## Supplementary material

The following pages contain: A more details about the three datasets used in this work, B more details about the architecture of the predicate classifier and hyperparameter optimization, C additional relationship detection experiments and ablation studies, and D additional qualitative results from the three datasets.

# A Datasets

Table 5: Comparison of the datasets used in this work

| | Number of images | | Vocabulary size | | | Unique triplets | |
|---|---|---|---|---|---|---|---|
| | Train | Test | Subject | Predicate | Object | Train | Test |
| HICO-DET [6] | 38118 | 9658 | 1 | 117 | 80 | 600 | 600 |
| VRD [24] | 4006 | 1001 | 100 | 70 | 100 | 6672 | 2741 |
| UnRel [30] | - | 1071 | 100 | 70 | 100 | - | 76 |

### A.1 HICO-DET

The Humans Interacting with Common Objects dataset [7], in its detection version [6], is available at `http://www-personal.umich.edu/~ywchao/hico`. The subject of the relationships is always a *person*. The object vocabulary is the same as MS-COCO [23]. Its predicates indicate human-object interactions, e.g. *carry*. Some images from MS-COCO are also contained in HICO-DET, but the authors made sure that the test set of HICO-DET has no overlap with MS-COCO. We warn future users to ignore the EXIF rotation tags present on some of the images, in fact all bounding boxes are annotated w.r.t. the non-rotated images. See table 5 for a comparison of dataset and vocabulary size.
We use the pre-trained object detector made available through the `detectron2` implementation [43] of Faster R-CNN [36]. Since the object detector is an important part of visual relationship detection pipelines, we report object detection metrics obtained for this dataset in table 6.

person eat sandwich
person sit on bench

person drive motorcycle
person hold motorcycle

Fig. 6: Ground-truth triplet annotations from the HICO-DET dataset

### A.2 Visual Relationship Detection dataset

The Visual Relationship Detection Dataset (VRD) [24] is available at `https://cs.stanford.edu/people/ranjaykrishna/vrd`. Its images and annotations correspond to those in the Scene Graph dataset [18], but the vocabularies of objects and predicates have been carefully curated, e.g. figure 7 We warn future users to ignore the EXIF rotation tags present on some of the images, in fact all bounding boxes are annotated w.r.t. the non-rotated images. Also, we note that for some images the annotation file contains 0 objects and 0 relationships. See table 5 for a comparison of dataset and vocabulary size.

Since no pre-trained model is publicly available for this dataset, we fine-tune an object detector based on `detectron2` [43]. Object detection metrics are reported in table 6 for future reference.

person has cellphone
tree in the front of building

person read book
backpack next to person

Fig. 7: Ground-truth triplet annotations from the VRD dataset

### A.3 Unusual Relationships dataset

The Unusual Relationships dataset (UnRel) [30] is available at `https://www.di.ens.fr/willow/research/unrel`. It is meant as an evaluation-only dataset for rare and unusual relationships, e.g. figure 8. See table 5 for a comparison of dataset and vocabulary size.

Since it shares the same object and predicate vocabulary of VRD, we use the same object detector, of which we report object detection metrics in table 6

person hold plane

dog drive car

Fig. 8: Ground-truth triplet annotations the UnRel dataset

Table 6: Object detection metrics for the datasets used in this work

| | Mean Average Precision | | | | | |
|---|---|---|---|---|---|---|
| | IoU@[0.5:0.95] | IoU@0.5 | IoU@0.75 | small | medium | large |
| HICO-DET [6] | 20.2 | 34.1 | 20.8 | 2.3 | 11.5 | 29.7 |
| VRD [24] | 21.2 | 35.3 | 22.6 | 4.9 | 14.3 | 25.0 |
| UnRel [30] | 21.0 | 35.3 | 22.6 | 4.9 | 14.3 | 25.0 |

| | Mean Average Recall | | | | | |
|---|---|---|---|---|---|---|
| | top-1 | top-10 | top-100 | small | medium | large |
| HICO-DET [6] | 30.3 | 39.3 | 40.2 | 11.6 | 29.2 | 48.6 |
| VRD [24] | 34.0 | 45.0 | 45.1 | 14.9 | 33.2 | 48.3 |
| UnRel [30] | 34.0 | 45.0 | 45.1 | 14.9 | 33.2 | 48.3 |

# B Architecture and hyperparameters

## B.1 Introduction to GNNs

In our work, an image is first represented as a fully-connected graph of objects and then processed through a graph neural network to predict predicates. Specifically, we use a message-passing implementation of graph convolution. At the input, each node $i$ is associated to a feature vector $\boldsymbol{v}_i$. Similarly, each edge $i \to j$ is associated to a feature vector $\boldsymbol{e}_{i,j}$. A global bias term $\boldsymbol{u}$ can be used to represent information that is not localized to any specific node/edge of the graph. With this graph representation, one layer of message passing performs the following updates.

1. For every edge $i \to j$, the edge vector is updated using a function $f^e$ that takes as input the adjacent nodes $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$, the edge itself $\boldsymbol{e}_{i,j}$ and the global attribute $\boldsymbol{u}$:
$$\boldsymbol{e}'_{i,j} = f^e\left(\boldsymbol{v}_i, \boldsymbol{v}_j, \boldsymbol{e}_{i,j}, \boldsymbol{u}\right)$$
2. For every node $i$, features from incident edges $\{\, \boldsymbol{e}'_{j,i} \,\}$ are aggregated using a pooling function $\text{agg}^{e\to v}$:
$$\bar{\boldsymbol{e}}'_i = \text{agg}^{e\to v}\left\{\, \boldsymbol{e}'_{j,i} \,\right\}$$
3. For every node $i$, the node feature vector is updated using a function $f^v$ that takes as input the aggregated incident edges $\bar{\boldsymbol{e}}'_i$, the node itself $\boldsymbol{v}_i$ and the global attribute $\boldsymbol{u}$:
$$\boldsymbol{v}'_i = f^v\left(\bar{\boldsymbol{e}}'_i, \boldsymbol{v}_i, \boldsymbol{u}\right)$$
4. All edges are aggregated using a pooling function $\text{agg}^{e\to u}$:
$$\bar{\boldsymbol{e}}' = \text{agg}^{e\to u}\left\{\, \boldsymbol{e}'_{i,j} \,\right\}$$

5. All nodes are aggregated using a pooling function $\text{agg}^{v \to u}$:

$$\bar{\boldsymbol{v}}' = \text{agg}^{v \to u} \{ \boldsymbol{v}_i' \}$$

6. The global feature vector is updated using a function $f^u$ of the aggregated edges $\bar{\boldsymbol{e}}'$, of the aggregated nodes $\bar{\boldsymbol{v}}'$ and of the global attribute $\boldsymbol{u}$:

$$\boldsymbol{u}' = f^u (\bar{\boldsymbol{e}}', \bar{\boldsymbol{v}}', \boldsymbol{u})$$

These convolutional layers can be stacked to increase the receptive fields of a node. However, in this work, we used a single layer to focus on pairwise relationships. Furthermore, we did not use a global attribute $\boldsymbol{u}$, which could encode for example context and background.

### B.2 Predicate classifier

For the predicate classifier we optimize the hyperparameters reported in table 7. Rather than performing a grid-search over the whole space, we perform a "guided" search: we iteratively perform parallel runs and only keep the best-performing combinations of parameters. This process of trial and elimination allows us to quickly prune unpromising regions of the search space.

Table 7: Hyperparameter space of the predicate classifier

| Parameter | Choices | Final value |
|---|---|---|
| **Optimizer** | | |
| Learning rate | $10^{-2}, 10^{-3}, 10^{-4}$ | $10^{-3}$ |
| Weight decay | $10^{-3}, 10^{-5}, 0$ | $10^{-5}$ |
| Max epochs | 35 | 18 |
| **Model** | | |
| Linear layers | 1, 2 | 1 |
| Linear features | 256, 512, 1024 | 1024 |
| Convolutional layers | 1, 2 | 2 |
| Convolutional kernels | 256, 512 | 256 |
| Pooling function | add, max, mean | max |
| Bias in $f_p$ | yes, no | yes |

The best set of hyperparameters is chosen to maximize `recall@5` over a held-out validation set (15% of training data). The train/val split is made at random for every training run. Random seeds are fixed at the beginning of each run and recorded for reproducibility. Note that `recall@5` refers to the image-level predicate predictions, and relationship detection metrics are not involved in the optimization of the predicate classifier.

On the test set of HICO-DET, relative to predicate classification only, these parameters achieve a mAP of 0.44, `recall@5` of 0.90 and `recall@10` of 0.96.

### B.3 ResNeXt baseline and Grad-CAM

We finetune a ResNeXt-50 [44] for predicate classification on the Visual Relationship Detection dataset. All parameters are initialized from an ImageNet [38] pretraining, except the final classification layer that is adapted to output 70-dimensional vector of predicate predictions and is initialized from a Normal distribution. Given an input image $\mathcal{I} \in [0,1]^{3\times H\times W}$, the convolutional architecture can be summarized as:

$$\boldsymbol{h} = \textsc{ResNeXt}(\mathcal{I}) \in \mathbb{R}^{2048\times\tilde{H}\times\tilde{W}} \qquad \text{backbone} \tag{17}$$

$$z_c = \frac{1}{\tilde{H}\tilde{W}} \sum_{i=1}^{\tilde{H}} \sum_{j=1}^{\tilde{W}} h_{c,i,j} \quad \forall c = 1,\ldots,2048 \qquad \text{global average pooling} \tag{18}$$

$$\boldsymbol{y} = \text{softmax}(\boldsymbol{W}\boldsymbol{z} + \boldsymbol{b}) \in [0,1]^K \qquad \text{classification} \tag{19}$$

where $\tilde{H}$ and $\tilde{W}$ represent the height and width of the feature volume extracted by the backbone before global average pooling.

We use Adam optimizer [19] to minimize the same loss of the GNN-based predicate classifier described in the main text. The learning rate is set to $10^{-3}$ for the classification layer and to $10^{-4}$ for the rest of the network.

We optimize only the number of epochs and whether the final layer should include a bias term or not. Based on performances on the validation set, the best hyperparameters are training for 6 epochs and including the bias. The final CNN-based model achieves similar `recall@5` as the GNN-based classifier on the test set for predicate classification.

Grad-CAM heatmaps as in figure 3 are produced by computing:

$$\alpha_c^k = \frac{1}{\tilde{H}\tilde{W}} \sum_{i=1}^{\tilde{H}} \sum_{j=1}^{\tilde{W}} \frac{\partial y_k}{\partial h_{c,i,j}} \quad \forall c = 1,\ldots,2048 \tag{20}$$

$$s_{i,j} = \textsc{ReLU}\left(\sum_{c=1}^{2048} \alpha_c^k h_{c,i,j}\right) \forall i = 1,\ldots,\tilde{H}; j = 1,\ldots,\tilde{W}. \tag{21}$$

Then the 2D vector $\boldsymbol{s}$ is upsampled to the $H \times W$ size of the input image, and its values are normalized to the range $[0,1]$.

### B.4 Training and inference

The graph neural network described in section 3.2 is trained to classify the predicates present in an image from image-level annotations.

**Algorithm 1:** Training Algorithm

**Input:** Pretrained object detector (`detectron2`),
Dataset of images with image-level predicate annotations.
**repeat**
Extract objects from image $\mathcal{I}$
Build a fully-connected image graph $\mathcal{G}$ using features from eq. 14, 15
Apply the predicate classifier to $\mathcal{G}$
Compute the predicate classification loss $\mathcal{L}$ (equation 10)
Minimize $\mathcal{L}$ using Adam optimizer
**until** *convergence*
**Output:** Trained predicate classifier

Once trained, the predicate classifier can be used for relationship detection. Specifically, each pred prediction is attributed to pairs of objects in the input by means of explanation, thus retrieving the full $\langle \text{subj}, \text{pred}, \text{obj} \rangle$ triplet.

**Algorithm 2:** Explanation-based Relationship Detection Algorithm

**Input:** Pretrained object detector (`detectron2`),
Trained predicate classifier,
Image of interest $\mathcal{I}$.
**if** *Predicate Detection* **then**
Extract ground-truth objects from image $\mathcal{I}$
**else if** *Phrase Detection* $\vee$ *Relationship Detection* **then**
Detect objects in $\mathcal{I}$ using the object detector
**end**
Build a fully-connected scene graph $\mathcal{G}$ using features from eq. 14, 15
Apply the predicate classifier to $\mathcal{G}$
Visual relations $\mathcal{R} \leftarrow \emptyset$
**for** *pred* $\in \{ N \textit{ top-scoring predicates} \}$ **do**
`/* Predicate predictions are explained in terms of relevant pairs of objects in the image graph` $\mathcal{G}$ `*/`
Compute node and edge relevances using eq. 11, 12
Score each $\langle \text{subj}, \text{obj} \rangle$ pair using equation 13
Multiply the score by the object detection scores of subj and obj
Multiply the score by the classification score of pred
Multiply the score by the relationship prior (equation 5)
Store high-scoring triplets $\langle \text{subj}, \text{pred}, \text{obj} \rangle$ in $\mathcal{R}$
**end**
**Output:** $K$ top-scoring visual relations from $\mathcal{R}$

# C Additional experiments

## C.1 Pooling function

As explained in appendix B, the pooling function for equation 9 is selected according to predicate classification performances (figure 9) on a 15% split of the training set. Figure 9 shows `recall@5` for *sum*, *max*, and *mean* pooling over 10 runs on the VRD dataset. Due to higher recall on the validation set, *max* pooling is selected and used for all results reported in the main text. We notice, however, that *mean* pooling performs closely to *max*.

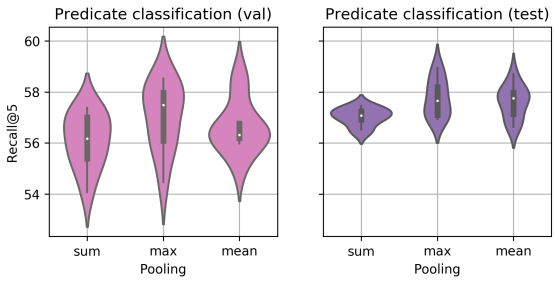


Fig. 9: `Recall@5` for predicate classification on VRD using different pooling functions. Validation set (15% of training) on the left, and test set on the right

To further test the role of pooling, we evaluated relationship detection metrics for several predicate classifiers trained using *sum*, *max*, and *mean* pooling. Figure 10 shows that *mean* pooling outperforms the other two, despite performing slightly worse for predicate classification.

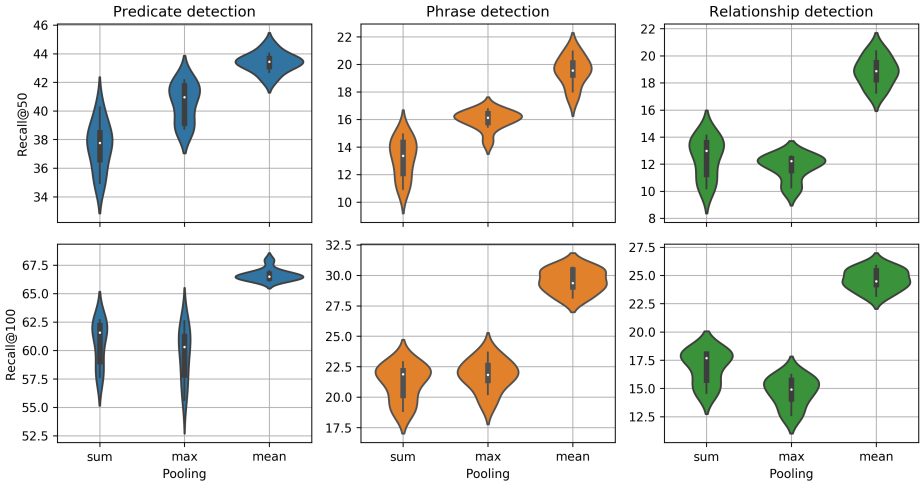


Fig. 10: `Recall@50` and `@100` for relationship detection on VRD using different pooling functions. *mean* pooling outperforms the other two, despite performing slightly worse for predicate classification

### C.2 Number of explained predicates

Given an image, the GNN classifier outputs a distribution of binary probabilities over the predicates contained in the image. To recover $\langle \text{subj}, \text{pred}, \text{obj} \rangle$ triplets, we consider the top $N$ predicates and *explain* them one at the time w.r.t. the input image graph. Therefore, the choice of $N$ influences the diversity of predicates contained in the detected relationships, e.g. if we only explained the top scoring predicate we could still recover many triplets but they would all share the same predicate.

For the main results, we set $N = 10$, assuming that in natural images the chance of having more than ten different predicates depicted in the same picture would be rather low. To further prove this point, in figure 11 we plot `recall@50` and `recall@100` for various choices of $N$ on the VRD dataset. Notably, considering very few predicates in the explanation phase, gives poor results on all three relationship detection scenarios. However, increasing $N$ to consider more predicate categories yields diminishing returns after $N = 20$.

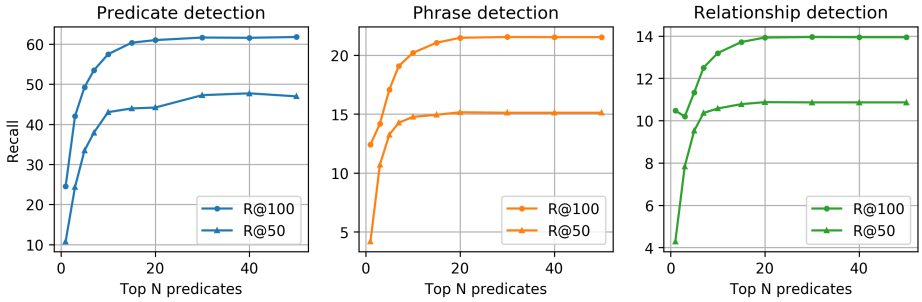


Fig. 11: Recall at 50 (R@50) and at 100 (R@100) on the VRD dataset as the number $N$ of predicates considered for explanation increases from 1 to 50. Diminishing returns are observed, with an elbow at approximately $N = 10$

### C.3 Relationship prior

As explained in section 3.4, a weakly-supervised method trained only on predicate labels is not able to learn the directionality of the relations, e.g. it could not distinguish *car on street* from *street on car*. Therefore, we introduced a simple relationship prior based on the frequency of relationships in a small subset of training data. Specifically, we compute:

$$\text{freq}(c_i, c_j | k) = \frac{|\{ \langle c_{\text{subj}}, b_{\text{subj}}, k, c_{\text{obj}}, b_{\text{subj}} \rangle \mid c_{\text{subj}} = c_i, c_{\text{obj}} = c_j, k_{\text{pred}} = k \}|}{|\{ \langle c_{\text{subj}}, b_{\text{subj}}, k, c_{\text{obj}}, b_{\text{subj}} \rangle \mid k_{\text{pred}} = k \}|}$$

In the main experiments, we use a 15% split of the training set to compute this prior, assuming that it would be enough to disambiguate most cases. In figure 12, we show how `recall@50` and `recall@100` on the VRD dataset change

according to the percentage of training triplets used to compute the relationship prior. For each percentage value, we plot the mean recall over 5 random subsets and shade the area corresponding to two standard deviations. We observe that all percentages obtain approximately the same recall, except for 0% that corresponds to a uniform prior. Notably, the randomness introduced when choosing a subset of the given percentage of training data has little effect on the result.

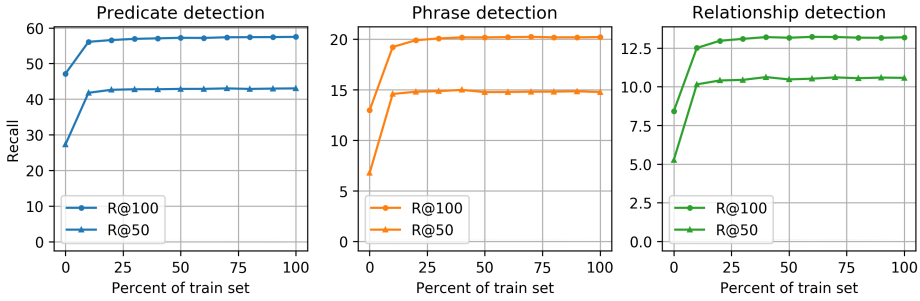


Fig. 12: Recall at 50 (R@50) and at 100 (R@100) on the VRD dataset as the percentage of training data used to compute the relationship prior increases. At each percentage, we run 5 evaluations and plot mean and two standard deviations. Each evaluation uses a different random subset to compute the prior. All percentages obtain approximately the same recall, except for 0% that corresponds to a uniform prior

## D Additional results

In this section we report additional qualitative results to evaluate the relationship detection pipeline. We include examples of: correct relationship detections, correct detections missing from the ground truth, incorrect detections due to object misclassification, and incorrect detection due to subject-object inversion, wrong choice of pair, or wrong predicate. All images in figures 13, 14 and 15 are chosen at random from the test sets of each dataset. Then, representative examples are chosen from the top 10 detections of each image (top 25 for UnRel).

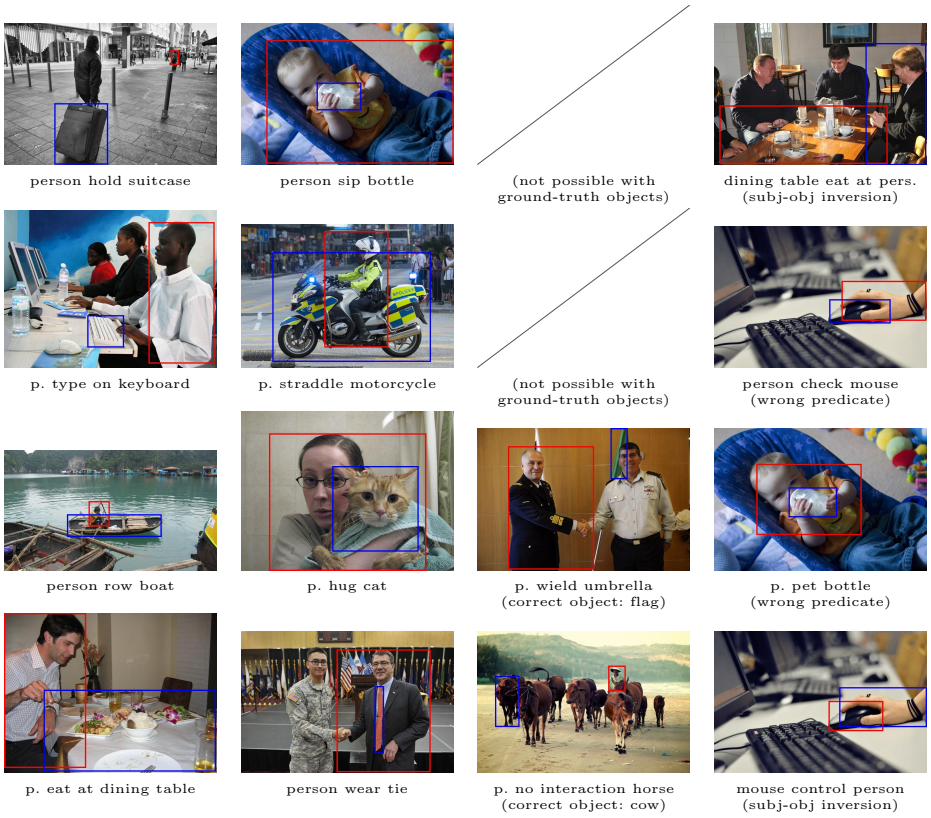


Fig. 13: **Additional detections on HICO-DET.** Top two rows use ground-truth objects, bottom two rows use Faster R-CNN objects. Subjects are framed in red, objects in blue. Left to right: correct relationship detection, correct but missing ground-truth, incorrect due to object misdetection, incorrect detection. Images are chosen at random from the test set, all depicted triplets are selected from the top 10 detections


Fig. 14: **Additional detections on VRD.** Odd rows use ground-truth objects, even rows use Faster R-CNN objects. Subjects are framed in red, objects in blue. Left to right: correct relationship detection, correct but missing ground-truth, incorrect due to object misdetection, incorrect detection. Images are chosen at random from the test set, all depicted triplets are selected from the top 10 detections of an image

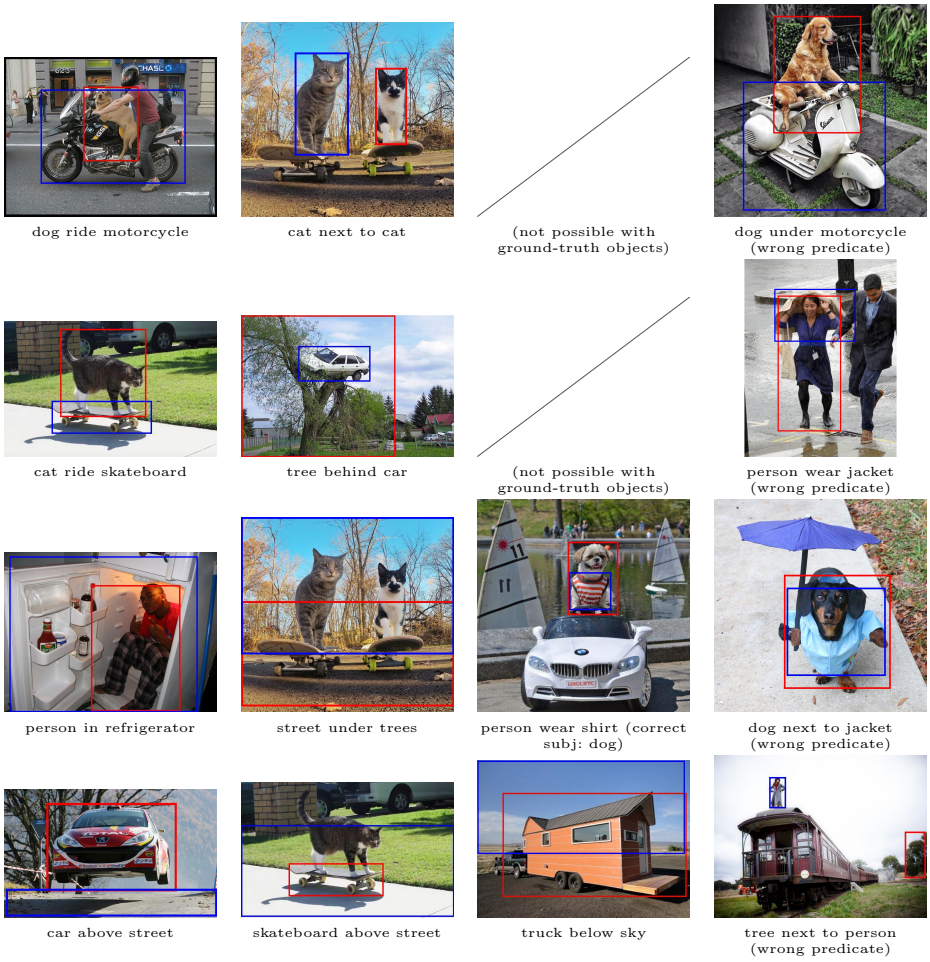


Fig. 15: **Additional detections on UnRel.** Top two rows use ground-truth objects, bottom two rows use Faster R-CNN objects. Subjects are framed in red, objects in blue. Left to right: correct relationship detection, correct but missing ground-truth, incorrect due to object misdetection, incorrect detection. Images are chosen at random from the test set, all depicted triplets are selected from the top 25 detections