Attention-based Context Aware Reasoning for Situation Recognition Supplementary Material

Thilini Cooray, Ngai-Man Cheung, Wei Lu
Singapore University of Technology and Design (SUTD)
thilini_cooray@mymail.sutd.edu.sg, {ngaiman_cheung, luwei}@sutd.edu.sg

1. Related work (Minor importance)

The most related work has been discussed in the main paper Section 2. Here we discuss some additional related work with minor importance to our work.

Visual Semantic Role Labelling (VSRL) is a task which goes hand in hand with Situation Recognition. VSRL was first introduced by Gupta and Malik [2] where they had annotated MSCOCO [6] dataset for 26 actions and localized 3 roles; agent, object and instrument. Motivation behind VSRL was to get a thorough understanding about actions by being able to reason on objects and people related to it. This vision was brought forward by Yatskar et al. [15] by introducing a more comprehensive dataset consisting of 504 actions and 190 unique semantic roles extracted from FrameNet [1].

Grounding semantic roles in images is another related task. Yang et al. [13] have proposed a method and a dataset to ground objects in video clips referring to semantic roles in a given sentence. This differs from our task as we do not use sentences to first find out verb and labels for its semantic roles. Silberer and Pinkal [10] have introduced another semantic role grounding dataset based on *Flickr30k Entities* [9] dataset. Their task is to select the most relevant region for each semantic role of the given frame from a set of image regions. Our proposed approach can be applied to this task as well. However we are unable to evaluate as the dataset is not released to the public.

2. Reasoning enhanced verb prediction - complete details

We have provided some description for reasoning enhanced *verb prediction* in Section 5.2 in the main paper. Here we provide complete details. We remark that our main contribution is *role prediction* (or frame recognition (FR)), which details have been discussed in Sections 3 and 4 of the main paper.

In this section we explain complete details about the TDA based verb prediction model which we report results in Table 1 row 2 under the title *Predicted Query Model* in our main paper.

2.1. Role label prediction component of the Verb model

TDA model expects a query condition $\mathbf{q}$ as we discussed in Equation 3 of the main paper in order to condition the image and find the relevant answer for the query. For verb prediction, we decided to form our query based on labels of the two most frequent roles in *imSitu* dataset; *Agent* and *Place*.

We decided to use a modified version of TDA based FR model to predict *Agent* and *Place* role labels, which are going to be input for our verb model. The reason we had to modify the original FR model is because, when the query is encoded in Equation 2 of the main paper, we use concatenation operation between verb embedding and role name embedding. However when we want to use this FR model to provide us label predictions of *Agent* and *Place* roles to input for verb prediction, FR model should have the capability to process queries which do not have verb embedding. Since concatenation operation cannot support this requirement, we replaced the original TDA FR model's concatenation in query encoding (Equation 2 - main paper) to an addition operation as follows:

$$\mathbf{q} = f_{role\_q}(\mathbf{w}_v + \mathbf{w}_r) \tag{1}$$

During model training, we use Equation 1 and we use the following after removing the verb embedding during inference.

$$\mathbf{q} = f_{role\_q}(\mathbf{w}_r) \tag{2}$$

First we train this model separately and this pretrained FR model is used to predict *Agent* and *Place* labels during verb model training.

2.2. TDA for verb prediction

We use the original TDA model only with slight modifications for verb prediction task. First we modify the query encoding step (Equation 2 - main paper) to our new query condition as follows.

$$\textbf{q\_verb} = f_{verb\_q}([\mathbf{w}_{agent}, \mathbf{w}_{place}]) \tag{3}$$

$[\cdot]$ is used to denote the concatenation. Embedding vectors for *Agent* and *Place* role labels are $\mathbf{w}_{agent}, \mathbf{w}_{place} \in \mathbb{R}^{d\_wemb}$. These embeddings are randomly initialized and learnt during model training. $agent \in \{1, \ldots, |N|\}$ is the *Agent* id and $place \in \{1, \ldots, |N|\}$ is the *Place* id.

Then we use this **q_verb** as our query and continue with the original TDA model from Equation 3-5 from the main paper. We observed from our experiments that normalization layers did not help for verb prediction like they did with FR. Therefore we did not execute Equation 6 from main paper for verb prediction model. As seen in the caption of the Table 1 of the main paper, when gold *Agent* and *Place* role labels are used, verb prediction accuracy is very high. But when we replace them with predicted labels, performance drops significantly due to the prediction errors of the FR model. Therefore we understood that completely relying the model on predicted role labels is unwise as they are known to be incorrect sometimes.

As a remedy to this, we decided to use hidden representations of *Agent* and *Place* role labels ($\mathbf{h}_{agent}, \mathbf{h}_{place} \in \mathbf{h}$) generated in Equation 6 of the main paper in this reasoning process as well. We use them as a soft query and fuse them to the original image encoding to provide more contextual information to support verb prediction. We generate this contextual information as follows

$$\textbf{soft\_query} = \mathbf{h}_{agent} + \mathbf{h}_{place} \tag{4}$$

$$\mathbf{E}_{flat} = \text{AvgPool}(\mathbf{E}_I)\mathbf{W}_{flt\_img} \tag{5}$$

$$\textbf{context} = \mathbf{E}_{flat} \circ \textbf{soft\_query} \tag{6}$$

where $\mathbf{W}_{flt\_img} \in \mathbb{R}^{d\_img \times d\_hidden}$. Then we add this with the output from Equation 5 of the main paper to obtain our final hidden representation ($\hat{\mathbf{h}}$) that will be sent to the classifier to get the verb prediction. This is the model we reported results in our main paper in Table 1 row 2.

$$\hat{\mathbf{h}} = \mathbf{h_u} + \textbf{context} \tag{7}$$

$$p_{verb} = \text{SoftMax}(f_{v\_classifier}(\hat{\mathbf{h}})) \tag{8}$$

We train this model with cross entropy loss as follows.

$$Loss = -\sum_{i=1}^{|V|} y_i \log(p_{verb}(i)) \tag{9}$$

$y_i \in \{0, 1\}$ is the ground truth encoding of the verb i. Also note that $p_{verb}(i) \in p_{verb}$. We get the final verb prediction (v_{pred}) as follows:

$$v_{pred} = \arg\max_i p_{verb}(i) \tag{10}$$

3. Complete Implementation Details

We implemented our models using PyTorch [8] framework. We use VGG-16 [11] as our backbone CNN architecture to encode images following all existing work [15, 14, 7, 5] for SR. We extract grid features of size $7 \times 7 \times 512$ after the final max pooling layer as our regions where $N_e = 49$. Final dimensions of different components of our models as follows: $d\_img = 512$, $d\_q = 1024$, $d\_wemb = 300$, $d\_hidden = 1024$, $B = 4$ and $\beta = 10$. Bias terms in all our equations have not been included for the simplicity of notation. Dropout value used in Equation 6 of the main paper is 0.1 and 0.5 is the dropout value used in both FR and verb classifiers. We trained our FR models to predict the most frequent 2000 nouns following [5] as it covers more than 95% of samples. $max\_role\_count = 6$ is the number of maximum roles exist in a frame of *imSitu* dataset. We train the model end-to-end including the CNN where CNN is finetuned with initial learning rate 5×10^{-5} and the rest of the model with learning rate of 1×10^{-3} using AdaMax [4] optimizer and Exponential scheduler. We used mini-batch size of 64 and obtained the best model by early stopping using development set performance. For CAQ and CAI models, we use the pre-trained TDA model to provide the hidden representations for context generation in Equation 9-10 and the rest of the model is trained end to end.

All our non-linear layers ($f_q, f_a, f_{pq}, f_{pi}, f_{cq}$ and f_{recon} from main paper and f_{role_q} and f_{verb_q} from this) are using gated hyperbolic tangent activation [12] as follows:

$$\tilde{\mathbf{y}} = tanh(\mathbf{x}\text{WeightNorm}(\mathbf{W})) \tag{11}$$

$$\mathbf{g} = \sigma(\mathbf{x}\text{WeightNorm}(\mathbf{W}')) \tag{12}$$

$$\mathbf{y} = \tilde{\mathbf{y}} \circ \mathbf{g} \tag{13}$$

Layer Name	f_input	f_output
f_q	$d\_wemb \times 2$	d_q
f_a	$d\_img + d\_q$	d_hidden
f_{pq}	d_q	d_hidden
f_{pi}	d_img	d_hidden
f_{cq}	$d\_hidden + d\_wemb \times 2$	d_q
f_{recon}	$d\_hidden \times max\_role\_count$	d_hidden
f_{role_q}	d_wemb	d_q
f_{verb_q}	$d\_wemb \times 2$	d_q

Table 1: Dimensions of all used non-linear layers.

Both $\mathbf{W}$ and $\mathbf{W}'$ have same dimensions $\mathbb{R}^{f\_input \times f\_output}$. Sigmoid function works as a gate to control each element of the input vector $\mathbf{x} \in \mathbb{R}^{f\_input}$

and output $\mathbf{y} \in \mathbb{R}^{f\_output}$. Table 1 includes exact dimensions we used for each non-linear layer.

$f_{classifier}$ and $f_{v\_classifier}$ use a Multilayer Perceptron network shown in the box below. For both $f_{classifier}$ and $f_{v\_classifier}$, $\mathbf{W}_{cls\_in} \in \mathbb{R}^{d\_hidden \times (d\_hidden \times 2)}$. For the last layer, the output size differs for each of them as the number of classes are N and V respectively. Therefore $\mathbf{W}^{roles}_{cls\_out} \in \mathbb{R}^{(d\_hidden \times 2) \times N}$ and $\mathbf{W}^{verb}_{cls\_out} \in \mathbb{R}^{(d\_hidden \times 2) \times V}$.

WeightNorm($\mathbf{W}_{cls\_in}$)
$ReLU$
Dropout
WeightNorm($\mathbf{W}_{cls\_out}$)

$f_{flatten\_img}$ contains a linear layer $\mathbf{W} \in \mathbb{R}^{(d\_img \times N_e) \times d\_hidden}$ followed by a BatchNorm [3] layer.

4. Ablation Studies

4.1. Impact of normalization layer

In our proposed models we use normalization indicated in Equation 6 of the main paper. We use normalization to reduce the magnitude of values output by element-wise multiplication operation mentioned in Equation 5 of the main paper. Element-wise multiplication can cause the magnitude of outputs vary drastically and this might cause the model to converge to local minimum [16]. Yu et al. [16] have used normalization to address that and in this section we empirically evaluate its impact on role and verb predictions.

TDA role model		
Feature	Value	Value-all
With normalization	72.96	37.60
Without normalization	72.47	36.85
CAQ role model		
Feature	Value	Value-all
With normalization	73.62	38.71
Without normalization	73.19	37.93
TDA verb model		
Feature	Top 1 Verb	Top 5 Verb
With normalization	34.29	61.92
Without normalization	34.83	61.87

Table 2: Impact of normalization on role and verb models.

From Table 2 we can observe that both TDA and CAQ FR models have achieved a 1% improvement when normalization layer is used. However for the verb model, the performance have reduced slightly with normalization. The reason for this could be, there are multiple queries reasoned against a single image encoding (all roles of the current frame) in the FR model. This can cause the magnitude of each of the neurons of output vector $\mathbf{h_u}$ to vary quite a lot for queries of the same image. Normalization has contributed to reduce this variation for some extend. On the other hand, verb model only has one query per image and our queries are very simple compared to natural language sentences. Hence when the normalization is added, it seems to have caused the verb model to underfit a bit and lose its performance.

4.2. Impact of context information on TDA verb model

Table 3 contains results on the improvement we obtained by adding soft-query based context information to TDA verb model. Model *TDA verb with context* is our final verb model which we report results in our main paper under Table 1 row 3. The difference it has with *TDA verb* model is that in *TDA verb*, we do not incorporate soft query based context for the reasoning process. We do not execute Equation 4 - 7 in *TDA verb* model and directly send $\mathbf{h_u}$ (output from Equation 5 of the main paper) to Equation 8 for verb predictions.

Model	Top 1 Verb	Top 5 Verb
TDA verb	34.83	61.87
TDA verb with context	35.70	62.19

Table 3: Performance comparison of soft-query based context incorporation to verb model.

The reason for this performance improvement is that when hidden representations of *Agent* and *Place* are used, it contains information about multiple potential role labels. Therefore even the final role label prediction was wrong causing our query $\mathbf{q\_verb}$ to be misleading, these hidden representations can contribute to correct it by incorporating secondary information which can provide clues on correct labels.

4.3. Performance of CAQ without attention

Table 4 compares the impact of attention based context generation on CAQ against TDA and CAQ which context generated without using attention (we call it CAQ without attention). CAQ without attention model does not execute Equation 9-11 in the main paper. It just sums up hidden representations of all neighbour roles together. CAQ without attention can improve TDA, which does not use any context adaptation. But it cannot surpass final CAQ (which uses attention) as the impact from each neighbour role to the current role differs from role to role. This can be qualitatively observed in the role dependency matrices in Figure 1.

Model	Value	Value-all
TDA	72.96	37.60
CAQ	73.62	38.71
CAQ without attention	73.54	38.32

Table 4: Performance comparison of CAQ for role prediction with and without attention against TDA.

4.4. Computational Efficiency

We compared the computational efficiency (Table 5) of proposed TDA and CAQ against Gated-GNN based SR model (GGNN) [5]. GGNN has the highest parameter count as it uses penultimate layer output from VGG-16 for image encoding while we use grid region features after last max-pooling layer. Although the non-CNN parameter count of GGNN is low, since GGNN is an iterative method, its computation time is high. TDA converged faster than role inter-dependency modeling approaches (GGNN and CAQ). However average running time of all models are of few seconds difference in our cluster of 1 GeForce GTX TITAN X and 1 GeForce GTX 1080 Ti.

Model	No of Total Trainable parameters	No of non-CNN parameters	Avg Training Time	Avg Evaluation Time
GGNN	148574225	10109905	15.72h	114.71s
TDA	28660369	13937233	9.87h	116.98s
CAQ	34955921	20232785	15.46h	120.18s

Table 5: Model efficiency comparison. Total trainable parameters include CNN and non-CNN parameters. CNN is image encoder, trained end-to-end with the rest of the models. “non-CNN” parameters : GNN - Parameters required for Gated-GNN, TDA - parameters used in Eq.2-7 in paper, CAQ - parameters required for all components in Section 4.1 in paper.

5. Qualitative Analysis

5.1. Comparison of CAQ to GNN with attention

Part of CAQ (Eq. 9-11 of main paper) has some similarity with GNN with attention (GNN-A): both techniques try to aggregate hidden representations of neighbour nodes (indicated as *context* in the paper) to be used for updating the current node representation. However, the entire CAQ differs from GNN-A significantly; in particular, the mechanism to update the current node (current semantic role) is very different.

GNN heavily relies on inter-node agreement for final node classification as it only uses the *context* for updating nodes. If a node displays a deviation from the normal pattern (Ex: for “Brushing”, in majority of samples where a person with a toothbrush, target is “teeth”. But for a few, the target is “finger nails”), GNN tends to suppress it by updating the original deviated node representation using its neighbourhood. The drawback of this updating mechanism is that the model tends to get highly biased to training set object co-occurrences. In contrast, CAQ uses the *context* only to update the query in its query based reasoning approach (Eq.12 in paper), avoiding directly updating node representation ($\mathbf{h}$ from Eq.6 in paper). Since the updated query has both the original question and context, we implicitly enable the model to decide which part of the query to focus when attending the image in Eq.3-4 in paper. Therefore CAQ has the ability to decide between independent query reasoning and inter-node agreement to mitigate the drawback in GNN.

5.2. Contribution of proposed contextualization module for improving inter-dependent query handling in SR

We have included further qualitative results extending from our main paper in this section. Page 6-8 contain additional examples of sample predictions, attention maps and role dependency matrices output from our TDA and CAQ models to showcase how CAQ has been able to improve its attention based reasoning and output accurate predictions compared to TDA, using contextual information. These samples are an extension to Figure 4 of the main paper. We discuss few examples in detail here to show how context helped to improve the performance in CAQ.

For verb “Weeding” (sample 1, Page 6), TDA has predicted the label for role *Tool* wrong. We can observe in the attention map, that TDA has highlighted the entire area around the man including his hands when finding the answer for role *Tool*, hence caused the prediction error. However TDA has correctly attended the image for roles *Place* and *Agent* and predicted them correctly. Next in CAQ, when the context is generated for role *Tool*, we can see from the role dependency matrix that *Agent* has the most impact for *Tool* and *Place* has second most. Thanks to the context provided by *Agent* and *Place*, we can observe that CAQ has been able to provide more focused attention to the “Hoe” and predict accurately.

Another example is verb “Nipping” in Page 7, sample 1. TDA has not been able to correctly locate which object from the image should be the answer for role *Item*. We can see this error from its attention map. However it has been able to correctly predict the *Agent* by locating the "Dog". From role dependency matrix we can see that, *Agent* provides the most information when generating context for *Item*. Using this context which provides the details that "Dog" is the *Agent*, CAQ has been able to correctly adjust its attention to clearly focus on the "Woman" and hence been able to correct its final prediction.

Final example we are discussing is verb "Fixing" (Page 7, sample 4). TDA has incorrectly predicted the used *Tool* as "Hand" due to the attention map which has highlighted the entire area of hands and wrench. However CAQ as been able to correct this error using the context generated from neighbour roles' information and focus directly to the *Tool*, "Wrench".

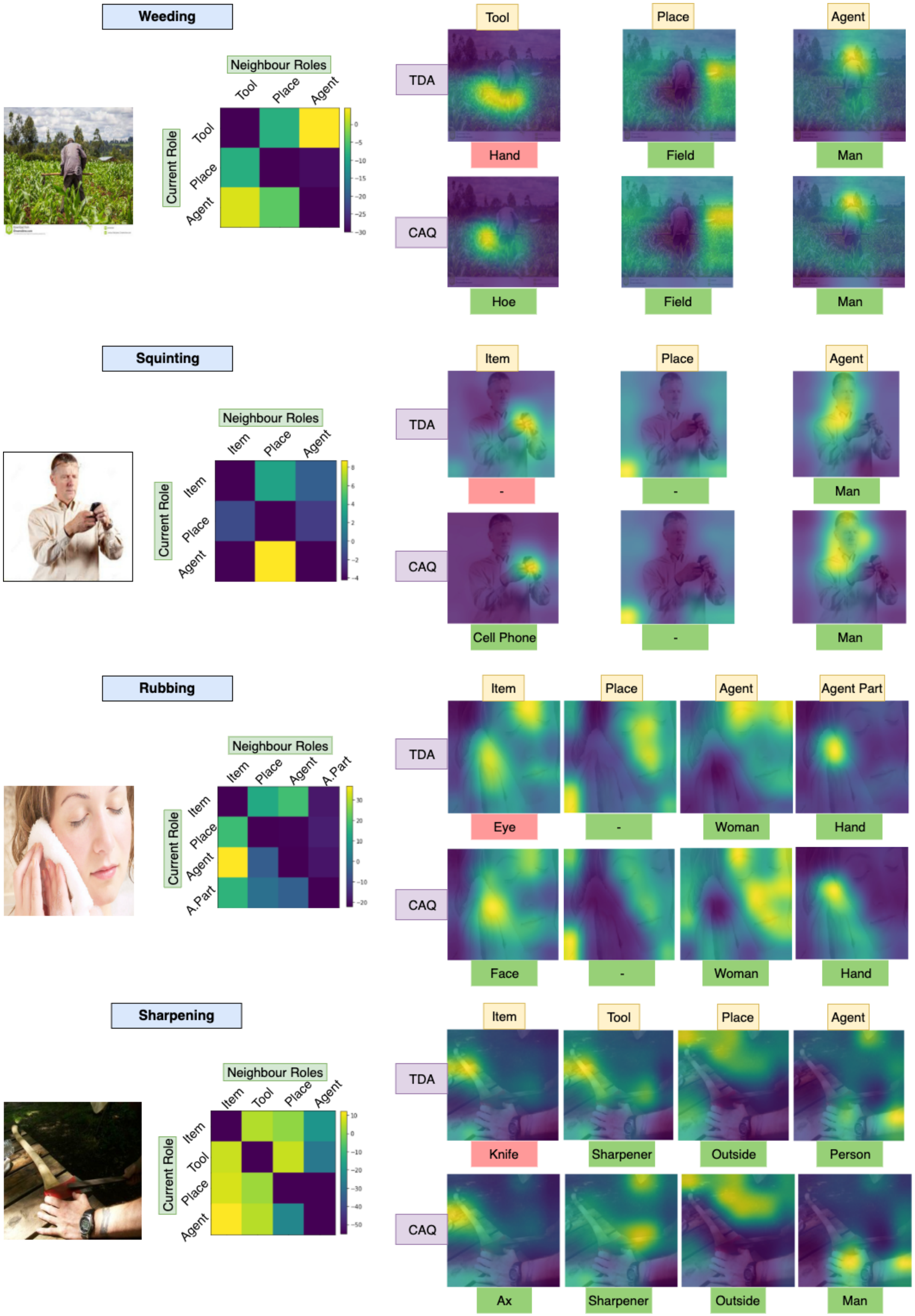
These samples emphasize how our proposed contextualization module contributes to improve inter-dependent query handing. The context generated using neighbour roles has proven to be able to guide the attention mechanism in CAQ to improve its answer localization more accurately than TDA, which only uses the verb_role embedding as guidance to generate attention.

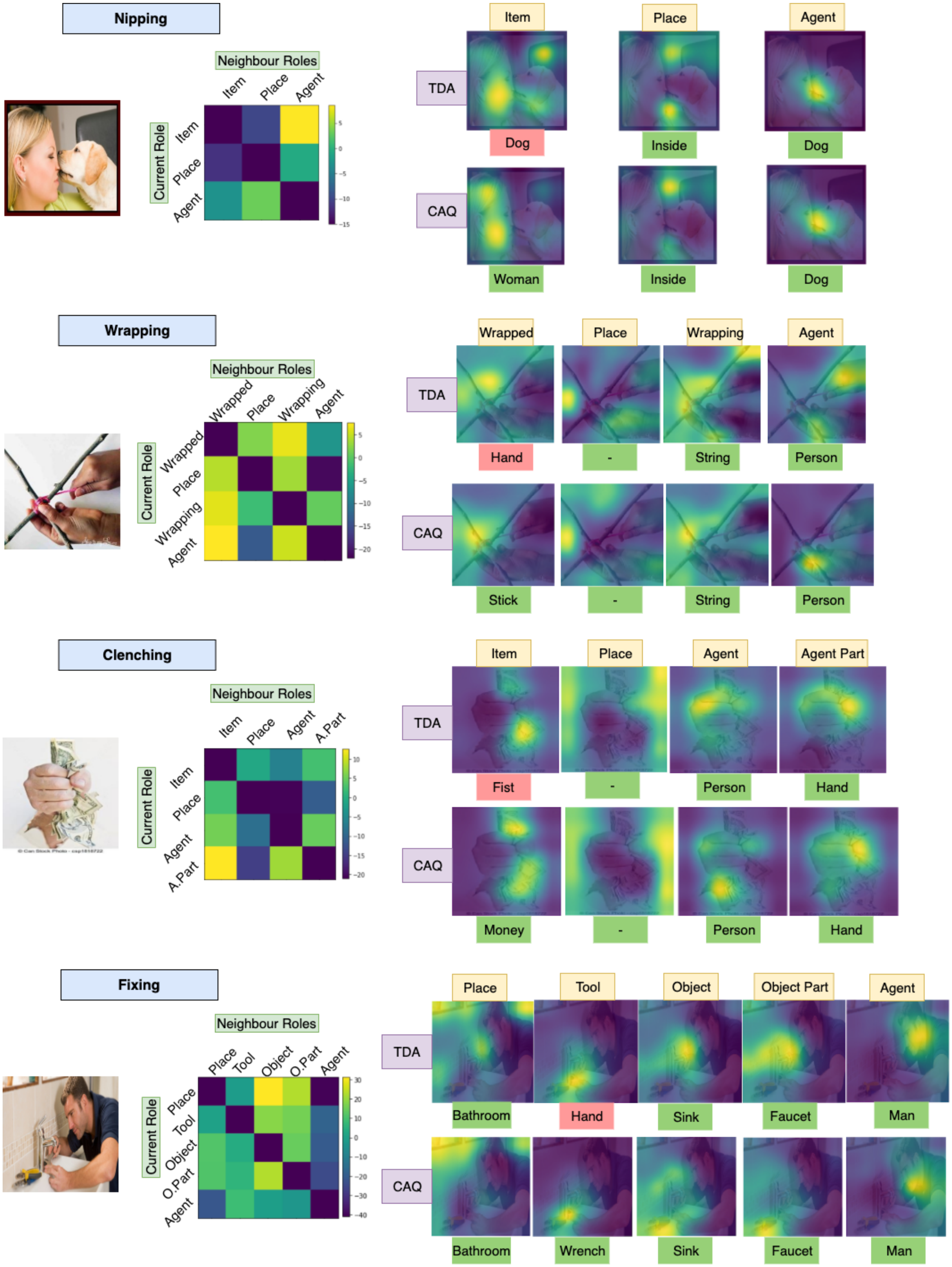
5.3. Role inter-dependency differences among verbs

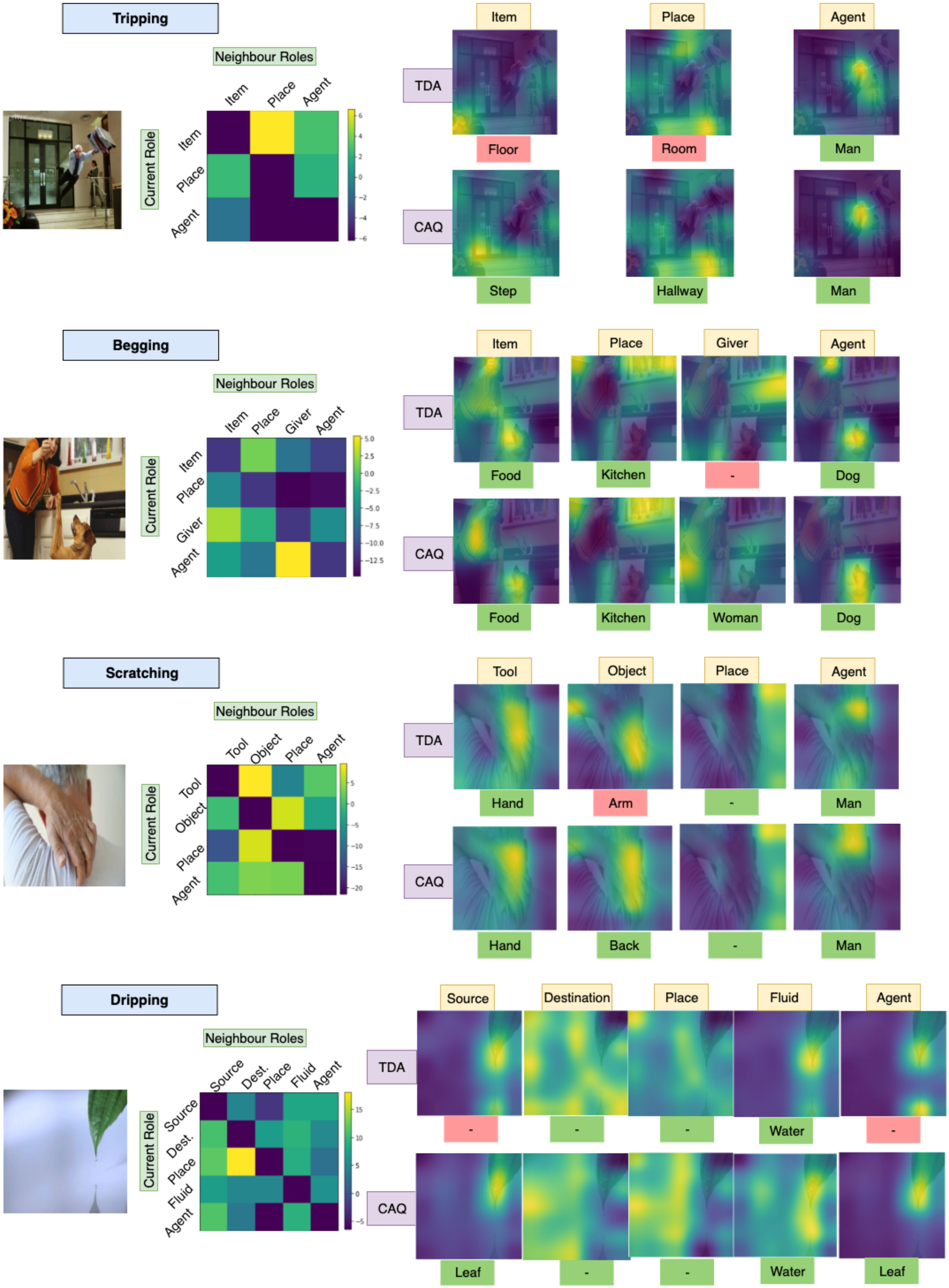
Next in Figure 1, we have role dependency matrices for several more verbs along with their sample images. These role dependency matrices are generated combining the unnormalized neighbour role weights generated for all roles in a single frame from Equation 9 of the main paper. Each row of our dependency matrix shows the current role, to which we generate the context using neighbour roles. Each column is for each neighbour role in the current frame. Each cell indicates the value which represents the impact a given neighbour role has on the current role. Diagonal elements have assigned the lowest value to indicate that current role does not consider itself when generating the context.

There are multiple subsets of roles that appear in many verbs. For an example the subset of roles {*Agent*, *Place*, *Item*} occur in "Opening", "Tugging" and "Carrying", while "Opening", "Applying", "Tuning" and "Spreading" share another subset of roles together which consists of {*Agent*, *Place*, *Tool*}. But do these roles get the same level of importance in every verb they appear? Do they even maintain the same correlation with their neighbour roles across the verbs they appear? This section is to highlight our observations on these matters according to the generated role dependency matrices by our proposed approach.

We observe based on our learnt role dependency matrices that, eventhough multiple verbs can have same subset of roles, the importance each role gets among its neighbours can vary based on each verb. For an example, eventhough *Item* being the role with the most impact for "Opening" and "Carrying", and *Agent* has the least impact for these verbs, *Agent* has the most impact for verb "Tugging" and *Item*'s impact is lesser. Role inter-dependency also shows a similar characteristic. For an example, *Place* is highly dependent on *Item* for "Opening". But when it comes to "Tugging" and "Carrying", *Place* has a relatively lesser dependency on *Item*.


Weeding
Neighbour Roles
Tool
Place
Agent
Current Role
Tool
Place
Agent
0
-5
-10
-15
-20
-25
-30
Tool
Place
Agent
TDA
Hand
Field
Man
CAQ
Hoe
Field
Man
Squinting
Neighbour Roles
Item
Place
Agent
Current Role
Item
Place
Agent
8
6
4
2
0
-2
-4
Item
Place
Agent
TDA
-
-
Man
CAQ
Cell Phone
-
Man
Rubbing
Neighbour Roles
Item
Place
Agent
A.Part
Current Role
Item
Place
Agent
A.Part
30
20
10
0
-10
-20
Item
Place
Agent
Agent Part
TDA
Eye
-
Woman
Hand
CAQ
Face
-
Woman
Hand
Sharpening
Neighbour Roles
Item
Tool
Place
Agent
Current Role
Item
Tool
Place
Agent
10
0
-10
-20
-30
-40
-50
Item
Tool
Place
Agent
TDA
Knife
Sharpener
Outside
Person
CAQ
Ax
Sharpener
Outside
Man


Nipping
Neighbour Roles
Item
Place
Agent
Current Role
Item
Place
Agent
5
0
−5
−10
−15
Item
Place
Agent
TDA
Dog
Inside
Dog
CAQ
Woman
Inside
Dog
Wrapping
Neighbour Roles
Wrapped
Place
Wrapping
Agent
Current Role
Wrapped
Place
Wrapping
Agent
5
0
−5
−10
−15
−20
Wrapped
Place
Wrapping
Agent
TDA
Hand
-
String
Person
CAQ
Stick
-
String
Person
Clenching
Neighbour Roles
Item
Place
Agent
A.Part
Current Role
Item
Place
Agent
A.Part
10
5
0
−5
−10
−15
−20
Item
Place
Agent
Agent Part
TDA
Fist
-
Person
Hand
CAQ
Money
-
Person
Hand
Fixing
Neighbour Roles
Place
Tool
Object
O.Part
Agent
Current Role
Place
Tool
Object
O.Part
Agent
30
20
10
0
−10
−20
−30
−40
Place
Tool
Object
Object Part
Agent
TDA
Bathroom
Hand
Sink
Faucet
Man
CAQ
Bathroom
Wrench
Sink
Faucet
Man



Tripping
Neighbour Roles
Item
Place
Agent
Current Role
Item
Place
Agent
6
4
2
0
−2
−4
−6
Item
Place
Agent
TDA
Floor
Room
Man
CAQ
Step
Hallway
Man
Begging
Neighbour Roles
Item
Place
Giver
Agent
Current Role
Item
Place
Giver
Agent
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
−12.5
Item
Place
Giver
Agent
TDA
Food
Kitchen
-
Dog
CAQ
Food
Kitchen
Woman
Dog
Scratching
Neighbour Roles
Tool
Object
Place
Agent
Current Role
Tool
Object
Place
Agent
5
0
−5
−10
−15
−20
Tool
Object
Place
Agent
TDA
Hand
Arm
-
Man
CAQ
Hand
Back
-
Man
Dripping
Neighbour Roles
Source
Dest.
Place
Fluid
Agent
Current Role
Source
Dest.
Place
Fluid
Agent
15
10
5
0
−5
Source
Destination
Place
Fluid
Agent
TDA
-
-
-
Water
-
CAQ
Leaf
-
-
Water
Leaf

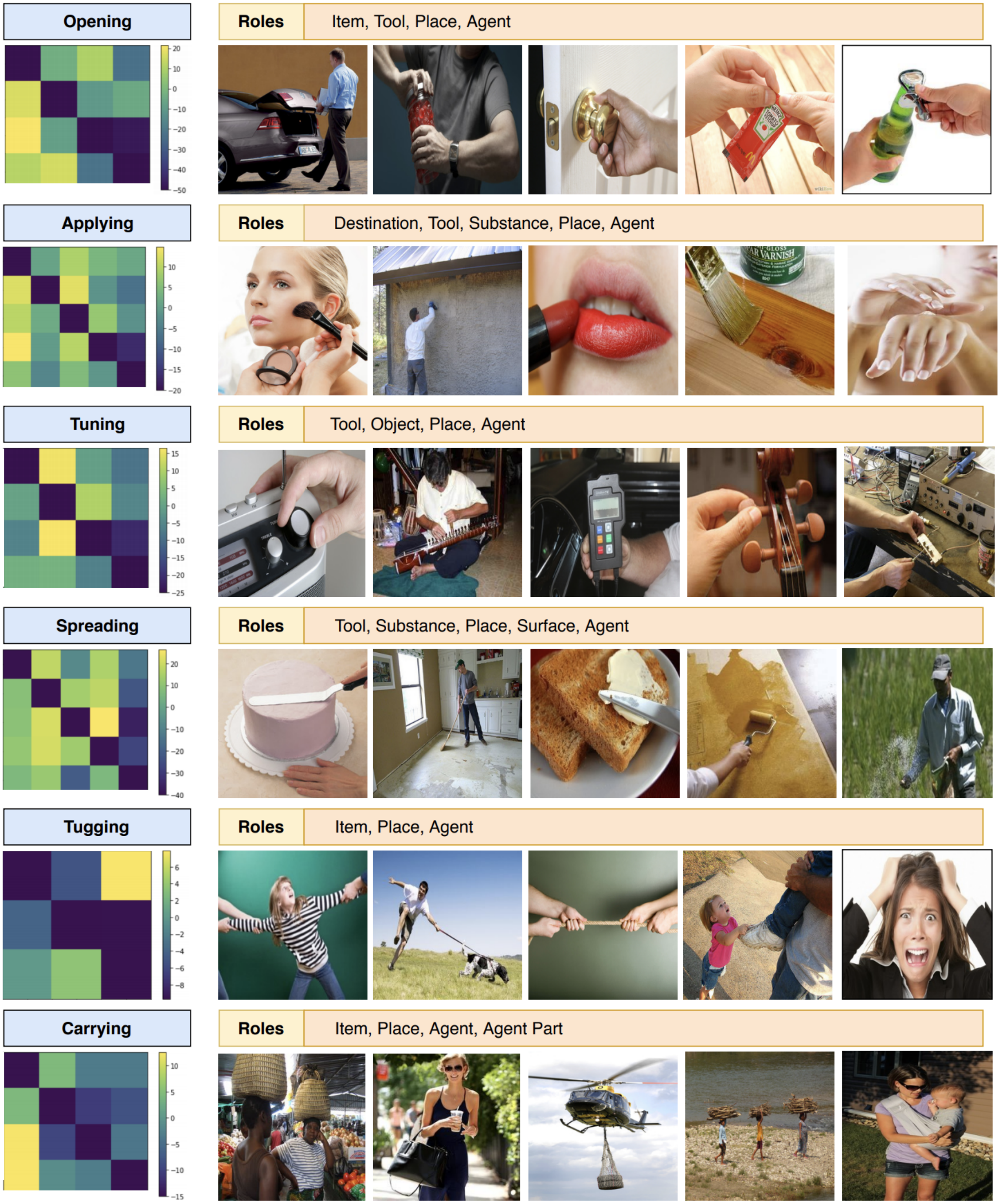
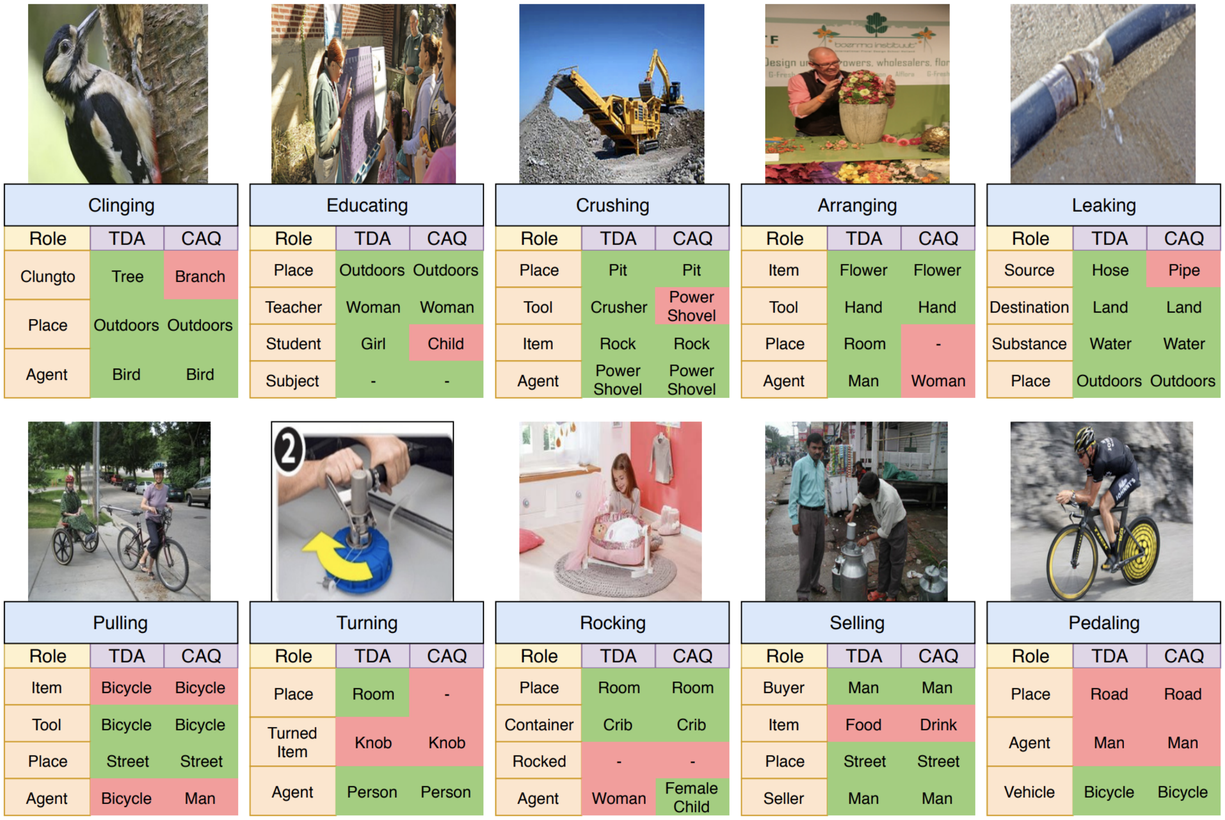


Figure 1: **Role Dependency Matrices** of more verbs with sample images which show different senses of verbs. Role list shows the order of roles occur in the matrix whose rows indicate the *Current Role* and each column shows the *Neighbour Roles*. These samples depict how the role with the most impact and role inter-dependencies vary from verb to verb.

6. Error Analysis

We discuss about the main reasons which caused our FR models to make wrong predictions in this section. We consider errors made by CAQ while TDA has the correct prediction, as well as errors that both TDA and CAQ have made which caused both of them to fail in particular samples. When we call a model has failed in a sample, we mean it has been unable to predict the entire frame (measured by *Value-all* criterion) correctly.



Clinging

Role	TDA	CAQ
Clungto	Tree	Branch
Place	Outdoors	Outdoors
Agent	Bird	Bird

Educating

Role	TDA	CAQ
Place	Outdoors	Outdoors
Teacher	Woman	Woman
Student	Girl	Child
Subject	-	-

Crushing

Role	TDA	CAQ
Place	Pit	Pit
Tool	Crusher	Power Shovel
Item	Rock	Rock
Agent	Power Shovel	Power Shovel

Arranging

Role	TDA	CAQ
Item	Flower	Flower
Tool	Hand	Hand
Place	Room	-
Agent	Man	Woman

Leaking

Role	TDA	CAQ
Source	Hose	Pipe
Destination	Land	Land
Substance	Water	Water
Place	Outdoors	Outdoors

Pulling

Role	TDA	CAQ
Item	Bicycle	Bicycle
Tool	Bicycle	Bicycle
Place	Street	Street
Agent	Bicycle	Man

Turning

Role	TDA	CAQ
Place	Room	-
Turned Item	Knob	Knob
Agent	Person	Person

Rocking

Role	TDA	CAQ
Place	Room	Room
Container	Crib	Crib
Rocked	-	-
Agent	Woman	Female Child

Selling

Role	TDA	CAQ
Buyer	Man	Man
Item	Food	Drink
Place	Street	Street
Seller	Man	Man

Pedaling

Role	TDA	CAQ
Place	Road	Road
Agent	Man	Man
Vehicle	Bicycle	Bicycle

Figure 2: Samples where our models made wrong predictions. **Top Row** : CAQ has made errors for samples TDA has correct predictions. **Bottom Row** : Both models have made wrong predictions according to ground truth annotations. Green is used to indicate correct predictions, red otherwise.

We observed that most errors have happened because of the variety of labels *imSitu* dataset has for visually similar objects. Wrong predictions caused by object classification errors are comparatively lesser. Figure 2 shows examples on this. Top row consists of examples where TDA has predicted correctly, but CAQ has made some errors. We can see other than the *Agent* prediction error of verb “Arranging”, all others have very similar predictions to the correct labels. However since the ground truth annotations do not have these labels included, they have marked as wrong. Same reason have caused in the bottom row also where both TDA and CAQ have failed to predict correctly. For verb “Pulling”, although models have misclassified “carriage” for a “bicycle” as the pulled *Item*, the *Agent* label predictions are very reasonable. But the ground truth only contains “cyclist” and “woman”, hence our predictions are marked wrong. However for the verb “Rocking”, both models have not been able to clearly identify the doll in the crib. When it comes to verb “Selling”, both models have not been able to deduce the *Item* should be “Milk” based on the look of the container. This is because the dataset does not have enough samples to support this information. For the verb “Pedaling”, both our predictions are very relevant. But as they differ from the ground truth, again the predictions are indicated as wrong. Even though *imSitu* has three annotations per image, it has not been able to cover all possible correct answers in some cases.

We believe grouping these vast variety of visually similar objects and narrowing down the possible answer space will be helpful in the future. Because it will allow future work to clearly separate out errors caused by models and address them.

References

[1] Collin F. Baker, Charles J. Fillmore, and John B. Lowe. The berkeley framenet project. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, COLING-ACL '98, August 10-14, 1998, Université de Montréal, Montréal, Quebec, Canada. Proceedings of the Conference.*, pages 86–90, 1998. 1

[2] Saurabh Gupta and Jitendra Malik. Visual semantic role labeling. *CoRR*, abs/1505.04474, 2015. 1

[3] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 448–456, 2015. 3

[4] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 2

[5] Ruiyu Li, Makarand Tapaswi, Renjie Liao, Jiaya Jia, Raquel Urtasun, and Sanja Fidler. Situation recognition with graph neural networks. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 4183–4192, 2017. 2, 4

[6] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, pages 740–755, 2014. 1

[7] Arun Mallya and Svetlana Lazebnik. Recurrent models for situation recognition. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 455–463, 2017. 2

[8] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017. 2

[9] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Computer Vision (ICCV), 2015 IEEE International Conference on*, pages 2641–2649. IEEE, 2015. 1

[10] Carina Silberer and Manfred Pinkal. Grounding semantic roles in images. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2616–2626, 2018. 1

[11] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015. 2

[12] Damien Teney, Peter Anderson, Xiaodong He, and Anton van den Hengel. Tips and tricks for visual question answering: Learnings from the 2017 challenge. *arXiv preprint arXiv:1708.02711*, 2017. 2

[13] Shaohua Yang, Qiaozi Gao, Changsong Liu, Caiming Xiong, Song-Chun Zhu, and Joyce Y. Chai. Grounded semantic role labeling. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 149–159, 2016. 1

[14] Mark Yatskar, Vicente Ordonez, Luke Zettlemoyer, and Ali Farhadi. Commonly uncommon: Semantic sparsity in situation recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[15] Mark Yatskar, Luke Zettlemoyer, and Ali Farhadi. Situation recognition: Visual semantic role labeling for image understanding. In *Conference on Computer Vision and Pattern Recognition*, 2016. 1, 2

[16] Zhou Yu, Jun Yu, Chenchao Xiang, Jianping Fan, and Dacheng Tao. Beyond bilinear: Generalized multimodal factorized high-order pooling for visual question answering. *IEEE Transactions on Neural Networks and Learning Systems*, 29(12):5947–5959, 2018. 3