Meta Spatio-Temporal Debiasing for Video Scene Graph Generation

Li Xu[1⋆], Haoxuan Qu[1⋆], Jason Kuen[2], Jiuxiang Gu[2], and Jun Liu[1⋆⋆]

[1] Singapore University of Technology and Design
{li_xu, haoxuan_qu}@mymail.sutd.edu.sg, jun_liu@sutd.edu.sg
[2] Adobe Research
{kuen, jigu}@adobe.com

1 Analysis of Our MVSGG Framework

The meta-optimization objective of our MVSGG framework is the following (see Eqn. 4 in our main paper):

$$\begin{aligned} &\min_{\omega}\ L_{m\_tr}(\omega) + \sum_{n=1}^{N} L_{m\_te}^{n}(\omega') \\ =&\min_{\omega}\ L_{m\_tr}(\omega) + \sum_{n=1}^{N} L_{m\_te}^{n}\big(\omega - \alpha\nabla_{\omega} L_{m\_tr}(\omega)\big) \end{aligned} \tag{1}$$

Also note that the first-order Taylor expansion can be summarized as:

$$f(x) \approx f(x_0) + f^{'}(x_0)\cdot(x - x_0) \tag{2}$$

where x_0 is close to x. Then inspired by the previous work [1], we can apply the first-order Taylor expansion to the second term in Eqn. 1. Specifically, we can let $x = \omega - \alpha\nabla_{\omega} L_{m\_tr}(\omega)$ and $x_0 = \omega$. Then, we can get:

$$L_{m\_te}^{n}\big(\omega - \alpha\nabla_{\omega} L_{m\_tr}(\omega)\big) \approx L_{m\_te}^{n}(\omega) + \nabla_{\omega} L_{m\_te}^{n}(\omega)\cdot(-\alpha\nabla_{\omega} L_{m\_tr}(\omega)) \tag{3}$$

By substituting the above Taylor expansion into Eqn. 1, the optimization objective of our framework can then be approximately formulated as:

$$\min_{\omega}\ L_{m\_tr}(\omega) + \sum_{n=1}^{N} L_{m\_te}^{n}(\omega) - \sum_{n=1}^{N} \alpha\Big(\nabla_{\omega} L_{m\_te}^{n}(\omega)\cdot\nabla_{\omega} L_{m\_tr}(\omega)\Big) \tag{4}$$

The first two terms in the above optimization objective aim to minimize the model losses on both meta-training data (the support set) and meta-testing data (the query sets), which is similar to conventional model training. Meanwhile, the last term is to maximize the dot product of the model gradients on meta-training data and meta-testing data.

⋆ Both authors contributed equally to the work.
⋆⋆ Corresponding Author

Specifically, as studied by the previous work [3], maximizing the dot product of gradients will encourage a higher direction similarity of the gradients. This means that here the last term in Eqn. 4 can regularize and encourage the model to reach a higher direction similarity of the gradients on meta-training data and meta-testing data, and thus encourages the model to learn features that are shared across the both sets of data. Meanwhile, since the data distributions w.r.t the biases are quite different across the meta-training and meta-testing data, by driving the model to learn the shared features across different data distributions, the model is guided to learn more generalizable features, instead of exploiting data biases.

Our experiments also show the efficacy of our meta-optimization scheme (see Table 7 of our main paper).

2 Additional Ablation Study

Impact of query sets construction strategy. In our framework, after randomly selecting a part of the training data as the support set, we use the remaining training data to construct various query sets where the data distribution of each query set is different from that of the support set w.r.t. one type of conditional bias. To investigate the efficacy of such a query sets construction strategy, we further evaluate the following two variants. The first variant (*Random Query Sets*) totally *randomly* selects data samples to construct each query set. We also evaluate another variant (*Uniform Query Sets*), in which each constructed query set follows an *uniform* data distribution w.r.t. the corresponding conditional bias. In this variant, for example, considering the query set for handling the spatial conditional bias w.r.t. the predicate based on the subject, the probability of the occurrence of each predicate given the same subject is nearly the same (i.e., uniform distribution).

As shown in Table 1, the second variant (*Uniform Query Sets*) outperforms the first one (*Random Query Sets*). We analyze that in the first variant, since the support set and query sets are both randomly constructed, the data distributions of the support set and query sets can be relatively close. However, compared to the first variant, the difference between the data distributions of the support set and query sets w.r.t. the biases is larger in the second variant, which thus benefits the model learning of more generalizable features so as to obtain better generalization performance.

Similarly, our method obtains the best performance compared to these two variants. This can be credited to that in our method, we make the difference between the data distributions of each query set and the support set *as large as possible* w.r.t. a type of conditional biases, i.e., such data distribution difference in our method is obviously larger than those of the both variants.

These experiment results demonstrate that our query sets construction strategy can effectively help our framework to well handle various types of conditional biases for better performance.

Table 1. We evaluate two variants to investigate the impact of query sets construction strategy in our framework.

Method	Relation Detection			Relation Tagging		
	mAP	R@50	R@100	P@1	P@5	P@10
Random Query Sets	29.47	19.77	23.10	72.00	54.80	40.85
Uniform Query Sets	30.30	20.18	23.45	74.50	55.20	41.10
Ours	**31.57**	**21.16**	**24.57**	**79.00**	**57.60**	**43.20**

Table 2. We evaluate a variant to investigate the impact of *meta training and testing*. The optimization objective for this variant is to minimize $L_{m\_tr}(\omega) + \sum_{n=1}^{N} L_{m\_te}^{n}(\omega)$ w.r.t. ω, i.e., replacing ω' with ω in Eqn. 4 in our paper.

Method	Relation Detection			Relation Tagging		
	mAP	R@50	R@100	P@1	P@5	P@10
Training w/o meta	29.79	20.08	23.21	72.00	54.20	40.80
Ours	**31.57**	**21.16**	**24.57**	**79.00**	**57.60**	**43.20**

Impact of meta training and testing. To investigate the efficacy of our *meta training and testing* scheme, we test a variant (*training w/o meta*) that trains the model on both the support set and query sets in the conventional manner without *meta training and testing*. Note that this variant still constructs the support set and query sets as in our framework. As shown in Table 2, our method outperforms this variant obviously, showing that by performing *meta training* over the support set followed by *meta testing* over the query sets and then leveraging the meta-optimization to update the model, our framework can effectively enhance model generalization performance.

Impact of the size of support set. In our framework, we randomly select 60% of the training set as the support set (60% *for support set*), and the remaining 40% of training set is used to construct query sets. Here we further evaluate the following two variants. One variant (50% *for support set*) uses 50% of the training set to construct the support set, and the remaining 50% part to construct the query sets. While another variant (70% *for support set*) uses 70% of the training set to construct the support set, and the remaining 30% part to construct the query sets. As shown in Table 3, our method and these two variants all outperform the baseline model (i.e., VidVRD-II [2]), demonstrating the robustness of our framework w.r.t the varying size of the support set.

Training time. We test the training time of our framework that trains the baseline network (TRACE [4]) with *meta training and testing*, and compare it to the training time of the baseline that trains the same network in the conventional training manner without *meta training and testing*, on AG dataset, as shown in Table 4. We conduct our experiments on an RTX-3090 GPU. Though our

Table 3. We evaluate different variants that use different proportions of the training set to construct the support set and query sets.

Method	Relation Detection			Relation Tagging		
	mAP	R@50	R@100	P@1	P@5	P@10
Baseline (VidVRD-II)	29.37	19.63	22.92	70.40	53.88	40.16
50% for support set	31.44	20.99	24.41	78.50	57.30	43.05
60% for support set	31.57	21.16	24.57	79.00	57.60	43.20
70% for support set	31.42	21.06	24.38	78.00	57.50	42.95

method achieves much better performance, it brings only relatively little increase (15.79%) of the training time.

Table 4. Comparison of the training time. Note that our method achieves significantly better performance than the baseline (see Table 4 and Table 5 in our main paper).

Method	Training Time
Baseline	38 hours
Ours	44 hours

3 Discussion of Our Meta Framework and Train-Validation

In our proposed framework, we first train the model using the support set (i.e., *meta training*), and then evaluate the model performance on the query sets (i.e., *meta testing*). Then the model evaluation performance on the query sets is utilized to provide a *generalization feedback and regularization* to drive the model training towards learning more generalizable features.

Meanwhile, in a classical train-validation scheme that is often used for hyper-parameter selection, we often train the model over the training set, and then evaluate and observe the model performance on the validation set.

From this perspective, our meta framework shares some similarities in concepts with the train-validation scheme, but it is worth noting that, our meta framework is totally different from the classical train-validation scheme, as discussed below.

In our work, we aim to optimize the model learning process, i.e., optimizing the network parameter training towards learning more generalizable features. This goal is not feasible to be achieved via a simple train-validation scheme which is often used for hyper-parameter selection. Meanwhile, in our framework, to achieve this goal, we get inspiration from meta learning (i.e., "learning to learn"), and design a novel meta learning-based training scheme. Concretely, we utilize the model generalization performance on the query sets to provide a *generalization feedback* that involves the second-order gradients, to the model

learning process. Via such a feedback, the model is driven to learn to automatically adjust the learning process to learn more generalizable features.

Moreover, to further drive the model to generalize well against various types of conditional biases, our framework splits the original training set to construct a support set (meta-training data) and multiple query sets (meta-testing data), so that the difference between the distributions of each constructed query set and the support set is *as large as possible* w.r.t. a type of conditional bias. This can help ensure that the biases in meta-training data no longer hold in meta-testing data, and thus the model needs to more focus on learning generalizable features during our *meta training and testing* scheme.

Due to the effectiveness of our meta framework with the above designs, our framework achieves superior performance on the evaluated benchmarks.

4 Qualitative Results

We present some qualitative results w.r.t. various types of spatio-temporal conditional biases in Fig 1. As shown, when handling testing samples where the spatio-temporal conditional biases do not hold, our framework performs better than the baseline model [2], demonstrating that our framework can effectively optimize the model to well generalize against various types of conditional biases.

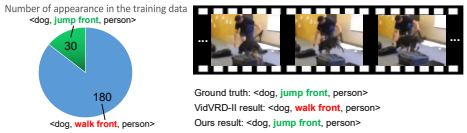
5 Query Sets Visualization

In our framework, we construct a support set and various query sets for *meta training and testing*, where each query set is designed to address a type of conditional bias. As shown in Fig. 2, we present a visualization of an example query set that is constructed to handle the spatial conditional bias w.r.t. the predicate conditioned on subject-object pair. We can observe that conditioned on each subject-object pair, the distribution of predicates in this query set is quite different from that of the support set. Similarly, we present a visualization of another example query set for handling the spatial conditional bias w.r.t. the object conditioned on subject-predicate pair in Fig. 3. As shown, conditioned on each subject-object pair, the distribution of objects in this query set is also quite different from that of the support set.

In this manner, by improving the model generalization performance on these two query sets after training on the support set, the trained model is encouraged to learn to generalize against the corresponding two types of conditional biases. Similarly, by enhancing the model performance on various query sets where each query set is distributionally different from the support set w.r.t a type of conditional bias, our framework can effectively train the model to learn to capture more generalizable features against various types of conditional biases.

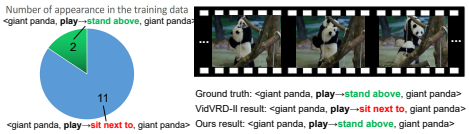
(a) Results w.r.t spatial conditional bias (*object-centered*).

(b) Results w.r.t spatial conditional bias (*predicate-centered*).

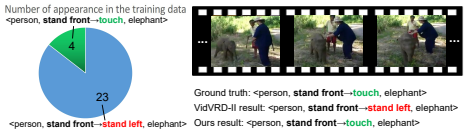


(c) Result w.r.t. spatial conditional bias. (*predicate-centered*).

(d) Results w.r.t spatial conditional bias (*subject-centered*).



(e) Results w.r.t temporal conditional bias (*forward case*).



(f) Results w.r.t temporal conditional bias (*forward case*).

(g) Results w.r.t temporal conditional bias (*backward case*).

Fig. 1. Qualitative results of our method and the baseline model [2]. As shown, our method demonstrates better performance against various types of spatio-temporal conditional biases.

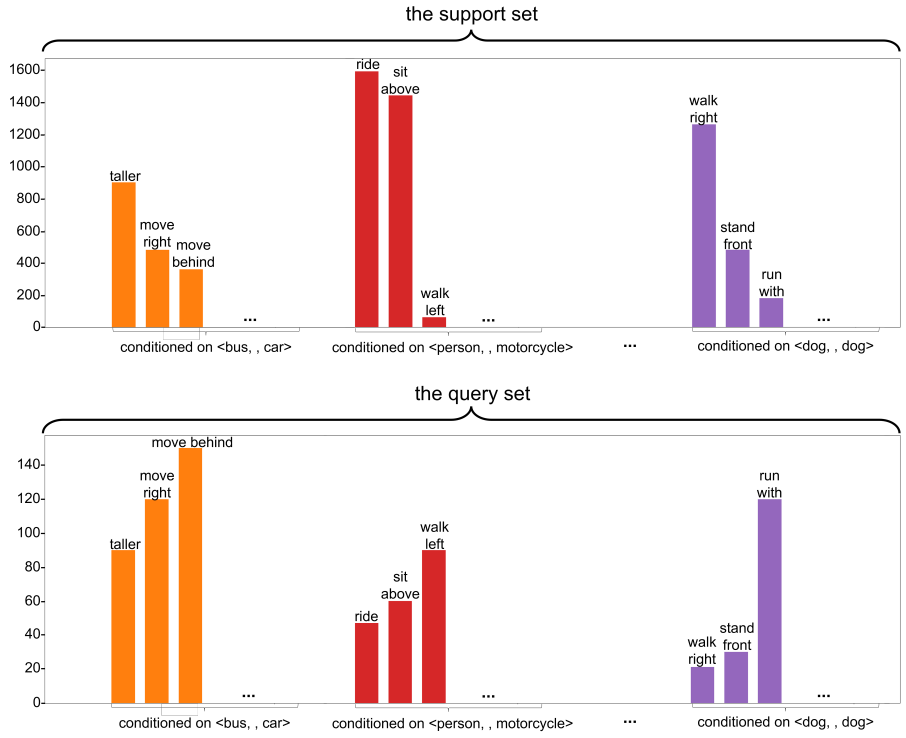


Fig. 2. Visualization of an example query set for handling the spatial conditional bias w.r.t. the **predicate conditioned on subject-object pair**. As shown above, conditioned on each subject-object pair, the distribution of predicates in the query set is different from that of the support set.

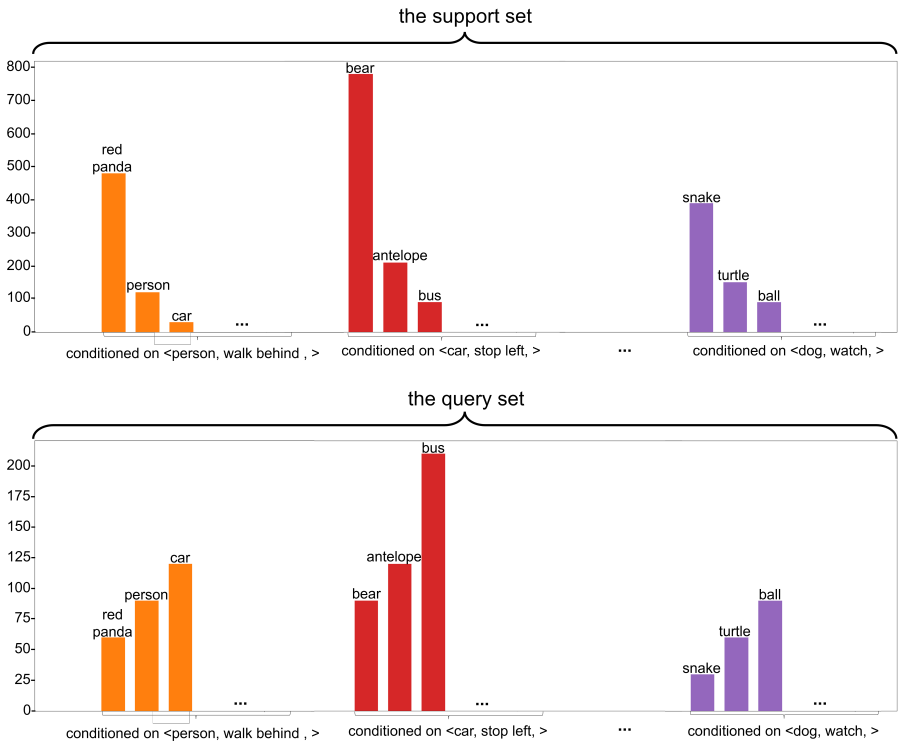


Fig. 3. Visualization of an example query set for handling the spatial conditional bias w.r.t. the **object conditioned on subject-predicate pair**. As shown above, conditioned on each subject-predicate pair, the distribution of objects in the query set is different from that of the support set.

References

1. Li, D., Yang, Y., Song, Y.Z., Hospedales, T.: Learning to generalize: Meta-learning for domain generalization. Proceedings of the AAAI Conference on Artificial Intelligence **32** (Apr 2018)
2. Shang, X., Li, Y., Xiao, J., Li, W., Chua, T.S.: Video visual relation detection via iterative inference. In: Proceedings of the 29th ACM international conference on Multimedia (2021)
3. Shi, Y., Seely, J., Torr, P., N, S., Hannun, A., Usunier, N., Synnaeve, G.: Gradient matching for domain generalization. In: International Conference on Learning Representations (2022), https://openreview.net/forum?id=vDwBW49HmO
4. Teng, Y., Wang, L., Li, Z., Wu, G.: Target adaptive context aggregation for video scene graph generation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13688–13697 (2021)