Supplementary Material for "VCD: Visual Causality Discovery for Cross-Modal Question Reasoning"

Yang Liu[1](✉), Ying Tan[1], Jingzhou Luo[1], and Weixing Chen[1]

Sun Yat-sen University, Guangzhou, China liuy856@mail.sysu.edu.cn, {tany86, luojzh5}@mail2.sysu.edu.cn,chen867820261@gmail.com

This supplementary document will further detail the following aspects in the submitted manuscript: 1. Adaptive Feature Fusion, 2. Derivation Details from Eq. (5)-(6), 3. Answer Prediction, 4. More Details of Datasets, 5. More Parameters Analysis, 6. More Comprehensive Visualization Results.

1 Adaptive Feature Fusion

For (v, q) and $(\hat{c}, q)$, their respective visual and linguistic outputs of the IVLT model are denoted as F, F_c and L, L_c, respectively. Inspired by the adaptive feature fusion in [5], since the linguistic features from different semantic roles are correlated, we build an adaptive linguistic feature fusion module that receives features from different semantic roles and learns a global context embedding, then this embedding is used to recalibrate the input features from different semantic roles. The linguistic features of nodes learned are $\{L_1, L_2\} = \{L, L_c\}$, where $L_k \in \mathbb{R}^{2d}(k = 1, \cdots, 2)$. To utilize the correlation among linguistic features, we concatenate these linguistic features and get joint representations G_u^k for each semantic role L_k through a fully-connected layer:

$$G_u^k = W_s^k[L_1, L_2,] + b_s^k, \;\; k = 1, \cdots, 2 \tag{1}$$

where $[\cdot, \cdot]$ denotes the concatenation operation, $G_u^k \in \mathbb{R}^{d_u}$ denotes the joint representation, W_s^k and b_s^k are weights and bias of the fully-connected layer. We choose $d_u = d$ to restrict the model capacity and increase its generalization ability. To make use of the global context information aggregated in the joint representations G_u^k, we predict excitation signal for it via a fully-connected layer:

$$E^k = W^k G_u^k + b^k, \;\; k = 1, \cdots, 2 \tag{2}$$

where W^k and b^k are weights and biases of the fully-connected layer. After obtaining the excitation signal $E^k \in \mathbb{R}^c$, we use it to recalibrate the input feature L_k adaptively by a simple gating mechanism:

$$\widetilde{L}_k = \delta(E^k) \odot L_k \tag{3}$$

where $\odot$ is channel-wise product operation for each element in the channel dimension, and $\delta(\cdot)$ is the ReLU function. In this way, we can allow the features of one semantic role to recalibrate the features of another semantic role while concurrently preserving the correlation among different semantic roles. Then, these refined linguistic feature

vectors $\{\widetilde{L}, \widetilde{L}_c\}$ are concatenated to form the final semantic-ware linguistic feature $\widetilde{L} = [\widetilde{L}, \widetilde{L}_c] \in \mathbb{R}^{2d}$.

To obtain the semantic-aware visual feature, we compute the visual feature $\widetilde{F}_k$ by individually conditioning each semantic role from the visual features $\{F_1, F_2\} = \{F, F_c\}$ to each semantic role from the refined linguistic features $\{\widetilde{L}_1^e, \widetilde{L}_2^e\} = \{\widetilde{L}^e, \widetilde{L}_c^e\}$ using the same operation as [4]. For each semantic role k $(k = 1, 2)$, the weighted semantic-aware visual feature is:

$$\begin{aligned} I_k &= \text{ELU}\big(W_k^I[W_k^f F_k, W_k^f F_k \odot W_k^l \widetilde{L}_k^e] + b_k^I\big) \\ \widetilde{F}_k &= \text{Softmax}(W_k^{I'} I_k + b_k^{I'}) \odot F_k \end{aligned} \tag{4}$$

Then, these semantic-aware visual features $\widetilde{F}_k$ $(k = 1, \cdots, 2)$ are concatenated to form the final semantic-aware visual feature $\widetilde{F} = [\widetilde{F}_1, \widetilde{F}_2] \in \mathbb{R}^{2d}$.

2 Derivation Details from Eq. (5)-(6)

For Eq. (6), we need to implement it in a deep framework. According to the previous works [7, 1], the visual question answering task can be transformed into the classification formulation. Therefore, we can parameterize the predictive distribution $P(A|V, M)$ as a network $g(\cdot)$ followed by a softmax layer, and thus get Softmax$[g(M, V)]$. As can be seen in Eq. (6), we need to sample V and M, and feed them into the network to complete $P(A|do(V), Q)$. However, the cost of the network forward pass for all of these samples is prohibitively expensive. To address this challenge, we apply Normalized Weighted Geometric Mean (NWGM) approximation [11, 6] to absorb the outer sampling into the feature level and thus only need to forward the "absorbed input" in the network for once. Actually, $\hat{M}$ is essentially an in-sample sampling process, where m denotes the selected knowledge from the current input sample V, $\hat{V}$ is essentially a cross-sample sampling process since it comes from the other samples. Therefore, both $\hat{M}$ and $\hat{V}$ can be calculated by attention networks.

We first show how to use Normalized Weighted Geometric Mean (NWGM) approximation to absorb the sampling into the network for deriving Eq. (7). Before introducing NWGM, we first revisit the calculation of a function $y(x)$'s expectation according to the distribution $P(x)$:

$$\mathbb{E}_x[y(x)] = \sum_x y(x)P(x) \tag{5}$$

which is the weighted arithmetic mean of $y(x)$ with $P(x)$ as the weights.

Correspondingly, the weighted geometric mean (WGM) of $y(x)$ with $P(x)$ as the weights is:

$$\text{WGM}(y(x)) = \prod_x y(x)^{p(x)} \tag{6}$$

where the weights $P(x)$ are put into the exponential terms. If $y(x)$ is an exponential function that $y(x) = exp[g(x)]$, we have:

$$\begin{aligned} \text{WGM}(y(x)) &= \prod_x y(x)^{P(x)} \\ &= \prod_x \exp[g(x)]^{P(x)} \\ &= \prod_x \exp[g(x)P(x)] \\ &= \exp[\sum_x g(x)P(x)] \\ &= \exp\{\mathbb{E}_x[g(x)]\} \end{aligned} \tag{7}$$

where the expectation $\mathbb{E}_x$ is absorbed into the exponential term. Based on this observation, we approximate the expectation of a function as the WGM of this function in the deep network whose last layer is a Softmax layer [11, 6]:

$$\mathbb{E}_x[y(x)] \approx \text{WGM}(y(x)) = \exp\{\mathbb{E}_x[g(x)]\} \tag{8}$$

where $y(x) = \exp[g(x)]$.

In our case, we treat $P(A|V, M)$ (Eq.(7)) as a predictive function and parameterize it by a network with a Softmax layer as the last layer:

$$P(A|V, M) = \text{Softmax}[g(V, M)] \propto \exp[g(V, M)] \tag{9}$$

Following Eq.(7) of the manuscript and $E_x[y(x)] \propto \text{WGM}(y(x)) = \exp\{\mathbb{E}_x[g(x)]\}$, we have:

$$\begin{aligned} &P(A|do(V), Q) \\ &= \sum_m P(M = m|V, Q) \sum_v P(V = v)P(A|V = v, M = m) \\ &= \mathbb{E}_{[M|V]}\mathbb{E}_v[P(A|M, V)] \\ &\approx \text{WGM}(P(A|V, M)) \\ &\approx \exp\{[g(\mathbb{E}_{[M|V]}[M], \mathbb{E}_v[V])]\} \end{aligned} \tag{10}$$

Note that, $P(A|V, M)$ is only proportional to $\exp[g(V, M)]$ instead of strictly equaling to, we only have $\text{WGM}(P(A|V, M)) \approx \exp\{[g(\mathbb{E}_{[M|V]}[M], \mathbb{E}_v[V])]\}$ instead of equaling to. Furthermore, to guarantee the sum of $P(A|do(V), Q)$ to be 1, we use a Softmax layer to normalize these exponential units:

$$P(A|do(V), Q) \approx \text{Softmax}(g(\mathbb{E}_{[M|V]}[M], \mathbb{E}_v[V])) \tag{11}$$

where the first part $\mathbb{E}_{[M|V]}[M]$ is in-sample sampling and the second part $\mathbb{E}_v[V]$ is cross-sample sampling. Since the Softmax layer normalizes these exponential terms, this is called the normalized weighted geometric mean (NWGM) approximation.

In a network, the variables V and M are represented by the embedding vectors and thus we use v and m to denote them. Following the convention in attention research

	Video	QA pairs	Count	Action	Transition	FrameQA
Train	62,846	139,414	26,843	20,475	52,704	39,392
Test	9,575	25,751	3,554	2,274	6,232	13,691
Total	71,741	165,165	30,397	22,749	58,936	53,083

Table 1. Statistics of the TGIF-QA dataset.

	Video	QA pairs	What	Who	How	When	Where
Train	1,200	30,933	19,485	10,479	736	161	72
Val	250	6,415	3,995	2,168	185	51	16
Test	520	13,157	8,149	4,552	370	58	28
Total	1,970	50,505	31,629	17,199	1,291	270	116

Table 2. Statistics of the MSVD-QA dataset.

where the attended vectors are usually represented in the matrix form, we also pack the estimated in-sample sampling and cross-sample sampling vectors to $\hat{V}$ and $\hat{M}$. In this way, we have:

$$P(A|do(V), Q) \approx \text{Softmax}(g(\hat{M}, \hat{V})) \tag{12}$$

which is given in Eq.(7) in the submitted manuscript.

To estimate $\hat{M}$, we usually calculate a query set from $V : Q_I = f(V)$ and use it in the Q-K-V operation. Similarly, to estimate $\hat{V}$, we can also calculate a query set as: $Q_c = h(V)$ and use it in the Q-K-V operation. In this way, we have Eq. (7) in the submitted manuscript:

$$\begin{aligned} &P(A|do(V), Q) \approx \text{Softmax}(g(\hat{M}, \hat{V})) \\ &= \text{Softmax}[g(\sum_m P(M = m|f(V))m, \sum_v P(V = v|h(V))v)] \end{aligned} \tag{13}$$

Note that although $P(V)$ in cross-sampling does not condition on any variable, we still require a query in Q-K-V operation, since without a query, the estimated result will degrade into a fixed single vector for each different input $V : \hat{v} = \sum_v P(v)v$, where $P(v)$ is the prior probability. We can also treat it as the strategy to increase the representation power of the whole model.

For visual front-door intervention, we only conduct intervention for visual modality, while the Q is already intervened in the former linguistic back-door intervention. Therefore, the $P(A|do(V), Q)$ will not influence Q.

3 Answer Prediction

The event-level visual question answering is usually divided into three types: open-ended, multi-choice, and counting. Here, we introduce how to infer the answer based on the semantic-aware visual feature $\widetilde{F}$ and linguistic feature $\widetilde{L}$.

Open-ended task is essentially a multi-label classification problem, which requires a model to predict an answer that belongs to a pre-defined answer set. We use an FC

	Video	QA pairs	What	Who	How	When	Where
Train	6,513	158,581	108,792	43,592	4,067	1,626	504
Val	497	12,278	8,337	3,439	344	106	52
Test	2,990	72,821	49,869	20,385	1,640	677	250
Total	10,000	243,680	166,998	67,416	6,051	2,409	806

Table 3. Statistics of the MSRVTT-QA dataset.

layer activated by softmax function to construct a classifier. The cross-entropy is used as the loss function:

$$\text{Loss} = -\sum_{i=1}^{A} y_i \log p_i \tag{14}$$

where A is the size of the pre-defined answer set. $y_i = 1$ if the i-th candidate is the right answer, otherwise $y_i = 0$. p_i is the predicted score of the i-th candidate answer.

Multi-choice task requires a model to choose an answer from several candidate answers given. We concatenate the question with each candidate answer individually before feed into our model. Then, a linear regression layer is employed to generate scores for all candidate answers. We use the hinge loss for pairwise comparisons between the scores of the correct answer s^p and incorrect answer s_i^n:

$$\text{Loss} = -\sum_{i=1}^{C-1} \max(0, m + s_i^n - s^p) \tag{15}$$

where C is the number of candidate answers and $i \leq C - 1$. m indicates that the score of the correct question-answer pair should be larger than any incorrect pair by a margin m.

Counting task requires a model to predict a number from a pre-defined number range. We use an FC layer to generate the number and Mean Square Error (MSE) is selected as the loss function:

$$\text{Loss} = (x - y)^2 \tag{16}$$

where x is the predicted number and y is the ground truth. The predicted values are rounded to the nearest integer.

4 More Details of Datasets

SUTD-TrafficQA [12]. This dataset consists of 62,535 QA pairs and 10,090 videos collected from traffic scenes. There are six challenging reasoning tasks including basic understanding, event forecasting, reverse reasoning, counterfactual inference, introspection and attribution analysis. The basic understanding task is to perceive and understand traffic scenarios at the basic level. The event forecasting task is to infer future events based on observed videos, and the forecasting questions query about the outcome of the current situation. The reverse reasoning task is to ask about the events that have happened before the start of a video. The counterfactual inference task queries the consequent outcomes of certain hypothesis that do not occur. The introspection task

is to test if models can provide preventive advice that could have been taken to avoid traffic accidents. The attribution task seeks the explanation about the causes of traffic events and infer the underlying factors.

TGIF-QA [3]. This dataset has 165K QA pairs collected from 72K animated GIFs. It has four tasks: repetition count, repeating action, state transition, and frame QA. Repetition count is a counting task that requires a model to count the number of repetitions of an action. Repetition action and state transition are multi-choice tasks with 5 optional answers. FrameQA is an open-ended task with a pre-defined answer set, which can be answered from a single video frame. Table 1 shows the statistics of the TGIF-QA dataset.

MSVD-QA [9]. This dataset is created from the Microsoft Research Video Description Corpus [2] that is widely used in the video captioning task. It consists of 50,505 algorithm-generated question-answer pairs and 1,970 trimmed video clips. Each video is approximately 10 seconds. It contains five question types: What, Who, How, When, and Where. The dataset is an open-ended task and divided into three splits: training, validation, and test. The statistics of the MSVD-QA dataset are presented in Table 2.

MSRVTT-QA [9]. This dataset is a larger dataset with more complex scenes, which is constructed from the MSRVTT dataset [10]. It contains 10,000 trimmed video clips of approximately 15 seconds each. A total of 243,680 question-answer paris contained in this dataset are automatically generated by the NLP algorithm. The dataset contains five question types: What, Who, How, When, and Where. The dataset is an open-ended task and divided into three splits: training, validation, and test. The statistics of the MSRVTT-QA dataset are presented in Table 3.

5 More Parameters Analysis

To validate whether our CMQR could generalize to different visual appearance and motion features, we evaluate the performance of the CMQR using different visual backbones, as shown in Table 4. These results validates that our CMQR generalizes well across both vision-language transformers and CNN backbones due to the learned causality-ware visual-linguistic representations. More importantly, the performance improvement of our CMQR is mainly attributed to our visual causality discovery model.

From Table 5, we can see that 8 MMA heads performs the best because more heads can facilitate the MMA module employ more perspectives between different modalities. For MTB layers, the optimal layer numbers are different for different datasets. For the dimension of hidden states, 512 is the best dimensionality of hidden states of the CMQR model due to its good compromise between the feature representation ability and model complexity.

6 More Comprehensive Visualization Results

To verify the ability of the CMQR in robust spatial-temporal relational reasoning, we grasp the visual-linguistic causal reasoning insight of the CMQR by inspecting some

	Method	Appearance	Motion	Accuracy
SUTD-QA	Eclipse [12]	ResNet-101	MobileNetV2	37.05
	Ours	XCLIP	XCLIP	**38.63** (+1.59)
	Ours	Swin-L	Video Swin-B	38.58 (+1.54)
	Ours	ResNet-101	ResNetXt-101	38.10 (+1.05)
MSVD-QA	DualVGR [8]	ResNet-101	ResNetXt-101	39.0
	Ours	XCLIP	XCLIP	**46.4** (+7.40)
	Ours	Swin-L	Video Swin-B	43.7 (+4.70)
	Ours	ResNet-101	ResNetXt-101	40.3 (+1.30)
MSRVTT-QA	HCRN [4]	ResNet-101	ResNeXt-101	35.6
	Ours	XCLIP	XCLIP	**38.9**(+3.30)
	Ours	Swin-L	Video Swin-B	38.6 (+3.00)
	Ours	ResNet-101	ResNeXt-101	37.0 (+1.40)

Table 4. Performance with different visual appearance and motion features on SUTD-TrafficQA, MSVD, and MSRVTT datasets.

		SUTD	TGIF (A)	TGIF (T)	TGIF (F)	TGIF (C)	MSVD	MSRVTT
MMA Heads	1	37.83	75.8	80.7	61.2	3.92	45.0	38.5
	2	38.17	75.7	79.7	60.6	3.96	44.7	38.5
	4	37.51	75.8	79.2	61.1	3.93	44.9	38.3
	8	**38.63**	**78.1**	**82.4**	**62.3**	**3.83**	**46.4**	**38.9**
MTB Layers	3	**38.63**	75.1	80.1	61.0	4.03	45.7	38.4
	4	37.84	76.6	80.2	61.6	3.96	45.3	38.7
	5	37.63	75.5	80.6	61.0	3.94	**46.4**	38.7
	6	37.73	76.2	80.8	61.4	4.12	45.9	**38.9**
	7	37.73	75.4	80.3	61.2	3.98	45.8	38.3
	8	37.58	**78.1**	**82.4**	**62.3**	**3.83**	45.5	38.6
Dimension	256	37.60	73.9	79.9	61.0	3.96	45.5	38.8
	512	**38.63**	**78.1**	**82.4**	**62.3**	**3.83**	**46.4**	**38.9**
	768	37.74	75.0	80.0	62.2	3.90	45.5	38.0

Table 5. Performance of CMQR with different MMA heads, MTB layers, and hidden state dimension on four datasets.

correct and failure examples from the SUTD-TrafficQA dataset and show the visualization results in Fig. 1. We respectively show how our model conducts robust spatial-temporal relational reasoning and how it reduce the spurious correlation.

Reliable reasoning. As shown in Fig. 1 (a), there exists the ambiguity problem that the dominant visual regions of the accident may be distracted by other visual concepts (i.e., different cars/vehicles on the road). In our CMQR, we learn the question-relevant visual-linguistic association by causal relational learning, thus we mitigate such ambiguity in our inference results where video-question-answer triplets exhibit a strong correlation between the dominant spatial-temporal scenes and the question semantics. This validates that the CMQR can focus on the right visual regions reliably when making decisions.

Removing bad confounding effect. In Fig. 1 (b), we present a case reflecting the confounding effect, where the visual regions of "van" are spuriously correlated with associated with the "sedan", due to their frequent co-occurrences. In other words, the model will hesitate about the region-object correspondence when encountered with the visual concepts of "van" and "motorbike". In our CMQR, we remove such confounding

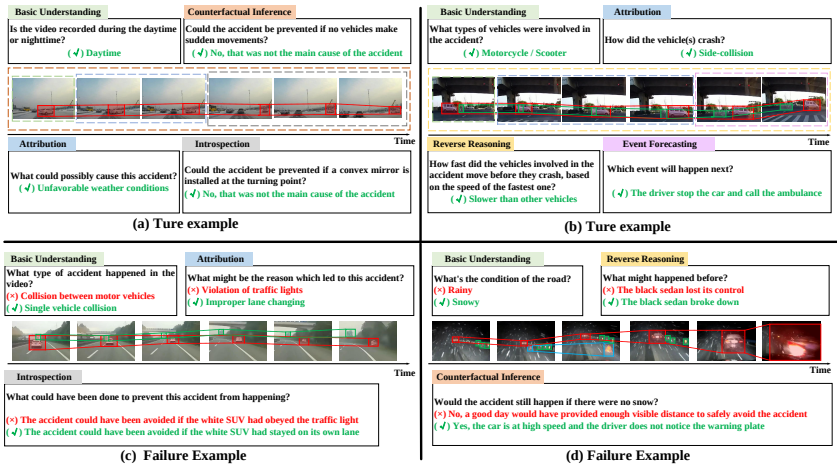

Fig. 1. Visualization of two visual-linguistic causal reasoning examples on the correct prediction cases from SUTD-TrafficQA dataset. Each video is accompanied by four question types that requires multi-level interaction and causal relations between the language and spatial-temporal structure of the video. The color windows denotes the concentrated causal scenes for the corresponding question types.

effect and pursue the true causality by adopting visual-linguistic causal intervention, and we show better dominant visual evidence and the question intention.

Generalization ability. From Fig. 1 (a)-(b), we can see that the CMQR can generalize well across different question types. which shows that the CMQR is question-sensitive to effectively capture the dominant spatial-temporal content in the videos by conducting robust and reliable spatial-temporal relational reasoning.

Introspective and counterfactual learning. For challenging question types like introspection and counterfactual inference, the CMQR model can faithfully introspect whether the attended scene reflects the logic behind the answering. This verifies that the CMQR can fully explore the causal, logic, and spatial-temporal structures of the visual and linguistic content, due to its promising ability of robust visual-linguistic causal reasoning that disentangles the spurious correlations of visual and linguistic modalities.

Additional failure cases. Moreover, we provide the failure examples in Fig. 1 (c)-(d), to have further insights into the limitations of our method. In Fig. 1 (c), our model mistakenly correlates the visual concept "suv" and the green "traffic plate" when conducting visual-linguistic reasoning. It is because the visual region of "traffic plate" appears like the "truck", while there only exists the white "suv" in the video. In Fig. 1 (d), it is hard to discriminate "rainy" and "snowy" due to the similar visual appearance in the video. And the "reflective stripes" along the road are mistakenly considered as the dominant visual concepts. Since our CMQR model contains no explicit object detection pipeline, some ambiguity visual concepts are challenging to be determined. Additionally, without external prior knowledge about traffic rules, some questions like "how to prevent the accident" and "the cause of the accident" are hard to answer. A possible

solution may be incorporating object detection and external knowledge of traffic rules into our method. We will explore it in our future work.

References

1. Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C.L., Parikh, D.: Vqa: Visual question answering. In: Proceedings of the IEEE international conference on computer vision. pp. 2425–2433 (2015)
2. Chen, D., Dolan, W.B.: Collecting highly parallel data for paraphrase evaluation. In: Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies. pp. 190–200 (2011)
3. Jang, Y., Song, Y., Yu, Y., Kim, Y., Kim, G.: Tgif-qa: Toward spatio-temporal reasoning in visual question answering. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2758–2766 (2017)
4. Le, T.M., Le, V., Venkatesh, S., Tran, T.: Hierarchical conditional relation networks for video question answering. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9972–9981 (2020)
5. Liu, Y., Wang, K., Liu, L., Lan, H., Lin, L.: Tcgl: Temporal contrastive graph for self-supervised video representation learning. IEEE Transactions on Image Processing **31**, 1978–1993 (2022)
6. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research **15**(1), 1929–1958 (2014)
7. Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: A neural image caption generator. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3156–3164 (2015)
8. Wang, J., Bao, B., Xu, C.: Dualvgr: A dual-visual graph reasoning unit for video question answering. IEEE Transactions on Multimedia (2021)
9. Xu, D., Zhao, Z., Xiao, J., Wu, F., Zhang, H., He, X., Zhuang, Y.: Video question answering via gradually refined attention over appearance and motion. In: Proceedings of the 25th ACM international conference on Multimedia. pp. 1645–1653 (2017)
10. Xu, J., Mei, T., Yao, T., Rui, Y.: Msr-vtt: A large video description dataset for bridging video and language. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5288–5296 (2016)
11. Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., Bengio, Y.: Show, attend and tell: Neural image caption generation with visual attention. In: International conference on machine learning. pp. 2048–2057 (2015)
12. Xu, L., Huang, H., Liu, J.: Sutd-trafficqa: A question answering benchmark and an efficient network for video reasoning over traffic events. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9878–9888 (2021)