A Multi-modal Debiasing Model with Dynamical Constraint for Robust Visual Question Answering

Yu Li[1], **Bojie Hu**[1, 2], **Fengshuo Zhang**[1], **Yahan Yu**[2], **Jian Liu**[1],
Yufeng Chen[1] and **Jinan Xu**[1]*
[1] Beijing Key Lab of Traffic Data Analysis and Mining, Beijing Jiaotong University, China
[2] Tencent Minority-Mandarin Translation, Beijing, China
{yuyuli, fengshuozhang, jianliu, jaxu, chenyf}@bjtu.edu.cn
bojiehu@tencent.com, jasmineyuyh@gmail.com

Abstract

Recent studies have pointed many well-developed Visual Question Answering (VQA) systems suffer from bias problem. Despite the remarkable performance gained on In-Distribution (ID) datasets, the VQA model might capture the superficial correlation from question to answer rather than showing real reasoning abilities. Therefore, when switching to Out-of-Distribution (OOD) dataset, whose test distribution is unknown or even reversed with the training set, significant drops appear. Efforts have been devoted to negative bias brought by language prior but are still limited by two aspects. First, most current debiasing methods achieve promising OOD generalization ability with a sacrifice of the ID performance. Second, they are restricted by exploiting comprehensive biases, since weakening the language bias is mainly focused and few works consider vision bias. In this paper, we investigate a straightforward way to mitigate bias problem for VQA task by subtracting bias score from VQA base score. Then we design two bias learning branches to detect more bias, which is combined with a dynamical constraint loss to alleviate the problem of over-correction and insufficient debiasing. We evaluate our method on the challenging VQA v2.0 and VQA-CP V2.0 datasets and achieve significant improvement.

1 Introduction

Visual Question Answering (VQA) (Antol et al., 2015) is a challenging task spanning both computer vision and natural language processing. The goal of VQA is to infer the answer based on a given image and a textual question, which is generally cast as a **classification problem**. Promising results on test set whose distribution is analogous with the training set, such as VQA v2.0 (Goyal et al., 2017), are generally favorable. However, latest studies (Agrawal et al., 2016; Goyal et al., 2017) have pointed out that many well-developed VQA models merely over-exploit the language prior from the training set to provide correct answers without reasoning. That is, the answer prediction might rely more on the correlation to the question and less on the image. For instance, in the VQA-CP v2.0 (Agrawal et al., 2018) training set, the answers of the question with the first few words "$how\ many \cdots$" are usually "2", and the answers of the specific question "$what's\ the\ color\ of\ the\ bananas?$" are almost all "$yellow$". Consequently, significant drops (Agrawal et al., 2018) are demonstrated while handling with the out-of-distribution test dataset.

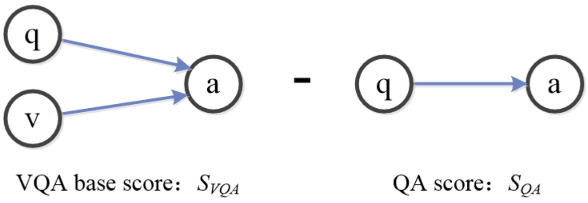

Figure 1: A straightforward way to mitigate the bias problem for VQA task is by subtracting the QA score from the VQA base score.

Recently, solutions for this problem can be categorized into two classes, namely, non-augmentation-based methods (Cadene et al., 2019; Ramakrishnan et al., 2018; Wu and Mooney, 2019; Selvaraju et al., 2019; Clark et al., 2019; Jing et al., 2020; Niu et al., 2021) and augmentation-based methods (Chen et al., 2020; Gokhale et al., 2020; Liang et al., 2020; Teney et al., 2020). The former seeks to weaken language bias or leverage visual grounding to increase the image dependency, while the latter aims to to balance the dis-tribution of training data. Moreover, most of the advanced debiasing methods still suffer from two issues, namely comprehensive bias detecting (Wen et al., 2021) and In-Distribution (ID) generalizability problems (Niu and Zhang, 2021).

In this work, firstly we explore a very straight-

[*] Jinan Xu is the corresponding author

Findings of the Association for Computational Linguistics: ACL 2023, pages 5032–5045
July 9-14, 2023 ©2023 Association for Computational Linguistics

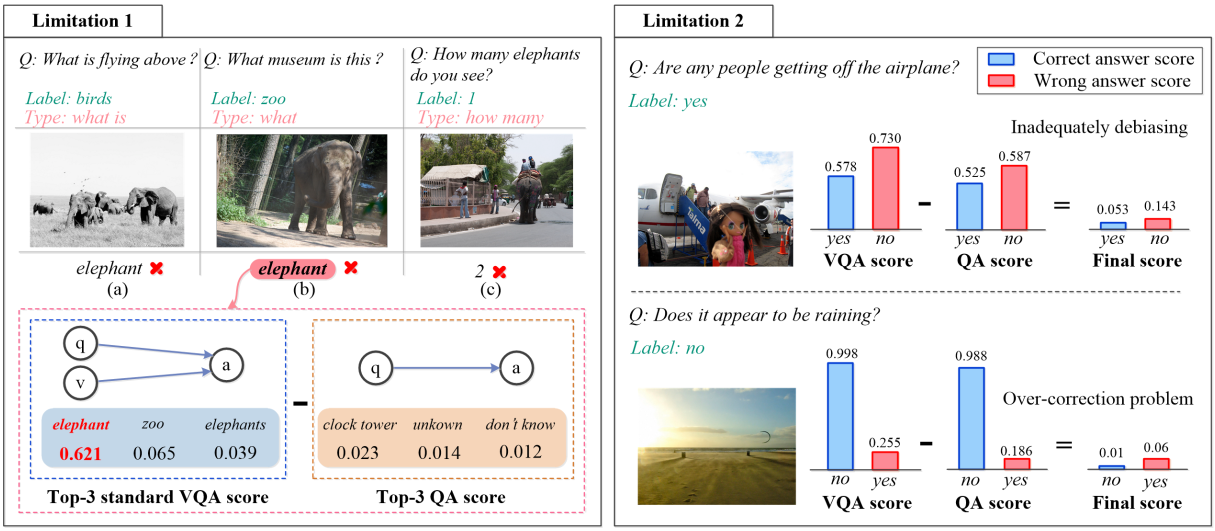

Figure 2: Examples from VQA-CP v2.0 test set when using the subtracting way for debiasing based on Updn model.

Model	ALL	Y/N	NUM.	Other
Updn (S_{VQA})	39.80	41.39	12.10	46.56
Updn (S_{VQA} - S_{QA})	**51.49**	**76.38**	**13.93**	**48.74**

Table 1: Debiasing effect on VQA-CP v2.0 dataset when applying our subtracting debiasing strategy (i.e., S_{VQA} - S_{QA}) to Updn (i.e., S_{VQA}). The best score is in **bold**.

forward solution to VQA bias, which is shown in Figure 1. Generally, we try to design different strategies for learning and inference via a VQA base model and a question answering (QA) model, beforehand. In the training procedure, these two models are separately optimized, and let S_{VQA}, S_{QA} denote the VQA base score and QA score, respectively. In the inference procedure, we calculate the debiased result by subtracting the S_{QA} from S_{VQA}. We apply such a simple strategy to the popular VQA model Updn (Anderson et al., 2018) on VQA-CP v2.0 dataset, and we find that the overall accuracy gains from 39.80% to 51.49% as shown in Table 1. Despite its remarkable performance, we still identify the above two major limitations in this strategy, which specifically reflect the following two aspects:

- **First, the model answers questions without comprehensively exploiting vision bias.** Figure 2 (a) & (b) in the left indicate the impact of the bias related to visual side, where the salient "*elephant*" object leads to wrong answers. According to our statistics, most of the irrelevant wrong answer "*elephant*" appear in "*what*" type questions, while the biased answers might be different in the questions belonging to "*how many*" type, such as Figure 2 (c). Therefore, both language and visual modalities might jointly bring about bias.

- **Second, strong uncertainty exists in the final score, since the base model and bias model are optimized separately.** That means the model cannot guarantee that the correct answer has the highest score after subtracting the bias effect. For this reason, the model still suffers from inadequate debiasing and overcorrection problems, which has been shown in the right part of Figure 2.

To solve the above problems, we propose a **M**ulti-modal **D**ebiasing model with **D**ynamical **C**onstraint (MDDC). For the first limitation, we construct two bias learning branches. Inspired by the way of using the single modality to identify unimodal-specific bias, we adopt a question-only branch for language bias. Unfortunately, such a strategy is unsuitable for the bias issue related to visual information. The reason is that the same image is usually used to answer various types of questions, thus the model cannot obtain the specific vision bias that involves necessary information to answer the question, but only an image-to-answer distribution bias. We assume that a more effective way is to provide some question clues for images to generate question-specific vision bias. Following this assumption, we design a special bias learning branch by incorporating prompts extracted from questions into an image-only answering model.

For the second limitation, we propose a dynam-

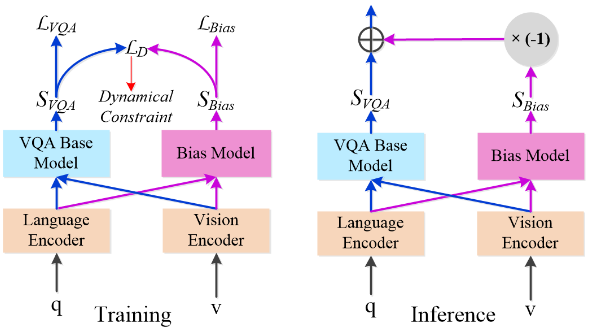

Figure 3: The overall framework of our debiasing model with dynamical constraint for VQA.

ical constraint loss to reduce the difference of the amount of information (Ramakrishnan et al., 2018) between the VQA base module and the bias module. In this way, we dynamically subtract the bias score according to the degree of bias. Therefore, we mitigate the problem of uncertain inference caused by the separate optimization of these two modules.

We evaluate the proposed MDDC on VQA v2.0 and VQA-CP v2.0 benchmarks. Experimental results on both datasets demonstrate that our debiasing strategy is competitive compared with mainstream baselines.

2 Related Work

Visual question answering has witnessed great progress, while a growing body of work (Agrawal et al., 2016; Goyal et al., 2017) has pointed out the drawbacks of reasoning ability and bias affect. In this section, we review recently proposed VQA debiasing approaches, which can be generally fall into non-augmentation-based methods and augmentation-based methods.

2.1 Non-augmentation-based Methods

One of the strategies is to introduce prior knowledge (i.e., human visual and textual explanations) to strengthen the visual grounding for VQA model. HINT (Selvaraju et al., 2019), SCR (Wu and Mooney, 2019) are proposed with a self-critical training objective that ensures the correct answers to match important image regions with the help of human explanations. Another common solution (Ramakrishnan et al., 2018; Cadene et al., 2019) is to design ensemble-based models, which adds an auxiliary QA model to identify bias. Ramakrishnan et al. (2018) propose an adversarial regularization method between the VQA base model and the question-only branch to overcome language bias. RUBi (Cadene et al., 2019) also leverages the QA model to capture language bias when unwanted regularities are identified. Wen et al. (2021) use both question-to-answer and vision-to-answer models to generate bias representations of two modalities. Niu et al. (2021) design a novel counterfactual inference framework to reduce language bias by subtracting the direct language effect from the VQA total causal effect. Guo et al. (2022) propose a loss re-scaling way to assign different weights to each answer according to the training data statistics.

2.2 Augmentation-based Methods

Recently, studies automatically generate additional question-image pairs to balance the distribution of training data. Chen et al. (2020) propose a method, CSS, to produce massive counterfactual samples by masking the critical objects and words. Mutant (Gokhale et al., 2020) generates the samples by semantic transformations of the original images or questions. Teney et al. (2020) and Zhu et al. (2020) obtain negative samples to balance the dataset without external annotations. Chen et al. (2022) design a knowledge distillation-based answer assignment to generate pseudo answers for each image-question pairs. However, it is important to note that the VQA-CP is proposed to evaluate whether the VQA model can distinguish between visual knowledge and language prior. Therefore, we expect that the model can be robust enough to make debiased inference under biased training.

3 Our Approach

In this section, we first describe the general architecture of our proposed MDDC model and then give the details for each component. Figure 3 depicts the overview of our approach, which consists of three major modules: (1) the standard VQA base module, which aims to indicate the probability belonging to each answer candidate; (2) the bias module, which aims to capture biases combining both questions and images simultaneously; (3) the dynamical constraint module, which aims to dynamically control the final prediction distribution.

3.1 Standard VQA Base Module

Given a dataset $\mathcal{D} = \{(v_i, q_i, a_i)\}_{i=1}^N$ which contains N samples, we define the i-th image $v_i \in \mathcal{V}$, the i-th question $q_i \in \mathcal{Q}$, and the i-th answer $a_i \in \mathcal{A}$. A standard VQA module is defined as:

$$p(a|v_i, q_i) = \sigma(f_{VQA}(e_v(v_i), e_q(q_i))) \quad (1)$$

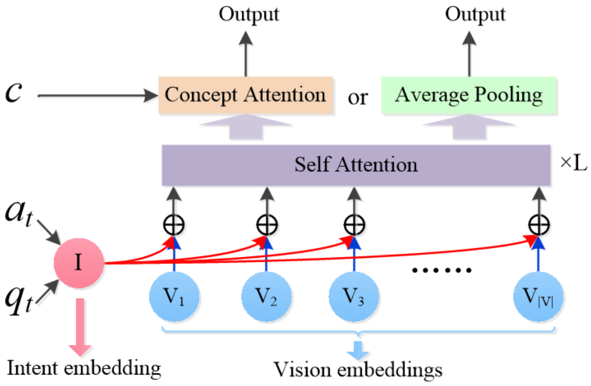

Figure 4: Question-guided vision bias module, where c, a_t, q_t represent the embeddings of concept, answer type and question type, respectively, V_i stands for the i-th region vector of the image.

where $e_v(\cdot)$ and $e_q(\cdot)$ denote image and question encoders, respectively, $f_{VQA}(\cdot)$ represents the mapping function which is learned to project the multi-modal feature to the answer space, $\sigma(\cdot)$ is the sigmoid function.

3.2 Bias Module

At the heart of our system is the design to obtain bias distributions. To make use of this intuition, we capture the language bias by using a QA model, as well as the vision bias by incorporating question clues into a vision-to-answer-only model.

3.2.1 Language Bias Learning

Language bias stands for the prior that produces the answer only according to the given question. For example, given a question q_i, we denote the language bias answer probability as:

$$p(a|q_i) = \sigma(f_Q(e_q(q_i))) \tag{2}$$

where $f_Q(\cdot)$ is a linear function to map the question representation to the answer space.

3.2.2 Question-guided Vision Bias Learning

We introduce a question-guided vision bias learning module for VQA debiasing, which is shown in Figure 4. Since merely using visual information is hard to obtain more targeted bias, a more flexible way is to guide images to generate answers with the intent and concepts of questions. The intent-level clues provide semantic enhancement on individual images, manifesting the goal of the question in a global view. Additionally, the concept-level clues can supplement more semantics to images, where the concept refers to a set of entities mentioned in the question. Here, we compute answer probability predicted by the question-guided vision bias module as follows:

$$p(a|v_i, a_t, q_t, c) = \sigma(f_F(e_v(v_i), a_t, q_t, c)) \tag{3}$$

where a_t, q_t and c stand for the answer type, question type and concepts of the question, respectively, $f_F(\cdot)$ is the function to combine these components and map the fusion representation to the answer space. Concretely, we fuse a_t and q_t via a gate mechanism to obtain the question intent vector which is later added with each image region embedding. Then, a multi-layer self-attention (Vaswani et al., 2017) is adopted to make interactive learning for the image features incorporated with intent clues. Finally, we get the vision bias output via a concept attention or average pooling operation. Note that the concept attention is a normal attention mechanism using c as the query to weight the image regions. However, we assume that not all questions are suitable for using concepts. For example, as for the *other* type question "$What\ color\ is\ the\ apple?$", it might be easy to answer "*red*" if the concept "*apple*" and intent "*what color*" are provided. But for the $number$ type questions, they are still hard to be answered even though given the intent and concept. Thus we only apply concept attention to $number$ type questions, and employ average pooling operation on yes/no and *other* type questions.

3.3 Dynamical Constraint

As mentioned above, it is necessary to build connection between the standard VQA base module and bias module. In this subsection, we introduce a dynamical constraint loss $\mathcal{L}_D$ to control the final distribution subtracted by the bias probability. Denote $\mathcal{B} = \{b_1, \ldots, b_M\}$ as the set of features extracted from M bias modules. We define s as the feature outputted from the VQA base module. Afterwards, $\mathcal{L}_D$ is computed as:

$$\mathcal{L}_D = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{A}\sum_{k=1}^{M}\beta_{ij}(I(a_j|s)_i - I(a_j|b_k)_i) \tag{4}$$

$$\beta_{ij} = p(a_j|s)_i \tag{5}$$

where N, A are the number of samples and the number of candidate answers, respectively, β is a dynamic control coefficient, $I(X|Y)$ represents the amount of information of X under the condition of Y. The goal of $\mathcal{L}_D$ is to decrease the uncertainty of the standard VQA module prediction and increase the probability uncertainty of the bias module according to the degree of bias. As for the former, it helps the VQA base module learn adequate knowledge disrupted by bias. As for the latter, it prompts the bias module to compute the appropriate bias score, for that different samples have varying degrees of impact from bias. Note that both the VQA probability score $p(a_j|s)$ and the bias probability score $p(a_j|b)$ satisfy the Bernoulli distribution since the sigmoid function is applied to the final output layer. Therefore, $\mathcal{L}_D$ is different from the Kullback-Leibler divergence (Doersch, 2016). More details are explained in Appendix A.

3.4 Training and Inference

Training. In the model training phase, we separately optimize the standard VQA module and bias module via the binary cross-entropy loss $bce(\cdot)$, which is defined as:

$$\mathcal{L}_B = bce(p(a|\mathbf{s}), y) + w \sum_{k=1}^{M} bce(p(a|b_k), y) \quad (6)$$

where w is a hyper-parameter to balance the base and bias components, and y is the target label. Then, the final loss function is computed as $\mathcal{L} = \mathcal{L}_B + \lambda \mathcal{L}_D$, where λ is the discount coefficient. Additionally, we stop the gradient backpropagation of the bias module to the language encoder and vision encoder in order to prevent the VQA base module from updating in a biased direction.

Inference. At the inference stage, the final score for the j-th answer $\Delta p(a_j)$ is distinct according to different answer types, which is defined as:

$$\Delta p(a_j)^t = p(a_j|s) - \sum_{k=1}^{M} \alpha_k^t p(a_j|b_k) \quad (7)$$

where t is the answer type (e.g., yes/no, $number$, and $other$), and α^t stands for the weight of t, which satisfies the condition of $\sum_{k=1}^{M} \alpha_k^t = 1$.

4 Experiment

4.1 Experimental Settings

Dataset. We conduct experiments on VQA-CP v2.0 dataset (Agrawal et al., 2018), which is proposed to evaluate the debiasing ability. Besides, we also validate the performance on VQA v2.0 (Goyal et al., 2017), to see the generalization ability on ID dataset. For both datasets, the questions are divided into three categories: yes/no, $number$ and $other$.

Base	Parameter	VQA-CP v2.0	VQA v2.0
Updn	lr	5e-4	5e-4
	batch size	256	256
	epoch	25	25
	$\alpha_1^t = \{\alpha^y, \alpha^n, \alpha^o\}$	{0.99, 0.01, 0.5}	{0.5, 0.5, 0.5}
	$\alpha_2^t = \{\alpha^y, \alpha^n, \alpha^o\}$	{0.01, 0.99, 0.5}	{0.5, 0.5, 0.5}
LXMERT	lr	5e-5	5e-5
	batch size	32	32
	epoch	10	10
	$\alpha_1^t = \{\alpha^y, \alpha^n, \alpha^o\}$	{0.99, 0.01, 0.5}	{0.5, 0.5, 0.5}
	$\alpha_2^t = \{\alpha^y, \alpha^n, \alpha^o\}$	{0.01, 0.99, 0.5}	{0.5, 0.5, 0.5}

Table 2: Important hyper-parameters list, where $a_k^t = \{a_1^t, a_2^t\}$ stands for the weight combinations of the language bias branch (i.e., a_1^t) and the question-guided vision bias branch (i.e., a_2^t) in Equation (7); α^y, α^n, α^o are severally denoted as the weights of the three question types, namely yes/no, $number$ and $other$.

Metric. Following previous work (Antol et al., 2015), the standard evaluation metric in VQA challenge is adopted, which is computed as:

$$Acc(ans) = min\left(1, \frac{\#humans\ provided\ ans}{3}\right) \quad (8)$$

where the $humans\ provided\ ans$ is the number of each answer that human annotated for question.

Hyper-Parameters and Environment. Optimal hyper-parameters are chosen via grid search. All the embeddings of question clues are randomly initialized. The intent extraction model is trained by fine-tuning BERT (Devlin et al., 2019), and the concepts are extracted by entity recognition tool. We use the Pytorch 1.40 framework to implement our model. All computations are done on NVIDIA Tesla V100 GPUs. Other important hyper-parameters are listed in Table 2.

4.2 Tested Backbones

We mainly implement our approach on two VQA backbones, namely Updn (Anderson et al., 2018) and LXMERT (Tan and Bansal, 2019).

Updn. the most popular VQA baseline, which firstly employs the pre-trained object detection model (Ren et al., 2015) to obtain features of salient image regions.

LXMERT. a multi-modal pre-training framework based on a cross-modality encoder from Transformers. In our experiments, we separately divide this backbone into two groups depending on whether loading pre-trained weights or not.

4.3 Baselines

We compare our model with existing mainstream bias reduction techniques, which can be grouped as

Models	Base	VQA-CP v2.0 test				VQA v2.0 val			
		All	Y/N	Num.	Other	All	Y/N	Num.	Other
SAN (Yang et al., 2016)	-	26.88	38.35	11.96	42.98	52.41	70.06	39.28	47.84
GVQA (Agrawal et al., 2018)	-	39.23	57.99	13.68	22.14	48.24	72.03	31.17	34.65
S-MRL (Cadene et al., 2019)	-	38.46	42.85	12.81	43.20	63.10	-	-	-
Updn (Anderson et al., 2018)	-	39.74	42.27	11.93	46.05	63.48	81.18	42.14	55.66
Updn † (Anderson et al., 2018)	-	39.80	41.39	12.10	46.56	**64.36**	82.02	43.31	56.49
AReg (Ramakrishnan et al., 2018)	Updn	41.17	65.49	15.48	35.48	62.75	79.84	42.35	55.16
GRL (Grand and Belinkov, 2019)	Updn	42.33	59.74	14.78	40.76	63.27	-	-	-
SCR (Wu and Mooney, 2019)	Updn	48.47	70.41	10.42	47.29	62.30	77.40	40.90	**56.50**
AttAlign (Selvaraju et al., 2019)	Updn	39.37	43.02	11.89	45.00	63.24	80.99	42.55	55.22
HINT (Selvaraju et al., 2019)	Updn	46.73	70.04	10.68	46.31	63.38	81.18	42.99	55.56
DLR (Jing et al., 2020)	Updn	48.87	70.99	18.72	45.57	57.96	76.82	39.33	48.54
RUBi (Cadene et al., 2019)	Updn	44.23	67.05	17.48	39.61	-	-	-	-
LM (Clark et al., 2019)	Updn	48.78	72.78	14.61	45.58	63.26	81.16	42.22	55.22
LMH (Clark et al., 2019)	Updn	52.73	72.95	**31.90**	47.79	56.35	65.06	37.63	54.69
CSS (Chen et al., 2020)	Updn	41.16	43.96	12.78	47.48	59.21	72.97	40.00	55.13
CF-VQA (HM) (Niu et al., 2021)	Updn	49.74	74.81	18.46	45.19	63.73	82.15	**44.29**	54.86
CF-VQA (SUM) (Niu et al., 2021)	Updn	53.55	**91.15**	13.03	44.97	63.54	**82.51**	43.96	54.30
Re-scaling (Guo et al., 2022)	Updn	47.09	68.42	21.71	42.88	55.50	64.22	39.61	53.09
MDDC (Ours)	Updn	**54.70**	83.58	19.93	**49.10**	63.33	81.64	42.56	54.88
LXMERT*† (Tan and Bansal, 2019)	-	40.91	41.91	13.71	47.85	**65.32**	**83.13**	**46.51**	**56.75**
MDDC (Ours)	LXMERT	**53.83**	**76.73**	**26.07**	**49.44**	64.03	82.15	45.64	55.10
LXMERT † (Tan and Bansal, 2019)	-	57.11	54.97	38.34	63.38	**75.87**	**91.46**	**59.61**	**68.31**
MDDC (Ours)	LXMERT	**69.77**	**87.88**	**52.80**	**64.93**	74.51	90.14	58.81	66.76

Table 3: Summary of results on VQA-CP v2.0 and VQA v2.0 datasets. † denotes our implementation, and ∗ stands for using the LXMERT model structure without loading multi-modal pre-trained weights. The best score is in **bold** and the second best is underlined.

Models	VQA-CP v2.0 test					VQA v2.0 val				
	All	Y/N	Num.	Other	ΔGap	All	Y/N	Num.	Other	ΔGap
Updn †	39.80	41.39	12.10	46.56	-	**64.36**	**82.02**	**43.31**	**56.49**	-
+ b_l	51.49	76.38	13.93	48.74	+ 11.69	62.41	81.17	42.47	53.40	- 1.95
+ b_l + $\mathcal{L}_D$	54.10	**84.27**	14.22	**49.23**	+ 14.30	62.65	81.40	42.92	53.61	- 1.71
+ b_l + b_v	52.15	77.00	16.61	48.87	+ 12.35	63.04	81.60	41.24	54.69	- 1.32
+ b_l + b_v + $\mathcal{L}_D$	**54.70**	83.58	**19.93**	49.10	**+ 14.90**	63.33	81.64	42.56	54.88	- **1.03**
LXMERT*	40.91	41.91	13.71	47.85	-	**65.32**	**83.13**	**46.51**	**56.75**	-
+ b_l	50.80	69.93	19.49	49.35	+ 9.89	62.98	81.86	45.53	53.22	- 2.34
+ b_l + $\mathcal{L}_D$	51.48	71.82	19.57	**49.56**	+ 10.57	63.17	81.69	45.60	53.73	- 2.15
+ b_l + b_v	52.41	72.94	25.63	48.98	+ 11.50	64.08	82.24	45.51	55.17	- **1.24**
+ b_l + b_v + $\mathcal{L}_D$	**53.83**	**76.73**	**26.07**	49.44	**+ 12.92**	64.03	82.15	45.64	55.10	- 1.29
LXMERT	57.11	54.97	38.34	63.38	-	**75.87**	**91.46**	**59.61**	**68.31**	-
+ b_l	69.09	87.42	52.73	63.96	+ 11.98	73.85	89.66	58.65	65.85	- 2.02
+ b_l + $\mathcal{L}_D$	69.30	**87.99**	50.85	64.55	+ 12.19	73.84	89.84	58.53	65.72	- 2.03
+ b_l + b_v	69.54	87.41	51.34	**65.15**	+ 12.43	74.62	90.34	59.04	66.77	- **1.25**
+ b_l + b_v + $\mathcal{L}_D$	**69.77**	87.88	**52.80**	64.93	**+ 12.66**	74.51	90.14	58.81	66.76	- 1.36

Table 4: Ablation study on VQA-CP v2.0 and VQA v2.0 Datasets. † denotes our implementation, and ∗ stands for using the LXMERT model structure without loading multi-modal pre-trained weights. The best score is in **bold** and the second best is underlined.

follows: (1) Methods incorporating human visual or textual explanation, including SCR (Agrawal et al., 2018), AttAlign (Selvaraju et al., 2019) and HINT (Selvaraju et al., 2019). (2) Adversarial regularization-based methods, including AReg (Ramakrishnan et al., 2018) and GRL (Grand and Belinkov, 2019). (3) Ensemble-based methods, including RUBi (Cadene et al., 2019), LM (Clark et al., 2019), LMH (Clark et al., 2019), Re-scaling (Guo et al., 2022). (4) Question encoding-based method DLR (Jing et al., 2020). (5) Counterfactual-based methods, including CF-VQA (Niu et al., 2021), CSS (Chen et al., 2020). In subsequent part, all the experimental results for the compared baselines are taken from their original papers.

4.4 Results

The results on VQA-CP v2.0 and VQA v2.0 are reported in Table 3.

Results on VQA-CP v2.0. Overall, our method achieves the best performance on VQA-CP v2.0 dataset compared with non-augmentation approaches. Drilling down to the question type, our method also gets competitive results. Specifically, we achieve the second-best results on yes/no type question, and the best results on $other$ type question. It is worth noting that our strategy obtains improvements of 6.90% and 4.13% across $number$ type and $other$ type question compared with CF-VQA (Niu et al., 2021) which also employs a subtracting way to reduce bias effect. We infer the reason is that our model detects more comprehensive biases from both language and vision aspects, and our dynamical constraint loss also plays a role in adjusting the final distribution.

Results on VQA v2.0. In consistent with what metioned in (Agrawal et al., 2018; Selvaraju et al., 2019; Ramakrishnan et al., 2018; Cadene et al., 2019; Chen et al., 2020; Niu et al., 2021) , we usually observe a drop after debiasing on VQA v2.0 because of the almost consistent distribution followed by training and test datasets. As a comparison, our debiasing strategy demonstrates strong robustness and achieves competitive results.

To sum up, these results not only show the effectiveness of our approach for reducing bias problem but also the value of the performance on ID dataset.

5 Analysis

5.1 Ablation Study

An ablation experiment would be informative to analyze the effects of the dynamical constraint loss (denoted as + $\mathcal{L}_D$), and the bias learning strategy, which can be taken apart as language bias learning (denoted as + b_l), and question-guided vision bias learning (denoted as + b_v). For fairness, all the models are trained under the same settings.

Table 4 lists the results on two datasets. It can be seen that coupling with all the components does really helpful on VQA-CP v2.0 (Agrawal et al., 2018), and can narrow the drop gap on VQA v2.0 (Goyal et al., 2017). When there is no multi-modal pre-trained knowledge (e.g., Updn (Anderson et al., 2018) and LXMERT* (Tan and Bansal, 2019)), on OOD dataset (i.e., VQA-CP v2.0), we find + b_v brings significant improvement on $number$ type question. A possible reason might be that the bias in $number$ type question is severely affected by images with language information (e.g., intent and concept) on VQA-CP v2.0. When leveraging pre-trained weights into LXMERT, there are still slight improvements brought by all the components. On the whole, our bias learning strategy (+ b_l + b_v) can detect more comprehensive biases than individual + b_l, and it narrows the performance gap on ID dataset (i.e., VQA v2.0). Fortunately, integrating the dynamical constraint improves the result across all base models on OOD dataset, and $\mathcal{L}_D$ does not have a significant negative impact on ID dataset.

Image	Intent	Concept	All	Y/N	Num.	Other
✓			52.85	85.13	12.44	47.02
✓	✓		53.65	**86.24**	12.27	47.91
✓		✓	53.10	86.01	12.46	47.00
	✓		53.53	84.94	12.83	48.23
✓	✓	✓	**54.70**	83.58	**19.93**	**49.10**

Table 5: The effects caused by different components in our question-guided vision bias learning module.

To conclude, it is always preferable to use all the components (+ b_l + b_v + $\mathcal{L}_D$), due to the superior performance. This proves the effectiveness of our bias learning strategy and dynamical constraint.

5.2 Vision Bias Learning Analysis

5.2.1 Impact of Question Clues

In the following set of experiments, we demonstrate the effectiveness of the question clues mentioned in our question-guided vision bias learning module on VQA-CP v2.0. Note that we only change three components (i.e., image, intent, concept) based on the overall Updn + MDDC model. As depicted in Table 5, we conclude that both the image and the question clues are necessary for debiasing. Concretely, leveraging intent feature is helpful for $other$ type question, based on which incorporating concept information via a concept-attention mechanism boosts the performance of $number$ type question from 12.27% to 19.93%. Such a phenomena indicates that the base model Updn might easily overfit the training set of VQA-CP v2.0 dataset and learn less valid knowledge for number recognition ability.

5.2.2 Impact of Layer Number

We further investigate the layer number of self-attention (Vaswani et al., 2017) in question-guided

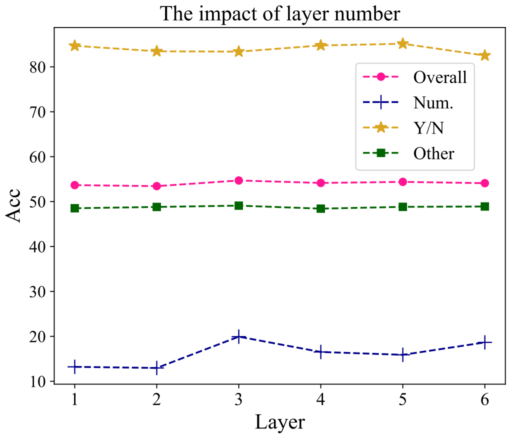

Figure 5: Impact under different layer numbers of self attention in question-guided vision bias learning branch.

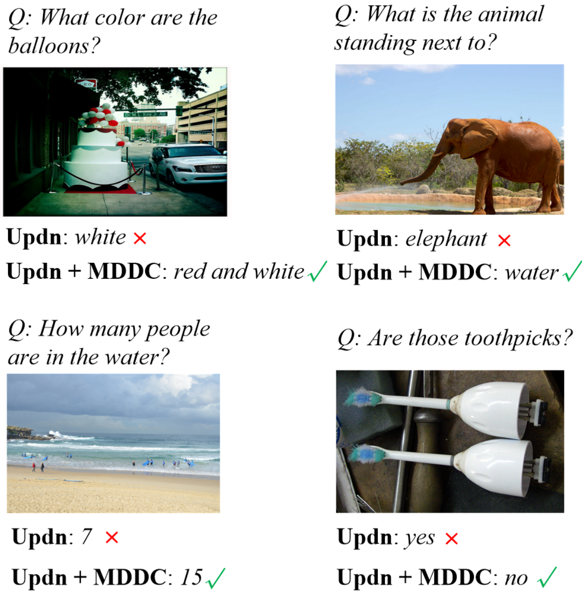
Q: What color are the balloons?
Updn: white ×
Updn + MDDC: red and white ✓
Q: What is the animal standing next to?
Updn: elephant ×
Updn + MDDC: water ✓
Q: How many people are in the water?
Updn: 7 ×
Updn + MDDC: 15 ✓
Q: Are those toothpicks?
Updn: yes ×
Updn + MDDC: no ✓

Figure 6: Qualitative comparison on VQA-CP v2.0.

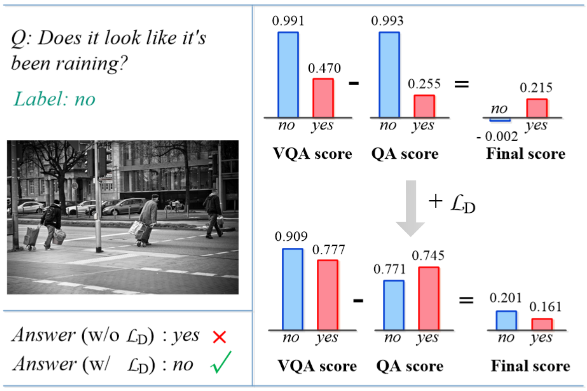

Figure 7: An example on VQA-CP v2.0 test split to verify the effectiveness of our dynamical constraint loss, when merely using language bias learning module. More examples are shown in Appendix B

vision bias learning module on VQA-CP v2.0 test split. Figure 5 shows the change of accuracy on the test set as the layer number increases, which is based on Updn model. A proper number of layers can make the model perform well on *number* type questions, which again verifies the effect of our vision bias learning strategy, while accuracies on the rest items are more stable. We find that the best results can be obtained when the number is equal to 3, and further numbers do not provide a significant performance improvement.

5.3 Qualitative Analysis

Debiasing qualitative examples on VQA-CP v2.0 are shown in Figure 6. By inspecting the results, we can further verify that our debiasing approach can address more comprehensive biases and dynamically adjust the final score. As illustrated in Figure 6, MDDC can successfully mitigate the biasd inference on all kinds of question types. For the example at the first row, MDDC overcomes the bias related to both question and image sides (i.e., "*white*", "*elephant*"). The two examples at the second row manifest the effects of MDDC on *number* and *yes/no* type questions. Another example based on Updn model in Figure 7 further illustrates the benefit brought by the dynamical constraint loss $\mathcal{L}_D$. Specifically, $\mathcal{L}_D$ helps to increase the difference between the VQA score and the QA score corresponding to the answer of "*no*", and it narrows the score gap of the wrong answer (i.e., "*yes*"), which promotes the final score of the correct answer to be the highest.

6 Conclusion

A robust visual question answering model with dynamical constraint is proposed for reducing as much multi-modal bias as possible. Compared with previous researches, we investigate a very straightforward way to obtain debiasing effect by subtracting bias score from VQA base score. On one hand, we design a language bias learning branch and a question-guided vision bias learning branch to detect comprehensive biases. On the other hand, a dynamical constraint loss is proposed related to the two bias branches to alleviate the over-correction and insufficient debiasing problems to some extent. Experimental results on VQA-CP v2.0 and VQA v2.0 datasets demonstrate the effectiveness of our proposed approach from both quantitative and qualitative perspectives.

Limitations

Our model introduces additional parameters in the question-guided vision bias module, compared with other methods. Moreover it is also worth exploring whether the question-guided vision bias module can improve *number* type questions in other OOD data sets.

Acknowledgements

The research work descried in this paper has been supported by the National Key R&D Program of China (2020AAA0108001) and the National Nature Science Foundation of China (No. 61976015, 61976016, 61876198 and 61370130). The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve this paper.

References

Aishwarya Agrawal, Dhruv Batra, and Devi Parikh. 2016. Analyzing the behavior of visual question answering models. *arXiv preprint arXiv:1606.07356*.

Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. 2018. Don't just assume; look and answer: Overcoming priors for visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4971–4980.

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6077–6086.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2425–2433.

Remi Cadene, Corentin Dancette, Matthieu Cord, Devi Parikh, et al. 2019. Rubi: Reducing unimodal biases for visual question answering. *Advances in Neural Information Processing Systems*, 32:841–852.

Long Chen, Xin Yan, Jun Xiao, Hanwang Zhang, Shiliang Pu, and Yueting Zhuang. 2020. Counterfactual samples synthesizing for robust visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10800–10809.

Long Chen, Yuhang Zheng, and Jun Xiao. 2022. Rethinking data augmentation for robust visual question answering. In *Proceedings of the European Conference on Computer Vision*, pages 95–112. Springer.

Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. 2019. Don't take the easy way out: Ensemble based methods for avoiding known dataset biases. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4069–4082, Hong Kong, China. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Carl Doersch. 2016. Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908*.

Tejas Gokhale, Pratyay Banerjee, Chitta Baral, and Yezhou Yang. 2020. Mutant: A training paradigm for out-of-distribution generalization in visual question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 878–892, Online. Association for Computational Linguistics.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6904–6913.

Gabriel Grand and Yonatan Belinkov. 2019. Adversarial regularization for visual question answering: Strengths, shortcomings, and side effects. In *Proceedings of the Second Workshop on Shortcomings in Vision and Language*, pages 1–13. Association for Computational Linguistics.

Yangyang Guo, Liqiang Nie, Zhiyong Cheng, Qi Tian, and Min Zhang. 2022. Loss re-scaling VQA: revisiting the language prior problem from a class-imbalance view. *IEEE Transactions on Image Processing*, 31:227–238.

Chenchen Jing, Yuwei Wu, Xiaoxun Zhang, Yunde Jia, and Qi Wu. 2020. Overcoming language priors in vqa via decomposed linguistic representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11181–11188.

Zujie Liang, Weitao Jiang, Haifeng Hu, and Jiaying Zhu. 2020. Learning to contrast the counterfactual samples for robust visual question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3285–3292, Online. Association for Computational Linguistics.

Yulei Niu, Kaihua Tang, Hanwang Zhang, Zhiwu Lu, Xian-Sheng Hua, and Ji-Rong Wen. 2021. Counterfactual vqa: A cause-effect look at language bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12700–12710.

Yulei Niu and Hanwang Zhang. 2021. Introspective distillation for robust question answering. *Advances in Neural Information Processing Systems*, 34:16292–16304.

Sainandan Ramakrishnan, Aishwarya Agrawal, and Stefan Lee. 2018. Overcoming language priors in visual question answering with adversarial regularization. *Advances in Neural Information Processing Systems*, 31:1541–1551.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, 28.

Ramprasaath R Selvaraju, Stefan Lee, Yilin Shen, Hongxia Jin, Shalini Ghosh, Larry Heck, Dhruv Batra, and Devi Parikh. 2019. Taking a hint: Leveraging explanations to make vision and language models more grounded. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2591–2600.

Hao Tan and Mohit Bansal. 2019. Lxmert: Learning cross-modality encoder representations from transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5100–5111. Association for Computational Linguistics.

Damien Teney, Ehsan Abbasnejad, Kushal Kafle, Robik Shrestha, Christopher Kanan, and Anton Van Den Hengel. 2020. On the value of out-of-distribution testing: An example of goodhart's law. *Advances in Neural Information Processing Systems*, 33:407–417.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

Zhiquan Wen, Guanghui Xu, Mingkui Tan, Qingyao Wu, and Qi Wu. 2021. Debiased visual question answering from feature and sample perspectives. *Advances in Neural Information Processing Systems*, 34:3784–3796.

Jialin Wu and Raymond Mooney. 2019. Self-critical reasoning for robust visual question answering. *Advances in Neural Information Processing Systems*, 32.

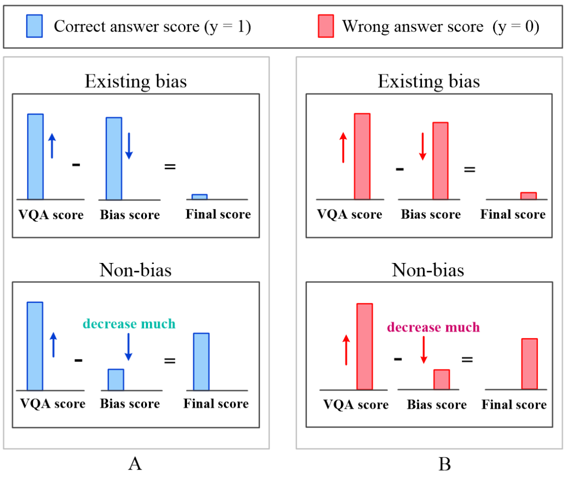

Figure 8: An explanation for our dynamical constraint during training procedure. Arrows (blue and red) indicate the upward and downward trends of probability scores after using dynamic constraint.

Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. 2016. Stacked attention networks for image question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 21–29.

Xi Zhu, Zhendong Mao, Chunxiao Liu, Peng Zhang, Bin Wang, and Yongdong Zhang. 2020. Overcoming language priors with self-supervised learning for visual question answering. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 1083–1089. International Joint Conferences on Artificial Intelligence Organization.

A A Theoretical Explanation

This section presents an approximately feasible theoretical explanation for our dynamical constraint $\mathcal{L}_D$. According to the definition, the total loss $\mathcal{L}$ ($\mathcal{L} = \mathcal{L}_B + \lambda\mathcal{L}_D$) can be combined like items and further simplified. For easier explanation, we extract the loss item of answer $a_i \in \mathcal{A}$ from a single sample, namely $\mathcal{L}(a_i)$ to illustrate, which is computed as:

$$\begin{aligned} \mathcal{L}(a_i) = &-(y_i + \lambda\beta_i)\log p(a_i|s) \\ &-(1-y_i)\log\left(1-p(a_i|s)\right) \\ &-(wy_i - \lambda\beta_i)\log p(a_i|b) \\ &-w(1-y_i)\log\left(1-p(a_i|b)\right) \end{aligned} \tag{9}$$

We assume the bias can be reflected as: the score of a common answer is extremely high or too low in the training phase, which affects the selection of the correct answer when evaluating on test set.

Here, we consider two boundary cases, depending on whether the target label of the current answer is 1 or 0 (i.e., $y = 1$ or $y = 0$).

If $y_i = 1$, both the VQA score $p(a_i|s)$ and the bias score $p(a_i|b)$ are optimized to 1. On this condition, $\mathcal{L}(a_i)$ can be transformed to:

$$\begin{aligned}\mathcal{L}(a_i) = &- \overbrace{(1 \underbrace{+\lambda\beta_i}_{\text{boost}}) \log p(a_i|s)}^{\text{VQA model learning}} \\ &- \underbrace{(w \overbrace{-\lambda\beta_i}^{\text{inhibition}}) \log p(a_i|b)}_{\text{bias learning}}\end{aligned} \tag{10}$$

Since $\beta_i = p(a_i|s)$, when $\beta_i \rightarrow 1$, the learning procedure of VQA base model is further boosted while the bias learning is inhibited. Due to the extremely long-tailed answer distribution, the training objective can be unbalanced across different answers. It indicates that the unbalanced bias knowledge might be overlearned from more samples during training. Thus, if the sample is severely biased, the bias score tends not to decrease much after suppression, and the VQA base score will be relatively less reserved after subtracting (as shown in Figure 8 A).

If $y_i = 0$, both $p(a_i|s)$ and $p(a_i|b)$ are optimized to 0, and the $\mathcal{L}(a_i)$ can be reduced to:

$$\begin{aligned}\mathcal{L}(a_i) = &\overbrace{\underbrace{-\lambda\beta_i \log p(a_i|s)}_{\text{inhibition}} - \log(1 - p(a_i|s))}^{\text{VQA model learning}} \\ &\underbrace{\overbrace{+\lambda\beta_i \log p(a_i|b)}^{\text{boost}} - w \log(1 - p(a_i|b))}_{\text{bias learning}}\end{aligned} \tag{11}$$

Intuitively, the term $-\lambda\beta_i \log p(a_i|s)$ inhibits $p(a_i|s) \rightarrow 0$, thus it can prevent the model from overfitting the training set to some extent. In addition, the item $+\lambda\beta_i \log p(a_i|b)$ boosts $p(a_i|b) \rightarrow 0$. Therefore, when the VQA base score of a wrong answer is high (i.e., B), the process of adjusting the prediction scores is similar to the condition of $y = 1$ (as shown in Figure 8 B).

In this way, during inference procedure, the final score of the biased answer might be more likely to decrease, while the unbiased answer tends to retain a relatively higher final score. In summary, such a strategy can help to prevent the model from overfitting the training set, and dynamically obtain a more appropriate final score.

B Illustrative Examples

In order to fully demonstrate the specific role of the dynamic constraint loss, we deliver more illustrative examples to show the probability predictions for each branch, as shown in Figure 9. We choose Updn as the backbone, and all the results are obtained under the same experimental settings.

For the cases in Figure 9, we find that when $\mathcal{L}_D$ is not added (w/o $\mathcal{L}_D$), strong uncertainty exists in the prediction results. The reason is that the VQA branch and the bias branches are trained separately, causing debiasing effect to be less significant in certain cases. By contrast, we explicitly introduce $\mathcal{L}_D$ to the model (w/ $\mathcal{L}_D$) and thus obtain satisfactory results.

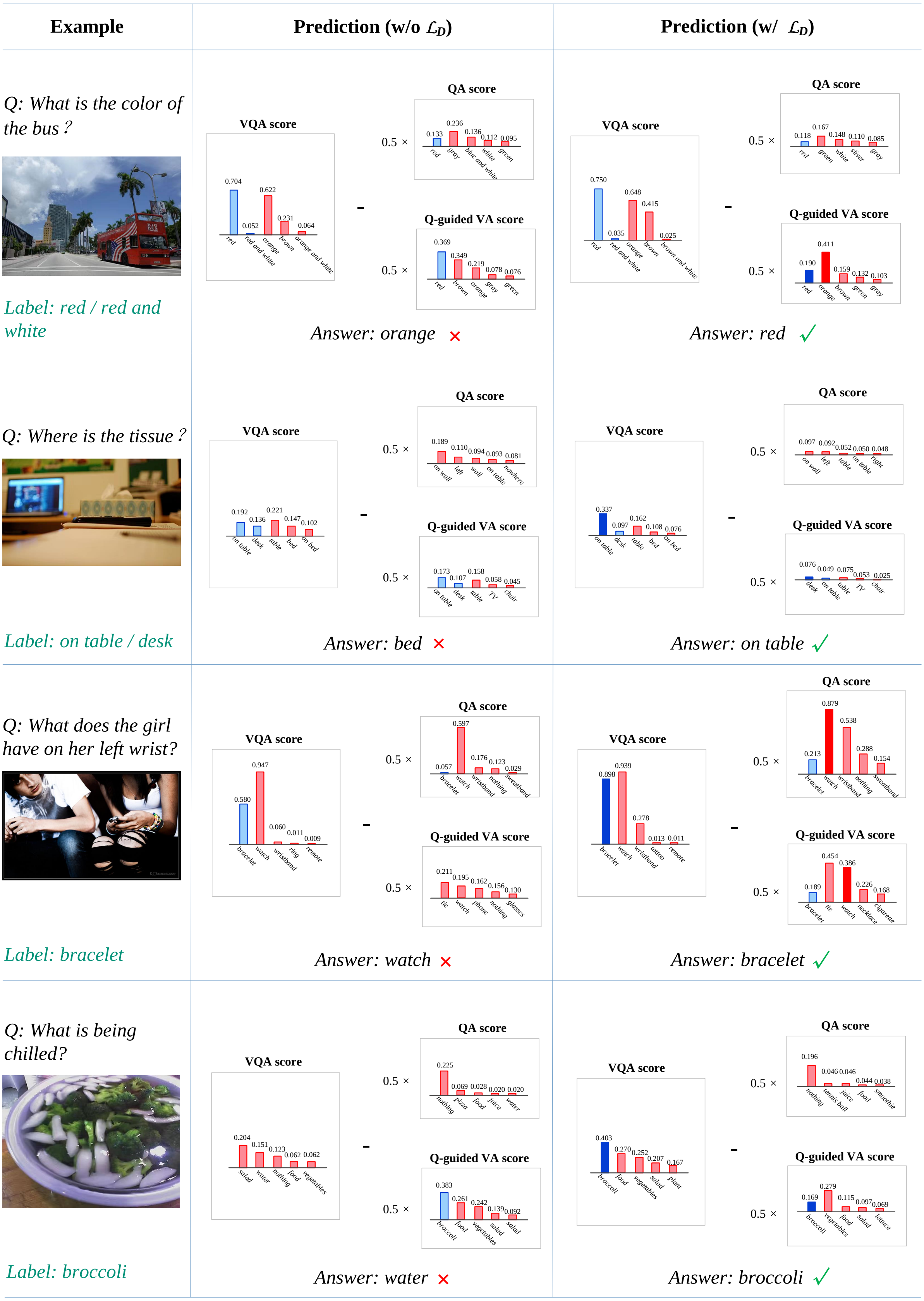

Figure 9: Examples on VQA-CP v2.0 test split to verify the effectiveness of the dynamical constraint loss, specifically for the standard VQA score, language bias score (i.e., QA score) and question-guided vision bias score (i.e., Q-guided VA score). In the top five rankings, the correct answer scores (in red) and incorrect answer scores (in blue) are given, where scores marked in dark red or dark blue stand for the scores corresponding to the answers that are changed more crucially after incorporating $\mathcal{L}_D$.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
the section: Limitation

☑ A2. Did you discuss any potential risks of your work?
the section: Limitation

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract; Section 1: Introduction

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☒ Did you use or create scientific artifacts?

Left blank.

☒ B1. Did you cite the creators of artifacts you used?
Left blank.

☒ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Left blank.

☒ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Left blank.

☒ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Left blank.

☒ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Left blank.

☒ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Left blank.

C ☑ Did you run computational experiments?

Section 4: Experiment

☒ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Limited by number of pages

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 4: Experiment

☒ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Limited by number of pages

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Section 4: Experiment

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☒ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Left blank.

☒ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Left blank.

☒ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Left blank.

☒ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Left blank.

☒ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Left blank.