An Empirical Study on the Language Modal in Visual Question Answering

Daowan Peng\textsuperscript{1,2}, Wei Wei\textsuperscript{1,2}, Xian-Ling Mao\textsuperscript{3}, Yuanyuan Fu\textsuperscript{2,4} and Dangyang Chen\textsuperscript{2,4}

\textsuperscript{1}Cognitive Computing and Intelligent Information Processing (CCIIIP) Laboratory, School of Computer Science and Technology, Huazhong University of Science and Technology
\textsuperscript{2}Joint Laboratory of HUST and Pingan Property & Casualty Research (HPL)
\textsuperscript{3}Department of Computer Science and Technology, Beijing Institute of Technology
\textsuperscript{4}Ping An Property & Casualty Insurance Company of China, Ltd
\{pengdw, weiw\}@hust.edu.cn, maoxl@bit.edu.cn, fuyuanyuan83@gmail.com, chendangyang273@pingan.com.cn

Abstract

Generalization beyond in-domain experience to out-of-distribution data is of paramount significance in the AI domain. Of late, state-of-the-art Visual Question Answering (VQA) models have shown impressive performance on in-domain data, partially due to the language priors bias which, however, hinders the generalization ability in practice. This paper attempts to provide new insights into the influence of language modality on VQA performance from an empirical study perspective. To achieve this, we conducted a series of experiments on six models. The results of these experiments revealed that, 1) apart from prior bias caused by question types, there is a notable influence of postfix-related bias in inducing biases, and 2) training VQA models with word-sequence-related variant questions demonstrated improved performance on the out-of-distribution benchmark, and the LXMERT even achieved a 10-point gain without adopting any debiasing methods. We delved into the underlying reasons behind these experimental results and put forward some simple proposals to reduce the models’ dependency on language priors. The experimental results demonstrated the effectiveness of our proposed method in improving performance on the out-of-distribution benchmark, VQA-CPv2. We hope this study can inspire novel insights for future research on designing bias-reduction approaches.

1 Introduction

Visuo-linguistic understanding is an important research topic in the field of multimodal machine learning. Vision Language (V+L) tasks, such as image caption [Karpathy and Fei-Fei, 2015], referring expression comprehension [Yu et al., 2016], natural language for visual reasoning [Suhr et al., 2017], visual entailment [Xie et al., 2018], visual commonsense reasoning [Zellers et al., 2019], and visual question answering (VQA) [Antol et al., 2015; Anderson et al., 2018], serve as proxy tasks for evaluating the capacity of a multi-modal system to achieve high-level multimodal learning and deeper visuo-linguistic understanding. This paper specifically focuses on the VQA task, which has been a long-standing challenge in the domains of computer vision and natural language processing. Previous research has shown that many state-of-the-art VQA models tend to rely excessively on easily learnable language priors instead of effectively reasoning based on the visual content within the images during training [Goyal et al., 2017; Jing et al., 2020; Wen et al., 2021]. As a result, these VQA models can achieve decent performance on in-distribution data by capturing superficial correlations in the language modality. However, over-reliance on language priors makes these models fragile and results in poor performance on out-of-distribution (OOD) data in real-world scenarios.

Recently, a broad variety of bias-reduction methods [Cadene et al., 2019; Clark et al., 2019; Liang et al., 2021; Han et al., 2021; Yulei et al., 2021] have been proposed, among which the commonly used approach involves adding a branch to capture language bias. For example, [Ramakrishnan et al., 2018] trained a base VQA model along with a question-only adversary to mitigate bias representation by allowing the question-only model to perform poorly. However, some of these methods may introduce extra costs during the inference stage. Besides, [Yulei et al., 2021] proposed...
a novel counterfactual inference framework based on causal effects. [Han et al., 2021] introduced a greedy gradient ensemble de-bias framework, where the bias model is forced to overfit the biased data distribution, allowing the base model to learn the general patterns. Apart from the model design side, various methods from the data end have been developed to reduce language priors bias. For instance, HINT [Selvaraju et al., 2019] and SCR [Wu and Mooney, 2019] utilize additional annotated data to enhance models’ visual-grounding capacity for better performance 1. CSS [Chen et al., 2020a] generated counterfactual samples by masking the decisive word in the question or crucial object in the image. [Liang et al., 2020] further improved the CSS method by employing contrastive learning to focus on the crucial elements. Thanks to the previous research on debiasing, some progress has been made in addressing the issue of language priors. However, in this paper, we aim to provide novel insights regarding the impact of language modality on performance in VQA tasks through empirical investigations. In this regard, we conducted a series of confirmatory experimental analysis to investigate prior bias issues. The empirical evidence revealed that, in comparison to the co-occurrence between question types and answers, the co-occurrence between objects and answers could potentially be a more significant factor in contributing to language bias. We also examine the state-of-the-art VQA models’ robustness to word-sequence-related disturbance of questions and found that models are resistant to such disturbance to some extent. Figure 1 shows an example. Moreover, we found that VQA models trained with variant questions demonstrated higher accuracy in the OOD evaluation. We conducted experiments to analyze the reasons behind this phenomenon and based on these findings, we proposed bias-reduction proposals to alleviate the language bias issue. To sum up, the main contributions of this paper are as follows:

• We provide empirical evidence demonstrating that language bias in VQA tasks is not solely caused by the co-occurrence of question types and answers, but also by the co-occurrence of visually-grounded concepts and answers, with the latter having a greater impact. Additionally, there may also exist multimodal bias.

• Extensive experiments reveal that models trained with variant questions outperform those trained with original questions. This improvement is attributed to the disruptions in the word sequence of questions, which impact the model’s learning of prior knowledge related to question types, leading to reduced bias-dependency learning.

• In light of the above findings, we propose de-biasing methods into multiple base VQA models by incorporating variant questions during training. The experimental results demonstrate significant performance enhancements on the VQA-CPv2 benchmark for the base models equipped with our proposed method.

2 Related Work

Visual Question Answering. As a high-level task that bridges the gap between computer vision and natural language processing, VQA [Antol et al., 2015; Yang et al., 2016; Agrawal et al., 2017; Anderson et al., 2018; Kim et al., 2018; Liu et al., 2022a] has received considerable attention from both the computer vision and natural language processing communities. Since the proposal of bottom-up and top-down (UpDn) attention mechanism [Anderson et al., 2018], it has been the de-facto standard baseline for the VQA task. [Kim et al., 2018] proposed a bilinear attention network (BAN) to efficiently compute multimodal representations. Additionally, [Yu et al., 2019] developed a deep modular co-attention network (MCAN) on top of the powerful Transformer [Vaswani et al., 2017], which models both intra- and inter-modal interactions simultaneously, making it a powerful baseline for the VQA task. In addition, pre-trained Vision Language Models (VLMs) [Tan and Bansal, 2019; Chen et al., 2020b; Su et al., 2020; Zhang et al., 2021; Zeng et al., 2022; Wang et al., 2022] that learn high-level multi-modal representations from large-scale data via a variety of pre-training tasks have demonstrated state-of-the-art performance in many Vision Language (V+L) tasks, including VQA. For instance, the DPT model [Liu et al., 2022b], which aligns the objectives of the pre-trained visual-language model with the specific requirements of the VQA task, has demonstrated improved generalizability and performance.

Bias and Robustness in VQA. The study of robustness in VQA is an important topic, particularly the issue of language bias, which significantly affects the OOD performance in VQA task. As such, an increasing number of bias-mitigation approaches [Cadene et al., 2019; Guo et al., 2021; Chen et al., 2020a; Han et al., 2021] and benchmarks [Agrawal et al., 2017; Agrawal et al., 2018; Kervadec et al., 2021a] have been proposed. [Cadene et al., 2019] built a question-only branch to capture the unwanted regularities by dynamically adjusting the loss. [Clark et al., 2019] trained a naive model that relied solely on dataset biases and then used an ensemble approach to incorporate a robust model that focused on other generalized patterns. [Liang et al., 2021] added a question-only branch to measure the intensity of language priors and then reshaped the objective function based on the loss of the question-only branch. [Lao et al., 2021] proposed the LP-Focal loss, which endows the cross-entropy loss with sample-level loss re-weights by building a question-only branch to capture language priors. [Yang et al., 2021] proposed a CCB method by building content and context branches to focus on local content and global context, respectively. On top of these two branches, a joint loss function with language bias optimizes the prediction. [Kervadec et al., 2021a] suggested that the standard evaluation metric is misleading by the overall accuracy under the unbalanced concepts and questions, thus they proposed a new benchmark consisting of a dataset and a new evaluation metric. Apart from the language bias issue, [Gokhale et al., 2020] found that VQA models could answer single questions but struggled to answer logical compositions of multiple such questions. Therefore, they constructed an augmentation of the VQA dataset by collecting logical com-

1 However, it has been revealed that the accuracy improvements of these methods result from the regularization effects [Shrestha et al., 2020]. Besides, collecting such human annotations can be expensive and burdensome.
position questions, including negation, conjunction, disjunction, and antonyms. [Shah et al., 2019] proposed a training scheme by exploiting cycle consistency to regularize the training process, which allows VQA models to become robust to linguistic variations. Besides, [Kervadec et al., 2021b] argued that noise and uncertainties in visual inputs are the main bottlenecks in VQA, which prevent the successful learning of reasoning capacities. SwapMix [Gupta et al., 2022] investigated the robustness of VQA models from the perspective of visual context. They swapped some irrelevant objects in the image and found VQA models are not robust for such visual context perturbation, indicating models over-rely on them to make predictions.

3 Empirical Analysis

3.1 Task Definition

The VQA task has been cast as a classification problem, where given an image, \(I\) and a question \(Q\), the objective is to predict an answer \(\hat{a}\) from all the candidate answers \(A\). This prediction is based on the image content and the context of the question. Without loss of generality, a VQA model can be formulated as a function transformation \(F : (Q, I) \rightarrow A\). The objective function \(p(.)\) is formulated as:

\[
\hat{a} = \arg \max_{a \in A} p(a|Q, I; \Theta),
\]

where \(\Theta\) denotes the model parameters. The common solution to predict the answer is via the cross-entropy loss

\[
L_{ce} = -\frac{1}{N} \sum_{i} a_i \log(p_i)
\]

\[
p_i = \text{Softmax}(Wh_i + b),
\]

where \(N\) denotes the number of samples, \(W\) and \(b\) are the learnable matrix and bias, \(h_i\) is the fused multi-modal feature.

3.2 Revisiting Question in VQA

Which Contributes More Bias?

The language priors bias in the VQA task is generally attributed to the co-occurrence of certain question-types and answers [Agrawal et al., 2018], and most bias-reduction methods are designed based on this hypothesis. In this section, we attempt to verify that the language bias issue is not solely due to the co-occurrence of question-types and answers through empirical analysis. To begin, we decompose the question into two parts: the question type (also known as the prefix) and the concepts (which include objects and other visually-grounded words or phrases in the question, also known as the postfix). We then examine their respective contributions to the final accuracy of the model. Intuitively, it is difficult to answer a question correctly if the question is incomplete (i.e., only the prefix or postfix is given). If an incomplete question is answered correctly, it suggests that there is some co-occurrence correlation between the incomplete portion and the corresponding answer, which indicates the presence of bias. These experiment settings are as follows. Dataset: We selected the widely-used VQAv2 benchmark [Goyal et al., 2017] and its OOD benchmark, VQA-CPv2 [Agrawal et al., 2018]. Base VQA models: In the experiment, we chose the most frequently used base models in the VQA task, which include attention-based models such as SAN [Yang et al., 2016] and UpDn [Anderson et al., 2018], bilinear attention network, BAN 2 [Kim et al., 2018], co-attention based model, MCAN [Yu et al., 2019], multi-modal pre-trained model, LXMERT [Tan and Bansal, 2019] and a question-only model (henceforth, Q-only). Among them, the LXMERT model uses BERT [Devlin et al., 2019] as the question encoder, while the other models use LSTM [Hochreiter and Schmidhuber, 1997] or GRU [Cho et al., 2014] as question encoders. Validation mode: We conducted two types of verification. The first type involved training models with original questions and evaluating them on either the prefix or postfix. The second type involved training models with either prefix or postfix and evaluating them on the original questions. All text inputs were padded or truncated to a fixed length using a predetermined character. Moreover, for the sake of simplicity in implementation, we used the questions with the prefix removed as postfix in our experiments.

The experimental results are presented in Table 1, and several important findings can be derived from these results. The results for the first type of verification mode are displayed in columns highlighted with a light green background in Table 1. We observed that all models with postfix inputs achieved better performance than those with prefix inputs on the VQA-CPv2 test split, particularly for the BAN, LXMERT, and MCAN models, where the postfix inputs contributed significantly more than the prefix. On the VQAv2 dataset, the BAN, LXMERT, and MCAN models performed slightly better with postfix inputs, while the prefix inputs resulted in better performance for the remaining models. The experimental results for the second type of verification mode are shown in columns against a light yellow background in Table 1. As can be seen, when models were trained with the postfix, their accuracy performances were better than those trained with the prefix for all the models except MCAN, which showed slightly lower performance on both VQA-CPv2 and VQAv2. The results from both verification modes in Table 1 indicate that postfix contribute more to the bias issue than prefixes. Moreover, we noticed that certain models (e.g., Q-only, UpDn and LXMERT) trained with postfix even outperformed those trained with the complete question on the VQA-CPv2 dataset. We conjecture that this could be attributed to the differences in question-type distributions between the training and test splits. Consequently, some models that do not learn information about the prefix of questions during the training process may exhibit better performance on VQA-CPv2.

Are There Any Other Kinds of Bias?

Also, Table 1 demonstrates that all models perform better than the Q-only model. Apart from the difference in model design, the Q-only model only takes the question as input, whereas others incorporate both the question and image as inputs. This allows these models to potentially depend on the co-occurrence between visual objects in the image and the keywords in the question. Therefore, we speculate that the bias issue exists not only in the language modality but also

\[ W^{l4-y} = \text{Use the 4-layers version of BAN in this paper.} \]
across multimodalities due to the fact that incomplete questions cannot be answered.

The Robustness to Variant Question

Robustness has always been a crucial concern in machine learning. [Cai et al., 2022] proposed a novel pre-training paradigm for language models. Their approach involves predicting the original order of perturbed words in text, aiming to enhance the model’s resilience to the text modality and improve its ability to comprehend text semantics. Inspired by their work, this paper investigated the robustness of state-of-the-art VQA models to disruptions in the questions. To achieve this, a series of confirmatory experiments were conducted to evaluate the robustness of VQA models. Specifically, we conducted the experiments on three types of variant questions. Given a question such as “what color is the flower?” with the prefix “what color is”, we defined three kinds of variant questions as follows:

- **variant-1**: = Concatenate(postfix, prefix), i.e., exchange the positions of the prefix and postfix, resulting in a variant such as “the flower color is what?”
- **variant-2**: = Random(question), i.e., shuffle the order of the words in the question randomly, resulting one of the possibilities as “the flower color is what?”
- **variant-3**: = Inverse(question), i.e., inverse the word sequence of the question, making the variant as “flower the is color what?”

To evaluate the model’s robustness to the variant questions, we define an evaluation metric Rob as follow,

\[
\%Rob = \frac{N_{rv,rq}}{N_{rq}} \times 100\%, \quad (3)
\]

where \(N_{rq}\) represents the number of correct predictions for the original questions, and \(N_{rv,rq}\) the number of both original questions and their variants that are correctly answered.

For the selection of VQA models, we have used the ones chosen in the previous section. The experimental results are presented in Table 2. The results on the VQA2 validation split demonstrate that all models experience varying degrees of performance degradation when evaluated on variant questions. Among them, MCAN and LXMERT show comparable performance on this in-distribution dataset. Furthermore, the results regarding robustness, Rob indicate that all models exhibit the best robustness for variant-1, followed by variant-2, and the worst for variant-3. Meanwhile, from the results of the VQA-CPv2 test split, an intriguing observation was that when tested with variant-1 questions, the test accuracy was even higher than that of the original questions in terms of the BAN, MCAN, and LXMERT models. The above experimental results lead us to consider a hypothesis, that is, variant questions may help alleviate language priors dependency.

To further verify this hypothesis, we conducted another series of experiments. In this experimental setup, we trained VQA models using variant questions as text inputs, and the resulting models are referred to as variant models.
Table 4: The ratio (in %) of prediction changes of variant models on VQA-CPv2 test split. The mark → (lower is better) measures the change in percentage (based on the predictions of original models) of questions that the original model can answer correctly while the variant models cannot; → (higher is better) represents the opposite case. The var(x) is abbreviated of variant-(x).

<table>
<thead>
<tr>
<th>Model</th>
<th>trained with</th>
<th>VQA-CPv2</th>
<th></th>
<th></th>
<th>VQAv2</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>All</td>
<td>Yes/No</td>
<td>Num</td>
<td>Other</td>
<td>All</td>
<td>Yes/No</td>
</tr>
<tr>
<td>Q-only</td>
<td>question</td>
<td>21.37</td>
<td>41.01</td>
<td>12.14</td>
<td>13.61</td>
<td>45.09</td>
<td>69.57</td>
</tr>
<tr>
<td></td>
<td>variant-1</td>
<td>27.80</td>
<td>53.64</td>
<td>39.32</td>
<td>11.09</td>
<td>41.81</td>
<td>68.21</td>
</tr>
<tr>
<td></td>
<td>variant-2</td>
<td>22.41</td>
<td>42.47</td>
<td>12.4</td>
<td>14.65</td>
<td>43.9</td>
<td>68.55</td>
</tr>
<tr>
<td></td>
<td>variant-3</td>
<td>25.90</td>
<td>66.02</td>
<td>1.34</td>
<td>11.62</td>
<td>32.38</td>
<td>65.69</td>
</tr>
<tr>
<td>SAN</td>
<td>question</td>
<td>40.70</td>
<td>41.62</td>
<td>13.14</td>
<td>47.77</td>
<td>62.78</td>
<td>78.69</td>
</tr>
<tr>
<td></td>
<td>variant-1</td>
<td>40.95</td>
<td>56.03</td>
<td>15.5</td>
<td>40.02</td>
<td>57.25</td>
<td>75.69</td>
</tr>
<tr>
<td></td>
<td>variant-2</td>
<td>41.32</td>
<td>43.21</td>
<td>13.17</td>
<td>48.06</td>
<td>61.47</td>
<td>76.96</td>
</tr>
<tr>
<td></td>
<td>variant-3</td>
<td>31.41</td>
<td>40.18</td>
<td>12.46</td>
<td>32.01</td>
<td>48.45</td>
<td>68.09</td>
</tr>
<tr>
<td>UpDn</td>
<td>question</td>
<td>41.55</td>
<td>42.91</td>
<td>13.56</td>
<td>48.55</td>
<td>65.56</td>
<td>82.87</td>
</tr>
<tr>
<td></td>
<td>variant-1</td>
<td>44.83</td>
<td>60.45</td>
<td>20.84</td>
<td>43.23</td>
<td>60.37</td>
<td>79.86</td>
</tr>
<tr>
<td></td>
<td>variant-2</td>
<td>42.33</td>
<td>44.89</td>
<td>13.35</td>
<td>48.94</td>
<td>64.22</td>
<td>81.5</td>
</tr>
<tr>
<td></td>
<td>variant-3</td>
<td>33.89</td>
<td>41.22</td>
<td>24.9</td>
<td>32.52</td>
<td>50.35</td>
<td>71.52</td>
</tr>
<tr>
<td>BAN</td>
<td>question</td>
<td>41.73</td>
<td>42.72</td>
<td>13.51</td>
<td>48.95</td>
<td>67.07</td>
<td>84.11</td>
</tr>
<tr>
<td></td>
<td>variant-1</td>
<td>47.19</td>
<td>61.21</td>
<td>18.41</td>
<td>47.74</td>
<td>63.81</td>
<td>81.39</td>
</tr>
<tr>
<td></td>
<td>variant-2</td>
<td>49.92</td>
<td>67.92</td>
<td>20.72</td>
<td>48.49</td>
<td>63.04</td>
<td>81.1</td>
</tr>
<tr>
<td></td>
<td>variant-3</td>
<td>45.87</td>
<td>66.81</td>
<td>15.04</td>
<td>43.37</td>
<td>55.19</td>
<td>75.51</td>
</tr>
<tr>
<td>LXMERT</td>
<td>question</td>
<td>43.29</td>
<td>46.37</td>
<td>15.38</td>
<td>49.34</td>
<td>65.67</td>
<td>83.31</td>
</tr>
<tr>
<td></td>
<td>variant-1</td>
<td>53.66</td>
<td>75.21</td>
<td>21.4</td>
<td>51.22</td>
<td>65.34</td>
<td>83.14</td>
</tr>
<tr>
<td></td>
<td>variant-2</td>
<td>43.57</td>
<td>46.49</td>
<td>15.71</td>
<td>49.68</td>
<td>65.29</td>
<td>83.19</td>
</tr>
<tr>
<td></td>
<td>variant-3</td>
<td>45.33</td>
<td>51.93</td>
<td>22.2</td>
<td>48.21</td>
<td>59.82</td>
<td>76.78</td>
</tr>
<tr>
<td>MCAN</td>
<td>question</td>
<td>43.73</td>
<td>42.6</td>
<td>15.69</td>
<td>52.02</td>
<td>68.65</td>
<td>85.91</td>
</tr>
<tr>
<td></td>
<td>variant-1</td>
<td>48.57</td>
<td>52.53</td>
<td>26.15</td>
<td>52.64</td>
<td>66.45</td>
<td>81.4</td>
</tr>
<tr>
<td></td>
<td>variant-2</td>
<td>48.79</td>
<td>55.79</td>
<td>22.00</td>
<td>52.48</td>
<td>66.65</td>
<td>81.93</td>
</tr>
<tr>
<td></td>
<td>variant-3</td>
<td>48.47</td>
<td>65.75</td>
<td>18.19</td>
<td>47.73</td>
<td>58.97</td>
<td>79.85</td>
</tr>
</tbody>
</table>

Table 3: The performance (in %) in terms of different variant models on VQA-CPv2 test split and VQAv2 validation split.

<table>
<thead>
<tr>
<th>Model</th>
<th>Q-only</th>
<th>SAN</th>
<th>UpDn</th>
<th>BAN</th>
<th>MCAN</th>
<th>LXMERT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>var1</td>
<td>var2</td>
<td>var3</td>
<td>var1</td>
<td>var2</td>
<td>var3</td>
</tr>
<tr>
<td></td>
<td>var1</td>
<td>var2</td>
<td>var3</td>
<td>var1</td>
<td>var2</td>
<td>var3</td>
</tr>
<tr>
<td></td>
<td>var1</td>
<td>var2</td>
<td>var3</td>
<td>var1</td>
<td>var2</td>
<td>var3</td>
</tr>
<tr>
<td></td>
<td>var1</td>
<td>var2</td>
<td>var3</td>
<td>var1</td>
<td>var2</td>
<td>var3</td>
</tr>
<tr>
<td>✓→ ✗</td>
<td>28.5</td>
<td>11.2</td>
<td>42.1</td>
<td>23.0</td>
<td>10.5</td>
<td>44.3</td>
</tr>
<tr>
<td>✗→ ✓</td>
<td>17.5</td>
<td>6.0</td>
<td>17.8</td>
<td>19.1</td>
<td>11.7</td>
<td>16.6</td>
</tr>
</tbody>
</table>

Table 4: The ratio (in %) of prediction changes of variant models on VQA-CPv2 test split. The mark ✓→ ✗ (lower is better) measures the ratio (based on the predictions of original models) of questions that the original model can answer correctly while the variant models cannot; ✗→ ✓ (higher is better) represents the opposite case. The var(x) is abbreviated of variant-(x).

The inference stage, we evaluated the performance of the variant models on original questions. The experimental results are presented in Table 3. As shown, some variant models achieved comparable performance to that of the original models on VQAv2 dataset. Furthermore, almost all the variant models achieve better performance on VQA-CPv2, except for SAN and UpDn, whose performances degrade when trained with variant-3. The LXMERT, fine-tuned for 20 epochs, even achieved a 10-point gain without adopting any debiasing methods when trained with variant-1 questions. Besides, a notable phenomenon can be observed in Table 3, which is that the accuracy changes of different variant models mainly occur in the “Yes/No” metric, although the changes demonstrated by different models vary.

3.3 Why Did the Performance Improve?

To figure out the reasons behind the accuracy improvements from these variant models on VQA-CPv2, we conducted further experimental analysis. Firstly, we performed a fine-grained analysis of the experimental results to investigate how the predictions of the variant models differed from those of the original models. To be more specific, we aimed to determine the number of predictions that changed from correct to incorrect and vice versa for the variant models.

The results are presented in Table 4, from which it can be seen that almost all of the variant-2 models have the smallest proportion of samples that were predicted correctly by the original model but predicted incorrectly by the variant models. The corresponding variant-1 model of the LXMERT model performs the best, with the smallest proportion of correct predictions flipped to incorrect ones, and it can also flip 27.6% of the incorrect predictions to correct ones. Besides, the BAN and MCAN models also demonstrate good performance regarding converting incorrect predictions to correct ones. Additionally, we conducted further analysis to examine the question types that correspond to the change in performance of the variant models. The results are depicted in Figure 2. It can be observed that the accuracy improvement
mainly stems from the “Yes/No” answer type, and the most frequent question type is “is there”. On the other hand, the decreased performance mainly comes from the “Other” answer type, and the most frequent corresponding question type is “what”. Based on the previous results, we aimed to investigate further which words in the question the variant models focus on and how this differs from the attention of the original models. To achieve this, one of the most straightforward ways is to compare the feature representations produced by the models and their corresponding variant models for the same question. We visualized the attention weights of the questions with respect to the models and their variant models. Specifically, the question was first fed into the model to obtain its feature representation. Then, we mapped the attention to each word of the question. Figure 3 presents two toy examples. The subplots (a) and (c) in Figure 3 show that the model trained using the original question input mode places a higher weight on question type, such as the examples “is this” and “what color is”. The abundance of questions that start with the phrase “is this” in the training dataset makes it easier for the models to learn these simple patterns. In contrast, for the models trained with variant questions, the aforementioned scenario does not occur, resulting in slightly smoother visualization feature representations, as shown in subplots (b) and (d) of Figure 3.

In addition, more detailed results are presented in Table 3. For instance, the results of the trained variant models on the in-distribution VQA v2 dataset reveal that almost every model’s performance, in terms of each answer type, has decreased to varying degrees when compared to the performance of the original models. This demonstrates that learning the pattern of the variant questions would negatively affect the performance of the original question on in-distribution data. However, the performance of variant models on the OOD test set is quite different. Almost all models showed improvements in the All metric, primarily due to the improvements in “Yes/No” and “Num” answer types. However, the models trained with variant-1 questions, including Q-only, SAN, BAN, and UpDn, showed a decrease in performance on the “Other” metric. In contrast, LXMERT improved its performance when trained with variant-1 questions, with an accuracy improvement of +1.88% on “Other” metric. The reason for these experimental results may lie in the different question encoders used by these models.

3.4 Other Property

Furthermore, we found that the trained variant models exhibit better semantic robustness than the original models. We calculated the semantic similarity between the encoded original questions and encoded the variant questions by:

$$\text{simi} = 1 - \frac{1}{N} \sum_{i} \cos < q_i, var_i >$$

where $N$ is the number of the samples, $q_i$ and $var_i$ denote the encoded original question and variant question, respectively. The results are shown in Figure 4.

4 How to Utilize These Traits?

The variant models exhibit more promising results on VQA-CPv2 compared to the original model. As demonstrated in the previous section, the variant models can avoid learning the inherent prior knowledge related to question types from the questions, allowing them to focus on other useful patterns. However, it should be noted that the syntactic structure of the variant questions may be incomplete or incorrect, and the semantics of the variant questions may even be entirely different. Therefore, the variant questions can only be utilized to aid in the design of de-biasing methods.

4.1 Proposals

To take advantage of the trait of variant models while preserving the semantics of the original question, one approach is to use the contrastive learning paradigm to combine the two encodings. In this approach, the variant question is treated as the positive sample, while negative samples are randomly sampled from the mini-batch during training. The process can be formulated as $\mathcal{L}_{con} = -\frac{1}{N} \sum_{i} \log(\frac{\text{sim}(h_i, h_{i}^{pos})}{\text{sim}(h_i, h_{i}^{neg}) + \sum_{j} \text{sim}(h_i, h_{j}^{neg})})$, where $h_i$ is the joint feature representation of two kinds modality, $h_i^{pos}$ and $h_i^{neg}$ represent the positive feature and negative feature, respectively, $\text{sim}(h_i, h_i^{pos}) = \frac{|h_i^{pos}||h_i|}{|h_i^{pos}||h_i|}$. During the training stage, the total optimization objective includes the VQA loss.
Figure 3: Examples of the visualization concerning the weight mapping to each word of the questions. The subplots (a) and (b) are the visualizations with respect to the question “Is this a cowboy hat?”, (a) is the result of the UpDn model trained with original questions, (b) is the result of UpDn model trained with variant questions. (c) and (d) are the visualizations with respect to the question “What color is the shirt?”, (c) is the result of UpDn model trained with original questions, (d) is the result of UpDn model trained with variant questions.

Figure 4: The semantic similarity between the encoded original questions and the encoded variant questions under different models. simi-(i) means the semantic similarity between the original question and the variant-(i) question.

\[ L_{ce} \] and the contrastive loss \( L_{con} \). Besides, another straightforward way is to combine the features of the original question with those of the variant question directly as a kind of data augmentation. Specifically, the joint features were encoded by a weighted combination of two kinds of features. Therefore, the resulting features cover richer patterns.

4.2 Experiments

In this section, we validate the effectiveness of the latter debiasing method on the OOD benchmark, i.e., VQA-CPv2. The proposed method is model-agnostic and can be combined with any other VQA model. Here, we also choose the most widely used base VQA models, SAN, UpDn, BAN, LXMERT, MCAN, and the Q-only model, as the baseline models. Regarding the implementation details in the training process, we adhere to the experimental settings of the open-source codes and do not modify other parameters such as learning rate, batch size, or optimizer. The experimental results are presented in Table 5. As evident from the results, all base models exhibited performance improvements when integrated with the method proposed in this paper. Moreover, the majority of models demonstrated enhancements in the “Other” metric, with only the UpDn and LXMERT models experiencing a slight decrease. This indicates that the base models combined with the proposed method can not only improve the simple pattern but also learn more difficult patterns. In addition, while the performance of some models combined with the proposed method may not be as good as the results of the variant models, the overall accuracy has improved significantly compared to the original models.

<table>
<thead>
<tr>
<th>Model</th>
<th>VQA-CPv2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>All</td>
</tr>
<tr>
<td>Q-only</td>
<td>21.37</td>
</tr>
<tr>
<td>Q-only+ours</td>
<td>26.3</td>
</tr>
<tr>
<td>SAN</td>
<td>40.70</td>
</tr>
<tr>
<td>SAN+ours</td>
<td>41.41</td>
</tr>
<tr>
<td>UpDn</td>
<td>41.53</td>
</tr>
<tr>
<td>UpDn+ours</td>
<td>44.95</td>
</tr>
<tr>
<td>BAN</td>
<td>41.73</td>
</tr>
<tr>
<td>BAN+ours</td>
<td>43.63</td>
</tr>
<tr>
<td>MCAN</td>
<td>43.73</td>
</tr>
<tr>
<td>MCAN+ours</td>
<td>44.89</td>
</tr>
<tr>
<td>LXMERT</td>
<td>43.29</td>
</tr>
<tr>
<td>LXMERT+ours</td>
<td>50.85</td>
</tr>
</tbody>
</table>

Table 5: The experimental results (Acc.%) of base VQA models combined with the proposed data augmentation method. Note that the LXMERT was fine-tuned for 20 epochs.

5 Conclusion and Future Work

In this paper, we investigate language modality in the VQA task through experimental analysis. The empirical findings indicated that the issue of language priors bias is not only related to question types alone, the postfix of questions even has a greater impact on language bias. Furthermore, we observed that variant models outperform original models on the VQA-CPv2 benchmark. We identified the underlying reasons for these results and proposed new debiasing methods based on these findings. The experimental results demonstrated that our method enhances the VQA models’ generalization ability. Our main purpose is not to pursue state-of-the-art results but to gain insights for designing bias-reduction methods. However, we only present some novel experimental findings, and we plan to provide in-depth theoretical analysis and probe other methods to leverage these traits in future work.
Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant No. 62276110, No. 62172039 and in part by the fund of Joint Laboratory of HUST and Pingan Property & Casualty Research (HPL). The authors would also like to thank the anonymous reviewers for their comments on improving the quality of this paper.

References


[Kervadec et al., 2021a] Corentin Kervadec, Grigory Antipov, Moez Baccouche, and Christian Wolf. Roses are red, violet are blue... but should vqa expect them to? In CVPR, pages 2776–2785, 2021.


