

# AdaVQA: Overcoming Language Priors with Adapted Margin Cosine Loss

Yangyang Guo<sup>1</sup>, Liqiang Nie<sup>1\*</sup>, Zhiyong Cheng<sup>2\*</sup>, Feng Ji<sup>3</sup>, Ji Zhang<sup>3</sup>, Alberto Del Bimbo<sup>4</sup>

<sup>1</sup>Shandong University

<sup>2</sup>Shandong Artificial Intelligence Institute

<sup>3</sup>Alibaba Group

<sup>4</sup>University of Florence

{guoyang.eric, nieliqiang, jason.zy.cheng}@gmail.com, {zhongxiu.jf, zj122146}@alibaba-inc.com, alberto.delbimbo@unifi.it

## Abstract

A number of studies point out that current Visual Question Answering (VQA) models are severely affected by the language prior problem, which refers to blindly making predictions based on the language shortcut. Some efforts have been devoted to overcoming this issue with delicate models. However, there is no research to address it from the view of the answer feature space learning, despite the fact that existing VQA methods all cast VQA as a classification task. Inspired by this, in this work, we attempt to tackle the language prior problem from the viewpoint of the feature space learning. An adapted margin cosine loss is designed to discriminate the frequent and the sparse answer feature space under each question type properly. In this way, the limited patterns within the language modality can be largely reduced to eliminate the language priors. We apply this loss function to several baseline models and evaluate its effectiveness on two VQA-CP benchmarks. Experimental results demonstrate that our proposed adapted margin cosine loss can enhance the baseline models with an absolute performance gain of 15% on average, strongly verifying the potential of tackling the language prior problem in VQA from the angle of the answer feature space learning.

## 1 Introduction

Visual Question Answering (VQA) targets at accurately answering natural language questions regarding a visual scene. Perceived as a more ‘AI-complete’ task, VQA has received critical attention from both computer vision and natural language processing communities. Recently, several large-scale benchmarks [Antol *et al.*, 2015; Goyal *et al.*, 2017] have been constructed to facilitate the advancement of VQA, followed by a number of elaborately designed visual reasoning models [Kazemi and Elqursh, 2017; Anderson *et al.*, 2018; Zhang *et al.*, 2018].

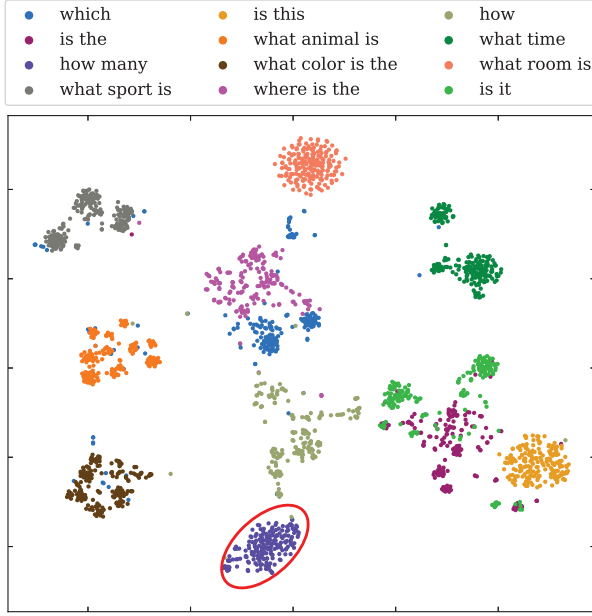
Despite its flourishing development, there still exists a major issue in VQA, that is, most VQA models tend to make predictions by taking the language shortcut between questions (or more specifically, question types<sup>1</sup>) and answers. For example, 2 takes large proportions of the answers to questions initiated with *how many* in the train set. In such case, the model will take the shortcut to leverage these language priors and directly yields answer 2 to *how many* questions when testing, regardless of both the remaining question words and the given image. As a result, the visual signal is largely under-explored, making the visual reasoning in VQA deteriorate into the a pure language matching problem without the consideration of ‘V’. Obviously, it is a contradiction to the initial motivation of this vision-and-language task.

Many research efforts have been devoted to overcoming this arduous problem, which can be roughly projected into two groups of methods: 1) balancing the biased dataset; and 2) correcting VQA models for alleviating the effects of language priors. In detail, the first category of methods contribute to either enlarging [Goyal *et al.*, 2017] or re-splitting [Agrawal *et al.*, 2018] the existing biased datasets. Notably, [Agrawal *et al.*, 2018] curated a diagnostic dataset - VQA-CP, wherein the answer distributions of per question type are significantly distinct between the train and test sets. It thus provides a favorable test-bed for estimating the language prior effect the VQA model encounters. Based on this dataset, some pioneering work has been conducted to correct the advanced VQA models for alleviating language prior problem [Ramakrishnan *et al.*, 2018; Chen *et al.*, 2020; Guo *et al.*, 2020]. A typical design of this category of methods is to attach an additional question-only branch, whereby the language prior is intentionally captured and further suppressed by the orthodox question-image branch.

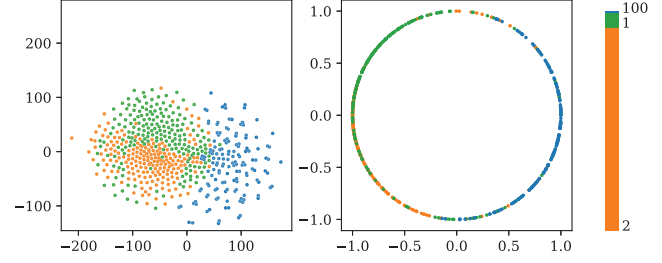
Due to the fact that the answers in most VQA datasets are composed of a few words, existing VQA models often cast VQA as a classification task, i.e., classifying the answer from candidates rather than progressively generating answer words to achieve better visual reasoning capability. However, what attracts our attention is that, there has been no research diving into investigating the learned feature space that each answer/class engages. We argue that distinguishing answers in

\*Corresponding Author: Liqiang Nie and Zhiyong Cheng.

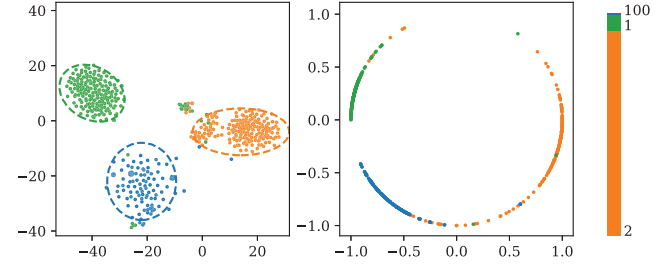
<sup>1</sup>We refer question type in VQA to the first few words in the given question.



(a) Answer manifold embedding with respect to question types.



(b) Answer manifold embedding of *how many* question type in Euclidean and angular spaces from the baseline.



(c) Answer manifold embedding of *how many* question type in Euclidean and angular spaces from our method.

Figure 1: Answer manifold embedding with respect to questions types in (a) and answers under *how many* question type in (b) and (c). Note that the numbers of each answer class are identical for better visualization. For (b) and (c), the answer distribution in the train set is demonstrated on the rightmost.

the learned feature space is the key to alleviate the language prior effect but existing models cannot well address this in the biased dataset. Inspired by this thought, we empirically visualize the 2-D embedding features of the answers to the question type *how many* in Fig 1b. It is obvious that different answers of the baseline are actually intertwined with each other in the two kinds of feature spaces. On the basis of this phenomenon, we make a further assumption: is it beneficial for overcoming the language prior problem if we properly separate the answers via manipulating their learned feature space?

To address this question, we propose to revisit the language prior problem from the viewpoint of the answer feature space learning. The primary goal of this work is to introduce an adapted margin for different answers in the angular space under the corresponding question type, which can effectively separate the answer embeddings. Towards this end, we firstly reformulate the softmax loss function as a cosine loss by  $L2$  normalizing both answer features  $\mathbf{x}$  and weight vectors  $\mathbf{W}_i$  [Liu *et al.*, 2017; Wang *et al.*, 2018a]. Consequently, the decision boundary can be computed through the cosine function of the angle  $\theta_i$  between  $\mathbf{x}$  and  $\mathbf{W}_i$ , i.e.,  $\mathbf{W}_i^T \mathbf{x} = \|\mathbf{W}_i\|_2 \cdot \|\mathbf{x}\|_2 \cdot \cos\theta_i = \cos\theta_i$ , s.t.  $\|\mathbf{W}_i\|_2 = 1$  and  $\|\mathbf{x}\|_2 = 1$ . Thereafter, an adapted margin  $m_i$  is employed to separate the answer features, i.e.,  $\cos\theta_i - m_i$ , where  $m_i$  is computed based on the train set statistics of answer  $a_i$  under the corresponding question type. In the ideal scenario, for each given question and its corresponding question type, frequent answers would span broader in the angular space with a smaller margin, while sparse answers span tighter in the angular space with a larger margin (see Fig. 1c). The intuition behind this is that frequent answers under certain ques-

tion types often go with more training samples, and thus a smaller margin is required to control a broader space to sufficiently include these answers. In contrast, the number of sparse answers is relatively small, requiring a more tighter feature space. Moreover, we find that the answer embeddings in the Euclidean space become more discriminative, and further alleviating the language prior effect.

In a nutshell, the proposed adapted margin cosine loss is beneficial for overcoming the language prior problem in VQA, making itself being model-agnostic and easy to implement. Besides, it introduces NO incremental parameters or computational cost when testing. To the best of our knowledge, this is the first attempt to directly optimize the feature space learning to attack this problem. Additionally, we apply this loss function over three baseline methods, verifying its effectiveness on two VQA-CP benchmarks. The experimental results demonstrate that the baselines equipped with our loss achieve an absolute gain of over **15%** averagely. The code is released for the re-implementation of this work<sup>2</sup>.

## 2 Related Work

In recent years, a considerable literature has thrived around the task of VQA [Antol *et al.*, 2015; Malinowski *et al.*, 2018; Goyal *et al.*, 2017; Wang *et al.*, 2018b; Zhang *et al.*, 2018]. From the very beginning, researchers proposed to encode the question and image into a common latent space, based on which the accurate answers can be identified through classification [Antol *et al.*, 2015; Malinowski *et al.*, 2015]. With the prevailing advent of the attention mechanism, different contributions of each image region [Yang *et al.*, 2016;

<sup>2</sup><https://github.com/guoyang9/AdaVQA>.

Fukui *et al.*, 2016; Kazemi and Elqursh, 2017; Anderson *et al.*, 2018] or each question word [Lu *et al.*, 2016; Yu *et al.*, 2017] for the answer prediction have been taken into consideration. Some studies have also explored the modular structure of questions for more explicit reasoning [Andreas *et al.*, 2016]. Nevertheless, questions in real world often require reasoning beyond what is in the image. To better handle such situation, FVQA [Wang *et al.*, 2018b] and OK-VQA [Marino *et al.*, 2019] develop new benchmarks as well as insightful approaches to incorporate structured and unstructured knowledge in VQA, respectively.

It is noted that many VQA models are influenced by the language prior problem [Agrawal *et al.*, 2018; Guo *et al.*, 2019]. Therefore, some efforts have been devoted to the biased dataset balancing. For instance, VQA v2 [Goyal *et al.*, 2017] adds complementary samples to the original VQA v1 dataset [Antol *et al.*, 2015] such that for each sample  $\langle v, q, a \rangle$ , another one with the same  $q$ , similar  $v$  yet a different  $a$  is generated. VQA-CP [Agrawal *et al.*, 2018] re-splits the VQA v1 and v2 datasets that the answer distribution of per question type is distinct between the train and test sets. In addition to the studies on the dataset side, methods devised for tackling this problem mostly try to diminish the answer prediction influence from the question encoding layer [Selvaraju *et al.*, 2019; Wu and Mooney, 2019; KV and Mittal, 2020; Jing *et al.*, 2020]. One popular solution essentially involves a new question-only training branch, wherein the answers are predicted based solely on the question input [Ramakrishnan *et al.*, 2018; Cadene *et al.*, 2019; Clark *et al.*, 2019; Chen *et al.*, 2020]. The answer prediction from this branch is further suppressed by the original question-image one. In addition, [Guo *et al.*, 2019] designed an evaluation metric to quantify the language prior effect in VQA models.

### 3 Proposed Method

#### 3.1 Problem Definition

Following the prevalent formulation, we consider the VQA task as a multi-label classification problem. That is, for a textual question  $Q$  regarding an image  $I$ , the objective function is given by:

$$\hat{a} = \arg \max_{a \in \Omega} p(a|Q, I; \Theta), \quad (1)$$

where  $\Omega$  and  $\Theta$  denote the answer set and the model parameters, respectively. Note that there can be multiple correct answers for each instance. The negative log likelihood loss function is then formulated as<sup>3</sup>:

$$\begin{aligned} L_s &= \sum_{i=1}^{|\Omega|} -a_i \log p_i \\ &= \sum_{i=1}^{|\Omega|} -a_i \log \frac{\exp(\mathbf{W}_i^T \mathbf{x})}{\sum_{j=1}^{|\Omega|} \exp(\mathbf{W}_j^T \mathbf{x})}, \end{aligned} \quad (2)$$

where  $\mathbf{W}$  and  $\mathbf{x}$  denote the weight matrix and feature vector directly adjacent to the answer prediction, respectively;

<sup>3</sup>The individual loss is considered instead of the total instances throughout this paper due to the space limitation.

$a_i \in [0, 1]$  represents the corresponding answer label. Besides, we remove the bias vector for simplicity as we also found it contributes little to the final model performance.

#### 3.2 Method Intuition

To date, most VQA methods tackling the language prior problem present to confront the prediction from a question-only branch. Little attention has been paid to explore the answer/class embedding in the learned feature space. In this work, we adopt a new viewpoint to overcome this problem, and validate its effectiveness in what follows. Specifically, we propose to place an adapted margin to separate the answer embeddings in the angular space properly based upon the statistics in the train set. Compared to the Euclidean space, the angular space is relatively easy to manipulate, as the margin range reduces from  $(-\infty, +\infty)$  to  $[-1, 1]$ . In the light of this, we firstly leverage the  $L_2$  normalization on weight vector  $\mathbf{W}_i$  and feature vector  $\mathbf{x}$  to ensure the posterior probability to be determined by the angle  $\theta_i$  between  $\mathbf{W}_i$  and  $\mathbf{x}$  (the answer feature space is converted from the Euclidean space to the angular space), i.e.,  $\|\mathbf{W}_i\|_2 = 1$  and  $\|\mathbf{x}\|_2 = 1$ . Accordingly, a modified normalized SoftMax loss (NSL) [Wang *et al.*, 2018a] can be obtained by,

$$L_{ns} = \sum_{i=1}^{|\Omega|} -a_i \log \frac{\exp s \cos \theta_i}{\sum_{j=1}^{|\Omega|} \exp s \cos \theta_j}, \quad (3)$$

where  $s$  is a scale factor for more stable computation. In order to achieve a more discriminative classification boundary, LMLC [Wang *et al.*, 2018a] introduces a fixed cosine margin to NSL,

$$L_{lmc} = \sum_{i=1}^{|\Omega|} -a_i \log \frac{\exp s(\cos \theta_i - m)}{\sum_{j \neq i}^{|\Omega|} \exp s \cos \theta_j + \exp s(\cos \theta_i - m)}, \quad (4)$$

where  $m$  implies the fixed cosine margin.

In implementation, we found that applying a fixed cosine margin cannot obtain satisfactory results. The key reason is that the answer distribution of each question type is highly biased, resulting in the incapability of learning a sufficient representation with a fixed margin in the angular space. In the next subsection, we will introduce a more sophisticated adapted margin cosine loss to overcome this issue.

#### 3.3 AdaVQA Formulation

The results in our experiments (see Sect. 4.2) explicitly demonstrate that a fixed cosine margin yields limited improvements or even deteriorates the model performance. Based upon this observation, we argue that an adapted cosine margin is more favorable for overcoming the language priors in VQA. In view of this, a new loss function is elaborately defined as:

$$\begin{cases} L_{AdaVQA} = \sum_{i=1}^{|\Omega|} -a_i \log \frac{\exp s(\cos \theta_i - m_i)}{\sum_{j=1}^{|\Omega|} \exp s(\cos \theta_j - m_j)}, \\ m_i = 1 - \bar{m}_i, \\ \bar{m}_i = \frac{n_i^k + \epsilon}{\sum_{j=1}^{|\Omega|} n_j^k + \epsilon}, \end{cases} \quad (5)$$

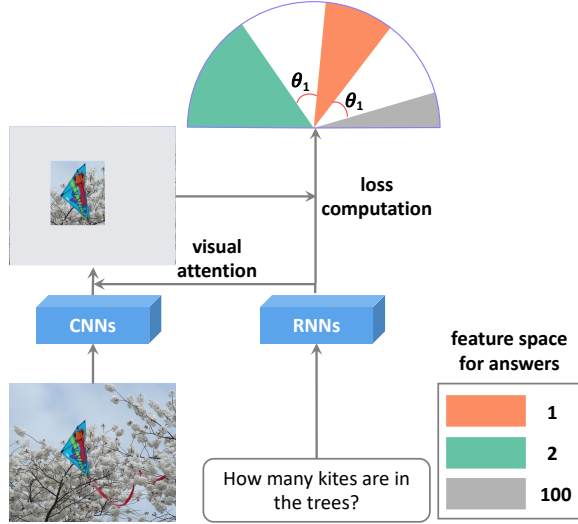


Figure 2: The training pipeline of AdaVQA. The input image and question is consumed via pre-trained convolutional neural networks (CNNs) and recurrent neural networks (RNNs), respectively. A visual attention module is then postfixed for identifying salient image regions pertaining to the given question. After normalizing the weight and feature vectors, an adapted cosine margin is introduced to produce suitable decision boundaries.

where  $m_i$  is the adapted margin for answer  $i$  based on the given question type,  $n_i^k$  denotes the number of answer  $i$  under the question type  $qt_k$  in the train set,  $\epsilon = 1e - 6$  is a hyper-parameter for avoiding computational overflow. The intuition behind this is that, for the current given question and its corresponding question type, the frequent answers span broader in the angular space (smaller margin), while sparse answers span tighter (larger margin). In other words, frequent answers imply more training samples, which require a broader feature space to sufficiently include these answers. On the contrast, a tighter feature space is acceptable for sparse answers as the number of training samples is much smaller. This setting enables the VQA models to place a more proper cosine margin in the angular answer feature space. Taking Fig. 1c as an example, answer 2 is more frequent than the other two answers (1 and 100) in the train set. As a result, it spans a larger feature space as we expect. In particular, Fig. 2 provides a general training pipeline of AdaVQA, where the feature space for the three answers (1, 2, 100) is learned based on the proposed loss function.

However, it will be a concern that such a margin restriction might damage the ability of the inter-type discrimination, i.e., the answer decision boundary among different question types becomes less discriminative. We therefore obtain the answer embeddings with respect to its question type and visualize them in Fig. 1a. From this figure, an obvious decision boundary pertaining to question types can be observed. One possible reason to this situation is that the cosine margin of each answer is computed according to the train set statistics under the given question type, while other answers outside this question type are kept untouched. This empirical setting inherently equip the model with the discrimination capability among different question types.

**Entropy threshold.** One major problem have yet to be addressed, i.e., not all questions within their corresponding question type would cause the language prior problem. Different from previous methods handling all questions without differentiation, we propose to employ an entropy threshold over each question type. Concretely, only the questions whose question type entropy is greater than the threshold will be considered for regularization. Specifically, the entropy for each question type is defined as:

$$e_{qt} = - \sum_{i=1}^{|\Omega|} \bar{m}_i^k \log_2 \bar{m}_i^k, \quad (6)$$

where  $\bar{m}_i^k$  is actually the occurrence probability of answer  $i$  under the question type  $qt_k$  in the train set.

**Partial derivatives.** We further provide the partial derivatives of the weight vector  $\mathbf{W}_i$  and feature vector  $\mathbf{x}$  from our loss function. Let  $p_i = s(\cos \theta_i - m_i) = s(\frac{\mathbf{W}_i^T \mathbf{x}}{\|\mathbf{W}_i\|_2 \cdot \|\mathbf{x}\|_2} - m_i)$ , and  $\hat{p}_i = \frac{\exp p_i}{\sum_{j=1}^{|\Omega|} \exp p_j}$ . The partial derivatives are obtained via,

$$\frac{\partial L_{AdaVQA}}{\partial \mathbf{x}} = \sum_{i=1}^{|\Omega|} (\sum_{j=1}^{|\Omega|} a_j \hat{p}_i - a_i) \cdot s \cdot \frac{\|\mathbf{x}\|_2 \mathbf{I} - \mathbf{x} \mathbf{x}^T}{\|\mathbf{x}\|_2^3} \frac{\mathbf{W}_i}{\|\mathbf{W}_i\|_2}, \quad (7)$$

and

$$\frac{\partial L_{AdaVQA}}{\partial \mathbf{W}_i} = \sum_{i=1}^{|\Omega|} (\sum_{j=1}^{|\Omega|} a_j \hat{p}_i - a_i) \cdot s \cdot \frac{\|\mathbf{W}_i\|_2 \mathbf{I} - \mathbf{W}_i \mathbf{W}_i^T}{\|\mathbf{W}_i\|_2^3} \frac{\mathbf{x}}{\|\mathbf{x}\|_2}. \quad (8)$$

**Lower bound for s.** The scale factor  $s$  is critical for the final answer feature learning. A too small  $s$  leads to an insufficient convergence as it limits the feature space span (as we found in our experiments that the loss goes ‘nan’ with a small  $s$ ). In view of this, a lower bound for  $s$  should be prescribed. Without loss of generality, let  $P_i$  denote the expected minimum of the answer class  $i$ , the lower bound is defined as,

$$s \geq - \frac{\ln(\frac{1}{P_i} - 1)}{2 - m_i - \frac{\sum_{j \neq i} m_j}{|\Omega| - 1}}. \quad (9)$$

### 3.4 Comparison with Different Loss Functions

We consider the binary-class scenario for intuitively illustrating the decision boundary from different loss functions. As can be seen from Fig. 3, the decision boundary of the plain SoftMax one can be negative, which is enlarged to be equal to zero of the NSL loss [Wang *et al.*, 2018a]. The LMLC [Wang *et al.*, 2018a] defines a fixed margin for different classes, which is not suitable in our case for overcoming the language prior problem. Regarding our AdaVQA, the sparser answer (green one) is mapped to a smaller feature space while the more frequent answer (orange one) engages a larger feature space.

## 4 Experiments

### 4.1 Experimental Setup

**Datasets.** To validate the effectiveness of the proposed loss function, we conducted extensive experiments on the two

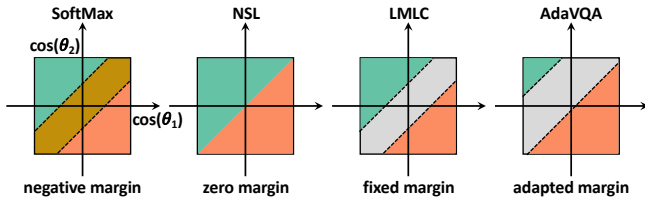


Figure 3: The visual comparison of decision boundary from different methods.

VQA-CP datasets: VQA-CP v2 and VQA-CP v1 [Agrawal *et al.*, 2018], which are two public benchmarks for estimating the models’ capability of overcoming the language prior problem in VQA.

**Evaluation metric.** The standard metric in VQA is adopted for evaluation [Antol *et al.*, 2015]. That is, for each predicted answer  $a$ , the accuracy is computed as,

$$Acc_a = \min(1, \frac{\# \text{humans that provide answer } a}{3}). \quad (10)$$

In addition, the answers are divided into three categories: *yes/no*, *number* and *other*.

**Implementation details.** We empirically applied our AdaVQA to three baselines: Strong-BL [Kazemi and Elqursh, 2017], Counter [Zhang *et al.*, 2018] and UpDn [Anderson *et al.*, 2018]. Different from most prior methods overcoming the language prior problem in VQA, for all the three baselines, we simply replaced the original loss with our AdaVQA and did NOT change any other settings, such as embedding size, learning rate, optimizer, and batch size.

## 4.2 Experimental Results

### Overall Accuracy

The experimental results on VQA-CP v2 and VQA-CP v1 are illustrated in Tab. 1 and Tab. 2, respectively. The main observations from these two tables are as follows:

- Our method achieves the best on the two benchmark datasets except for the *Other* answer category (SCR [Chen *et al.*, 2020] and LMH [Clark *et al.*, 2019] outperform ours on VQA-CP v2 and VQA-CP v1, respectively.). One possible reason for this is that most questions under this answer category are starting with ‘what ...’, and the answers are therefore much more diverse than other categories. Accordingly, less language prior will be introduced under this category since the answers are less biased.
- For all the three baselines, i.e., Strong-BL, Counter and UpDn, when applying our loss function, a significant performance improvement (15% on average) can be observed. For example, on the VQA-CP v2 dataset, Counter+Ours achieves an absolute performance gain of 12.23% compared with Counter; and on the VQA-CP v1 dataset, UpDn+Ours outperforms the baseline UpDn with 23.18%.
- Compared with other methods whose backbone model is also UpDn on the VQA-CP v2 dataset, our method (UpDn+Ours) surpasses them with a large margin, especially for the three newly accepted approaches VGQE, CSS and Decomp-LR.

Method	VQA-CP v2 test			
	Y/N	Num.	Other	All
Question-Only [2015]	35.09	11.63	07.11	15.95
SAN [2016]	38.35	11.14	21.74	24.96
NMN [2016]	38.94	11.92	25.72	27.47
MCB [2016]	41.01	11.96	40.57	36.33
Strong-BL†[2017]	40.04	12.01	37.60	34.41
HAN [2018]	52.25	13.79	20.33	28.65
Counter†[2018]	41.01	12.98	42.69	37.67
UpDn [2018]	42.27	11.93	46.05	39.74
UpDn†[2018]	49.78	14.07	43.42	40.79
GVQA [2018]	57.99	13.68	22.14	31.30
AdvReg [2018]	65.49	15.48	35.48	41.17
Rubi [2019]	68.65	20.28	43.18	47.11
LMH [2019]	-	-	-	52.05
LMH†[2019]	70.29	44.10	44.86	52.15
HINT [2019]	67.27	10.61	45.88	46.73
SCR [2019]	72.36	10.93	<b>48.02</b>	49.45
VGQE [2020]	66.35	27.08	46.77	50.11
CSS [2020]	43.96	12.78	47.48	41.16
Decomp-LR [2020]	70.99	18.72	45.57	48.87
Strong-BL+Ours	62.70	21.64	35.33	41.21
Counter+Ours	61.00	53.22	43.17	49.90
UpDn+Ours	<b>72.47</b>	<b>53.81</b>	45.58	<b>54.67</b>

Table 1: Accuracy comparisons with respect to different answer categories over the VQA-CP v2 dataset. Regarding the method category, the first group denotes plain approaches, the second group represents methods directly applied on the UpDn baseline, and the approaches from the last group are with our loss function. ‘-’ and ‘†’ denote the numbers are unavailable and our implementation, respectively. The best performance in current splits is highlighted in bold.

Method	VQA-CP v1 test			
	Y/N	Num.	Other	All
Question-only [2015]	35.72	11.07	08.34	20.16
SAN [2016]	35.34	11.34	24.70	26.88
NMN [2016]	38.85	11.23	27.88	29.64
MCB [2016]	37.96	11.80	39.90	34.39
Strong-BL†[2017]	40.04	12.01	37.60	34.41
Counter†[2018]	41.01	12.98	42.69	37.67
UpDn†[2018]	43.76	12.49	42.57	38.02
GVQA [2018]	64.72	11.87	24.86	39.23
AdvReg [2018]	74.16	12.44	25.32	43.43
LMH†[2019]	76.61	29.05	<b>43.38</b>	54.76
Strong-BL+Ours	72.64	27.61	36.11	49.84
Counter+Ours	72.01	<b>49.28</b>	42.60	55.92
UpDn+Ours	<b>91.17</b>	41.34	39.38	<b>61.20</b>

Table 2: Accuracy comparisons with respect to different answer categories over the VQA-CP v1 dataset. ‘†’ denotes our implementation. The best performance in current splits is highlighted in bold.



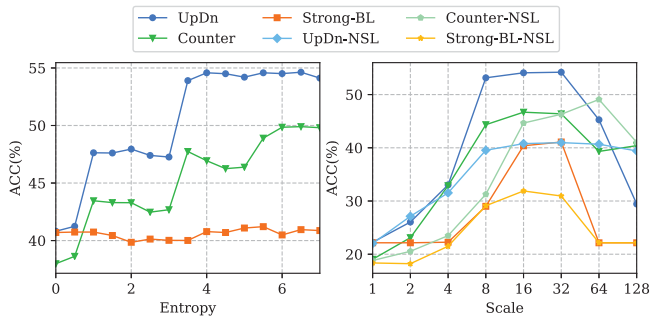


Figure 4: Test set accuracy with respect to different question type entropy and scales on the VQA-CP v2 dataset.

Model	Margin	Strong-BL	Counter	UpDn
Baseline	-	34.41	37.67	40.79
NSL	-	31.89	49.08	40.97
Fixed Margin	0.1	34.42	38.03	40.62
	0.3	34.38	38.21	40.68
	0.5	34.29	38.07	40.88
	0.7	34.22	38.15	40.98
	0.9	34.38	38.10	41.02
Adapted Margin	<i>adapted</i>	41.21	49.90	54.67

Table 3: Effectiveness validation of the proposed loss function on three baselines over the VQA-CP v2 dataset.

## Ablation Study

For a deeper understanding of our AdaVQA, we further provided detailed ablation studies on the VQA-CP v2 dataset. From the results in Tab. 3 and Fig. 4, we can see that:

- As shown in Tab. 3, when using the  $L2$  normalization on both the weight vector and the feature vector, the de facto NSL, the results are inconsistent amongst different methods. For example, the Counter with NSL surpasses the baseline with 11.41%, while Strong-BL with NSL even leads to a little bit deterioration of the performance.
- We then leveraged the fixed margin to yield effective feature discrimination, where the margins are tuned from 0.1 to 0.9 with a step size of 0.2. However, the results are not favorable (the improvement is limited), which validates the evidence that a fixed margin is not suitable for overcoming the language prior problem. In contrast, when replacing the fixed margin with our adapted one, the model can substantially outperform the one equipped with fixed margins, which additionally proves the superiority of AdaVQA.
- We investigated the different influence of the question type entropy and scales in Fig. 4. From this figure, we found that 1) an appropriate entropy should be considered, since a very large one will not bring more gains or even cause performance degradation; and 2) a too small scale is insufficient for learning the feature space, while too large ones will also lead to unsatisfactory results.

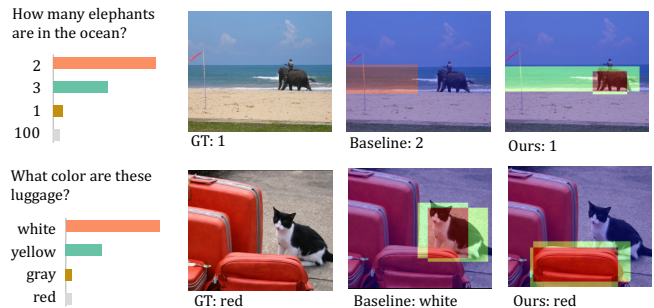


Figure 5: Visualization of the baselines method (UpDn) with and without our proposed loss function. The key answer distributions under the question’s corresponding question type are illustrated on the leftmost column, followed by the attention maps from baseline and ours on the last two columns.

## Case Study

Finally, we visualized two successful cases from our method and illustrated them in Fig. 5. Regarding the first case, as the answer 2 takes large proportion under the question type *how many* in the train set, the baseline model thus yields an answer 2 to this question. In contrast, our AdaVQA corrects this mistake, producing a more reasonable attention map (focusing on the elephant object). As for the second case, the baseline model wrongly predicting the answer *white* mainly attribute to *white* being more frequent under the question type *what color* in the train set. Besides, too much attention has been paid to the cat region, which contributes another reason to the incorrect answer, as the fur of the cat is white and black. On the contrary, our AdaVQA can guide the model to pay more attention to the target object - luggage, leading to the correct answer *red*.

## 5 Conclusion

We propose to tackle the language prior problem in VQA from a novel view of the answer feature space learning, which has not been explored in previous studies. To achieve this goal, we design an adapted margin cosine loss to discriminate the answers via properly characterizing the answer feature space. Concretely, for the given question, the frequent and sparse answers with respect to the corresponding question type are learned to span broader and tighter on the angular space, respectively. Extensive experiments over two benchmarks validate the effectiveness of the proposed loss function on three baselines.

It is worth mentioning that this work does not present an advanced model for pursuing the SOTA results in VQA. We instead expect that, with this adapted margin cosine loss, future research can focus more on the enhancement of the visual reasoning capability without too much influence of the language prior problem.

## Acknowledgments

This work is supported by the National Key Research and Development Project of New Generation Artificial Intelligence, No.:2018AAA0102502.

## References

- [Agrawal *et al.*, 2018] Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. Don’t just assume; look and answer: Overcoming priors for visual question answering. In *CVPR*, 2018.
- [Anderson *et al.*, 2018] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*, 2018.
- [Andreas *et al.*, 2016] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. Neural module networks. In *CVPR*, 2016.
- [Antol *et al.*, 2015] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *ICCV*, 2015.
- [Cadene *et al.*, 2019] Remi Cadene, Corentin Dancette, Matthieu Cord, Devi Parikh, et al. Rubi: Reducing unimodal biases for visual question answering. In *NeurIPS*, 2019.
- [Chen *et al.*, 2020] Long Chen, Xin Yan, Jun Xiao, Hanwang Zhang, Shiliang Pu, and Yueting Zhuang. Counterfactual samples synthesizing for robust visual question answering. In *CVPR*, 2020.
- [Clark *et al.*, 2019] Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. Don’t take the easy way out: ensemble based methods for avoiding known dataset biases. In *EMNLP*, 2019.
- [Fukui *et al.*, 2016] Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding. In *EMNLP*, 2016.
- [Goyal *et al.*, 2017] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *CVPR*, 2017.
- [Guo *et al.*, 2019] Yangyang Guo, Zhiyong Cheng, Liqiang Nie, Yibing Liu, Yinglong Wang, and Mohan Kankanhalli. Quantifying and alleviating the language prior problem in visual question answering. In *SIGIR*, 2019.
- [Guo *et al.*, 2020] Yangyang Guo, Liqiang Nie, Zhiyong Cheng, and Qi Tian. Loss-rescaling vqa: revisiting language prior problem from a class-imbalance view. In *arXiv preprint arXiv:2010.16010*, 2020.
- [Jing *et al.*, 2020] Chenchen Jing, Yuwei Wu, Xiaoxun Zhang, Yunde Jia, and Qi Wu. Overcoming language priors in vqa via decomposed linguistic representations. In *AAAI*, 2020.
- [Kazemi and Elqursh, 2017] Vahid Kazemi and Ali Elqursh. Show, ask, attend, and answer: A strong baseline for visual question answering. In *arXiv preprint arXiv:1704.03162*, 2017.
- [KV and Mittal, 2020] Gouthaman KV and Anurag Mittal. Reducing language biases in visual question answering with visually-grounded question encoder. In *arXiv preprint arXiv:2007.06198*, 2020.
- [Liu *et al.*, 2017] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *CVPR*, 2017.
- [Lu *et al.*, 2016] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. In *NeurIPS*, 2016.
- [Malinowski *et al.*, 2015] Mateusz Malinowski, Marcus Rohrbach, and Mario Fritz. Ask your neurons: A neural-based approach to answering questions about images. In *ICCV*, 2015.
- [Malinowski *et al.*, 2018] Mateusz Malinowski, Carl Doersch, Adam Santoro, and Peter Battaglia. Learning visual question answering by bootstrapping hard attention. In *ECCV*, 2018.
- [Marino *et al.*, 2019] Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *CVPR*, 2019.
- [Ramakrishnan *et al.*, 2018] Sainandan Ramakrishnan, Aishwarya Agrawal, and Stefan Lee. Overcoming language priors in visual question answering with adversarial regularization. In *NeurIPS*, 2018.
- [Selvaraju *et al.*, 2019] Ramprasaath R Selvaraju, Stefan Lee, Yilin Shen, Hongxia Jin, Shalini Ghosh, Larry Heck, Dhruv Batra, and Devi Parikh. Taking a hint: Leveraging explanations to make vision and language models more grounded. In *ICCV*, 2019.
- [Wang *et al.*, 2018a] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. In *CVPR*, 2018.
- [Wang *et al.*, 2018b] Peng Wang, Qi Wu, Chunhua Shen, Anthony Dick, and Anton van den Hengel. Fvqa: Fact-based visual question answering. *TPAMI*, 2018.
- [Wu and Mooney, 2019] Jialin Wu and Raymond Mooney. Self-critical reasoning for robust visual question answering. In *NeurIPS*, 2019.
- [Yang *et al.*, 2016] Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. Stacked attention networks for image question answering. In *CVPR*, 2016.
- [Yu *et al.*, 2017] Zhou Yu, Jun Yu, Jianping Fan, and Dacheng Tao. Multi-modal factorized bilinear pooling with co-attention learning for visual question answering. In *ICCV*, 2017.
- [Zhang *et al.*, 2018] Yan Zhang, Jonathon Hare, and Adam Prügel-Bennett. Learning to count objects in natural images for visual question answering. In *ICLR*, 2018.