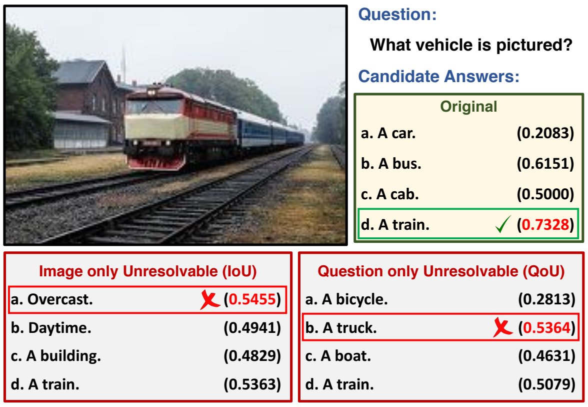
Being Negative but Constructively:
Lessons Learnt from Creating Better Visual Question Answering Datasets

Wei-Lun Chao[*], Hexiang Hu[*], Fei Sha
University of Southern California
Los Angeles, California, USA
weilunchao760414@gmail.com, hexiang.frank.hu@gmail.com, feisha@usc.edu

Abstract

Visual question answering (Visual QA) has attracted a lot of attention lately, seen essentially as a form of (visual) Turing test that artificial intelligence should strive to achieve. In this paper, we study a crucial component of this task: how can we design good datasets for the task? We focus on the design of multiple-choice based datasets where the learner has to select the right answer from a set of candidate ones including the target (i.e. the correct one) and the decoys (i.e. the incorrect ones). Through careful analysis of the results attained by state-of-the-art learning models and human annotators on existing datasets, we show that the design of the decoy answers has a significant impact on how and what the learning models learn from the datasets. In particular, the resulting learner can ignore the visual information, the question, or both while still doing well on the task. Inspired by this, we propose automatic procedures to remedy such design deficiencies. We apply the procedures to re-construct decoy answers for two popular Visual QA datasets as well as to create a new Visual QA dataset from the Visual Genome project, resulting in the largest dataset for this task. Extensive empirical studies show that the design deficiencies have been alleviated in the remedied datasets and the performance on them is likely a more faithful indicator of the difference among learning models. The datasets are released and publicly available via http://www.teds.usc.edu/website_vqa/.



Question:
What vehicle is pictured?
Candidate Answers:

Original	
a. A car.	(0.2083)
b. A bus.	(0.6151)
c. A cab.	(0.5000)
d. A train.	✓ (0.7328)

Image only Unresolvable (IoU)	
a. Overcast.	✗ (0.5455)
b. Daytime.	(0.4941)
c. A building.	(0.4829)
d. A train.	(0.5363)

Question only Unresolvable (QoU)	
a. A bicycle.	(0.2813)
b. A truck.	✗ (0.5364)
c. A boat.	(0.4631)
d. A train.	(0.5079)

Figure 1: An illustration of how the shortcuts in the Visual7W dataset (Zhu et al., 2016) should be remedied. In the original dataset, the correct answer "A train" is easily selected by a machine as it is far often used as the correct answer than the other decoy (negative) answers. (The numbers in the brackets are probability scores computed using eq. (2)). Our two procedures — QoU and IoU (cf. Sect. 4) — create alternative decoys such that both the correct answer and the decoys are highly likely by examining either the image or the question **alone**. In these cases, machines make mistakes unless they consider all information **together**. Thus, the alternative decoys suggested our procedures are better designed to gauge how well a learning algorithm can understand all information equally well.

1 Introduction

Multimodal information processing tasks such as image captioning (Farhadi et al., 2010; Ordonez et al., 2011; Xu et al., 2015) and visual question answering (Visual QA) (Antol et al., 2015) have gained a lot of attention recently. A number of significant advances in learning algorithms have been made, along with the development of nearly two dozens of datasets in this very active research domain. Among those datasets, popular ones include MSCOCO (Lin et al., 2014; Chen et al., 2015), Visual Genome (Krishna et al., 2017), VQA (Antol et al., 2015), and several others. The overarching objective is that a learning machine needs to go beyond understanding different modalities of information separately (such as image recognition alone) and to learn how to correlate them in order to perform well on those tasks.

[*]Equal contributions

To evaluate the progress on those complex and more AI-like tasks is however a challenging topic. For tasks involving language generation, developing an automatic evaluation metric is itself an open problem (Anderson et al., 2016; Kilickaya et al., 2017; Liu et al., 2016; Kafle and Kanan, 2017b). Thus, many efforts have concentrated on tasks such as *multiple-choice* Visual QA (Antol et al., 2015; Zhu et al., 2016; Jabri et al., 2016) or selecting the best caption (Hodosh et al., 2013; Hodosh and Hockenmaier, 2016; Ding et al., 2016; Lin and Parikh, 2016), where the selection accuracy is a natural evaluation metric.

In this paper, we study how to design high-quality multiple choices for the Visual QA task. In this task, the machine (or the human annotator) is presented with an image, a question and a list of candidate answers. The goal is to select the correct answer through a consistent understanding of the image, the question and each of the candidate answers. As in any multiple-choice based tests (such as GRE), designing what should be presented as negative answers — we refer them as *decoys* — is as important as deciding the questions to ask. We all have had the experience of exploiting the elimination strategy: *This question is easy — none of the three answers could be right so the remaining one must be correct!*

While a clever strategy for taking exams, such “shortcuts” prevent us from studying faithfully how different learning algorithms comprehend the meanings in images and languages (e.g., the quality of the embeddings of both images and languages in a semantic space). It has been noted that machines can achieve very high accuracies of selecting the correct answer without the visual input (i.e., the image), the question, or both (Jabri et al., 2016; Antol et al., 2015). Clearly, the learning algorithms have overfit on incidental statistics in the datasets. For instance, if the decoy answers have rarely been used as the correct answers (to any questions), then the machine can rule out a decoy answer with a binary classifier that determines whether the answers are in the set of the correct answers — note that this classifier does not need to examine the image and it just needs to memorize the list of the correct answers in the training dataset. See Fig. 1 for an example, and Sect. 3 for more and detailed analysis.

We focus on minimizing the impacts of exploiting such shortcuts. We suggest a set of principles for creating decoy answers. In light of the amount of human efforts in curating existing datasets for the Visual QA task, we propose two procedures that revise those datasets such that the decoy answers are better designed. In contrast to some earlier works, the procedures are fully automatic and do not incur additional human annotator efforts. We apply the procedures to revise both Visual7W (Zhu et al., 2016) and VQA (Antol et al., 2015). Additionally, we create new multiple-choice based datasets from COCOQA (Ren et al., 2015) and the recently released VQA2 (Goyal et al., 2017) and Visual Genome datasets (Krishna et al., 2017). The one based on Visual Genome becomes the largest multiple-choice dataset for the Visual QA task, with more than one million image-question-candidate answers triplets.

We conduct extensive empirical and human studies to demonstrate the effectiveness of our procedures in creating high-quality datasets for the Visual QA task. In particular, we show that machines need to use all three information (image, questions and answers) to perform well — any missing information induces a large drop in performance. Furthermore, we show that humans dominate machines in the task. However, given the revised datasets are likely reflecting the true gap between the human and the machine understanding of multimodal information, we expect that advances in learning algorithms likely focus more on the task itself instead of overfitting to the idiosyncrasies in the datasets.

The rest of the paper is organized as follows. In Sect. 2, we describe related work. In Sect. 3, we analyze and discuss the design deficiencies in existing datasets. In Sect. 4, we describe our automatic procedures for remedying those deficiencies. In Sect. 5 we conduct experiments and analysis. We conclude the paper in Sect. 6.

2 Related Work

Wu et al. (2017) and Kafle and Kanan (2017b) provide recent overviews of the status quo of the Visual QA task. There are about two dozens of datasets for the task. Most of them use real-world images, while some are based on synthetic ones. Usually, for each image, multiple questions and their corresponding answers are generated. This can be achieved either by human annotators, or with an automatic procedure that uses captions or question templates and detailed image annotations. We concentrate on 3 datasets: VQA (Antol et al., 2015), Visual7W (Zhu et al., 2016), and Visual Genome (Krishna et al., 2017). All of them use images from MSCOCO (Lin et al., 2014).

Besides the pairs of questions and correct answers, VQA, Visual7W, and visual Madlibs (Yu et al., 2015) provide decoy answers for each pair so that the task can be evaluated in multiple-choice selection accuracy. *What decoy answers to use* is the focus of our work.

In VQA, the decoys consist of human-generated plausible answers as well as high-frequency and random answers from the datasets. In Visual7W, the decoys are all human-generated plausible ones. Note that, humans generate those decoys by *only looking at the questions and the correct answers but **not** the images*. Thus, the decoys might be unrelated to the corresponding images. A learning algorithm can potentially examine the image alone and be able to identify the correct answer.

In visual Madlibs, the questions are generated with a limited set of question templates and the detailed annotations (e.g., objects) of the images. Thus, similarly, a learning model can examine the image alone and deduce the correct answer.

We propose automatic procedures to revise VQA and Visual7W (and to create new datasets based on COCOQA (Ren et al., 2015), VQA2 (Goyal et al., 2017), and Visual Genome) such that the decoy generation is carefully orchestrated to prevent learning algorithms from exploiting the shortcuts in the datasets by overfitting on incidental statistics. In particular, our design goal is that a learning machine needs to understand all the 3 components of an image-question-candidate answers triplet in order to make the right choice — ignoring either one or two components will result in drastic degradation in performance.

Our work is inspired by the experiments in (Jabri et al., 2016) where they observe that machines without looking at images or questions can still perform well on the Visual QA task. Others have also reported similar issues (Goyal et al., 2017; Zhang et al., 2016; Johnson et al., 2017; Agrawal et al., 2016; Kafle and Kanan, 2017a; Agrawal et al., 2018), though not in the multiple-choice setting. Our work extends theirs by providing more detailed analysis *as well as automatic procedures* to remedy those design deficiencies.

Besides Visual QA, VisDial (Das et al., 2017) and Ding et al. (2016) also propose automatic ways to generate decoys for the tasks of multiple-choice visual captioning and dialog, respectively.

Recently, Lin and Parikh (2017) study active learning for Visual QA: i.e., how to select informative image-question pairs (for acquiring annotations) or image-question-answer triplets for machines to "learn" from. On the other hand, our work further focuses on designing better datasets for "evaluating" a machine.

3 Analysis of Decoy Answers' Effects

In this section, we examine in detail the dataset Visual7W (Zhu et al., 2016), a popular choice for the Visual QA task. We demonstrate how the deficiencies in designing decoy questions impact the performance of learning algorithms.

In multiple-choice Visual QA datasets, a training or test example is a triplet that consists of an image I, a question Q, and a candidate answer set A. The set A contains a target T (the correct answer) and K decoys (incorrect answers) denoted by D. An IQA triplet is thus $\{\mathrm{I}, \mathrm{Q}, \mathrm{A} = \{\mathrm{T}, \mathrm{D}_1, \cdots, \mathrm{D}_K\}\}$. We use C to denote either the target or a decoy.

3.1 Visual QA models

We investigate how well a learning algorithm can perform when supplied with different modalities of information. We concentrate on the one hidden-layer MLP model proposed in (Jabri et al., 2016), which has achieved state-of-the-art results on the dataset Visual7W. The model computes a scoring function $f(c, i)$

$$f(c, i) = \sigma(\boldsymbol{U} \max(0, \boldsymbol{W} g(c, i)) + b) \qquad (1)$$

over a candidate answer c and the multimodal information i, where g is the joint feature of (c, i) and $\sigma(x) = 1/(1 + \exp(-x))$. The information i can be null, the image (I) alone, the question (Q) alone, or the combination of both (I+Q).

Given an IQA triplet, we use the penultimate layer of ResNet-200 (He et al., 2016) as visual features to represent I and the average WORD2VEC embeddings (Mikolov et al., 2013) as text features to represent Q and C. To form the joint feature $g(c, i)$, we just concatenate the features together. The candidate $c \in$ A that has the highest $f(c, i)$ score in prediction is selected as the model output.

We use the standard training, validation, and test splits of Visual7W, where each contains 69,817, 28,020, and 42,031 examples respectively.

Information used	Machine	Human
random	25.0	25.0
A	52.9	-
I + A	62.4	75.3
Q + A	58.2	36.4
I + Q + A	65.7	88.4

Table 1: Accuracy of selecting the right answers out of 4 choices (%) on the Visual QA task on Visual7W.

Each question has 4 candidate answers. The parameters of $f(c, i)$ are learned by minimizing the binary logistic loss of predicting whether or not a candidate c is the target of an IQA triplet. Details are in Sect. 5 and the Supplementary Material.

3.2 Analysis results

Machines find shortcuts Table 1 summarizes the performance of the learning models, together with the human studies we performed on a subset of 1,000 triplets (c.f. Sect. 5 for details). There are a few interesting observations.

First, in the row of "A" where only the candidate answers (and whether they are right or wrong) are used to train a learning model, the model performs significantly better than random guessing and humans (52.9% vs. 25%) — humans will deem each of the answers equally likely *without* looking at both the image and the question! Note that in this case, the information i in eq. (1) contains nothing. The model learns the specific statistics of the candidate answers in the dataset and exploits those. Adding the information about the image (i.e., the row of "I+A"), the machine improves significantly and gets close to the performance when all information is used (62.4% vs. 65.7%). There is a weaker correlation between the question and the answers as "Q+A" improves over "A" only modestly. This is expected. In the Visual7W dataset, the decoys are generated by human annotators as plausible answers to the questions without being shown the images — thus, many decoy answers do not have visual groundings. For instance, a question of "what animal is running?" elicits equally likely answers such as "dog", "tiger", "lion", or "cat", while an image of a dog running in the park will immediately rule out all 3 but the "dog", see Fig. 1 for a similar example. Thus, the performance of "I+A" implies that many IQA triplets can be solved by object, attribute or concept detection on the image, without understanding the questions. This is indeed the case also for humans — humans can achieve 75.3% by considering "I+A" and not "Q". Note that the difference between machine and human on "I+A" are likely due to their difference in understanding visual information.

Note that human improves significantly from "I+A" to "I+Q+A" with "Q" added, while the machine does so only marginally. The difference can be attributed to the difference in understanding the question and correlating with the answers between the two. Since each image corresponds to multiple questions or have multiple objects, solely relying on the image itself will not work well in principle. Such difference clearly indicates that in the Visual QA model, the language component is weak as the model cannot fully exploit the information in "Q", making a smaller relative improvement 5.3% (from 62.4% to 65.7%) where humans improved relatively 17.4%.

Shortcuts are due to design deficiencies We probe deeper on how the decoy answers have impacted the performance of learning models.

As explained above, the decoys are drawn from all plausible answers to a question, irrespective of whether they are visually grounded or not. We have also discovered that the targets (i.e., correct answers) are infrequently used as decoys.

Specifically, among the 69,817 training samples, there are 19,503 unique correct answers and each one of them is used about 3.6 times as correct answers to a question. However, among all the $69,817 \times 3 \approx 210K$ decoys, each correct answer appears 7.2 times on average, far below a chance level of 10.7 times ($210K \div 19,503 \approx 10.7$). This disparity exists in the test samples too. Consequently, the following rule, computing each answer's likelihood of being correct,

$$P(\text{correct}|\text{C}) = \begin{cases} 0.5, & \text{if C is never seen in training,} \\ \frac{\text{\# times C as target}}{\text{\# times C as target} + (\text{\# times C as decoys})/K}, & \text{otherwise,} \end{cases} \tag{2}$$

should perform well. Essentially, it measures how unbiased C is used as the target and the decoys. Indeed, it attains an accuracy of 48.73% on the test data, far better than the random guess and is close to the learning model using the answers' information only (the "A" row in Table 1).

Good rules for designing decoys Based on our analysis, we summarize the following guidance rules to design decoys: (1) **Question only Unresolvable (QoU)**. The decoys need to be equally plausible to the question. Otherwise, machines can rely on the correlation between the question and candidate answers to tell the target from decoys, even without the images. Note that this is a principle that is being followed by most datasets. (2) **Neutrality**. The decoys answers should be equally likely used as the correct answers. (3) **Image only Unresolvable (IoU)**. The decoys need to be plausible to the image. That is, they should appear in the image, or there exist questions so that the decoys can be treated as targets to the image. Otherwise, Visual QA can be resolved by objects, attributes, or concepts detection in images, even without the questions.

Ideally, each decoy in an IQA triplet should meet the three principles. **Neutrality** is comparably easier to achieve by *reusing terms in the whole set of targets as decoys.* On the contrary, a decoy may hardly meet **QoU** and **IoU** simultaneously[1]. However, as long as all decoys of an IQA triplet meet **Neutrality** and some meet **QoU** and others meet **IoU**, the triplet as a whole still achieves the three principles — a machine ignoring either images or questions will likely perform poorly.

4 Creating Better Visual QA Datasets

In this section, we describe our approaches of remedying design deficiencies in the existing datasets for the Visual QA task. We introduce two automatic and widely-applicable procedures to create new decoys that can prevent learning models from exploiting incident statistics in the datasets.

4.1 Methods

Main ideas Our procedures operate on a dataset that already contains image-question-target (IQT) triplets, i.e., we do not assume it has decoys already. For instance, we have used our procedures to create a multiple-choice dataset from the Visual Genome dataset which has no decoy. We assume that each image in the dataset is coupled with "multiple" QT pairs, which is the case in nearly all the existing datasets. Given an IQT triplet (I, Q, T), we create two sets of decoy answers.

- **QoU-decoys**. We search among all other triplets that have similar questions to Q. The **targets** of those triplets are then collected as the decoys for T. As the targets to similar questions are likely plausible for the question Q, QoU-decoys likely follow the rules of **Neutrality** and **Question only Unresolvable (QoU)**. We compute the average WORD2VEC (Mikolov et al., 2013) to represent a question, and use the cosine similarity to measure the similarity between questions.
- **IoU-decoys**. We collect the **targets** from other triplets of the *same* image to be the decoys for T. The resulting decoys thus definitely follow the rules of **Neutrality** and **Image only Unresolvable (IoU)**.

We then combine the triplet (I, Q, T) with QoU-decoys and IoU-decoys to form an IQA triplet as a training or test sample.

Resolving ambiguous decoys One potential drawback of automatically selected decoys is that they may be semantically similar, ambiguous, or rephrased terms to the target (Zhu et al., 2016). We utilize two filtering steps to alleviate it. First, we perform string matching between a decoy and the target, deleting those decoys that contain or are covered by the target (e.g., "daytime" vs. "during the daytime" and "ponytail" vs. "pony tail").

Secondly, we utilize the WordNet hierarchy and the Wu-Palmer (WUP) score (Wu and Palmer, 1994) to eliminate semantically similar decoys. The WUP score measures how similar two *word senses* are (in the range of $[0, 1]$), based on the depth of them in the taxonomy and that of their least common subsumer. We compute the similarity of two strings according to the WUP scores in a similar manner to (Malinowski and Fritz, 2014), in which the WUP score is used to evaluate Visual QA performance. We eliminate decoys that have higher WUP-based similarity to the target. We use the NLTK toolkit (Bird et al., 2009) to compute the similarity. See the Supplementary Material for more details.

Other details For QoU-decoys, we sort and keep for each triplet the top N (e.g., 10,000) similar triplets from the entire dataset according to the question similarity. Then for each triplet, we compute the WUP-based similarity of each potential decoy to the target successively, and accept those with similarity below 0.9 until we have K decoys. We choose 0.9 according to (Malinowski and Fritz, 2014). We also perform such a check among selected decoys to ensure they are not very similar to each other. For IoU-decoys, the potential decoys are sorted randomly. The WUP-based

[1] E.g., in Fig 1, for the question "What vehicle is pictured?", the only answer that meets both principles is "train", which is the correct answer instead of being a decoy.

similarity with a threshold of 0.9 is then applied to remove ambiguous decoys.

4.2 Comparison to other datasets

Several authors have noticed the design deficiencies in the existing databases and have proposed "fixes" (Antol et al., 2015; Yu et al., 2015; Zhu et al., 2016; Das et al., 2017). No dataset has used a procedure to generate IoU-decoys. We empirically show that how the IoU-decoys significantly remedy the design deficiencies in the datasets.

Several previous efforts have generated decoys that are similar in spirit to our **QoU**-decoys. Yu et al. (2015), Das et al. (2017), and Ding et al. (2016) automatically find decoys from similar questions or captions based on question templates and annotated objects, tri-grams and GLOVE embeddings (Pennington et al., 2014), and paragraph vectors (Le and Mikolov, 2014) and linguistic surface similarity, respectively. The later two are for different tasks from Visual QA, and only Ding et al. (2016) consider removing semantically ambiguous decoys like ours. Antol et al. (2015) and Zhu et al. (2016) ask humans to create decoys, given the questions and targets. As shown earlier, such decoys may disobey the rule of **Neutrality**.

Goyal et al. (2017) augment the VQA dataset (Antol et al., 2015) (by human efforts) with additional IQT triplets to eliminate the shortcuts (language prior) in the open-ended setting. Their effort is complementary to ours on the multiple-choice setting. Note that an extended task of Visual QA, visual dialog (Das et al., 2017), also adopts the latter setting.

5 Empirical Studies

5.1 Dataset

We examine our automatic procedures for creating decoys on five datasets. Table 2 summarizes the characteristics of the three datasets we focus on.

VQA Real (Antol et al., 2015) The dataset uses images from MSCOCO (Lin et al., 2014) under the same training/validation/testing splits to construct IQA triplets. Totally 614,163 IQA triplets are generated for 204,721 images. Each question has 18 candidate answers: in general 3 decoys are human-generated, 4 are randomly sampled, and 10 are randomly sampled frequent-occurring targets. *As the test set does not indicate the targets, our studies focus on the training and validation sets.*

Dataset	# of Images			# of triplets			# of decoys
Name	train	val	test	train	val	test	per triplet
VQA	83k	41k	81k	248k	121k	244k	17
Visual7W	14k	5k	8k	69k	28k	42k	3
VG	49k	19k	29k	727k	283k	433k	-

Table 2: Summary of Visual QA datasets.

Visual7W Telling (Visual7W) (Zhu et al., 2016) The dataset uses 47,300 images from MSCOCO (Lin et al., 2014) and contains 139,868 IQA triplets. Each has 3 decoys generated by humans.

Visual Genome (VG) (Krishna et al., 2017) The dataset uses 101,174 images from MSCOCO (Lin et al., 2014) and contains 1,445,322 IQT triplets. No decoys are provided. Human annotators are asked to write diverse pairs of questions and answers freely about an image or with respect to some regions of it. On average an image is coupled with 14 question-answer pairs. We divide the dataset into non-overlapping 50%/20%/30% for training/validation/testing. Additionally, we partition such that each portion is a "superset" of the corresponding one in Visual7W, respectively.

VQA2 (Goyal et al., 2017) and COCOQA (Ren et al., 2015) We describe the datasets and experimental results in the Supplementary Material.

Creating decoys We create 3 QoU-decoys and 3 IoU-decoys for every IQT triplet in each dataset, following the steps in Sect. 4.1. In the cases that we cannot find 3 decoys, we include random ones from the original set of decoys for VQA and Visual7W; for other datasets, we randomly include those from the top 10 frequently-occurring targets.

5.2 Setup

Visual QA models We utilize the MLP models mentioned in Sect. 3 for all the experiments. We denote **MLP-A**, **MLP-QA**, **MLP-IA**, **MLP-IQA** as the models using A (Answers only), Q+A (Question plus Answers), I+A (Image plus Answers), and I+Q+A (Image, Question and Answers) for multimodal information, respectively. The hidden-layer has 8,192 neurons. We use a 200-layer ResNet (He et al., 2016) to compute visual features which are 2,048-dimensional. The ResNet is pre-trained on ImageNet (Russakovsky et al., 2015). The WORD2VEC feature (Mikolov et al., 2013) for questions and answers are 300-dimensional, pre-trained on Google News[2]. The

[2] We experiment on using different features in the Supplementary Material.

parameters of the MLP models are learned by minimizing the binary logistic loss of predicting whether or not a candidate answer is the target of the corresponding IQA triplet. Please see the Supplementary Material for details on optimization.

We further experiment with a variant of the spatial memory network (denoted as **Attention**) (Xu and Saenko, 2016) and the **HieCoAtt** model (Lu et al., 2016) adjusted for the multiple-choice setting. Both models utilize the attention mechanism. Details are listed in the Supplementary Material.

Evaluation metric For VQA and VQA2, we follow their protocols by comparing the picked answer to 10 human-generated targets. The accuracy is computed based on the number of exactly matched targets (divided by 3 and clipped at 1). For others, we compute the accuracy of picking the target from multiple choices.

Decoy sets to compare For each dataset, we derive several variants: (1) **Orig**: the original decoys from the datasets, (2) **QoU**: **Orig** replaced with ones selected by our QoU-decoys generating procedure, (3) **IoU**: **Orig** replaced with ones selected by our IoU-decoys generating procedure, (4) **QoU +IoU**: **Orig** replaced with ones combining **QoU** and **IoU**, (5) **All**: combining **Orig**, **QoU**, and **IoU**.

User studies Automatic decoy generation may lead to ambiguous decoys as mentioned in Sect. 4 and (Zhu et al., 2016). We conduct a user study via Amazon Mechanic Turk (AMT) to test humans' performance on the datasets after they are remedied by our automatic procedures. We select 1,000 IQA triplets from each dataset. Each triplet is answered by three workers and in total 169 workers get involved. The total cost is $215 — the rate for every 20 triplets is $0.25. We report the average human performance and compare it to the learning models'. See the Supplementary Material for more details.

5.3 Results

The performances of learning models and humans on the 3 datasets are reported in Table 3, 4, and 5[3].

Method	Orig	IoU	QoU	IoU +QoU	All
MLP-A	52.9	27.0	34.1	17.7	15.6
MLP-IA	62.4	27.3	55.0	23.6	22.2
MLP-QA	58.2	84.1	40.7	37.8	31.9
MLP-IQA	65.7	84.1	57.6	52.0	45.1
HieCoAtt∗	63.9	-	-	51.5	-
Attntion∗	65.9	-	-	52.8	-
Human	88.4	-	-	84.1	-
Random	25.0	25.0	25.0	14.3	10.0

∗: based on our implementation or modification

Table 3: Test accuracy (%) on Visual7W.

Effectiveness of new decoys A better set of decoys will force learning models to integrate all 3 pieces of information — images, questions and answers — to make the correct selection from multiple-choices. In particular, they should prevent learning algorithms from exploiting shortcuts such that partial information is sufficient for performing well on the Visual QA task.

Table 3 clearly indicates that those goals have been achieved. With the Orig decoys, the relatively small gain from MLP-IA to MLP-IQA suggests that the question information can be ignored to attain good performance. However, with the IoU-decoys which require questions to help to resolve (as image itself is inadequate to resolve), the gain is substantial (from 27.3% to 84.1%). Likewise, with the QoU-decoys (question itself is not adequate to resolve), including images information improves from 40.7% (MLP-QA) substantially to 57.6% (MLP-IQA). Note that with the Orig decoys, this gain is smaller (58.2% vs. 65.7%).

It is expected that MLP-IA matches better QoU-decoys but not IoU-decoys, and MLP-QA is the other way around. Thus it is natural to combine these two decoys. What is particularly appealing is that MLP-IQA improves noticeably over models learned with partial information on the combined IoU +QoU-decoys (and "All" decoys[4]). Furthermore, using answer information only (MLP-A) attains about the chance-level accuracy.

On the VQA dataset (Table 4), the same observations hold, though to a lesser degree. On any of the IoU or QoU columns, we observe substan-

[3] We note that in Table 3, the 4.3% drop of the human performance on IoU +QoU, compared to Orig, is likely due to that IoU +QoU has more candidates (7 per question). Besides, the human performance on qaVG cannot be directly compared to that on the other datasets, since the questions on qaVG tend to focus on local image regions and are considered harder.

[4] We note that the decoys in Orig are not trivial, which can be seen from the gap between All and IoU +QoU. Our main concern on Orig is that for those questions that machines can accurately answer, they mostly rely on only partial information. This will thus hinder designing machines to fully comprehend and reason from multimodal information. We further experiment on random decoys, which can achieve **Neutrality** but not the other two principles, to demonstrate the effectiveness of our methods in the Supplementary Material.

Method	Orig	IoU	QoU	IoU +QoU	All
MLP-A	31.2	39.9	45.7	31.2	27.4
MLP-IA	42.0	39.8	55.1	34.1	28.7
MLP-QA	58.0	84.7	55.1	54.4	50.0
MLP-IQA	64.6	85.2	65.4	63.7	58.9
HieCoAtt∗	63.0	-	-	63.7	-
Attntion∗	66.0	-	-	66.7	-
Human	88.5†	-	-	89.0	-
Random	5.6	25.0	25.0	14.3	4.2

∗: based on our implementation or modification
†: taken from (Antol et al., 2015)

Table 4: Accuracy (%) on the validation set in VQA.

Method	IoU	QoU	IoU +QoU
MLP-A	29.1	36.2	19.5
MLP-IA	29.5	60.2	25.2
MLP-QA	89.3	45.6	43.9
MLP-IQA	89.2	64.3	58.5
HieCoAtt∗	-	-	57.5
Attntion∗	-	-	60.1
Human	-	-	82.5
Random	25.0	25.0	14.3

∗: based on our implementation or modification

Table 5: Test accuracy (%) on qaVG.

Datasets	Decoys	Best w/o	qaVG model	
		using qaVG	initial	fine-tuned
Visual7W	Orig	65.7	60.5	**69.1**
	IoU +QoU	52.0	58.1	**58.7**
	All	45.1	48.9	**51.0**
VQA	Orig	64.6	42.2	**65.6**
	IoU +QoU	63.7	47.9	**64.1**
	All	58.9	37.5	**59.4**

Table 6: Using models trained on qaVG to improve Visual7W and VQA (Accuracy in %).

tial gains when the complementary information is added to the model (such as MLP-IA to MLP-IQA). All these improvements are much more visible than those observed on the original decoy sets.

Combining both Table 3 and 4, we notice that the improvements from MLP-QA to MLP-IQA tend to be lower when facing IoU-decoys. This is also expected as it is difficult to have decoys that are simultaneously both IoU and QoU — such answers tend to be the target answers. Nonetheless, we deem this as a future direction to explore.

Differences across datasets Contrasting Visual7W to VQA (on the column IoU +QoU), we notice that Visual7W tends to have bigger improvements in general. This is due to the fact that VQA has many questions with "Yes" or "No" as the targets — the only valid decoy to the target "Yes" is "No", and vice versa. As such decoys are already captured by Orig of VQA ('Yes" and "No" are both top frequently-occurring targets), adding other decoy answers will not make any noticeable improvement. In Supplementary Material, however, we show that once we remove such questions/answers pairs, the degree of improvements increases substantially.

Comparison on Visual QA models As presented in Table 3 and 4, MLP-IQA is on par with or even outperforms **Attention** and **HieCoAtt** on the Orig decoys, showing how the shortcuts make it difficult to compare different models. By eliminating the shortcuts (i.e., on the combined IoU +QoU-decoys), the advantage of using sophisticated models becomes obvious (**Attention** outperforms MLP-IQA by 3% in Table 4), indicating the importance to design advanced models for achieving human-level performance on Visual QA.

For completeness, we include the results on the Visual Genome dataset in Table 5. This dataset has no "Orig" decoys, and we have created a multiple-choice based dataset **qaVG** from it for the task — it has over 1 million triplets, the largest dataset on this task to our knowledge. On the combined IoU +QoU-decoys, we again clearly see that machines need to use all the information to succeed.

With **qaVG**, we also investigate whether it can help improve the multiple-choice performances on the other two datasets. We use the MLP-IQA trained on qaVG with both IoU and QoU decoys to initialize the models for the Visual7W and VQA datasets. We report the accuracies before and after fine-tuning, together with the best results learned solely on those two datasets. As shown in Table 6, fine-tuning largely improves the performance, justifying the finding by Fukui et al. (2016).

5.4 Qualitative Results

In Fig. 2, we present examples of image-question-target triplets from **V7W**, **VQA**, and **VG**, together with our IoU-decoys (A, B, C) and QoU-decoys (D, E, F). G is the target. The predictions by the corresponding MLP-IQA are also included. Ignoring information from images or questions makes it extremely challenging to answer the triplet correctly, even for humans.

Our automatic procedures do fail at some triplets, resulting in ambiguous decoys to the targets. See Fig. 3 for examples. We categorized those failure cases into two situations.

- Our filtering steps in Sect. 4 fail, as observed in the top example. The WUP-based similarity re-

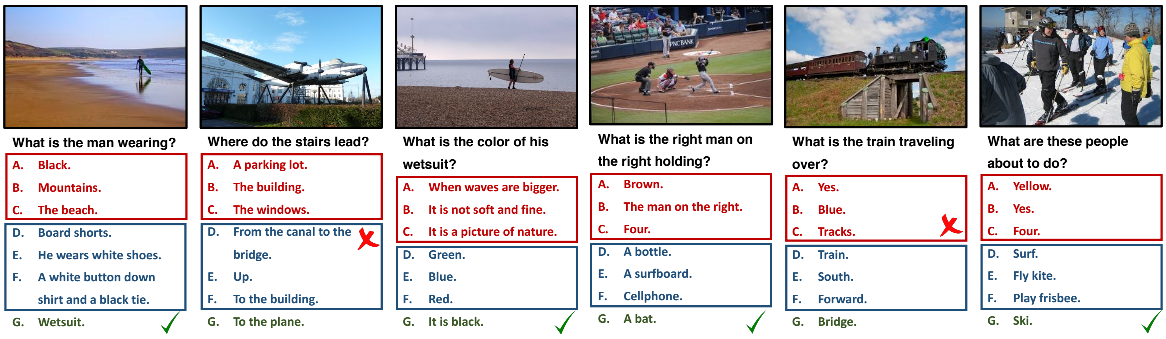


Figure 2: Example image-question-target triplets from Visual7W, VQA, and VG, together with our IoU-decoys (A, B, C.) and QoU-decoys (D, E, F). G is the target. Machine's selections are denoted by green ticks (correct) or red crosses (wrong).

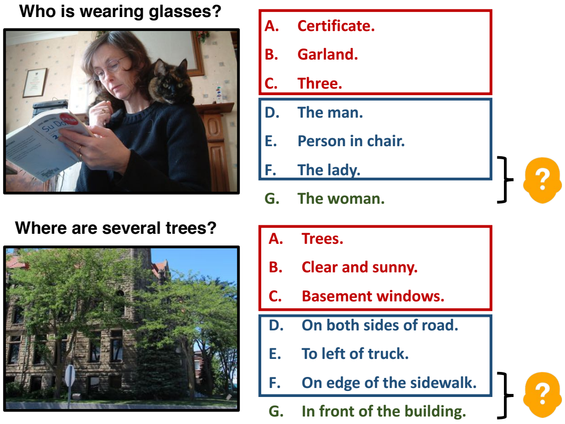


Figure 3: Ambiguous examples by our IoU-decoys (A, B, C) and QoU-decoys (D, E, F). G is the target. Ambiguous decoys F are marked.

lies on the WordNet hierarchy. For some semantically similar words like "lady" and "woman", the similarity is only 0.632, much lower than that of 0.857 between "cat" and "dog". This issue can be alleviated by considering alternative semantic measures by WORD2VEC or by those used in (Das et al., 2017; Ding et al., 2016) for searching similar questions.

- The question is ambiguous to answer. In the bottom example in Fig. 3, both candidates D and F seem valid as a target. Another representative case is when asked about the background of a image. In images that contain sky and mountains in the distance, both terms can be valid.

6 Conclusion

We perform detailed analysis on existing datasets for multiple-choice Visual QA. We found that the design of decoys can inadvertently provide "shortcuts" for machines to exploit to perform well on the task. We describe several principles of constructing good decoys and propose automatic procedures to remedy existing datasets and create new ones. We conduct extensive empirical studies to demonstrate the effectiveness of our methods in creating better Visual QA datasets. The remedied datasets and the newly created ones are released and available at http://www.teds.usc.edu/website_vqa/.

Acknowledgments

This work is partially supported by USC Graduate Fellowship, NSF IIS-1065243, 1451412, 1513966/1632803, 1208500, CCF-1139148, a Google Research Award, an Alfred. P. Sloan Research Fellowship and ARO# W911NF-12-1-0241 and W911NF-15-1-0484.

References

Aishwarya Agrawal, Dhruv Batra, and Devi Parikh. 2016. Analyzing the behavior of visual question answering models. In *EMNLP*.

Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. 2018. Don't just assume; look and answer: Overcoming priors for visual question answering. In *CVPR*.

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2016. Spice: Semantic propositional image caption evaluation. In *ECCV*.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *ICCV*.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. 2015. Microsoft coco captions:

Data collection and evaluation server. *arXiv preprint arXiv:1504.00325* .

Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José MF Moura, Devi Parikh, and Dhruv Batra. 2017. Visual dialog. In *CVPR*.

Nan Ding, Sebastian Goodman, Fei Sha, and Radu Soricut. 2016. Understanding image and text simultaneously: a dual vision-language machine comprehension task. *arXiv preprint arXiv:1612.07833* .

Ali Farhadi, Mohsen Hejrati, Mohammad Amin Sadeghi, Peter Young, Cyrus Rashtchian, Julia Hockenmaier, and David Forsyth. 2010. Every picture tells a story: Generating sentences from images. In *ECCV*.

Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. 2016. Multimodal compact bilinear pooling for visual question answering and visual grounding. In *EMNLP*.

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 *CVPR*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*.

Micah Hodosh and Julia Hockenmaier. 2016. Focused evaluation for image description with binary forced-choice tasks. In *ACL Workshop*.

Micah Hodosh, Peter Young, and Julia Hockenmaier. 2013. Framing image description as a ranking task: Data, models and evaluation metrics. *Journal of Artificial Intelligence Research* .

Allan Jabri, Armand Joulin, and Laurens van der Maaten. 2016. Revisiting visual question answering baselines. In *ECCV*.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. 2017. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *CVPR*.

Kushal Kafle and Christopher Kanan. 2017a. An analysis of visual question answering algorithms. In *ICCV*.

Kushal Kafle and Christopher Kanan. 2017b. Visual question answering: Datasets, algorithms, and future challenges. *Computer Vision and Image Understanding* .

Mert Kilickaya, Aykut Erdem, Nazli Ikizler-Cinbis, and Erkut Erdem. 2017. Re-evaluating automatic metrics for image captioning. In *EACL*.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV* .

Quoc V Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *ICML*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *ECCV*.

Xiao Lin and Devi Parikh. 2016. Leveraging visual question answering for image-caption ranking. In *ECCV*.

Xiao Lin and Devi Parikh. 2017. Active learning for visual question answering: An empirical study. *arXiv preprint arXiv:1711.01732* .

Chia-Wei Liu, Ryan Lowe, Iulian V Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *EMNLP*.

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2016. Hierarchical question-image co-attention for visual question answering. In *NIPS*.

Mateusz Malinowski and Mario Fritz. 2014. A multi-world approach to question answering about real-world scenes based on uncertain input. In *NIPS*.

Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In *NIPS*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*.

Vicente Ordonez, Girish Kulkarni, and Tamara L Berg. 2011. Im2text: Describing images using 1 million captioned photographs. In *NIPS*.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*.

Mengye Ren, Ryan Kiros, and Richard Zemel. 2015. Exploring models and data for image question answering. In *NIPS*.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015. ImageNet large scale visual recognition challenge. *IJCV* .

Qi Wu, Damien Teney, Peng Wang, Chunhua Shen, Anthony Dick, and Anton van den Hengel. 2017. Visual question answering: A survey of methods and datasets. *Computer Vision and Image Understanding* .

Zhibiao Wu and Martha Palmer. 1994. Verbs semantics and lexical selection. In *ACL*.

Huijuan Xu and Kate Saenko. 2016. Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. In *ECCV*.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C Courville, Ruslan Salakhutdinov, Richard S Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*.

Licheng Yu, Eunbyung Park, Alexander C Berg, and Tamara L Berg. 2015. Visual madlibs: Fill in the blank description generation and question answering. In *ICCV*.

Peng Zhang, Yash Goyal, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2016. Yin and yang: Balancing and answering binary visual questions. In *CVPR*.

Yuke Zhu, Oliver Groth, Michael Bernstein, and Li Fei-Fei. 2016. Visual7w: Grounded question answering in images. In *CVPR*.

Supplementary Material

In this Supplementary Material, we provide details omitted in the main text.

- Sect. A: Details on the MLP-based models and the attention-based models (Sect. 3.1 and 5.2 of the main text).
- Sect. B: WUPS-based similarity for filtering out ambiguous decoys (Sect. 4.1 of the main text).
- Sect. C: Detailed results on VQA (Antol et al., 2015) w/o question-answer (QA) pairs that have Yes/No as the targets (Sect. 5.3 of the main text).
- Sect. D: Experiments on VQA2 (Goyal et al., 2017) and COCOQA (Ren et al., 2015) (Sect. 5 of the main text).
- Sect. E: Details on user studies (Sect. 5.2 of the main text).
- Sect. F: Analysis on different question and answer embeddings (Sect. 5.2 of the main text).
- Sect. G: Analysis on random decoys (Sect. 5.3 of the main text).

A Details on the MLP-based models and the attention-based models

As mentioned in the main text, we benchmark the performance of popular Visual QA models on our remedied dataset. Here we provide the details about those models we experimented and its corresponding training configurations.

A.1 The simple MLP-based model

The one hidden-layer MLP model used in our experiments has 8,192 hidden units, exactly following (Jabri et al., 2016). It contains a batch normalization layer before ReLU, and a dropout layer after ReLU. We set the dropout rate to be 0.5.

The input to the model is the concatenated features of images, questions, and answers, as shown in Fig. 4. We change all characters to lowercases and all integer numbers within $[0, 10]$ to words before computing WORD2VEC. We perform ℓ_2 normalization to features of each information before concatenation.

A.2 A variant of SMem (Attention*)

In the main text we experiment with a straightforward attention model similar to the spatial memory network (SMem) (Xu and Saenko, 2016), as shown in Fig. 5 (a). Instead of computing the visual attention for each word in the question, we directly compute the visual attention for the entire question using its average WORD2VEC embedding. We then concatenate the resulting visual features with the feature of the question and a candidate answer (in the same way as the MLP-based model in Sect. A.1) as the input to train a one-hidden-layer MLP for binary classification.

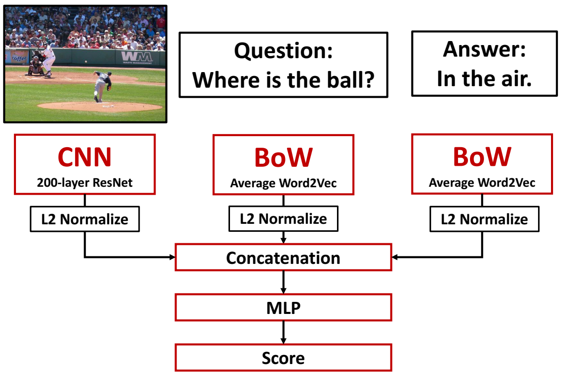


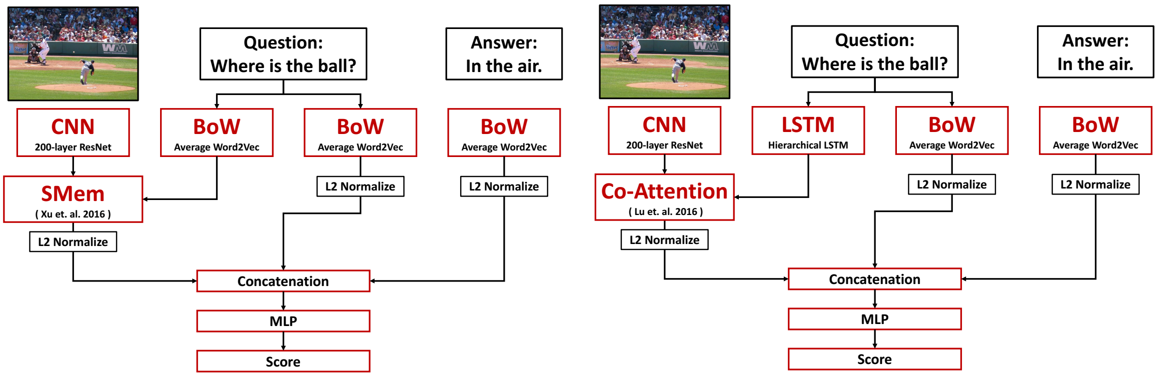
Figure 4: Illustration of MLP-based models.

A.3 A variant of HieCoAtten (HieCoAtten*)

Beyond the Attention*, we also experimented HieCoAtt*, a variant of the model proposed by Lu et al. (2016) (as shown in Fig. 5 (b)). Our model inherits all components in (Lu et al., 2016) that are related to computing the joint multi-modal embedding (from images and questions). To adapt to the multiple-choice setting, we discard the multi-way classifier in the original HieCoAtt and use its penultimate activations as feature for images. Similarly, we then concatenate this together with the features of questions and candidate answers, and input it to a one-hidden-layer MLP, following exact the configuration as Sect. A.1.

A.4 Optimization

We train all our models using stochastic gradient based optimization method with mini-batch size of 100, momentum of 0.9, and the stepped learning rate policy: the learning rate is divided by 10 after every M mini-batches. We set the initial learning rate to be 0.01 (we further consider 0.001 for the case of fine-tuning in Sect. 5.3 of the main text). For each model, we train with at most 600,000 iterations. We treat M and the number of iterations as hyper-parameters of training. We tune the hyper-parameters on the validation set.



(a) Attention* (SMem + MLP)

(b) HieCoAtt* (HieCoAtt + MLP)

Figure 5: Illustration of attention-based Visual QA models.

Within each mini-batch, we sample 100 IQA triplets. For each triplet, we randomly choose to use QoU-decoys or IoU-decoys when training on IoU +QoU, or QoU-decoys or IoU-decoys or Orig when training on All. We then take the target and **3** decoys for each triplet to train the binary classifier (i.e., minimize the logistic loss). Specifically on VQA, which has 17 Orig decoys for a triplet, we randomly choose 3 decoys out of them. That is, 100 triplets in the mini-batch corresponds to 400 examples with binary labels. This procedure is to prevent *unbalanced training*, where machines simply learn to predict the dominant label, as suggested by Jabri et al. (2016).

We note that in all the experiments in the main text, we use the *same type of decoy sets for training and testing*.

B WUP-based similarity for filtering out ambiguous decoys

We use the Wu-Palmer (WUP) score (Wu and Palmer, 1994), which characterizes the *word sense* similarity, to filter out ambiguous decoys to the target (correct answer). The WUP score is computed based on the WordNet hierarchy. Essentially, it measures the similarity of two *nodes* (i.e., synsets) in the hierarchy. As a *word* might correspond to multiple nodes, we measure the word similarity as follows:

$$WUP(w_1, w_2) = \max_{(n_1,n_2)\in N_1\times N_2} WUP(n_1, n_2), \tag{3}$$

where N_1 and N_2 are the sets of nodes that words w_1 and w_2 correspond to, respectively. That is, the word similarity is based on the most similar pair of nodes from both words. We consider only the NOUN and ADJ nodes for tractable computation.

Since a candidate answer may contain more than one word (i.e., a word sequence), we compute the similarity between two word sequences WS_1 and WS_2 as follows

$$\begin{aligned} &WUP(WS_1, WS_2) = \\ &\max\{\prod_{w_1\in WS_1} \max_{w_2\in WS_2} WUP(w_1, w_2), \qquad (4)\\ &\prod_{w_2\in WS_2} \max_{w_1\in WS_1} WUP(w_1, w_2)\}. \end{aligned}$$

This formulations is highly similar to the one proposed by Malinowski and Fritz et al. (2014) for evaluating open-ended Visual QA. The main difference is that we use “max” rather than “min” to compute the final score. Note that our purpose of using the WUP score is to filter out ambiguous decoys to the target. For example, we consider “a cute cat” to be ambiguous to “cat”. Using eq. (4) gives a similarity 1, which can not be achieved by taking “min”.

B.1 Analysis on the coverage

Among all the 139,868 IQA triplets in Visual7W, the target answers of 137,557 of them (98%) can find corresponding nodes in the WordNet hierarchy, so the scores can be computed. For VQA, the ratio is 97%. For qaVG, the ratio is 99%.

C Detailed results on VQA w/o QA pairs that have Yes/No as the targets

As mentioned in Sect. 5.3 of the main text, the validation set of VQA contains 45,478 QA pairs (out of totally 12,1512 pairs) that have Yes or No as the correct answers. The only reasonable decoy to Yes

Method	Orig	IoU	QoU	IoU +QoU	All
MLP-A	28.8	42.9	34.5	23.6	15.8
MLP-IA	43.0	44.8	53.2	35.5	28.5
MLP-QA	45.8	80.7	39.3	38.2	31.9
MLP-IQA	55.6	81.8	56.6	53.7	46.5
HieCoAtt∗	54.8	-	-	55.6	-
Attention∗	58.5	-	-	58.6	-
Human-IQA	-	-	-	85.5	-
Random	5.6	25.0	25.0	14.3	4.2

∗: based on our implementation or modification

Table 7: Accuracy (%) on VQA$^-$-2014val, which contains 76,034 triplets.

is No, and vice versa — any other decoy could be easily recognized in principle. Since both of them are among top 10 frequently-occurring answers, they are already included in the Orig decoys — our IoU-decoys and QoU-decoys can hardly make noticeable improvement. We thus remove all those pairs (denoted as Yes/No QA pairs) to investigate the improvement on the remaining pairs, for which having multiple choices makes sense. We denote the subset of VQA as VQA$^-$ (we remove Yes/No pairs in both training and validation set).

We conduct the same experiments as in Sect. 5.3 of the main text on VQA$^-$. Table 7 summarizes the machines' as well as humans' results. Compared to Table 4 of the main text, most of the results drop, which is expected as those removed Yes/No pairs are considered simpler and easier ones — their *effective* random chance is 50%. The exception is for the MLP-IA models, which performs roughly the same or even better on VQA$^-$, suggesting that Yes/No pairs are somehow difficult to MLP-IA. This, however, makes sense since without the questions (e.g., those start with "Is there a ..." or "Does the person ..."), a machine cannot directly tell if the correct answer falls into Yes or No, or others.

We see that on VQA$^-$, the improvement by our IoU-decoys and QoU-decoys becomes significant. The gain brought by images on QoU (from 39.3% to 56.6%) is much larger than that on Orig (from 45.8% to 55.6%). Similarly, the gain brought by questions on IoU (from 44.8% to 81.8%) is much larger than that on Orig (from 43.0% to 55.6%). After combining IoU-decoys and QoU-decoys as in IoU +QoU and All, the improvement by either including images to MLP-QA or including questions to MLP-IA is noticeable higher than that on Orig. Moreover, even with only 6 decoys, the performance by MLP-A on IoU +QoU is already lower than that on Orig, which has 17 decoys, demonstrating the effectiveness of our decoys in preventing machines from overfitting to the incidental statistics. These observations together demonstrate how our proposed ways for creating decoys improve the quality of multiple-choice Visual QA datasets.

D More experimental results on VQA2 and COCOQA

D.1 Dataset descriptions

COCOQA (Ren et al., 2015) This dataset contains in total 117,684 auto-generated IQT triplets with no decoy answers. Therefore, we create decoys using our proposed approach and follow the original data split, leading to a training set and a testing set with 78,736 IQA triplets and 38,948 IQA triplets, respectfully.

VQA2 (Goyal et al., 2017) VQA2 is a successive dataset of VQA, which pairs each IQT triplet with a complementary one to reduce the correlation between questions and answers. There are 443,757 training IQT triplets and 214,354 validation IQT triplets, with no decoys. We generate decoys using our approach and follow the original data split to organize the data. We do not consider the test split as it does not indicate the targets (correct answers).

D.2 Experimental results

For both datasets, we conduct the same experiments as in Sect. 5.3 of the main text using the MLP-based models. As shown in Table 8, we clearly see that with only answers being visible to the model (MLP-A), the performance is close to random (on the column of IoU +QoU-decoys), and far from observing all three sources of information (MLP-IQA). Meanwhile, models that can observe either images and answers (MLP-IA) or questions and answers (MLP-QA) fail to predict as good as the model that observe all three sources of information. Results in Table 9 also shows a similar trend. These empirical observations meet our expectation and again verify the effectiveness of our proposed methods for creating decoys.

Besides, we also perform a more in-depth experiment on VQA2, removing triplets with Yes/No as the target. We name this subset as VQA2$^-$. Table 10 shows the experimental results on VQA2$^-$. Comparing to Table 9, we see that the overall performance for each model decreases as the dataset

Method	IoU	QoU	IoU +QoU
MLP-A	70.3	31.7	26.6
MLP-IA	73.4	73.3	60.7
MLP-QA	91.5	52.5	51.4
MLP-IQA	93.1	78.3	75.9
Random	25.0	25.0	14.3

Table 8: Test accuracy (%) on COCOQA.

Method	IoU	QoU	IoU +QoU
MLP-A	37.7	41.9	27.7
MLP-IA	37.9	54.4	30.5
MLP-QA	84.2	48.3	48.1
MLP-IQA	86.3	63.0	61.1
Random	25.0	25.0	14.3

Table 9: Test accuracy (%) on VQA2-2017val.

becomes more challenging on average. Specifically, the model that observes question and answer on VQA2$^-$ performs much worse than that on VQA2 (37.2% vs. 48.1%).

E Details on user studies

As mentioned in Sect. 5.2 of the main text, we provide details on user studies. Fig. 6 shows our user interface. We perform the studies using Amazon Mechanic Turk (AMT) on Visual7W (Zhu et al., 2016), VQA (Antol et al., 2015) and Visual Genome (VG) (Krishna et al., 2017). We mainly evaluate on our IoU-decoys and QoU-decoys (combined together).

For each dataset, we randomly sample 1,000 image-question-target triplets together with the corresponding IoU-decoys and QoU-decoys to evaluate human performance. For each of these triplets, three workers are assigned to select the most correct candidate answer according to the image and the question. We compute the average accuracy of these workers and report them in Table 3, 4 and 5 of the main text and Table 7.

We also conduct human evaluation using the Orig decoys of Visual7W so as to investigate the difference between human-generated and automatically generated decoys. We also study how humans will perform given only partial information (i.e., images + candidate answers or questions + candidate answers), again using the Orig decoys of Visual7W. The corresponding interfaces are shown in Fig. 7 and 8. For these studies, we use the same set of 1,000 triplets used to evaluate our created decoys for fair comparison. We make sure that no worker works on the same triplet across the four studies on Visual7W. Results are reported in Table 1 of the main text.

Method	IoU	QoU	IoU +QoU
MLP-A	39.8	33.7	21.3
MLP-IA	40.3	53.0	31.0
MLP-QA	84.8	37.6	37.2
MLP-IQA	85.9	56.1	53.8
Random	25.0	25.0	14.3

Table 10: Test accuracy (%) on VQA2$^-$-2017val, which contains 134,813 triplets.

Method	GloVe	Translation	word2vec
MLP-A	18.0	18.0	17.7
MLP-IA	23.6	23.2	23.6
MLP-QA	38.1	38.3	37.8
MLP-IQA	52.5	51.4	52.0
Random	14.3	14.3	14.3

Table 11: Test accuracy (%) on Visual7W, comparing different embeddings for questions and answers. The results are reported for the IoU +QoU-decoys.

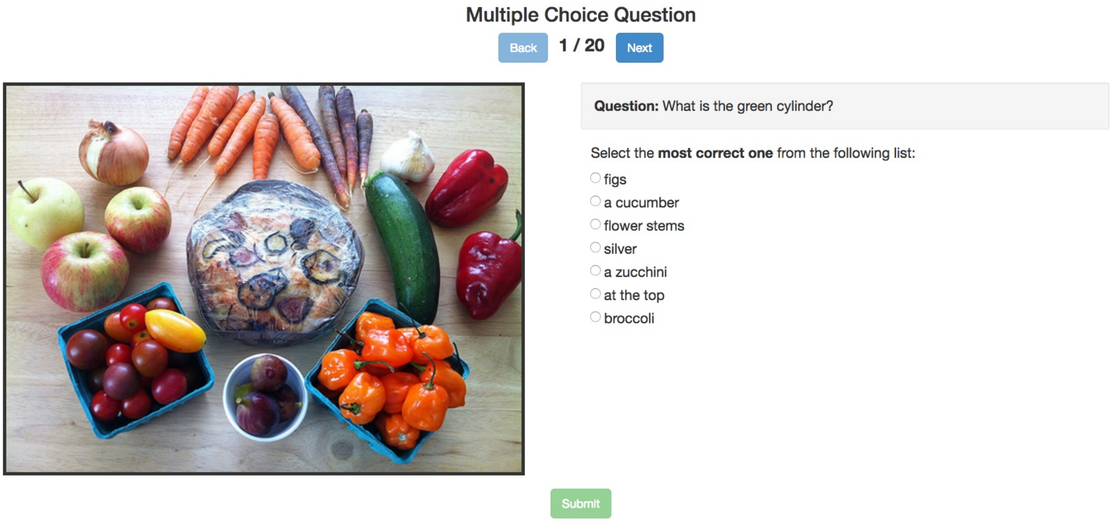
In summary, 169 workers are involved in our studies. The total cost is $215 — the rate for every 20 triplets is $0.25. On our IoU-decoys and QoU-decoys, humans achieve 84.1%, 89.0%, and 82.5% on Visual7W, VQA, and VG, respectively. Compared to the human performance on the Orig decoys that involve human effort in creation (i.e., 88.4% on Visual7W, and 88.5% on VQA as reported in (Antol et al., 2015)), these results suggest that the ways we create the decoys and the filtering steps mentioned in Sect. 4.2 lead to high-quality datasets with limited ambiguity.

F Analysis on different question and answer embeddings

We consider GloVe (Pennington et al., 2014) and the embedding learned from translation (McCann et al., 2017) on both question and answer embeddings. The results on Visual7W (IoU + QoU, compared to Table 3 of the main text that uses word2vec) are in Table 11. We do not observe significant difference among different embeddings, which is likely due to that both the questions and answers are short (averagely 7 words for questions and 2 for answers).

G Analysis on random decoys

We conduct the analysis on sampling random decoys, instead of our IoU-decoys and QoU-decoys, on Visual7W. We collect 6 additional random decoys for each **Orig** IQA triplet so the answer set will contain 10 candidates, the same as **All** in Table 3 of the main text. We consider two strategies: (A) uniformly random decoys from unique correct



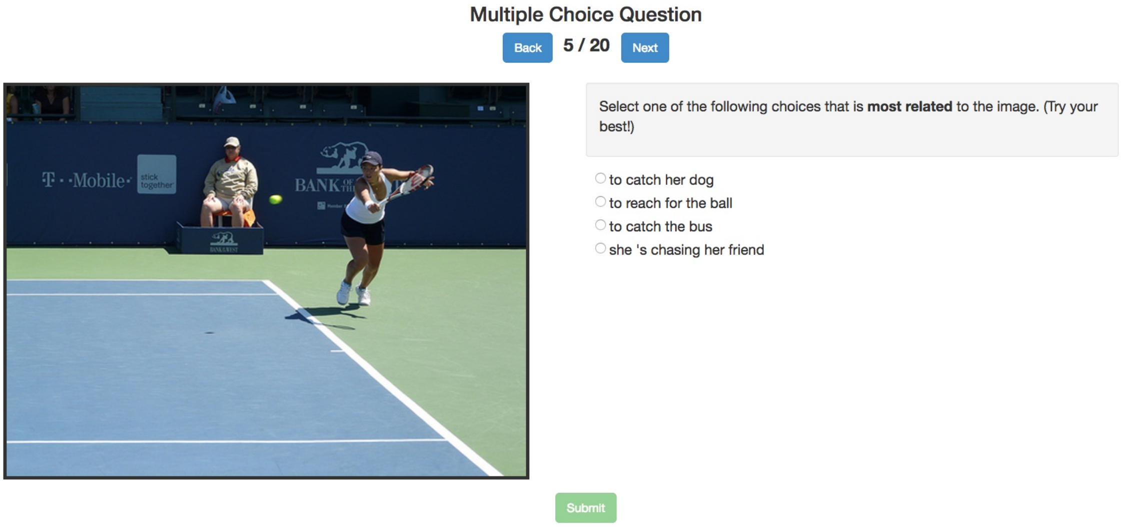
Multiple Choice Question

Question: What is the green cylinder?

Select the **most correct one** from the following list:

○ figs
○ a cucumber
○ flower stems
○ silver
○ a zucchini
○ at the top
○ broccoli

Figure 6: User interface for human evaluation on Visual7W (IoU-decoys+QoU-decoys).

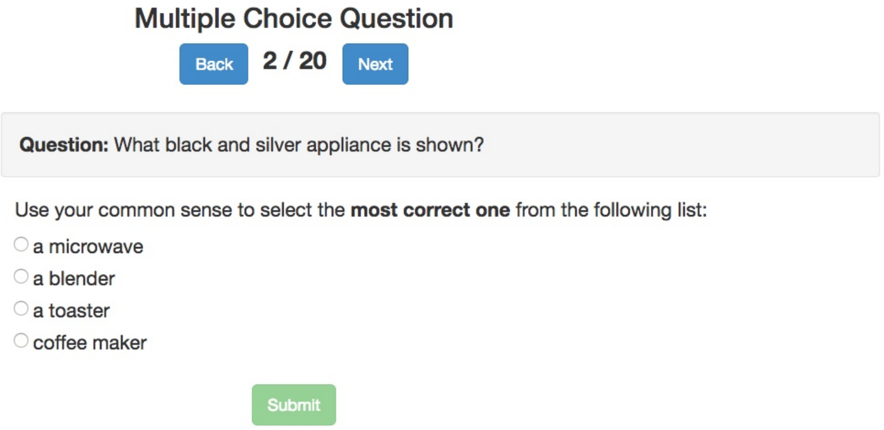


Multiple Choice Question

Select one of the following choices that is **most related** to the image. (Try your best!)

○ to catch her dog
○ to reach for the ball
○ to catch the bus
○ she 's chasing her friend

Figure 7: User interface for human evaluation on Visual7W (Orig decoys), where questions are blocked.



Multiple Choice Question

Question: What black and silver appliance is shown?

Use your common sense to select the **most correct one** from the following list:

○ a microwave
○ a blender
○ a toaster
○ coffee maker

Figure 8: User interface for human evaluation on Visual7W (Orig decoys), where images are not blocked.

Method	(A)	(B)	All
MLP-A	39.6	11.6	15.6
MLP-IA	53.4	40.3	22.2
MLP-QA	52.3	50.3	31.9
MLP-IQA	61.5	60.2	45.1
Random	10.0	10.0	10.0

Table 12: Test accuracy (%) on Visual7W, comparing different random decoy strategies to our methods: (A) Orig + uniformly random decoys from unique correct answers, (B) Orig + weighted random decoys w.r.t. their frequencies, and All (Orig+IoU +QoU).

answers, and (B) weighted random decoys w.r.t. their frequencies. The results are in Table 12. We see that different random strategies lead to drastically different results. Moreover, compared to the **All** column in Table 3 of the main text, we see that our methods lead to a larger relative gap between MLP-IQA to MLP-IA and MLP-QA than both random strategies, demonstrating the effectiveness of our methods in creating decoys.