Supplementary Material:
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

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

- Sect. S1: Details on the MLP-based models and the attention-based models (Sect. 3.1 and 5.2 of the main text).
- Sect. S2: WUPS-based similarity for filtering out ambiguous decoys (Sect. 4.1 of the main text).
- Sect. S3: 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. S4: Experiments on VQA2 (Goyal et al., 2017) and COCOQA (Ren et al., 2015) (Sect. 5 of the main text).
- Sect. S5: Details on user studies (Sect. 5.2 of the main text).
- Sect. S6: Analysis on different question and answer embeddings (Sect. 5.2 of the main text).
- Sect. S7: Analysis on random decoys (Sect. 5.3 of the main text).

S1 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.

S1.1 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.

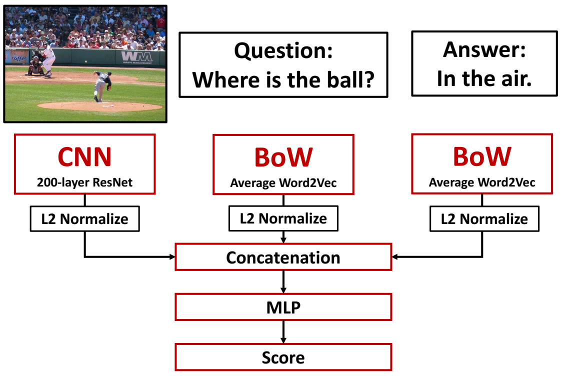


Figure F1: Illustration of MLP-based models.

The input to the model is the concatenated features of images, questions, and answers, as shown in Fig. F1. 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.

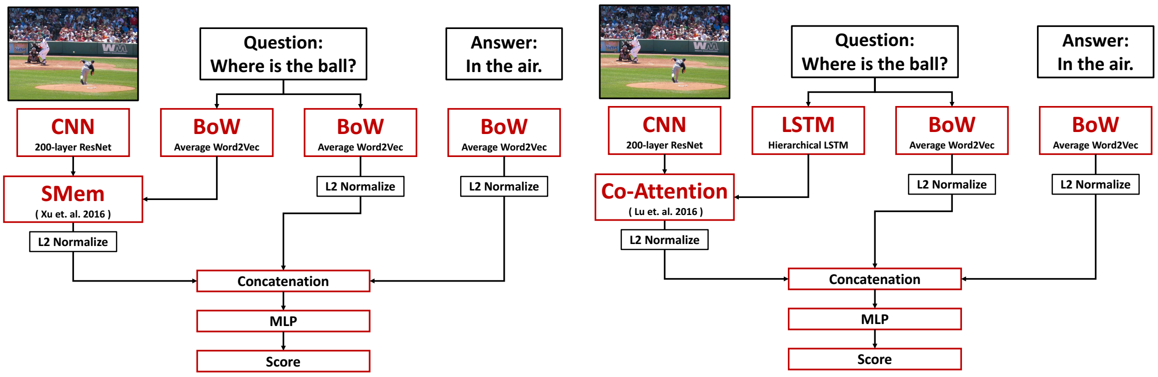
S1.2 A variant of SMem (Attention*)

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. F2 (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. S1.1) as the input to train a one-hidden-layer MLP for binary classification.

S1.3 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. F2 (b)). Our model

[*] Equal contributions



(a) Attention* (SMem + MLP)

(b) HieCoAtt* (HieCoAtt + MLP)

Figure F2: Illustration of attention-based Visual QA models.

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. S1.1.

S1.4 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.

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*.

S2 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:

$$WUP(w_1, w_2) = \max_{(n_1,n_2)\in N_1\times N_2} WUP(n_1, n_2), \tag{1}$$

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

$$\begin{aligned} &WUP(WS_1, WS_2) = \\ &\max\{\prod_{w_1\in WS_1} \max_{w_2\in WS_2} WUP(w_1, w_2), \\ &\quad \prod_{w_2\in WS_2} \max_{w_1\in WS_1} WUP(w_1, w_2)\}. \end{aligned} \tag{2}$$

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. (2) gives a similarity 1, which can not be achieved by taking "min".

S2.1 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%.

S3 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 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 T1 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.

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 T1: Accuracy (%) on VQA^{-}-2014val, which contains 76,034 triplets.

S4 More experimental results on VQA2 and COCOQA

S4.1 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).

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 T2: 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 T3: Test accuracy (%) on VQA2-2017val.

S4.2 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 T2, 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 T3 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 T4 shows the experimental results on VQA2$^-$. Comparing to Table T3, we see that the overall performance for each model decreases as the dataset 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%).

S5 Details on user studies

As mentioned in Sect. 5.2 of the main text, we provide details on user studies. Fig. F3 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 T1.

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

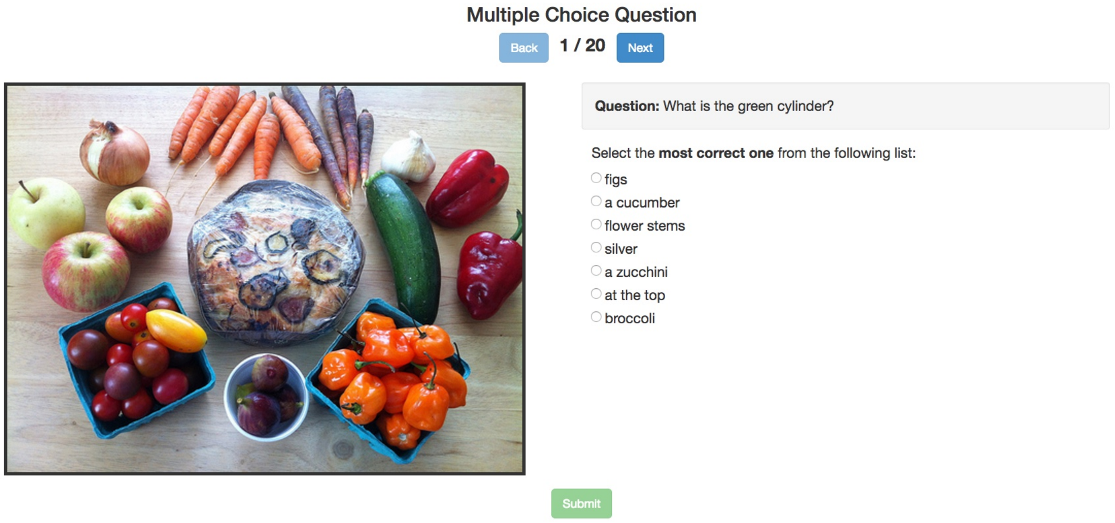
Table T4: Test accuracy (%) on VQA2$^-$-2017val, which contains 134,813 triplets.

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. F4 and F5. 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.

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.

S6 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 T5. We do not ob-



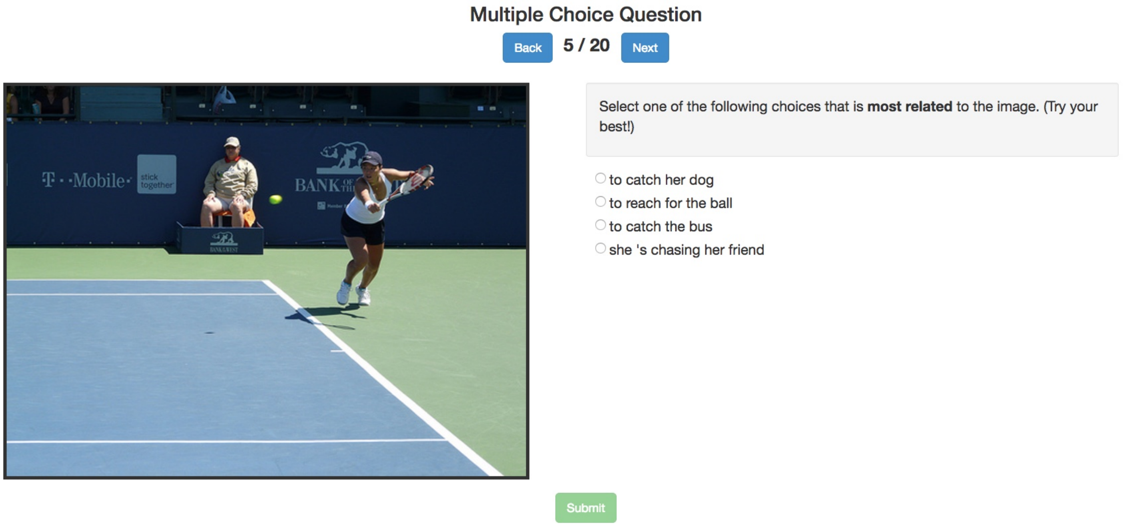
Multiple Choice Question

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 F3: User interface for human evaluation on Visual7W (IoU-decoys+QoU-decoys).

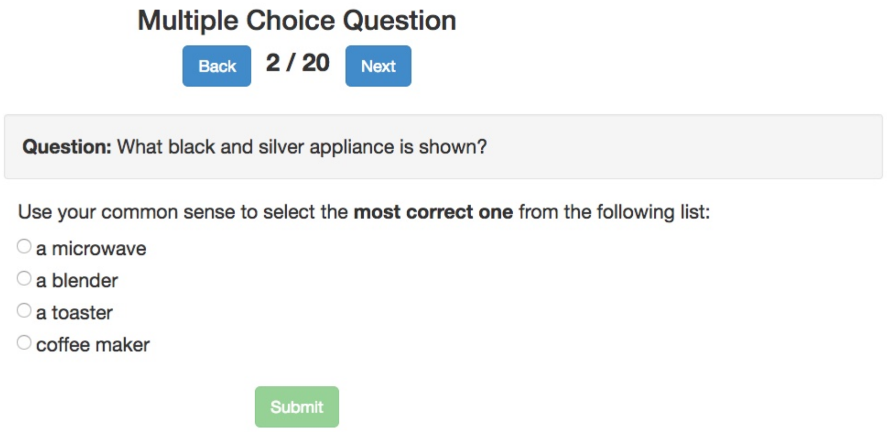


Multiple Choice Question

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 F4: 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 F5: User interface for human evaluation on Visual7W (Orig decoys), where images are not blocked.

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 T5: Test accuracy (%) on Visual7W, comparing different embeddings for questions and answers. The results are reported for the IoU +QoU-decoys.

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 T6: 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).

serve 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).

S7 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 answers, and (B) weighted random decoys w.r.t. their frequencies. The results are in Table T6. 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.

References

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

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*.

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

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* .

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*.

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*.

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*.

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