Augmenting Image Question Answering Dataset by Exploiting Image Captions

Masashi Yokota, Hideki Nakayama
Graduate School of Information Science and Technology,
The University of Tokyo
Tokyo, JP
{yokota, nakayama}@nlab.ci.i.u-tokyo.ac.jp

Abstract

Image question answering (IQA) is one of the tasks that need rich resources, i.e. supervised data, to achieve optimal performance. However, because IQA is a challenging task that handles complex input and output information, the cost of naive manual annotation can be prohibitively expensive. On the other hand, it is thought to be relatively easy to obtain relevant pairs of an image and text in an unsupervised manner (e.g., crawling Web data). Based on this expectation, we propose a framework to augment training data for IQA by generating additional examples from unannotated pairs of an image and captions. The important constraint that a generated IQA example must satisfy is that its answer must be inferable from the corresponding image and question. To satisfy this, we first select a possible answer for a given image by randomly extracting an answer from corresponding captions. Then we generate the question from the triplets of the image, captions and fixed answer. In experiments, we test our method on the Visual Genome dataset varying the ratio of seed supervised data and demonstrate its effectiveness.

Keywords: Semi-Supervised Learning, Self-labeling, Image Question Generation, Image Question Answering

1. Introduction

The objective of image question answering (IQA) is to predict a correct answer given an image and a question. To achieve a satisfactory performance, we generally require a large amount of human-annotated data which is expensive to prepare. One of the possible ways to mitigate the burden of collecting annotations is generating a pseudo dataset, which is one of the effective approaches in semi-supervised learning. Although this approach has been well-known to benefit a variety of tasks such as image classification (Lee, 2013), neural machine translation (He et al., 2016), and reading comprehension (Yang et al., 2017), its effectiveness has not yet been demonstrated for IQA.

In this study, we address the resource problem in IQA by proposing a framework of pseudo data generation exploiting captions as auxiliary information. In IQA, a generated pseudo dataset that comprises triplets consisting of an image I, question Q, and answer A should satisfy an important constraint: the components of a triplet (I, Q, A) must be relevant to each other. In other words, the answer A must be inferable from an image-question pair (I, Q). We address the problem of component relevancy by focusing on text captions C as the additional information. Our expectation is that pairs of mutually relevant image and captions are relatively easily obtainable from the Web (e.g. Wikipedia, BBC News, etc.). These captions are expected to contain information in images such as object colors, human actions, relationships between objects, etc. We first fix the answer A by sampling a token from captions as a possible answer for an image, and then generate a question conditioned by the triplet (I, C, A). For example, as shown in Fig. 1, we randomly select the token "bat" from the caption "A man holding a baseball bat." and then use the token as an answer. Finally, we generate the question "What is the man holding?" using the given image, caption and selected answer.

Our overall semi-supervised IQA framework consists of three phases:

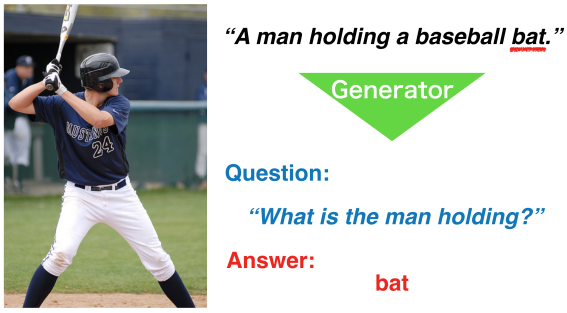

Figure 1: The example of a generated IQA triplet. The answer, underlined in red, is extracted from the given caption "A man holding a baseball bat." The generator produces the question using the image, caption and extracted answer.

(1) training an image question generation (IQG) model with a small annotated (i.e. original) dataset $\{(I, Q, A, C)\}$,

(2) generating a pseudo IQA dataset which consists of triplets $\{(I, Q, A)\}$ from the additional unannotated data $\{(I, C)\}$,

(3) training an IQA model with both the original and pseudo IQA datasets.

Figure 1 shows an example of a pseudo IQA triplet (I, Q, A) generated by our framework. The overview of the phase (2) and (3) is shown in Fig. 2.

Our experiments show that pseudo data generated by our method improves IQA performance as compared to the baseline trained with the original dataset and outperforms other naive data-augmentation methods.

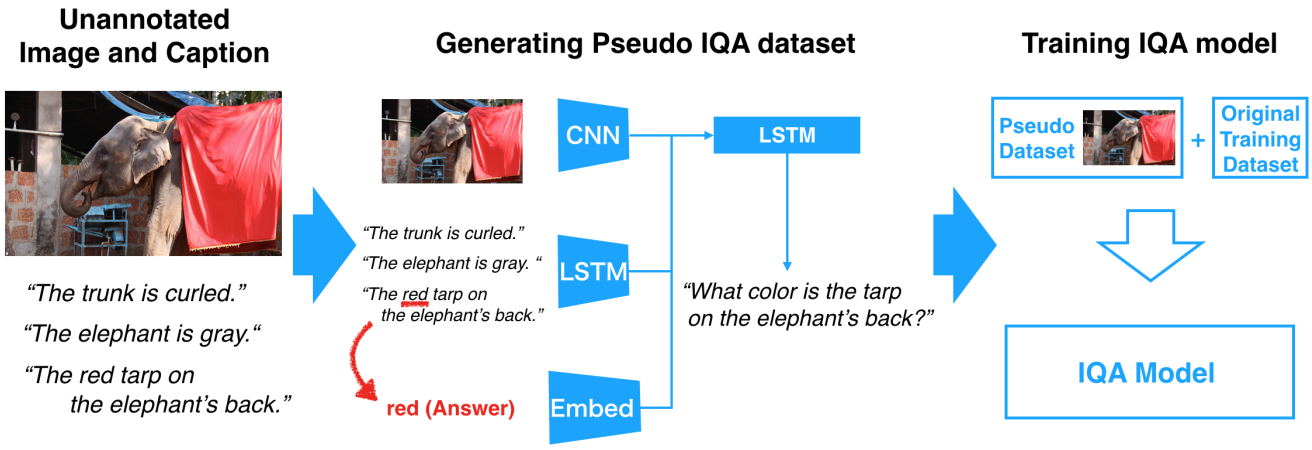

Figure 2: Overview of phase (2) and (3) of our framework. In phase (2), we extract an answer A from captions C associated with an image I, and subsequently generate a question from triplet (I, C, A). In phase (3), we train an IQA model with an original training dataset and generated pseudo dataset.

2. Related Work

Semi-supervised learning has achieved outstanding performance typically in single modal tasks, e.g. image classification and machine translation. The semi-supervised learning approaches exploiting unlabeled data can be categorized into four types: graph embedding methods (Yang et al., 2016), manifold learning methods (Rifai et al., 2011), generative models (Kingma et al., 2015; Rasmus et al., 2015) and self-labeling methods (Lee, 2013; He et al., 2016).

Roughly speaking, the first three approaches aim to obtain better representations or decision boundaries using unlabeled data. However, it is difficult to directly apply these methods to multi-input tasks because multi-modal data often has complex structures, making it unclear how to improve representations or decision boundaries efficiently.

The self-labeling approaches utilize unlabeled data by labeling it with the classes predicted by the model. Yang et al. (2017) has improved the performance of reading comprehension task with this approach. Their framework has two models: the question answering (QA) model and question generation (QG) model. The QA model aims to predict a correct answer from a given question and document. The goal of the QG model is to generate QA pairs from unlabeled documents. In the training phase, the QA model is updated using supervised learning with both the original dataset and generated pseudo dataset. The QG model is updated using reinforcement learning whose rewards are the accuracy of the QA model. Inspired by their work, our framework consists of the QA model and QG model. There are two differences between their work and our framework: using a multi-modal dataset and applying a supervised manner to both the QA model and QG model. Because of using only a supervised manner, our method is simpler than their work.

3. Question Generation

We train an IQG model with an original supervised training dataset consisting of quadruplets $\{(I, Q, A, C)\}$. Using the trained model, a pseudo IQA dataset is then generated from the additional unannotated data $\{(I, C)\}$ where we extract answers $\{A\}$ from captions $\{C\}$, and then generate questions from triplets $\{(I, C, A)\}$ by using the IQG model.

3.1. Image Question Generation Model

Our IQG model generates a question from a triplet (I, C, A). The model consists of sub networks of an image encoder, an answer encoder, a caption encoder and a question decoder. The image encoder and answer encoder are a convolutional neural network (CNN) and a word embedding layer, respectively. The caption encoder and the question decoder are both recurrent neural networks. Similarly to (Xu et al., 2015), the decoder predicts a question with an attention mechanism. Figure 3 shows the overview of the IQG model.

Image Encoder

We use a CNN to extract image representation vectors $\boldsymbol{X}$ that consist of L vectors from an image,

$$\boldsymbol{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_L\}. \quad (1)$$

Each representation vector $\boldsymbol{x}_i$ corresponds to the local regional vector of the image.

Caption Encoder

Each sentence in a caption is a sequence of word tokens. We encode each caption using a Long Short-Term Memory Network (LSTM) (Hochreiter and Schmidhuber, 1997). Let the number of captions relevant to the image be K. The caption representation vectors $\boldsymbol{D}$ are computed as follows:

$$\boldsymbol{D} = \{\boldsymbol{d}_1, ..., \boldsymbol{d}_K\}, \boldsymbol{d}_i = \mathrm{LSTM}(\boldsymbol{S}_i), \quad (2)$$

where $\boldsymbol{S}_i$ is the i-th sentence of the caption and $\boldsymbol{d}_i$ is the hidden state of LSTM given $\boldsymbol{S}_i$.

Answer Encoder

We tokenize and embed the input answer to obtain the answer feature vector $\boldsymbol{a}$,

$$\boldsymbol{a} = \boldsymbol{E}\boldsymbol{w}, \quad (3)$$

where $\boldsymbol{E}$ is a lookup table and $\boldsymbol{w}$ is a one-hot vector of the input answer.

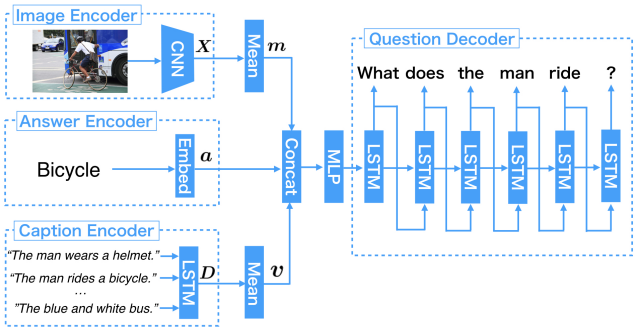

Figure 3: Overview of the IQG model which consists of three encoders and a decoder. Its input is a triplet (I, Q, A). $\boldsymbol{X}$ and $\boldsymbol{D}$ are the image feature and caption feature encoded respectively by a CNN and LSTM. The answer feature vector $\boldsymbol{a}$ is encoded by an embedding layer. $\boldsymbol{m}$ is the average of $\boldsymbol{X}$, and $\boldsymbol{v}$ is the average of $\boldsymbol{D}$. Using $\boldsymbol{m}$, $\boldsymbol{v}$ and $\boldsymbol{a}$ through the multilayer perceptron (MLP), we initialize the memory cell and hidden state of a question decoder which is an LSTM.

Question Decoder

We use an LSTM as a question decoder. $\boldsymbol{c}_t$ and $\boldsymbol{h}_t$ are the memory cell and hidden state of the question decoder at the timestep t. The initial memory cell $\boldsymbol{c}_0$ and hidden state $\boldsymbol{h}_0$ of the question decoder are given by

$$\boldsymbol{c}_0 = \mathrm{F}([\boldsymbol{m}; \boldsymbol{v}; \boldsymbol{a}]), \tag{4}$$

$$\boldsymbol{h}_0 = \mathrm{G}([\boldsymbol{m}; \boldsymbol{v}; \boldsymbol{a}]), \tag{5}$$

where $[\,\cdot\,;\,\cdot\,]$ represents vector concatenation, F and G indicate two-layer perceptrons, $\boldsymbol{m}$ is the average of $\boldsymbol{X}$, and $\boldsymbol{v}$ is the average of $\boldsymbol{D}$.

The decoder uses a soft attention mechanism (Xu et al., 2015) to utilize the features of the image and captions. At timestep t, the image context vector $\boldsymbol{y}_t$ and caption context vector $\boldsymbol{z}_t$ are given by

$$\boldsymbol{y}_t = \sum_{l=1}^{k} \alpha_{t,l} \boldsymbol{x}_l, \tag{6}$$

$$\alpha_{t,l} = \mathrm{softmax}(\mathrm{Q}([\boldsymbol{h}_{t-1}; \boldsymbol{a}; \boldsymbol{x}_l])), \tag{7}$$

$$\boldsymbol{z}_t = \sum_{l=1}^{k} \beta_{t,l} \boldsymbol{d}_l, \tag{8}$$

$$\beta_{t,l} = \mathrm{softmax}(\mathrm{R}([\boldsymbol{h}_{t-1}; \boldsymbol{a}; \boldsymbol{d}_l])), \tag{9}$$

where $\alpha_{t,l}$ and $\beta_{t,l}$ are image and caption attention weights, respectively, and Q and R are two-layer perceptrons.

The hidden state of the decoder is updated using the expression by the following rule:

$$\boldsymbol{h}_t = \mathrm{LSTM}([\boldsymbol{q}_{t-1}; \boldsymbol{h}_{t-1}; \boldsymbol{y}_t; \boldsymbol{z}_t; \boldsymbol{a}]). \tag{10}$$

The question $\boldsymbol{q}$ is a sequence of word tokens

$$\boldsymbol{q} = (\boldsymbol{q}_1, ..., \boldsymbol{q}_n), \tag{11}$$

where n is the question length. The decoder computes the probability $p(q_t|\boldsymbol{I}, \boldsymbol{D}, \boldsymbol{a})$ of a word token in the following way:

$$p(q_t|\boldsymbol{I}, \boldsymbol{D}, \boldsymbol{a}) = \mathrm{softmax}(\mathrm{H}(\boldsymbol{q}_{t-1}; \boldsymbol{h}_t; \boldsymbol{y}_t; \boldsymbol{z}_t; \boldsymbol{a})), \tag{12}$$

where H is a two-layer perceptron.

For training the IQG model, we minimize the softmax cross-entropy loss of each output by Stochastic Gradient Descent.

3.2. IQA Dataset Generation Procedure

We explain how to generate a pseudo IQA dataset from the additional unannotated data $\{(I, C)\}$ using the trained IQG model. The key question here is how to get plausible answers to use as inputs for question generation. Our dataset approach consists of two steps. We first sample the answers $\{A\}$ from N-grams of captions $\{C\}$ that contain any of the answer classes appearing in the original (seed) training data. The reason for using N-grams is that the target answer types in this study are words or phrases, e.g. "apple", "playing tennis" etc. Subsequently, we generate questions $\{Q\}$ from triplets $\{(I, C, A)\}$ using the IQG model. Finally, we use the generated triplets $\{(I, Q, A)\}$ as a pseudo IQA dataset.

4. Experiment

We conduct experiments using various sizes of the supervised training dataset to see how our data-augmentation framework improves IQA performance with respect to the amount of available annotated data.

4.1. Settings

The Visual Genome (VG) (Krishna et al., 2016) dataset is used in our experiments. As the VG dataset does not have a specified test split, we split the whole dataset into partitions of 20%/10%/10%/60% for training/validation/test/generation set, respectively. The training/validation/test datasets consist of quadruplets $\{(I, Q, A, C)\}$. The generation dataset contains pairs $\{(I, C)\}$. We use the training dataset to train both IQG and IQA models. The validation dataset is used to determine the hyperparameters of both models. The generation dataset is used as a pool of unannotated data which is used for generating pseudo IQA data. The test dataset is only used for the evaluation of final IQA model trained on the training dataset plus generated pseudo training dataset.

We use a question vocabulary consisting of words appearing over 100 times in the questions in the training dataset. A caption vocabulary is prepared in the same way as the question vocabulary. We replace out-of-vocabulary words with a special token ⟨UNK⟩. Answers appearing more than 100 times in the training dataset are used as answer classes. To evaluate an IQA model, we calculate the average classification accuracy on the test dataset. We use a subset of the test dataset consisting of examples whose answers are included in the answer classes defined above.

4.2. Baselines

We describe two simple baselines that produce a pseudo dataset.

A paraphrase-based approach generates question paraphrases of the original dataset. Given a triplet (I, Q, A)

of the original dataset, the method generates a paraphrase of the given question. We use the lexical paraphrases from the PPDB (Ganitkevitch et al., 2013) dataset (size S). If a question contains a word that appears in the PPDB dataset, the method generates a question by paraphrasing the word.
An object-detection-based approach generates pseudo IQA pairs by using YOLO V2 (Redmon and Farhadi, 2017), an object detection model. An object detection model can locate objects in a given image. We can generate an IQA pair by using the predicted information: the class of the object, and the number of the objects. We generate questions by using two templates: ("What is in the picture?", "*class*") and ("How many *class* are there?", "*number*"). For example, if the model detects one "cat" in the image, we generate two QA pairs: (question, answer) = ("What is in the picture?", "cat"), ("How many cats are there?", "one").

4.3. Implementation Details

We explain the implementation details of both the IQG and IQA models.

4.3.1. IQG Model

We use features from the pre-trained Resnet-152 (He et al., 2015) as the image representation vectors for the IQG model. All input images are rescaled to 448×448. The image features are extracted from the last convolutional layer (and their shape is $14 \times 14 \times 2048$). The sizes of the hidden state of the caption encoder and question decoder are 512. The dimensionality of the answer representation vector is 1024.
We optimize our IQG model with SMORMS3 (Funk, 2015) optimizer. The learning rate is set to 1.0×10^{-4}, and the batch size is 50. K, the number of captions relevant to an image is 15. Dropout (rate 0.5) (Srivastava et al., 2014) is applied to each layer for regularization. We also use weight decay of 1.0×10^{-5}.

4.3.2. IQA Model

We use DeeperLSTMQ, which is the VQA (Agrawal et al., 2015) baseline model, as the IQA model. We use the same hyperparameters and training manner described in the original paper.

4.4. Results and Analysis

Table 1 shows the results of our methods using subsets of training data (*Seed*) of varying sizes: 75K, 100K, 150K, 200K, 300K. $|L|$ and $|U|$ denote the sizes of the seed and pseudo dataset, respectively. *Para*, *OD* and *QG* denote the pseudo datasets generated by the paraphrase-based method, object-detection-based method and our proposed method, respectively.
The results show that each data-augmentation method improves the performance of IQA models as compared to the simple supervised learning (*Seed*). Although the improvement is not always significant, it becomes more prominent when the size of *Seed* data is small. Particularly, our method decreases the test error by 1.24% in the case of the annotated data of size 75K. This result suggests the effectiveness of our approach in the context of low-resource learning.

$\|L\|$	$\|U\|$	Training Data	Acc. [%]
75K	-	*Seed*	36.68
	300K	*Seed* + *Para*	37.28
		Seed + *OD*	37.76
		Seed + *QG*	**37.84**
100K	-	*Seed*	38.61
	300K	*Seed* + *Para*	39.03
		Seed + *OD*	38.95
		Seed + *QG*	**39.09**
150K	-	*Seed*	40.03
	300K	*Seed* + *Para*	40.26
		Seed + *OD*	40.13
		Seed + *QG*	**40.36**
200K	-	*Seed*	40.97
	300K	*Seed* + *Para*	41.04
		Seed + *OD*	41.09
		Seed + *QG*	**41.28**
300K	-	*Seed*	41.95
	300K	*Seed* + *Para*	41.96
		Seed + *OD*	41.98
		Seed + *QG*	**42.00**

Table 1: The IQA model performance on the test dataset of Visual Genome with various sizes of the annotated data. $|L|$ is the annotated dataset size, and $|U|$ is the size of generated data. *Seed* denotes the original training dataset. *Para* and *OD* stand for the pseudo datasets produced by the baselines. *QG* is the pseudo dataset produced by our proposed method.

Moreover, we can observe the effectiveness of captions from the results. From a theoretical viewpoint, it is not very surprising that our method outperforms the baselines since our method uses additional information (captions) to generate questions. However, considering that we can now easily obtain image-caption pairs from the web, we believe that our method is a practically promising approach to improve an IQA model with less human effort.

5. Conclusion

We have proposed a method to generate a pseudo IQA dataset which we believe is a promising approach to tackle resource problem in IQA. The key notion in our approach is that, to satisfy the component relevancies of triplets $\{(I, Q, A)\}$, we extract plausible answers A from external captions, and then produce questions from triplets $\{(I, C, A)\}$. Our model outperforms simple supervised-only baseline as well as other data-augmentation methods, which indicates the effectiveness of image-caption pairs for question generation.

Acknowledgements

We thanks the anonymous reviewers for their constructive comments on this work. This work was supported by JSPS KAKENHI Grant Number 16H05872.

References

Agrawal, A., Lu, J., Antol, S., Mitchell, M., Zitnick, C. L., Batra, D., and Parikh, D. (2015). Vqa: Visual question answering. In *Proc. of ICCV*.

Funk, S. (2015). Rmsprop loses to smorms3 - beware the epsilon! Technical report.

Ganitkevitch, J., Durme, B. V., and Callison-Burch, C. (2013). PPDB: The paraphrase database. In *Proc. of NAACL*.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition. Technical report.

He, D., Xia, Y., Qin, T., Wang, L., Yu, N., Liu, T.-Y., and Ma, W.-Y. (2016). Dual learning for machine translation. In *Proc. of NIPS*.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9:1735–1780.

Kingma, D. P., Mohamed, S., Rezende, D. J., and Welling, M. (2015). Semi-supervised learning with deep generative models. In *Proc. of NIPS*.

Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.-J., Shamma, D. A., Bernstein, M. S., and Fei-Fei, L. (2016). Visual genome: Connecting language and vision using crowdsourced dense image annotations. *arXiv preprint arXiv:1602.07332*.

Lee, D.-H. (2013). Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks. In *Proc. of ICML*.

Rasmus, A., Berglund, M., Honkala, M., Valpola, H., and Raiko, T. (2015). Semi- supervised learning with ladder networks. In *Proc. of NIPS*.

Redmon, J. and Farhadi, A. (2017). Yolo9000: Better, faster, stronger. In *Proc. of CVPR*.

Rifai, S., Dauphin, Y. N., Vincent, P., Bengio, Y., and Muller, X. (2011). The manifold tangent classifier. In *Proc. of NIPS*.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15:1929–1958.

Xu, K., Ba, J. L., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., Zemel, R. S., and Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. In *Proc. of ICMP*.

Yang, Z., Cohen, W., and dinov, R. S. (2016). Revisiting semi-supervised learning with graph embeddings. In *Proc. of ICML*.

Yang, Z., Hu, J., Salakhutdinov, R., and Cohen, W. W. (2017). Semi-supervised qa with generative domain-adaptive nets. *arXiv preprint arXiv:1702.02206*.