Learning to Caption Images Through a Lifetime by Asking Questions

Tingke Shen[1,2] Amlan Kar[1,2] Sanja Fidler[1,2,3]
[1]Vector Institute [2]University of Toronto [3]NVIDIA
{shenkev, amlan, fidler}@cs.toronto.edu

Abstract

In order to bring artificial agents into our lives, we will need to go beyond supervised learning on closed datasets to having the ability to continuously expand knowledge. Inspired by a student learning in a classroom, we present an agent that can continuously learn by posing natural language questions to humans. Our agent is composed of three interacting modules, one that performs captioning, another that generates questions and a decision maker that learns when to ask questions by implicitly reasoning about the uncertainty of the agent and expertise of the teacher. As compared to current active learning methods which query images for full captions, our agent is able to ask pointed questions to improve the generated captions. The agent trains on the improved captions, expanding its knowledge. We show that our approach achieves better performance using less human supervision than the baselines on the challenging MSCOCO [14] dataset.

1. Introduction

Imagine a child that sees a crocodile for the first time. She may likely ask what the animal is called, or where it can be encountered outside the zoo, but probably does not need to be told that it is green or has four legs, and that its sharp teeth can pose danger. Children (and even adults) learn from teachers in an active way: asking questions about concepts that they are unfamiliar or uncertain about. In doing so, they make learning more efficient – the child who acquires exactly the information they are missing – and the teacher who answers the question instead of needing to explain many aspects of a concept in full detail. As A.I. becomes more and more integrated in our everyday lives, be it in the form of personal assistants or household robots [28, 17, 23], they too should actively seek out missing information from humans – by asking questions in the form of natural language which non-experts can understand and answer.

Most existing work on scene understanding tasks such as VQA [5, 25, 29, 6] and captioning [14, 21, 3] have focused on a closed world setting, i.e. consuming the knowledge provided by a labeled dataset. On the other hand, the goal of active learning is to be able to continuously update the model by seeking for the relevant data to be additionally labeled by a human [22]. Most active learning approaches, however, ask the human to provide a full labeling of an example, and the main challenge is in identifying the examples to be labeled, to ensure annotation efficiency. In our work, we go beyond this, by endowing the model with the ability to ask for a particular aspect of a label, and do so in natural language in order to unambiguously identify the missing information.

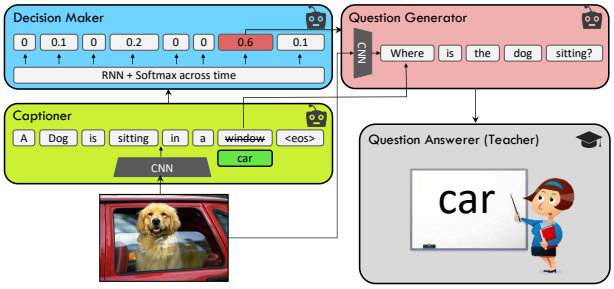

Figure 1. Learning to describe images by asking questions. Our model learns in a lifetime learning setting, by actively seeking for missing information. We jointly learn when and what to ask, and learn from the teacher's answers. Our model poses questions in natural language.

We focus on the task of image captioning as a proxy task for scene understanding. In order to describe an image, a model needs to generate words describing the objects, their attributes, actions, and possibly relationships and interactions between objects. This is inherently a multi-task problem. In this paper, our goal is to allow a captioning agent to actively ask questions about the aspects of the image it is uncertain about, in a lifetime learning setting in which examples arrive sequentially and continually. Thus, instead of having humans provide captions for each new image, our agent aims to ask a minimal set of questions for the human to answer, and learn to caption from these answers.

Our model consists of three modules: a captioning module, a decision making module that learns whether to ask and what to ask about, and a question generation module. At training time when the captioner produces each word, the decision module decides for which concept, if any, to ask about. If the agent decides to ask, the question module produces a question, which the teacher answers. All three modules are implemented as neural networks. They are updated continuously with the data arriving in batches: the captioning module is updated using the captions improved

by the answers from the teacher, while the decision module is updated based on the current uncertainty of the captioning module. For efficiency reasons, our teacher to answer questions is a QA bot. At test time the captioning model describes new images without asking questions.

In summary, our contributions are:

- A new Learning by Asking Questions paradigm in which captioning, question generating, and decision modules interact in order to learn in over a lifetime. The advantage of LBAQ is it improves the efficiency of data collection.
- A novel decision maker module, trained with reinforcement learning (RL) that decides whether and what to ask a question about by implicitly reasoning about the uncertainty of the agent and knowledge of the teacher.

We showcase our method on MSCOCO [14]. We provide insights into the behavior of our approach, and discuss open challenges ahead. To the best of our knowledge, this is the first time that natural language question asking has been explored in a lifetime learning setting with real-world images. Please visit our project page http://aidemos.cs.toronto.edu/lbaq/ for demo and code release.

2. Related Work

We provide a short overview of (inter)active learning approaches, and outline our main contributions.

Active learning. The goal of active learning is to intelligently seek labels for unlabelled data from an oracle in order to maximize learning while reducing the annotation cost. An agent predicts which sample, if labelled, will give the most useful learning signal as measured by performance on the test set. Strategies for active learning include uncertainty sampling, query by committee and expected model change [22]. Unlike the typical active learning setting where an agent asks the oracle for a full data label (which would be a full caption in our scenario), our method learns to ask pointed questions to retrieve partial labels, *i.e*. missing key words that compose a caption. Our model thus needs to not only learn when to ask, but also what to ask, and how to distill the received answer into a complex multi-task module (captioner).

Learning by Asking Questions is an exciting direction with notable contemporary work. Prior approaches typically differ in task, methodology (are questions natural or templated? how does the agent utilize the feedback?) and environment (synthetic vs real). [18] learns to answer questions by asking questions. Image and the generated question are treated as an unlabelled sample and an oracle provides an answer to form a novel training pair. This simplifies the learning by asking framework by bypassing the challenges of free-form conversation and interpreting the teacher's answer, because QA can be directly used as training data. Our work generalizes over this framework by using question-asking as a support task to the main task, in our case image captioning, which leads to a more general, and significantly more challenging scenario. Furthermore, [18] operates in CLEVR [8], a synthetic environment and questions are limited to programs rather than natural language.

[31] explores question asking for visual recognition. Given an image, a graph of objects, attributes and relationships is continually updated as the agent asks questions. However, questions are limited to templates, and training is done in synthetic environments with a limited set of objects and relationships. [26] uses questions to explore new object classes for image classification. However, [26] does not retrain their classifier. Our work differs from [31, 26] by proposing a way for the agent to learn in a lifetime setting.

In [11], the agent learns whether to ask questions to the teacher to efficiently solve dialogue tasks. The student's goal is to maximize the accuracy of answering the teacher's questions while reducing the cost (to the teacher) of asking for hints. We extend this line of thinking by letting the agent learn what to ask about in addition to whether to ask.

Vision and Language. Our work tackles captioning [30, 21, 3], visual question answering (VQA) [25, 6, 10], and visual question generation (VQG) [12, 19]. However, most of these works have focused on a closed dataset setting. Our main goal here is not in designing a novel architecture for each module (captioning, VQG, VQA), but rather focusing on the interaction of the modules and the teacher in order to learn in a continual, active setting. Related to us is [15], where a teacher observes the captioning agent in a continual setting, and gives natural language feedback when errors occur. The agent then learns to improve based on this signal. In our work, the agent is the one seeking advice, thus making the teaching process more efficient.

3. Our Approach

Our goal is to train an image captioning model in the active learning setting with minimal human supervision. We approach the problem by endowing the agent with the ability to ask questions, and learn from the teacher's answers. However, question asking is only a tool for retrieving information during training; at test time, the captioner operates without needing to ask questions. We first provide an intuitive overview of our interactive training procedure, describing the lifetime learning setting, namely how the agent learns from data arriving in a sequence of batches. Next, we provide details of how the agent queries for, and learns from, answers and feedback from the teacher. Finally, we describe the implementation of our agent's modules.

3.1. Lifetime Learning

We imagine a lifetime learning setting where data arrives in chunks. This is analogous to a student who learns over multiple classes in a semester. The first chunk D_w has complete ground truth (GT), *i.e*. human written captions. We refer to it as the warmup chunk. The agent learns from the re-

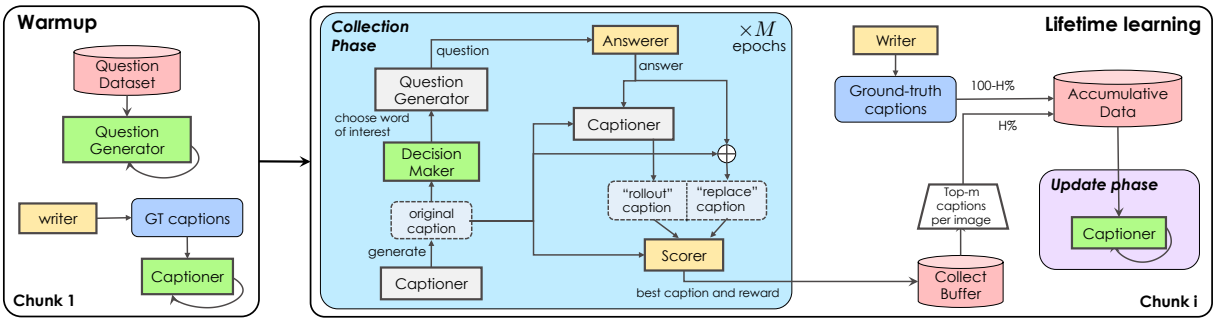

Figure 2. Modules being updated (green), modules held fixed (grey), teacher (yellow). Writer is a teacher that produces full GT captions. Captioner begins by warming up on the first chunk containing all GT captions (left panel). Learning by asking questions (right panel) occurs in two phases: collection and update. In collection phase, the captioner generates a caption, the decision maker chooses when to ask a question, the question generator generates a question and the teacher provides an answer. Answer is used to create two new captions. Captions are collected and used to train the captioner in the update phase.

maining K unlabelled chunks $D_u = [D_{u1}, D_{u2}, \ldots, D_{uK}]$ with partial supervision from the teacher. We first train the question generator and pretrain the captioner on the warmup chunk. For each unlabelled chunk, the agent iterates between two phases: querying the teacher, and learning from the collected information.

In the **(caption) collection phase**, the agent interacts with the teacher using two modules: a decision maker, and a question generator. The agent attempts to caption a new image in an unlabelled chunk, and decides whether to replace words with answers obtained by asking questions. The agent collects the improved captions and uses them to train the captioner in the **update phase**. In collection phase, feedback from the teacher is also used to train the decision maker to make better decisions about whether/when to ask. The process is illustrated in Fig 2, and summarized in Alg 1.

3.2. Notation

Let $\mathbf{w} = (w_1, w_2, \ldots, w_L)$ denote a caption of length L, and I an image. The **captioning module** $C(\mathbf{w}|I)$ computes a probability distribution over the words in a sentence, *i.e.* $p_{\theta_C}(\mathbf{w}|I)$. We further compute $\mathbf{c} = (c_1, c_2, \ldots, c_L)$, denoting an array of contexts computed by the captioner (details in Sec 3.5). The context helps the decision maker decide what concepts to ask about, and the question generator to ask relevant questions. Let the context used by the decision maker and question generator be called c^{DM} and c^q, respectively. The **decision module** $DM(t|\mathbf{c})$ computes a multinomial distribution $p_{\theta_{DM}}(t|c^{DM})$ indicating the probability of a word position t in the caption at which the question should be asked. We allow t to index a special `<eos>` position representing the case where no question should be asked. The **question generation module** $Q(\mathbf{q}|I, c_t^q)$ computes the probability distribution $p_{\theta_q}(\mathbf{q}|I, c_t^q)$ over a question $\mathbf{q}$. The details about the modules are presented in Sec 3.5.

3.3. Caption Collection Phase

In the collection phase, the agent attempts to improve captions generated from its own policy by querying the teacher. For each round, the agent makes multiple passes over a chunk. Given an image, the agent generates a caption, and the decision maker decides whether and when (at which word) to ask a question to the teacher. The teacher answers the question, which the agent uses to create a new caption (details in Section 3.3.1). The teacher scores both new and old captions and the agent stores the captions in a buffer D_c. At the same time, the agent uses the scores from the teacher to make online updates to the decision maker to pick better time steps for asking questions (Section 3.3.2).

The collected captions will be used in the update phase by the agent to distill the teacher's knowledge back into the captioner. However, the agent could encounter difficult images that cannot be improved by asking questions. Empirically we find the agent cannot improve on images containing objects in unusual settings, or if the caption generated from the captioner's policy is missing multiple key concepts. Therefore, we allow the agent to "give up" if the improved caption is bad, and the teacher writes a new caption. This is analogous to a student asking for a full explanation from the teacher after class if he did not understand a concept. For every image, the agent considers the top m captions from the buffer D_c for training. It keeps the top $H\%$ of images-caption tuples based on the average caption reward over m captions. For the other 100-$H\%$ images, the agent "gives up" and is given m GT captions. In practice, we choose $m = 2$ out of the 5 MSCOCO captions. The `KeepBestAndGiveUp` subroutine in Algorithm 1 summarizes how the agent selects training data for the captioner.

3.3.1 Interacting with the Teacher Details

Given an image, the captioner produces the complete initial caption $\mathbf{w}^0$ and context $\mathbf{c}^0$ by a greedy rollout from $p_{\theta_C}(.|I)$. The decision module then makes a decision by sampling from $p_{\theta_{DM}}(.|c^{DM})$. Words other than nouns, verbs, and adjectives are masked out. Let w_t be the word for which the decision module decides to ask a question. The question generator produces a question and the agent receives an answer a. The agent then replaces word w_t in $\mathbf{w}^0$ with a and predicts a new caption $\mathbf{w}_{ro}^1 = (w_1 \ldots w_{t-1}, a, w'_{t+1}, \ldots w'_L)$, by rolling out the rest of the caption from position t using the previous hidden state h_{t-1}

Algorithm 1 Lifetime learning

procedure LIFETIME(D_w, D_u)
  train: C, Q, V ▷ train captioner, question generator, QA-bot
  initialize: DM ▷ initialize decision maker
  $D \leftarrow D_w$
  $D_u = [D_{u1}, D_{u2}, \dots D_{uK}]$
  for D_{uk} in D_u **do** ▷ begin lifetime learning
    $D_c \leftarrow [\,]$ ▷ collection phase
    for epoch $= 1$ to Number of Passes over Chunk **do**
      for I in D_{uk} **do**
        $\mathbf{w}, r^{1:N}, t^{1:N} \leftarrow$ `SeekTeacher`(I)
        $\mathbf{w}^*, (r^*)^{1:N} \leftarrow$ `SeekTeacher`$(I,$ greedy=True$)$
        D_c += $(\mathbf{w}, r, \mathbf{w}^*, r^*)$ ▷ collect caps. and rewards
        θ_{DM} += $\sum_{n=1}^{N}[r^n - (r^*)^n]\nabla \log p_{\theta_{DM}}(t^n|\mathbf{c}^{n-1})$
    $D \leftarrow$ `KeepBestAndGiveUp`(D_c, H)
    train: C on D using $L(\theta_C)$ ▷ update phase

of the captioner and a. If the teacher's answer is a rare word for the agent, the agent may diverge from any sensible trajectory. For this reason, we give the agent the option of doing a one-word-replace of the expert's answer, *i.e.* $\mathbf{w}^1_{re} = (w_1 \dots w_{t-1}, a, w_{t+1}, \dots w_L)$.

Finally the teacher scores both the original and the two improved captions, by giving each a numeric reward r. The process can be repeated by asking a second question and replacing another word at step $t' > t$. In general, the agent can ask up to N questions for a single caption. In practice, we observe $N = 1$ to work best in our experiments. We keep N in the following for the generality of exposition. The interaction process is summarized in Algorithm 2.

3.3.2 Learning When to Ask Questions

As the agent queries the teacher, it trains the decision maker online to make better decisions. The teacher provides a scalar, non-differentiable reward. Hence we update decision maker using REINFORCE [24]. We baseline the reward with the greedy decision reward $(r^*)^0$ (*i.e.*, what the improved-caption would have been had DM sampled greedily), following the self-critical policy gradient [21]. See line 11 in Alg 1. In the general case with N questions asked, the gradient for the parameters of the decision maker θ_{DM} is:

$$\sum_{n=1}^{N}[r^n - (r^*)^n]\nabla \log p_{\theta_{DM}}(t^n|\mathbf{c}^{n-1}) \tag{1}$$

In this work we did not update the question generator in lifetime learning because jointly training the decision maker and question generator is a hierarchical RL problem. Reward accreditation is challenging because the agent needs to learn to differentiate DM choosing a bad time step from DM choosing a good time step but question generator generating a bad question.

3.4. Captioner Update Phase

After the collection phase, the agent trains the captioning module on the collected captions. We assume the agent has full access to past data D and is retrained from scratch. We retrain from scratch to avoid the added complexity of applying learning-without-forgetting techniques since our model has many moving parts already. Future works can look at how to efficiently learn on the new data. D contains warmup GT captions, collected captions, and GT captions from "giving up". The captioner is retrained using a joint loss over the captions stored in D,

$$L(\theta_C) = -\sum_{\mathbf{w}\in D} r_{\mathbf{w}} \log p_{\theta_C}(\mathbf{w}|I) - \lambda \sum_{\mathbf{w}^*\in D} \log p_{\theta_C}(\mathbf{w}^*|I) \tag{2}$$

where $\mathbf{w}$ are collected captions, $\mathbf{w}^*$ are GT captions, $r_{\mathbf{w}}$ is the score given by the teacher for $\mathbf{w}$, and λ is a tuned hyperparameter. In practice, we set λ to the 90^{th} percentile reward of the collected captions, assuming that ground truth captions are generally better than collected captions.

Algorithm 2 Interacting with the teacher

procedure SEEKTEACHER(I, GREEDY=FALSE)
  $\mathbf{w}^0, \mathbf{c}^0 \leftarrow C(\cdot|I)$ ▷ compute caption and context
  $r^0 \leftarrow$ `TeacherScore`$(\mathbf{w}^0)$
  for $n = 1$ to N **do**
    $t^n \leftarrow DM(\cdot|\mathbf{c}^{DM,n-1}$, greedy$)$ ▷ DM samples step
    $\mathbf{q} \leftarrow Q(\cdot|I, c_{t^n}^{q,n-1})$ ▷ generate question
    $a \leftarrow V(\cdot|I, \mathbf{q})$ ▷ teacher provides answer
    $\mathbf{w}^n_{ro}, \mathbf{c}^n \leftarrow [\mathbf{w}^{n-1}_{0:t^n-1}, a, C(\cdot|I, h_{t^n-1}, a)]$ ▷ roll new cap.
    $\mathbf{w}^n_{re} \leftarrow [\mathbf{w}^{n-1}_{0:t^n-1}, a, \mathbf{w}^{n-1}_{t^n+1:}]$
    $r^n_{ro} \leftarrow$ `TeacherScore`$(\mathbf{w}^n_{ro})$ ▷ teacher scores caption
    $r^n_{re} \leftarrow$ `TeacherScore`$(\mathbf{w}^n_{re})$
    $\mathbf{w}^n, r^n \leftarrow \max\{r^{n-1}, r^n_{ro}, r^n_{re}\}$
  return $\mathbf{w}^N, r^{n=1:N}, t^{n=1:N}$

3.5. Implementation Details

Captioning module. $C(\mathbf{w}|I)$ is implemented as an attention CNN-RNN model [30]. We additionally predict a part-of-speech (POS) tag at each time step to inform the question generator what type of question should be asked and the decision maker whether to ask. Captioner is trained using MLE with teacher forcing and scheduled sampling.

Question generation module. $Q(\mathbf{q}|I, c^q_t)$ is also implemented as a CNN-RNN and conditions on the context at time t. Specifically, c^q_t consists of: POS distribution which determines the "question type", the attention weights predicted by the captioner which guide the question generator to look, an encoding of the caption which provides global context and prevents asking for redundant concepts, and the position encoding for t. We found it helpful to allow the question generator to re-attend rather than fully rely on the captioner's attention. We train the question generator on a novel dataset, using MLE with teacher forcing and scheduled sampling similar to the captioner (details in Appendix).

Decision module. The decision maker $DM(t|\mathbf{c})$ is implemented as a multilayer perceptron (MLP) with Softmax output. Context c^{DM} consists of the POS distribution, an encoding of the caption, and uncertainty metrics computed from top-k words predicted by the captioner:

- Cosine similarity between the embedding of the top-1 word and all other $k-1$ words.

- Cosine similarity between each top-k word and the embedding of the entire sentence (implemented as the sum of word embeddings).
- Minimum distance of each top-k word to another word.

Entropy is a natural way to measure the uncertainty of the captioner. However, the model can predict synonyms which increase entropy but do not suggest that the model is uncertain. Therefore, for each time step we take the word embeddings of the top-k words and compute their relative distances as a secondary measure of uncertainty. We use $k = 6$. In ablation studies, we show that these statistics alone can capture the uncertainty of the cap. Training a neural network on these stats further improves performance.

Teacher module. We imagine our agent in a human-in-the-loop setting where a teacher answers natural language questions, chooses the best caption out of a few alternatives, scores it, and writes GT captions if necessary. The teacher consists of two parts: a VQA bot $V(a|I, \mathbf{q})$ implemented following [25] and a caption scorer composed of a linear combination of BLEU [20], ROUGE [13], METEOR [2], and CIDEr [27]. We call the reward from the caption scorer the `Mix` score, and denote it by r. We discuss challenges to using a synthetic teacher in Sections 4.3 and 4.6.

4. Experiments

We evaluate our approach on the challenging MSCOCO dataset [14], and compare it to intelligent baselines. We perform detailed ablation studies that verify our choices and give insight into how our model behaves.

We follow the standard Karpathy split [9] that contains 117,843 training, 5K validation and 5K test images. We randomly split the training set into warmup and lifetime learning chunks. In our experiments, we vary the size of the warmup, and the number of lifetime chunks, to analyze the model behavior under different regimes. There are 5 GT captions for each image in the warmup set. At the end of lifetime learning, there are $m = 2$ collected or GT captions for each image in the lifetime set.

Image features are extracted with ResNet-101 trained on ImageNet [4] [7]. Vocabulary sizes for the captioner, question generator and VQA are 11253, 9755 and 3003, respectively. We use the Stanford NLP parser to get GT POS labels [16]. The decision maker only considers a subset of tags (listed in Appendix) for asking questions.

4.1. Training Details

The synthetic teacher (VQA bot) was trained on the VQA2.0 dataset [1], following a simplified implementation of [25] using a multi-answer binary cross entropy loss function. The VQA model achieves 64.2% on the VQA2.0 val split without ensembling. We train the question generator by combining data from MSCOCO and VQA2.0. (Implementation details in App.) A natural concern is that training the question generator on images the captioner sees during lifetime learning will cause the que. gen. to "lookup" GT questions. We find this to not be the case (see Figure 8). In general, the questions generated for an image are diverse, generic and rarely match GT questions (see Appendix for more examples). The entire training process takes 2.5 longer than supervised learning baselines, mostly because we retrain the captioner from scratch. This slowdown can be overcome in future works by using learning-without-forgetting techniques.

4.2. Cost of Human Supervision

We first perform a human study to understand human cost associated with every interaction type with the agent. We choose to measure "human effort" as the time taken for a task. In our experiment, a human teacher has three possible tasks: produce a full caption, answer a question, and score a caption. Table 4 shows that on average it takes 5.2 and 4.6 times longer to caption than score a caption or answer a question. To compute the cost of human supervision, we normalize the cost of each task to caption scoring. Hence the agent incurs one point of supervision for each caption scored, 1.13 for each question answered, and 5.2 for each caption written. In practice, we assume no cost when the VQA module answers a question. A human teacher would charge the agent for answers but would also give better answers. In the experiments to follow, we use *Human Supervision* as a metric for cost incurred by querying a human.

4.3. Learning by Asking Questions

In Table 1 we evaluate our lifetime learner, *aka* "inquisitive student" (IS), against training only on GT data on the test split. All results are reported using greedy decoding. Our model was trained with a 10% warmup chunk, 3 unlabelled chunks and $H = 70\%$ collect percentage. For each setting we report the best model out of three with different random seeds on the test set. We report two GT baselines: *Equal GT* – the same number of GT captions as our model but no additional collected captions from the teacher, and *All GT* – GT captions are used for all images (same number of captions as our model).

In order to evaluate the benefits of asking questions, we introduce Mute Student (MS), a lifetime learner that interacts with the teacher by only receiving feedback on whether captions are good (does not ask questions). MS is trained in exactly the same lifetime setting as IS, but samples multiple captions from the captioner's current distribution rather than ask questions to construct new captions to be rated by the teacher. The best captions are still collected and used to train for the next round. All models have the same hyperparameters and captioning architecture and are trained on all images to ensure fairness. GT % (captions) and (human) Supervision % are reported relative to *All GT*.

Compared to *Equal GT*, our lifetime model achieves 5

Method	$H\%$	GT %	Supervision %	Mix	CIDEr	METEOR	ROUGE	BLEU4	BLEU2
Equal GT	-	45.2 %	45.2 %	98.9	91.5	24.7	52.3	28.0	53.4
All GT	-	100 %	100 %	101.7	96.4	25.1	52.9	28.8	54.9
Inquisitive Student	70%	45.2 %	73.5 %	**103.9**	**98.0**	**25.4**	**53.8**	**30.5**	**57.1**
Mute Student	70%	45.2 %	72.6 %	102.2	95.9	25.2	53.4	29.3	55.9

Table 1. Evaluation on *test*. Our model was trained with 10% warmup and 3 unlabelled chunks. Methods see all images at least once for fairness. **Note:** (Best of 3 runs) 100% GT corresponds to 46% of the MSCOCO training captions because only 2 (out of 5) captions are used for each image in the lifetime chunks.

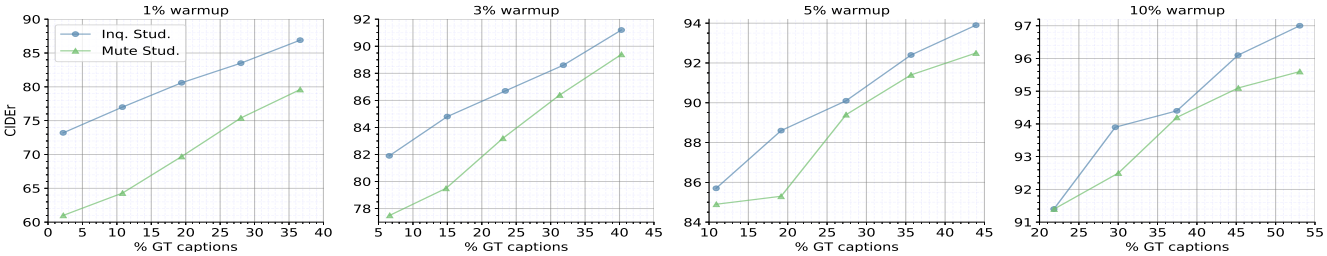

Figure 3. Caption quality on test. Both models are decoded greedily.For each plot, GT % is varied by changing the percentage of captions $H\%$ collected by the agent. % GT captions is reported relative to *All GT*.

`Mix` and 6.5 CIDEr higher which shows that for an agent with a fixed budget of GT captions, additionally learning from collected captions can significantly improve performance. Compared to *All GT*, our model achieves 2.2 `Mix` or 1.6 CIDEr higher score while using only 45.2% of GT captions and 73.5% of human supervision. This means that training on teacher-improved captions not only achieves greater efficiency but also leads to higher performance than training on GT captions. We find this to be a particularly strong and interesting result.

IS also beats MS, which demonstrates that question-asking is beneficial. This is investigated further in Fig. 3. We vary the amount of GT captions by adjusting the percentage H of collected captions. We call an agent that trusts its teacher-improved captions often (and rarely gives up) a "confident" learner. Confident learners use less human supervision. An agent that begins lifetime learning earlier with only a small warmup set is an "eager" learner.

IS outperforms MS in almost all settings but the difference is greater if the agents are eager. Fig. 3 shows that at 10% warmup the gap is 1.4 CIDEr (97 vs 95.6) but as we reduce to 1% warmup, the gap becomes 12.7 CIDEr (77 vs 64.3). This supports the intuition that asking questions benefits learners with less experience. In addition, a more eager learner ultimately reaches lower performance for the same amount of supervision. For about 30% GT captions IS achieves 93.9 CIDEr in the 10% warmup setting and 83.5 CIDEr in the 1% warmup setting. We hypothesize this is because the quality of sentence continuations, or rollouts after receiving the teacher's answer, worsens if the agent pretrains on less data. Furthermore, a very eager learner may make too many mistakes to fix by asking only one question.

Selected examples are shown in Fig 4. The first four examples are positive and show asking questions helps fix incorrect words and retrieve novel concepts. In the fifth example, the reward is lower for the new caption even though it is good according to human judgment. Auto-eval metrics do not reward the agent for relevant, novel captions that don't match words in the reference captions. A human teacher with more flexible scoring could encourage the agent to learn more diverse captions and a larger vocabulary.

4.4. Learning New Concepts

1%, 3% and 10% warmup datasets contain only 30%, 47%, and 70% of the captioning vocabulary respectively. The remaining words/concepts are explored in lifetime learning. Fig. 5 shows the number of unique words used by a captioner evaluated on the val split at the end of lifetime learning. We found a dependency between training epochs and vocabulary size and therefore took all models at the same epoch. We baseline against mute student. IS has a larger knowledge base than MS at all % GT as it uses more unique noun, verb and total words than MS, showing IS is able to learn new vocabulary.

In Table 3 we compare the vocabulary of lifetime learners to *All GT*. *All GT* has a larger vocabulary than lifetime learners. This is intuitive because *All GT* has more GT captions and therefore sees more varied data. IS only receives a single word answer given an image, whereas *All GT* receives a complete caption label containing on average 10.5 words. For the same reason, in Fig. 5 the agents' vocabulary decreases as % GT decreases.

Another way to measure the usefulness of teacher's answers is to compute how often it repeats a concept the captioner already knows. Table 2 shows how frequently the answer from the teacher appears in the top-k words predicted by the captioner at the time step where the question is asked (ATopk). Note that this is approximate because the captioner may predict the answer at a different step. In the first round of lifetime training, 26.3% of teacher answers appeared in the top-5 words predicted by the captioner. Hence, 73.7% of the time, the agent is sees an unfamiliar or novel concepts. Over the lifetime, ATopk increases as the stu-

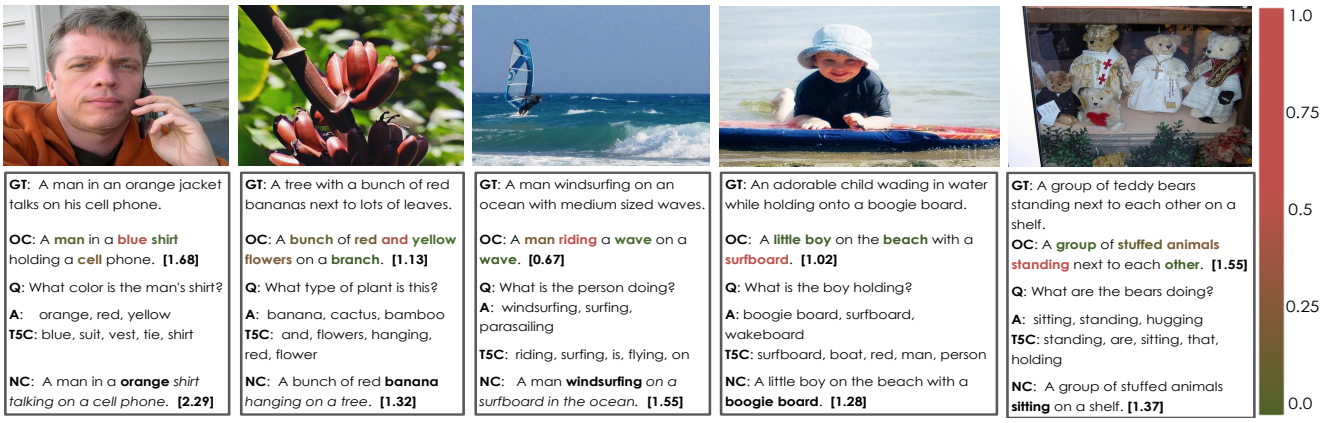

Figure 4. T5C: top-5 words predicted by captioner at the word when question is asked. Rewards are in square brackets. Colors in OC indicate probability the decision maker will ask about a word (scale is on right). Left 4 are positive examples, right is failed (pointing to weaknesses of auto-eval metric). NC is the "rollout" caption. Even when one word (answer) is replaced, multiple words can be updated because the captioner samples the rest of the sentence conditioned on the answer.

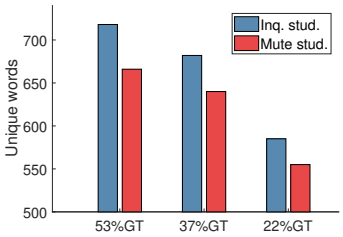

Figure 5. Num. of unique words used by captioner evaluated on val at the end of lifetime learning. Models trained with 10% warmup and 3 chunks.

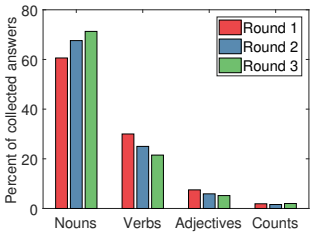

Figure 6. Distribution of teacher answer types over rounds. The model was trained using 10% warmup, $H = 70\%$ and 3 chunks.

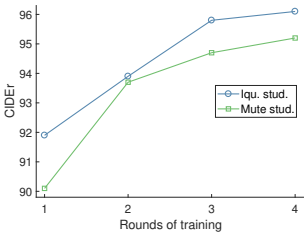

Figure 7. Performance on val vs the number of total chunks (plus the warmup). Models were trained using 10% warmup and $H = 70\%$.

dent's knowledge catches up to that of the teacher.

4.5. Analyzing the Modules

Question Generator. We conducted a human study (Fig. 11) using Amazon Mechanical Turk (AMT) to evaluate the quality of generated questions. Annotators rated 500 images-question pairs by answering questions if they were good or flagging questions as "not understandable" or "irrelevant to the image". The questions were randomly selected questions that the question generator asked while trying to caption. The images were not seen by the question generator during its training. 82.4% of questions were rated "good" and answered. This is a promising result and suggests that learning by asking can be adapted to use human teachers instead of a QA bot.

Fig. 8 shows generated questions at different time steps in a caption. In general, generated questions tend to be diverse, and generic. It's important for questions to be generic so that the teacher can answer with a wide range of possible concepts and possibly new concepts. We also rarely observe the generated questions to be the same as the GT questions. More examples in Appendix.

Decision Maker. To test the decision maker, we look directly at the scores of the refined captions it produces, rather than those of the final captions after retraining the captioner. This lets us to precisely observe the ablated performance of the DM. Table 9 evaluates different decision maker strategies. We first train captioning and question generation modules. The baseline is the performance of the captioner without asking questions. The other settings use various decision maker models to ask a question to improve captions. Learned models are trained using RL on a single chunk of unlabelled data. Scores are shown for the *val split*.

The full model gives 6.5 CIDEr improvement over no question asking. Picking the time step with maximum entropy is not a very good strategy. It is only 0.3 CIDEr better than picking a random step. This is because the model can predict synonyms which increase the entropy but do not indicate the model is uncertain. Adding closeness metrics yields 1.0 CIDEr improvement over maximum entropy, showing that taking into account the closeness of words in embedding space gives a better measure of uncertainty. In all cases, learning improves performance, with the best learned model achieving 3.1 CIDEr higher than the best non-learned model. We use the full model as our decision maker for all experiments.

4.6. Understanding the Model

Number of chunks. Fig. 7 shows that as the number of chunks increases, performance increases (for similar human supervision). This is intuitive because more chunks means the agent sees fewer images before adapting the captioner.

C: Two **cats sit in** a room with a **cat**.
Q1: What animal is in the photo? **A**: cat
Q2: What are the cats doing? **A**: looking out window
Q3: Are these cats sitting or outside? **A**: inside
Q4: What are the cats looking at? **A**: window
GTQ: What animals are shown?
GTQ: How many cats are there?

C: A **train sitting on** the **tracks**.
Q1: What is the yellow object? **A**: train
Q2: Is this train moving or coming? **A**: going
Q3: Is the train in or outside? **A**: outside
Q4: Where is the train? **A**: station
GTQ: What color are the train doors on the right?
GTQ: What shape are the windows?

C: Three **people** are playing **with a large** frisbee.
Q1: Who is holding the frisbee? **A**: boy
Q2: What kind of game are they playing? **A**: frisbee
Q3: What is the man in the blue shirt holding? **A**: frisbee
Q4: What color is the frisbee? **A**: blue

C: A **cat** laying **on a bed** with a **pillow** and a pillow.
Q1: What is on top of the suitcase? **A**: cat
Q2: Is the cat inside or inside? **A**: inside
Q3: What kind of cat is on the left? **A**: gray
Q4: Where is the cat? **A**: suitcase
Q5: What is on the left of the suitcase? **A**: cat

Figure 8. Questions generated from different words in the generated caption (colors match words to questions). Highlighted questions retrieve answers that are novel to the caption. Left 2 images are seen by question gen. during training (GTQ are GT questions used for training), right 2 are not. Generated questions tend to be diverse and different from GT ones.

Round	ATop3	ATop5	ATop10
1	17.7	26.3	37.4
2	24.1	34.2	46.9
3	27.4	38.3	50.7

Table 2. Frequency (in %) of teacher answers that occur in captioning module's predictions during lifetime training. Calculated from agent's collected captions in each round.

Model	Nouns	Verbs	Adj.
IS	527	97	53
MS	491	86	48
All GT	680	127	47

Table 3. Number of unique words used by each model on val. Lifetime learners are trained with 10% warmup, $H = 60\%$, 3 chunks.

Task	Avg. time (s)	Std. (s)	Time ratio
Captioning	34.4	21.8	1.0
Scoring	6.6	2.2	5.2
Answering	7.6	3.7	4.6

Table 4. Time taken by humans to perform tasks: captioning, scoring a caption, answering a question. Time ratio is relative to captioning. $N = 27$ humans surveyed, $n_c = 270$ captions written, $n_q = 675$ questions answered, $n_s = 675$ captions scored.

Method	Mix	C	B4
No questions	86.4	74.1	22.1
Random	88.3	76.2	22.2
Entropy	88.9	76.5	22.4
Unc. metrics	89.6	77.5	22.5
Unc. metrics learned	90.8	79.3	23.2
Full learned	91.9	80.6	23.7

Figure 9. Ablating the decision maker. *Entropy* is picking the time step with highest top-k word entropy. *Unc. metrics* includes entropy and words closeness (Sec. 3.5). *Unc. metrics learned* adds a MLP to predict the best time step for asking. *Full learned* additionally includes POS and an encoding of the caption as input.

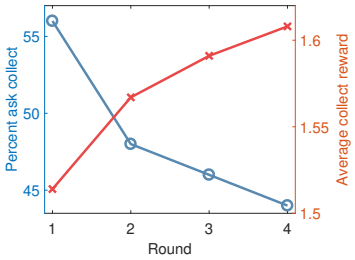

Figure 10. Changes to collected captions over rounds. Model trained with 10% warmup, $H = 70\%$, 3 chunks.

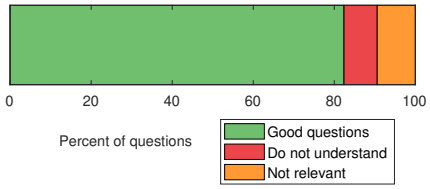

Figure 11. AMT study to judge the quality of the generated questions. Given an image and a question, annotators were asked to answer the question if it is good, or flag it as "not understandable" or "not relevant". Generally the questions were good.

The number of chunks cannot be too large because we retrain the captioner from scratch after every chunk.

Catching up to the teacher. Fig. 10 shows the percent of collected captions that improved by asking questions (left axis) and average reward of collected captions (right axis) versus num. consumed chunks. Over time, the agent is able to improve fewer and fewer captions by querying the teacher. Furthermore, the largest increase in collected reward occurs in the first round. These observations suggest that the teacher's knowledge is exhausted over time.

Types of answers. In Fig. 6 we see the distribution of answer types from the teacher. Over time, the student asks for more nouns, and less verbs and adjectives. We hypothesize this is because the agent is learning verbs and adjectives early on before moving onto nouns.

5. Conclusion

In this paper, we addressed the problem of active learning for the task of image captioning. In particular, we allow the agent to ask for a particular concept related to the image that it is uncertain about, and not require the full caption from the teacher. Our model is composed of three modules, *i.e.* captioning, decision making and question posing, which interact with each other in a lifetime learning setting. Learning and teaching efficiency is shown to be improved on the MS-COCO dataset. Our work is the first step towards a more natural learning setting in which data arrives continuously, and robots learn from humans through natural language questions and feedback. There are many challenges ahead in making the lifetime model learning more efficient, and incorporating real humans in the loop.

Acknowledgements Supported by the DARPA Explainable AI (XAI) program. We thank NVIDIA for their donation of GPUs. We thank Relu Patrascu for infrastructure support, David Acuna and Seung Wook Kim for fruitful discussion. SF acknowledges the Canada CIFAR AI Chair award at Vector Institute.

References

[1] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2425–2433, 2015. 5

[2] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005. 5

[3] Bo Dai, Sanja Fidler, Raquel Urtasun, and Dahua Lin. Towards diverse and natural image descriptions via a conditional gan. *arXiv preprint arXiv:1703.06029*, 2017. 1, 2

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009. 5

[5] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[6] Akshay Kumar Gupta. Survey of visual question answering: Datasets and techniques. *arXiv preprint arXiv:1705.03865*, 2017. 1, 2

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[8] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 1988–1997. IEEE, 2017. 2

[9] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137, 2015. 5

[10] Seung Wook Kim, Makarand Tapaswi, and Sanja Fidler. Progressive reasoning by module composition. *arXiv preprint arXiv:1806.02453*, 2018. 2

[11] Jiwei Li, Alexander H. Miller, Sumit Chopra, Marc'Aurelio Ranzato, and Jason Weston. Learning through dialogue interactions by asking questions. *arXiv:1612.04936*, 2016. 2

[12] Yikang Li, Nan Duan, Bolei Zhou, Xiao Chu, Wanli Ouyang, Xiaogang Wang, and Ming Zhou. Visual question generation as dual task of visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6116–6124, 2018. 2

[13] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out*, 2004. 5

[14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 1, 2, 5

[15] Huan Ling and Sanja Fidler. Teaching machines to describe images via natural language feedback. *arXiv preprint arXiv:1706.00130*, 2017. 2

[16] Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60, 2014. 5

[17] Maja J Matarić. Socially assistive robotics: Human augmentation versus automation. *Science Robotics*, 2(4):eaam5410, 2017. 1

[18] Ishan Misra, Ross Girshick, Rob Fergus, Martial Hebert, Abhinav Gupta, and Laurens van der Maaten. Learning by asking questions. *arXiv preprint arXiv:1712.01238*, 2017. 2

[19] Nasrin Mostafazadeh, Ishan Misra, Jacob Devlin, Margaret Mitchell, Xiaodong He, and Lucy Vanderwende. Generating natural questions about an image. *arXiv preprint arXiv:1603.06059*, 2016. 2

[20] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002. 5

[21] Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jarret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. *arXiv preprint arXiv:1612.00563*, 2016. 1, 2, 4

[22] Burr Settles. Active learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(1):1–114, 2012. 1, 2

[23] Edgar Simo-Serra, Sanja Fidler, Francesc Moreno-Noguer, and Raquel Urtasun. Neuroaesthetics in fashion: Modeling the perception of fashionability. In *CVPR*, volume 2, page 6, 2015. 1

[24] Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in neural information processing systems*, pages 1057–1063, 2000. 4

[25] Damien Teney, Peter Anderson, Xiaodong He, and Anton van den Hengel. Tips and tricks for visual question answering: Learnings from the 2017 challenge. *arXiv preprint arXiv:1708.02711*, 2017. 1, 2, 5

[26] Kohei Uehara, Antonio Tejero-De-Pablos, Yoshitaka Ushiku, and Tatsuya Harada. Visual question generation for class acquisition of unknown objects. *arXiv preprint arXiv:1808.01821*, 2018. 2

[27] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575, 2015. 5

[28] Oriol Vinyals and Quoc Le. A neural conversational model. *arXiv preprint arXiv:1506.05869*, 2015. 1

[29] Qi Wu, Damien Teney, Peng Wang, Chunhua Shen, Anthony Dick, and Anton van den Hengel. Visual question answering: A survey of methods and datasets. *Computer Vision and Image Understanding*, 163:21–40, 2017. 1

[30] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning*, pages 2048–2057, 2015. 2, 4

[31] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Visual curiosity: Learning to ask questions to learn visual recognition. *arXiv preprint arXiv:1810.00912*, 2018. 2