Mind Your Language: Learning Visually Grounded Dialog in a Multi-Agent Setting

Anonymous EMNLP submission

Abstract

The task of visually grounded dialog involves learning goal-oriented cooperative dialog between autonomous agents who exchange information about a scene through several rounds of questions and answers. We posit that requiring agents to adhere to rules of human language while also maximizing information exchange is an ill-posed problem, and observe that humans do not stray from a common language, because they are social creatures and have to communicate with many people everyday, and it is far easier to stick to a common language even at the cost of some efficiency loss. Using this as inspiration, we propose and evaluate a multi-agent dialog framework where each agent interacts with, and learns from, multiple agents, and show that this results in more relevant and coherent dialog (as judged by human evaluators) without sacrificing task performance (as judged by quantitative metrics).

1 Introduction

Intelligent assistants like Siri and Alexa are increasingly becoming an important part of our daily lives, be it in the household, the workplace or in public places. As these systems become more advanced, we will have them interacting with each other to achieve a particular goal (Leviathan, 2018). We want these conversations to be interpretable to humans for the sake of transparency and ease of debugging. Having the agents communicate in natural language is one of the most universal ways of ensuring interpretability. This motivates our work on goal-driven agents which interact in coherent language understandable to humans.

To that end, this paper builds on work by (Das et al., 2017) on goal-driven visual dialog agents. The task is formulated as a conversation between two agents, a Question (Q-) and an Answer (A-) bot. The A-Bot is given an image, while the Q-Bot is given only a caption to the image. Both agents share a common objective, which is for the Q-Bot to form an accurate mental representation of the unseen image using which it can retrieve, rank or generate that image. This is facilitated by the exchange of 10 pairs of questions and answers between the two agents, using a shared vocabulary. (Das et al., 2017) trained the agents first in isolation via supervision from the VisDial dataset (Das et al., 2016), followed by making them interact and adapt to each other via reinforcement learning by optimizing for better task performance. While trying to maximize performance, the agents learn to communicate in non-grammatical and semantically meaningless sentences in order to maximize the exchange of information. This reduces transparency of the AI system to human observers and is undesirable. We address this problem by proposing a **multi-agent dialog framework** where each agent interacts with multiple agents. This is motivated by our observation that humans adhere to syntactically and semantically coherent language, which we hypothesize is because they have to interact with an entire community, and having a private language for each person would be extremely inefficient. We show that our multi-agent (with multiple Q-Bots and multiple A-Bots) dialog system results in more coherent and human-interpretable dialog between agents, without compromising on task performance, which also validates our hypothesis. This makes them seem more helpful, transparent and trustworthy. We will make our code available as open-source.[1]

2 Objective

The game involves two collaborative agents a question bot (Q-bot) and an answer bot (A-bot). The A-Bot is provided an image, I (represented as a feature embedding y_{gt} extracted by, say, a pretrained

---

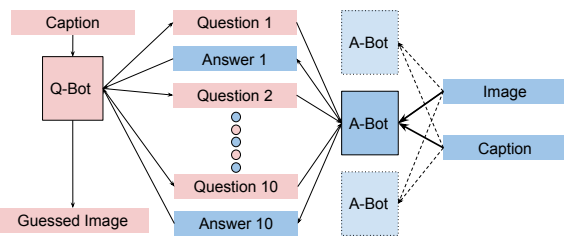[1] https://github.com/anonuser/repo.git

1

Figure 1: Multi-Agent (with 1 Q-Bot, 3 A-Bots) Dialog Framework

CNN model (Simonyan and Zisserman, 2014)), while the Q-Bot is provided with only a caption of the image. The Q-Bot is tasked with estimating a vector representation $\hat{y}$ of I, which is used to retrieve that image from a dataset. Both agents receive a common penalty from the environment which is equal to the error in $\hat{y}$ with respect to y_{gt}. Thus, an unlimited number of games may be simulated without human supervision, motivating the use of reinforcement learning in this framework.

Our primary focus for this work is to ensure that the agents' dialog remains coherent and understandable while also being informative and improving task performance. For concreteness, an example of dialog that is informative yet incoherent: question: "do you recognize the guy and age is the adult?", answered with: "you couldn't be late teens, his". The example shows that the bots try to extract and convey as much information as possible in a single question/answer (sometimes by incorporating multiple questions or answers into a single statement). But in doing so they lose basic semantic and syntactic structure.

3 Relevant Work

Most of the major works which combine vision and language have traditionally been focusing on the problem of image captioning (((Kiros et al., 2014), (Xu et al., 2015), (Vinyals et al., 2014), (Johnson et al., 2015), (Lu et al., 2016), (Yao et al., 2016)) and visual question answering ((Agrawal et al., 2017), (Zhang et al., 2016), (Goyal et al., 2017), (Yang et al., 2016)). The problem of visual dialog is relatively new and was first introduced by (Das et al., 2016) who also created the VisDial dataset to advance the research on visually grounded dialog. The dataset was collected by pairing two annotators on Amazon Mechanical Turk to chat about an image. They formulated the task as a 'multi-round' VQA task and evaluated individual responses at each round in an image guessing setup. In a subsequent work by (Das et al., 2017) they proposed a Reinforcement Learning based setup where they allowed the Question bot and the Answer bot to have a dialog with each other with the goal of correctly predicting the image unseen to the Question bot. However, in their work they noticed that the reinforcement learning based training quickly lead the bots to diverge from natural language. In fact (Kottur et al., 2017) recently showed that language emerging from two agents interacting with each other might not even be interpretable or compositional. Our multi-agent framework aims to alleviate this problem and prevent the bots from developing a specialized language between them. Interleaving supervised training with reinforcement learning also helps prevent the bots from becoming incoherent and generating non-sensical dialog. Recent work has also proposed using such goal driven dialog agents for other related tasks including negotiation (Lewis et al., 2017) and collaborative drawing (Kim et al., 2017). We believe that our work can easily extend to those settings as well. (Lu et al., 2017) proposed a generative-discriminative framework for visual dialog where they train only an answer bot to generate informative answers for ground truth questions. These answers were then fed to a discriminator, which was trained to rank the generated answer among a set of candidate answers. This is a major restriction of their model as it can only be trained when this additional information of candidate answers is available, which restricts it to a supervised setting. Furthermore, since they train only the answer bot and have no question bot, they cannot simulate an entire dialog which also prevents them from learning by self-play via reinforcement. (Wu et al., 2017) further improved upon this generative-discriminative framework by formulating the discriminator as a more traditional GAN (Goodfellow et al., 2014), where the adversarial discriminator is tasked to distinguish between human generated and machine generated dialogs. Furthermore, unlike (Lu et al., 2017) they modeled the discriminator using an attention network which also utilized the dialog history in predicting the next answer allowing it to maintain coherence and consistency across dialog turns.

4 Agent Architectures

We briefly describe the agent architectures and leave the details for the appendix.

4.1 Question Bot Architecture

The question bot architecture we use is inspired by the answer bot architecture in (Das et al., 2017) and (Lu et al., 2017) but the individual units have been modified to provide more useful representations. Similar to the original architecture, our Q-Bot, shown in Fig. 2a, also consists of 5 parts, (a) fact encoder, (b) state-history encoder, (c) caption encoder, (d) image regression network and (e) question decoder. The fact encoder is modeled using a Long-Short Term Memory (LSTM) network, which encodes the previous question-answer pair into an embedding F_t^Q. We modify the state-history encoder to incorporate a two-level hierarchical encoding of the dialog. It uses the fact embedding F_t^Q at each time step to compute attention over the history of dialog, $(F_1^Q, F_2^Q, F_3^Q ... F_{t-1}^Q)$ and produce a history encoding S_t^Q. The key modification (compared to (Lu et al., 2017)) in our architecture is the addition of a separate LSTM to compute a caption embedding C. This is key to ensuring that the hierarchical encoding does not exclusively attend on the caption while generating the history embedding, and prevents the occurrence of repetitive questions in a dialog since the encoding now has an adequate representation of the previous facts. The caption embedding is then concatenated with F_t^Q and S_t^Q, and passed through multiple fully connected layers to compute the state-history encoder embedding e_t and the predicted image feature embedding $\hat{y}_t = f(S_t^Q)$. The encoder embedding, e_t^Q is fed to the question decoder, another LSTM, which generates the question, q_t. For all LSTMs and fully connected layers in the model we use a hidden layer size of 512. The image feature vector is 4096 dimensional. The word embeddings and the encoder embeddings are 300 dimensional.

4.2 Answer Bot Architecture

The architecture for A-Bot, also inspired from (Lu et al., 2017), shown in Fig. 2b, is similar to that of the Q-Bot. It has 3 components: (a) question encoder, (b) state-history encoder and (c) answer decoder. The question encoder computes an embedding, Q_t for the question to be answered, q_t. The history encoding $(F_1^A, F_2^A, F_3^A ... F_t^A) \rightarrow S_t^A$ uses a similar two-level hierarchical encoder, where the attention is computed using the question embedding Q_t. The caption is passed on to the A-Bot as the first element of the history, which is why we do not use a separate caption encoder. Instead,

(a) Encoder architecture for Q-Bot

(b) Encoder architecture for A-Bot
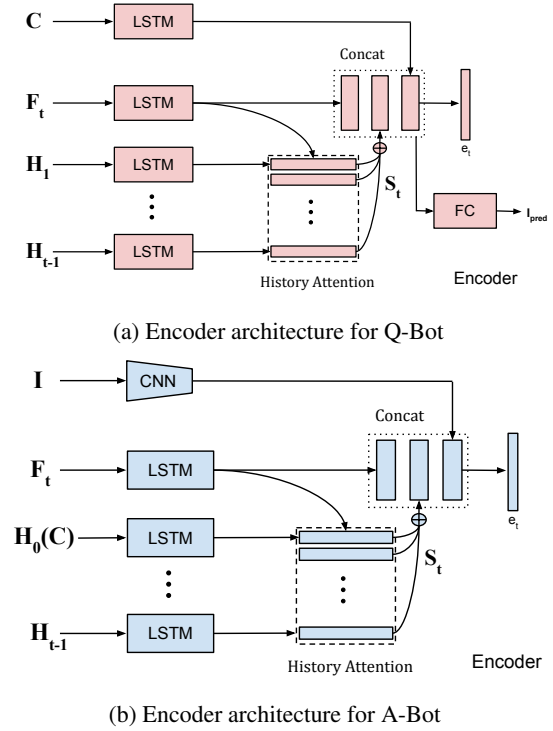
Figure 2

we use the fc7 feature embedding of a pretrained VGG-16 (Simonyan and Zisserman, 2014) model to compute the image embedding I. The three embeddings S_t^A, Q_t, I are concatenated and passed through another fully connected layer to extract the encoder embedding e_t^A. The answer decoder, which is another LSTM, uses this embedding e_t^A to generate the answer a_t. Similar to the Q-Bot, we use a hidden layer size of 512 for all LSTMs and fully connected layers. The image feature vector coming from the CNN is 4096 dimensional (FC7 features from VGG16). The word embeddings and the encoder embeddings are 300 dimensional.

5 Training

We follow the training process proposed in (Das et al., 2017). Two agents, a Q-Bot and an A-Bot are first trained in isolation, by supervision from the VisDial dataset. After this supervised pretraining for 15 epochs over the data, we smoothly transition the agents to learn by reinforcement via a curriculum. Specifically, for the first K rounds of dialog for each image, the agents are trained using supervision from the VisDial dataset. For the remaining 10-K rounds, however, they are trained via reinforcement learning. K starts at 9 and is linearly annealed to 0 over 10 epochs. The individual phases of training will be described below, with

3

some details in the appendix.

5.1 Supervised pre-training

In the supervised part of training, both the Q-Bot and A-Bot are trained separately. Both the Q-Bot and A-Bot are trained with a Maximum Likelihood Estimation (MLE) loss computed using the ground truth questions and answers, respectively, for every round of dialog. The Q-Bot simultaneously optimizes another objective: minimizing the Mean Squared Error (MSE) loss between the true and predicted image embeddings. The ground truth dialogs and image embeddings come from the Vis-Dial dataset.

Given the true dialog history, image features and ground truth question, the A-Bot generates an answer to that question. Given the true dialog history and the previous question-answer pair, the Q-Bot is made to generate the next question to ask the A-Bot. However, there are multiple problems with this training scheme. First, MLE is known to result in models that generate repetitive dialogs and often produce generic responses. Second, the agents are never allowed to interact during training. Thus, when they interact during testing, they end up facing out of distribution questions and answers, and produce unexpected responses. Third, the sequentiality of dialog is lost when the agents are trained in an isolated, supervised manner.

5.2 Reinforcement Learning Setup

To alleviate the issues pointed out with supervised training, we let the two bots interact with each other via self-play (no ground-truth except images and captions). This interaction is also in the form of questions asked by the Q-Bot, and answered in turn by the A-Bot, using a shared vocabulary. The state space is partially observed and asymmetric, with the Q-Bot observing $\{c, q_1, a_1 \ldots q_{10}, a_{10}\}$ and the A-Bot observing the same, plus the image I. Here, c is the caption, and q_i, a_i is the ith dialog pair exchanged where $i = 1 \ldots 10$. The action space for both bots consists of all possible output sequences of a specified maximum length (Q-Bot: 16, A-Bot: 9 as specified by the dataset) under a fixed vocabulary (size 8645). Note that these parameter values are chosen to comply with the VisDial dataset. Each action involves predicting words sequentially until a stop token is predicted, or the generated statement reaches the maximum length. Additionally, Q-Bot also produces a guess of the visual representation of the input image (VGG fc-7

embedding of size 4096). Both Q-Bot and A-Bot share the same objective and get the same reward to encourage cooperation. Information gain in each round of dialog is incentivized by setting the reward as the **change in distance** of the predicted image embedding to the ground-truth image representation. This means that a QA pair is of high quality only if it helps the Q-Bot make a better prediction of the image representation. Both policies are modeled by neural networks, as discussed in Section 4.

However, as noted above, this RL optimization problem is ill-posed, since rewarding the agents for information exchange does not motivate them to stick to the rules and conventions of the English language. Thus, we follow the elaborate curriculum scheme described above, despite which the bots are still observed to diverge from natural language and produce non-grammatical and incoherent dialog. Thus, we propose a multi bot architecture to help the agents interact in diverse and coherent, yet informative, dialog.

Learning Algorithm: A dialog round at time t consists of the following steps: 1) The Q-Bot, conditioned on the state encoding, generates a question q_t, 2) A-Bot updates its state encoding with q_t and then generates an answer a_t, 3) Both Q-Bot and A-Bot encode the completed exchange as a fact embedding, 4) Q-Bot updates its state encoding to incorporate this fact and finally 5) Q-Bot predicts the image representation for the unseen image conditioned on its updated state encoding.

Similar to (Das et al., 2016), we use the REIN-FORCE (Williams, 1992) algorithm that updates network parameters in response to experienced rewards. The per-round rewards maximized are:

$$r_t(s_t^Q, (q_t, a_t, y_t)) = l(\hat{y}_{t-1}, y^{gt}) - l(\hat{y}_t, y^{gt}) \quad (1)$$

This reward is positive if the distance between image representation generated at time t is closer to the ground truth than the representation at time $t - 1$, hence incentivizing information gain at each round of dialog. The REINFORCE update rule ensures that a (q_t, a_t) exchange that is informative has its probabilities pushed up. Do note that the image feature regression network f is trained directly via supervised gradient updates on the L-2 loss.

5.3 Multi-Agent Dialog Framework (MADF)

In this section we describe our proposed Multi-Agent Dialog architecture in detail. The motivation

---

Algorithm 1 Multi-Agent Dialog Framework (MADF)

1: **procedure** MULTIBOTTRAIN
2:    **while** train_iter < max_train_iter **do**   ▷ Main Training loop over batches
3:       $Qbot \leftarrow random\_select\,(Q_1, Q_2, Q_3....Q_q)$
4:       $Abot \leftarrow random\_select\,(A_1, A_2, A_3....A_a)$ ▷ Either q or a is equal to 1
5:       $history \leftarrow (0, 0, ...0)$ ▷ History initialized with zeros
6:       $fact \leftarrow (0, 0, ...0)$ ▷ Fact encoding initialized with zeros
7:       $\Delta image\_pred \leftarrow 0$ ▷ Tracks change in Image Embedding
8:       $Qz_1 \leftarrow Ques\_enc(Qbot, fact, history, caption)$
9:       **for** t in 1:10 **do**   ▷ Have 10 rounds of dialog
10:          $ques_t \leftarrow Ques\_gen(Qbot, Qz_t)$
11:          $Az_t \leftarrow Ans\_enc(Abot, fact, history, image, ques_t, caption)$
12:          $ans_t \leftarrow Ans\_gen(Abot, Az_t)$
13:          $fact \leftarrow [ques_t, ans_t]$ ▷ Fact encoder stores the last dialog pair
14:          $history \leftarrow concat(history, fact)$ ▷ History stores all previous dialog pairs
15:          $Qz_t \leftarrow Ques\_enc(Qbot, fact, history, caption)$
16:          $image\_pred \leftarrow image\_regress(Qbot, fact, history, caption)$
17:          $R_t \leftarrow (target\_image - image\_pred)^2 - \Delta image\_pred$
18:          $\Delta image\_pred \leftarrow (target\_image - image\_pred)^2$
19:       **end for**
20:       $\Delta(w_{Qbot}) \leftarrow \frac{1}{10}\sum_{t=1}^{10}\nabla_{\theta_{Qbot}}\left[G_t \log p(ques_t, \theta_{Qbot}) - \Delta image\_pred\right]$
21:       $\Delta(w_{Abot}) \leftarrow \frac{1}{10}\sum_{t=1}^{10}G_t \nabla_{\theta_{Abot}} \log p(ans_t, \theta_{Abot})$
22:       $w_{Qbot} \leftarrow w_{Qbot} + \Delta(w_{Qbot})$ ▷ REINFORCE and Image Loss update for Qbot
23:       $w_{Abot} \leftarrow w_{Abot} + \Delta(w_{Abot})$ ▷ REINFORCE update for Abot
24:    **end while**
25: **end procedure**

---

behind this is the observation that allowing a pair of agents to interact with each other and learn via reinforcement for too long leads to them developing an idiosyncratic private language which does not adhere to the rules of human language, and is hence not understandable by human observers. We claim that if instead of allowing a single pair of agents to interact, we were to make the agents more social, and make them *interact and learn from multiple other agents*, they would be disincentivized to develop a private language, and would have to conform to the common language.

In particular, we create either multiple Q-bots to interact with a single A-bot, or multiple A-bots to interact with a single Q-bot. All these agents are initialized with the learned parameters from the supervised pretraining as described in Section 5.1. Then, for each batch of images from the VisDial dataset, we randomly choose a Q-bot to interact with the A-bot, or randomly choose an A-bot to interact with the Q-bot, as the case may be. The two chosen agents then have a complete dialog consisting of 10 question-answer pairs about each of

those images, and update their respective weights based on the rewards received (as per Equation 1) during the conversation, using the REINFORCE algorithm. This process is repeated for each batch of images, over the entire VisDial dataset. It is important to note that histories are *not shared* across batches. MADF can be understood in detail using the pseudocode in Algorithm 1.

6 Experiments and Results

6.1 Dataset description

We use the VisDial 0.9 dataset for our task introduced by (Das et al., 2016). The data is collected using Amazon Mechanical Turk by pairing 2 annotators and asking them to chat about the image as a multi round VQA setup. One of the annotators acts as the questioner and has access to only the caption of the image and has to ask questions from the other annotator who acts as the answerer and must answer the questions based on the visual information from the actual image. This dialog repeats for 10 rounds at the end of which the questioner has to guess what the image was. We perform our

Model	MRR	Mean Rank	R@1	R@5	R@10
Answer Prior (Das et al., 2016)	0.3735	26.50	23.55	48.52	53.23
MN-QIH-G (Das et al., 2016)	0.5259	17.06	42.29	62.85	68.88
HCIAE-G-DIS (Lu et al., 2017)	0.547	14.23	44.35	65.28	71.55
Frozen-Q-Multi (Das et al., 2017)	0.437	21.13	N/A	53.67	60.48
CoAtt-GAN (Wu et al., 2017)	0.5578	14.4	**46.10**	**65.69**	71.74
SL(Ours)	**0.610**	**5.323**	34.74	57.67	**72.68**
RL - 1Q,1A(Ours)	0.459	7.097	16.04	54.69	72.34
RL - 1Q,3A(Ours)	0.601	5.495	34.83	57.47	72.48
RL - 3Q,1A(Ours)	0.590	5.56	33.59	57.73	72.61

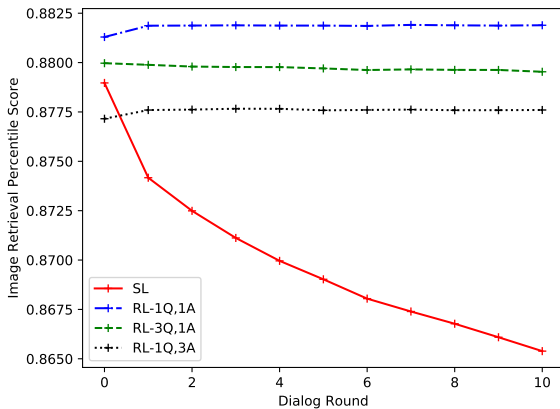Table 1: Comparison of Metrics with Literature

Figure 3: The percentile scores of the ground truth image compared to the entire test set of 40k images. The X-axis denotes the dialog round number (from 1 to 10), while the Y-axis denotes the image retrieval percentile score.

experiments on VisDial v0.9 (the latest available release) containing 83k dialogs on COCO-train and 40k on COCO-val images, for a total of 1.2M dialog question-answer pairs. We split the 83k into 82k for train, 1k for validation, and use the 40k as test, in a manner consistent with (Das et al., 2016). The caption is considered to be the first round in the dialog history.

6.2 Evaluation Metrics

We evaluate the performance of our model using 6 metrics, proposed by (Das et al., 2017): **1) Mean Reciprocal Rank (MRR)**, **2) Mean Rank**, **3) Recall@1**, **4) Recall@5**, **5) Recall@10** and **6) Image Retrieval Percentile**. Mean Rank and MRR compute the average rank (and its reciprocal, respectively) assigned to the answer generated by the A-bot, over a set of 100 candidate answers for each question (also averaged over all the 10 rounds). Recall@k computes the percentage of answers with

rank less than k. Image Retrieval percentile is a measure of how close the image prediction generated by the Q-bot is to the ground truth. All the images in the test set are ranked according to their distance from the predicted image embedding, and the rank of the ground truth embedding is used to calculate the image retrieval percentile.

Table 1 compares the Mean Rank, MRR, Recall@1, Recall@5 and Recall@10 of our agent architecture and dialog framework (below the horizontal line) with previously proposed architectures (above the line). SL refers to the agents after only the isolated, supervised training of Section 5.1. RL-1Q,1A refers to a single, idiosyncratic pair of agents trained via reinforcement as in Section 5.2. RL-1Q,3A and RL-3Q,1A refer to social agents trained via our Multi-Agent Dialog framework in Section 5.3, with 1Q,3A referring to 1 Q-Bot and 3 A-Bots, and 3Q,1A referring to 3 Q-Bots and 1 A-Bot. It can be seen that our agent architectures clearly outperform all previously proposed generative architectures in MRR, Mean Rank and R@10, but not in R@1 and R@5. This indicates that our approach produces consistently good answers (as measured by MRR, Mean Rank and R@10), even though they might not be the best possible answers (as measured by R@1 and R@5). SL has the best MRR and Mean Rank, which drops drastically in RL-1Q,1A. The agents trained by MADF recover and are able to outperform all previously proposed models. Fig. 3 shows image retrieval percentile scores over dialog rounds. The percentile score decreases monotonically for SL, but is stable for all versions using RL.

6.3 Human Evaluation

There are no quantitative metrics to comprehensively evaluate dialog quality, hence we do a hu-

	Metric	N	Supervised	RL 1Q,1A	RL 1Q,3A	RL 3Q,1A
1	Q-Bot Relevance	8	2.5	2.75	**2**	2.75
2	Q-Bot Grammar	8	**2.25**	2.875	2.5	2.375
3	A-Bot Relevance	12	2.5	2.583	2.25	**1.67**
4	A-Bot Grammar	12	1.92	3.5	**1.83**	2.25
5	Overall Coherence	20	2.8	3.05	**2.3**	**1.85**

Table 2: Human Evaluation Results - Mean Rank (Lower is better). **N** refers to the number of human evaluators involved in the ranking.

man evaluation of the generated dialog. There are 5 metrics we evaluate on: 1) Q-Bot Relevance, 2) Q-Bot Grammar, 3)A-Bot Relevance, 4) A-Bot Grammar and 5) Overall Dialog Coherence. We evaluate 4 Visual Dialog systems, trained via: 1) **Supervised Learning (SL)**, 2) **Reinforce for 1 Q-Bot, 1 A-Bot (RL-1Q,1A)**, 3) **Reinforce for 1 Q-Bot, 3 A-Bots (RL-1Q,3A)** and 4) **Reinforce for 3 Q-Bots, 1 A-Bot (RL-3Q,1A)**. A total of 20 evaluators (randomly chosen students) were shown the caption and the 10 QA-pairs generated by each system for one of 4 randomly chosen images, and asked to give an ordinal ranking (from 1 to 4) for each metric. If the evaluator was also given access to the image, she was asked only to evaluate metrics 3, 4 and 5, while if the evaluator was not shown the image, she was asked only to evaluate metrics 1, 2 and 5. Table 2 contains the average ranks obtained on each metric (lower is better).

The results convincingly prove our hypothesis that having multiple A-Bots to interact and learn from will improve the Q-Bot, and vice versa. This is because having multiple A-Bots to interact with gives the Q-Bot access to a variety of diverse dialog, leading to more stable updates with lower bias. The results confirm this, with Q-Bot Relevance rank being best in RL-1Q,3A, and A-Bot Relevance rank being best in RL-3Q,1A. These two dialog systems, which were trained via MADF, also have the best overall dialog coherence by a significant margin over RL-1Q,1A and SL. We show two of the examples shown to the human evaluators in Figure 4. The trends observed in the scores given by human evaluators is also clearly visible in these examples. MADF agents are able to model the human responses much better than the other agents. It can also be seen that although the RL-1Q,1A system has greater diversity in its responses, the quality of those responses is greatly degraded, with the A-Bot's answers especially being both non-grammatical and irrelevant. In Section 5.1, we

discussed how the MSE loss used in SL results in models which generate repetitive dialog, which can be seen in Fig. 4. Consider the first image, where in the SL QA-generations, the Q-Bot repeats the same questions multiple times, and gets inconsistent answers from the A-Bot for the same question. By contrast, all 10 QA-generations for RL-3Q,1A are grammatically correct. The Q-Bot's questions are very relevant to the image being considered, and the A-Bot's answers appropriate and correct.

7 Discussion and Conclusion

In this paper we propose a novel Multi-Agent Dialog Framework (MADF), inspired from human communities, to improve the dialog quality of AI agents. We show that training 2 agents with supervised learning can lead to uninformative and repetitive dialog. Furthermore, we observe that the task performance (measured by the image retrieval percentile scores) for the system trained via supervision only deteriorates as dialog round number increases. We hypothesize that this is because the agents were trained in isolation and never allowed to interact during supervised learning, which leads to failure during testing when they encounter out of distribution samples (generated by the other agent, instead of ground truth) for the first time. We show how allowing a single pair of agents to interact and learn from each other via reinforcement learning dramatically improve their percentile scores, which additionally does not deteriorate over multiple rounds of dialog, since the agents have interacted with one another and been exposed to the other's generated questions or answers. However, the agents, in an attempt to improve task performance end up developing their own private language which does not adhere to the rules and conventions of human languages, and generates non-grammatical and non-sensical statements. As a result, the dialog system loses interpretability and sociability. To alleviate this issue, we propose a

7

The little girl is standing with skis on her feet

Human-Human Dialogue

what color are the skis ?	A UNK color
Are there any other people?	Not that i can see
Is this outdoors?	Yes
Do you see snow?	Yes
Is it currently snowing?	No I don't think so
Is she on a slope or hill?	No i don't think so
Do you see trees?	Yes
Do you see the sky?	No
Is she wearing gloves?	Yep
Is she wearing a hat?	yes

SL

is the girl close to a race ?	no , woman is riding
'is she wearing a helmet ?	no
is she wearing a hat ?	She is wearing a headband
what color is it ?	I would say it's black
'is she wearing goggles ?	Can't see her face
'is she wearing goggles ?	Not sure
what color is her skis ?	Dark, hard to tell i would guess
is she wearing a hat ?	No
what color does she have ?	White
and what is the girl wearing ?	A white striped long sleeve shirt and blue

RL - 1Q, 1A

is there more than 1 skier ?	Yes
oh how many UNK do you see ?	A few
is the person holding a pipe of so me sort of basket ?	Yes, there is a bowtie
is lady going fast ?	I can not tell
how many is there ?	It's not silver people are in grocery
is there snow ?	Is covered by snow covered by snow
how many animals do you see ?	Of the 1, and a 2
is there a lot of snow ?	Yes tons of snow everywhere covered
are they all together ?	Yes 1 is a darker 1
is there a lift ?	Just a few skiing out yet

RL - 1Q, 3A

are there any other people around ?	No
is it snowing ?	Can't tell
is it a sunny day ?	Yes
is the woman wearing a dress ?	Yes
what color is it ?	It has a white meat and girl on
is she by herself of snow ?	Just lady, woman and dog
what color is her jacket ?	Orange
does it look like a competition ?	Yes
is it day or night ?	Daytime
is it sunny ?	yes

RL - 3Q, 1A

how old is the woman ?	About 16
is she in a ski resort ?	I don't think so
can you see the sky ?	No
is she alone ?	Yes
what color are the gloves ?	Silver
is there snow on the ground ?	No snow
is the photo very clear ?	Yes
is she wearing a helmet ?	No
is it snowing ?	it looks like it is stopped
is it sunny ?	No really

2 men in a grassy park playing frisbee

Human-Human Dialogue

How many men?	2
Are they playing frisbee?	Yes
What color frisbee?	White
Just white?	Yes
Do men look skilled?	No
Why not?	Playing
They don't seem good?	Guess not, I said no
Is frisbee in air?	No
Who has it?	Person 1
Is he bigger than other?	no

SL

are the boys playing volleyball ?	yes , a professional boy maybe in their late 20 's
how old do they look ?	no , i don't see buildings
do you see any buildings ?	yes , both men and women
are they male ?	no , can only see the bunch of
can you see any cars ?	there are more people than 3
can you tell how many women	no , they 're wood
are sitting there or playing ?	women
are they combing leather ?	they are black
what color are the men 's shoes ?	'the snowboarder is white
what color are the women ?	yes
are the boys well groomed ?	

RL - 1Q, 1A

do you recognize 1 ball ?	looks like a around
that you could you are seeing controllers ?	that i can see the fair
do you recognize the guy and age is the adult ?	you couldn't be late teens , his
do you recognize them ?	motorcycle but UNK tint and UNK
how many people are there ?	35
do you do they fence famous ?	It looks black and white
do you see fire ?	I'm not sure because it's is
is the fire dog UNK ?	Yes
do they have long manes ?	Black and white, it looks
can you see other people watching them ?	Yes there is a people

RL - 1Q, 3A

How old do the men appear?	30s
Is this at a beach?	No
Do they have on bathing suits?	No
How old are they?	Young adults
What color frisbee?	White
Do they have a regular ball shirt on?	1 of them do
With how old are they?	Mid 30s
Is there other people in the pic?	Yes, there is a man behind the him
How many of them are playing?	2
What is the woman doing?	sitting

RL - 3Q, 1A

What color is umbrella?	Black with a blue stripe
What are they wearing?	T shirts and jeans
What color is frisbee?	White
What are they doing?	Sitting on the beach, talking
Are they all holding rackets?	Yes
Are there any other people?	Yes
What color is the frisbee?	Creamy green
Are there any other people?	Yes a lot
Are the people tall?	Looks very tall
Are they in a field?	no

Figure 4: Two randomly selected images from the VisDial dataset followed by the ground truth (human) and generated dialog about that image for each of our 4 systems (SL, RL-1Q,1A, RL-1Q,3A, RL-3Q,1A). These images were also used in the human evaluation results shown in Table 2

.

multi-agent dialog framework based on self-play reinforcement learning, where a single A-Bot is allowed to interact with multiple Q-Bots and vice versa. Through a human evaluation study, we show that this leads to significant improvements in dialog quality measured by relevance, grammar and coherence. This is because interacting with multiple agents prevents any particular pair from maximizing performance by developing a private language, since it would harm performance with all the other agents.

8 Future Work

We plan to explore several other multi bot architectural settings and perform a more thorough human evaluation for qualitative analysis of our dialog. We also plan on incorporating an explicit perplexity based reward term in our reinforcement learning setup to further improve the dialog quality. We will also experiment with using a discriminative answer decoder which uses information of the possible answer candidates to rank the generated answer with respect to all the candidate answers and use the ranking performance to train the answer decoder.

Another avenue for future exploration is to use a richer image feature embedding to regress on. Currently, we use a regression network to compute the estimated image embedding which represents the Q-Bot's understanding of the image. We plan to implement an image generation GAN which can use this embedding as a latent code to generate an image which can be visualized. While the MADF in its current form only works if we have multiple Q-Bots or multiple A-Bots but not both, future work could possibly look at incorporating that into the framework, while ensuring that the updates do not become too unstable.

References

Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Devi Parikh, and Dhruv Batra. 2017. Vqa: Visual question answering. *Int. J. Comput. Vision*, 123(1):4–31.

Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José M. F. Moura, Devi Parikh, and Dhruv Batra. 2016. Visual dialog. *CoRR*, abs/1611.08669.

Abhishek Das, Satwik Kottur, José M. F. Moura, Stefan Lee, and Dhruv Batra. 2017. Learning cooperative visual dialog agents with deep reinforcement learning. *CoRR*, abs/1703.06585.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc.

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 *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Justin Johnson, Andrej Karpathy, and Fei-Fei Li. 2015. Densecap: Fully convolutional localization networks for dense captioning. *CoRR*, abs/1511.07571.

Jin-Hwa Kim, Devi Parikh, Dhruv Batra, Byoung-Tak Zhang, and Yuandong Tian. 2017. Codraw: Visual dialog for collaborative drawing. *arXiv preprint arXiv:1712.05558*.

Ryan Kiros, Ruslan Salakhutdinov, and Richard S. Zemel. 2014. Unifying visual-semantic embeddings with multimodal neural language models. *CoRR*, abs/1411.2539.

Satwik Kottur, José M. F. Moura, Stefan Lee, and Dhruv Batra. 2017. Natural language does not emerge 'naturally' in multi-agent dialog. *CoRR*, abs/1706.08502.

Yaniv Leviathan. 2018. Google duplex: An ai system for accomplishing real-world tasks over the phone.

Mike Lewis, Denis Yarats, Yann N Dauphin, Devi Parikh, and Dhruv Batra. 2017. Deal or no deal? end-to-end learning for negotiation dialogues. *arXiv preprint arXiv:1706.05125*.

Jiasen Lu, Anitha Kannan, Jianwei Yang, Devi Parikh, and Dhruv Batra. 2017. Best of both worlds: Transferring knowledge from discriminative learning to a generative visual dialog model. *CoRR*, abs/1706.01554.

Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. 2016. Knowing when to look: Adaptive attention via A visual sentinel for image captioning. *CoRR*, abs/1612.01887.

Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2014. Show and tell: A neural image caption generator. *CoRR*, abs/1411.4555.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. In *Reinforcement Learning*, pages 5–32. Springer.

Qi Wu, Peng Wang, Chunhua Shen, Ian D. Reid, and Anton van den Hengel. 2017. Are you talking to me? reasoned visual dialog generation through adversarial learning. *CoRR*, abs/1711.07613.

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. *CoRR*, abs/1502.03044.

Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. 2016. Stacked attention networks for image question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 21–29.

Ting Yao, Yingwei Pan, Yehao Li, Zhaofan Qiu, and Tao Mei. 2016. Boosting image captioning with attributes. *CoRR*, abs/1611.01646.

Peng Zhang, Yash Goyal, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2016. Yin and Yang: Balancing and answering binary visual questions. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

9

A Agent Architectures

A.1 Q-Bot

As discussed in Section 4.1, the Q-bot architecture has 5 components, the first 4 of which belong to the question encoder.

1. **Fact Encoder:** The fact encoder is a unidirectional LSTM which is given the previous question-answer pair (q_{t-1}, a_{t-1}) as input. The LSTM generates a fact embedding $F_t^Q \in R^{512}$.

2. **State/History Encoder:** The encoder first computes the fact embeddings $H_t^Q = (F_1^Q, F_2^Q, F_3^Q...F_{t-1}^Q)$, using an LSTM akin to the fact encoder described above. We pass these embeddings and F_t^Q computed by the Fact Encoder through a fully connected layer, generating attention weights which are used to attend over H_t^Q, producing the history embedding $S_t^Q \in R^{512}$. Notice that this results in a two-level hierarchical encoding of the dialog $(q_t, a_t) \rightarrow F_t^Q$ and $(F_1^Q, F_2^Q, F_3^Q...F_t^Q) \rightarrow S_t^Q$.

3. **Caption Encoder:** This is a unidirectional LSTM which is given the image caption c as input. The LSTM generates a caption embedding $C^Q \in R^{512}$.

4. **Feature Regression Network:** $\{F_t^Q, S_t^Q, C^Q\}$ are concatenated to produce an embedding E_t^Q. This is passed through 2 fully connected layers with dropout to produce $\hat{y}_t$ from the current encoded state $\hat{y}_t = f(S_t^Q)$.

5. **Question Decoder:** The hidden state of this LSTM is initialized with the hidden state of the fact encoder. E_t^Q is passed through a fully connected layer to generate e_t^Q, which is used to update the hidden state of the LSTM of the question decoder. The question q_t is then generated by sequentially sampling words (either via teacher forcing during supervised pretraining or via autoregressive generation during RL and evaluation).

Note that we use a dropout of 0.5 in all the LSTMs during training. All LSTM hidden layers sizes are 512, and the image embedding size is 4096. The input word embedding size is 300.

A.2 A-Bot

As discussed in Section 4.2, the A-bot architecture has 3 components.

1. **Question Encoder:** The question encoder is a unidirectional LSTM which is given the current question q_t generated by the Q-Bot as input. The LSTM generates a question embedding $Q_t^A \in R^{512}$.

2. **State/History/Image Encoder:** The encoder first computes the fact embeddings $H_t^A = (F_1^A, F_2^A, F_3^A...F_{t-1}^A)$, using an LSTM akin to the fact encoder described above. By passing these embeddings and the Q_t^A computed by the Question Encoder through a fully connected layer, attention weights are generated which are used to attend over H_t^A, producing the history embedding $S_t^A \in R^{512}$. Notice that this results in a two-level hierarchical encoding of the dialog $(q_t, a_t) \rightarrow F_t^A$ and $(F_1^A, F_2^A, F_3^A...F_t^A) \rightarrow S_t^A$. $\{F_t^A, S_t^A, y_{gt}\}$ are then concatenated to produce an embedding E_t^A.

3. **Answer Decoder:** The hidden state of this LSTM is initialized with the hidden state of the question encoder. E_t^A is passed through a fully connected layer to generate e_t^A, which is used to update the hidden state of the LSTM of the answer decoder. The answer a_t is then generated by sequentially sampling words (either via teacher forcing during supervised pretraining or via autoregressive generation during RL and evaluation).

Note that we use a dropout of 0.5 in all the LSTMs during training. All LSTM hidden layers sizes are 512, and the image embedding size is 4096. The input word embedding size is 300.

B Training Details

B.1 Supervised Pre-Training

Both Q-Bot and A-Bot are trained in isolation for 15 epochs, via supervision from the VisDial dataset.

B.1.1 Q-Bot:

At time step t, the Q-Bot's fact encoder is fed with the ground-truth QA pair for $t-1$, the state/history encoder is fed with all the ground-truth QA pairs up to $t-1$ and the caption encoder is given the true image caption c_{gt} as input. These encoders

then generate their respective embeddings which are fed into the feature regression network and the question decoder to produce $\hat{y}_t$ and update the hidden state of the question decoder respectively. The Q-Bot is then trained by maximizing the likelihood $p(q_{gt}|E_t^Q)$ of the training data q_{gt}, computed using the softmax probabilities given by the question decoder. Simultaneously, the Mean Squared Error (MSE) loss between the predicted image embedding and ground truth is also minimized. Effectively, the loss

$$\mathcal{L}_Q = -\sum \log p(q_{gt}|E_t^Q) + (\hat{y}_t - y_{gt})^2$$

is minimized.

B.1.2 A-Bot:

At time step t, the A-Bot's question encoder is fed with the ground-truth question for t, the state/history/image encoder is fed with all the ground-truth QA pairs up to $t-1$ and the image I. These encoders then generate their respective embeddings which are fed into the answer decoder to produce a_t. The A-Bot is trained by maximizing the likelihood $p(a_{gt}|E_t^A)$ of the training data a_{gt}, computed using the softmax probabilities given by the answer decoder. Effectively, the loss

$$\mathcal{L}_A = -\sum \log p(a_{gt}|E_t^A)$$

is minimized.

B.2 Reinforcement Learning

The Q-Bot is given only the caption c_{gt} and the A-Bot is given only the image I and caption c_{gt} as inputs.

B.2.1 Q-Bot:

At time step t, the Q-Bot's fact encoder is fed with the generated QA pair for $t-1$, the state/history encoder is fed with all the generated QA pairs up to $t-1$ and the caption encoder is given the true image caption c_{gt} as input. These encoders then generate their respective embeddings which are fed into the feature regression network and the question decoder to produce $\hat{y}_t$ and q_t respectively. The change in distance between $\hat{y}_t$ and y_{gt} due to the current QA-pair is given as a reward to Q-Bot (Eqn. 1), which it uses to train itself via REINFORCE. Simultaneously, the Mean Squared Error (MSE) loss between the predicted image embedding and ground truth is also minimized via supervision. Effectively, the loss

$$\mathcal{L}_Q = -\sum G_t \log p(q_t|E_t^Q) + (\hat{y}_t - y_{gt})^2$$

is minimized, where $G_t = \sum_{k=0}^{10-t} \gamma^k r_{t+k+1}$ indicates the Monte-Carlo return at step t, and γ is a discount factor equal to 0.99.

B.2.2 A-Bot:

At time step t, the A-Bot's question encoder is fed with the generated question q_t, the state/history/image encoder is fed with all the generated QA pairs up to $t-1$ and the image I. These encoders then generate their respective embeddings which are fed into the answer decoder to produce a_t. The A-Bot also receives the same reward as the Q-Bot, and trains itself via REINFORCE. Effectively, the loss

$$\mathcal{L}_A = -\sum G_t \log p(a_t|E_t^A)$$

is minimized, where $G_t = \sum_{k=0}^{10-t} \gamma^k r_{t+k+1}$ indicates the Monte-Carlo return at step t, and γ is a discount factor equal to 0.99.

11