The Devil is in the Details: A Magnifying Glass for the GuessWhich Visual Dialogue Game

Alberto Testoni
University of Trento
alberto.testoni@unitn.it

Ravi Shekhar
Queen Mary University of London
r.shekhar@qmul.ac.uk

Raquel Fernández
University of Amsterdam
raquel.fernandez@uva.nl

Raffaella Bernardi
University of Trento
raffaella.bernardi@unitn.it

Abstract

Grounded conversational agents are a fascinating research line on which important progress has been made lately thanks to the development of neural network models and to the release of visual dialogue datasets. The latter have been used to set visual dialogue games which are an interesting test bed to evaluate conversational agents. Researchers' attention is on building models of increasing complexity, trained with computationally costly machine learning paradigms that lead to higher task success scores. In this paper, we take a step back: We use a rather simple neural network architecture and we scrutinize the GuessWhich task, the dataset, and the quality of the generated dialogues. We show that our simple Questioner agent reaches state-of-the art performance, that the evaluation metric commonly used is too coarse to compare different models, and that high task success does not correspond to high quality of the dialogues. Our work shows the importance of running detailed analyses of the results to spot possible models' weaknesses rather than aiming to outperform state-of-the-art scores.

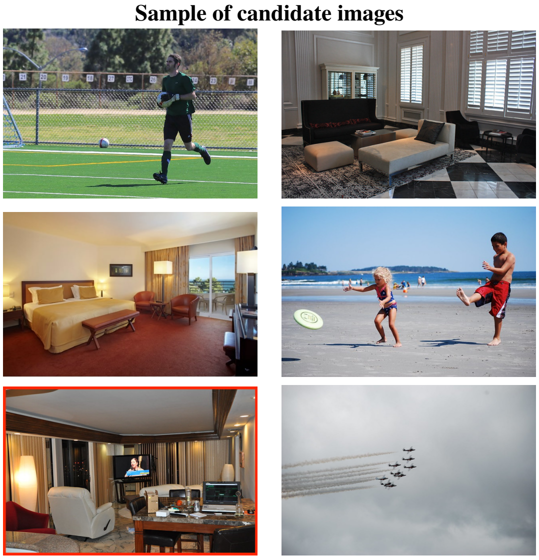
1 Introduction

The development of conversational agents that ground language into visual information is a challenging problem that requires the integration of dialogue management skills with multimodal understanding. Recently, visual dialogue settings have entered the scene of the Machine Learning and Computer Vision communities thanks to the construction of visually-grounded human-human dialogue datasets (Mostafazadeh et al., 2017; Das et al., 2017a; de Vries et al., 2017) against which neural network models have been challenged. Artificial agents have been developed to learn either to ask or answer questions. Most of the work has focused on developing better Answerer agents, with a few exceptions (e.g., Manuvinakurike et al., 2017; Zhang et al., 2018; Jiaping et al., 2018; Sang-Woo et al., 2019; Shekhar et al., 2019). Interesting and efficient machine learning methods (such as hierarchical co-attentions and adversarial learning) have been put at work to improve the Answerer agent (Lu et al., 2017b,a; S. and D., 2018; Kottur et al., 2018; Wu et al., 2018; Yang et al., 2019; Gan et al., 2019). Also when work has been proposed to highlight weaknessed of the available datasets, this has been done from the perspective of the Answerer (Massiceti et al., 2019). Much less is known about the Questioner agent, on which our work focuses.

The Questioner is evaluated through visually-grounded dialogue games like `GuessWhat?!` and `GuessWhich` introduced by de Vries et al. (2017) and Das et al. (2017b), respectively.[1] The two games share the idea of having two agents, a Questioner and an Answerer, playing together so that the Questioner, by asking questions to the Answerer, at the end of the game can make its guess on what is the object or which is the image they have been speaking about; however, the two games differ in many respects. Crucially in `GuessWhich` the Questioner sees a description (i.e., a caption) of the target image it will have to guess at the end of the game, but does not see any of the candidate images among which it has to select the target one (see Figure 1 for an example). Most, if not all, the work proposed for these two games heavily relies on Reinforcement Learning (RL).

The purpose of this work is to dive into the `GuessWhich` task and dataset through a simple Questioner model trained in a supervised setting, with a standard encoder-decoder architecture. The model learns to process the image caption and the dialogue history (the sequence of question-answer

[1] The name `GuessWhich` has been used only lately by Chattopadhyay et al. (2017) to evaluate the Answerer agent playing the game with a Human. We take the liberty to use it for the game when played by two agents.



Caption: *A room with a couch, tv monitor and a table*

Questioner	Answerer
1. what kind of room is it?	family room home office
2. is tv on?	yes
3. can you see what is playing on tv?	woman who is singing or giving a speech close caption on bottom
4. is there anything on table?	yes all kinds of stuff
5. are there any people in room?	no there aren't
6. are there any windows?	sliding glass door i believe
7. are there any lamps or lights?	yes modern floor lamp to left of chair
8. is there carpet on floor?	yes there is
9. what color is carpet?	off white with darker design running through
10. what kind of chair is visible?	white leather reclining chair

Figure 1: GuessWhich: two Bots are given a caption describing an image that one of the two bots (the answerer) sees while the other (the questioner) does not see. The Questioner has to ask 10 questions about the image and then select among about 10K candidates the image they have been speaking about. The dialogues given as example were generated by AMT workers, who were asked to chit-chat about the image, without having to select the target image at the end. The target image is the one on the left corner on the bottom, marked by the red box.

pairs), to generate questions, and to retrieve the target image at the end of the game by ranking the candidate images. We show that a simple model like ours outperforms state-of-the-art (SoA) models based on RL. Most importantly, by scrutinizing the model, we show that the SoA result obtained hides important weaknesses of the model and of the dataset:

- The question generator plays a rather minor role on task-success performance.
- The dialogues do not help much to guess the image, in the test phase. During training, they play the role of a *language incubator*, i.e., they help enrich the linguistic skills of the model, but the most informative linguistic input to guess the image is its caption.
- The distribution of game difficulty in the dataset is rather skewed: our simple model performs very well on half of the games, while half of the games appear to have issues that make them intrinsically difficult.

2 Related Work

Reinforcement Learning (RL) has become the default paradigm in visually-grounded dialogue. Strub et al. (2018) and Das et al. (2017b) show that RL improves the Questioner's task success with respect to supervised learning (SL) in both `GuessWhat?!` and `GuessWhich`. Two crucial components of the Questioner in visual dialogue guessing games are the question generator and the guesser. Shekhar et al. (2019) show that by training these two components jointly good performance can be achieved, and that a level of task success comparable to that attained by RL-based models can be reached by training the two modules cooperatively (i.e., with generated dialogues). Furthermore, Shekhar et al. (2019) show the linguistic poverty of the dialogues generated with RL methods, highlighting the importance of going beyond task success in evaluating visually-grounded Questioner agents. Inspired by this work, we study how far a simple model can go within the `GuessWhich` game and how the dialogue history is exploited in such a game.

Jiaping et al. (2018) propose a Questioner model based on hierarchical RL which, besides using RL to play the `GuessWhich` game, learns to decide when to stop asking questions and guess the image. In their approach, questions are retrieved (rather than generated) and the model is trained and evaluated on 20 pre-selected candidate images (instead of the full list of around 10K candidates as in the original game). A decision-making module has been introduced also by Shekhar et al. (2018), who train a discriminative model to play the `GuessWhat?!` game end-to-end without RL. In `GuessWhat?!`, the Questioner model has to identify a target object among 20 candidate objects within an image. Thanks to the decider module, SoA results are achieved with shorter dialogues.

In the original GuessWhich game, the image has to be guessed among a very high number of candidates (∼10k); moreover, neither the target nor the other candidate images are seen during the dialogue. Hence the role of the decider module is vanished in such a setting, since the agent will never be sure to have gathered enough information to distinguish the target from the other images. As we focus on the original GuessWhich game, we do not include a decision-making module in our Questioner model. The number of questions is set to 10, as with the human players (see the next section).

Finally, a novel model is proposed by Sang-Woo et al. (2019), where the Questioner exploits a probabilistic calculus to select the question that brings about the most information gain. Their code has just been released. Hence, we leave for the future a thorough comparison with this approach.

3 Task and Dataset

We evaluate our model on the GuessWhich game proposed by Das et al. (2017b), which is based on the Visual Dialogue (VisDial) dataset by Das et al. (2017a).[2]

VisDial is the dataset used to play the GuessWhich game. It consists of 68K images from MS-COCO (Lin et al., 2014) of which 50,729 and 7663 are used for the training and validation set, respectively, and 9628 are used for the test set. There is no image overlap across the three sets. Each image is paired with one dialogue. The dialogues have been collected through Amazon Mechanical Turk (AMT) by asking subjects to chat in real-time about an image. The two AMT workers were assigned distinct roles: the questioner, who does not see the image but sees an MS-COCO caption of it, has to imagine the scene and ask questions about it; the answerer, who sees the image and the caption, has to answer the other player's questions. The workers are allowed to end the chat after 10 rounds of question-answer pairs. An example of a dialogue by AMT workers is shown in Figure 1.

GuessWhich is a two-player game proposed by Das et al. (2017b). Two agents, Q-bot and A-Bot, have to play the role of the Questioner and the Answerer AMT workers in VisDial, but at the end of the dialogue the Qbot has to guess the target image among a set of candidates (this task-oriented aspect was not present in the human data collection). The authors have released two versions of the test set: one with the original MS-COCO ground-truth captions and one with captions automatically generated with Neuraltalk (Karpathy and Fei-Fei, 2015) using the implementation by Vinyals and Le (2015). Usually, models are trained with the ground-truth captions and evaluated using the generated ones to check their robustness.

4 Models

We focus on developing a model of the Questioner agent. As the Answerer, we use the A-bot model by Das et al. (2017b) described below.

4.1 The Answerer Model

The A-Bot by Das et al. (2017b) is based on a Hierarchical Recurrent Encoder-Decoder neural network. It consists of three 2-layered LSTM encoders with 512-d hidden states and one LSTM decoder: A *question encoder* encodes the question asked by the Q-Bot; a *history encoder* takes, at each turn t, (i) the encoded question Q_t, (ii) the VGG image features (recall that the Answerer does see the image, unlike the Questioner), and (iii) the previous question-answer pair encodings to produce a state-embedding of the question being asked that is grounded on the image and contextualized over the dialogue history; an *answer decoder* takes the state encoding of the history encoder and generates an answer by sampling words; a *fact encoder* encodes the question-answer pairs.

The VGG features are obtained from a CNN pretrained on ImageNet (Russakovsky et al., 2015). The vocabulary contains all tokens that occur at least 5 times in the training set; its size is 7,826 tokens. The model is trained with a cross-entropy loss. We use the code released in the authors' Github page.

4.2 State of the Art Questioner Models

The Q-Bot by Das et al. (2017b) has a similar structure to the A-bot described above and shares its vocabulary, but it does not receive the image features as input. The goal of Q-Bot is to generate a question based on the caption and the dialogue history (the sequence of previous question-answer pairs). To this end, an encoder receives first the caption and then the question-answer pairs sequential-

[2] We use the version v0.5 available from the authors' github at https://github.com/batra-mlp-lab/visdial-rl.

ly; it outputs a state-embedding at t that is jointly used by the decoder (an LSTM which learns to generate the next question) and by a Feature Regression Network (FRN, a fully connected layer which learns to approximate the visual vector of the target image). The decoder and the FRN are updated at every turn.

In the supervised learning (SL) phase, the two agents (A-Bot and Q-Bot) are separately trained under a Maximum Likelihood Estimation objective on the train set of VisDial human-human dialogues for 60 epochs. The FRN of the Q-Bot is trained to regress to the true image representation at each turn using Mean Square Error, i.e. l_2 loss. We will refer to this setting as Q-Bot-SL.

In the Reinforcement Learning (RL) phase, the Q-Bot and A-Bot are initialized by the models trained with SL for 15 epochs and then are fine-tuned with RL gradually by continuing SL for the first k rounds, and with RL for the $10 - k$ rounds, and annealing down k by 1 at every epoch. The authors have released the versions in which the model is trained with RL for 10 and 20 epochs. The reward is given to the two bots at each turn jointly. It is based on the change in distance (l_2) between the image representation produced by the FRN of Q-Bot and the true image vector before and after a round of dialogue. The total reward is a function only of the initial and final states. We will refer to this setting as Q-Bot-RL.

Recently, Sang-Woo et al. (2019) have proposed an interesting new model, AQM+, within the Answerer in Questioner's Mind (AQM) framework introduced by Lee et al. (2017). Their Questioner asks questions based on an approximated probabilistic model of the Answerer, generating the question that gives the maximum information gain. The authors evaluate two versions of their model corresponding to the Q-Bot-SL and Q-Bot-RL settings described above: the two agents are trained (a) independently using human data (hence, AQM+/indA) or (b) together using the generated data (AQM+/depA).

4.3 Our Questioner Model

The architecture of our model is similar to the Q-Bot model of Das et al. (2017b) with two important differences: (i) the Encoder receives the caption at each turn, as it happens with humans who can reread the caption each time they ask a new question, and (ii) in the training phase, the image regression module "sees" the visual vector of the target image only once, at the end of the game (i.e., as is the case for the human participants, there is no direct visual feedback during the dialogue).

As illustrated in Figure 2, in our model the Encoder receives two linguistic features: one for the caption and one for the dialogue. These features are obtained through two independent LSTM networks (Cap-LSTM and QA-LSTM) whose hidden states of 1024 dimensions are scaled through two linear layers to get linguistic features of 512-d. These two representations are passed to the Encoder: they are concatenated and scaled through a linear layer with a *tanh* activation function. The final layer (viz. the dialogue state) is given as input to both the question decoder (QGen) and the Guesser module. QGen employs an LSTM network to generate the token sequence for each question. The Guesser module acts as a feature regression network (FRN): it takes as input the dialogue hidden state produced by the Encoder, and passes it through two linear layers with a ReLU activation function on the first layer. The final representation is a 4096-d vector which corresponds to the fc7 VGG representation of the target image. In contrast to the FRN by Das et al. (2017b), as mentioned above, our Guesser "sees" the ground-truth image only at the end of the game.

We use the same vocabulary as the A-Bot model. We apply the supervised training paradigm of Das et al. (2017b) and refer to our simple Questioner model as ReCap.

5 Experiment and Results

In this section, we present our experimental setup and report the results obtained, comparing them to the state of the art. We also analyse the role of the caption and the dialogue, as well the joint training regime on the performance of the model.

5.1 Evaluation Metrics and Implementation

Following Das et al. (2017b), we report the Mean Percentile Rank (MPR) of the target image, which is computed from the mean rank position of the target image among all the candidates. An MPR of e.g., 95% means that, on average, the target image is closer to the one chosen by the model than the 95% of the candidate images. Hence, in the VisDial test set with 9628 candidates, 95% MPR corresponds to a mean rank of 481.4, and a difference of +/– 1% MPR corresponds to –/+ 96.28 mean rank, which is a substantial difference. The chance level is 50.00

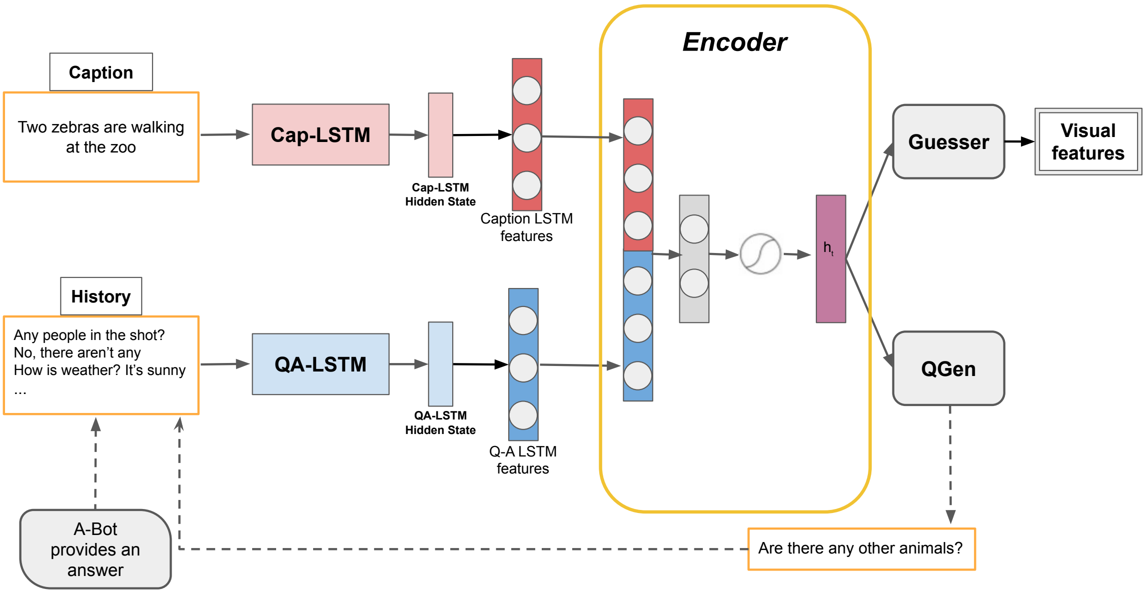


Figure 2: The ReCap questioner model: A simple encoder-decoder architecture that builds a hidden state of the dialogue by combining the representation of the caption and the dialogue; it rereads the caption at each turn while processing the dialogue history incrementally. The hidden state is used by the decoder (Question Generator) to generate the follow-up question at each turn, and by the Guesser to select the target image at the end of the game.

MPR, viz., 4814 mean rank position.

Our ReCap model has been trained for 41 epochs. Like Das et al. (2017b), our QGen and Guesser are trained jointly. However, following Shekhar et al. (2019), we use a modulo-n training regime, where n indicates after how many epochs of QGen training the Guesser is updated – we use $n = 5$. For the Q-Bot by Das et al. (2017b), we report the results we have obtained using the code released by the authors since they are higher than those reported in their paper.[3]

5.2 Comparison with SoA Models

Following Das et al. (2017b); Sang-Woo et al. (2019), we evaluate the models on the version of the test set containing captions generated with NeuralTalk2. As already shown by these authors, at test time, SoA models achieve rather good performance at round 0, i.e., just being exposed to the caption, without the dialogue history. For instance, the Q-Bot-SL trained on both captions and dialogues, when tested only on the caption achieves 89.11 MPR; in other words, it obtains just 2.08% less than what the same model achieves with the full 10-round dialogue. As we can see in Table 1, the same holds for all the models we consider.

	MPR@0	MPR@10
Chance	50.00	50.00
Q-Bot-SL	89.11	91.19
Q-Bot-RL	95.72	94.19
AQM+/indA	88.50	94.64
AQM+/depA	88.50	97.45
ReCap	89.38	95.54

Table 1: Models tested with captions generated with NeuralTalk2. We evaluate the Mean Percentile Rank (MPR) of the models when receiving only the caption (at round 0) or the full dialogue (round 10). The results of the AQM model are from Sang-Woo et al. (2019).

Two things stand out regarding the performance of our model ReCap: First, although it is simpler, it obtains results 4.35% higher than Q-Bot-SL and comparable to Q-Bot-RL (ReCap +1.35%) as well as to the "supervised" version of the AQM model (ReCap +0.90%). Its performance is only lower than the more complex version of AQM (– 1.91%). Second, our model appears to be able to exploit the dialogue beyond the caption to a larger degree than Q-Bot-SL and Q-Bot-RL, as evidenced by the larger difference between the results at round 0 and round 10.

5.3 Role of the Caption and the Dialogue

Given the results by Das et al. (2017b); Sang-Woo et al. (2019) with respect to the high performance obtained by the model at round 0 with just the

[3] In the authors' github, there are various versions of the code: the QBot-RL model trained with 10 vs. 20 epochs, starting from the pre-trained Q-Bot-SL, and with and without optimizing the delta parameter. We use the code without the delta parameter, since it is the one explained in the paper, and with 20 epochs since it gives better results than the other one.

		GEN	GT
ReCap	MPR@0	89.38	87.95
	MPR@10	95.54	95.65
Q-Bot-SL	MPR@0	89.11	87.53
	MPR@10	91.19	89.00
Q-Bot-RL	MPR@0	95.72	94.84
	MPR@10	95.43	94.19

	MPR@10
Guesser caption	49.99
Guesser dialogue	49.99
Guesser caption + dialogue	94.92
Guesser+QGen	94.84
ReCap	95.65
Guesser-USE caption	96.90

Table 2: **Left**: Comparison of models performance when tested on generated (GEN) vs. ground truth (GT) captions **Right**: Ablation study of ReCap reporting MPR: We evaluate the Guesser when trained by receiving only the GT caption (Guesser caption); only the GT dialogues (Guesser dialogue) or both the GT caption and the GT dialogues (Guesser caption + dialogue). Furthermore, we report the results of QGen and Guesser trained separately (Guesser + QGen). Finally, Guesser-USE caption shows the MPR obtained by the Gusser when using pre-trained linguistic features.

caption, we aim to better understand the role of the captions and the dialogues in GuessWhich.

First of all, we check how the models behave on the ground truth captions (GT) of MS-COCO. As we can see from Table 2 (left), having the generated captions instead of the GT ones, facilitates the task (all models experience a gain in performance of around 1 to 2% MPR with GEN). However, at round 10, our model is somewhat more stable: it is less affected by the use of GT vs. generated captions than the other two versions of Q-Bot.

Secondly, we check whether the lack of improvement through the dialogue rounds is due to the quality of the dialogues. Therefore, we run the evaluation on the GT dialogues. Figure 3 reports the performance of our ReCap model when tested with the GT dialogues at each question-answer round, and compares it with the performances obtained by ReCap and the two Q-Bot models with the generated dialogues. As we can see, using the generated or GT dialogues does not affect ReCap's performance very much: Also with human dialogues, after round 3 the performance does not increase significantly. Of course, these results do not say anything about the quality of the dialogues generated by the models, but they show that the per-round pattern common to all the models is not due to the linguistic quality of the dialogues.

Finally, we evaluate our Guesser trained and tested when receiving as input only the caption (Guesser caption), only the GT dialogues (Guesser dialogue) or both (Guesser caption and dialogue). Interestingly, as we can see from Table 2 (right), training the model only on the caption or only on the dialogue does not provide the Guesser with enough information to perform the task: it stays at chance level. Instead, training it with both types of linguistic input doubles its performance (from 49.99 to 94.92). Based on these findings, we check also how the Guesser, trained and tested only on the caption, performs when the caption embedding is obtained using pre-trained and frozen linguistic features from the Universal Sentence Encoder (USE) by Cer et al. (2018) (Guesser-USE caption). This model reaches 96.90 MPR. This shows that in ReCap the caption and the dialogue play a complementary role: the caption provides a good description of the image but it is not enough by itself to train the model lexical knowledge; whereas the dialogues improve the linguistic knowledge of the encoder but do not provide a self-contained description of the image since they were produced as a follow-up to the image caption.

5.4 Role of the Joint Learning

By comparing the results in Table 1 and Table 2, we can see that QGen plays a rather minor role: already the Guesser alone reaches 94.92 MPR. Below we verify whether the multi-task setting in which the Guesser and QGen modules are trained improves the task success. Shekhar et al. (2019) show that the accuracy of a model that jointly learns to ask a question and guess the target object in the GuessWhat?! game obtains a 9% increase over its counterpart in which the two modules are trained separately. We check whether this result holds for the GuessWhich game too and compare ReCap with its counterpart with the two modules trained independently (Guesser+QGen). As we can see from Table 2, the joint training brings an increase of +0.81% MPR, viz. a lower increase than the one found in the GuessWhat?! game. We conjecture that this difference is due to the fact that the Guesser in GuessWhich does not have access to the distractor images during the dialogue, viz., the

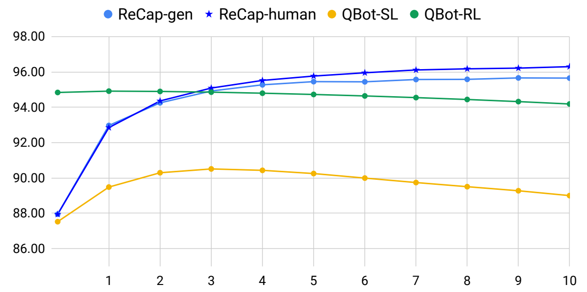


Figure 3: MPR distribution per dialogue round: comparison of ReCap model tested on human dialogues vs. ReCap and QBot models tested on generated dialogues.

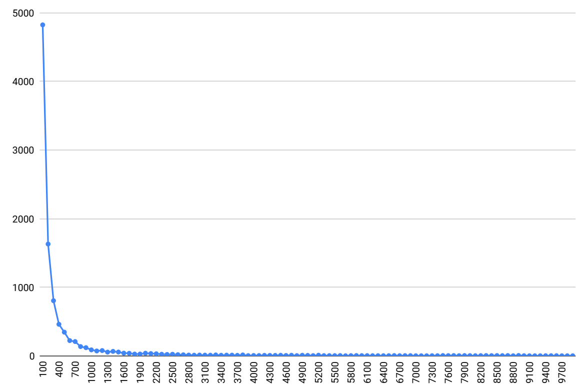


Figure 4: Distribution of rank assigned to the target image by ReCap tested on human dialogues. Each column aggregates 300 ranks.

candidate images that it has to learn to distinguish from the target image.

6 Analysis

To better understand why the dialogues do not help to rank the target image higher, below we further analyse the dataset.

Analysis of the Ranking For each of the 9628 images in the test set, we look into the ranks chosen by the ReCap model tested on the human dialogues. As shown Figure 4, the distribution is very skewed. On the one hand, in 126 games the target image has been ranked below chance level (below rank 4814); these images effect the MPR quite negatively. On the other hand, half of the games played by our simple model have a rank lower than 100, which means approximately 99 percentile or higher. Of these, 1032 are ranked above the 10th position.

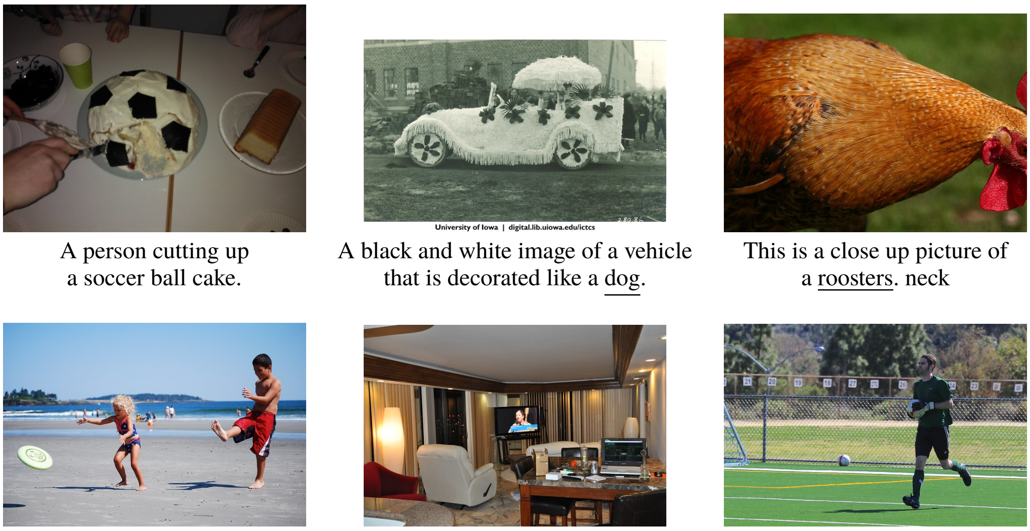
Qualitative analysis of the 126 instances ranked below chance level has revealed that they are mostly cases of dialogues about images whose objects are hard to recognize, where the caption contains wrong information or unknown words (see examples in Figure 5.) Interestingly, Gusser-USE caption has failed to rank high only 60 of these 126 outliers. For instance, the example in Figure 5 (up right) with the unknown word "roosters" is ranked at position 1082 by Guesser-USE caption and at 7006 by ReCap. As for the 1032 games with highly ranked target images, they concern images where the main objects are easily identifiable and are mentioned in the caption.

Analysis of the Visual Space To further understand the MPR results obtained by the models, we have carried out an analysis of the visual space of the candidate images. To check how the images in the high vs. low position in the rank differ, we have looked into their neighbourhood in the semantic space. We see that the highly ranked images have a denser neighbourhood that the ones ranked low, where density is defined as mean cosine distance between the image visual vector and its 20 closest neighbours. There is a 0.61 Spearman correlation, with p-value < 0.05, between the rank of the retrieved image and the density of the neighbourhood.

Analysis of the Dialogues Following Shekhar et al. (2019), we look into the quality of the dialogues generated by the models by computing lexical diversity, measured as type/token ratio over all games; question diversity, measured as the percentage of unique questions over all games, and the percentage of games with at least one question repeated verbatim within the dialogue. Table 3 reports the statistics for our ReCap model and Q-bot. As we can see, ReCap produces a much richer and less repetitive output than both versions of Q-Bot. In particular, it has a more diverse vocabulary, generates more unique questions, and repeats questions within the same dialogue at a much lower rate. An illustration of these difference in the dialogue quality is provided by the example in Table 4.

	Lexical diversity	Question diversity	% Games with repeated Q's
Q-bot-SL	0.11	1.66	100
Q-bot-RL	0.05	0.35	100
ReCap	0.45	31.25	41.66
Humans	1.85	46.70	1.06

Table 3: Linguistic analysis of the generated dialogues. For lexical and question diversity higher scores are better, while lower scores are better for percentage of games with repeated questions.



A person cutting up a soccer ball cake.

A black and white image of a vehicle that is decorated like a dog.

This is a close up picture of a roosters. neck

Two children are playing on the beach with a frisbee.

A room with a couch, tv monitor and a table.

A man that is on a soccer field with a ball.

Figure 5: **Top:** Images ranked lower than 8000 by ReCap: the objects are difficult to identified (left), the caption contains wrong information (middle) or unknown words (right); the wrong and the unknown words are underlined. **Bottom:** Images ranked higher than 10 by ReCap: the captions mention the main objects which are easily identifiable.

ReCap		QBot-SL		QBot-RL	
1. is tv flat screen?	no, it is small screen of some sort	1. is tv on?	no	1. any people?	no
2. does table have wood table?	yes, it does	2. is there couch?	yes	2. is it sunny?	yes
3. any people?	no people	3. what color is couch?	brown	3. is it sunny?	yes
4. is tv on?	no	4. is there a table?	yes	4. is it sunny?	yes
5. what color is couch?	brown	5. is there a table?	yes	5. is it sunny?	yes
6. are there any boots?	no, there aren't any	6. is there a table?	yes	6. is it sunny?	yes
7. can you see wintows?	no	7. is there a table?	yes	7. is it sunny?	yes
8. does table have wood on table?	no it doesn't	8. what color is table?	brown	8. is it sunny?	yes
9. what color are walls?	white	9. is there table?	yes	9. is it sunny?	yes
10. does table look clean?	not really	10. what color is table?	brown	10. is it sunny?	yes

Table 4: Examples of dialogues produced by the ReCap and Q-Bot models for the game about the image in Figure 5 (bottom, middle) which has been highly ranked by ReCap.

7 Conclusion

We have presented a simple model of the GuessWhich Questioner player. We have shown that it achieves SoA task-success scores. We have used this model as a magnifying glass to scrutinize the GuessWhich task and dataset aiming to further understand the model's results and, by so doing, to shed light on the task, the dataset, and the evaluation metric.

Our in-depth analysis shows that the dialogues play the role of a language incubator for the agent, i.e., they simply enrich its linguistic skills, and do not really help in guessing the target image. Furthermore, the difficulty distribution of the GuessWhich datapoints seems to be rather skewed: on the one hand, our model performs very well on half of the games; on the other hand, there are outliers which have intrinsic difficulty and have a high impact on the final score. All this shows that the metric used in previous work to evaluate the models is way too coarse and obscures important aspects. Finally, we have shown that the linguistic quality of the dialogues produced by our simple model is substantially higher than that of the dialogues generated by SoA models.

References

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. 2018. Universal sentence encoder. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (System Demonstrations)*.

Prithvijit Chattopadhyay, Deshraj Yadav, Viraj Prabhu, Arjun Chandrashekharan, Abhishek Das, Stefan Lee, Dhruv Batra, and Devi Parikh. 2017. Evaluating visual conversational agents via cooperative human-ai games. In *The fifth AAAI Conference on Human Computation and Crowdsourcing (HCOMP)*.

Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José M.F. Moura, Devi Parikh, and Dhruv Batra. 2017a. Visual Dialog. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Abhishek Das, Satwik Kottur, José M.F. Moura, Stefan Lee, and Dhruv Batra. 2017b. Learning cooperative visual dialog agents with deep reinforcement learning. In *International Conference on Computer Vision (ICCV)*.

Zhe Gan, Yu Cheng, Ahmed EI Kholy, Linjie Li, Jingjing Liu, and Jianfeng Gao. 2019. Multi-step reasoning via recurrent dual attention for visual dialog. Https://arxiv.org/abs/1902.00579.

Zhang Jiaping, Zhao Tiancheng, and Yu Zhou. 2018. Multimodal hierarchical reinforcement learning policy for task-oriented visual dialog. In *Proceeding of the SigDial Conference*, pages 140–150. Association for Computational Linguistics.

Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3128–3137.

Satwik Kottur, José M. F. Moura, Devi Parikh, Dhruv Batra, and Marcus Rohrbach. 2018. Visual coreference resolution in visual dialog using neural module networks. In *Proceedings of ECCV 2018*.

Sang-Woo Lee, Yujung Heo, and Byoung-Tak Zhang. 2017. Answerer in questioner's mind for goal-oriented visual dialogue. In *NIPS Workshop on Visually-Grounded Interaction and Language (ViGIL)*.

T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, Dollar, P., and C. L. Zitnick. 2014. Microsoft COCO: Common objects in context. In *Proceedings of ECCV (European Conference on Computer Vision)*.

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

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2017b. Hierarchical question-image co-attention for visual question answering. In *Conference on Neural Information Processing Systems (NIPS)*.

Ramesh Manuvinakurike, David DeVault, and Kallirroi Georgila. 2017. Using reinforcement learning to model incrementality in a fast-paced dialogue game. In *Proceedings of the SIGDIAL 2017 Conference*.

Daniela Massiceti, Puneet K. Dokania, N. Siddharth, and Philip H.S. Torr. 2019. Visual dialogue without vision or dialogue. In *NeurIPS Workshop on Critiquing and Correcting Trends in Machine Learning*.

Nasrin Mostafazadeh, Chris Brockett, Bill Dolan, Michel Galley, Jianfeng Gao, Georgios P. Spithourakis, and Lucy Vanderwende. 2017. Image-grounded conversations: Multimodal context for natural question and response generation. In *Proceedings of the The 8th International Joint Conference on Natural Language Processing*.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. 2015. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252.

Zarrieß S. and Schlangen D. 2018. Decoding strategies for neural referring expression generation. In *Proceedings of INLG 2018*.

Lee Sang-Woo, Gao Tong, Yang Sohee, Yao Jaejun, and Ha Jung-Woo. 2019. Large-scale answerer in questioner's mind for visual dialog question generation. In *ICLR*.

Ravi Shekhar, Tim Baumgärtner, Aashish Venkatesh, Elia Bruni, Raffaella Bernardi, and Raquel Fernández. 2018. Ask no more: Deciding when to guess in referential visual dialogue. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING)*, pages 1218–1233.

Ravi Shekhar, Aashish Venkatesh, Tim Baumgärtner, Elia Bruni, Barbara Plank, Raffaella Bernardi, and Raquel Fernández. 2019. Beyond task success: A closer look at jointly learning to see, ask, and guesswhat. In *NAACL*.

Florian Strub, Mathieu Seurin, Ethan Perez, Harm de Vries, Jérémie Mary, Philippe Preux, Aaron Courville, and Olivier Pietquin. 2018. Visual reasoning with multi-hop feature modulation. In *Proceedings of ECCV*.

Oriol Vinyals and Quoc V. Le. 2015. A neural conversational model. In *Proceedings of the 31st International Conference on Machine Learning*.

Harm de Vries, Florian Strub, Sarath Chandar, Olivier Pietquin, Hugo Larochelle, and Aaron C. Courville. 2017. Guesswhat?! Visual object discovery through multi-modal dialogue. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Qi Wu, Peng Wang, Chunhua Shen, Ian Reid, and Anton van den Hengel. 2018. Are you talking to me? reasoned visual dialog generation through adversarial learning. In *In Proceedings of CVPR 2018*.

Tianhao Yang, Zheng-Jun Zha, and Hanwang Zhang. 2019. Making history matter: History-advantage sequence training for visual dialog. Https://arxiv.org/abs/1902.09326.

Junjie Zhang, Qi Wu, Chunhua Shen, Jian Zhang, Jianfeng Lu, and Anton van den Hengel. 2018. Goal-oriented visual question generation via intermediate rewards. In *Proceedings of the European Conference of Computer Vision (ECCV)*.