Recurrent Attention Network with Reinforced Generator for Visual Dialog

HEHE FAN[*], Center for Artificial Intelligence, University of Technology Sydney, Australia and Baidu Research, China
LINCHAO ZHU, Center for Artificial Intelligence, University of Technology Sydney, Australia
YI YANG, Center for Artificial Intelligence, University of Technology Sydney, Australia
FEI WU, College of Computer Science, Zhejiang University, China

In Visual Dialog, an agent has to parse temporal context in the dialog history and spatial context in the image to hold a meaningful dialog with humans. For example, to answer "what is the man on her left wearing?", the agent needs to: 1) analyze the temporal context in the dialog history to infer who is being referred to as "her"; 2) parse the image to attend "her"; 3) uncover the spatial context to shift the attention to "her left" and check the apparel of the man. In this paper, we use a dialog network to memorize the temporal context and an attention processor to parse the spatial context. Since the question and the image are usually very complex, which makes it difficult for the question to be grounded with a single glimpse, the attention processor attends to the image multiple times to better collect visual information. In the Visual Dialog task, the generative decoder (G) is trained under the word-by-word paradigm, which suffers from the lack of a sentence-level training. We propose to reinforce G at the sentence level using the discriminative model (D), which aims to select the right answer from a few candidates, to ameliorate the problem. Experimental results on the VisDial dataset demonstrate the effectiveness of our approach.

CCS Concepts: • **Computing methodologies** → **Computer vision tasks**.

Additional Key Words and Phrases: visual dialog, vision and language, reinforcement learning, deep learning.

ACM Reference Format:
Hehe Fan, Linchao Zhu, Yi Yang, and Fei Wu. 2018. Recurrent Attention Network with Reinforced Generator for Visual Dialog. *J. ACM* 37, 4, Article 111 (August 2018), 16 pages. https://doi.org/10.1145/1122445.1122456

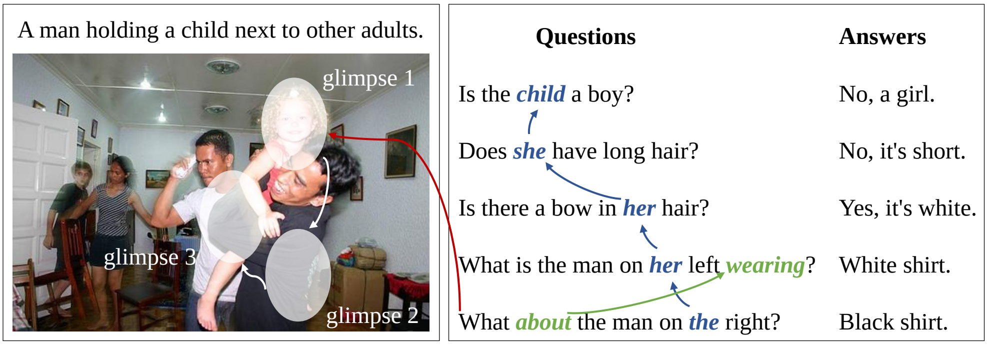
1 INTRODUCTION

This paper focuses on Visual Dialog [5, 6, 19, 32, 40], in which an agent perceives the environment visually and communicates with humans in natural language. When presented with an image, a dialog history, and a question about the visual content of the image, the agent answers the question in natural language. Unlike Visual Question Answering (VQA) [2, 21, 28, 39], where there are no follow-up questions, the questions in Visual Dialog are usually temporally ordered with narrative

[*] Part of this work was done when Hehe Fan was an intern at Baidu Research.

Authors' addresses: Hehe Fan, Center for Artificial Intelligence, University of Technology Sydney, 15 Broadway, Ultimo, Sydney, NSW, 2007, Australia, Baidu Research, No.1-No.5 No.10 Xibeiwang East Road, Haidian District, Beijing, 100193, China, hehe.fan@student.uts.edu.au; Linchao Zhu, Center for Artificial Intelligence, University of Technology Sydney, 15 Broadway, Ultimo, Sydney, NSW, 2007, Australia, linchao.zhu@uts.edu.au; Yi Yang, Center for Artificial Intelligence, University of Technology Sydney, 15 Broadway, Ultimo, Sydney, NSW, 2007, Australia, yi.yang@uts.edu.au; Fei Wu, College of Computer Science, Zhejiang University, 38 Zhedalu, Xihu, Hangzhou, Zhejiang, 310007, China, wufei@cs.zju.edu.cn.

0004-5411/2018/8-ART111 $15.00
https://doi.org/10.1145/1122445.1122456


A man holding a child next to other adults.

Questions	Answers
Is the *child* a boy?	No, a girl.
Does *she* have long hair?	No, it's short.
Is there a bow in *her* hair?	Yes, it's white.
What is the man on *her* left *wearing*?	White shirt.
What *about* the man on *the* right?	Black shirt.

Fig. 1. An example of Visual Dialog. Given an image and its caption, the agent answers a series of questions. The dialog history is crucial for enabling the agent to conduct a dialog. For example, to answer the last question, the agent has to infer what "the" and "about" refer to. The agent takes multiple glimpses to attend to the man on the left given the complex layout of the image and the dialog history.

structures. Visual Dialog posses the ability to hold a meaningful dialog with humans in natural language about visual content, which benefits a variety of applications.

It is necessary to carefully address two main problems in Visual Dialog. First, how to enable the agent to effectively parse the temporal context, *i.e.*, the dialog history, to understand the current question accurately. As shown in Figure 1, the agent has to infer what the two words "the" and "about" refer to in order to answer the last question. Second, how to attend to the region of interest in an image given a question and its temporal context, so that the agent has a comprehensive understanding of the rich visual content.

In this paper, we propose a recurrent attention network for Visual Dialog, which attempts to address the two problems by an encoder-decoder architecture. For the encoder, we use a dialog network, which is a Long Short-Term Memory (LSTM) [13], to memorize the temporal context of the dialog history. The dialog network triggers a question signal that includes the question and dialog history information. Then the signal is passed to an attention processor.

The attention processor is another LSTM integrated with an attention mechanism [3, 43], to parse the visual spatial context. Guided by the dialog network signal, the attention processor grounds the question in the image by iteratively glimpsing the visual content multiple times. Lastly, a state vector is generated by incorporating the multiple glimpses and passed to the decoder to generate or select an answer.

There are two decoders in Visual Dialog [5], a generative model (G) which generates answers in natural language, and a discriminative model (D) to select the best answers from candidates. A major problem in optimizing G is the lack of appropriate loss function for training. Typical word-by-word generation is trained to fit an answer at word level. However, two answers can be semantically different even though they are similar by word to word matching. For example, the sentence "he is 4 years old" is very similar to the ground-truth "he is 14 years old" by word-level matching, but the meanings are very different.

To ameliorate the problem, we propose to guide the training of G at sentence level by D. Our premise is that D is capable of measuring answers at sentence level, its knowledge therefore can be used as a complement of word-level training for G. In addition, the training of D is more robust because the training loss can be computed by directly checking whether the selected answers are exactly the ground truths. Since transferring information from D to G is non-differentiable, we use Reinforcement Learning [35, 36] to achieve the goal by reward. Experiment on the VisDial dataset [5] demonstrates the effectiveness of the attention processor and the training method that reinforces G by D.

2 RELATED WORK

Vision and Language: There is increased interest in the field of multimodal learning [44] for bridging computer vision [7, 10] and natural language understanding, such as image captioning [8, 11, 12, 15, 34, 37, 38, 46], video captioning [17, 24, 31, 42], text-to-image coreference/grounding [14, 16, 25, 26, 30], Visual Question Answering (VQA) [2, 4, 21, 39, 41, 47] and Visual Dialog [5, 6, 19, 32, 40]. These tasks typically involve attention mechanisms.

The difference between Visual Dialog and VQA is that the questions in Visual Dialog are dependent on the history and often strongly contextual. In Visual Dialog, a question-answer pair is called a "fact" and the previous facts constitute "history". In contrast to [5, 19], Das *et al.* [6] trained two agents to communicate in natural language using an "image guessing" game. We follow [5, 19] by only implementing the answerer agent but evaluating the agent at dialog level, in which no ground-truth answer is available during evaluation.

Attention in Vision and Language: A number of prior works have applied attention mechanisms [3, 43] to vision and language tasks [1, 5, 20, 24, 45, 48]. For example, Yang *et al.* [45] presented stacked attention networks (SANs) that learned to answer natural language questions from images. Lu *et al.* [20] proposed two co-attention mechanisms (HieCoAtt) for VQA that jointly performed question-guided visual attention and image-guided question attention. Lu *et al.* also proposed a History-Conditioned Image Attentive Encoder (HCIAE) [19] for Visual Dialog that first used the current question to attend to the exchanges in the history, and then used the question and attended history to attend to the image, so as to obtain the final encoding. Wu *et al.* [40] applied the co-attention mechanism (CoAtt) to Visual Dialog by adding attentions on the dialog history. Different with these attention mechanisms, our recurrent attention network queries the image by several glimpses in each question-answering round. The core of the recurrent attention network is an attention processor that is implemented by an LSTM with an attention mechanism. For each glimpse, the attention processor takes the attention query and the last attention result as input and outputs the new attention result. The last state of the LSTM is used to initialize the decoder's state.

Reinforcement Learning: Many breakthroughs in reinforcement learning have recently been made. It is generally accepted that there are two families of model-free methods for reinforcement learning. The first consists of value-based methods, where an action value function is learned to guide the agent to act in the next time step. One algorithm is Q-learning [35], *e.g.*, DQN[23], which aims to directly approximate the optimal action value function. In contrast to value-based methods, the second type are policy-based methods which directly parameterize the policy π, such as REINFORCE [36]. Standard REINFORCE updates the policy parameters in the gradient ascent direction.

REINFORCE has been widely used in computer vision and natural language processing. Mnih *et al.* [22] developed the recurrent visual attention model which learns the spatial attention policies for image classification. Fan *et al.* [9] applied REINFORCE to efficient video classification. Ranzato *et al.* [27] proposed a sequence level training method for recurrent neural networks with REINFORCE. Rennie *et al.* [29] used the sequence level training for image captioning. Das *et al.* used REINFORCE to train two agents to communicate in natural language for Visual Dialog. Wu *et al.* [40] used adversarial REINFORCE with an intermediate reward to encourage the generator to generate responses as human generated dialogs. In this paper, we use the basic REINFORCE to provide a sentence-level training for the generator G.

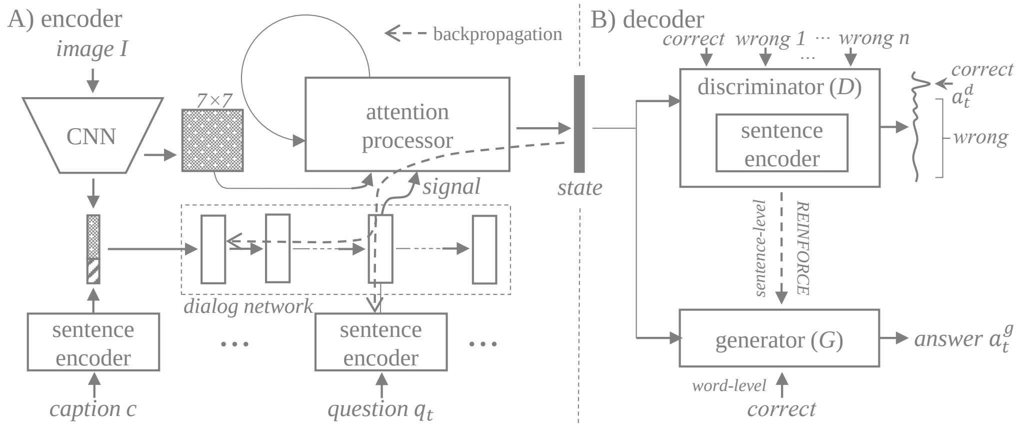


Fig. 2. The architecture of the proposed recurrent attention network. The encoder (A) consists of a sentence encoder, a dialog network and an attention processor. The sentence encoder embeds a natural language sentence to a one-dimensional vector. The dialog network is designed to parse the temporal context of dialog. Since LSTM is able to model sequences, we leverage an LSTM to model question dependency in a dialog. According to the order of questions, the dialog network can understand pronouns in a sequence of questions. Specially, at each question-answering round, the dialog network takes an embedded question as input. Then, according to its memory, the dialog network emits a signal that integrates the question and the temporal context. The attention processor grounds the signal by glimpsing the image multiple times and generates a state vector to the decoder. The given image and caption are used to initialize the state of the dialog network. The decoder (B) includes a discriminator (*D*) and a generator (*G*). The *D* selects best answers from candidates. The *G* directly generates answers in natural language. In addition to the word-by-word training, the *G* is further improved by *D* at sentence level.

3 MODEL

In this section, we describe the details of our recurrent attention network. As shown in Figure 2, the network consists of an encoder and a decoder. Before the first question-answering round, the agent is given the image I and a caption c of the image. In each round t, the encoder analyzes the dialog history, integrates the current question q_t, observes the image and generates a state vector for the decoder. According to the state vector, the discriminator decoder D then selects the best answer a_t^d from a set of candidates. The generator decoder G generates an answer a_t^g for the question q_t in natural language. During training, we also utilize D to improve G by reinforcement learning.

3.1 Encoder

Dialog network: The dialog network is responsible for modeling the temporal context in the dialog history. Since a dialog is a sequence of question-answer pairs along time, we use an LSTM to memorize the temporal context. At each question-answering round, the dialog network takes the question embedding vector as input and generates a signal that integrates the temporal context and the question. At the beginning of a dialog, we concatenate the image feature and the caption embedding to initialize the state of the dialog network.

Note that we do not encode the previously generated answers back to the dialog network. The reasons are as follows: 1) In this paper, we evaluate models with a new protocol, in which the ground-truth answers are not available to the agent. Since the self-generated answers are not reliable, we do not use them as “facts”. 2) The generated answers are dependent on the encoder.

ALGORITHM 1: Recurrent Attention Mechanism

```
Input  : spatial image feature $\boldsymbol{F} \in \mathcal{R}^{d \times k^2}$; signal emitted by dialog network $\boldsymbol{q} \in \mathcal{R}^{d'}$;
         attention parameters of the attention processor $\boldsymbol{W_1} \in \mathcal{R}^{d' \times s}$, $\boldsymbol{W_2} \in \mathcal{R}^{a \times d}$, $\boldsymbol{M} \in \mathcal{R}^{a \times s}$, $\boldsymbol{v} \in \mathcal{R}^{a}$;
         Long Short-Term Memory of the attention processor LSTM;
         number of attention steps $T$.
Output : state $\boldsymbol{h}$.
Initialize: $t \leftarrow 0$.
$\boldsymbol{h} \leftarrow \boldsymbol{q}\boldsymbol{W_1}$ // initialization for the LSTM hidden state, which is an $s$-dimensional vector
$\boldsymbol{q} \leftarrow \frac{\boldsymbol{q}}{||\boldsymbol{q}||_2}$ // l2 normalization for signal $\boldsymbol{q}$
while $t < T$ do
    $t \leftarrow t + 1$
    $\boldsymbol{\alpha} \leftarrow \boldsymbol{v}^T \tanh(\boldsymbol{W_2}\boldsymbol{F} + \boldsymbol{M}\boldsymbol{h}\mathbb{1}^T)$ // generate attention vector, whose length is $k^2$
    $\boldsymbol{\alpha} \leftarrow \text{softmax}(\boldsymbol{\alpha})$ // attention weight
    $\boldsymbol{f} \leftarrow \boldsymbol{F}\boldsymbol{\alpha}^T$ // apply attention weight to the spatial image feature and generate a attended feature
      vector whose length is $d$
    $\boldsymbol{f} \leftarrow \frac{\boldsymbol{f}}{||\boldsymbol{f}||_2}$ // l2 normalization for attended feature vector
    $\boldsymbol{i} \leftarrow \text{concatenate}(\boldsymbol{f}, \boldsymbol{q})$
    $\boldsymbol{h} \leftarrow \text{LSTM}(\boldsymbol{i}, \boldsymbol{h})$
end
```

The information that the decoder needs is produced by the encoder. Therefore, there is no need to encode the previously generated answers back to the dialog network.

Attention processor: The attention processor takes the signal emitted by the dialog network as input and parses the spatial context in the image to generate a state vector for the decoder. The attention processor is implemented by an LSTM with an attention mechanism at each glimpse. The LSTM memorizes the attention history and collects the visual information for the question. Before the first glimpse, the state of our attention processor is initialized by the signal emitted by the dialog network. For each glimpse, the attention processor uses the LSTM's state to attend to the image. The attention result will be used as input for the next glimpse. Since the LSTM memorizes the attention history, the attention processor is capable of avoiding repeated attentions for future glimpses. Figure 3 illustrates how the attention processor grounds a question in an image. Although sometimes it is difficult to precisely define how many glimpses are actually needed to correctly answer a question, a single-step attention mechanism is usually not enough.

We use the spatial features $\boldsymbol{F} \in \mathcal{R}^{d \times k \times k}$ from the activation of a Convolutional Neural Network (ConvNet) layer as the image representations to be attended. Before the model attends the the spatial feature $\boldsymbol{F} \in \mathcal{R}^{d \times k \times k}$, we reshape $\boldsymbol{F}$ to $\mathcal{R}^{d \times k^2}$. For the VGG-16 network [33], we use the output of pool$_5$ layer in which $d = 512$ and $k = 7$.

Suppose s is the LSTM hidden state size, a is the length of attention vector and $\mathbb{1} \in \mathcal{R}^{k^2}$ is a vector with all elements set to 1. The signal emitted by dialog network is $\boldsymbol{q} \in \mathcal{R}^{d'}$. Our attention processor contains an LSTM and a few trainable parameters $\boldsymbol{W_1} \in \mathcal{R}^{d' \times s}$, $\boldsymbol{W_2} \in \mathcal{R}^{a \times d}$, $\boldsymbol{M} \in \mathcal{R}^{a \times s}$, $\boldsymbol{v} \in \mathcal{R}^{a}$. We show the details of the recurrent attention mechanism in Alg 1. The $\alpha^T \in \mathcal{R}^{k}$ is the attention weight over the spatial image feature $\boldsymbol{F}$. Since $\boldsymbol{f}$ (visual features) and $\boldsymbol{q}$ (semantic features) are from two different feature spaces, we normalize them before concatenating them to a vector.

3.2 Decoder

There are two kinds of models for the decoder in Visual Dialog, *e.g.*, the discriminative model (D) and the generative model (G). The discriminator aims to select the ground-truth answer from a set

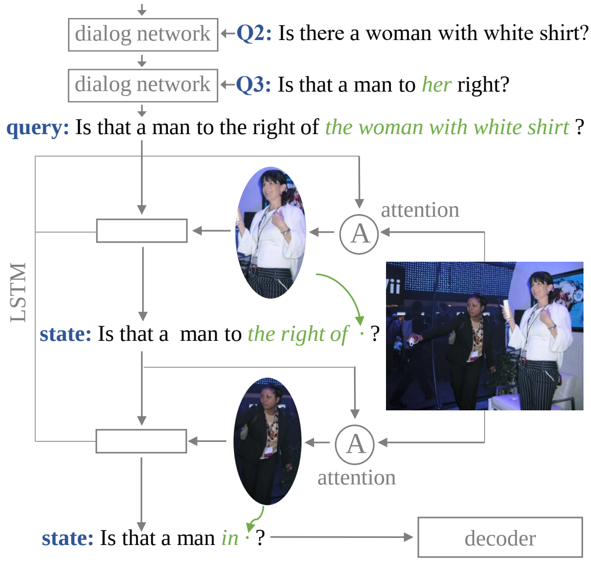

dialog network
Q2: Is there a woman with white shirt?
dialog network
Q3: Is that a man to her right?
query: Is that a man to the right of the woman with white shirt ?
LSTM
attention
A
state: Is that a man to the right of · ?
A
attention
state: Is that a man in · ?
decoder

Fig. 3. Illustration of how the attention processor grounds a question in an image. First, according to dialog history, the dialog network parse the pronoun “her". Then, based on the output of the dialog network, the attention processor glimpses the image to answer the question. Fig. 5 demonstrates three example of the behavior of our attention processor.

of candidates at sentence level, while the generator directly generates a sentence and maximizes the likelihood of each word in the ground-truth answer. Unlike [5], in which the two models are trained and evaluated separately, we train D and G jointly. Furthermore, we use D to improve G at sentence level. In this section, we denote the output state vector of encoder as $\boldsymbol{h}_0$.

Discriminator (D): The discriminator computes dot-product similarities between $\boldsymbol{h}_0$ and each of the embedded answer candidates. As shown in Figure 4 (A), the similarities are fed into a softmax function during training to compute the posterior probability over the candidates. The network is trained to maximize the log-likelihood of the ground-truth answer. We denote $f(\cdot)$ as the sentence encoder and $\boldsymbol{s}_r$ as the ground-truth answer, then training D involves minimizing:

$$L_D = -\log p(f(\boldsymbol{s}_r)|\boldsymbol{h}_0) \propto -f(\boldsymbol{s}_r) \cdot \boldsymbol{h}_0, \tag{1}$$

In this step, both the sentence encoder f and the dialog network are updated. During evaluation, candidates are ranked based on their posterior probabilities.

Generator (G): We denote the ground-truth answer $\boldsymbol{s}_r$ as a sequence of words $[w_1, w_2, ..., w_T]$, and training G involves minimizing:

$$\begin{aligned} L_G &= -\log p(\boldsymbol{s}_r|\boldsymbol{h}_0) = -\log p(w_1, w_2, ..., w_T|\boldsymbol{h}_0) \\ &= -\log \prod_{t=1}^{T} p(w_t|w_1, w_2, ..., w_{t-1}, \boldsymbol{h}_0) = -\sum_{t=1}^{T} \log p(w_t|w_1, w_2, ..., w_{t-1}, \boldsymbol{h}_0) \end{aligned} \tag{2}$$

Since Eq. (2) aims to maximize the likelihood of each word in the ground-truth answer, we call this method word-level training. Usually, the term $p(w_t|w_1, ..., w_{t-1}, \boldsymbol{h}_0)$ is modeled by an LSTM that takes $\boldsymbol{h}_0$ as its initial state:

$$p(\cdot|w_1, w_2, ..., w_{t-1}, \boldsymbol{h}_0), \boldsymbol{h}_t = LSTM(w_{t-1}, \boldsymbol{h}_{t-1}) \tag{3}$$

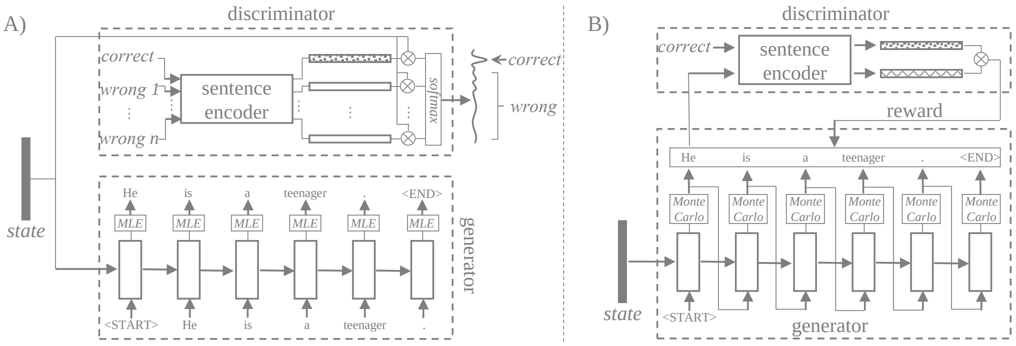


Fig. 4. The architecture of the decoder. (A) Joint training of discriminator (D) and generator (G). (B) Reinforcing G by D. Because D can measure answers at sentence level, it can be used to evaluate and guide G's behavior during training. Because this process is not differentiable, reinforcement learning is used.

where w_0 is the start symbol <START>. What's more, the distribution $p(\cdot|w_1, w_2, ..., w_{t-1}, \boldsymbol{h}_0)$ is a parametric function of $\boldsymbol{h}_t$. LSTM first generates the current hidden state $\boldsymbol{h}_t$ and then emits the distribution by a fully-connected layer according to $\boldsymbol{h}_t$. For simplicity, we use $\pi(\cdot|\boldsymbol{h}_t)$ to denote this distribution:

$$\pi(\cdot|\boldsymbol{h}_t) = p(\cdot|w_1, w_2, ..., w_{t-1}, \boldsymbol{h}_0) \tag{4}$$

The current word is generated by $w_t = \operatorname{argmax}_w \pi(\cdot|\boldsymbol{h}_t)$. During training, the previous ground-truth words are given. When conducting the evaluation, the previous ground-truth words are unavailable and are generated by maximum likelihood estimation (MLE). The $\log\pi$ is used to rank candidate answers as log-likelihood scores.

The generator and the sentence encoder share the same word embedding matrix. Compared with discriminative models, generative decoders are more practical for realistic applications but often achieve lower accuracy than discriminative models.

Reinforcing G by D: The answer to a question is usually not unique. However, training with Eq. (2) will make the agent focus only on the unique answer provided by the dataset. As a result, the agent may mistakenly treat semantically similar sentences as wrong answers. Since D can measure answers at sentence level, we additionally use D to evaluate and guide G's behavior during training. Suppose $g(\cdot)$ is the generative function to be learned, the sentence-level training is to :

$$L_J = -\log p(f(g(\boldsymbol{h}_0))|\boldsymbol{h}_0) \approx -\log p(f(g(\boldsymbol{h}_0))|f(s_r)) \tag{5}$$

Since $f(\cdot)$ can map semantically similar sentences to similar embeddings, G has the freedom to generate a more reasonable distribution that does not only fit the unique ground-truth answer in the training dataset. While $f(g(\boldsymbol{h}_0))$ is non-differentiable, we use reinforcement learning to transfer this information by reward.

Reinforcement learning is about an agent interacting with an environment, and learning an optimal policy, by trial and error, for sequence decision making. We consider the sentence generation as a process of word sequence decision making. At each time step t, G selects a word w_t from the word space $\mathcal{W}$ according to its internal hidden state $\boldsymbol{h}_t$ and policy that maps the hidden state space to the word space. Here, the policy is π in Eq. (4). Then, G updates its state to $\boldsymbol{h}_{t+1}$ and selects the next word w_{t+1}. After reaching the terminator <END>, G receives a reward r provided by D, as shown in Figure 4 (B). The goal of reinforcement learning is to learn the optimal policy that maximizes such rewards. Suppose $\boldsymbol{s}$ is an answer generated by G and $\boldsymbol{\mathcal{S}}$ is the sentence space. The objective of reinforcement learning is to maximize:

$$J(\theta) = \mathop{\mathbb{E}}_{s \sim \mathcal{S}} r(s|\boldsymbol{h}_0) \tag{6}$$

where θ represents the parameters of the policy π and $J(\cdot)$ is the expected reward under the distribution of possible sentences. The gradients of the objective are:

$$\nabla_\theta J = \mathop{\mathbb{E}}_{s \sim \mathcal{S}} \nabla_\theta \log\pi(s|\boldsymbol{h}_0) r(s|\boldsymbol{h}_0) = \mathop{\mathbb{E}}_{s \sim \mathcal{S}} \left[\sum_{t=1}^{T} \nabla_\theta \log\pi(w_t|\boldsymbol{h}_t) r(s|\boldsymbol{h}_0) \right] \tag{7}$$

The gradients can be backpropagated to G's LSTM and the entire encoder via $\boldsymbol{h}_0$.

Since $\boldsymbol{h}_0 \approx f(\boldsymbol{s}_r)$, we use $\boldsymbol{s}_r$ to replace $\boldsymbol{h}_0$. Then the reward function is:

$$r(\boldsymbol{s}|\boldsymbol{s}_r) = f(\boldsymbol{s}) \cdot f(\boldsymbol{s}_r) \tag{8}$$

The reward will encourage ($r(\boldsymbol{s}|\boldsymbol{s}_r) > 0$) or discourage ($r(\boldsymbol{s}|\boldsymbol{s}_r) < 0$) the generation of $\boldsymbol{s}$. Furthermore, $\boldsymbol{s}_1$ will be encouraged more than $\boldsymbol{s}_2$ when $r(\boldsymbol{s}_1|\boldsymbol{s}_r) > r(\boldsymbol{s}_2|\boldsymbol{s}_r)$.

Since the dimension of the possible sentence space $\mathcal{S}$ can be very high, it is impossible to optimize Eq. (2) directly. Following REINFORCE [36], we use Monte Carlo sampling to approximate the policy gradients:

$$w_t \sim \pi(\cdot|\boldsymbol{h}_t) \tag{9}$$

Note that the word-level training also optimizes the policy π and can significantly reduce the sentence space $\mathcal{S}$ by forcing G to generate grammatically correct answers. Therefore, the word-level training can be considered as supplementary to the sentence-level training.

In training, we minimize the following hybrid loss:

$$L = \alpha L_D + \beta L_G - \gamma J \tag{10}$$

where α, β and γ are three positive factors.

4 EXPERIMENTS

Dataset: We evaluate our proposed approach on the VisDial v0.9 dataset [5], which contains 83k dialogs on COCO-train and 40k on COCO-val images. Each dialog contains one image, one caption and 10 question-answering rounds. The captions are from the COCO dataset [18]. The questions and answers are collected by pairing two questioners and answerers on Amazon Mechanical Turk to chat about an image. The questioners see only the image captions and the image remains hidden to them. Their task is to ask questions about the hidden images to 'imagine the scenes better'. According to the image and caption, the answerers answer questions asked by their chat partner. Every question is coupled with 100 candidate answer options, of which only one option is correct.

Evaluation Protocol: In [5], the agent takes the ground-truth answers of all previous questions as input information to generate an answer for a new question. When testing the t-th round of a dialog, all ground-truth answers from round 1 to round $t-1$ are provided to the system. In addition, for generator, candidate answer options are used as teacher forcing for each question. In real-world applications, however, the ground truth answers are always not available. Therefore, we adopt a different evaluation setting from [5]. The difference is that the ground-truth answers to the history questions are not provided to the agent. The agent uses the answers generated (G) or selected (D) by itself or does not use them. The models are asked to sort the 100 candidate answer options for each question. D uses posterior probabilities to rank these answer options, and G uses the log-likelihood of these options for ranking. The model is evaluated on the following retrieval metrics: (1) mean rank of human response (lower is better), (2) recall@k, *i.e.*, existence of the human response in top-k ranked responses (higher is better), and (3) mean reciprocal rank (MRR) of the human response (higher is better).

Implementation Details: All LSTMs in the model are single layer with $512d$ hidden state. For the sentence encoder, we adopt the dynamic recurrent neural network structure. We use VGG-16 [33] to extract image features. Images are resized to 224 × 224 pixels. We use the output of pool5 (7 × 7 × 512) for attention and the output of fc7 (4096) to initialize the dialog network's state. The vocabulary consists of 8,836 words and each word occurs at least five times in the training dataset. We use the Adam optimizer with the learning rate of 0.001 and clip the gradients by 5.0. Consistent with [5], we split the 83k training dataset into 82k for train, 1k for val, and use the 40k as test. Each model is trained for 80 epochs. We have implemented our algorithm using both PaddlePaddle and TensorFlow, which have shown similar performance. In this paper, we report the TensorFlow accuracy.

4.1 Evaluation of the attention processor

To evaluate the attention processor, we design three baseline models in this section. We train and evaluate the discriminative model and the generative model separately. The baseline models are as follows,

- **Plain Dialog Network (PDN):** Only the sentence encoder and the dialog network are included for each question-answering round. The output of the dialog network is directly used to initialize the decoder's state.
- **Image Dialog Network (IDN):** The image information is added on the basis of the Plain Dialog Network for each question-answering round. The output of the dialog network and the image feature are first normalized and then concatenated to initialize the decoder's state.
- **Attention Dialog Network (ADN):** The image information in the Image Dialog Network is replaced by the attention mechanism as Alg. 1: 1-1. The output of the dialog network and attended image feature are also normalized before being concatenated to initialize the decoder's state.

For the above baseline models, at the beginning of a dialog, the image feature and text caption embedding are concatenated to initialize the state of the dialog network. For the proposed attention processor, the number of attention steps T is set to $\{1, 2, 3, 4, 5\}$ respectively. The difference between the ADN model and the proposed attention processor with T=1 is that the attention processor includes an additional LSTM. Correctly, the ADN model outputs the attention result $\boldsymbol{f}$ concatenated with the output of the dialog network $\boldsymbol{q}$. The attention processor outputs the state of the LSTM, *i.e.*, the $\boldsymbol{h}$.

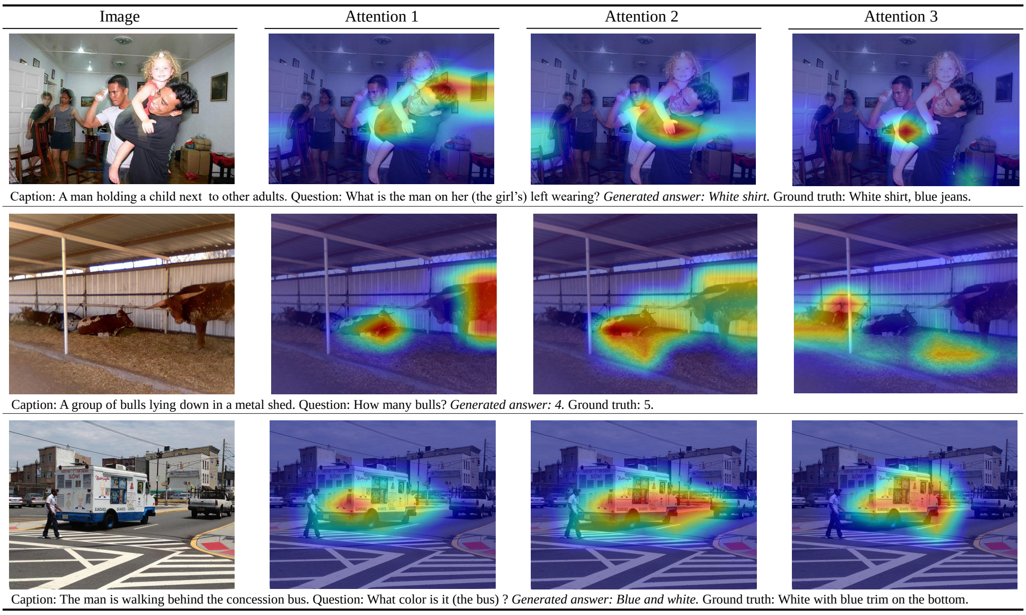
We run experiments at three times. Mean and standard deviation are reported in Table 1. Among the three baselines, ADN achieves the highest accuracy. IDN achieves the second highest accuracy. PDN achieves the lowest accuracy. For example, for the discriminative model, the recall@1 are 42.75, 42.96 and 43.56, respectively. The proposed recurrent attention mechanism significant outperforms the traditional attention mechanism (ADN), especially with the discriminative model. For example, for the discriminative model, our attention processor with $T = 1$ achieves 46.22 at recall@1, which outperforms ADN (43.56) by 2.66.

Forth, for our attention processor with the generative model, accuracy is improved with the increase in the number of attention steps. Our model usually achieves best accuracy when $T = 3$ or $T = 4$.

To further research the attention processor's behavior, we illustrate a few attention examples in Figure 5. In this experiment, we set $T = 3$. First, we observe from Attention 1 that our dialog network is capable of correctly parsing the temporal context in dialog. In the top example, the agent knows that "on her left" means "on the girl's left". In the bottom example, the agent knows that "it" means "the bus". Second, the attention processor exhibits different behavior according to

Table 1. Experimental results on the VisDial dataset. The number of attention steps T for the recurrent attention network are set to $\{1, 2, 3, 4, 5\}$. joint: D and G jointly training. att: recurrent attention network. RL: reinforce G by D.

Models			MRR	R@1	R@5	R@10	Mean
Generative model	baseline	PDN	0.4993±0.0003	40.96±0.03	58.32±0.01	60.98±0.08	22.85±0.02
		IDN	0.5023±0.0008	41.72±0.06	58.44±0.01	61.26±0.11	22.81±0.07
		ADN	0.5035±0.0002	42.00±0.10	58.56±0.03	61.32±0.03	22.75±0.01
	joint	PDN	0.5003±0.0004	40.93±0.01	58.47±0.12	61.13±0.05	22.83±0.02
		IDN	0.5026±0.0005	41.81±0.05	58.77±0.11	61.35±0.04	22.74±0.04
		ADN	0.5034±0.0002	42.17±0.08	58.64±0.07	61.43±0.02	22.71±0.08
	RL	PDN	0.5002±0.0008	41.04±0.04	58.52±0.05	61.12±0.12	22.76±0.01
		IDN	0.5032±0.0010	41.93±0.12	59.02±0.09	61.48±0.06	22.75±0.02
		ADN	0.5045±0.0003	42.04±0.05	59.09±0.04	61.70±0.05	22.64±0.02
	joint+RL	PDN	0.5009±0.0003	41.15±0.13	58.57±0.09	61.20±0.10	22.75±0.04
		IDN	0.5044±0.0005	42.09±0.05	59.04±0.14	61.61±0.08	22.73±0.06
		ADN	0.5053±0.0007	42.17±0.03	59.13±0.07	61.78±0.12	22.66±0.03
	att	$T = 1$	0.5028±0.0003	41.77±0.04	58.33±0.03	60.74±0.04	22.81±0.02
		$T = 2$	0.5092±0.0002	42.61±0.00	58.83±0.02	61.34±0.03	22.72±0.05
		$T = 3$	**0.5105±0.0001**	**42.79±0.03**	**58.90±0.02**	**61.53±0.03**	**22.62±0.02**
		$T = 4$	0.5098±0.0004	42.70±0.03	58.77±0.05	61.45±0.03	22.67±0.01
		$T = 5$	0.5093±0.0002	42.59±0.09	58.74±0.03	61.29±0.02	22.70±0.04
	att+joint	$T = 1$	0.5129±0.0005	43.15±0.05	58.97±0.10	61.55±0.05	22.56±0.04
		$T = 2$	0.5132±0.0002	43.23±0.04	59.07±0.02	61.58±0.03	22.47±0.02
		$T = 3$	0.5135±0.0002	43.23±0.03	59.08±0.07	61.62±0.03	22.40±0.03
		$T = 4$	**0.5140±0.0004**	**43.26±0.02**	**59.13±0.04**	**61.73±0.09**	**22.28±0.07**
		$T = 5$	0.5132±0.0003	43.15±0.03	59.02±0.04	61.52±0.07	22.49±0.04
	att+RL	$T = 1$	0.5131±0.0003	43.18±0.01	58.95±0.08	61.55±0.08	22.53±0.03
		$T = 2$	0.5136±0.0003	43.26±0.05	59.13±0.02	61.57±0.05	22.48±0.05
		$T = 3$	**0.5141±0.0002**	**43.29±0.04**	59.15±0.03	61.64±0.02	22.41±0.02
		$T = 4$	0.5138±0.0002	43.29±0.07	**59.18±0.04**	**61.70±0.00**	**22.34±0.01**
		$T = 5$	0.5133±0.0001	43.17±0.03	58.90±0.02	61.39±0.04	22.52±0.02
	att+joint+RL	$T = 1$	0.5142±0.0004	43.26±0.02	59.15±0.05	61.820±0.02	22.40±0.03
		$T = 2$	0.5145±0.0003	43.37±0.03	59.29±0.09	61.88±0.08	22.37±0.03
		$T = 3$	0.5148±0.0001	43.39±0.05	**59.38±0.02**	61.97±0.06	22.32±0.04
		$T = 4$	**0.5154±0.0002**	**43.42±0.02**	59.31±0.03	**62.04±0.05**	**22.24±0.02**
		$T = 5$	0.5140±0.0001	43.32±0.01	59.08±0.03	61.87±0.01	22.41±0.04
Discriminative model	baseline	PDN	0.5701±0.0010	42.75±0.02	74.27±0.02	83.17±0.07	6.16±0.02
		IDN	0.5727±0.0007	42.96±0.01	74.16±0.02	83.45±0.13	6.03±0.01
		ADN	0.5832±0.0005	43.56±0.02	74.84±0.05	83.94±0.05	5.96±0.03
	joint	PDN	0.5703±0.0007	42.80±0.04	74.33±0.00	83.66±0.08	6.13±0.02
		IDN	0.5725±0.0012	42.95±0.04	74.17±0.02	83.51±0.05	6.05±0.04
		ADN	0.5836±0.0008	43.73±0.07	74.98±0.05	84.03±0.03	5.92±0.05
	att	$T = 1$	**0.6011±0.0004**	**46.22±0.02**	**76.99±0.04**	**86.11±0.02**	**5.18±0.02**
		$T = 2$	0.6007±0.0004	46.10±0.01	76.78±0.02	86.05±0.03	5.22±0.03
		$T = 3$	0.6004±0.0006	46.12±0.05	76.40±0.07	85.90±0.06	5.27±0.02
		$T = 4$	0.6004±0.0011	46.14±0.04	76.47±0.01	85.94±0.03	5.24±0.01
		$T = 5$	0.5998±0.0006	46.04±0.03	76.10±0.03	85.84±0.07	5.30±0.04
	att+joint	$T = 1$	0.6025±0.0003	46.33±0.03	77.07±0.11	86.26±0.05	5.12±0.04
		$T = 2$	**0.6040±0.0005**	**46.52±0.04**	**77.10±0.04**	**86.34±0.07**	**5.10±0.03**
		$T = 3$	0.6024±0.0002	46.45±0.04	76.91±0.02	86.14±0.12	5.18±0.02
		$T = 4$	0.6006±0.0004	46.19±0.03	76.80±0.02	86.09±0.08	5.20±0.00
		$T = 5$	0.5998±0.0006	46.06±0.02	76.64±0.07	86.11±0.06	5.22±0.04



Caption: A man holding a child next to other adults. Question: What is the man on her (the girl's) left wearing? *Generated answer: White shirt.* Ground truth: White shirt, blue jeans.

Caption: A group of bulls lying down in a metal shed. Question: How many bulls? *Generated answer: 4.* Ground truth: 5.

Caption: The man is walking behind the concession bus. Question: What color is it (the bus) ? *Generated answer: Blue and white.* Ground truth: White with blue trim on the bottom.

Fig. 5. Visualization examples of how the attention processor grounds questions in images.

different questions. In the top example, the attention gradually moves from the middle between "the girl" and "the man on her left" to "the man on her left", and finally focuses on the body of "the man on her left". For the middle example, the attention processor counts the bulls in the image sequentially. In the bottom example, the attention moves from the tail of the bus to the head to check the color.

4.2 Evaluation of the Entire Model

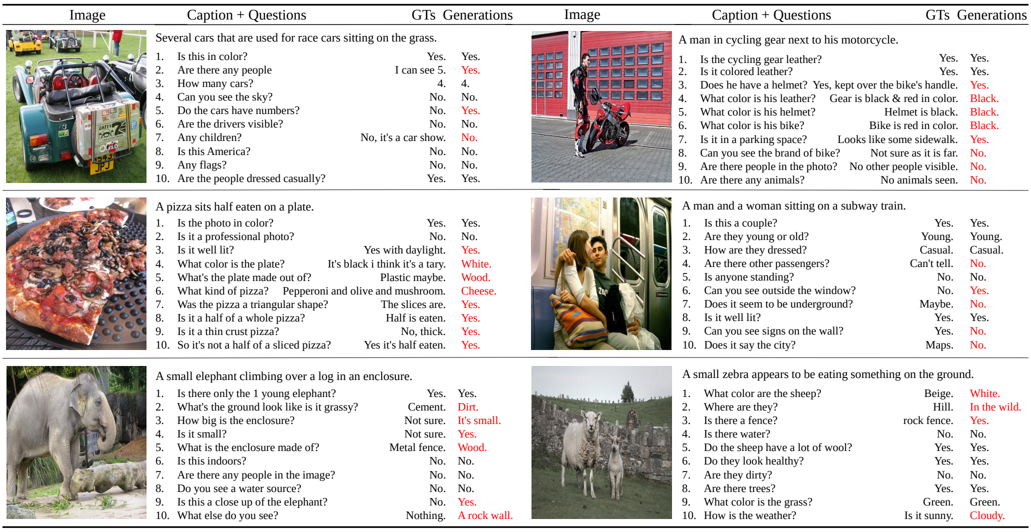
In this section, we evaluate the training method proposed in Eq. (10). In Section 4.1, we find that D converges faster than G. Therefore, we set $\alpha = 0.5$ and $\beta = 1.0$. We first evaluate the discriminator-generator joint training method ($\gamma = 0$) and then evaluate the proposed sentence-level training which reinforces G by D ($\gamma \neq 0$). The number of attention steps T is tested from $\{1, 2, 3, 4, 5\}$.

The results of the discriminator-generator joint training are shown in Table 1. Compared with separate training ("att"), this method, although simple, significantly improves the accuracy for both D and G. For the generative model, the best recall@1 by separate training ("att") is 42.79 while is 43.26 by joint training ("att+joint").

Next, we evaluate the sentence-level training method. In this experiment, we set $\gamma = 0.1$. Since this method aims to improve G and has little influence on D in the experiment, we only show the generative model's results in Table 1. As we can see from the results, sentence-level training ("att+joint+RL") further improves the accuracy on the basis of discriminator-generator joint training, especially for recall@5 and recall@10. For the generative model with three attention steps, the G improves 0.30 ("att+joint") at recall@5.

4.3 Comparison with other models

We have proposed an encoder-decoder model for Visual Dialog. For the encoder, we use a dialog network to memorize the temporal context of the dialog history and an attention processor to



Several cars that are used for race cars sitting on the grass.

	Caption + Questions	GTs	Generations
1.	Is this in color?	Yes.	Yes.
2.	Are there any people	I can see 5.	Yes.
3.	How many cars?	4.	4.
4.	Can you see the sky?	No.	No.
5.	Do the cars have numbers?	No.	Yes.
6.	Are the drivers visible?	No.	No.
7.	Any children?	No, it's a car show.	No.
8.	Is this America?	No.	No.
9.	Any flags?	No.	No.
10.	Are the people dressed casually?	Yes.	Yes.

A man in cycling gear next to his motorcycle.

	Caption + Questions	GTs	Generations
1.	Is the cycling gear leather?	Yes.	Yes.
2.	Is it colored leather?	Yes.	Yes.
3.	Does he have a helmet?	Yes, kept over the bike's handle.	Yes.
4.	What color is his leather?	Gear is black & red in color.	Black.
5.	What color is his helmet?	Helmet is black.	Black.
6.	What color is his bike?	Bike is red in color.	Black.
7.	Is it in a parking space?	Looks like some sidewalk.	Yes.
8.	Can you see the brand of bike?	Not sure as it is far.	No.
9.	Are there people in the photo?	No other people visible.	No.
10.	Are there any animals?	No animals seen.	No.

A pizza sits half eaten on a plate.

	Caption + Questions	GTs	Generations
1.	Is the photo in color?	Yes.	Yes.
2.	Is it a professional photo?	No.	No.
3.	Is it well lit?	Yes with daylight.	Yes.
4.	What color is the plate?	It's black i think it's a tary.	White.
5.	What's the plate made out of?	Plastic maybe.	Wood.
6.	What kind of pizza?	Pepperoni and olive and mushroom.	Cheese.
7.	Was the pizza a triangular shape?	The slices are.	Yes.
8.	Is it a half of a whole pizza?	Half is eaten.	Yes.
9.	Is it a thin crust pizza?	No, thick.	Yes.
10.	So it's not a half of a sliced pizza?	Yes it's half eaten.	Yes.

A man and a woman sitting on a subway train.

	Caption + Questions	GTs	Generations
1.	Is this a couple?	Yes.	Yes.
2.	Are they young or old?	Young.	Young.
3.	How are they dressed?	Casual.	Casual.
4.	Are there other passengers?	Can't tell.	No.
5.	Is anyone standing?	No.	No.
6.	Can you see outside the window?	No.	Yes.
7.	Does it seem to be underground?	Maybe.	No.
8.	Is it well lit?	Yes.	Yes.
9.	Can you see signs on the wall?	Yes.	No.
10.	Does it say the city?	Maps.	No.

A small elephant climbing over a log in an enclosure.

	Caption + Questions	GTs	Generations
1.	Is there only the 1 young elephant?	Yes.	Yes.
2.	What's the ground look like is it grassy?	Cement.	Dirt.
3.	How big is the enclosure?	Not sure.	It's small.
4.	Is it small?	Not sure.	Yes.
5.	What is the enclosure made of?	Metal fence.	Wood.
6.	Is this indoors?	No.	No.
7.	Are there any people in the image?	No.	No.
8.	Do you see a water source?	No.	No.
9.	Is this a close up of the elephant?	No.	Yes.
10.	What else do you see?	Nothing.	A rock wall.

A small zebra appears to be eating something on the ground.

	Caption + Questions	GTs	Generations
1.	What color are the sheep?	Beige.	White.
2.	Where are they?	Hill.	In the wild.
3.	Is there a fence?	rock fence.	Yes.
4.	Is there water?	No.	No.
5.	Do the sheep have a lot of wool?	Yes.	Yes.
6.	Do they look healthy?	Yes.	Yes.
7.	Are they dirty?	No.	No.
8.	Are there trees?	Yes.	Yes.
9.	What color is the grass?	Green.	Green.
10.	How is the weather?	Is it sunny.	Cloudy.

Fig. 6. Examples of generated answers by our model. The red color indicates the generated answer is different with the ground truth.

Table 2. Comparison with Late Fusion (LF) Encoder, Hierarchical Recurrent Encoder with Attention (HREA), Memory Network (MN) and History-Conditioned Image Attentive Encoder (HCIAE).

Model	Generative Model					Discriminative Model				
	MRR	R@1	R@5	R@10	Mean	MRR	R@1	R@5	R@10	Mean
LF [5]	0.5033	41.69	58.39	60.94	22.87	0.5794	44.00	74.03	83.11	6.09
HREA [5]	0.5032	41.80	58.20	60.77	22.52	0.5762	43.37	74.72	84.09	5.71
MN [5]	0.5004	41.41	58.19	60.64	23.18	0.5763	43.79	73.76	83.80	6.23
HCIAE-G-DIS [19]	0.5094	42.21	59.01	61.33	22.38	-	-	-	-	-
HCIAE-D-NP-ATT [19]	-	-	-	-	-	0.5905	44.98	76.11	85.12	5.63
Ours	**0.5148**	**43.39**	**59.38**	**61.97**	**22.32**	**0.6024**	**46.45**	**76.91**	**86.34**	**5.10**

glimpse the image. For the decoder, we train the discriminator and generator jointly. We further use discriminator to guide the generator at the sentence level. Experiments show that our model significantly improves accuracy compared to the LF, HREA, MN and HCIAE models.

We compare our model with three models proposed in [5]. They are Late Fusion (LF) Encoder, Hierarchical Recurrent Encoder with Attention (HREA) and Memory Network (MN). We implement the three models by ourselves and apply it to our setting. As strong competitors, we also include History-Conditioned Image Attentive Encoder (HCIAE) [19] models. For HCIAE models, the HCIAE-G(enerative)-DIS(criminator), which is trained under the mixed MLE and discriminator loss (knowledge transfer), is the best generative model, while the HCIAE-D(iscriminative)-NP(air)-ATT(entive), which is trained under the n-pair discriminative loss and using the self-attentive answer encoding, is the best discriminative model. Therefore, we use these two models to represent HCIAE models. Note that, we do not directly compare our results to those numbers reported in [5, 19] because we do not use the history ground-truth answers for evaluation. Instead, as we illustrated in the evaluation protocol, we use the answers selected or generated by the models. For our model, we set $T = 3$ since glimpsing three times achieves best performance on the validation dataset. The results are shown in Table 2.

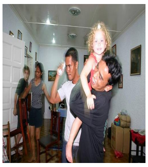
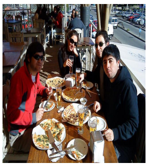
Image	Caption + Questions	GTs	Ours	With GT Answers		Without GT Answers	
				MN	HCIAE-G-DIS	MN	HCIAE-G-DIS
	A man holding a child next to other adults.						
	Is this a family photo?	Yes.	Yes.	Yes.	Yes.	Yes.	Yes.
	Is everyone dressed up?	Yes.	Yes.	Yes.	Yes.	Yes.	Yes.
	How many adults?	3.	3.	4.	4.	4.	4.
	What is the child wearing?	Red shirt.	Red shirt.	Red shirt.	Red shirt.	Black shirt.	I can't tell.
	Are the outfits matching?	No.	No.	No.	No.	No.	No.
	Is the child a boy?	No, a girl.	No, a girl.	No.	No.	No.	No.
	Does she have long hair?	No, it's short.	No.	No.	No.	No.	No.
	Is there a bow in her hair?	Yes, it's white.	Yes.	No.	No.	No.	No.
	What is the man on the left wearing?	White shirt, blue jeans.	White shirt.	White shirt.	White shirt.	White shirt.	White shirt.
	A group of people sitting around a table eating pizza.						
	Are these friends?	Yes.	Yes.	Yes.	Yes.	Yes.	Yes.
	Is this inside?	Yes.	Yes.	Yes.	Yes.	Yes.	Yes.
	Are they facing the camera?	Yes.	Yes.	Yes.	Yes.	Yes.	Yes.
	Is this a restaurant?	Yes.	Yes.	Yes.	Yes.	Yes.	Yes.
	Are they smiling?	Yes.	Yes.	No.	Yes.	No.	Yes.
	Is this in color?	Yes.	Yes.	Yes.	Yes.	Yes.	Yes.
	Are there more than 10?	No.	No.	No.	No.	No.	I think so.
	Are they in their 20s?	30's.	Yes.	Yes.	Maybe.	Yes.	Maybe.
	Is it daytime?	Yes.	Yes.	No.	Yes.	No.	Yes.

Fig. 7. Examples of generated answers with and without ground-truth answers. The red color indicates the generated answer is different with the ground truth. Since our framework do not use answers, there is no difference between whether to use ground-truth answers or not. For other methods, leveraging ground-truth answers can help agent to generate more reasonable answers. However, ground-truth answers are not available in practice.

Table 3. Comparison with other methods with the evaluation protocol proposed in [5], in which the ground-truth answers to the history questions are available to the agent.

Model	Generative Model					Discriminative Model				
	MRR	R@1	R@5	R@10	Mean	MRR	R@1	R@5	R@10	Mean
LF [5]	0.5204	42.04	61.78	67.66	16.84	0.5807	43.82	74.68	84.07	5.78
HER [5]	0.5237	42.29	62.18	67.92	17.07	0.5846	44.67	74.50	84.22	5.72
HREA [5]	0.5242	42.29	62.33	68.17	16.79	0.5868	44.82	74.81	84.36	5.66
MN [5]	0.5259	42.29	62.85	68.88	17.06	0.5965	45.55	76.22	85.37	5.46
SAN-QI [45]	-	-	-	-	-	0.5764	43.44	74.26	83.72	5.88
HieCoAtt-QI [20]	-	-	-	-	-	0.5788	43.51	74.49	83.96	5.84
AMEM [32]	-	-	-	-	-	0.6160	47.74	78.04	86.84	4.99
HCIAE-G-DIS [19]	0.5467	44.35	65.28	71.55	**14.23**	-	-	-	-	-
HCIAE-D-NP-ATT [19]	-	-	-	-	-	0.6222	48.48	78.75	87.59	4.81
Ours	0.5450	44.40	65.23	70.88	14.54	0.6227	48.63	78.62	87.49	4.95

As can be seen from the results, our method outperforms all three models in [5]. For example, for the generative model evaluated by recall@1, our model outperforms LF by 1.7, HREA by 1.59 and MN by 1.98. Our model also outperforms the two HCIAE models in [19]. For example, for the generative model evaluated by recall@1, our model outperforms HCIAE-G-DIS by 1.18. For the discriminative model evaluated by recall@1, our model outperforms HCIAE-D-NP-ATT by 1.47.

We present some examples of answers generated by our model in Figure 6. Occasionally, the model generates more accurate answers than the ground truths. For the example in the "elephant" dialog, when asked "what else do you see?", the model generates "a rock wall", which is more accurate than the ground truth "nothing". For the example in the last dialog, when asked "how is the weather", the model generates "cloudy", which is more accurate than the ground truth "it's sunny".

We also compare our model with other methods with the evaluation strategy proposed in [5], in which the ground-truth answers to the history questions are available to the agent. Apart from the LF, HERA, MN and HCIAE models, we also include the Hierarchical Recurrent Encoder (HER) [5], Stacked Attention Network with Question and Image (SAN-QI) [45], Hierarchical Question-Image

Co-Attention (HieCoAtt-QI) [20], AMEM [32] and Co-Attention [40] models in this experiment. Numbers are exactly as reported in prior works. In this experiment, the number of glimpses is set to 3. The results are shown in Table 3. Compared with Table 2, the ground-truth answers to the history question help the agent to answer the future questions. For example, the ground-truth history answers help HCIAE-D-NP-ATT to reduce mean rank by 1.16. Since the HCIAE [19] and Co-Attention [40] models bravely exploit the ground-truth question-answer pairs, *i.e.*, facts, and apply attention mechanisms to the facts, they achieve high accuracies. However, our model considers the answers unreliable and does not well uses them, thus it does not outperform the HCIAE and Co-Attention models in this experiment.

5 CONCLUSION

We have proposed an encoder-decoder model for Visual Dialog. For the encoder, we use a dialog network to memorize the temporal context of the dialog history and an attention processor to glimpse the image. For the decoder, we train the discriminator and generator jointly. We further use discriminator to guide the generator at the sentence level. Experiments show that our model significantly improves accuracy compared to the LF, HREA, MN and HCIAE models.

Acknowledgement. The work of Yi Yang was in part supported by ARC DP200100938. The work of Fei Wu was in part supported by NSFC61625107.

REFERENCES

[1] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. 2016. Neural Module Networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. 39–48. https://doi.org/10.1109/CVPR.2016.12

[2] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. VQA: Visual Question Answering. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*. 2425–2433. https://doi.org/10.1109/ICCV.2015.279

[3] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv* abs/1409.0473 (2014).

[4] Yalong Bai, Jianlong Fu, Tiejun Zhao, and Tao Mei. 2018. Deep Attention Neural Tensor Network for Visual Question Answering. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XII*. 21–37. https://doi.org/10.1007/978-3-030-01258-8_2

[5] Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José M. F. Moura, Devi Parikh, and Dhruv Batra. 2017. Visual Dialog. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*. 1080–1089. https://doi.org/10.1109/CVPR.2017.121

[6] Abhishek Das, Satwik Kottur, José M. F. Moura, Stefan Lee, and Dhruv Batra. 2017. Learning Cooperative Visual Dialog Agents with Deep Reinforcement Learning. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*. 2970–2979. https://doi.org/10.1109/ICCV.2017.321

[7] Yuhang Ding, Hehe Fan, Mingliang Xu, and Yi Yang. 2020. Adaptive Exploration for Unsupervised Person Re-Identification. *ACM Transactions on Multimedia Computing, Communications, and Applications TOMCCAP* 16, 1 (2020). https://doi.org/10.1145/3369393

[8] Jeff Donahue, Lisa Anne Hendricks, Marcus Rohrbach, Subhashini Venugopalan, Sergio Guadarrama, Kate Saenko, and Trevor Darrell. 2017. Long-Term Recurrent Convolutional Networks for Visual Recognition and Description. *IEEE Trans. Pattern Anal. Mach. Intell.* 39, 4 (2017), 677–691. https://doi.org/10.1109/TPAMI.2016.2599174

[9] Hehe Fan, Zhongwen Xu, Linchao Zhu, Chenggang Yan, Jianjun Ge, and Yi Yang. 2018. Watching a Small Portion could be as Good as Watching All: Towards Efficient Video Classification. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden.* 705–711. https://doi.org/10.24963/ijcai.2018/98

[10] Hehe Fan, Liang Zheng, Chenggang Yan, and Yi Yang. 2018. Unsupervised Person Re-identification: Clustering and Fine-tuning. *ACM Transactions on Multimedia Computing, Communications, and Applications TOMCCAP* 14, 4 (2018). https://doi.org/10.1145/3243316

[11] Hao Fang, Saurabh Gupta, Forrest N. Iandola, Rupesh Kumar Srivastava, Li Deng, Piotr Dollár, Jianfeng Gao, Xiaodong He, Margaret Mitchell, John C. Platt, C. Lawrence Zitnick, and Geoffrey Zweig. 2015. From captions to visual concepts

and back. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015.* 1473–1482. https://doi.org/10.1109/CVPR.2015.7298754

[12] Q. Feng, Y. Wu, H. Fan, C. Yan, M. Xu, and Y. Yang. 2020. Cascaded Revision Network for Novel Object Captioning. *IEEE Transactions on Circuits and Systems for Video Technology* (2020). https://doi.org/10.1109/TCSVT.2020.2965966

[13] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation* 9, 8 (1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

[14] Ronghang Hu, Marcus Rohrbach, and Trevor Darrell. 2016. Segmentation from Natural Language Expressions. In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part I.* 108–124. https://doi.org/10.1007/978-3-319-46448-0_7

[15] Andrej Karpathy and Li Fei-Fei. 2017. Deep Visual-Semantic Alignments for Generating Image Descriptions. *IEEE Trans. Pattern Anal. Mach. Intell.* 39, 4 (2017), 664–676. https://doi.org/10.1109/TPAMI.2016.2598339

[16] Chen Kong, Dahua Lin, Mohit Bansal, Raquel Urtasun, and Sanja Fidler. 2014. What Are You Talking About? Text-to-Image Coreference. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014.* 3558–3565. https://doi.org/10.1109/CVPR.2014.455

[17] Yehao Li, Ting Yao, Yingwei Pan, Hongyang Chao, and Tao Mei. 2018. Jointly Localizing and Describing Events for Dense Video Captioning. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018.* 7492–7500.

[18] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: Common Objects in Context. In *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V.* 740–755. https://doi.org/10.1007/978-3-319-10602-1_48

[19] 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. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA.* 313–323.

[20] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2016. Hierarchical Question-Image Co-Attention for Visual Question Answering. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain.* 289–297.

[21] Mateusz Malinowski, Marcus Rohrbach, and Mario Fritz. 2017. Ask Your Neurons: A Deep Learning Approach to Visual Question Answering. *International Journal of Computer Vision* 125, 1-3 (2017), 110–135. https://doi.org/10.1007/s11263-017-1038-2

[22] Volodymyr Mnih, Nicolas Heess, Alex Graves, and Koray Kavukcuoglu. 2014. Recurrent Models of Visual Attention. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada.* 2204–2212.

[23] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin A. Riedmiller, Andreas Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. 2015. Human-level control through deep reinforcement learning. *Nature* 518, 7540 (2015), 529–533. https://doi.org/10.1038/nature14236

[24] Pingbo Pan, Zhongwen Xu, Yi Yang, Fei Wu, and Yueting Zhuang. 2016. Hierarchical Recurrent Neural Encoder for Video Representation with Application to Captioning. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016.* 1029–1038. https://doi.org/10.1109/CVPR.2016.117

[25] Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2017. Flickr30k Entities: Collecting Region-to-Phrase Correspondences for Richer Image-to-Sentence Models. *International Journal of Computer Vision* 123, 1 (2017), 74–93. https://doi.org/10.1007/s11263-016-0965-7

[26] Vignesh Ramanathan, Armand Joulin, Percy Liang, and Fei-Fei Li. 2014. Linking People in Videos with "Their" Names Using Coreference Resolution. In *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I.* 95–110. https://doi.org/10.1007/978-3-319-10590-1_7

[27] Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2015. Sequence Level Training with Recurrent Neural Networks. *arXiv* abs/1511.06732 (2015).

[28] Mengye Ren, Ryan Kiros, and Richard S. Zemel. 2015. Exploring Models and Data for Image Question Answering. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada.* 2953–2961.

[29] Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jarret Ross, and Vaibhava Goel. 2017. Self-Critical Sequence Training for Image Captioning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017.* 1179–1195. https://doi.org/10.1109/CVPR.2017.131

[30] Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele. 2016. Grounding of Textual Phrases in Images by Reconstruction. In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part I.* 817–834. https://doi.org/10.1007/978-3-319-46448-0_49

[31] Anna Rohrbach, Marcus Rohrbach, Niket Tandon, and Bernt Schiele. 2015. A dataset for Movie Description. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015.* 3202–3212. https://doi.org/10.1109/CVPR.2015.7298940

[32] Paul Hongsuck Seo, Andreas Lehrmann, Bohyung Han, and Leonid Sigal. 2017. Visual Reference Resolution using Attention Memory for Visual Dialog. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA.* 3722–3732.

[33] Karen Simonyan and Andrew Zisserman. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* abs/1409.1556 (2014).

[34] Anqi Wang, Haifeng Hu, and Liang Yang. 2018. Image Captioning with Affective Guiding and Selective Attention. *TOMCCAP* 14, 3 (2018), 73:1–73:15. https://doi.org/10.1145/3226037

[35] Christopher J. C. H. Watkins and Peter Dayan. 1992. Q-learning. *Machine Learning* 8 (1992), 279–292. https://doi.org/10.1007/BF00992698

[36] Ronald J. Williams. 1992. Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning. *Machine Learning* 8 (1992), 229–256. https://doi.org/10.1007/BF00992696

[37] Jie Wu, Haifeng Hu, and Yi Wu. 2018. Image Captioning via Semantic Guidance Attention and Consensus Selection Strategy. *ACM Transactions on Multimedia Computing, Communications, and Applications TOMCCAP* 14, 4 (2018). https://doi.org/10.1145/3271485

[38] Qi Wu, Chunhua Shen, Lingqiao Liu, Anthony R. Dick, and Anton van den Hengel. 2016. What Value Do Explicit High Level Concepts Have in Vision to Language Problems?. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016.* 203–212.

[39] Qi Wu, Chunhua Shen, Peng Wang, Anthony R. Dick, and Anton van den Hengel. 2018. Image Captioning and Visual Question Answering Based on Attributes and External Knowledge. *IEEE Trans. Pattern Anal. Mach. Intell.* 40, 6 (2018), 1367–1381. https://doi.org/10.1109/TPAMI.2017.2708709

[40] Qi Wu, Peng Wang, Chunhua Shen, Ian D. Reid, and Anton van den Hengel. 2018. Are You Talking to Me? Reasoned Visual Dialog Generation through Adversarial Learning. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt lake city, Utah, USA, June 18-22, 2018.*

[41] Yu Wu, Lu Jiang, and Yi Yang. 2020. Revisiting EmbodiedQA: A Simple Baseline and Beyond. *IEEE Transactions on Image Processing* 29 (2020), 3984–3992. https://doi.org/10.1109/TIP.2020.2967584

[42] Jun Xu, Ting Yao, Yongdong Zhang, and Tao Mei. 2017. Learning Multimodal Attention LSTM Networks for Video Captioning. In *Proceedings of the 2017 ACM on Multimedia Conference, MM 2017, Mountain View, CA, USA, October 23-27, 2017.* 537–545. https://doi.org/10.1145/3123266.3123448

[43] 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. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015.* 2048–2057.

[44] Yan Yan, Feiping Nie, Wen Li, Chenqiang Gao, Yi Yang, and Dong Xu. 2016. Image Classification by Cross-Media Active Learning With Privileged Information. *IEEE Trans. Multimedia* 18, 12 (2016), 2494–2502. https://doi.org/10.1109/TMM.2016.2602938

[45] Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alexander J. Smola. 2016. Stacked Attention Networks for Image Question Answering. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016.* 21–29. https://doi.org/10.1109/CVPR.2016.10

[46] Dongfei Yu, Jianlong Fu, Tao Mei, and Yong Rui. 2017. Multi-level Attention Networks for Visual Question Answering. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017.* 4187–4195. https://doi.org/10.1109/CVPR.2017.446

[47] Linchao Zhu, Zhongwen Xu, Yi Yang, and Alexander G. Hauptmann. 2017. Uncovering the Temporal Context for Video Question Answering. *Int. J. Comput. Vis.* 124, 3 (2017), 409–421. https://doi.org/10.1007/s11263-017-1033-7

[48] Yuke Zhu, Oliver Groth, Michael S. Bernstein, and Li Fei-Fei. 2016. Visual7W: Grounded Question Answering in Images. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016.* 4995–5004. https://doi.org/10.1109/CVPR.2016.540