Towards Diverse and Natural Image Descriptions via a Conditional GAN

Bo Dai[1]   Sanja Fidler[23]   Raquel Urtasun[234]   Dahua Lin[1]

[1]Department of Information Engineering, The Chinese University of Hong Kong
[2]University of Toronto   [3]Vector Institute   [4]Uber Advanced Technologies Group

db014@ie.cuhk.edu.hk   fidler@cs.toronto.edu   urtasun@cs.toronto.edu   dhlin@ie.cuhk.edu.hk

Abstract

Despite the substantial progress in recent years, the image captioning techniques are still far from being perfect. Sentences produced by existing methods, e.g. those based on RNNs, are often overly rigid and lacking in variability. This issue is related to a learning principle widely used in practice, that is, to maximize the likelihood of training samples. This principle encourages high resemblance to the "ground-truth" captions, while suppressing other reasonable descriptions. Conventional evaluation metrics, e.g. BLEU and METEOR, also favor such restrictive methods. In this paper, we explore an alternative approach, with the aim to improve the naturalness and diversity – two essential properties of human expression. Specifically, we propose a new framework based on Conditional Generative Adversarial Networks (CGAN), which jointly learns a generator to produce descriptions conditioned on images and an evaluator to assess how well a description fits the visual content. It is noteworthy that training a sequence generator is nontrivial. We overcome the difficulty by Policy Gradient, a strategy stemming from Reinforcement Learning, which allows the generator to receive early feedback along the way. We tested our method on two large datasets, where it performed competitively against real people in our user study and outperformed other methods on various tasks.

1. Introduction

Generating descriptions of images has been an important task in computer vision. Compared to other forms of semantic summary, *e.g.* object tagging, linguistic descriptions are often richer, more comprehensive, and a more natural way to convey image content. Along with the recent surge of deep learning technologies, there has been remarkable progress in image captioning over the past few years [11, 29–32]. Latest studies on this topic often adopt a combination of an LSTM or its variant and a CNN. The former is to produce the word sequences while the latter is to capture the visual features of the images.

The advance in image captioning has been marked as a

Figure 1: This figure shows two images with descriptions generated by humans, an LSTM net trained with our GAN-based framework (*G-GAN*), and an LSTM net trained with MLE (*G-MLE*). The last two columns compare the metric values of BLEU-3 and *E-GAN*, the evaluator trained using our method. As we can see, the sentences generated by *G-GAN* are more natural and demonstrate higher variability, as compared to those by *G-MLE*. Also, the *E-GAN* metrics are more consistent with human's evaluations, while BLEU only favors those that significantly overlap with the training samples in detailed wording.

prominent success of AI[1]. It has been reported [29, 30] that with certain metrics, like BLEU [23] or CIDEr [28], state-of-the-art techniques have already surpassed human's performance. A natural question to ask is then: *has the problem of generating image descriptions been solved?* Let us take a step back, and look at a sample of the current results. Figure 1 shows two vivid scenes together with three sentences produced by the Encoder-and-Decoder model [29] (marked as "G-MLE"), a state-of-the-art caption generator. Though faithfully describing the content of the images, these sentences feel rigid, dry, and lacking in vitality.

This is not surprising. Our brief survey (see Section 2)

---

[1]ARTIFICIAL INTELLIGENCE AND LIFE IN 2030, https://ai100.stanford.edu/2016-report

shows that existing efforts primarily focus on *fidelity*, while other essential qualities of human languages, *e.g.* *naturalness* and *diversity*, have received less attention. More specifically, mainstream captioning models, including those based on LSTMs [8], are mostly trained with the (conditional) maximum likelihood objective. This objective encourages the use of the *n-grams* that appeared in the training samples. Consequently, the generated sentences will bear high resemblance to training sentences in *detailed wording*, with very limited variability in expression [4]. Moreover, conventional evaluation metrics, such as BLEU [23], METEOR [14], ROUGE [18], and CIDEr [28], tend to favor this *"safe"* but restricted way. Under these metrics, sentences that contain matched n-grams would get substantially higher scores than those using variant expressions [1]. This issue is manifested by the fact that human descriptions get considerably lower scores.

Motivated to move beyond these limitations, we explore an alternative approach in this work. We wish to produce sentences that possess three properties: (1) **Fidelity**: the generated descriptions should reflect the visual content faithfully. Note that we desire the fidelity in *semantics* instead of *wording*. (2) **Naturalness**: the sentences should *feel* like what real people would say when presented with the image. In other words, when these sentences are shown to a real person, she/he would ideally not be able to tell that they are machine-generated. (3) **Diversity**: the generator should be able to produce notably different expressions given an image – just like human beings, different people would describe an image in different ways.

Towards this goal, we develop a new framework on top of the Conditional GAN [22]. GAN has been successfully used in image generation. As reported in previous works [9, 24], they can produce *natural* images nearly indistinguishable from real photos, freely or constrained by conditions. This work studies a different task for the GAN method, namely, generating *natural* descriptions conditioned on a given image. To our best knowledge, this is the first time the GAN method is used for image description.

Applying GANs to text generation is nontrivial. It comes with two significant challenges due to the special nature of linguistic representation. First, in contrast to image generation, where the transformation from the input random vector to the produced image is a deterministic continuous mapping, the process of generating a linguistic description is a *sequential sampling* procedure, which samples a *discrete* token at each step. Such operations are *non-differentiable*, making it difficult to apply back-propagation directly. We tackle this issue via *Policy Gradient*, a classical method originating from reinforcement learning [27]. The basic idea is to consider the production of each word as an *action*, for which the reward comes from the evaluator. By approximating the stochastic policy with a parametric function

approximator, we allow gradients to be back-propagated.

Second, in the conventional GAN setting, the generator would receive feedback from the evaluator when an entire sample is produced. For sequence generation, this would lead to several difficulties in training, including *vanishing gradients* and *error propagation*. To mitigate such difficulties, we devise a mechanism that allows the generator to get early feedback. Particularly, when a description is *partly* generated, our framework would calculate an approximated *expected future reward* through Monte Carlo rollouts [34]. Empirically, we found that this significantly improves the efficiency and stability of the training process.

Overall, our contributions can be briefly summarized as follows: (1) We explore an alternative approach to generate image descriptions, which, unlike most of the previous work, encourages not only *fidelity* but also *naturalness* and *diversity*. (2) From a technical standpoint, our approach relies on the conditional GAN method to learn the generator, instead of using MLE, a paradigm widely adopted in state-of-the-art methods. (3) Our framework not only results in a generator that can produce natural and diverse expressions, but also yields a description evaluator at the same time, which, as we will show in our experiments, is substantially more consistent with human evaluation.

2. Related Work

Generation. Generating descriptions for images has been a long standing topic in computer vision. Early studies mostly adopted *detection-based* approaches. Such methods first detect visual concepts (*e.g.* object categories, relationships, and attributes) using CRFs [2, 6, 12], SVMs [16], or CNNs [5, 17], then generate descriptions thereon using simple methods, such as sentence templates [12, 16], or by retrieving relevant sentences from existing data [5, 6, 13, 15].

In recent years, the Encoder-and-Decoder paradigm proposed in [29] became increasingly popular. Many state-of-the-art frameworks [21, 29–32, 35] for this task adopt the *maximum likelihood* principle for learning. Such a framework usually works as follows. Given an image I, it first derives a feature representation $\mathbf{f}(I)$, and then generates the words $w_1, \ldots, w_T$ sequentially, following a Markov process conditioned on $\mathbf{f}(I)$. The model parameters are learned via maximum likelihood estimation (MLE), *i.e.* maximizing the conditional log-likelihood of the training samples, as:

$$\sum_{(I_i, S_i) \sim \mathcal{D}} \sum_{t=0}^{T_i} \log p\left(w_i^{(t)} | \mathbf{f}(I), w_i^{(t-1)}, \ldots, w_i^{(t-n)}\right) \quad (1)$$

Here, I_i and $S_i = (w_i^{(0)}, \ldots, w_i^{(T_i)})$ are the image and the corresponding descriptive sentence of the i-th sample, and n is the order of the Markov chain – the distribution of the current word depends on n preceding words. Along
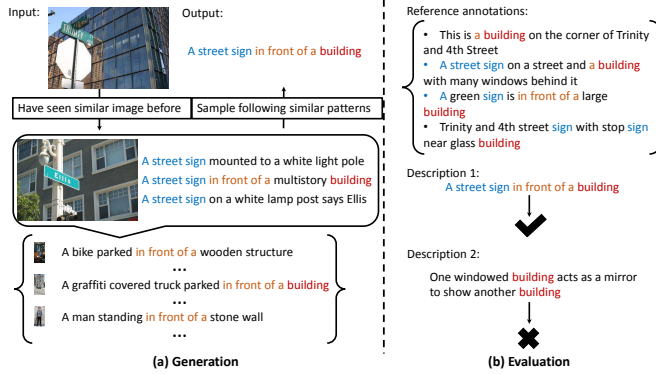
Figure 2: We illustrate the procedures of image description generation and evaluation for state-of-the-art approaches. While the generation procedure tends to follow observed patterns, the evaluation procedure also favors this point. Best viewed in color.

Figure 3: Examples of images with two semantically similar descriptions, selected from ground-truth annotations. While existing metrics assign higher scores to those with more matched n-grams, *E-GAN* gives scores consistent with human evaluation.

with the popularity of deep neural networks, latest studies often adopt neural networks for both image representation and language modeling. For example, [30] uses a CNN for deriving the visual features $\mathbf{f}(I)$, and an LSTM [8] net to express the sequential relations among words. Despite the evolution of the modeling choices, the maximum likelihood principle remains the predominant learning principle.

As illustrated in Figure 2, when similar images are presented, the sentences generated by such a model often contain *repeated* patterns [3]. This is not surprising – under the MLE principle, the joint probability of a sentence is, to a large extent, determined by whether it contains the frequent n-grams from the training set. Therefore, the model trained in this way will tend to produce such n-grams. In particular, when the generator yields a few of words that match the prefix of a frequent n-gram, the remaining words of that n-gram will likely be produced following the Markov chain.

Evaluation. Along with the development of the generation methods, various evaluation metrics have been proposed to assess the quality of the generated sentences. Classical metrics include BLEU [23] and ROUGE [18], which respectively focuses on the precision and recall of n-grams. Beyond them, METEOR [14] uses a combination of both the precision and the recall of n-grams. CIDEr [28] uses weighted statistics over n-grams. As we can see, such metrics mostly rely on matching n-grams with the *"ground-truths"*. As a result, sentences that contain frequent n-grams will get higher scores as compared to those using variant expressions, as shown in Figure 3. Recently, a new metric SPICE [1] was proposed. Instead of matching between n-grams, it focuses on those linguistic entities that reflect visual concepts (*e.g.* objects and relationships). However, other qualities, *e.g.* the naturalness of the expressions, are not considered in this metric.

Our Alternative Way. Previous approaches, including both generation methods and evaluation metrics, primarily focus on the *resemblance* to the training samples. While this is a *safe* way to generate plausible descriptions, it is *limited*. For example, when presented an image, different people would probably give different descriptions that do not overlap much in the wording patterns. This diversity in expression is an essential property of human languages, which, however, is often overlooked in previous works (both generation and evaluation). In this work, we explore an alternative approach – instead of emphasizing n-gram matching, we aim to improve the *naturalness* and *diversity*, *i.e.* generating sentences that feel like what real people would say, rather than focusing on word-by-word matching. Specifically, our approach jointly trains a generator G and an evaluator E in an adversarial way, where G is to produce natural descriptions, while E is to distinguish irrelevant or artificial descriptions from natural ones.

From a technical standpoint, our approach is based on the conditional GAN approach. GANs [7] and conditional GANs [22] are popular formulations for learning generators. For computer vision, GAN was originally introduced to generate images [24]. In a recent work [34], a text generator based on the GAN method was proposed. Note that this is an unconstrained generator that does not take into account any conditions. Hence, it can not be directly used for generating descriptions for images – in this task, the relevance of the generated text to the given image is essential. To our best knowledge, this is the first study that explores the use of *conditional* GAN in generating image descriptions.

3. Framework

We propose a new framework for generating image descriptions based on the conditional GAN [22] method, which consists of a generator G, and an evaluator E. Given
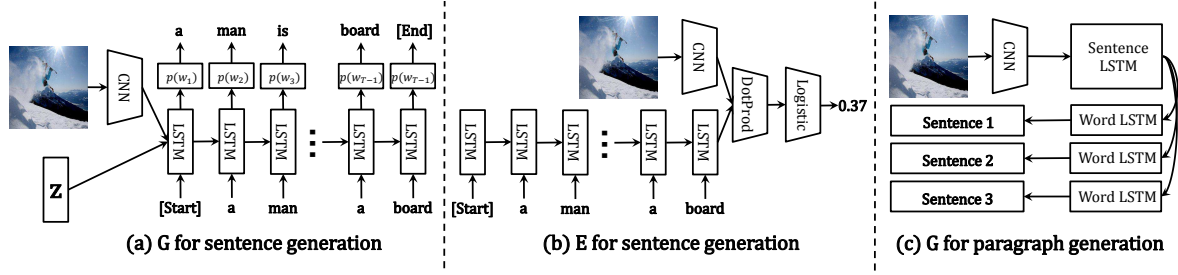
Figure 4: The structures of the generator G for both single sentences and paragraphs, and the evaluator E for single sentences.

an image I, the former is for generating *natural* and *semantically relevant* descriptions; while the latter is for evaluating how well a sentence or paragraph describes I. We start with generating single sentences as descriptions, and then extend our framework to paragraph generation.

3.1. Overall Formulation

Our framework contains a *generator* G and a *evaluator* E, whose structures are respectively shown in Figure 4 (a) and (b). It is worth noting that our framework is orthogonal to works that focus on architectural designs of the G and the E. Their structures are not restricted to the ones introduced in this paper. In our framework, given an image I, the generator G takes two inputs: an image feature $\mathbf{f}(I)$ derived from a convolutional neural network (CNN) and a random vector $\mathbf{z}$. In particular, we follow the setting in Neu-ralTalk2[2], adopting *VGG16* [26] as the CNN architecture. The random vector $\mathbf{z}$ allows the generator to produce different descriptions given an image. One can control the *diversity* by tuning the variance of $\mathbf{z}$. With both $\mathbf{f}(I)$ and $\mathbf{z}$ as the initial conditions, the generator relies on an LSTM [8] net as a decoder, which generates a sentence, word by word. Particularly, the LSTM net assumes a sequence of latent states $(s_0, s_1, \dots)$. At each step t, a word w_t is drawn from the conditional distribution $p(w|s_t)$.

The evaluator E is also a neural network, with an architecture similar to G but operating in a different way. Given an image I and a descriptive sentence $S = (w_0, w_1, \dots)$, it embeds them into vectors $\mathbf{f}(I)$ and $\mathbf{h}(S)$ of the same dimension, respectively via a CNN and an LSTM net. Then the *quality* of the description, *i.e.* how well it describes I, is measured by the dot product of the embedded vectors, as

$$r_{\boldsymbol{\eta}}(I, S) = \sigma\left(\langle \mathbf{f}(I, \boldsymbol{\eta}_I), \mathbf{h}(S, \boldsymbol{\eta}_S)\rangle\right). \quad (2)$$

Here, $\boldsymbol{\eta} = (\boldsymbol{\eta}_I, \boldsymbol{\eta}_S)$ denotes the evaluator parameters, and σ is a logistic function that turns the dot product into a probability value in $[0, 1]$. Note that while the CNN and the LSTM net in E have the same structure as those in G, their parameters are not tied with each other.

For this framework, the learning objective of G is to generate descriptions that are *natural*, *i.e.* indistinguishable from what humans would say when presented with the same image; while the objective of E is to distinguish between artifical descriptions (*i.e.* those from G) and the real ones (*i.e.* those from the training set). This can be formalized into a minimax problem as follows:

$$\min_{\boldsymbol{\theta}} \max_{\boldsymbol{\eta}} \mathcal{L}(G_{\boldsymbol{\theta}}, E_{\boldsymbol{\eta}}). \quad (3)$$

Here, $G_{\boldsymbol{\theta}}$ and $E_{\boldsymbol{\eta}}$ are a generator with parameter $\boldsymbol{\theta}$ and an evaluator with parameter $\boldsymbol{\eta}$. The objective function $\mathcal{L}$ is:

$$\mathbb{E}_{S \sim \mathcal{P}_I} \left[\log r_{\boldsymbol{\eta}}(I, S)\right] + \mathbb{E}_{\mathbf{z} \sim \mathcal{N}_0} \left[\log(1 - r_{\boldsymbol{\eta}}(I, G_{\boldsymbol{\theta}}(I, \mathbf{z})))\right]. \quad (4)$$

Here, $\mathcal{P}_I$ denotes the descriptive sentences for I provided in the training set, $\mathcal{N}_0$ denotes a standard normal distribution, and $G_{\boldsymbol{\theta}}(I, \mathbf{z})$ denotes the sentence generated with I and $\mathbf{z}$. The overall learning procedure alternates between the updating of G and E, until they reach an equilibrium.

This formulation reflects an essentially different philosophy in *how to train a description generator* as opposed to those based on MLE. As mentioned, our approach aims at the *semantical relevance* and *naturalness*, *i.e.* whether the generated descriptions feel like what human would say, while the latter focuses more on word-by-word patterns.

3.2. Training G: Policy Gradient & Early Feedback

As mentioned, unlike in conventional GAN settings, the production of sentences is a discrete sampling process, which is *nondifferentiable*. A question thus naturally arises - how can we *back-propagate the feedback* from E under such a formulation? We tackle this issue via *Policy Gradient* [27], a technique originating from reinforcement learning. The basic idea is to consider a sentence as a sequence of *actions*, where each word w_t is an action. The choices of such "actions" are governed by a *policy* $\boldsymbol{\pi_\theta}$.

With this interpretation, the generative procedure works as follows. It begins with an empty sentence, denoted by $S_{1:0}$, as the initial state. At each step t, the *policy* $\boldsymbol{\pi_\theta}$ takes the conditions $\mathbf{f}(I)$, $\mathbf{z}$, and the preceding words $S_{1:t-1}$ as inputs, and yields a conditional distribution $\boldsymbol{\pi_\theta}(w_t|\mathbf{f}(I), \mathbf{z}, S_{1:t-1})$ over the extended vocabulary,

namely all words plus an indicator of sentence end, denoted by e. This computation is done by moving forward along the LSTM net by one step. From this conditional distribution, an action w_t will be sampled. If $w_t = e$, the sentence will be terminated, otherwise w_t will be appended to the end. The *reward* of this sequence of actions S is $r_{\boldsymbol{\eta}}(I, S)$, the score given by the evaluator E.

Now, we have defined an action space, a policy, and a reward function, and it seems that we are ready to apply the reinforcement learning method. However, there is a serious issue here – a sentence can only be evaluated when it is *completely* generated. In other words, we can only see the reward at the end. We found empirically that this would lead to a number of practical difficulties, *e.g.* gradients vanishing along a long chain and overly slow convergence in training.

We address this issue through *early feedback*. To be more specific, we evaluate an *expected future reward* as defined below when the sentence is *partially* generated:

$$V_{\boldsymbol{\theta},\boldsymbol{\eta}}(I, \mathbf{z}, S_{1:t}) = \mathbb{E}_{S_{t+1:T} \sim G_{\boldsymbol{\theta}}(I, \mathbf{z})}[r_{\boldsymbol{\eta}}(I, S_{1:t} \oplus S_{t+1:T})]. \tag{5}$$

where $\oplus$ represents the concatenation operation. Here, the expectation can be approximated using Monte Carlo rollouts [34]. Particularly, when we have a part of the sentence $S_{1:t}$, we can continue to sample the remaining words by simulating the LSTM net until it sees an end indicator e. Conducting this conditional simulation for n times would result in n sentences. We can use the evaluation score averaged over these simulated sentences to approximate the *expected future reward*. To learn the generator $G_{\boldsymbol{\theta}}$, we use maximizing this expected reward $V_{\boldsymbol{\theta},\boldsymbol{\eta}}$ as the learning objective. Following the argument in [27], we can derive the gradient of this objective w.r.t. $\boldsymbol{\theta}$ as:

$$\tilde{\mathbb{E}} \left[\sum_{t=1}^{T_{\max}} \sum_{w_t \in \mathcal{V}} \nabla_{\boldsymbol{\theta}} \pi_{\boldsymbol{\theta}}(w_t | I, \mathbf{z}, S_{1:t-1}) \cdot V_{\boldsymbol{\theta}',\boldsymbol{\psi}}(I, \mathbf{z}, S_{1:t} \oplus w_t) \right]. \tag{6}$$

Here, $\mathcal{V}$ is the vocabulary, $T_{\max}$ is the max length of a description, and $\tilde{\mathbb{E}}$ is the mean over all simulated sentences within a mini-batch. $\boldsymbol{\theta}'$ is a copy of the generator parameter $\boldsymbol{\theta}$ at the begining of the update procedure of the generator. During the procedure, the generator will be updated multiple times, and each update will use the same set of parameters ($\boldsymbol{\theta}'$) to compute Eq (5).

Overall, using policy gradients, we make the generator trainable with gradient descent. Using expected future reward, we can provide early feedback to the generator along the way, thus substantially improving the effectiveness of the training process. Note that policy gradients have also been used in image description generation in [20,25]. These works, however, adopt conventional metrics, *e.g.* BLEU and CIDEr as rewards, instead of relying on GAN. Hence, their technical frameworks are fundamentally different.

3.3. Training E: Naturalness & Relevance

The primary purpose of E is to determine how well a description S describes a given image I. A good description needs to satisfy two criteria: *natural* and *semantically relevant*. To enforce both criteria, inspired by [24] we extend Eq (4) to consider three types of descriptions for each training image I: (1) $\mathcal{S}_I$: the set of descriptions for I provided by human, (2) $\mathcal{S}_G$: those from the generator $G_{\boldsymbol{\theta}}$, and (3) $\mathcal{S}_{\backslash I}$: the human descriptions for different images, which is uniformly sampled from all descriptions that are not associated with the given image I. To increase the scores for the descriptions in $\mathcal{S}_I$ while suppressing those in the others, we use a joint objective formulated as:

$$\max_{\boldsymbol{\eta}} \quad \mathcal{L}_E(\boldsymbol{\eta}) = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}_E(I_i; \boldsymbol{\eta}). \tag{7}$$

Here, N is the number of training images. The term for each image I_i is given by:

$$\begin{aligned} \mathcal{L}_E(I; \boldsymbol{\eta}) = \ & \mathbb{E}_{S \in \mathcal{S}_I} \log r_{\boldsymbol{\eta}}(I, S) \\ & + \alpha \cdot \mathbb{E}_{S \in \mathcal{S}_G} \log(1 - r_{\boldsymbol{\eta}}(I, S)) \\ & + \beta \cdot \mathbb{E}_{S \in \mathcal{S}_{\backslash I}} \log(1 - r_{\boldsymbol{\eta}}(I, S)). \end{aligned} \tag{8}$$

The second term forces the evaluator to distinguish between the human descriptions and the generated ones, which would in turn provide useful feedbacks to $G_{\boldsymbol{\theta}}$, pushing it to generate more *natural* descriptions. The third term, on the other hand, ensures the *semantic relevance*, by explicitly suppressing mismatched descriptions. The coefficients α and β are to balance the contributions of these terms, whose values are empirically determined on the validation set.

3.4. Extensions for Generating Paragraphs

We also extend our framework to generate *descriptive paragraphs* by adopting a *Hierarchical LSTM* design. Specifically, our extended design is inspired by [11]. As shown in part (c) of Figure 4, it comprises two LSTM levels – a *sentence-level* LSTM net and a *word-level* LSTM net. Given the conditions $\mathbf{f}(I)$ and $\mathbf{z}$, to produce a paragraph, it first generates a sequence of vectors based on $\mathbf{f}(I)$, each encoding the topics of a sentence. Then for each sentence, it generates the words conditioned on the corresponding topic and the random vector $\mathbf{z}$.

For evaluating a paragraph, the evaluator E also adopts a hierarchical design, but reversing the steps. Given an image I and a paragraph P, it first embeds each sentence into a vector via a word-level LSTM net, and then embeds the entire paragraph by combining the sentence embeddings via a sentence-level LSTM net. Finally, it computes the score by taking the dot product between the paragraph embedding $\mathbf{p}$ and the image representation $\mathbf{f}(I)$, and turning it into a probability as $\sigma(\mathbf{p}^T \mathbf{f}(I))$, where σ is the logistic function.
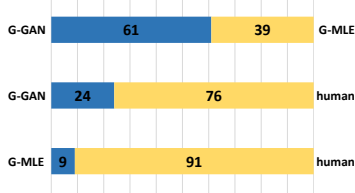
Figure 5: The figure shows the human comparison results between each pair of generators. With names of the generators placed at each side of the comparison, the blue and orange areas respectively indicate percentages of the generator in the left and right being the better one.

		R@1	R@3	R@5	R@10
S	G-MLE	5.06	12.28	18.24	29.30
	G-GAN	**14.30**	**30.88**	**40.06**	**55.82**
P	G-MLE	9.88	20.12	27.30	39.94
	G-GAN	**12.04**	**23.88**	**30.70**	**41.78**

Table 2: The recalls of image rankings for different generators. Here recalls is the ratio of the original image being in the top-k in the ranked lists. The ranks are based on the similarities (S) between a image and a description, estimated by *E-GAN*, as well as the log-likelihoods (P), computed by different generators.

After pretraining, we fix the sentence-level LSTM net of G and only update the word-level LSTM net of G during the CGAN learning procedure. This can effectively reduce the cost of Monte Carlo rollouts. With a fixed sentence-level LSTM net, the policy gradients for each sentence will be computed separately, following the steps in Sec 3.2. Other parts of the training procedure remain the same.

4. Experiment

Datasets   We conducted experiments to test the proposed framework on two datasets: (1) *MSCOCO* [19], which contains $82,081$ training images and $40,137$ validation images. (2) *Flickr30k* [33], which contains $31,783$ images in total. We followed the split in [10], which has $1,000$ images for validation, $1,000$ for testing, and the rest for training. In both datasets, each image has at least 5 ground truth sentences. Note that our experiments involve comparison between human descriptions and model-generated ones. As we have no access to the ground-truth annotations of the testing images in MSCOCO, for this dataset, we use the training set for both training and validation, and the validation set for testing the performance.

Experimental settings   To process the annotations in each dataset, we follow [10] to remove non-alphabet characters, convert all remaining characters to lower-case, and replace all the words that appeared less than 5 times with a special word *UNK*. As a result, we get a vocabulary of size $9,567$ on MSCOCO, and a vocabulary of size $7,000$ on Flickr30k. All sentences are truncated to contain at most

16 words during training. We respectively pretrain G using standard MLE [29], for 20 epochs, and E with supervised training based on Eq (8), for 5 epoches. Subsequently, G and E are jointly trained, where each iteration consists of one step of G-update followed by one step of E-update. We set the mini-batch size to 64, the learning rate to 0.0001, and $n = 16$ in Monte Carlo rollouts. When testing, we use beam search based on the expected rewards from E-GAN, instead of the log-likelihoods, which we found empirically leads to better results.

Models   We compare three methods for sentence generation: (1)**Human**: a sentence randomly sampled from ground-truth annotations of each image is used as the output of this method. Other human-provided sentences will be used as the references for metric evaluation. This baseline is tested for the purpose of comparing human-provided and model-generated descriptions. (2)**G-MLE**: a generator trained based on MLE [29] is used to produce the descriptions. This baseline represents the state-of-the-art of mainstream methods. (3)**G-GAN**: the same generator trained by our framework proposed in this paper, which is based on the conditional GAN formulations.

For both *G-MLE* and *G-GAN*, *VGG16* [26] is used as the image encoders. Activations at the `fc7` layer, which are of dimension 4096, are used as the image features and fed to the description generators. Note that *G-GAN* also takes a random vector **z** as input. Here, **z** is a 1024-dimensional vector, whose entries are sampled from a standard normal distribution.

Evaluation metrics   We consider multiple evaluation metrics, including six conventional metrics BLEU-3 and BLEU-4 [23], METEOR [14], ROUGE_L [18], CIDEr [28], SPICE [1], and two additional metrics relevant to our formulation: *E-NGAN* and *E-GAN*. Particularly, *E-GAN* refers to the evaluator trained using our framework, *E-NGAN* refers to the evaluator trained according to Eq (8) without updating the generator alternatively. In other words, it is trained to distinguish between human-provided sentences and those generated by an MLE-based model.

Table 1 lists the performances of different generators under these metrics. On both datasets, the sentences produced by *G-MLE* receive considerably higher scores than those provided by human, on nearly all conventional metrics. This is not surprising. As discussed earlier, such metrics primarily focus on n-gram matching w.r.t. the references, while ignoring other important properties, *e.g.* naturalness and diversity. These results also clearly suggest that these metrics may not be particularly suited when evaluating the overall quality of the generated sentences. On the contrary, *E-GAN* regards *Human* as the best generator, while *E-NGAN* regards *G-GAN* as the best one. These two metrics obviously take into account more than just n-gram matching.

		BLEU-3	BLEU-4	METEOR	ROUGE_L	CIDEr	SPICE	E-NGAN	E-GAN
COCO	human	0.290	0.192	0.240	0.465	0.849	**0.211**	0.527	**0.626**
	G-MLE	**0.393**	**0.299**	**0.248**	**0.527**	**1.020**	0.199	0.464	0.427
	G-GAN	0.305	0.207	0.224	0.475	0.795	0.182	**0.528**	0.602
Flickr	human	0.269	0.185	0.194	0.423	0.627	0.159	0.482	**0.464**
	G-MLE	**0.372**	**0.305**	**0.215**	**0.479**	**0.767**	**0.168**	0.465	0.439
	G-GAN	0.153	0.088	0.132	0.330	0.202	0.087	**0.582**	0.456

Table 1: This table lists the performances of different generators on MSCOCO and Flickr30k. On BLEU-{3,4}, METEOR, ROUGE_L, CIDEr, and SPICE, *G-MLE* is shown to be the best among all generators, surpassing human by a significant margin. While *E-NGAN* regard *G-GAN* as the best generator, *E-GAN* regard *human* as the best one.

z_1	a baseball player holds a bat up to hit the ball	a man riding a snowboard down a slope	a group of people sitting around a table having a meal in a restaurant	a group of men dressed in suits posing for a photo
z_2	a baseball player holding white bat and wear blue baseball uniform	a person standing on a snowboard sliding down a hill	a young man sitting at a table with coffee and a lot of food	a couple of men standing next to each other wearing glasses
z_3	a professional baseball player holds up his bat as he watches	a man is jumping over a snow covered hill	a pretty young man sitting next to two men in lots of people	some people dressed in costume and cups

Figure 6: This figure shows example images with descriptions generated by *G-GAN* with different **z**.

User study & qualitative comparison To fairly evaluate the quality of the generated sentences as well as how *consistent* the metrics are with human's perspective, we conducted a user study. Specifically, we invited 30 human evaluators to compare the outputs of different generators. Each time, a human evaluator would be presented an image with two sentences from different methods and asked to choose the better one. Totally, we collected about 3,000 responses.

The comparative results are shown in Figure 5: From human's views, *G-GAN* is better than *G-MLE* in 61% of all cases. In the comparison between human and models, *G-MLE* only won in 9% of the cases, while *G-GAN* won in over 24%. These results clearly suggest that the sentences produced by *G-GAN* are of considerably higher quality, *i.e.* being more natural and semantically relevant. The examples in Figure 7 also confirm this assessment. Particularly, we can see when *G-MLE* is presented with similar images, it tends to generate descriptions that are almost the same. On the contrary, *G-GAN* describes them with more distinctive and diverse ones. We also varied **z** to study the capability of *G-GAN* in giving *diverse* descriptions while maintaining the semantical relatedness. The qualitative results are listed in Figure 6.

For the evaluation metrics, the assessments provided by *E-GAN* are the most consistent with human's evaluation, where the Kendall's rank correlation coefficient between *E-GAN* and *HE* is 0.14, while that for CIDEr and SPICE are -0.30 and -0.25. Also note that *E-GAN* yields a larger numerical gap between scores of human and those of other generators as compared to *E-NGAN*, which suggests that adversarial training can improve the discriminative power of the evaluator.

Evaluation by retrieval To compare the *semantic relevance*, we conducted an experiment using generated descriptions for retrieval. Specifically, we randomly select 5,000 images from the MSCOCO validation set; and for each image, we use the generated description as a query, ranking all 5,000 images according to the similarities between the images and the descriptions, computed by *E-GAN*, as well as the log-likelihoods. Finally, we compute the recall of the original image that appeared in the top-k ranks. The results for $k = 1, 3, 5, 10$ are listed in Table 2, where *G-GAN* is shown to provide more discriminative descriptions, outperforming *G-MLE* by a large margin across all cases.

Failure Analysis We analyzed failure cases and found that a major kind of errors is the inclusion of incorrect details. *e.g.* colors (red/yellow hat), and counts (three/four people). A possible cause is that there are only a few samples for each particular detail, and they are not enough to make the generator capture these details reliably. Also, the focus on diversity and overall quality may also encourage the generator to include more details, with the risk of some details being incorrect.

G-MLE	a man flying through the air while riding a snowboard	a man flying through the air while riding a snowboard	a man flying through the air while riding a snowboard	a man flying through the air while riding a skateboard
G-GAN	a man on a skateboard in a snowy park	a man skiing down the slope near a mountain	a man performing a grind trick on a skateboard ramp	a man with stunts on his skis in the snow
G-MLE	a group of people standing around a boat	a group of people sitting around a table	a group of people sitting at a table	a group of people sitting around a living room
G-GAN	the bench is sitting on the ground by the water	a group of people watching each other	a table with a lot of stuff on it	furnished living room with furniture and built area

Figure 7: This figure lists some images and corresponding descriptions generated by *G-GAN* and *G-MLE*. *G-MLE* tends to generate similar descriptions for similar images, while *G-GAN* generates better distinguishable descriptions for them.

	human	G-GAN, z_1	G-GAN, z_2	G-MLE
	people are on motorcycles. there are green cars behind them. the signs are all brown with chinese written on it.	men are riding on a motorcycle. the man is wearing tan boots, and a white and blue jacket with beige stripes on. the street is made of cobblestone. there are tall bright green trees on the sidewalk.	two people are riding motorcycles. there are many trees on the sidewalk. there is a red and white painted letter on the side of the ledge. tall buildings are on the background.	a man is riding a bike. there are trees on the sidewalk. there are people walking on the sidewalk. there is a tall building in the background.
	A baseball player is swinging a bat. He is wearing a black helmet and a black and white uniform. A catcher is behind him wearing a gray uniform. The catcher has a brown glove on his hand. Two men can be seen standing behind a green fence.	a baseball player in a white and blue uniform is holding a white bat. there is a umpire behind the batter in the blue and white uniform. he is getting ready to catch the ball. there is a crowd of people behind him watching him.	men are on a baseball field on a sunny day. the player is wearing a black and white uniform. there is a catcher behind him. the field is green with brown dirt and white shiny lines.	a baseball player is standing on a baseball field. he is wearing a blue helmet on his head. the catcher is wearing a black and gray uniform. the court is green with white lines.

Figure 8: Examples of images with different descriptive paragraphs generated by a human, *G-GAN* with different **z**, and *G-MLE*.

Paragraph Generation We also tested our framework on paragraph generation (See Sec 3.4). We use the dataset provided by [11], which contains $14,575$ training images, $2,487$ validation images, and $2,489$ testing images. Example results are shown in Figure 8. Again, we found that G-GAN can produce diverse and more natural descriptions as compared to G-MLE, which tends to follow similar patterns across sentences.

5. Conclusion

This paper presented an alternative approach to generating image descriptions. Compared to existing methods, which are mostly focused on the match of detailed wording, our approach, instead, aims to improve the overall quality, which involves *semantic relevance*, *naturalness*, and *diversity*. Some of these properties are often overlooked in pre-

vious efforts. We proposed a formulation based on conditional GAN that jointly trains a generator G and an evaluator E, and applied Policy Gradient and early feedbacks to tackle the technical challenges in end-to-end training. On both MSCOCO and Flickr30k, the proposed method produced descriptions that are more natural, diverse, and semantically relevant as compared to a state-of-the-art MLE-based model. This is clearly demonstrated in our user studies, qualitative examples, and retrieval applications. Our framework also provides an evaluator that is more consistent with human's evaluation.

References

[1] P. Anderson, B. Fernando, M. Johnson, and S. Gould. Spice: Semantic propositional image caption evaluation. In *European Conference on Computer Vision*, pages 382–398. Springer, 2016. 2, 3, 6

[2] B. Dai, Y. Zhang, and D. Lin. Detecting visual relationships with deep relational networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2

[3] J. Devlin, H. Cheng, H. Fang, S. Gupta, L. Deng, X. He, G. Zweig, and M. Mitchell. Language models for image captioning: The quirks and what works. *arXiv preprint arXiv:1505.01809*, 2015. 3

[4] J. Devlin, S. Gupta, R. Girshick, M. Mitchell, and C. L. Zitnick. Exploring nearest neighbor approaches for image captioning. *arXiv preprint arXiv:1505.04467*, 2015. 2

[5] H. Fang, S. Gupta, F. Iandola, R. K. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. C. Platt, et al. From captions to visual concepts and back. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1473–1482, 2015. 2

[6] A. Farhadi, M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth. Every picture tells a story: Generating sentences from images. In *European conference on computer vision*, pages 15–29. Springer, 2010. 2

[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, page 26722680, 2014. 3

[8] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 2, 3, 4

[9] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. *arXiv preprint arXiv:1611.07004*, 2016. 2

[10] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3128–3137, 2015. 6

[11] J. Krause, J. Johnson, R. Krishna, and L. Fei-Fei. A hierarchical approach for generating descriptive image paragraphs. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 5, 8

[12] G. Kulkarni, V. Premraj, V. Ordonez, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg. Babytalk: Understanding and generating simple image descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12):2891–2903, 2013. 2

[13] P. Kuznetsova, V. Ordonez, A. C. Berg, T. L. Berg, and Y. Choi. Collective generation of natural image descriptions. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 359–368. Association for Computational Linguistics, 2012. 2

[14] M. D. A. Lavie. Meteor universal: Language specific translation evaluation for any target language. *ACL 2014*, page 376, 2014. 2, 3, 6

[15] R. Lebret, P. O. Pinheiro, and R. Collobert. Simple image description generator via a linear phrase-based approach. *arXiv preprint arXiv:1412.8419*, 2014. 2

[16] S. Li, G. Kulkarni, T. L. Berg, A. C. Berg, and Y. Choi. Composing simple image descriptions using web-scale n-grams. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, pages 220–228. Association for Computational Linguistics, 2011. 2

[17] Y. Li, W. Ouyang, B. Zhou, K. Wang, and X. Wang. Scene graph generation from objects, phrases and region captions. In *ICCV*, 2017. 2

[18] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL-04 workshop*, volume 8. Barcelona, Spain, 2004. 2, 3, 6

[19] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pages 740–755. Springer, 2014. 6

[20] S. Liu, Z. Zhu, N. Ye, S. Guadarrama, and K. Murphy. Optimization of image description metrics using policy gradient methods. *arXiv preprint arXiv:1612.00370*, 2016. 5

[21] J. Lu, C. Xiong, D. Parikh, and R. Socher. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. *arXiv preprint arXiv:1612.01887*, 2016. 2

[22] M. Mirza and S. Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014. 2, 3

[23] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002. 1, 2, 3, 6

[24] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative adversarial text to image synthesis. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 3, 2016. 2, 3, 5

[25] S. J. Rennie, E. Marcheret, Y. Mroueh, J. Ross, and V. Goel. Self-critical sequence training for image captioning. *arXiv preprint arXiv:1612.00563*, 2016. 5

[26] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 4, 6

[27] R. S. Sutton, D. A. McAllester, S. P. Singh, Y. Mansour, et al. Policy gradient methods for reinforcement learning with function approximation. In *NIPS*, volume 99, pages 1057–1063, 1999. 2, 4, 5

[28] R. Vedantam, C. Lawrence Zitnick, and D. Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4566–4575, 2015. 1, 2, 3, 6

[29] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3156–3164, 2015. 1, 2, 6

[30] K. Xu, J. Ba, R. Kiros, K. Cho, A. C. Courville, R. Salakhutdinov, R. S. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, volume 14, pages 77–81, 2015. 1, 2, 3

[31] Z. Yang, Y. Yuan, Y. Wu, W. W. Cohen, and R. R. Salakhutdi-nov. Review networks for caption generation. In *Advances in Neural Information Processing Systems*, pages 2361–2369, 2016. 1, 2

[32] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo. Image caption-ing with semantic attention. In *Proceedings of the IEEE Con-ference on Computer Vision and Pattern Recognition*, pages 4651–4659, 2016. 1, 2

[33] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier. From im-age descriptions to visual denotations: New similarity met-rics for semantic inference over event descriptions. *Transac-tions of the Association for Computational Linguistics*, 2:67–78, 2014. 6

[34] L. Yu, W. Zhang, J. Wang, and Y. Yu. Seqgan: sequence gen-erative adversarial nets with policy gradient. *arXiv preprint arXiv:1609.05473*, 2016. 2, 3, 5

[35] L. Zhou, C. Xu, P. Koch, and J. J. Corso. Image caption generation with text-conditional semantic attention. *arXiv preprint arXiv:1606.04621*, 2016. 2