

# No Metrics Are Perfect: Adversarial Reward Learning for Visual Storytelling

Xin Wang\*, Wenhui Chen\*, Yuan-Fang Wang, William Yang Wang

University of California, Santa Barbara

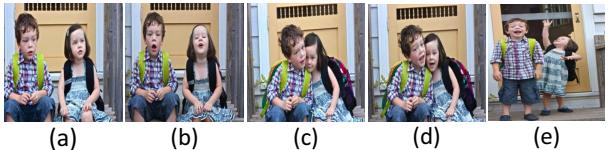
{xwang, wenhuchen, yfwang, william}@cs.ucsb.edu

## Abstract

Though impressive results have been achieved in visual captioning, the task of generating abstract stories from photo streams is still a little-tapped problem. Different from captions, stories have more expressive language styles and contain many imaginary concepts that do not appear in the images. Thus it poses challenges to behavioral cloning algorithms. Furthermore, due to the limitations of automatic metrics on evaluating story quality, reinforcement learning methods with hand-crafted rewards also face difficulties in gaining an overall performance boost. Therefore, we propose an Adversarial REward Learning (AREL) framework to learn an implicit reward function from human demonstrations, and then optimize policy search with the learned reward function. Though automatic evaluation indicates slight performance boost over state-of-the-art (SOTA) methods in cloning expert behaviors, human evaluation shows that our approach achieves significant improvement in generating more human-like stories than SOTA systems.<sup>1</sup>

## 1 Introduction

Recently, increasing attention has been focused on visual captioning (Chen et al., 2015, 2016; Xu et al., 2016; Wang et al., 2018c), which aims at describing the content of an image or a video. Though it has achieved impressive results, its capability of performing human-like understanding is still restrictive. To further investigate machine’s



### Captions:

- (a) A small boy and a girl are sitting together.
- (b) Two kids sitting on a porch with their backpacks on.
- (c) Two young kids with backpacks sitting on the porch.
- (d) Two young children that are very close to one another.
- (e) A boy and a girl smiling at the camera together.

**Story #1:** The brother and sister were ready for the first day of school. They were excited to go to their first day and meet new friends. They told their mom how happy they were. They said they were going to make a lot of new friends. Then they got up and got ready to get in the car.

**Story #2:** The brother did not want to talk to his sister. The siblings made up. They started to talk and smile. Their parents showed up. They were happy to see them.

Figure 1: An example of visual storytelling and visual captioning. Both captions and stories are shown here: each image is captioned with one sentence, and we also demonstrate two diversified stories that match the same image sequence.

capabilities in understanding more complicated visual scenarios and composing more structured expressions, visual storytelling (Huang et al., 2016) has been proposed. Visual captioning is aimed at depicting the concrete content of the images, and its expression style is rather simple. In contrast, visual storytelling goes one step further: it summarizes the idea of a photo stream and tells a story about it. Figure 1 shows an example of visual captioning and visual storytelling. We have observed that stories contain rich emotions (*excited, happy, not want*) and imagination (*siblings, parents, school, car*). It, therefore, requires the capability to associate with concepts that do not explicitly appear in the images. Moreover, stories are more subjective, so there barely exists standard

\* Equal contribution

<sup>1</sup>Code is released at <https://github.com/littlekobe/AREL>

templates for storytelling. As shown in Figure 1, the same photo stream can be paired with diverse stories, different from each other. This heavily increases the evaluation difficulty.

So far, prior work for visual storytelling (Huang et al., 2016; Yu et al., 2017b) is mainly inspired by the success of visual captioning. Nevertheless, because these methods are trained by maximizing the likelihood of the observed data pairs, they are restricted to generate simple and plain description with limited expressive patterns. In order to cope with the challenges and produce more human-like descriptions, Rennie et al. (2017) have proposed a reinforcement learning framework. However, in the scenario of visual storytelling, the common reinforced captioning methods are facing great challenges since the hand-crafted rewards based on string matches are either too biased or too sparse to drive the policy search. For instance, we used the METEOR (Banerjee and Lavie, 2005) score as the reward to reinforce our policy and found that though the METEOR score is significantly improved, the other scores are severely harmed. Here we showcase an adversarial example with an average METEOR score as high as 40.2:

*We had a great time to have a lot of the.  
They were to be a of the. They were to be in  
the. The and it were to be the. The, and it  
were to be the.*

Apparently, the machine is gaming the metrics. Conversely, when using some other metrics (e.g. BLEU, CIDEr) to evaluate the stories, we observe an opposite behavior: many relevant and coherent stories are receiving a very low score (nearly zero).

In order to resolve the strong bias brought by the hand-coded evaluation metrics in RL training and produce more human-like stories, we propose an Adversarial REward Learning (AREL) framework for visual storytelling. We draw our inspiration from recent progress in inverse reinforcement learning (Ho and Ermon, 2016; Finn et al., 2016; Fu et al., 2017) and propose the AREL algorithm to learn a more intelligent reward function. Specifically, we first incorporate a Boltzmann distribution to associate reward learning with distribution approximation, then design the adversarial process with two models – a **policy model** and a **reward model**. The policy model performs the primitive actions and produces the story sequence, while the reward model is responsible for learning

the implicit reward function from human demonstrations. The learned reward function would be employed to optimize the policy in return.

For evaluation, we conduct both automatic metrics and human evaluation but observe a poor correlation between them. Particularly, our method gains slight performance boost over the baseline systems on automatic metrics; human evaluation, however, indicates significant performance boost. Thus we further discuss the limitations of the metrics and validate the superiority of our AREL method in performing more intelligent understanding of the visual scenes and generating more human-like stories.

Our main contributions are four-fold:

- We propose an adversarial reward learning framework and apply it to boost visual story generation.
- We evaluate our approach on the Visual Storytelling (VIST) dataset and achieve the state-of-the-art results on automatic metrics.
- We empirically demonstrate that automatic metrics are not perfect for either training or evaluation.
- We design and perform a comprehensive human evaluation via Amazon Mechanical Turk, which demonstrates the superiority of the generated stories of our method on relevance, expressiveness, and concreteness.

## 2 Related Work

**Visual Storytelling** Visual storytelling is the task of generating a narrative story from a photo stream, which requires a deeper understanding of the event flow in the stream. Park and Kim (2015) has done some pioneering research on storytelling. Chen et al. (2017) proposed a multi-modal approach for storyline generation to produce a stream of entities instead of human-like descriptions. Recently, a more sophisticated dataset for visual storytelling (VIST) has been released to explore a more human-like understanding of grounded stories (Huang et al., 2016). Yu et al. (2017b) proposes a multi-task learning algorithm for both album summarization and paragraph generation, achieving the best results on the VIST dataset. But these methods are still based on behavioral cloning and lack the ability to generate more structured stories.

**Reinforcement Learning in Sequence Generation** Recently, reinforcement learning (RL) has gained its popularity in many sequence generation tasks such as machine translation (Bahdanau et al., 2016), visual captioning (Ren et al., 2017; Wang et al., 2018b), summarization (Paulus et al., 2017; Chen et al., 2018), etc. The common wisdom of using RL is to view generating a word as an action and aim at maximizing the expected return by optimizing its policy. As pointed in (Ranzato et al., 2015), traditional maximum likelihood algorithm is prone to exposure bias and label bias, while the RL agent exposes the generative model to its own distribution and thus can perform better. But these works usually utilize hand-crafted metric scores as the reward to optimize the model, which fails to learn more implicit semantics due to the limitations of automatic metrics.

**Rethinking Automatic Metrics** Automatic metrics, including BLEU (Papineni et al., 2002), CIDEr (Vedantam et al., 2015), METEOR (Banerjee and Lavie, 2005), and ROUGE (Lin, 2004), have been widely applied to the sequence generation tasks. Using automatic metrics can ensure rapid prototyping and testing new models with fewer expensive human evaluation. However, they have been criticized to be biased and correlate poorly with human judgments, especially in many generative tasks like response generation (Lowe et al., 2017; Liu et al., 2016), dialogue system (Bruni and Fernández, 2017) and machine translation (Callison-Burch et al., 2006). The naive overlap-counting methods are not able to reflect many semantic properties in natural language, such as coherence, expressiveness, etc.

**Generative Adversarial Network** Generative adversarial network (GAN) (Goodfellow et al., 2014) is a very popular approach for estimating intractable probabilities, which sidestep the difficulty by alternately training two models to play a min-max two-player game:

$$\min_D \max_G \mathbb{E}_{x \sim p_{data}} [\log D(x)] + \mathbb{E}_{z \sim p_z} [\log D(G(z))],$$

where  $G$  is the generator and  $D$  is the discriminator, and  $z$  is the latent variable. Recently, GAN has quickly been adopted to tackle discrete problems (Yu et al., 2017a; Dai et al., 2017; Wang et al., 2018a). The basic idea is to use Monte Carlo policy gradient estimation (Williams, 1992) to update the parameters of the generator.

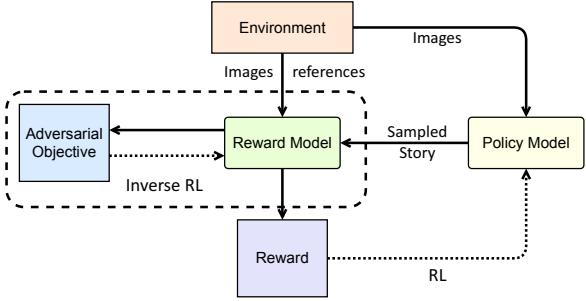


Figure 2: AREL framework for visual storytelling.

**Inverse Reinforcement Learning** Reinforcement learning is known to be hindered by the need for an extensive feature and reward engineering, especially under the unknown dynamics. Therefore, inverse reinforcement learning (IRL) has been proposed to infer expert’s reward function. Previous IRL approaches include maximum margin approaches (Abbeel and Ng, 2004; Ratliff et al., 2006) and probabilistic approaches (Ziebart, 2010; Ziebart et al., 2008). Recently, adversarial inverse reinforcement learning methods provide an efficient and scalable promise for automatic reward acquisition (Ho and Ermon, 2016; Finn et al., 2016; Fu et al., 2017; Henderson et al., 2017). These approaches utilize the connection between IRL and energy-based model and associate every data with a scalar energy value by using Boltzmann distribution  $p_\theta(x) \propto \exp(-E_\theta(x))$ . Inspired by these methods, we propose a practical AREL approach for visual storytelling to uncover a robust reward function from human demonstrations and thus help produce human-like stories.

### 3 Our Approach

#### 3.1 Problem Statement

Here we consider the task of visual storytelling, whose objective is to output a word sequence  $W = (w_1, w_1, \dots, w_T)$ ,  $w_t \in \mathbb{V}$  given an input image stream of 5 ordered images  $I = (I_1, I_2, \dots, I_5)$ , where  $\mathbb{V}$  is the vocabulary of all output token. We formulate the generation as a markov decision process and design a reinforcement learning framework to tackle it. As described in Figure 2, our AREL framework is mainly composed of two modules: a **policy model**  $\pi_\beta(W)$  and a **reward model**  $R_\theta(W)$ . The policy model takes an image sequence  $I$  as the input and performs sequential actions (choosing words  $w$  from the vocabulary  $\mathbb{V}$ ) to form a narrative story  $W$ . The reward model

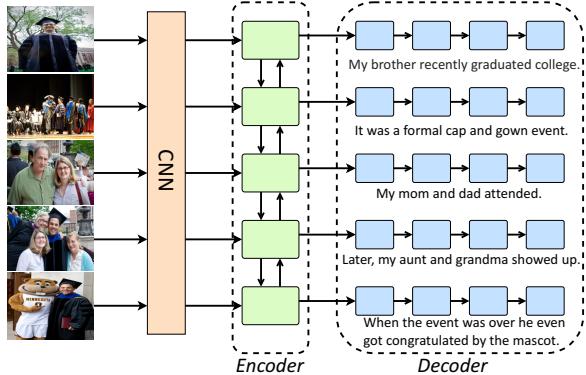


Figure 3: Overview of the policy model. The visual encoder is a bidirectional GRU, which encodes the high-level visual features extracted from the input images. Its outputs are then fed into the RNN decoders to generate sentences in parallel. Finally, we concatenate all the generated sentences as a full story. Note that the five decoders share the same weights.

is optimized by the adversarial objective (see Section 3.3) and aims at deriving a human-like reward from both human-annotated stories and sampled predictions.

### 3.2 Model

**Policy Model** As is shown in Figure 3, the policy model is a CNN-RNN architecture. We first feed the photo stream  $I = (I_1, \dots, I_5)$  into a pretrained CNN and extract their high-level image features. We then employ a visual encoder to further encode the image features as context vectors  $h_i = [\overleftarrow{h}_i; \overrightarrow{h}_i]$ . The visual encoder is a bidirectional gated recurrent units (GRU).

In the decoding stage, we feed each context vector  $h_i$  into a GRU-RNN decoder to generate a sub-story  $W_i$ . Formally, the generation process can be written as:

$$s_t^i = \text{GRU}(s_{t-1}^i, [w_{t-1}^i, h_i]), \quad (1)$$

$$\pi_\beta(w_t^i | w_{1:t-1}^i) = \text{softmax}(W_s s_t^i + b_s), \quad (2)$$

where  $s_t^i$  denotes the  $t$ -th hidden state of  $i$ -th decoder. We concatenate the previous token  $w_{t-1}^i$  and the context vector  $h_i$  as the input.  $W_s$  and  $b_s$  are the projection matrix and bias, which output a probability distribution over the whole vocabulary  $\mathbb{V}$ . Eventually, the final story  $W$  is the concatenation of the sub-stories  $W_i$ .  $\beta$  denotes all the parameters of the encoder, the decoder, and the output layer.

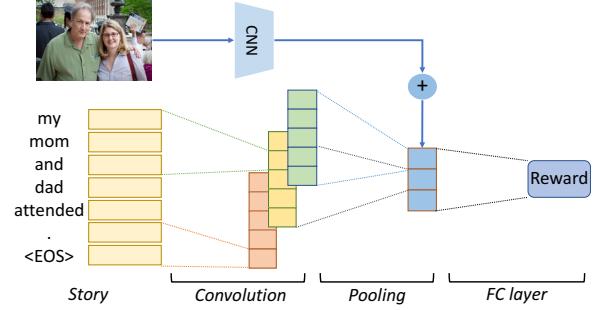


Figure 4: Overview of the reward model. Our reward model is a CNN-based architecture, which utilizes convolution kernels with size 2, 3 and 4 to extract bigram, trigram and 4-gram representations from the input sequence embeddings. Once the sentence representation is learned, it will be concatenated with the visual representation of the input image, and then be fed into the final FC layer to obtain the reward.

**Reward Model** The reward model  $R_\theta(W)$  is a CNN-based architecture (see Figure 4). Instead of giving an overall score for the whole story, we apply the reward model to different story parts (sub-stories)  $W_i$  and compute partial rewards, where  $i = 1, \dots, 5$ . We observe that the partial rewards are more fine-grained and can provide better guidance for the policy model.

We first query the word embeddings of the sub-story (one sentence in most cases). Next, multiple convolutional layers with different kernel sizes are used to extract the n-grams features, which are then projected into the sentence-level representation space by pooling layers (the design here is inspired by Kim (2014)). In addition to the textual features, evaluating the quality of a story should also consider the image features for relevance. Therefore, we then combine the sentence representation with the visual feature of the input image through concatenation and feed them into the final fully connected decision layer. In the end, the reward model outputs an estimated reward value  $R_\theta(W)$ . The process can be written in formula:

$$R_\theta(W) = \phi(W_r(f_{conv}(W) + W_i I_{CNN}) + b_r), \quad (3)$$

where  $\phi$  denotes the non-linear projection function,  $W_r, b_r$  denote the weight and bias in the output layer, and  $f_{conv}$  denotes the operations in CNN.  $I_{CNN}$  is the high-level visual feature extracted from the image, and  $W_i$  projects it into the

sentence representation space.  $\theta$  includes all the parameters above.

### 3.3 Learning

**Reward Boltzmann Distribution** In order to associate story distribution with reward function, we apply EBM to define a Reward Boltzmann distribution:

$$p_\theta(W) = \frac{\exp(R_\theta(W))}{Z_\theta}, \quad (4)$$

Where  $W$  is the word sequence of the story and  $p_\theta(W)$  is the approximate data distribution, and  $Z_\theta = \sum_W \exp(R_\theta(W))$  denotes the partition function. According to the energy-based model (Le-Cun et al., 2006), the optimal reward function  $R^*(W)$  is achieved when the Reward-Boltzmann distribution equals to the “real” data distribution  $p_\theta(W) = p^*(W)$ .

**Adversarial Reward Learning** We first introduce an empirical distribution  $p_e(W) = \frac{\mathbb{1}(W \in D)}{|D|}$  to represent the empirical distribution of the training data, where  $D$  denotes the dataset with  $|D|$  stories and  $\mathbb{1}$  denotes an indicator function. We use this empirical distribution as the “good” examples, which provides the evidence for the reward function to learn from.

In order to approximate the Reward Boltzmann distribution towards the “real” data distribution  $p^*(W)$ , we design a min-max two-player game, where the Reward Boltzmann distribution  $p_\theta$  aims at maximizing its similarity with empirical distribution  $p_e$  while minimizing that with the “faked” data generated from policy model  $\pi_\beta$ . On the contrary, the policy distribution  $\pi_\beta$  tries to maximize its similarity with the Boltzmann distribution  $p_\theta$ . Formally, the adversarial objective function is defined as

$$\max_{\beta} \min_{\theta} KL(p_e(W)||p_\theta(W)) - KL(\pi_\beta(W)||p_\theta(W)). \quad (5)$$

We further decompose it into two parts. First, because the objective  $J_\beta$  of the story generation policy is to maximize its similarity with the Boltzmann distribution  $p_\theta$ , the optimal policy that minimizes KL-divergence is thus  $\pi(W) \sim \exp(R_\theta(W))$ , meaning if  $R_\theta$  is optimal, the optimal  $\pi_\beta = \pi^*$ . In formula,

$$J_\beta = -KL(\pi_\beta(W)||p_\theta(W)) = \mathbb{E}_{W \sim \pi_\beta(W)} [R_\theta(W)] - \log Z_\theta + H(\pi_\beta(W)), \quad (6)$$

---

#### Algorithm 1 The AREL Algorithm.

---

```

1: for episode  $\leftarrow 1$  to N do
2:   collect story  $W$  by executing policy  $\pi_\theta$ 
3:   if Train-Reward then
4:      $\theta \leftarrow \theta - \eta \times \frac{\partial J_\theta}{\partial \theta}$  (see Equation 9)
5:   else if Train-Policy then
6:     collect story  $\tilde{W}$  from empirical  $p_e$ 
7:      $\beta \leftarrow \beta - \eta \times \frac{\partial J_\beta}{\partial \beta}$  (see Equation 9)
8:   end if
9: end for

```

---

where  $H$  denotes the entropy of the policy model. On the other hand, the objective  $J_\theta$  of the reward function is to distinguish between human-annotated stories and machine-generated stories. Hence it is trying to minimize the KL-divergence with the empirical distribution  $p_e$  and maximize the KL-divergence with the approximated policy distribution  $\pi_\beta$ :

$$\begin{aligned} J_\theta = & KL(p_e(W)||p_\theta(W)) - KL(\pi_\beta(W)||p_\theta(W)) \\ = & \sum_W [p_e(W)R_\theta(W) - \pi_\beta(W)R_\theta(W)] \\ & + \log Z_\theta - \log Z_\theta - H(p_e) + H(\pi_\beta), \end{aligned} \quad (7)$$

Since  $H(\pi_\beta)$  and  $H(p_e)$  are irrelevant to  $\theta$ , we denote them as constant  $C$ . It is also worth noting that with negative sampling in the optimization of the KL-divergence, the computation of the intractable partition function  $Z_\theta$  is bypassed. Therefore, the objective  $J_\theta$  can be further derived as

$$J_\theta = \mathbb{E}_{W \sim p_e(W)} [R_\theta(W)] - \mathbb{E}_{W \sim \pi_\beta(W)} [R_\theta(W)] + C. \quad (8)$$

Here we propose to use stochastic gradient descent to optimize these two models alternately. Formally, the gradients can be written as

$$\begin{aligned} \frac{\partial J_\theta}{\partial \theta} &= \mathbb{E}_{W \sim p_e(W)} \left[ \frac{\partial R_\theta(W)}{\partial \theta} \right] - \mathbb{E}_{W \sim \pi_\beta(W)} \left[ \frac{\partial R_\theta(W)}{\partial \theta} \right], \\ \frac{\partial J_\beta}{\partial \beta} &= \mathbb{E}_{W \sim \pi_\beta(W)} (R_\theta(W) - \log \pi_\beta(W) - b) \frac{\partial \log \pi_\beta(W)}{\partial \beta}, \end{aligned} \quad (9)$$

where  $b$  is the estimated baseline to reduce variance during REINFORCE training.

**Training & Testing** As described in Algorithm 1, we introduce an alternating algorithm to train these two models using stochastic gradient descent. During testing, the policy model is used with beam search to produce the story.

## 4 Experiments and Analysis

### 4.1 Experimental Setup

**VIST Dataset** The VIST dataset (Huang et al., 2016) is the first dataset for sequential vision-to-language tasks including visual storytelling, which consists of 10,117 Flickr albums with 210,819 unique photos. In this paper, we mainly evaluate our AREL method on this dataset. After filtering the broken images<sup>2</sup>, there are 40,098 training, 4,988 validation, and 5,050 testing samples. Each sample contains one story that describes 5 selected images from a photo album (mostly one sentence per image). And the same album is paired with 5 different stories as references. In our experiments, we used the same split settings as in (Huang et al., 2016; Yu et al., 2017b) for a fair comparison. During our experiments, we apply two kinds of non-linear functions  $\phi$  for the discriminator, namely SoftSign function ( $f(x) = \frac{x}{1+|x|}$ ) and Hyperbolic function ( $f(x) = \frac{\sinh x}{\cosh x}$ ). We found that unbounded non-linear functions like ReLU function (Glorot et al., 2011) will lead to severe vibrations and instabilities during training, therefore we resort to the bounded functions.

**Evaluation Metrics** In order to comprehensively evaluate our method on storytelling dataset, we adopt both the automatic metrics and human evaluation as our criterion. Four diverse automatic metrics are used in our experiments: BLEU, METEOR, ROUGE-L, and CIDEr. We utilize the open source evaluation code<sup>3</sup> used in (Yu et al., 2017b). For human evaluation, we employ the Amazon Mechanical Turk to perform two kinds of user studies (see Section 4.3 for more details).

**Training Details** We employ pretrained ResNet-152 model (He et al., 2016) to extract image features from the photostream. We built a vocabulary of size 9,837 to include words appearing more than three times in the training set. More training details can be found at Appendix B.

### 4.2 Automatic Evaluation

In this section, we compare our AREL method with the state-of-the-art methods as well as standard reinforcement learning algorithms on auto-

<sup>2</sup>There are only 3 (out of 21,075) broken images in the test set, which basically has no influence on the final results. Moreover, Yu et al. (2017b) also removed the 3 pictures, so it is a fair comparison.

<sup>3</sup>[https://github.com/lichengunc/vist\\_eval](https://github.com/lichengunc/vist_eval)

Method	B-1	B-2	B-3	B-4	M	R	C
Huang et al.	-	-	-	-	31.4	-	-
Yu et al.	-	-	21.0	-	34.1	29.5	7.5
XE-ss	62.3	38.2	22.5	13.7	34.8	<b>29.7</b>	8.7
GAN	62.8	38.8	23.0	14.0	35.0	29.5	9.0
AREL-s-50	62.9	38.4	22.7	14.0	34.9	29.4	9.1
AREL-t-50	63.4	39.0	23.1	<b>14.1</b>	<b>35.2</b>	29.6	9.5
AREL-s-100	<b>64.0</b>	38.6	22.3	13.2	35.1	29.3	<b>9.6</b>
AREL-t-100	63.8	<b>39.1</b>	<b>23.2</b>	<b>14.1</b>	35.0	29.5	9.4

Table 1: Automatic evaluation on the VIST dataset. We report BLEU (B), METEOR (M), ROUGH-L (R), and CIDEr (C) scores of the SOTA systems and the models we implemented, including XE-ss, GAN and AREL. AREL-s-N denotes AREL models with SoftSign as output activation and alternate frequency as N, while AREL-t-N denoting AREL models with Hyperbolic as the output activation (N = 50 or 100).

matic evaluation metrics. Then we further discuss the limitations of the hand-crafted metrics on evaluating human-like stories.

#### Comparison with SOTA on Automatic Metrics

In Table 1, we compare our method with Huang et al. (2016) and Yu et al. (2017b), which report achieving best-known results on the VIST dataset. We first implement a strong baseline model (XE-ss), which share the same architecture with our policy model but is trained with cross-entropy loss and scheduled sampling. Besides, we adopt the traditional generative adversarial training for comparison (GAN). As shown in Table 1, our XE-ss model already outperforms the best-known results on the VIST dataset, and the GAN model can bring a performance boost. We then use the XE-ss model to initialize our policy model and further train it with AREL. Evidently, our AREL model performs the best and achieves the new state-of-the-art results across all metrics.

But, compared with the XE-ss model, the performance gain is minor, especially on METEOR and ROUGE-L scores. However, in Sec. 4.3, the extensive human evaluation has indicated that our AREL framework brings a significant improvement on generating human-like stories over the XE-ss model. The inconsistency of automatic evaluation and human evaluation lead to a suspect that these hand-crafted metrics lack the ability to fully evaluate stories’ quality due to the complicated characteristics of the stories. Therefore, we conduct experiments to analyze and discuss the

Method	B-1	B-2	B-3	B-4	M	R	C
XE-ss	62.3	38.2	22.5	13.7	34.8	<b>29.7</b>	8.7
BLEU-RL	62.1	38.0	22.6	13.9	34.6	29.0	8.9
METEOR-RL	<b>68.1</b>	35.0	<u>15.4</u>	<u>6.8</u>	<b>40.2</b>	30.0	<u>1.2</u>
ROUGE-RL	58.1	<u>18.5</u>	<u>1.6</u>	<u>0</u>	27.0	<b>33.8</b>	<u>0</u>
CIDEr-RL	61.9	37.8	22.5	13.8	34.9	29.7	8.1
AREL (best)	<b>63.8</b>	<b>39.1</b>	<b>23.2</b>	<b>14.1</b>	<b>35.0</b>	29.5	<b>9.4</b>

Table 2: Comparison with different RL models with different metric scores as the rewards. We report the average scores of the AREL models as AREL (avg). Although METEOR-RL and ROUGE-RL models achieve the highest scores on their own metrics, the underlined scores are severely damaged. Actually, they are gaming their own metrics with nonsense sentences.

defects of the automatic metrics in section 4.2.

**Limitations of Automatic Metrics** String-match-based automatic metrics are not perfect and fail to evaluate some semantic characteristics of the stories (e.g. expressiveness and coherence). In order to confirm our conjecture, we utilize automatic metrics as rewards to reinforce the model with policy gradient. The quantitative results are demonstrated in Table 1.

Apparently, METEOR-RL and ROUGE-RL are severely ill-posed: they obtain the highest scores on their own metrics but damage the other metrics severely. We observe that these models are actually overfitting to a given metric while losing the overall coherence and semantical correctness. Same as METEOR score, there is also an adversarial example for ROUGE-L<sup>4</sup>, which is nonsense but achieves an average ROUGE-L score of 33.8.

Besides, as can be seen in Table 1, after reinforced training, BLEU-RL and CIDEr-RL do not bring a consistent improvement over the XE-ss model. We plot the histogram distributions of both BLEU-3 and CIDEr scores on the test set in Figure 5. An interesting fact is that there are a large number of samples with nearly zero score on both metrics. However, we observed those “zero-score” samples are not pointless results; instead, lots of them make sense and deserve a better score than zero. Here is a “zero-score” example on BLEU-3:

*I had a great time at the restaurant today.  
The food was delicious. I had a lot of food.  
The food was delicious. I had a great time.*

<sup>4</sup>An adversarial example for ROUGE-L: *we the was a . and to the . we the was a . and to the . we the was a . and to the . we the was a . and to the . we the was a . and to the .*

Method	Win	Lose	Unsure
XE-ss	22.4%	71.7%	5.9%
BLEU-RL	23.4%	67.9%	8.7%
CIDEr-RL	13.8%	80.3%	5.9%
GAN	34.3%	60.5%	5.2%
AREL	<b>38.4%</b>	<b>54.2%</b>	<b>7.4%</b>

Table 3: Turing test results.

The corresponding reference is

*The table of food was a pleasure to see!  
Our food is both nutritious and beautiful!  
Our chicken was especially tasty! We love  
greens as they taste great and are healthy!  
The fruit was a colorful display that tantalized  
our palette.*

Although the prediction is not as good as the reference, it is actually coherent and relevant to the theme “food and eating”, which showcases the defeats of using BLEU and CIDEr scores as a reward for RL training.

Moreover, we compare the human evaluation scores with these two metric scores in Figure 5. Noticeably, both BLEU-3 and CIDEr have a poor correlation with the human evaluation scores. Their distributions are more biased and thus cannot fully reflect the quality of the generated stories. In terms of BLEU, it is extremely hard for machines to produce the exact 3-gram or 4-gram matching, so the scores are too low to provide useful guidance. CIDEr measures the similarity of a sentence to the majority of the references. However, the references to the same image sequence are photostream different from each other, so the score is very low and not suitable for this task. In contrast, our AREL framework can learn a more robust reward function from human-annotated stories, which is able to provide better guidance to the policy and thus improves its performances over different metrics.

**Visualization of The Learned Rewards** In Figure 6, we visualize the learned reward function for both ground truth and generated stories. Evidently, the AREL model is able to learn a smoother reward function that can distinguish the generated stories from human annotations. In other words, the learned reward function is more in line with human perception and thus can encourage the model to explore more diverse language styles and expressions.

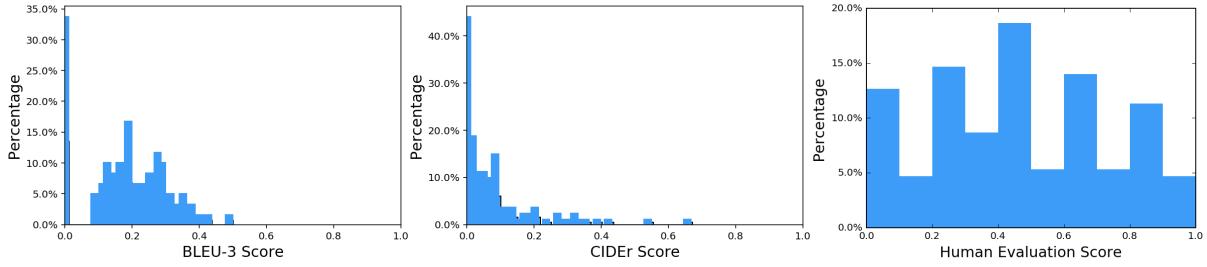


Figure 5: Metric score distributions. We plot the histogram distributions of BLEU-3 and CIDEr scores on the test set, as well as the human evaluation score distribution on the test samples. We use the Turing test results to calculate the human evaluation scores (see Section 4.3). Basically, 0.2 score is given if the generated story wins the Turing test, 0.1 for tie, and 0 if losing. Each sample has 5 scores from 5 judges, and we use the sum as the human evaluation score, so it is in the range [0, 1].

Choice (%)	AREL vs XE-ss			AREL vs BLEU-RL			AREL vs CIDEr-RL			AREL vs GAN		
	AREL	XE-ss	Tie	AREL	BLEU-RL	Tie	AREL	CIDEr-RL	Tie	AREL	GAN	Tie
Relevance	<b>61.7</b>	25.1	13.2	<b>55.8</b>	27.9	16.3	<b>56.1</b>	28.2	15.7	<b>52.9</b>	35.8	11.3
Expressiveness	<b>66.1</b>	18.8	15.1	<b>59.1</b>	26.4	14.5	<b>59.1</b>	26.6	14.3	<b>48.5</b>	32.2	19.3
Concreteness	<b>63.9</b>	20.3	15.8	<b>60.1</b>	26.3	13.6	<b>59.5</b>	24.6	15.9	<b>49.8</b>	35.8	14.4

Table 4: Pairwise human comparisons. The results indicate the consistent superiority of our AREL model in generating more human-like stories than the SOTA methods.

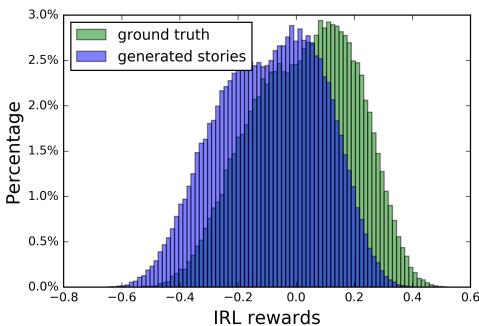


Figure 6: Visualization of the learned rewards on both the ground-truth stories and the stories generated by our AREL model. The generated stories are receiving lower averaged scores than the human-annotated ones.

**Comparison with GAN** We here compare our method with vanilla GAN (Goodfellow et al., 2014), whose update rules for the generator can be generally classified into two categories. We demonstrate their corresponding objectives and ours as follows:

$$\begin{aligned} GAN1 : \quad J_\beta &= \mathbb{E}_{W \sim p_\beta} [-\log R_\theta(W)], \\ GAN2 : \quad J_\beta &= \mathbb{E}_{W \sim p_\beta} [\log(1 - R_\theta(W))], \\ \text{ours} : \quad J_\beta &= \mathbb{E}_{W \sim p_\beta} [-R_\theta(W)]. \end{aligned}$$

As discussed in Arjovsky et al. (2017), *GAN1* is prone to the unstable gradient issue and *GAN2*

is prone to the vanishing gradient issue. Analytically, our method does not suffer from these two common issues and thus is able converge to optimum solutions more easily. From Table 1 we can observe slight gains of using AREL over GAN with automatic metrics, but we further deploy human evaluation for a better comparison.

### 4.3 Human Evaluation

Automatic metrics cannot fully evaluate the capability of our AREL method. Therefore, we perform two different kinds of human evaluation studies on Amazon Mechanical Turk: Turing test and pairwise human evaluation. For both tasks, we use 150 stories (750 images) sampled from the test set, each assigned to 5 workers to eliminate human variance. We batch six items as one assignment and insert an additional assignment as a sanity check. Besides, the order of the options within each item is shuffled to make a fair comparison.

**Turing Test** We first conduct five independent Turing tests for XE-ss, BLEU-RL, CIDEr-RL, GAN, and AREL models, during which the worker is given one human-annotated sample and one machine-generated sample, and needs to decide which is human-annotated. As shown in Table 3, our AREL model significantly outperforms all the other baseline models in the Turing test: it has much more chances to fool AMT worker (the

					
XE-ss	We took a trip to the mountains.	There were many different kinds of different kinds.	We had a great time. He was a great time.	We had a great time. He was a great time.	It was a beautiful day.
AREL	The family decided to take a trip to the countryside.	There were so many different kinds of things to see.	The family decided to go on a hike.	I had a great time.	At the end of the day, we were able to take a picture of the beautiful scenery.
Human-created Story	We went on a hike yesterday.	There were a lot of strange plants there.	I had a great time.	We drank a lot of water while we were hiking.	The view was spectacular.

Figure 7: Qualitative comparison example with XE-ss. The direct comparison votes (AREL:XE-ss:Tie) were 5:0:0 on Relevance, 4:0:1 on Expressiveness, and 5:0:0 on Concreteness.

ratio is AREL:XE-ss:BLEU-RL:CIDEr-RL:GAN = 45.8%:28.3%:32.1%:19.7%:39.5%), which confirms the superiority of our AREL framework in generating human-like stories. Unlike automatic metric evaluation, the Turing test has indicated a much larger margin between AREL and other competing algorithms. Thus, we empirically confirm that metrics are not perfect in evaluating many implicit semantic properties of natural language. Besides, the Turing test of our AREL model reveals that nearly half of the workers are fooled by our machine generation, indicating a preliminary success toward generating human-like stories.

**Pairwise Comparison** In order to have a clear comparison with competing algorithms with respect to different semantic features of the stories, we further perform four pairwise comparison tests: AREL vs XE-ss/BLEU-RL/CIDEr-RL/GAN. For each photostream, the worker is presented with two generated stories and asked to make decisions from the three aspects: relevance<sup>5</sup>, expressiveness<sup>6</sup> and concreteness<sup>7</sup>. This head-to-head compete is designed to help us understand in what aspect our model outperforms the competing algorithms, which is displayed in Table 4.

Consistently on all the three comparisons, a large majority of the AREL stories trumps the competing systems with respect to their relevance,

<sup>5</sup>Relevance: the story accurately describes what is happening in the image sequence and covers the main objects.

<sup>6</sup>Expressiveness: coherence, grammatically and semantically correct, no repetition, expressive language style.

<sup>7</sup>Concreteness: the story should narrate concretely what is in the image rather than giving very general descriptions.

expressiveness, and concreteness. Therefore, it empirically confirms that our generated stories are more relevant to the image sequences, more coherent and concrete than the other algorithms, which however is not explicitly reflected by the automatic metric evaluation.

#### 4.4 Qualitative Analysis

Figure 7 gives a qualitative comparison example between AREL and XE-ss models. Looking at the individual sentences, it is obvious that our results are more grammatically and semantically correct. Then connecting the sentences together, we observe that the AREL story is more coherent and describes the photo stream more accurately. Thus, our AREL model significantly surpasses the XE-ss model on all the three aspects of the qualitative example. Besides, it won the Turing test (3 out 5 AMT workers think the AREL story is created by a human). In the appendix, we also show a negative case that fails the Turing test.

## 5 Conclusion

In this paper, we not only introduce a novel adversarial reward learning algorithm to generate more human-like stories given image sequences, but also empirically analyze the limitations of the automatic metrics for story evaluation. We believe there are still lots of improvement space in the narrative paragraph generation tasks, like how to better simulate human imagination to create more vivid and diversified stories.

## Acknowledgment

We thank Adobe Research for supporting our language and vision research. We would also like to thank Licheng Yu for clarifying the details of his paper and the anonymous reviewers for their thoughtful comments. This research was sponsored in part by the Army Research Laboratory under cooperative agreements W911NF09-2-0053. The views and conclusions contained herein are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notice herein.

## References

- Pieter Abbeel and Andrew Y Ng. 2004. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning*, page 1. ACM.
- Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein gan. *arXiv preprint arXiv:1701.07875*.
- Dzmitry Bahdanau, Philemon Brakel, Kelvin Xu, Anirudh Goyal, Ryan Lowe, Joelle Pineau, Aaron Courville, and Yoshua Bengio. 2016. An actor-critic algorithm for sequence prediction. *arXiv preprint arXiv:1607.07086*.
- Santanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.
- Elia Bruni and Raquel Fernández. 2017. Adversarial evaluation for open-domain dialogue generation. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 284–288.
- Chris Callison-Burch, Miles Osborne, and Philipp Koehn. 2006. Re-evaluation the role of bleu in machine translation research. In *11th Conference of the European Chapter of the Association for Computational Linguistics*.
- Wenhu Chen, Guanlin Li, Shuo Ren, Shujie Liu, Zhirui Zhang, Mu Li, and Ming Zhou. 2018. Generative bridging network for neural sequence prediction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1706–1715. Association for Computational Linguistics.
- Wenhu Chen, Aurélien Lucchi, and Thomas Hofmann. 2016. Bootstrap, review, decode: Using out-of-domain textual data to improve image captioning. *CoRR*.
- Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. 2015. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*.
- Zhiqian Chen, Xuchao Zhang, Arnold P. Boedihardjo, Jing Dai, and Chang-Tien Lu. 2017. Multi-modal storytelling via generative adversarial imitation learning. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 3967–3973.
- Bo Dai, Sanja Fidler, Raquel Urtasun, and Dahua Lin. 2017. Towards diverse and natural image descriptions via a conditional gan. In *The IEEE International Conference on Computer Vision (ICCV)*.
- Chelsea Finn, Paul Christiano, Pieter Abbeel, and Sergey Levine. 2016. A connection between generative adversarial networks, inverse reinforcement learning, and energy-based models. *arXiv preprint arXiv:1611.03852*.
- Justin Fu, Katie Luo, and Sergey Levine. 2017. Learning robust rewards with adversarial inverse reinforcement learning. *arXiv preprint arXiv:1710.11248*.
- Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323.
- Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.
- Peter Henderson, Wei-Di Chang, Pierre-Luc Bacon, David Meger, Joelle Pineau, and Doina Precup. 2017. Optiongan: Learning joint reward-policy options using generative adversarial inverse reinforcement learning. *arXiv preprint arXiv:1709.06683*.
- Jonathan Ho and Stefano Ermon. 2016. Generative adversarial imitation learning. In *Advances in Neural Information Processing Systems*, pages 4565–4573.
- Ting-Hao K. Huang, Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Jacob Devlin, Aishwarya Agrawal, Ross Girshick, Xiaodong He, Pushmeet Kohli, Dhruv Batra, et al. 2016. Visual

- storytelling. In *15th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2016)*.
- Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.
- Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, and F Huang. 2006. A tutorial on energy-based learning. *Predicting structured data*, 1(0).
- Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out*.
- Chia-Wei Liu, Ryan Lowe, Iulian V Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. *arXiv preprint arXiv:1603.08023*.
- Ryan Lowe, Michael Noseworthy, Iulian V Serban, Nicolas Angelard-Gontier, Yoshua Bengio, and Joelle Pineau. 2017. Towards an automatic turing test: Learning to evaluate dialogue responses. *arXiv preprint arXiv:1708.07149*.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. 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.
- Cesc C Park and Gunhee Kim. 2015. Expressing an image stream with a sequence of natural sentences. In *Advances in Neural Information Processing Systems*, pages 73–81.
- Romain Paulus, Caiming Xiong, and Richard Socher. 2017. A deep reinforced model for abstractive summarization. *arXiv preprint arXiv:1705.04304*.
- Marc’Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2015. Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732*.
- Nathan D Ratliff, J Andrew Bagnell, and Martin A Zinkevich. 2006. Maximum margin planning. In *Proceedings of the 23rd international conference on Machine learning*, pages 729–736. ACM.
- Zhou Ren, Xiaoyu Wang, Ning Zhang, Xutao Lv, and Li-Jia Li. 2017. Deep reinforcement learning-based image captioning with embedding reward. In *Proceeding of IEEE conference on Computer Vision and Pattern Recognition (CVPR)*.
- Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575.
- Jing Wang, Jianlong Fu, Jinhui Tang, Zechao Li, and Tao Mei. 2018a. Show, reward and tell: Automatic generation of narrative paragraph from photo stream by adversarial training. *AAAI*.
- Xin Wang, Wenhui Chen, Jiawei Wu, Yuan-Fang Wang, and William Yang Wang. 2018b. Video captioning via hierarchical reinforcement learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Xin Wang, Yuan-Fang Wang, and William Yang Wang. 2018c. Watch, listen, and describe: Globally and locally aligned cross-modal attentions for video captioning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 795–801. Association for Computational Linguistics.
- Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256.
- Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. Msrvtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2017a. Seqgan: Sequence generative adversarial nets with policy gradient. In *AAAI*, pages 2852–2858.
- Licheng Yu, Mohit Bansal, and Tamara Berg. 2017b. Hierarchically-attentive rnn for album summarization and storytelling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 966–971, Copenhagen, Denmark. Association for Computational Linguistics.
- Brian D Ziebart. 2010. *Modeling purposeful adaptive behavior with the principle of maximum causal entropy*. Carnegie Mellon University.
- Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, and Anind K Dey. 2008. Maximum entropy inverse reinforcement learning. In *AAAI*, volume 8, pages 1433–1438. Chicago, IL, USA.

## Appendix

### A Error Analysis

**Failure Case in Turing Test** In Figure 8, we presented a negative example that failed the Turing test (4 out of 5 made the correct decision). Compared with the human-generated story, our AREL story lacked emotion and imagination and thus can be easily distinguished. For example, the real human gave the band a nickname “very loud band” and told a more amusing story. Though we have made encouraging progress on generating human-like stories, further research of creating diversified stories is still needed.

**Data Bias** From the experiments, we observe that there exist some severe data bias issues in the VIST dataset, such as gender bias and event bias. In the training set, the ratio of male and female’s appearances is 2.06:1, and it is 2.16:1 in the test set. the models aggravate the gender bias to 3.44:1. Besides, because all the images are collected from Flickr, there is also an event bias issue. We count three most frequent events: party, wedding, and graduation, whose ratios are 6.51:2.36:1 on the training set and 4.54:2.42:1 on the test set. However, their ratio on the testing results is 10.69:2.22:1. Clearly, the models tend to magnify the influence of the largest majority. These bias issues remain to be studied for future work.

### B Training Details

Our model is implemented on PyTorch and consists of two parts – a policy model and a reward model. The policy model is implemented with a multiple-RNN architecture. Each RNN model is responsible for generating a sub-story for each photo in the stream. But the weights are tied to minimize the memory consumption. The image features are extracted from the pre-trained ResNet-152 model<sup>8</sup>. The visual encoder receives the ResNet-152 features and uses recurrent neural network to understand the temporal dynamics and represents them as hidden state vectors, which is further fed into the decoder to generate stories. The reward model is based on convolutional neural network and uses convolution kernels to extract semantic features for prediction. Here we give the detailed description of our system:

- **Visual Encoder:** the visual encoder is a bi-directional GRU model with hidden dimension of 256 for each direction. we concatenate the bi-directional states and form a 512 dimension vector for the story generator. The input album is composed of five images, and each image is used as separate input to different RNN decoders.
- **Decoder:** The decoder is a single-layer GRU model with hidden dimension of 512. The recurrent decoder model receives the output from the visual encoder as the first input, and then at the following time steps, it receives the last predicted token as input or uses the ground truth as input. During scheduled sampling, we use a sampling probability to decide which action to take.
- **Reward Model:** we use a convolutional neural network to extract n-gram features from the story embedding and stretch them into a flattened vector. The embedding size of input story is 128, and the filter dimension of CNN is also 128. Here we use three kernels with window size 2, 3, 4, each with a stride size of 1. We use a pooling size of 2 to shrink the extracted outputs and flatten them as a vector. Finally, we project this vector into a single cell indicating the predicted reward value.

During training, we first pre-train a schedule-sampling model with a batch size of 64 with NVIDIA Titan X GPU. The warm-up process takes roughly 5-10 hours, and then we select the best model to initialize our AREL policy model. Finally, we use alternating training strategy to optimize both the policy model and the reward model with a learning rate of 2e-4 using Adam optimization algorithm. During test time, we use a beam size of 3 to approximate the whole search space, we force the beam search to proceed more than 5 steps and no more than 110 steps. Once we reach the EOS token, the algorithm stops and we compare the results with human-annotated corpus using 4 different automatic evaluation metrics.

### C Amazon Mechanical Turk

We used AMT to perform two surveys, one picks a more human-like story. We asked the worker to answers 8 questions within 30 minutes, and we pay 5 workers to work on the same sheet to

<sup>8</sup><https://github.com/KaimingHe/deep-residual-networks>

					
<b>XE-ss</b>	I went to the party last week.	The band played a lot of music.	[female] and [female] were having a great time.	[male] and [male] are having a great time at the party.	We had a great time at the party.
<b>AREL</b>	My friends and I went to a party.	The band played a lot of music.	[female] and [male] were having a good time .	[male] and [male] are the best friends in the world.	After a few drinks, everyone was having a great time.
<b>Human-created Story</b>	My first party in the dorm!	There was a very loud band called "very loud band".	my friend [female] had enough. She took my hand and led me to the kitchen where we couldn't hear.	[male] and [male] cornered me and asked me out on a date with them both .	Party! We all danced until passed out .

Figure 8: Failure case in Turing test. 4 out of 5 workers correctly recognized the human-created story and 1 person mistakenly chose AREL story.

eliminate human-to-human bias. Here we demonstrate the Turing survey form in [Figure 9](#). Besides, we also perform a head-to-head comparison with other algorithms, we demonstrate the survey form in [Figure 10](#).

**Survey Instructions** (Click to expand)

Read the following image streams and compare two stories in the aspect of matching, coherence, and concreteness.

**Given a photo stream, select a story which is more likely to be generated by human**

**Q1** Read the following image stream to answer the questions



**A.** the park was so crowded in the morning . the venue was filled with antsy people . the graduates word glossy black gowns . this faculty member gave a excited speech . we gathered together to share roses and balloons .

**B.** today was the day of the graduation ceremony . there were a lot of people there . everyone was very excited . the dean gave a speech to the graduates . everyone was very happy to be there .

Which story is generated by human?

A

B

Unsure

Figure 9: Turing Survey Form

### Survey Instructions (Click to expand)

Read the following image streams and compare two stories in the aspect of matching, coherence, and concreteness.



**Relevance:** the story **accurately describes what is happening** in the image stream and covers the main objects appearing in the images.

**Expressiveness:** coherence, grammatically and semantically correct, **no repetition, expressive language style**

**Concreteness:** the story should **narrate concretely what is in the image** rather than giving very general descriptions.

Good example: the students gathered to listen to the presenters give lectures . there was several presenters on hand to speak . they spoke to the crowd with new ideas . the students listened with interest . some of the students took notes as the presenters spoke .

Bad example (repetition): today was the day . i was very happy to see them . she was very happy to be there . they were all very happy to see him . this is a picture of a group .

Bad example (too abstract): this is a picture of a speaker . the speaker was very good . everyone is happy to be there . everyone was very happy . everyone was very happy .

**Q1** Read the following image stream to answer the questions



**A.** the graduation ceremony was held in the auditorium . there were a lot of people there . i was so proud of me . the dean of the school gave a speech to the graduates . everyone was so happy to be married .

**B.** today was the day of the graduation ceremony . there were a lot of people there . everyone was very excited . the dean gave a speech to the graduates . everyone was very happy to be there .

Which story better describe the images?

A     B     Tie

Which story is more coherent?

A     B     Tie

Which story is more concrete?

A     B     Tie

Figure 10: Pairwise Comparison Form