CgT-GAN: CLIP-guided Text GAN for Image Captioning

Jiarui Yu*
yjr@mail.ustc.edu.cn
University of Science and Technology of China
Hefei, China

Haoran Li*
lihaoran747@126.com
University of Science and Technology of China
Hefei, China

Yanbin Hao†
haoyanbin@hotmail.com
University of Science and Technology of China
Hefei, China

Bin Zhu
andrewzhu1216@gmail.com
Singapore Management University
Bras Basah, Singapore

Tong Xu
tongxu@ustc.edu.cn
University of Science and Technology of China
Hefei, China

Xiangnan He†
xiangnanhe@gmail.com
University of Science and Technology of China
Hefei, China

ABSTRACT

The large-scale visual-language pre-trained model, Contrastive Language-Image Pre-training (CLIP), has significantly improved image captioning for scenarios without human-annotated image-caption pairs. Recent advanced CLIP-based image captioning without human annotations follows a text-only training paradigm, *i.e.*, reconstructing text from shared embedding space. Nevertheless, these approaches are limited by the training/inference gap or huge storage requirements for text embeddings. Given that it is trivial to obtain images in the real world, we propose CLIP-guided text GAN (CgT-GAN), which incorporates images into the training process to enable the model to "see" real visual modality. Particularly, we use adversarial training to teach CgT-GAN to mimic the phrases of an external text corpus and CLIP-based reward to provide semantic guidance. The caption generator is jointly rewarded based on the caption naturalness to human language calculated from the GAN's discriminator and the semantic guidance reward computed by the CLIP-based reward module. In addition to the cosine similarity as the semantic guidance reward (*i.e.*, CLIP-cos), we further introduce a novel semantic guidance reward called CLIP-agg, which aligns the generated caption with a weighted text embedding by attentively aggregating the entire corpus. Experimental results on three subtasks (ZS-IC, In-UIC and Cross-UIC) show that CgT-GAN outperforms state-of-the-art methods significantly across all metrics. Code is available at https://github.com/Lihr747/CgtGAN.

CCS CONCEPTS

• **Computing methodologies** → **Natural language generation**; **Computer vision tasks**.

KEYWORDS

Image captioning; CLIP; Reinforcement learning; GAN

ACM Reference Format:
Jiarui Yu, Haoran Li, Yanbin Hao, Bin Zhu, Tong Xu, and Xiangnan He. 2023. CgT-GAN: CLIP-guided Text GAN for Image Captioning. In *Proceedings of the 31st ACM International Conference on Multimedia (MM '23), October 29–November 3, 2023, Ottawa, ON, Canada.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3581783.3611891

1 INTRODUCTION

Recently, CLIP (Contrastive Language-Image Pretraining) [47] has revolutionized the multi-modal domain by aligning images and text in a joint embedding space. CLIP has been shown to benefit numerous multi-modal tasks, such as VQA [52], text-to-image synthesis [49], and referring image segmentation [63]. In the studied image captioning, CLIP-based methods are also well explored in scenarios with paired training data. Prior works, such as those employing CLIP as a backbone [4, 40], or as a semantic enrichment technique [28], have shown improved captioning performance. Nevertheless, these methods require paired training, and the human pairwise label has a stake in the performance. In this work, we explore the feasibility of using CLIP for a more challenging task: image captioning without human-labeled pairs. That is, during training, we only utilize images and an external text corpus. The goal of the task is to generate a caption that textually describes a given image by leveraging unpaired images and sentences.

Current image captioning methods without annotations fall into two categories: concept-based and CLIP-based. Concept-based methods use computer vision techniques to discover various visual conceptual clues within images and then map the words to the caption. Existing methods [12, 16, 26] extract objects, scenes, and attributes by using deep models pre-trained on other related tasks such as object detection [21] and scene graph generation [70], and advocate rewarding the generated image caption for containing the detected visual concepts. Their performances often rely heavily on the quality of visual concept extraction. Also, they are incapable of capturing complex object interactions by using such text narration that explicitly refers to visual concept [16].

CLIP-based methods utilize the CLIP model as an oracle visual-language alignment tool, enabling captioning without human labels. One line of such methods [55, 58] controls a pre-trained generative

*Both authors contributed equally to this research.
†Yanbin Hao and Xiangnan He are both the corresponding authors.

MM '23, October 29–November 3, 2023, Ottawa, ON, Canada.

ACM ISBN 979-8-4007-0108-5/23/10…$15.00
https://doi.org/10.1145/3581783.3611891

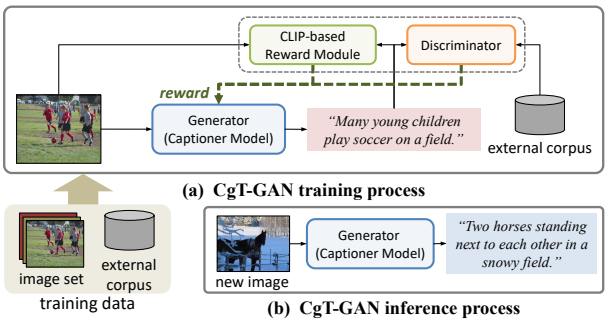


Figure 1: An illustration of our proposed CgT-GAN. (a) Rewards from the CLIP-based reward module (semantic guidance) and from the discriminator (naturalness score) are combined to guide the generator (captioner). (b) Take out the generator for new image inference.

language model (LM) to produce image captions in a zero-shot manner. However, its performance is subpar due to the generative LM's poor fit for the captioning task, despite requiring no additional data. Another line of approaches [17, 27, 42] involves training a captioner with text-only data and reconstructing text from the CLIP text embedding. This is feasible since the text embedding shares the same cosine space with the image embedding, but these methods are often constrained by either training/inference gaps or substantial storage for textual embeddings (for projecting visual embedding to textual embedding). The final series of methods [69, 75] combine images and text corpus to compute image-text similarity using CLIP. The models are then rewarded to enhance caption grammar or identify highly correlated pseudo image-text pairs.

In real-world applications, the images are usually easy to obtain. We, therefore, adopt the same settings as PL-UIC [75] and ESPER-style [69], which involve the simultaneous use of images and an external corpus during training. Interestingly, even though additional images are adopted, existing solutions that use both images and sentences are inferior to text-only methods. This phenomenon motivated us to explore a more effective way of using CLIP to understand the visual modality and predict accurate captions. In this paper, as shown in Figure 1, we propose a CLIP-guided text generative adversarial network (CgT-GAN), integrating CLIP [47] into the text GAN [68] in a more effective manner to continually guide the image-to-caption generation. Specifically, CLIP is not only for image feature encoding but also for semantic guidance rewarding. To enhance the text generation capacity of text GAN, we exploit the transformer-based language model GPT-2 [48] as the generator to generate conditional synthetic caption and the improved BERT-RoBERTa [33] as the discriminator to guess between real and fake sentences. The network is trained in an end-to-end fashion without making any extra effort on entity detection or pseudo labelling, where the CLIP-based reward is combined with the naturalness computed by the discriminator to jointly train the generator.

Furthermore, for the reward module, we explore two rewarding strategies: CLIP-cos and CLIP-agg. CLIP-cos calculates the cosine similarity of the image-caption pair directly and rewards the generator accordingly. Inspired by [27], we additionally propose a more effective reward option, CLIP-agg, which attentively aggregates text embeddings in corpus with CLIP to guide the captioner generation.

Our contribution is three-fold:

(1) Compared to text-only CLIP-based methods, we adopt images in the training stage, which makes the captioner "see" real visuals to minimize the training/inference input domain gap and improve performance.

(2) Different from current CLIP-based RL or adversarial learning methods, we embed a CLIP-based reward module in a text GAN framework. Two reward strategies are proposed and analyzed for effective rewarding in an end-to-end fashion.

(3) Our model constantly outperforms the existing methods on three subtasks without human-labeled pairs: zero-shot, in-domain unpaired, and cross-domain unpaired image captioning.

2 RELATED WORK

2.1 Image Captioning

Image captioning (IC) model learns to describe images with manually annotated image-caption pairs. Harnessing on these labels, IC training naturally focuses on maximizing the probability of correct caption. The widely used architecture of IC models is the encoder-decoder[25, 61], where the encoder captures visual content and the decoder generates caption. Riding on the structured network, attention mechanism [59] is also adopted to pay varying attention to image parts and tokens [2, 8, 38, 43, 66, 67]. Apart from model designing, advanced learning paradigms, such as reinforcement learning [50] and adversarial learning[7, 10], are utilized to further improve the vision-faith and text-realism for captions.

In contrast to traditional IC, image captioning without human annotations has recently drawn researchers' attention, where models cannot access any labelled image-text pairs. Thus, its key challenge is *how to align vision and language when pairwise annotations are unavailable*. Successful attempts [12, 16, 26] mainly follow the pipeline of teaching the caption generator to *speak* human language and reducing the image-text mismatch. The goal of speaking human language is mostly achieved by utilizing either text reconstruction [18, 32, 39] or adversarial learning [5, 6, 12, 53]. Their major difference lies in the domain alignment of vision-language. Before the era of CLIP, the most representative solution is to identify features, *e.g.*, visual concepts, that are common to both image and text. By harvesting visual objects or entities from images in advance, these methods can thus generate captions that contain the same visual concepts. Specifically, two kinds of concept-based alignment strategies are typically used. The first one is to learn a concept-to-sentence translator to ensure the caption being concept-related to image [5, 12, 18, 20, 39, 71, 74]. The other one performs concept matching in a joint embedding space, where image and text embeddings will be pulled closer if they share the same concepts, otherwise, pushed apart [26, 53]. These concept-based approaches assure the caption of containing visual concepts, nevertheless, they become ineffectual in modeling complex object correlations. To address this issue, research efforts are also made to explicitly construct scene graphs based on image objects to enrich visual context [6, 13, 16, 32]. Though providing more contextual information, the construction of a scene graph always requires complex preprocessing (*e.g.*, object detection and relation formulation) and may overlook the global understanding of image content as compared to CLIP, which possesses rich vision-language knowledge.

2.2 CLIP for Image Captioning

Contrastive Language–Image Pre-training [47] (CLIP) is a large-scale multi-modal pre-training model comprised of two sub-modules: image encoder and text encoder, which aligns image and text in a joint embedding space through cosine similarity. CLIP has been actively exploited in many multimedia tasks[22, 24, 41, 45, 62, 63]. In the studied image captioning, the most straightforward way of using CLIP is to adopt the image encoder for more expressive feature extraction [40, 57, 58, 69, 75], and the text encoder for improving caption grammar [9] or semantic comprehension [28].

As CLIP is trained on a web-scale image-text dataset by contrastively maximizing the visual-semantic similarity, both its image encoder and text encoder inherently possess the knowledge of cross-modal alignment. Consequently, some works [17, 27, 42, 55, 58, 69, 75] consider using CLIP to teach/train a generative model for image captioning without human labelling. These works vary in the use-pattern of CLIP guidance. Specifically, [55, 58] adjusts the language model by assessing the relatedness of each token to an image with CLIP at inference time. [17, 27, 42] share a similar idea of training the captioner to reconstruct text from CLIP visual-language space. These approaches only utilize text corpus during training, making them data-efficient. In detail, [17, 42] address the modality gap by adding noise to the textual embeddings, while [27] projects visual embedding into textual space when inference. However, the former two methods still suffer from the input difference between training and inference, and the latter requires additional textual embedding storage when projecting embedding from visual space to textual space. In contrast to these text-only methods, the proposed CgT-GAN adopts images and an external corpus during training. This arrangement allows the model to "see" visual modality during training, which overcomes the mentioned imperfections. Similar setting to ours, [69] employs a combination of CLIP reward and text likelihood reward to jointly guide the generator's learning, while [75] utilizes CLIP to obtain high-quality pseudo pairwise labels. Compared to these works, the CLIP-based reward module is novelly incorporated in a text GAN and provides a visual-semantic reward to guide the caption generator. Additionally, two rewarding strategies are proposed and explored in three subtasks.

3 METHODOLGY

In this section, we elaborate on the details of CLIP-guided text GAN (CgT-GAN). Figure 2(a) depicts the overall framework of CgT-GAN. CgT-GAN is composed of two modules, a text GAN module and a CLIP-based reward module. Instead of pure reinforcement learning or using pseudo labels by previous works [69, 75], CgT-GAN is learned through the GAN framework, where its generator is optimized with a simple but effective reward.

3.1 Problem Formulation

In this work, we focus on image captioning without using human-annotated image-caption pairs. Denote $\mathbb{I} = \{\mathcal{I}_i\}_{i=1}^{N_I}$ as a set of images, where N_I indicates the total number of images. In addition, an external text corpus containing a set of sentences $\mathbb{S} = \{\mathcal{S}_i\}_{i=1}^{N_S}$, where N_S denotes the total number of sentences, is generally employed to provide rich linguistic knowledge and teach the captioner to mimic human language naturally. The goal is thus to learn a mapping function $\mathcal{G} : \mathcal{I} \rightarrow \mathcal{C}$ by using $\mathbb{I}$ and $\mathbb{S}$, but without any pairwise labelling. Here, $\mathcal{C}$ refers to the generated caption.

3.2 Text GAN Module

Similar to the vanilla GAN [14] and its application variants [34, 73], the text GAN module in our work is also composed of a generator G and a discriminator D. The training mechanism follows the adversarial way.

Generator. The generator G is trained to generate a natural language caption $\mathcal{C}$ for an input image $\mathcal{I}$. For such an image-to-text generation task, the input image feature representation is ardently expected to contain rich language-aware information. We thus adopt the CLIP model, which well masters the vision-language prior knowledge through training on web-collected image-sentence data, to extract the visual feature. Specifically, given an image $\mathcal{I}$, we firstly obtain the feature embedding $\boldsymbol{e}^{\mathcal{I}} \in \mathbb{R}^{d_1}$ by using the frozen CLIP image encoder (implemented with ViT-L/14 [11]). Then, similar to ClipCap [40], a two-layer multi-layer perceptron (MLP) is employed to output a set of vectors, denoted as visual prompts, $\mathbf{P} = [\boldsymbol{p}_1, \boldsymbol{p}_2, \cdots, \boldsymbol{p}_k]$ where $\mathbf{P} \in \mathbb{R}^{k \times d_2}$. d_1 and d_2 refer to the dimension of feature and prompt embeddings. Formally, we have

$$\boldsymbol{e}^{\mathcal{I}} = \text{CLIP-ImageEncoder}(\mathcal{I}), \tag{1}$$

$$\boldsymbol{p}_1, \boldsymbol{p}_2, \cdots, \boldsymbol{p}_k = \text{MLP}(\boldsymbol{e}^{\mathcal{I}}). \tag{2}$$

Finally, the pre-trained generative language model GPT-2 [48] is utilized to instantiate the caption generator G. Here, GPT-2 takes the k visual prompts $\{\boldsymbol{p}_i\}_{i=1}^{k}$ as the input tokens and continuously predicts the next word. Specifically, GPT-2 generates the t-th caption word $\boldsymbol{c}_t$ as:

$$\boldsymbol{c}_t = \arg\max_i P(\boldsymbol{w}_i | \boldsymbol{p}_1, \boldsymbol{p}_2, \cdots, \boldsymbol{p}_k, \boldsymbol{c}_1, \boldsymbol{c}_2, \cdots, \boldsymbol{c}_{t-1}), \tag{3}$$

where $\boldsymbol{c}_1, \boldsymbol{c}_2, \cdots, \boldsymbol{c}_{t-1}$ are prefix tokens predicted before t-th step, $\boldsymbol{w}_i$ is the i-th entry token in GPT-2's word dictionary and P is the conditional probability. The sentence decoding stops when the sequence is as long as enough or meets the end-of-sequence ("EOS") token. We set the max length as 20 and the "EOS" token as ".".

Discriminator. The discriminator D is to distinguish between real and fake (generated) sentences, *i.e.*, $\mathcal{S}$ and $\mathcal{C}$. In practice, we employ another pre-trained natural language understanding model RoBERTa followed by a two-layer MLP as our discriminator. Similar to traditional GAN, the discriminator D is trained to make a judgment about how real the generated caption is, providing feedback to make the generator G generate more human-like sentences. Concretely, the caption $\mathcal{C}$ and the real sentence $\mathcal{S}$ sampled from the external corpus are separately fed into RoBERTa for obtaining discriminative representation and then passed to the MLP to calculate their naturalness (a scalar value that measures how close the sentence is to natural language) [10]. The computational process is formally described as follows:

$$\begin{aligned} f_D(\mathcal{S}) &= \text{MLP}(\text{RoBERTa}(\mathcal{S})), \\ f_D(\mathcal{C}) &= \text{MLP}(\text{RoBERTa}(\mathcal{C})). \end{aligned} \tag{4}$$

After obtaining the naturalness scores $f_D(\mathcal{S})$ and $f_D(\mathcal{C})$, the generator and discriminator can be optimized alternatively by adversarial training. However, due to the discreteness of generated caption, the gradient cannot be directly backpropagated from the

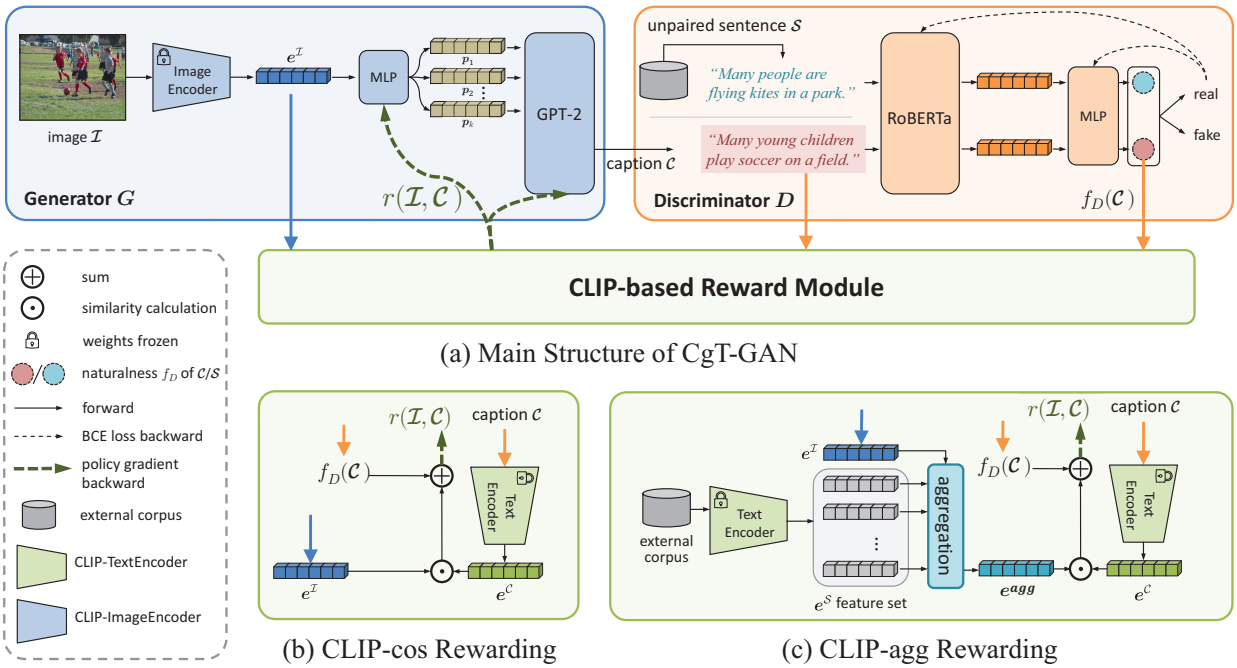


Figure 2: (a) Overall framework of CgT-GAN. CgT-GAN is composed of a text GAN module containing a caption generator G (blue block) and a discriminator D (orange block) and a CLIP-based reward module (green block). Leveraging on the proposed CgT-GAN, the adversarial loss for real/fake sentence discrimination and a combined reward for both language naturalness and image-caption alignment are introduced. (b) and (c) depict details of the CLIP-based reward module with CLIP-cos and CLIP-agg rewarding strategies. The former computes the cosine similarity as semantic guidance, while, the latter encourages closer distance between the generated text embedding with an aggregation embedding of the corpus.

discriminator to the generator. To tackle this problem, we regard the GAN learning as a reinforcement learning (RL) and use the policy gradient to train the network. The model training will be explained in detail in the section below.

3.3 CLIP-based Reward Module

The text GAN can only make the generated caption more human-like. The remaining key problem is how to align the caption with the image. In other words, the generated caption should semantically describe the image content. Recall that image-caption pairwise data is unavailable in our settings. Therefore, we propose a CLIP-based reward module to achieve image-caption semantic alignment, which produces a *semantic guidance reward* to further adjust the generator G. For the reward module, two rewarding strategies are proposed: **CLIP-cos** and **CLIP-agg**.

CLIP-cos. CLIP-cos simply calculates the CLIP similarity between the image $\mathcal{I}$ and the generated caption C, *i.e.*, the cosine similarity of their embeddings. Specifically, as shown in Figure 2(b), the generated caption C is fed into the frozen CLIP text encoder, resulting in a text embedding vector $\boldsymbol{e}^C$ as:

$$\boldsymbol{e}^C = \text{CLIP-TextEncoder}(C). \tag{5}$$

Afterwards, given the CLIP-based image embedding $\boldsymbol{e}^{\mathcal{I}}$ and caption embedding $\boldsymbol{e}^C$, the cosine similarity can be easily calculated by

$$r_{\cos}(\mathcal{I}, C) = \cos(\boldsymbol{e}^{\mathcal{I}}, \boldsymbol{e}^C) = \frac{\boldsymbol{e}^{\mathcal{I}} \cdot \boldsymbol{e}^C}{\left|\boldsymbol{e}^{\mathcal{I}}\right| \left|\boldsymbol{e}^C\right|}. \tag{6}$$

We regard the cosine similarity $\cos(\boldsymbol{e}^{\mathcal{I}}, \boldsymbol{e}^C)$ as the CLIP-cos reward $r_{\cos}$ used for the text GAN. As CLIP is pre-trained for vision-language matching by cosine score, the reward can provide robust semantic guidance.

CLIP-agg. As analyzed in [29], CLIP segregates image and text embeddings into two narrow, discrete cone-shaped spaces, known as *modality gap*. This indicates that the cross-modal alignment of CLIP-cos may be inefficient. In contrast, as shown in Figure 2(c), the aggregation operation of CLIP-agg enables the image embedding to be alternatively represented by a weighted sum of its associated text embeddings in the corpus. Subsequently, the CLIP-agg reward encourages a closer distance between the generated caption embedding and the aggregated textual embedding, facilitating image-text alignment within the shared CLIP text embedding space. To calculate the reward, we first compute the text embeddings $\{\boldsymbol{e}^{\mathcal{S}_i}\}$ of the external corpus. Then, we obtain an image-aware aggregated textual embedding $\boldsymbol{e}^{\text{agg}}$ through CLIP embeddings attention-weighted summation (aggregation in Figure 2(c)):

$$\boldsymbol{e}^{\mathcal{S}_i} = \text{CLIP-TextEncoder}(\mathcal{S}_i), i = 1, 2, \cdots, N_S, \tag{7}$$

$$\boldsymbol{e}^{\text{agg}} = \sum_{i=1}^{N_S} \frac{\exp\left(\cos(\boldsymbol{e}^{\mathcal{S}_i}, \boldsymbol{e}^{\mathcal{I}})/\tau\right)}{\sum_{k=1}^{N_S} \exp\left(\cos(\boldsymbol{e}^{\mathcal{S}_k}, \boldsymbol{e}^{\mathcal{I}})/\tau\right)} * \boldsymbol{e}^{\mathcal{S}_i}, \tag{8}$$

where N_S is the number of corpus sentences and τ is the temperature coefficient. The CLIP-agg reward r_{agg} takes both cosine similarity and L_1 penalty between $\boldsymbol{e}^{\text{agg}}$ and $\boldsymbol{e}^C$ into consideration,

as they are both in the text domain. It is denoted as follows:

$$r_{\text{agg}}(\mathcal{I}, C) = \cos(\boldsymbol{e}^{C}, \boldsymbol{e}^{\text{agg}}) - L_1(\boldsymbol{e}^{C}, \boldsymbol{e}^{\text{agg}}). \tag{9}$$

This ensures that the generated caption is not only semantically similar to the image context (cos) but also minimizes the difference in text embedding space between the generated text and the aggregation of the corpus sentences (L_1). Note that the CLIP encoders are fixed during training, and the aggregated text embedding $\boldsymbol{e}^{\text{agg}}$ can be calculated offline. Therefore, no additional computation for Eq. (7) and Eq. (8) is incurred during the training process.

We choose CLIP-agg as the default semantic guidance rewarding strategy for our CgT-GAN. Both the two strategies and their combination are compared in the experiment section.

3.4 Model Training

There are two network blocks in CgT-GAN, *i.e.*, the generator G and discriminator D, that require training. Prior to adversarial training, the generator is initialized by training it to reconstruct sentences from the given corpus using their CLIP embeddings. This stage is referred to as *initialization*. Afterwards, the pre-trained modules GPT-2 and RoBERTa will be fine-tuned empirically during the *adversarial training* stage. The discriminator D calculates the naturalness f_D for real/fake sentences, which is optimized through binary cross-entropy loss minimization, as follows:

$$\begin{aligned}\min_{\varphi} - \mathbb{E}_{\mathcal{S}\sim p_{\text{corpus}}} \left[\log \sigma\left(f_{D_\varphi}(\mathcal{S})\right)\right] \\ - \mathbb{E}_{C\sim G_\theta}\left[\log\left(1-\sigma\left(f_{D_\varphi}(C)\right)\right)\right],\end{aligned} \tag{10}$$

where $\mathcal{S}$ is the real sentence with a corpus distribution p_{corpus}, C is the generated caption sampled from G, φ denotes the parameters of D and σ is the sigmoid function.

We regard the generator training as an RL problem. In particular, our generator G can be viewed as a policy, which predicts the next word ("action") based on the current visual prompts and tokens ("state"). Therefore, the policy gradient can be approximated by using the REINFORCE algorithm [56] as follows:

$$\begin{aligned}\nabla J(\theta) &= \mathbb{E}_{C^s\sim G_\theta(C|\mathcal{I})}(R(\mathcal{I}, C^s) - R(\mathcal{I}, \hat{C}))\nabla \log G_\theta(C^s|\mathcal{I}), \\ G_\theta(C^s|\mathcal{I}) &= \prod_{t=1}^{n} G_\theta(C_t^s|\mathcal{I}, C_{1:t-1}^s),\end{aligned} \tag{11}$$

where C^s is the caption sampled from the generator with each token being selected using the estimated probability G_θ, $\hat{C}$ is the predicted caption (*i.e.*, each token is selected with the highest probability) under the inference algorithm, and $R(\cdot)$ is the reward function. The above gradient computation follows the self-critical sequence training (SCST) method [50], which normalizes the reward utilizing the output of the generator's own test-time inference algorithm. SCST can achieve high performance on image captioning involving the test-mode inference (the baseline $R(\mathcal{I}, \hat{C})$) into the training promotes the training/test time consistency. In the context of vanilla text GANs, the reward function R can be defined as the naturalness score $f_D(C)$ assigned by the discriminator D to the generated caption, as shown in Eq. (4). However, in our scenario, the objective of training the generator G is two-fold: to produce captions that are both highly natural (*i.e.*, resembling human language) and semantically consistent with the corresponding image. The overall reward function $R(\mathcal{I}, C)$ is thus composed of two components: the naturalness score $f_D(C)$, as computed by the discriminator D, and the semantic guidance reward r_* calculated by the reward module:

$$R(\mathcal{I}, C) = f_D(C) + r_*(\mathcal{I}, C). \tag{12}$$

r_* can be $r_{\cos}$ (Eq. (6)) and r_{agg} (Eq. (9)). Unless specified otherwise, we adopt r_{agg} as r_*.

4 EXPERIMENTS

In this section, we first introduce the datasets, evaluation metrics and tasks, and then make a thorough examination to answer the following four research questions:

- **RQ1**: How does CgT-GAN perform compared with current state-of-the-art methods?
- **RQ2**: How does the performance of CgT-GAN vary with different reward combinations, and which rewarding strategy yields the best results?
- **RQ3**: Can competitive performance be achieved by simpler CLIP usage or generative training on web-scale noisy pairs?
- **RQ4**: How does CgT-GAN perform with CLIP of different scales?

4.1 Experimental Setup

Dataset. We use two different image caption datasets, *i.e.*, MSCOCO Caption Dataset [31] and Flickr30k [46]. MSCOCO contains 123,287 images with each image being annotated with five descriptions. Flickr30k has 31,783 images collected from Flickr website and also attaches five sentences to each image. We adopt the commonly used data split [38]. For external corpus, the used ShutterStock(SS)[12], Google Conceptual Captions(CC3M)[51], Flickr30k and MSCOCO training datasets contain 2.3M, 3.3M, 145k, and 557k sentences, respectively. Note that SS and CC3M are collected from the web, and Flickr30k and MSCOCO corpus are created by human labellers.

Evaluation. Five commonly used evaluation metrics, including BLEU-4 [44], ROUGE [30], METEOR [3], CIDEr [60] and SPICE [1] are adopted for measuring the performance of methods from various perspectives. CLIP-based metrics like CLIP-S and refCLIP-S[19] are not considered, because these scores share some similarities with our reward, thus unable to reflect real performance.

Tasks. Following DeCap[27], we conduct experiments on three distinct tasks. (1) Zero-shot image captioning (ZS-IC)[1] (2) In-domain unpaired image captioning (In-UIC). (3) Cross-domain unpaired image captioning (Cross-UIC). In ZS-IC, we use sentence corpora crawled from the web, while in In-UIC and Cross-UIC, we use descriptions from an image caption training dataset as the corpus. Cross-UIC requires that the images and the sentence corpus come from different datasets, whereas In-UIC means the images and the corpora belong to the same dataset but without pairwise information. Formally, the image captioning task is expressed in the form of *X images* ↔ *Y captions*, indicating that the images are sourced from the X dataset, while the captions are sourced from the Y dataset during the training phase. Concretely, we construct two ZS-IC tasks: *MSCOCO images* ↔ *SS captions* and *MSCOCO images* ↔ *CC3M captions*, two Cross-UIC tasks: *Flickr30k images* ↔ *MSCOCO captions* and *MSCOCO images* ↔ *Flickr30k captions* and two In-UIC

[1]The zero-short task is defined by DeCap[27], which chooses webly-collected corpora for use. In our experiment, we additionally access images in the training set.

Table 1: Performance comparison on the test split of the MSCOCO datasets under zero-shot captioning setting (with SS and CC3M corpus). Items in grey are CLIP-based methods.

Method	B.-4	M.	R.	C.	S.
MSCOCO images ↔ SS captions					
UIC-GAN [12]	5.6	12.4	28.7	28.6	8.1
R^2M [18]	6.4	13.0	31.3	29.0	9.1
IGGAN [6]	6.5	13.1	30.5	28.8	8.2
TSGAN [71]	6.9	13.0	32.3	28.9	8.3
[12] + Honda *et al.* [20]	7.1	14.1	35.2	35.7	9.2
PL-UIC [75]	10.0	16.2	35.8	45.8	11.6
DeCap [27]	8.9	17.5	—	50.6	13.1
CgT-GAN (ours)	**11.1**	**19.0**	**37.2**	**58.6**	**14.5**
MSCOCO images ↔ CC3M captions					
SME-Emb [26]	6.5	12.9	35.1	22.7	—
R^2M [18]	8.3	14.0	35.0	29.3	9.6
Honda *et al.* [20]	7.6	13.5	**37.3**	31.8	8.4
DeCap [27]	8.8	16.0	—	42.1	10.9
CgT-GAN (ours)	**10.9**	**16.9**	35.2	**49.8**	**12.5**

tasks: *MSCOCO images* ↔ *MSCOCO captions* and *Flickr30k images* ↔ *Flickr30k captions*. It is important to note that the images in the validation and test sets are unseen during training. Details of model optimization can be found in the supplementary material.

4.2 Comparison with the State-of-the-Art Methods (RQ1)

We compare CgT-GAN with various state-of-the-art (SOTA) methods, which can be categorized into two groups: concept-based methods and CLIP-based methods. The former detects objects to bridge the gap between visual and textual information. The latter aligns visuals and language with the help of CLIP. We designated a gray background in the table to distinguish CLIP-based methods, which include *text-only methods* such as DeCap [27], CapDec [42], and CLOSE [17], as well as *methods that employ unpaired images and texts*, such as PL-UIC [75] and ESPER [69]. The comparisons of results are conducted separately on the aforementioned three settings: zero-shot image captioning (ZS-IC), in-domain unpaired image captioning (In-UIC), and cross-domain unpaired image captioning (Cross-UIC), following the same protocols.

Results on the ZS-IC task. Table 1 presents the performance comparison of two zero-shot image captioning tasks. Our CgT-GAN consistently achieves the best results among all the competing methods on two ZS-IC tasks in terms of BLEU-4, METEOR, CIDEr and SPICE, outperforming other GAN methods (*e.g.*, TSGAN) and the advanced CLIP-based method (*e.g.*, PL-UIC and DeCap). For instance, on the *MSCOCO↔SS* task, CgT-GAN obtains 19.0/58.6/14.5 METEOR/CIDEr/SPICE scores, which are 34.8%/64.1%/57.6% better than the ensemble of Honda *et al.*+UIC-GAN and 8.6%/15.8%/10.7% better than DeCap, respectively. These results demonstrate the impressive capacity of CgT-GAN in generating lifelike image captions.

CLIP offers richer and more diverse visual-language knowledge than competing methods such as UIC-GAN, R2M, and SME-Emb, which only use category-limited visual concepts. Compared to CLIP-based methods, our training paradigm is more efficient than PL-UIC with CLIP-filtered pseudo labels. Additionally, we observed that utilizing unlabelled image data during training significantly improves CgT-GAN's performance compared to text-only methods like DeCap. We also notice that CgT-GAN performs slightly worse than Honda *et al.* on ROUGE with the CC3M corpus. This is mainly because Honda *et al.* additionally utilizes pre-detected visual concepts, which are the key focus of ROUGE. In contrast, CgT-GAN is learned end-to-end without making any extra effort on entity detection. Furthermore, CgT-GAN trained with the SS corpus achieved higher scores than with the CC3M corpus because the SS corpus crawled using MSCOCO eighty keywords can provide more supportive sentences for generator guidance.

Table 2: Performance comparison on the test split of the MSCOCO and Flickr30k datasets under the cross-domain unpaired setting. Items in grey are CLIP-based methods.

Method	B.-4	M.	R.	C.	S.
Flickr30k images ↔ MSCOCO captions					
SME-Emb [26]	7.9	13.0	32.8	9.9	—
UIC-GAN [12] from [5]	8.3	13.3	33.4	14.2	—
R^2M [18]	11.7	13.7	35.9	18.1	8.3
SCS [5]	13.0	14.1	37.8	18.1	—
DeCap [27]	16.3	17.9	—	35.7	11.1
CapDec [42]	**17.3**	18.6	42.7	35.7	—
CgT-GAN (ours)	**17.3**	**19.6**	**43.9**	**47.5**	**12.9**
MSCOCO images ↔ Flickr30k captions					
CapDec[42]	9.2	16.3	36.7	27.3	—
DeCap[27]	12.1	18.0	—	44.4	10.9
CgT-GAN (ours)	**15.2**	**19.4**	**40.9**	**58.7**	**13.4**

Results on the Cross-UIC task. Note that images and the corpus are from the different datasets on the Cross-UIC setting. Table 2 shows the performance of the models on the image-corpora cross-domain settings of Flickr30k and MSCOCO. CgT-GAN outperforms other models on Cross-UIC task, demonstrating a similar performance trend to that on the ZS-IC task. Interestingly, with the same MSCOCO images, the use of Flickr30k corpus from human labellers only brings 0.2% CIDEr relative improvement (58.6 → 58.7) compared to the SS corpus. This suggests that web-collected corpora may work as well as human descriptions for our model.

Results on the In-UIC task. The In-UIC setting involves training images and corpus from the same dataset to test the upper bound of performance without supervision. Our experiments are conducted on Flickr30k and MSCOCO, as presented in Table 3. To the best of our knowledge, our proposed CgT-GAN is the first to surpass a CIDEr score of 100 on the MSCOCO task with a vanilla CLIP backbone. CgT-GAN outperforms other reinforcement learning methods based on CLIP, such as ESPER-Style [69], and CLIP text-only training methods, like CLOSE [17], which further demonstrates the advantages of our reinforcement strategy and the benefits of incorporating images in the training process. It is also certain that our CgT-GAN performs better than a group of traditional visual concept-based methods, like SCS and Graph-Align. Surprisingly, we found that our results on the MSCOCO dataset under the In-UIC setting (CIDEr score of **108.1**) are close to those of a similar generator, ClipCap[40], trained using image-caption pairs (CIDEr score of **113.1**).

Table 3: Performance comparison on the test split of the MSCOCO and Flickr30k datasets under the in-domain unpaired setting. Items in grey are CLIP-based methods.

Method	B.-4	M.	R.	C.	S.
MSCOCO images ↔ MSCOCO captions					
Pivoting[15]	5.4	13.2	—	17.7	—
SSR[54]	11.1	14.2	—	28.2	—
Coarse-SRE[32]	16.5	14.3	33.4	37.2	10.6
Fine-SRE[32]	19.7	17.4	41.9	49.7	13.3
UIC-GAN[12]	18.6	17.9	43.1	54.9	11.1
R^2M [18] from [53]	16.0	17.3	39.7	48.4	11.2
TSGAN[71]	18.9	18.2	43.3	55.2	11.3
SME-Emb[26]	19.3	20.2	45.0	61.8	12.9
MemGAN[53]	20.0	19.9	45.1	63.6	12.9
IGGAN[6]	21.9	21.1	46.5	64.0	14.5
Graph-Align[16]	21.5	20.9	47.2	69.5	15.0
SCS[5]	22.8	21.4	47.7	74.7	15.1
[16] + Fine-SRE[32]	21.8	22.1	48.4	75.7	16.1
PL-UIC [75]	25.0	22.6	49.4	77.9	15.2
ESPER-Style [69]	21.9	21.9	—	78.2	—
DeCap [27]	24.7	25.0	—	91.2	18.7
CapDec [42]	26.4	25.1	51.8	91.8	—
CLOSE [17]	28.6	25.2	—	95.4	18.1
CgT-GAN (ours)	**30.3**	**26.9**	**54.5**	**108.1**	**20.5**
Flickr30k images ↔ Flickr30k captions					
UIC-GAN [12] from [5]	10.8	14.2	33.4	15.4	—
SCS [5]	14.3	15.6	38.5	20.5	—
CapDec [42]	17.7	20.0	43.9	39.1	—
DeCap [27]	21.2	21.8	—	56.7	15.2
CgT-GAN (ours)	**24.1**	**22.6**	**48.2**	**64.9**	**16.1**

Table 4: Performance changes with different rewards on MSCOCO test split under the ZS-IC (with CC3M captions) and the In-UIC (with MSCOCO captions) settings.

Init.	f_D	r_*	MSCOCO ↔ CC3M					MSCOCO ↔ MSCOCO				
			B.-4	M.	R.	C.	S.	B.-4	M.	R.	C.	S.
✓			2.7	11.1	25.6	12.6	5.6	6.5	14.3	33.0	24.8	7.9
✓	✓		4.9	10.3	28.5	13.6	4.8	22.1	22.4	47.6	75.6	15.4
✓		✓	4.8	**17.6**	33.3	28.2	**12.8**	11.3	22.4	41.3	46.4	16.5
	✓	✓	9.8	16.6	34.6	47.1	12.3	**30.4**	26.3	54.1	105.8	20.0
✓	✓	✓	**10.9**	16.9	**35.2**	**49.8**	12.5	30.3	**26.9**	**54.5**	**108.1**	**20.5**

4.3 Ablation Study (RQ2)

The reward function $R(I, C)$ is the key of CgT-GAN, consisting of two components: the naturalness score f_D and the semantic guidance reward r_* (Eq. (12)). In this subsection, we perform an ablation study to explore the influence of various combinations of rewards and rewarding strategies. The experiments are conducted on two distinct tasks: MSCOCO images with noisy descriptions (CC3M) and images descriptions in the same domain (MSCOCO).

Naturalness reward and semantic guidance reward. As shown in Table 4, the use of a single reward, either the naturalness f_D or the semantic guidance reward r_* results in less satisfactory performance. However, their combination significantly enhances the performance, validating our intention to jointly strengthen language naturalness and visual-language alignment. This finding confirms that both rewards achieve their intended goals and complement each other for image captioning. Moreover, the initialization stage produces a fairly good generator, enhancing adversarial training (Init. + f_D + r_*) compared to that without init (f_D + r_*).

Table 5: Performance changes with different rewarding strategies on MSCOCO test split under the ZS-IC (with CC3M captions) and the In-UIC (with MSCOCO captions) settings.

Strategy	MSCOCO ↔ CC3M					MSCOCO ↔ MSCOCO				
	B.-4	M.	R.	C.	S.	B.-4	M.	R.	C.	S.
CLIP-cos	7.9	16.3	33.1	39.7	11.4	22.9	24.7	49.3	88.9	18.9
CLIP-agg	**10.9**	16.9	**35.2**	**49.8**	12.5	**30.3**	**26.9**	**54.5**	**108.1**	**20.5**
Reward-mix	10.5	**17.0**	34.8	49.0	**12.7**	28.7	26.4	53.1	103.8	20.3

Rewarding Strategy. In our main experiments, we selected the CLIP-agg strategy as the default option. We also tested another strategy, CLIP-cos, as described in Section 3.3. Additionally, we averaged the two rewards with equal weight to create a new strategy, Reward-mix. The results presented in Table 5 show that the CLIP-agg strategy outperforms the other two strategies. Furthermore, we notice slower convergence and inferior performance when applying CLIP-cos, because of the modality gap[29] between visual and language modalities, which reduces the efficiency of CLIP-cos reward. While the CLIP-agg guides the generator with a textual embedding, eliminating the modality gap. Another interesting finding is that the Reward-mix strategy achieves competitive performance to CLIP-agg, even better SPICE. We speculate CLIP-cos may provide reliable guidance when the corpus is noisy or out-of-domain. For further experiments, please refer to the appendix.

4.4 Comparasion with Simpler CLIP Utilization and Web-scale Generative Training (RQ3)

As CLIP is a powerful image-text alignment model, there are simpler methods that adopt CLIP to tackle the unpaired image captioning task. We consider two CLIP baselines: **(1) CLIP-retrieval**, where the predicted caption for each image in the test set is the corpus sentence with the highest CLIP similarity; **(2) CLIP-pseudo**, where pseudo labels are generated for each image in the training set using CLIP-retrieval on the text corpus and training is conducted using these pseudo labels. Besides, generative models pre-trained on web-collected noisy image-text pairs also seem to meet the "without human labelling" requirement. Thus, we select **(3) SimVLM** [64] as a comparison, a transformer-based visual language model (VLM) generatively trained on billions of web-collected image-caption pairs[23]. Since SimVLM infers in a zero-shot manner, we compare it with CgT-GAN on ZS-IC setting.

We compare CgT-GAN with the above baselines in Table 6. CgT-GAN obtains much better performance than simpler CLIP-based methods: CLIP-retrieval and CLIP-pseudo. Moreover, CgT-GAN outperforms the SimVLM trained on 1.8B web-collected image-caption pairs. The comparison shows our effective usage of CLIP and unpaired data.

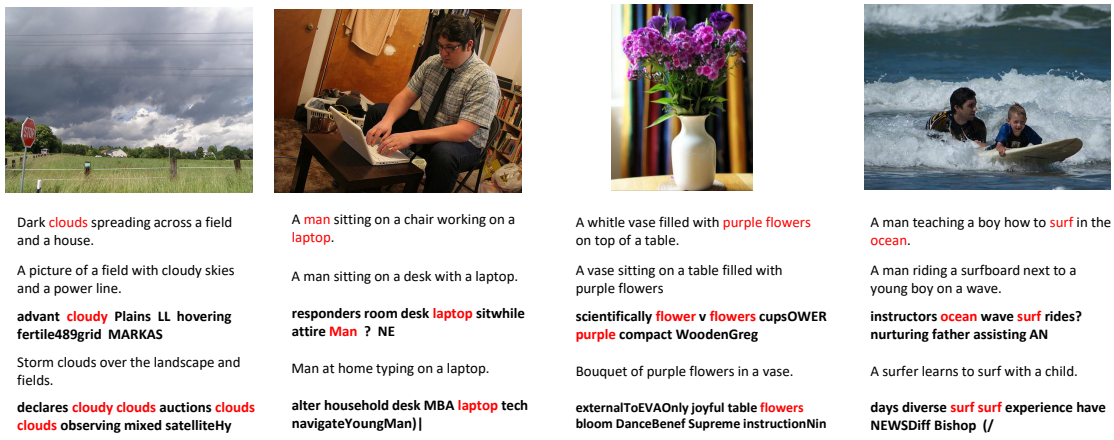


Figure 3: Prompt explanations and corresponding predicted captions on MSCOCO test split. (CC3M) and (MSCOCO) denote training with CC3M and MSCOCO captions, respectively. Red fonts show that visual prompts perceive the main image content.

Table 6: Performance comparison with generative pre-training methods and simpler CLIP-based methods using MSCOCO dataset (test set) under the ZS-IC (with CC3M captions) and the In-UIC (with MSCOCO captions) settings.

Method	MSCOCO ↔ CC3M					MSCOCO ↔ MSCOCO				
	B.-4	M.	R.	C.	S.	B.-4	M.	R.	C.	S.
CLIP-retrieval	5.5	13.8	27.7	31.0	10.0	13.1	20.8	40.7	58.0	15.2
CLIP-pseudo	9.7	15.4	**35.4**	38.9	10.2	19.7	22.5	47.2	74.1	16.3
SimVLM[64]	**11.2**	14.7	—	32.2	8.5	—	—	—	—	—
CgT-GAN	10.9	**16.9**	35.2	**49.8**	**12.5**	**30.3**	**26.9**	**54.5**	**108.1**	**20.5**

Table 7: Performance changes with variant scale backbones using MSCOCO dataset (test set) under In-UIC setting.

Method	Backbone	B.-4	M.	R.	C.	S.
ESPER[69]	ViT-B/32	21.9	21.9	—	78.2	—
CLOSE[17]	ViT-B/32	—	—	—	91.1	—
DeCap[27]	ViT-B/32	24.7	25.0	—	91.2	18.7
CgT-GAN	ViT-B/32	**27.4**	**25.1**	**52.0**	**96.9**	**18.9**
CapDec[42]	R50×4	26.4	25.1	51.8	91.8	—
CLOSE[17]	R50×4	—	—	—	92.0	—
CgT-GAN	R50×4	**27.2**	**25.5**	**52.3**	**99.9**	**19.1**
CLOSE[17]	ViT-L/14	28.6	25.2	—	95.4	18.1
CgT-GAN	ViT-L/14	**30.3**	**26.9**	**54.5**	**108.1**	**20.5**

4.5 CgT-GAN with Variant Backbones (RQ4)

Current advanced CLIP-based state-of-the-arts employ CLIP encoders with varying scales, making it challenging to conduct a fair performance comparison. To ensure a fairer comparison, we conduct additional experiments to evaluate CgT-GAN with varying CLIP backbones. Table 7 summarizes the performance of CgT-GAN and CLIP-based baseline methods with different CLIP backbones. The results show that CgT-GAN performs better when CLIP backbone scales up. Moreover, our proposed CgT-GAN outperforms CLIP-based SOTAs with the same backbone.

4.6 Visual Prompts Explanation

Visual prompts $\{\boldsymbol{p}_i\}_{i=1}^{k}$ are computed from the image embedding and suit for the GPT-2 sentence generation model. In other words, visual prompts are expected to work as semantic tokens to make GPT-2 generate visual-semantic consistent captions. Here, we try to understand how close the visual prompts are to the real word embeddings. For this, we compute the cosine similarity (similar to ClipCap [40]) of a visual prompt and a real word embedding (from GPT-2 dictionary) and select the closest word for observation. Results are shown in figure 3. It can be found that visual prompts surprisingly align the main image content with concept words, like "clouds", "flowers", "surf" and "laptop". Additional case studies are provided in the appendix.

5 CONCLUSION

In this paper, we have presented the CLIP-guided text GAN (CgT-GAN), which utilizes the CLIP to guide image-to-caption generation. CLIP in this paper is not only used for image encoding but also for semantic guidance. CgT-GAN allows the generator to "see" real images during training but does not require any human-annotated image-caption pairs. In CgT-GAN, we examine two types of CLIP-based semantic guidance rewards to enhance caption generating, including the cosine similarity reward CLIP-cos and the newly proposed text embedding aggregation reward CLIP-agg. The CLIP-based reward is finally combined with the GAN's reward to guide the generator learning in a simple and effective manner. Through extensive experiments, CgT-GAN outperforms all competing methods in three subtasks. We also showcase that the visual prompts can correspond to the salient features in the image, thereby revealing how the generator works. We want to mention here that the proposed learning fashion may also be incorporated into other GAN networks, which will be our future work.

6 ACKNOWLEDGMENTS

This work was supported by the National Key Research and Development Program of China under Grant 2020YFB1406703.

REFERENCES

[1] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2016. Spice: Semantic propositional image caption evaluation. In *ECCV*. 382–398.

[2] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*. 6077–6086.

[3] Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*. 65–72.

[4] Manuele Barraco, Marcella Cornia, Silvia Cascianelli, Lorenzo Baraldi, and Rita Cucchiara. 2022. The unreasonable effectiveness of CLIP features for image captioning: an experimental analysis. In *CVPR Workshop*. 4662–4670.

[5] Huixia Ben, Yingwei Pan, Yehao Li, Ting Yao, Richang Hong, Meng Wang, and Tao Mei. 2021. Unpaired image captioning with semantic-constrained self-learning. *TMM* 24 (2021), 904–916.

[6] Shan Cao, Gaoyun An, Zhenxing Zheng, and Qiuqi Ruan. 2020. Interactions guided generative adversarial network for unsupervised image captioning. *Neurocomputing* 417 (2020), 419–431.

[7] Chen Chen, Shuai Mu, Wanpeng Xiao, Zexiong Ye, Liesi Wu, and Qi Ju. 2019. Improving image captioning with conditional generative adversarial nets. In *AAAI*. 8142–8150.

[8] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, Wei Liu, and Tat-Seng Chua. 2017. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In *CVPR*. 5659–5667.

[9] Jaemin Cho, Seunghyun Yoon, Ajinkya Kale, Franck Dernoncourt, Trung Bui, and Mohit Bansal. 2022. Fine-grained image captioning with clip reward. In *Findings of NAACL*. 517–527.

[10] Bo Dai, Sanja Fidler, Raquel Urtasun, and Dahua Lin. 2017. Towards diverse and natural image descriptions via a conditional gan. In *ICCV*. 2970–2979.

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*.

[12] Yang Feng, Lin Ma, Wei Liu, and Jiebo Luo. 2019. Unsupervised image captioning. In *CVPR*. 4125–4134.

[13] Jiahui Gao, Yi Zhou, LH Philip, Shafiq Joty, and Jiuxiang Gu. 2022. UNISON: Unpaired Cross-Lingual Image Captioning. In *AAAI*. 10654–10662.

[14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Commun. ACM* 63, 11 (2020), 139–144.

[15] Jiuxiang Gu, Shafiq Joty, Jianfei Cai, and Gang Wang. 2018. Unpaired image captioning by language pivoting. In *ECCV*. 503–519.

[16] Jiuxiang Gu, Shafiq Joty, Jianfei Cai, Handong Zhao, Xu Yang, and Gang Wang. 2019. Unpaired image captioning via scene graph alignments. In *ICCV*. 10323–10332.

[17] Sophia Gu, Christopher Clark, and Aniruddha Kembhavi. 2022. I Can't Believe There's No Images! Learning Visual Tasks Using only Language Data. *arXiv preprint arXiv:2211.09778* (2022).

[18] Dan Guo, Yang Wang, Peipei Song, and Meng Wang. 2021. Recurrent relational memory network for unsupervised image captioning. In *IJCAI*. 920–926.

[19] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2021. CLIPScore: A Reference-free Evaluation Metric for Image Captioning. In *EMNLP*. 7514–7528.

[20] Ukyo Honda, Yoshitaka Ushiku, Atsushi Hashimoto, Taro Watanabe, and Yuji Matsumoto. 2021. Removing Word-Level Spurious Alignment between Images and Pseudo-Captions in Unsupervised Image Captioning. In *EACL*. 3692–3702.

[21] Jonathan Huang, Vivek Rathod, Chen Sun, Menglong Zhu, Anoop Korattikara, Alireza Fathi, Ian Fischer, Zbigniew Wojna, Yang Song, Sergio Guadarrama, et al. 2017. Speed/accuracy trade-offs for modern convolutional object detectors. In *CVPR*. 7310–7311.

[22] Ajay Jain, Ben Mildenhall, Jonathan T Barron, Pieter Abbeel, and Ben Poole. 2022. Zero-shot text-guided object generation with dream fields. In *CVPR*. 867–876.

[23] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*. 4904–4916.

[24] Chen Ju, Tengda Han, Kunhao Zheng, Ya Zhang, and Weidi Xie. 2022. Prompting Visual-Language Models for Efficient Video Understanding. In *ECCV*. 105–124.

[25] Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *CVPR*. 3128–3137.

[26] Iro Laina, Christian Rupprecht, and Nassir Navab. 2019. Towards unsupervised image captioning with shared multimodal embeddings. In *ICCV*. 7414–7424.

[27] Wei Li, Linchao Zhu, Longyin Wen, and Yi Yang. 2023. DeCap: Decoding CLIP Latents for Zero-Shot Captioning via Text-Only Training. In *ICLR*.

[28] Yehao Li, Yingwei Pan, Ting Yao, and Tao Mei. 2022. Comprehending and ordering semantics for image captioning. In *CVPR*. 17990–17999.

[29] Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. 2022. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. In *NeurIPS*. 17612–17625.

[30] Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *ACL*. 74–81.

[31] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *ECCV*. 740–755.

[32] Fenglin Liu, Meng Gao, Tianhao Zhang, and Yuexian Zou. 2019. Exploring semantic relationships for image captioning without parallel data. In *ICDM*. 439–448.

[33] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* (2019).

[34] Zhenguang Liu, Kedi Lyu, Shuang Wu, Haipeng Chen, Yanbin Hao, and Shouling Ji. 2021. Aggregated multi-gans for controlled 3d human motion prediction. In *AAAI*. 2225–2232.

[35] Zhenguang Liu, Pengxiang Su, Shuang Wu, Xuanjing Shen, Haipeng Chen, Yanbin Hao, and Meng Wang. 2021. Motion prediction using trajectory cues. In *ICCV*. 13299–13308.

[36] Zhenguang Liu, Shuang Wu, Shuyuan Jin, Shouling Ji, Qi Liu, Shijian Lu, and Li Cheng. 2022. Investigating pose representations and motion contexts modeling for 3D motion prediction. *TPAMI* 45, 1 (2022), 681–697.

[37] Ilya Loshchilov and Frank Hutter. 2018. Decoupled Weight Decay Regularization. In *ICLR*.

[38] Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. 2017. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In *CVPR*. 375–383.

[39] Zihang Meng, David Yang, Xuefei Cao, Ashish Shah, and Ser-Nam Lim. 2022. Object-Centric Unsupervised Image Captioning. In *ECCV*. 219–235.

[40] Ron Mokady, Amir Hertz, and Amit H Bermano. 2021. Clipcap: Clip prefix for image captioning. *arXiv preprint arXiv:2111.09734* (2021).

[41] Medhini Narasimhan, Anna Rohrbach, and Trevor Darrell. 2021. CLIP-It! language-guided video summarization. In *NeurIPS*. 13988–14000.

[42] David Nukrai, Ron Mokady, and Amir Globerson. 2022. Text-Only Training for Image Captioning using Noise-Injected CLIP. In *EMNLP findings*. 4055–4063.

[43] Yingwei Pan, Ting Yao, Yehao Li, and Tao Mei. 2020. X-linear attention networks for image captioning. In *CVPR*. 10971–10980.

[44] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. In *ACL*. 311–318.

[45] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. 2021. Styleclip: Text-driven manipulation of stylegan imagery. In *ICCV*. 2085–2094.

[46] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2015. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*. 2641–2649.

[47] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *ICML*. 8748–8763.

[48] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog* 1, 8 (2019), 9.

[49] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125* (2022).

[50] Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *CVPR*. 7008–7024.

[51] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*. 2556–2565.

[52] Haoyu Song, Li Dong, Weinan Zhang, Ting Liu, and Furu Wei. 2022. CLIP Models are Few-Shot Learners: Empirical Studies on VQA and Visual Entailment. In *ACL*. 6088–6100.

[53] Peipei Song, Dan Guo, Jinxing Zhou, Mingliang Xu, and Meng Wang. 2022. Memorial GAN With Joint Semantic Optimization for Unpaired Image Captioning. *TCyber* (2022).

[54] Yuqing Song, Shizhe Chen, Yida Zhao, and Qin Jin. 2019. Unpaired cross-lingual image caption generation with self-supervised rewards. In *ACM MM*. 784–792.

[55] Yixuan Su, Tian Lan, Yahui Liu, Fangyu Liu, Dani Yogatama, Yan Wang, Lingpeng Kong, and Nigel Collier. 2022. Language models can see: plugging visual controls in text generation. *arXiv preprint arXiv:2205.02655* (2022).

[56] Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

[57] Mingkang Tang, Zhanyu Wang, Zhenhua Liu, Fengyun Rao, Dian Li, and Xiu Li. 2021. Clip4caption: Clip for video caption. In *ACM MM*. 4858–4862.

[58] Yoad Tewel, Yoav Shalev, Idan Schwartz, and Lior Wolf. 2022. ZeroCap: Zero-Shot Image-to-Text Generation for Visual-Semantic Arithmetic. In *CVPR*. 17918–17928.

[59] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*. 5998–6008.

[60] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *CVPR*. 4566–4575.

[61] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2016. Show and tell: Lessons learned from the 2015 mscoco image captioning challenge. *TPAMI* 39, 4 (2016), 652–663.

[62] Wenjie Wang, Xinyu Lin, Fuli Feng, Xiangnan He, and Tat-Seng Chua. 2023. Generative recommendation: Towards next-generation recommender paradigm. *arXiv preprint arXiv:2304.03516* (2023).

[63] Zhaoqing Wang, Yu Lu, Qiang Li, Xunqiang Tao, Yandong Guo, Mingming Gong, and Tongliang Liu. 2022. Cris: Clip-driven referring image segmentation. In *CVPR*. 11686–11695.

[64] Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. 2022. SimVLM: Simple Visual Language Model Pretraining with Weak Supervision. In *ICLR*.

[65] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*. 5288–5296.

[66] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*. 2048–2057.

[67] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. 2018. Exploring visual relationship for image captioning. In *ECCV*. 684–699.

[68] Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2017. Seqgan: Sequence generative adversarial nets with policy gradient. In *AAAI*.

[69] Youngjae Yu, Jiwan Chung, Heeseung Yun, Jack Hessel, JaeSung Park, Ximing Lu, Prithviraj Ammanabrolu, Rowan Zellers, Ronan Le Bras, Gunhee Kim, et al. 2022. Multimodal Knowledge Alignment with Reinforcement Learning. *arXiv preprint arXiv:2205.12630* (2022).

[70] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. 2018. Neural motifs: Scene graph parsing with global context. In *CVPR*. 5831–5840.

[71] Yucheng Zhou, Wei Tao, and Wenqiang Zhang. 2021. Triple sequence generative adversarial nets for unsupervised image captioning. In *ICASSP*. 7598–7602.

[72] Bin Zhu and Chong-Wah Ngo. 2020. CookGAN: Causality based text-to-image synthesis. In *CVPR*. 5519–5527.

[73] Bin Zhu, Chong-Wah Ngo, Jingjing Chen, and Yanbin Hao. 2019. R2gan: Cross-modal recipe retrieval with generative adversarial network. In *CVPR*. 11477–11486.

[74] Peipei Zhu, Xiao Wang, Yong Luo, Zhenglong Sun, Wei-Shi Zheng, Yaowei Wang, and Changwen Chen. 2022. Unpaired Image Captioning by Image-level Weakly-Supervised Visual Concept Recognition. *TMM* (2022), 1–15.

[75] Peipei Zhu, Xiao Wang, Lin Zhu, Zhenglong Sun, Wei-Shi Zheng, Yaowei Wang, and Changwen Chen. 2023. Prompt-based learning for unpaired image captioning. *TMM* (2023), 1–15.

APPENDIX

A IMPLEMENTATION DETAILS

We select GPT-2 model [48] provided by huggingface[2] as the generator. The number of visual prompts is $k = 10$. The MLP of the generator has 2 layers, where the channel numbers of the hidden layer and the output layer are 3840 and 7680, respectively. The 7680 dimension output is reshaped to 10×768 as visual prompt vectors. For the discriminator, we employ RoBERTa-base with a pooler layer[3] as our RoBERTa[33] model. The MLP of the discriminator also has 2 layers, where the layer sizes are 384 and 1, respectively. All MLPs in our implementation take *tanh* as the activate function. We optimize our CgT-GAN by AdamW[37] with $\epsilon = 10^{-8}$, $\beta = (0.9, 0.999)$ and weight decay = 0.05 on weights. The learning rate of the generator and the discriminator is set to 10^{-5}. We set 150 warmup steps for the generator while the discriminator has no warmup steps. The batch-size is set to 128 for MSCOCO image dataset and 32 for Flickr30k image dataset. The mean of the policy gradient is estimated by sampling 5 times from the generator. For both sampling and inference, the beam size of the generator is set to 1. In practice, we train our CgT-GAN with the reward from the discriminator only for the early 150 steps, *i.e.*, only using $f_D(C)$ for training. Then we linearly increase the ratio of $r_*(I, C)$ until it reaches the ratio of $f_D(C)$ in the following 2350 steps.

For the generator initialization stage, the batch-size is set to 16 for Flickr30k captions and 32 for other text corpora. The training configuration is set as: learning rate 2×10^{-5}, AdamW optimizer with the same configs with GAN training, and 5000 steps warm-up to stabilize the training.

The current implementation focuses on image captioning. We want to mention here that the CLIP (or other cross-modal alignment models)-guided GAN framework could be extended to various generative multimodal applications [22, 35, 36, 45, 57, 72], particularly when dealing with situations where paired data is unavailable. That will be the future work. The CgT-GAN is trained on 2×A40 GPUs, while the initialization is runing on a single A40 GPU. We use the official COCO evaluation tools[4] to calculate all metrics.

B MORE EXPERIMENTAL ANALYSIS

B.1 CLIP-agg Components and Temperature.

Since we set CLIP-agg as the default strategy, we further analyzed each component and temperature influence in the CLIP-agg reward. The CLIP-agg reward is composed of two components, cosine similarity and L_1 penalty, as expressed in Eq. (9). Cosine similarity encourages paired embeddings to have similar semantics, consistent with the CLIP training objective. On the other hand, L_1 distance penalty brings caption and aggregative embeddings closer to each other in Euclidean space. Our analysis, presented in Table 8, shows that the combination performs best and the two components are complementary. The performance also varies with temperature τ, as shown in Figure 4. The temperature balances the diversity and accuracy of the supporting embeddings. A larger τ considers broader text embeddings, while a smaller τ gives more attention to closer embeddings in the aggregation. Therefore, when using a web-crawled corpus, the CLIP-agg strategy prefers a higher τ to increase the accuracy of the aggregated embedding.

Table 8: Performance changes with different CLIP-agg components on MSCOCO test split under the In-UIC setting.

cosine similarity	L_1 penalty	B.-4	M.	R.	C.	S.
✓		30.2	26.6	53.9	107.3	20.3
	✓	29.4	26.8	53.7	105.8	**20.5**
✓	✓	**30.3**	**26.9**	**54.5**	**108.1**	**20.5**

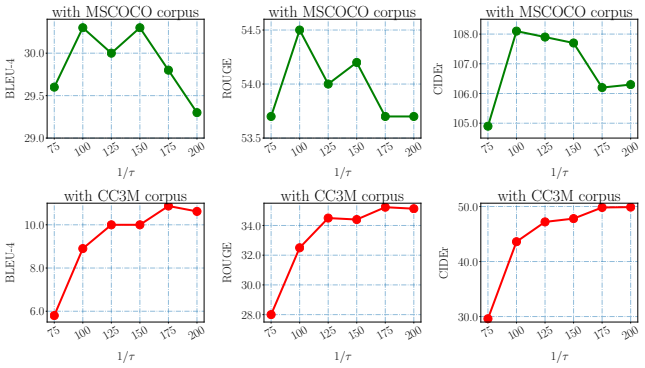


Figure 4: Performance changes with different CLIP-agg temperatures on MSCOCO test split using CC3M (red lines) and MSCOCO captions (green lines).

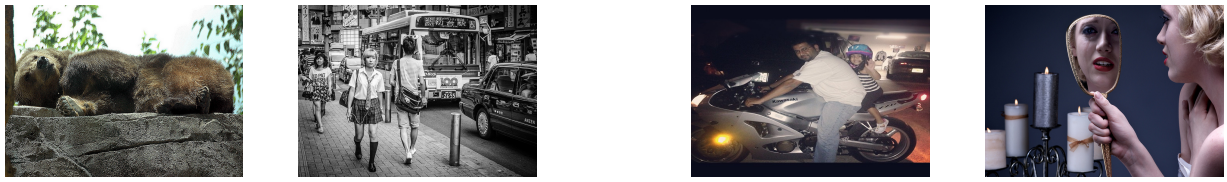
Table 9: Performance changes on MSCOCO test split with different corpora (CC3M and MSR-VTT) and different rewarding strategies.

Strategy	MSCOCO ↔ CC3M					MSCOCO ↔ MSR-VTT				
	B.-4	M.	R.	C.	S.	B.-4	M.	R.	C.	S.
CLIP-cos	7.9	16.3	33.1	39.7	11.4	8.1	**17.4**	36.3	38.2	11.5
CLIP-agg	**10.9**	16.9	**35.2**	**49.8**	12.5	10.7	16.9	38.6	44.4	11.7
Reward-mix	10.5	**17.0**	34.8	49.0	**12.7**	**11.3**	17.2	**39.0**	**47.2**	**11.8**

B.2 Robustness to Corpus Domain Variation

During our CgT-GAN training stage, the generator is instructed to mimic the sentences in the corpus. However, there might be a large gap between the image distribution and the corpus distribution in a specific task, which has the potential to impair performance. As a result, it is important to evaluate the robustness of CgT-GAN in the face of corpus domain variations. We conducted additional experiments using the MSCOCO images ↔ MSR-VTT [65] captions setting to simulate such situations. MSR-VTT [65] is a video captioning dataset that primarily focuses on describing actions and events within the videos. Therefore, it has a significantly different caption distribution compared to the MSCOCO image dataset,

[2]https://huggingface.co/docs/transformers/v4.21.1/en/model_doc/gpt2#transformers.GPT2LMHeadModel
[3]https://huggingface.co/docs/transformers/model_doc/roberta#transformers.RobertaModel
[4]https://github.com/tylin/coco-caption

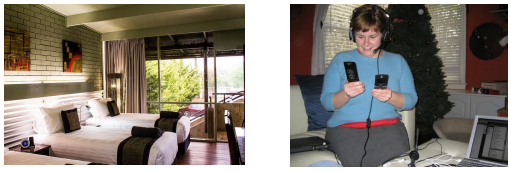


GroundTruth	A large brown bear laying on top of a giant rock.	A group of people walking on a street by some cars.	The man is giving the young girl a ride on his motorcycle.	A crying woman looking at herself in a mirror.
DeCap(CC3M)	**a bear, a common animal, is resting in the medium photo**	**Japanese city : street photography of person on the streets**	**DeCap (MSCOCO)** **A man and a woman riding a motorcycle together.**	**A woman looking into a mirror with her teeth.**
CgT-GAN (CC3M)	**A grizzly bear sleeps on a rock in a natural park.**	**Young girl and man walking on the street.**	The father is riding a motorcycle with his daughter.	Sad woman looking at herself in a mirror.
CgT-GAN (MSCOCO)	A brown bear laying on top of a large rock.	A girl walking down a street past a man.	**A man and a little girl sitting on a motorcycle.**	**A woman is holding her reflection in a mirror.**

Figure 5: Caption examples on MSCOCO test split of CgT-GAN. CgT-GAN (CC3M) and CgT-GAN (MSCOCO) denote training with CC3M and MSCOCO captions, respectively. Boldface fonts in the first two cases show the comparison between DeCap and our CgT-GAN with CC3M corpus. Those in the last two show comparison between CgT-GAN and Decap with MSCOCO corpus.

Table 10: Parameter count comparison with other methods. CIDEr scores are obtained under MSCOCO In-UIC setting.

Method	Encoder Version	Encoder Params.	Generator Params.	CIDEr
ESPER [69]	ViT-B/32	88M	156M	78.2
CgT-GAN	ViT-B/32	88M	156M	96.9
CapDec[42]	R50×4	87M	182M	91.8
CgT-GAN	R50×4	87M	156M	99.9
CLOSE[17]	ViT-L/14	304M	225M	95.4
CgT-GAN	ViT-L/14	304M	157M	108.1



GroundTruth	**Three beds** all sitting in a row up against a brick wall.	A woman has headphones on and is holding **two cell phones**.
Prediction	A bedroom with two beds sitting next to each other.	A woman sitting and holding a cell phone in her hands.

Figure 6: Failure cases on MSCOCO test split of CgT-GAN using In-UIC setting.

which mainly consists of static visual content. Results presented in Table 9 show that CgT-GAN achieves comparable results with two diverse corpora (video captioning dataset MSR-VTT and image captioning dataset CC3M), indicating the robustness of our method in adapting to multiple corpora, even in the presence of distribution gaps. Notably, Reward-mix achieves the highest performance in the MSCOCO images ↔ MSR-VTT captions setting, suggesting that the combination of CLIP-cos and CLIP-agg exhibits high robustness to the distribution gap.

B.3 Parameter Count Comparison

We calculate the parameters of CgT-GAN essential components in the *inference stage*, i.e., the image encoder (CLIP-ImageEncoder) and the generator (MLP + GPT2). The comparison of these parameters is presented in Table 10. From the results, it can be observed that our CgT-GAN has either fewer or equivalent model parameters compared to other methods, yet it achieves significantly better performance (CIDEr).

B.4 Case Study

We show four typical examples in Figure 5 to qualitatively compare the caption results. As can be seen, our CgT-GAN outperforms DeCap in terms of mimicking human language, especially when trained with CC3M. In the first two examples, CgT-GAN (CC3M) successfully describes the image content and produces more fluent captions than DeCap (CC3M), which indicates that adversarial learning leads to better performance than text-only methods on noisy corpora. Moreover, we discover CgT-GAN is observant to identify details and spatial relations, as shown in the last two cases, where CgT-GAN recognizes the "little girl" and comprehends that the "woman" in the mirror is a reflection. By comparing CgT-GAN (MSCOCO) and CgT-GAN (CC3M), we observe that CgT-GAN (CC3M) is more contextually imaginative, like "natural park" (the first case), "daughter" (the third case) and "sad" (the last case) due to the diverse training text. We also present the typical failure cases in Figure 6, which provide insights into the potential limitation of our proposed CgT-GAN. Through the analysis of the generated captions under the In-UIC setting, it can be found that CgT-GAN encounters challenges in accurately counting objects in some cases. One possible reason is that CLIP-based visual embedding primarily focuses on high-level semantics.