Fine-grained Image Captioning with CLIP Reward

Jaemin Cho¹ Seunghyun Yoon² Ajinkya Kale³ Franck Dernoncourt²
Trung Bui² Mohit Bansal¹
¹UNC Chapel Hill ²Adobe Research ³Adobe Inc.

{jmincho, mbansal}@cs.unc.edu {syoon, akale, franck.dernoncourt, bui}@adobe.com

Abstract

Modern image captioning models are usually trained with text similarity objectives. However, since reference captions in public datasets often describe the most salient common objects, models trained with the text similarity objectives tend to ignore specific and detailed aspects of an image that distinguish it from others. Towards more descriptive and distinctive caption generation, we propose to use CLIP, a multi-modal encoder trained on huge image-text pairs from the web, to calculate multi-modal similarity and use it as a reward function. We also propose a simple finetuning strategy of CLIP text encoder to improve grammar that does not require extra text annotation. This completely eliminates the need for reference captions during the reward computation. To comprehensively evaluate descriptive captions, we introduce FineCapEval, a new dataset for caption evaluation with fine-grained criteria: overall, background, object, relations. In our experiments on text-to-image retrieval and FineCapEval, the proposed CLIP-guided model generates more distinctive captions than the CIDEr-optimized model. We also show that our unsupervised grammar finetuning of the CLIP text encoder alleviates the degeneration problem of the naive CLIP reward. Lastly, we show human analysis where the annotators strongly prefer CLIP reward to CIDEr and MLE objectives on diverse criteria.

1 Introduction

Describing an image with its detailed, distinguishing aspects is crucial for many applications, such as creating text keys for image search engine and accessibility for the visual impaired. The standard deep learning approaches train an image-conditioned language model by maximizing the textual similarity between generated and reference captions (Vinyals et al., 2015; Xu et al., 2015; Rennie et al., 2017; Anderson et al., 2018). However, the reference captions of public datasets often describe only the most salient objects in images. This makes models trained to maximize textual similarity with reference captions tend to generate less distinctive captions that ignore the fine detailed aspects of an image that distinguishes it from others.

To alleviate the problem, we propose to use CLIP (Radford et al., 2021), a multi-modal encoder model trained on large image-text data (mostly English) collected from the web, by using its similarity scores (Hessel et al., 2021) as rewards (Sec. 3.1). In addition, we propose a CLIP text encoder fine-tuning strategy with synthetic negative caption augmentation to improve the grammar of captioning model, without any extra text annotations (Sec. 3.2). Note that our approach completely eliminates the need for reference captions during reward computation. To comprehensively evaluate descriptive captions, we also introduce FineCapEval, a new dataset that measures captioning in diverse aspects: overall, background, object, and relation between objects (Sec. 4).

In our experiments on MS COCO (Lin et al., 2014) dataset, we show that the captions from models trained with CLIP reward are more distinctive and contain more detailed information compared to
the captions from CIDEr (Vedantam et al., 2015)-optimized models. The CLIP-guided captions even achieve the higher text-to-image retrieval performance than reference captions that are originally paired with images. We also show that our text encoder finetuning significantly improves caption grammars by removing degeneration artifacts such as word repetition. In fine-grained caption evaluation with FineCapEval and human analysis, we show our CLIP based rewards outperform text similarity objectives by a large margin on all categories.

2 Related Works

Image Captioning Metrics. Traditionally, captions have been evaluated with n-gram or scene-graph based similarity metrics such as BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), METEOR (Banerjee and Lavie, 2005), CIDEr (Vedantam et al., 2015), and SPICE (Anderson et al., 2015). However, such metrics often fail to capture paraphrased expressions due to the limited number of reference captions or scene-graphs. To tackle the problem, recent works including BERTScore (Zhang et al., 2019), ViLBERTScore (Lee et al., 2020a), UMIC (Lee et al., 2021), and CLIPScore (Hessel et al., 2021) propose to use relevance scores computed by language or multi-modal models pretrained on large data.

Objectives for Image Captioning. Standard deep learning based image captioning approaches train models with maximum likelihood estimation (MLE) objective. Ranzato et al. (2016) point that MLE suffers from exposure bias problem. To tackle exposure bias, Bengio et al. (2015) propose a curriculum learning strategy called scheduled sampling. Ranzato et al. (2016) propose to train models by directly maximizing the textual similarity between generated and reference captions with REINFORCE (Williams, 1992). Rennie et al. (2017); Luo (2020) propose self-critical sequence training (SCST) approach by normalizing rewards to stabilize its high variance.

Recent studies have observed that reference-trained captioning models often neglect important information from images (Dai et al., 2017; Wang et al., 2017). Lee et al. (2020b) use a visual question answering model’s accuracy as a reward, encouraging models to generate captions that include information sufficient to answer a visual question. Dai and Lin (2017); Luo et al. (2018); Liu et al. (2018) use image-text retrieval model’s self-retrieval score as a reward and combine them with n-gram based metrics, encouraging captioning models to generate captions that are distinctive to each input image.

Note that these works require careful balancing between self-retrieval and text similarity objectives for stable training. In contrast, by finetuning CLIP text encoder (Sec. 3.2), our approach removes the need of reference caption and text similarity metrics for reward computation.

3 Methods

3.1 CLIP-guided Image Captioning

We propose to use the relevance score between image and text calculated by CLIP (Radford et al., 2021). Following Hessel et al. (2021), we use CLIP-S as our reward: $CLIP-S(I, c) = w \max(f(I), f^T(c)), 0$ where $I, c$ are image and caption, $f^T$ are CLIP’s image and text encoders, and $w$ is set to 2.5. By maximizing the multimodal similarity of CLIP, which is a contrastively trained model, image captioning models are encouraged to generate captions that contain more distinctive information about the input image. Fig. 1 (a) illustrates this training strategy.

Following Rennie et al. (2017), we optimize our captioning model $P_b(c|I)$ with REINFORCE (Williams, 1992) with self-critical baseline. We approximate the gradient of the expected reward for generated caption $\hat{c}$, where rewards from beam search are normalized with the baseline rewards $b$ from the greedy decoding $\hat{c}_{\text{greedy}}$: $\nabla_{\theta} \mathbb{E}_{\hat{c} \sim P_b(c|I)}[R(I, \hat{c})] \approx (R(I, \hat{c}_{\text{beam}}) - R(I, \hat{c}_{\text{greedy}})) \nabla_{\theta} \log P_b(\hat{c}_{\text{beam}}|I)$ where $R(I, c) = CLIP-S(I, c)$.

3.2 Improving Grammar with CLIP Text Encoder Finetuning

Since CLIP is not trained with a language modeling objective, the captioning model trained with CLIPS reward often generates grammatically incorrect (e.g., repeated words) captions (See Table 3). We inject grammatical knowledge to CLIP’s text encoder with synthetic negative captions, generated by randomly repeating/removing/inserting/swapping/shuffling tokens of the reference captions. We provide the implementation details of such operations in appendix. We introduce a 2-layer per-
We introduce FineCapEval, a new dataset for caption evaluation in four different aspects. To construct FineCapEval, we collect 500 images from the MS COCO (Lin et al., 2014) test2015 split and Conceptual Caption (Sharma et al., 2018) val split, respectively. Then, for each image, we ask 5 human annotators to write phrases of 1) background, 2) objects (and their attributes; i.e., color, shape, etc.), 3) relation between objects (i.e., spatial relation), and 4) a detailed caption that includes all three aspects. See details of data collection process in appendix. In total, FineCapEval consists of 1,000 images with 5,000 annotations for each of the 4 criteria. In Table 1, we show samples of FineCapEval dataset.

5 Experiments
We train CLIP-Res50Transformer captioning model (Shen et al., 2022) with different rewards: MLE, CIDEr, CLIP-S, CIDER+CLIP-S, CLIP-S+Grammar. Following previous works, we conduct experiment on MS COCO (Lin et al., 2014) English captioning dataset with Karpathy split (Karpathy and Fei-Fei, 2015). We evaluate the model with n-gram based metrics, embedding based metrics, text-to-image retrieval scores, and FineCapEval. We also conduct human evaluation with five criteria to understand the human preference of the generated captions in diverse aspects.

Model Architecture and Training. We use the CLIP-Res50Transformer (Shen et al., 2022) as our captioning model architecture. The model consists of CLIP-Res50 for visual feature extraction and a transformer encoder-decoder for conditional language model. We resize images in 224x224 to extract 2048-dimensional visual features. The transformer consists of 6-layer encoder and 6-layer decoder. We train our the model with MLE objective for 15 epochs and further train with different rewards for 25 epochs (total 40 epochs), which takes within 1 day with 8 V100 GPUs. We use beam size 5 for beam search decoding. We implement a training pipeline with PyTorch (Paszke et al., 2017), PyTorch Lightning3, and HuggingFace Transformers (Wolf et al., 2020).

N-gram based Metrics. For N-gram based metrics, we report BLEU-4 (Papineni et al., 2002), CIDEr (Vedantam et al., 2015) METEOR (Banerjee and Lavie, 2005), and ROUGE-L (Lin, 2004).

Embedding-based Metrics. We report BERT-S (Zhang et al., 2019) and CLIP-S/RefCLIP-S (Hesse et al., 2022).
Table 2: Performance on MS COCO Karpathy test split. *The first caption out of 5 reference captions are used to calculate retrieval scores. R@K refers to the recall-K of the reference image. Rword refers to the word-level recall for background (Bg.), object (Obj.) and relation (Rel.) criteria (see Sec. 4 for details).

Table 3 shows that the degeneration (e.g., repeating words) of CLIP-S reward is successfully mitigated by adding the grammar reward (CLIP-S+Grammar). Table 2 shows that adding grammar reward significantly increases all text similarity metrics (e.g., +60 for CIDEr).

6.3 Fine-grained Caption Evaluation

FineCapEval. The rightmost four columns of Table 2 show that the captions with CLIP-S and CLIP-S+Grammar significantly outperform the captions with CIDEr on all four criteria of FineCapEval: overall, background, object, relation. The gap is smallest in object criterion, which implies MS COCO reference captions describe more object information than background or relation between objects.

Human Evaluation. Table 4 shows human evaluation results on five criteria: overall, background,
Table 3: Captions generated by models with different rewards on MS COCO Karpathy test split images.

<table>
<thead>
<tr>
<th>Criteria</th>
<th>CLIP-S + Grammar</th>
<th>Win</th>
<th>Lose</th>
<th>Tie</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall</td>
<td>v.s. MLE</td>
<td>49.0</td>
<td>41.8</td>
<td>9.2</td>
</tr>
<tr>
<td></td>
<td>v.s. CIDEr</td>
<td>51.0</td>
<td>30.8</td>
<td>18.2</td>
</tr>
<tr>
<td>Background</td>
<td>v.s. MLE</td>
<td>52.8</td>
<td>35.0</td>
<td>12.2</td>
</tr>
<tr>
<td></td>
<td>v.s. CIDEr</td>
<td>53.9</td>
<td>25.4</td>
<td>20.6</td>
</tr>
<tr>
<td>Object</td>
<td>v.s. MLE</td>
<td>52.0</td>
<td>36.6</td>
<td>11.4</td>
</tr>
<tr>
<td></td>
<td>v.s. CIDEr</td>
<td>55.2</td>
<td>32.8</td>
<td>12.0</td>
</tr>
<tr>
<td>Attribute</td>
<td>v.s. MLE</td>
<td>57.2</td>
<td>36.8</td>
<td>6.0</td>
</tr>
<tr>
<td></td>
<td>v.s. CIDEr</td>
<td>55.8</td>
<td>37.2</td>
<td>7.0</td>
</tr>
<tr>
<td>Relation</td>
<td>v.s. MLE</td>
<td>44.6</td>
<td>44.2</td>
<td>11.2</td>
</tr>
<tr>
<td></td>
<td>v.s. CIDEr</td>
<td>49.2</td>
<td>39.6</td>
<td>11.2</td>
</tr>
</tbody>
</table>

Table 4: Human pairwise preference evaluation results.

We introduce a novel training strategy for image captioning models by maximizing multimodal similarity score of CLIP and finetuning its text encoder to improve grammar. The use of CLIP reward eliminates the need for reference captions and their bias for reward computation. We also introduce FineCapEval, a dataset for fine-grained caption evaluation. We demonstrate the effectiveness of our proposed method based on improvements in text-to-image retrieval, FineCapEval, and human evaluation on fine-grained criteria along with qualitative examples. Future works involve finetuning CLIP reward models with desired writing styles for different applications and improving the synthetic augmentation process by using external data suitable for grammars with advanced linguistics expertise.

8 Ethical Considerations

The CLIP models we used are trained on millions of image-text pairs collected from the web. Birhane et al. (2021) shows that such large-scale datasets often contain problematic and explicit image-text pairs. As the CLIP model card suggests, using CLIP reward for training image captioning models is intended as a research output, and any deployed use case of the models is out of scope.

Our captioning models and CLIP models are trained on English datasets; its use should be limited to English language use cases. As our proposed method is not limited to English and easily extended to other languages, future work will explore the extensions in various languages.

Acknowledgements

We thank the reviewers for their valuable comments. This work was partially done while JC was interning at Adobe Research and later extended at UNC, where it was supported by ARO Award W911NF2110220, DARPA MCS Grant N66001-19-2-4031, and NSF-CAREER Award 1846185. The views contained in this article are those of the authors and not of the funding agency.

---

6https://github.com/openai/CLIP/blob/main/model-card.md
References


Bo Dai, Sanja Fidler, Raquel Urtasun, and Dahua Lin. 2017. Towards Diverse and Natural Image Descriptions via a Conditional GAN. In ICCV.


Xihui Liu, Hongsheng Li, Jing Shao, Dapeng Chen, and Xiaogang Wang. 2018. Show, tell and discriminate: Image captioning by self-retrieval with partially labeled data. In ECCV.


Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. 2015. CIDEr: Consensus-based Image Description Evaluation. In CVPR.


C FineCapEval Details

Data Collection. To create a fine-grained description of the image, we ask annotators to write a caption that should describe target images’ 1) background, 2) objects and their attributes (i.e., color, shape, etc.), and 3) the relationship between the objects if any (i.e., spatial relation). Furthermore, we ask the annotators to write metadata containing which words/phrases in their writing belong to the three criteria. We also provide annotators with guidelines in writing a caption as follows: 1) There should be a single sentence describing the image. 2) The image may be a photo, an illustration or a pure background. 3) Pay close attention to local and global events in the image. 4) Descriptions should be at least ten words for each image. 5) Avoid the subject description of the image (i.e., a dog runs “very fast”), a man feels “successful”). 6) Avoid known entities such as specific locations (i.e., Eiffel Tower), time (i.e., 4 pm), event (i.e., Halloween), proper name. 7) In describing people, use only man/woman/boy/girl if clear; otherwise, use person/child. All annotators are hired by a professional crowdsourcing platform TELUS. The crowdsourcing company obtained consents from the crowdsourcing company obtained consents from the crowdworkers before the annotation process and conducted the ethical reviews. We collect English captions and all the annotators are native English speakers living in the US. We pay 5,400 USD, including 1) caption creation (5k samples) and 2) quality assurance process that manually examines 50% of the created caption by different workers.

Word-level Recall $R_{word}$. In Alg. 2, we show Python implementation of word-level recall $R_{word}$. In summary, $R_{word}$ measures how many words from each of the reference phrases are included in a generated caption on average.

D Human Evaluation Details

We conduct pairwise evaluation of human preference, as shown in the Sec. 5. For each image, we show two captions generated from two models: ours (CLIP-S + Grammar) and the baseline (MLE/CIDEr). A human worker selects a caption that better describes the image in terms of five criteria: overall, background, object, attribute, and relation. For each criterion, we use 50 images from FineCapEval, and the two options are randomly and evenly shuffled. We also provide ‘Tie’ option
Algorithm 1 Python implementation of negative text generation (main paper Sec. 3.2)

```python
from random import randint, choice, shuffle

def repeat(tokens, n_max_gram=3, n_max_repeat=3): # repeat n-grams
    n_gram = randint(1, n_max_gram)
    repeat_idx = randint(0, len(tokens) - n_gram)
    repeated = tokens[repeat_idx:repeat_idx+n_gram]
    n_repeat = randint(1, n_max_repeat)
    for _ in range(n_repeat):
        insert_idx = randint(0, len(tokens))
        tokens = tokens[:insert_idx]+repeated+tokens[insert_idx:]
    return tokens

def remove(tokens, n_max_gram=3): # remove n-grams
    n_gram = randint(1, n_max_gram)
    remove_idx = randint(0, len(tokens) - n_gram)
    tokens = tokens[:remove_idx] + tokens[remove_idx + n_gram:]
    return tokens

def insert(tokens, vocab, n_max_tokens=3): # insert tokens
    n_insert_token = randint(1, n_max_tokens)
    for _ in range(n_insert_token):
        insert_idx = randint(0, len(tokens) - 1)
        insert_tok = choice(vocab)
        tokens = tokens[:insert_idx]+[insert_tok]+tokens[insert_idx:]
    return tokens

def swap(tokens, vocab, n_max_tokens=3): # swap tokens
    n_swap_tokens = randint(1, n_max_tokens)
    for _ in range(n_swap_tokens):
        swap_token_idx = randint(0, len(tokens) - 1)
        swap_token = choice(vocab)
        while swap_token == tokens[swap_token_idx]:
            swap_token = choice(vocab)
        tokens[swap_token_idx] = swap_token
    return tokens

def _shuffle(tokens): # shuffle tokens
    shuffle(tokens)
    return tokens

def generate_negative_text(text, vocab): # main function
    tokens = text.split()
    neg_type = choice(['repeat', 'remove', 'insert', 'swap', 'shuffle'])
    if neg_type == 'repeat': tokens = repeat(tokens)
    elif neg_type == 'remove': tokens = remove(tokens)
    elif neg_type == 'insert': tokens = insert(tokens, vocab)
    elif neg_type == 'swap': tokens = swap(tokens, vocab)
    elif neg_type == 'shuffle': tokens = _shuffle(tokens)
    return " ".join(tokens)
```

Algorithm 2 Python implementation of word-level recall $R_{\text{word}}$ computation (main paper Sec. 5)

```python
def calculate_word_recall(pred_id2sent, gt_id2phrases):
    """
    pred_id2sent: dict of generated captions (dict[int, str])
    gt_id2phrases: dict of reference phrases (dict[int, list[str]])
    """
    n_total = 0
    total_score = 0
    for id, gt_phrases in gt_id2phrases.items():
        pred_sent = pred_id2sent[id]
        score = 0
        for gt_phrase in gt_phrases:
            word_score = 0
            for gt_word in gt_phrase.split():
                if gt_word in pred_sent:
                    word_score += 1
            score += word_score / len(gt_phrase.split())
        score /= len(gt_phrases)
        total_score += score
        n_total += 1
    word_recall = total_score / n_total * 100
    return word_recall
```

Figure 2: The screenshot of human evaluation process for ‘object’ criterion (main paper Sec. 5).
Table 5: More captions generated by models with different rewards on MS COCO Karpathy test split images.

<table>
<thead>
<tr>
<th>Image</th>
<th>Reward</th>
<th>Captions</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a)</td>
<td>CIDEr</td>
<td>a group of boats parked in the water on a lake</td>
</tr>
<tr>
<td></td>
<td>CLIP-S</td>
<td>several rows of boats parked near a canal mountains horizon area and a mountain horizon area horizon ear motion</td>
</tr>
<tr>
<td></td>
<td>CLIP-S+Grammar</td>
<td>a lot of boats parked on the grass next to the lake with the hills behind</td>
</tr>
<tr>
<td></td>
<td>Reference Captions</td>
<td>A blue boat docked on a green lush shore. A small marina with boats docked there a group of boats sitting together with no one around Some boats parked in the water at a dock boats sitting around the side of a lake by a tree</td>
</tr>
<tr>
<td>(b)</td>
<td>CIDEr</td>
<td>a zebra standing in the snow next to a brick wall</td>
</tr>
<tr>
<td></td>
<td>CLIP-S</td>
<td>a adult zebra wearing black and grey stripes standing near a brick wall area with grey stance position stance</td>
</tr>
<tr>
<td></td>
<td>CLIP-S+Grammar</td>
<td>a large black and grey zebra standing together in the snowy ground next to a stone</td>
</tr>
<tr>
<td></td>
<td>Reference Captions</td>
<td>A zebra is standing outside in the snowOne zebra standing in snow near a stone wallA zebra is standing in a snowy fieldA zebra stands in snow in front of a wallA zebra standing alone in the snow with a stone block wall and wooden fence behind it</td>
</tr>
<tr>
<td>(c)</td>
<td>CIDEr</td>
<td>a black dog sitting next to a plate of food</td>
</tr>
<tr>
<td></td>
<td>CLIP-S</td>
<td>black black dog with macaroni and macaroni plate with pasta and pasta on a wooden floor plate position position position</td>
</tr>
<tr>
<td></td>
<td>CLIP-S+Grammar</td>
<td>a black dog sitting next to a plate of food on the wood floor</td>
</tr>
<tr>
<td></td>
<td>Reference Captions</td>
<td>Shaggy dog gets dinner served on a plate.A small black dog standing over a plate of food.A small dog eating a plate of broccoli.A black dog being given broccoli to eatThere is a dog staring at a plate of food</td>
</tr>
<tr>
<td>(d)</td>
<td>CIDEr</td>
<td>two elephants standing next to a tree in a zoo</td>
</tr>
<tr>
<td></td>
<td>CLIP-S</td>
<td>two adult and baby elephant near a tree enclosure area with a tree area enclosure motion stance ear stance</td>
</tr>
<tr>
<td></td>
<td>CLIP-S+Grammar</td>
<td>a large elephant playing with a tree in the dirt field with rocks behind it</td>
</tr>
<tr>
<td></td>
<td>Reference Captions</td>
<td>An elephant standing under the shade of a treeAn elephant standing in the middle of a rocky environmentAn elephant is alone in a wooded enclosureAn elephant standing in a shaded cleaning in a wooded areaAn elephant walks alone past some big rocks boulders in an open field</td>
</tr>
<tr>
<td>(e)</td>
<td>CIDEr</td>
<td>a group of people riding bikes down a city street</td>
</tr>
<tr>
<td></td>
<td>CLIP-S</td>
<td>several cyclists moving and bicycles near a restaurant and a blue advertisement outside a red brick building motion stance p</td>
</tr>
<tr>
<td></td>
<td>CLIP-S+Grammar</td>
<td>a group of people riding their bikes on the busy street with a blue sign</td>
</tr>
<tr>
<td></td>
<td>Reference Captions</td>
<td>people on bicycles ride down a busy streetA group of people are riding bikes down the street in a bike lane bike riders passing Burger King in city streetA group of bicyclists are riding in the bike laneBicyclists on a city street, most not using the bike lane</td>
</tr>
<tr>
<td>(f)</td>
<td>CIDEr</td>
<td>a man riding a bike next to a train</td>
</tr>
<tr>
<td></td>
<td>CLIP-S</td>
<td>older adult male riding a bicycle near a red and commuter train passing a train station motion stance ear stance</td>
</tr>
<tr>
<td></td>
<td>CLIP-S+Grammar</td>
<td>a person walking on a bike next to a red passenger train on the road</td>
</tr>
<tr>
<td></td>
<td>Reference Captions</td>
<td>A man on a bicycle riding next to a train A man is riding a bicycle but there is a train in the backgroundA red and white train and a man riding a bicycle a guy that is riding his bike next to a trainA man riding a bike past a train traveling along tracks</td>
</tr>
<tr>
<td>(g)</td>
<td>CIDEr</td>
<td>a window of an airport with planes on the runway</td>
</tr>
<tr>
<td></td>
<td>CLIP-S</td>
<td>several rows of planes parked outside a terminal window area with fog outside a terminal window motion position area motions</td>
</tr>
<tr>
<td></td>
<td>CLIP-S+Grammar</td>
<td>a lot of airplanes parked on a wet airport terminal</td>
</tr>
<tr>
<td></td>
<td>Reference Captions</td>
<td>An airport filled with planes sitting on tarmacs. The view of runway from behind the windows of airportPlanes on a wet tarmac unloading at arrival gatesWindow view from the inside of airplanes, baggage carrier and tarmac</td>
</tr>
</tbody>
</table>

to choose when the two captions are equally good or bad. For each criterion, we recruit 10 annotators 1) who are located in the Great Britain or the United States 2) HIT approval rate above 80% and 3) Number of HITs approved greater than 1000, from Amazon Mechanical Turk. We pay the annotators 0.03 USD per selection, which roughly corresponds to 11 USD/hour. In Fig. 2, we provide the screenshot for ‘object’ criterion for example.

E  Licenses

For all artifacts, we remain within their respective license agreements. Here, we list the licenses:

- **MS COCO - CC 4.0** - [https://cocodataset.org/#termsofuse](https://cocodataset.org/#termsofuse)
- **Conceptual Captions** - [https://github.com/google-research-datasets/](https://github.com/google-research-datasets/)
conceptual-captions/blob/
master/LICENSE

- CLIP-ViL - MIT - https://github.com/clip-vil/CLIP-ViL/blob/master/LICENSE