Simple Weakly-Supervised Image Captioning via CLIP's Multimodal Embeddings

Derek Tam **Colin Raffel** **Mohit Bansal**

UNC Chapel Hill
{dtredsox, craffel, mbansal}@cs.unc.edu

Abstract

CLIP (Radford et al. 2021) enables strong performance in zero-shot image classification and other single-modality tasks through multi-modal pre-training. Recently, ClipCap (Mokady, Hertz, and Bermano 2021) demonstrated how the vision encoder of CLIP could be fed into GPT-2 to perform image captioning. In this work, we propose `WS-ClipCap`, which extends ClipCap to perform weakly-supervised image captioning by training only on the text from image captions. During training, `WS-ClipCap` encodes image captions using CLIP's text encoder. Then, during inference, `WS-ClipCap` encodes images using CLIP's vision encoder. Due to CLIP's joint embedding space for different modalities, the image and text representations are similar and can be interchanged. `WS-ClipCap` outperforms MAGIC (Su et al. 2022) substantially (which trains only on textual image captions) and performs on par with ESPER (Yu et al. 2022) (which trains only on images) while being significantly simpler than both. We also analyze how the performance of `WS-ClipCap` is affected by the distribution shift between CLIP's multi-modal embeddings and investigate several ways of correcting the distribution mismatch.

Introduction

The CLIP model (Radford et al. 2021) showed how contrastive multimodal pre-training can be used to produce shared embeddings for vision and text in a joint embedding space. Recently, Mokady, Hertz, and Bermano (2021) introduced ClipCap, which combines the vision encoder of CLIP and the text decoder of GPT-2 (Radford et al. 2019) to perform image captioning. ClipCap is trained through supervised text-caption pairs, which can be expensive to collect. The fact that CLIP aims to produce text and image embeddings in a shared space suggests the possibility that ClipCap could be trained on text alone. To explore this possibility, we propose `WS-ClipCap` which extends ClipCap to weakly-supervised image captioning by leveraging the joint embedding space of CLIP. Specifically, `WS-ClipCap` trains on unpaired image captions only by using CLIP's text encoder to get a (multi-modal) representation of text and then using GPT-2 to reconstruct a different matching image caption that corresponds to the same image. During inference, CLIP's vision encoder is used to get a multi-modal representation of an image and then GPT-2 is used to generate the image caption. Our method relies on the fact that CLIP's vision and language encoder map to a shared embedding space, which implies that the text encoder and vision encoder should be interchangeable between training and inference (as shown in fig. 1). Our approach enables training ClipCap with only unpaired textual image captions.

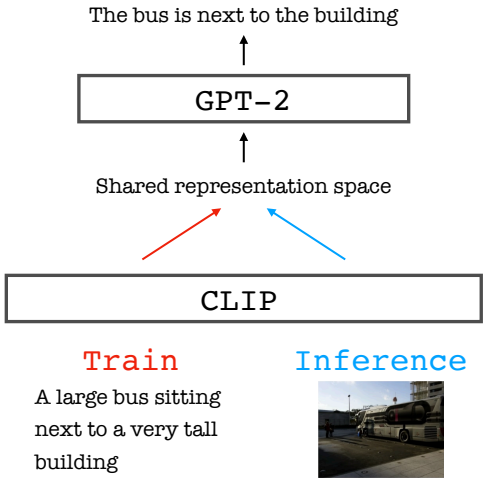

Figure 1: `WS-ClipCap` is trained by feeding in captions into CLIP and then decoding the text. During inference, it feeding in the image and decodes the text, leveraging CLIP's joint embedding space to swap encoders during training and inference.

Experimentally, `WS-ClipCap` performs strongly compared to contemporaneous methods for weakly-supervised/weakly-supervised image captioning. `WS-ClipCap` outperforms MAGIC (Su et al. 2022) substantially, which also trains on unpaired image captions only. This improvement could be thanks to the fact that `WS-ClipCap` leverages additional supervision from matching captions (i.e. paraphrases) that correspond to the same image, which MAGIC cannot use. `WS-ClipCap` also performs on par with ESPER (Yu et al. 2022), which trains on images only, while being much simpler without requiring any reinforcement learning to train.

Though `WS-ClipCap` attains reasonable weakly-

supervised image captioning performance, it does not match the performance of supervised image captioning with ClipCap. In line with recent work (Liang et al. 2022), this suggests that CLIP's representations are not truly multimodal. We therefore perform additional analysis to understand the distribution mismatch between image/caption pairs. We experiment with three methods for correcting the distribution mismatch: aligning their means, aligning via rotations, and aligning via optimal transport. Though none of these methods can correct the distribution mismatch, aligning the means performs the best and reduces the distribution mismatch slightly.

Related Work

ClipCap. Our work is primarily based off ClipCap (Mokady, Hertz, and Bermano 2021) which combines CLIP and GPT-2 by connecting the vision encoder of CLIP and the text decoder of GPT-2 using prompt tuning (Lester, Al-Rfou, and Constant 2021). CLIP (Radford et al. 2021) trains a vision and language encoder on images and their captions using a contrastive loss that aims to images and text to a shared embedding space. GPT-2 (Radford et al. 2019) is an auto-regressive generative language model that was trained on text scraped from the web.

Leveraging the Multimodality of CLIP. Nukrai, Mokady, and Globerson (2022) concurrently propose CapDec, which is very similar to `WS-ClipCap` but CapDec also injects noise into CLIP's representations when training. While CapDec focuses on correcting the modality mismatch during later fine-tuning by injecting noise, we focus on aligning the mismatched distributions between image/-caption pairs. Song et al. (2022) also propose a method with similar motivation to `WS-ClipCap` that leverages the multimodality of CLIP to swap the text encoder during training with the image encoder during inference. However, they work on a different task of visual entailment and therefore use a completely different architecture based on CLIP.

Learning from Limited Labeled Examples. Tewel et al. (2021) propose ZeroCap for zero shot image-to-text generation by using gradient information during inference. Frozen (Tsimpoukelli et al. 2021) combines a vision model and a pre-trained language model for few-shot learning, but does not leverage the multimodality of any particular model. There have been several works proposed for unsupervised image captioning from images only (Yu et al. 2022) and weakly-supervised image captioning from image captions only (Su et al. 2022). Most of these methods leverage CLIP to compute a similarity score between images and generated text as supervision for the model while we focus on the joint embedding space of CLIP.

CLIP's embeddings. There have been several recent works analyzing CLIP's embeddings. Liang et al. (2022) showed CLIP's text and image embeddings lie in two different cones that are separated by a gap. So et al. (2022) propose closing the gap in embeddings between the different modalities and improving zero-shot retrieval accuracy by finetuning CLIP with Mixup of image and text representations.

`WS-ClipCap`

Before presenting `WS-ClipCap`, we provide a detailed description of ClipCap, the method upon which `WS-ClipCap` is based. Let f_t and f_v be the text and vision encoders of the CLIP model respectively, f_g be the GPT-2 model, and assume we are given an image-text pair (x, y), where x is an image, and y is the image caption consisting of tokens $y_0, y_1, ...y_n$. ClipCap first computes the representation of an image using CLIP: $f_v(x)$. Then, it adds an MLP h which is trained to project CLIP's representation $f_v(x)$ of the paired image x into the same space as GPT's text embeddings. The loss is shown in eq. (1) where CE corresponds to the cross entropy loss. The parameters in CLIP are frozen while the parameters in GPT-2 and the MLP are updated.

$$p(y_i|x) = f_g\Big(h\big(f_v(x)\big), y_{<i}\Big)$$
$$L(x, y) = \sum_{i=1}^{n} CE(p(y_i|x), y_i) \quad (1)$$

`WS-ClipCap` is largely similar to ClipCap, with the primary difference being that we train using an weakly-supervised objective rather than a supervised cross-entropy loss. During training, `WS-ClipCap` encodes a given (unpaired) image caption with CLIP's text encoder to get a representation, feeds it through an MLP, and then decodes a different caption corresponding to the same image using GPT-2. Note that different captions corresponding to the same image are available in the image captioning datasets since they annotate each image with multiple captions. Let y' be a different image caption than y that also corresponds to the image x. The loss is otherwise the same except for the modified objective as shown in eq. (2)

$$p(y_i|x) = f_g\Big(h\big(f_t(y')\big), y_{<i}\Big) \quad (2)$$

During inference, we encode a query image with CLIP's image encoder to get an image representation and then decode the image representation using GPT-2. Because CLIP's image and text encoders produce embeddings in a joint space, the representation for an image and its corresponding caption should ideally be similar and therefore decoding from GPT-2 should produce the same output whether it is fed with the image or caption embedding. Crucially, CLIP's weights are frozen so the text representation won't be updated during training. This prevents the text representations from drifting away from the image representations.

Results

We run experiments for image captioning on MS-COCO (Lin et al. 2014) and Flickr30k (Hodosh, Young, and Hockenmaier 2013) following the Karparthy split. We use the standard evaluation metrics from COCOEvalCap.[1] We compare against the following baselines:

[1] https://github.com/tylin/coco-caption

(a) Training (b) Inference

Figure 2: We illustrate the training (fig. 2a) and inference (fig. 2b) of `WS-ClipCap`. During training, an image is fed in through CLIP's vision encoder and projected into a prompt embeddings for GPT-2 while during inference, an image caption is fed in through CLIP's text encoder and projected into a prompt embeddings for GPT-2.

Method	BLEU@1	BLEU@2	BLEU@3	BLEU@4	METEOR	ROUGE-L	CIDEr	SPICE
MAGIC (Su et al. 2022)*	56.8	–	–	12.9	17.4	39.9	49.3	11.3
ESPER (Yu et al. 2022)*	–	–	–	21.9	21.9	–	**78.2**	–
`WS-ClipCap`	65.5	46.7	32.1	**22.1**	**22.2**	48.0	74.6	14.9
– matched captions	50.3	30.0	17.0	9.6	15.2	37.5	33.7	8.6
ClipCap (Mokady, Hertz, and Bermano 2021)	74.0	57.2	42.7	31.5	26.8	54.7	106.6	19.9

Table 1: Image captioning results on MS-COCO. ClipCap is trained on supervised data. `WS-ClipCap` and MAGIC train on image captions only. ESPER trains on images only. ZeroCap does not have any training. * indicates reported numbers from paper.

Method	BLEU@1	BLEU@2	BLEU@3	BLEU@4	METEOR	ROUGE-L	CIDEr	SPICE
MAGIC (Su et al. 2022)	44.5	–	–	6.4	13.1	31.6	20.4	7.1
`WS-ClipCap`	**53.2**	35.9	23.7	**15.7**	**19.3**	**41.8**	**36.5**	**12.9**
ClipCap (Mokady, Hertz, and Bermano 2021)	68.0	49.6	35.2	24.8	22.2	48.6	57.9	15.8

Table 2: Image captioning results on Flickr30k. ClipCap is trained on supervised data. `WS-ClipCap` and MAGIC train on image captions only.

- MAGIC (Su et al. 2022), which trains a language model on the image captions and then steers the decoding during inference based on the CLIP similarity of the image and generated tokens.

- ESPER (Yu et al. 2022), which uses reinforcement learning to train a model on images only using CLIP's similarity of the image and generated text samples as a reward.

The MS-COCO results are shown in table 1 and the Flickr30k results are shown in table 2. Overall, we see `WS-ClipCap` outperforms other weakly-supervised image captioning methods which train on captions only and performs on par with ESPER which trains on images only and requires more complicated training with reinforcement learning. However, it still lags a bit behind ClipCap which is trained with full supervision. We also see that removing matched captions significantly decreases the performance of `WS-ClipCap`. Under this scenario, `WS-ClipCap` is trained by just encoding and decoding the same caption. Because `WS-ClipCap` uses CLIP directly in the model (in contrast to other unsupervised/weakly-supervised image captioning methods which use CLIP indirectly as a scoring function), it benefits from the matched image captions that can improve CLIP's text representations.

Analysis

The fact that `WS-ClipCap` underperforms ClipCap suggests that CLIP's embeddings might not be truly multimodal. We therefore analyze the distribution mismatch between CLIP's image and text representations to see the effect it has on `WS-ClipCap`. To do so, we use Mixup (Zhang et al. 2017; So et al. 2022) to interpolate image/caption embeddings, where given an image representation x_i, text representation x_t, and mixup ratio λ, the interpolated representation is $\lambda x_t + (1-\lambda)x_i$, where λ represents the proportion of the final embeddings coming from the text embedding.

Our results are shown in fig. 3. `WS-ClipCap` evaluated on text does very well, but as we shift the evaluation to images, performance monotonically decreases. This can be due to either 1) the loss of information in the image compared to the image caption or 2) a distribution mismatch between text representation and mixup representation.

We determine the reason by looking at ClipCap evaluated on Mixup representations. ClipCap does well when evaluated on images but does even better on a mixed up image and text representation with $\lambda = 0.5$. Even if we introduce a distribution mismatch by evaluating on interpolated embeddings, the extra information in the image caption nullifies any decrease in performance from the distributional mismatch and actually improves performance. However, this only holds till $\lambda = 0.5$ and afterwards, the effect of the distribution mismatch nullifies any gain from the image caption.

When we evaluate ClipCap on text, the performance drops to roughly the same amount as when using `WS-ClipCap`. This means the distribution mismatch between training and inference caused by the distribution mismatch between individual image caption pairs may be the main cause for performance degradation between ClipCap and `WS-ClipCap`.

Figure 3: Performance of `WS-ClipCap` trained with images or text and evaluated on mixed up image and text representations.

Correcting the Distribution Mismatch

We study three methods for correcting the distribution mismatch between CLIP's image and text representations: aligning the means of the distributions, aligning the representations via rotations, and aligning the representations via optimal transport.

Aligning the means. Previous works (Liang et al. 2022) have shown that CLIP's text and image representations are separated by a modality gap and propose aligning the means to remove this modality gap. Following (Liang et al. 2022), we shift the image embeddings such that they have the same mean as the text embeddings. The average performance of `WS-ClipCap` improves from 40.8 to 42.1 but still does not come close to ClipCap's performance of 52.4.

Aligning via rotations. We also try aligning the representations by rotating them, motivated by PCA. PCA computes principal directions, which are orthogonal directions ordered by how much variance they explain. We want a rotation such that the corresponding principal directions in each modality are aligned. For example, the top principal direction for images should be aligned with the top principal direction for text, the second principal direction for images should be aligned with the second principal direction for text, and so on. Rotating the representations in such a way requires computing the principal component scores, or the magnitude of the representation in each of its principal directions. The principal component scores can be computed efficiently via SVD. Given a matrix $X \in \mathbb{R}^{N \text{x} D}$ where N is the number of image/caption pairs and D is the dimension, SVD computes the left singular, diagonal, and right singular matrix U, D, V respectively where $X = UDV^T$ The principal component score is UD. The average performance of `WS-ClipCap` drops from 40.8 to 15.8 using rotated representations. We hypothesize rotations do not work well since it does not take into account the distribution of points along each direction of variance

Aligning via Optimal Transport. Finally, we consider aligning the representations in each modality via optimal transport. Given two probability measures p_1 and p_2 and a cost matrix C where C_{ij} represents the cost of moving element i in the support of p_1 to element j in the support of p_2, optimal transport computes a transport plan T where

$$T^* = \arg\min_T \sum_{i=0}^{|p_1|} \sum_{j=0}^{|p_2|} T_{ij} C_{ij}$$

For the cost matrix, we use the L2 distance between the representations where C_{ij} represents the distance between the i^{th} image representation and the j^{th} text representation. We assume uniform probability measures p_1 and p_2. To align the image representations, we multiply them by T^*. The average performance of `WS-ClipCap` drops from 40.8 to 0.0, where the model generates non-sensible text. We hypothesize that optimal transport does not work in aligning the representations since the cost matrix alone, which is all optimal transport knows about the distribution mismatch, does not transmit enough information about the distribution mismatch.

Overall, we find that though none of the three methods proposed can correct the distribution mismatch completely, aligning the means can improve performance slightly.

Conclusion

We present `WS-ClipCap` for weakly-supervised image captioning based on ClipCap. `WS-ClipCap` leverages the multimodality of CLIP by using the text encoder during training and the vision encoder during inference. Despite its simplicity, `WS-ClipCap` outperforms other weakly-supervised image captioning methods trained on text and performs on par with other unsupervised image captioning methods trained on images. However, `WS-ClipCap` does not perform as well as ClipCap, which we conclude to be due to the distribution mismatch between image/caption pairs. We experiment with three ways to correct the distribution mismatch including via rotations, optimal transport, and shifting the means. We find that shifting the means of the different modalities to align the means performs the best in reducing the distribution mismatch. Future directions include: (1) correcting the distribution mismatch between image/captions pairs such that `WS-ClipCap` can perform on par with ClipCap, (2) pre-training multimodal embeddings to not have a distribution mismatch, and (3) applying `WS-ClipCap` to multimodal models where the structure of representations for each modality differ (i.e. an image is represented with one embedding while an image caption is represented with a set of token embeddings).

References

Hodosh, M.; Young, P.; and Hockenmaier, J. 2013. In *Framing image description as a ranking task: Data, models and evaluation metrics.*

Lester, B.; Al-Rfou, R.; and Constant, N. 2021. The Power of Scale for Parameter-Efficient Prompt Tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 3045–3059. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics.

Liang, W.; Zhang, Y.; Kwon, Y.; Yeung, S.; and Zou, J. 2022. Mind the Gap: Understanding the Modality Gap in Multi-modal Contrastive Representation Learning.

Lin, T.-Y.; Maire, M.; Belongie, S.; Bourdev, L.; Girshick, R.; Hays, J.; Perona, P.; Ramanan, D.; Zitnick, C. L.; and Dollár, P. 2014. Microsoft COCO: Common Objects in Context. In *European Conference on Computer Vision*.

Mokady, R.; Hertz, A.; and Bermano, A. H. 2021. ClipCap: CLIP Prefix for Image Captioning.

Nukrai, D.; Mokady, R.; and Globerson, A. 2022. Text-Only Training for Image Captioning using Noise-Injected CLIP. *arXiv preprint arXiv:2211.00575*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; Krueger, G.; and Sutskever, I. 2021. Learning Transferable Visual Models From Natural Language Supervision.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; and Sutskever, I. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8): 9.

So, J.; Oh, C.; Shin, M.; and Song, K. 2022. Multi-Modal Mixup for Robust Fine-tuning.

Song, H.; Dong, L.; Zhang, W.-N.; Liu, T.; and Wei, F. 2022. CLIP Models are Few-shot Learners: Empirical Studies on VQA and Visual Entailment.

Su, Y.; Lan, T.; Liu, Y.; Liu, F.; Yogatama, D.; Wang, Y.; Kong, L.; and Collier, N. 2022. Language Models Can See: Plugging Visual Controls in Text Generation. *arXiv preprint arXiv:2205.02655*.

Tewel, Y.; Shalev, Y.; Schwartz, I.; and Wolf, L. 2021. Zero-shot image-to-text generation for visual-semantic arithmetic. *arXiv preprint arXiv:2111.14447*.

Tsimpoukelli, M.; Menick, J.; Cabi, S.; Eslami, S. M. A.; Vinyals, O.; and Hill, F. 2021. Multimodal Few-Shot Learning with Frozen Language Models.

Yu, Y.; Chung, J.; Yun, H.; Hessel, J.; Park, J.; Lu, X.; Ammanabrolu, P.; Zellers, R.; Bras, R. L.; Kim, G.; et al. 2022. Multimodal Knowledge Alignment with Reinforcement Learning. *arXiv preprint arXiv:2205.12630*.

Zhang, H.; Cisse, M.; Dauphin, Y. N.; and Lopez-Paz, D. 2017. mixup: Beyond Empirical Risk Minimization.