ClipCap: CLIP Prefix for Image Captioning

Ron Mokady* Amir Hertz* Amit H. Bermano
The Blavatnik School of Computer Science, Tel Aviv University

2 contributions —>
(1) lighter model (frozen CLIP & GPT2)
(2) fine-tuned model

Abstract

Image captioning is a fundamental task in vision-language understanding, where the model predicts a textual informative caption to a given input image. In this paper, we present a simple approach to address this task. We use CLIP encoding as a prefix to the caption, by employing a simple mapping network, and then fine-tunes a language model to generate the image captions. The recently proposed CLIP model contains rich semantic features which were trained with textual context, making it best for vision-language perception. Our key idea is that together with a pre-trained language model (GPT2), we obtain a wide understanding of both visual and textual data. Hence, our approach only requires rather quick training to produce a competent captioning model. Without additional annotations or pre-training, it efficiently generates meaningful captions for large-scale and diverse datasets. Surprisingly, our method works well even when only the mapping network is trained, while both CLIP and the language model remain frozen, allowing a lighter architecture with less trainable parameters. Through quantitative evaluation, we demonstrate our model achieves comparable results to state-of-the-art methods on the challenging Conceptual Captions and nocaps datasets, while it is simpler, faster, and lighter. Our code is available in https://github.com/rmokady/CLIP_prefix_caption.

1. Introduction

In image captioning, the task is to provide a meaningful and valid caption for a given input image in a natural language. This task poses two main challenges. [1] The first is semantic understanding. This aspect ranges from simple tasks such as detecting the main object, to more involved ones, such as understanding the relations between depicted parts of the image. For example, in the top-left image of Fig. 1, the model understands that the object is a gift. [2] The second challenge is the large number of possible ways to describe a single image. In this aspect, the training dataset typically dictates the preferable option for a given image.

*Equal contribution.

A politician receives a gift from politician.

A collage of different colored ties on a white background.

Silhouette of a woman practicing yoga on the beach at sunset.

Aerial view of a road in autumn.

Figure 1. Our ClipCap model produces captions depcting the respective images. Here, the results are of a model that was trained over the Conceptual Captions dataset.

Many approaches have been proposed for image captioning [4, 9, 13, 19, 34, 35, 42, 44, 47]. Typically, these works utilize an encoder for visual cues and a textual decoder to produce the final caption. Essentially, this induces the need to bridge the challenging gap between the visual and textual representations. For this reason, such models are resource hungry. They require extensive training time, a large number of trainable parameters, a massive dataset, and in some cases even additional annotations (such as detection results), which limit their practical applicability.

old works —> separate encoder (visual) and decoder (textual) —> so there's a gap and we need to align them

Excessive training time is even more restrictive for applications that require several training procedures. For instance, training multiple captioning models over various datasets could provide different users (or applications) with different captions for the same image. Additionally, given fresh samples, it is desirable to update the model routinely with the new data. Therefore, a *lightweight* captioning model is preferable. Specifically, a model with faster training times and fewer trainable parameters would be beneficial, especially if it does not require additional supervision.

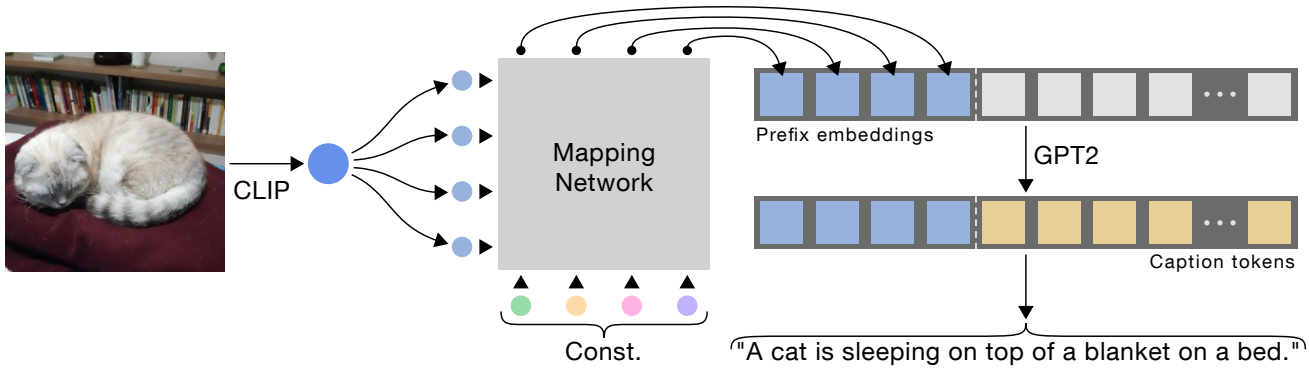


Figure 2. Overview of our transformer-based architecture, enabling the generation of meaningful captions while both CLIP and the language model, GPT-2, are frozen. To extract a fixed length prefix, we train a lightweight transformer-based mapping network from the CLIP embedding space and a learned constant to GPT-2. At inference, we employ GPT-2 to generate the caption given the prefix embeddings. We also suggest a MLP-based architecture, refer to Sec. 3 for more details.

In this paper, we leverage powerful vision-language pre-trained models to simplify the captioning process. More specifically, we use the CLIP (Contrastive Language-Image Pre-Training) encoder, recently introduced by Radford et al. [29]. CLIP is designed to impose a shared representation for both images and text prompts. It is trained over a vast number of images and textual descriptions using a contrastive loss. Hence, its visual and textual representations are well correlated. As we demonstrate, this correlation saves training time and data requirements.

major benefit with CLIP

As illustrated in Fig. 2, our method produces a prefix for each caption by applying a mapping network over the CLIP embedding. This prefix is a fixed size embeddings sequence, concatenated to the caption embeddings. These are fed to a language model, which is fine-tuned along with the mapping network training. At inference, the language model generates the caption word after word, starting from the CLIP prefix. This scheme narrows the aforementioned gap between the visual and textual worlds, allowing the employment of a simple mapping network. To achieve even a lighter model, we introduce another variant of our method, where we train only the mapping network, while both CLIP and the language model are kept frozen. By utilizing the expressive transformer architecture, we successfully produce meaningful captions, while imposing substantially less trainable parameters. Our approach is inspired by Li et al. [20], which demonstrates the ability to efficiently adapt a language model for new tasks by concatenating a learned prefix. We use GPT-2 [30] as our language model, which has been demonstrated to generate rich and diverse texts.

inference is auto-regressive with CLIP embed as the start token

could refer to a transformer-based model, adept at capturing and expressing complex patterns within data.

As our approach exploits the rich visual-textual representation of CLIP, our model requires significantly lower training time. For instance, we train our model on a single Nvidia GTX1080 GPU for 80 hours over the three million samples of the massive Conceptual Captions dataset. Nevertheless, our model generalizes well to complex scenes, as can be seen in Fig. 1 (e.g., practicing yoga on the beach at sunset). We evaluate our method extensively, demonstrating successful realistic and meaningful captions. Even though our model requires less training time, it still achieves comparable results to state-of-the-art approaches over the challenging Conceptual Captions [33] and nocaps [1] datasets, and marginally lower for the more restricted COCO [7, 22] benchmark. In addition, we provide a thorough analysis of the required prefix length and the effect of fine-tuning the language model, including interpretation of our produced prefixes. Overall, our main contributions are as follow:

- A lightweight captioning approach that utilizes pre-trained frozen models for both visual and textual processing.
- Even when the language model is fine-tuned, our approach is simpler and faster to train, while demonstrating comparable results to state-of-the-art over challenging datasets.

2. Related Works

Recently, Radford et al. [29] presented a novel approach, known as CLIP, to jointly represent images and text descriptions. CLIP comprises two encoders, one for visual cues and one for text. It was trained over more than 400 million image-text pairs guided by unsupervised contrastive loss, resulting in rich semantic latent space shared by both visual and textual data. Many works have already used CLIP successfully for computer vision tasks that require the understanding of some auxiliary text, such as generating or editing an image based on a natural language condition [5, 14, 28]. In this paper, we utilize the powerful CLIP model for the task of image captioning. Note that our method does not employ the CLIP's textual encoder, since there is no input text, and the output text is generated by a

language model.

Commonly, image captioning [34] models first encode the input pixels as feature vectors, which are then used to produce the final sequence of words. Early works utilize the features extracted from a pre-trained classification network [6, 9, 13, 42], while later works [4, 19, 47] exploit the more expressive features of an object detection network [31]. Though a pre-trained object detection network is available for the popular COCO benchmark [7, 22], it is not necessarily true for other datasets. This implies that most methods would require additional object detection annotations to operate over new and diverse datasets. To further leverage the visual cues, an attention mechanism is usually utilized [4, 6, 42] to focus on specific visual features. Moreover, recent models apply self-attention [16, 43] or use an expressive visual Transformer [12] as an encoder [23]. Our work uses the expressive embedding of CLIP for visual representation. Since CLIP was trained over an extremely large number of images, we can operate on any set of natural images without additional annotations.

To produce the caption itself, a textual decoder is employed. Early works have used LSTM variants [8, 38, 39], while recent works [16, 26] adopted the improved transformer architecture [36]. Built upon the transformer, one of the most notable works is BERT [11], demonstrating the dominance of the newly introduced paradigm. With this paradigm, the language model is first pre-trained over a large data collection to solve an auxiliary task. Then, the model is fine-tuned for a specific task, where additional supervision is used. As our visual information resides in the prefix, we utilize a powerful auto-regressive language model, GPT-2 [30]. Considering the training loss term, earlier works adopt the effective cross-entropy, while contemporary methods also apply self-critical sequence training [15, 32, 45]. That is, an additional training stage to optimize the CIDEr metric. We deliberately refrain from this optimization to retain a quick training procedure.

Most close to ours, are works that employ vision-and-language pre-training to create a shared latent space of both vision and text [19, 25, 35, 46, 47]. Zhou et al. [47] use visual tokens extracted from object detector as a prefix to caption tokens. The entire model is then pre-trained to perform prediction utilizing the BERT [11] architecture. Li et al. [19] and Zhang et al. [46] also utilize BERT, but require the additional supervision of object tags. Hence, these methods are limited to datasets in which such object detectors or annotations are available. The approach of Wang et al. [40] mitigate the need for supplementary annotations, but still perform an extensive pre-train process with millions of image-text pairs, resulting in a lengthy training time. This exhaustive pre-training step is required to compensate for the lack of joint representation of language and vision, which we inherently obtained by employing CLIP.

3. Method

We start with our problem statement. Given a dataset of paired images and captions $\{x^i, c^i\}_{i=1}^N$, our goal is to learn the generation of a meaningful caption for an unseen input image. We can refer to the captions as a sequence of tokens $c^i = c_1^i, \ldots, c_\ell^i$, where we pad the tokens to a maximal length ℓ. Our training objective is then the following:

$$\max_\theta \sum_{i=1}^{N} \log p_\theta(c_1^i, \ldots, c_\ell^i | \, x^i), \tag{1}$$

where θ denotes the model's trainable parameters. Our key idea is to use the rich semantic embedding of CLIP, which contains, virtually, the essential visual data, as a condition. Following recent works [47], we consider the condition as a prefix to the caption. Since the required semantic information is encapsulated in the prefix, we can utilize an autoregressive language model that predicts the next token without considering future tokens. Thus, our objective can be described as:

$$\max_\theta \sum_{i=1}^{N} \sum_{j=1}^{\ell} \log p_\theta(c_j^i | \, x^i, c_1^i, \ldots, c_{j-1}^i) \tag{2}$$

3.1. Overview

An illustration of our method is provided in Fig. 2. We use GPT-2 (large) as our language model, and utilize its tokenizer to project the caption to a sequence of embeddings. To extract visual information from an image x^i, we use the visual encoder of a pre-trained CLIP [29] model. Next, we employ a light mapping network, denoted F, to map the CLIP embedding to k embedding vectors:

$$p_1^i, \ldots, p_k^i = F(\text{CLIP}(x^i)). \tag{3}$$

Where each vector p_j^i has the same dimension as a word embedding. We then concatenate the obtained visual embedding to the caption c^i embeddings:

$$Z^i = p_1^i, \ldots, p_k^i, c_1^i, \ldots, c_\ell^i. \tag{4}$$

During training, we feed the language model with the prefix-caption concatenation $\{Z^i\}_{i=1}^N$. Our training objective is predicting the caption tokens conditioned on the prefix in an autoregressive fashion. To this purpose, we train the mapping component F using the simple, yet effective, cross-entropy loss:

$$\mathcal{L}_X = -\sum_{i=1}^{N} \sum_{j=1}^{\ell} \log p_\theta(c_j^i | \, p_1^i, \ldots, p_k^i, c_1^i, \ldots, c_{j-1}^i). \tag{5}$$

We now turn to discuss two variants of our method regarding the additional fine-tuning of the language model and their implications.

3.2. Language model fine-tuning

Our main challenge during training is to translate between the representations of CLIP and the language model. Even though both models develop a rich and diverse representation of text, their latent spaces are independent, as they were not jointly trained. Moreover, each captioning dataset incorporates a different style, which may not be natural for the pre-trained language model. Hence, we propose fine-tuning the language model during the training of the mapping network. This provides additional flexibility for the networks and yields a more expressive outcome.

However, fine-tuning the language model naturally increases the number of trainable parameters substantially. Thus, we present an additional variant of our approach, in which we keep the language model fixed during training. Our attempt to adjust a frozen language model is inspired by the work of Li and Liang [20]. In their work, they accommodate such a pre-trained model to an unfamiliar task by learning only a prefix. Such prefix is automatically optimized to steer the language model towards the new objective during a standard training procedure. Following this approach, we suggest avoiding the fine-tuning to realize an even lighter model, where only the mapping network is trained. As presented in Section 4, our model not only produces realistic and meaningful captions, but also achieves superior results for some of the experiments without fine-tuning the language model. Note that fine-tuning CLIP does not benefit resulting quality, but does increase training time and complexity. We hence postulate that the CLIP space already encapsulates the required information, and adapting it towards specific styles does not contribute to flexibility.

3.3. Mapping Network Architecture

Our key component is the mapping network, which translates the CLIP embedding to the GPT-2 space. When the language model is simultaneously fine-tuned, the mapping is less challenging, as we easily control both networks. Therefore, in this case, we can employ a simple Multi-Layer Perceptron (MLP). We have achieved realistic and meaningful captions even when utilizing only a single hidden layer, as CLIP is pre-trained for a vision-language objective.

case 1

Nevertheless, when the language model is frozen, we propose utilizing the more expressive transformer [36] architecture. The transformer enables global attention between input tokens while reducing the number of parameters for long sequences. This allows us to improve our results by increasing prefix size, as shown in Section. 4. We feed the transformer network with two inputs, the visual encoding of CLIP and a learned constant input. The constant has a dual role, first, to retrieve meaningful information from CLIP embedding through the multi-head attention. Second, it learns to adjust the fixed language model to the new data. This is demonstrated in Section. 4, where we offer interpretability for our generated prefix. As can be seen, when the language model is fixed, the transformer mapping network learns a meticulous set of embeddings without any textual meaning. These are optimized to tame the language model.

case 2

importance of constants

fixed model can lead to embeddings without any meaning

3.4. Inference

During inference, we extract the visual prefix of an input image x using the CLIP encoder and the mapping network F. We start generating the caption conditioned on the visual prefix, and predict the next tokens one by one, guided by the language model output. For each token, the language model outputs probabilities for all vocabulary tokens, which are used to determine the next one by employing a greedy approach or beam search.

4. Results

Datasets. We use the *COCO-captions* [7, 22], *nocaps* [1], and *Conceptual Captions* [33] datasets. We split the former according to the Karpathy et al. [17] split, where the training set contains $120,000$ images and 5 captions per image. Since COCO is limited to 80 classes, the nocaps dataset is designed to measure generalization to unseen classes and concepts. It contains only validation and test sets, with the training utilizing COCO itself. The nocaps dataset is divided to three parts — *in-domain* contains images portraying only COCO classes, *near-domain* contains both COCO and novel classes, and *out-of-domain* consists of only novel classes. As suggested by Li et al. [19], we evaluate the model using only the validation set. Though some methods utilize object tags of the novel classes, we only consider the setting of no additional supervision, as we find it more applicable in practice. Therefore, we do not employ a constrained beam search [2]. The Conceptual Captions dataset consists of 3M pairs of images and captions, harvested from the web and post-processed. It is considered to be more challenging than COCO due to the larger variety of styles of both the images and the captions, while not limited to specific classes. To focus on the concepts, specific entities in this dataset are replaced with general notions. For example, in Fig. 1, the names are replaced with "politician". For evaluation, we use the validation set, consisting of 12.5K images, as the test set is not publicly available. Consequently, we did not use this set for validation.

this is a good split to guage the performance on differnet levels

tried to make it more generalised so it can more useful

Baselines. We compare our method to the state-of-the-art works of Li et al. [19] (known as Oscar), Vision-Language Pre-training model (VLP) [47], and the eminent work of Anderson et al. [4], denoted BUTD. These models first produce visual features using an object detection network [31]. BUTD then utilizes an LSTM to generate the captions, while VLP and Oscar employ a transformer, trained simi-

(A) **Conceptual Captions**

Model	ROUGE-L ↑	CIDEr ↑	SPICE ↑	#Params (M) ↓	Training Time ↓
VLP	24.35	77.57	16.59	115	1200h (V100)
Ours; MLP + GPT2 tuning	**26.71**	**87.26**	**18.5**	156	80h (GTX1080)
Ours; Transformer	25.12	71.82	16.07	**43**	**72h** (GTX1080)

massive drop in training time

(B) **nocaps**

Model	in-domain		near-domain		out-of-domain		Overall		Params↓	Time↓
	CIDEr↑	SPICE ↑	CIDEr	SPICE	CIDEr	SPICE	CIDEr	SPICE		
BUTD [4]	74.3	11.5	56.9	10.3	30.1	8.1	54.3	10.1	52	960h
Oscar [19]	79.6	**12.3**	66.1	**11.5**	45.3	**9.7**	63.8	**11.2**	135	74h
Ours; MLP + GPT2 tuning	79.73	12.2	**67.69**	11.26	**49.35**	**9.7**	65.7	11.1	156	7h
Ours; Transformer	**84.85**	12.14	66.82	10.92	49.14	9.57	**65.83**	10.86	**43**	**6h**

(C) **COCO**

Model	B@4 ↑	METEOR ↑	CIDEr ↑	SPICE ↑	#Params (M) ↓	Training Time ↓
BUTD [4]	36.2	27.0	113.5	20.3	52	960h (M40)
VLP [47]	36.5	28.4	117.7	21.3	115	48h (V100)
Oscar [19]	**36.58**	**30.4**	**124.12**	**23.17**	135	74h (V100)
Ours; Transformer	33.53	27.45	113.08	21.05	**43**	**6h** (GTX1080)
Ours; MLP + GPT2 tuning	32.15	27.1	108.35	20.12	156	7h (GTX1080)

(D) **Ablation**

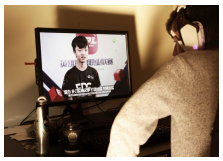
Ours; Transformer + GPT2 tuning	32.22	27.79	109.83	20.63	167	7h (GTX1080)
Ours; MLP	27.39	24.4	92.38	18.04	32	6h (GTX1080)

Table 1. Quantitative evaluation. As can be seen, our method achieves comparable results for both nocaps and Conceptual Captions with much faster training time.

Ground Truth	A man with a red helmet on a small moped on a dirt road	A young girl inhales with the intent of blowing out a candle.	A man on a bicycle riding next to a train.	a wooden cutting board topped with sliced up food.	A kitchen is shown with a variety of items on the counters.
Oscar	a man riding a motorcycle down a dirt road.	a woman sitting at a table with a plate of food.	a woman riding a bike down a street next to a train.	a woman sitting at a table with a plate of food.	a kitchen with a sink, dishwasher and a window.
Ours; MLP + GPT2 tuning	a man riding a motorcycle on a dirt road.	a woman is eating a piece of cake with a candle.	a man is standing next to a train.	a row of wooden cutting boards with wooden spoons.	a kitchen with a sink, stove, and window.
Ours; Transformer	a man is riding a motorbike on a dirt road.	a young girl sitting at a table with a cup of cake.	a man is standing next to a train.	a wooden table with a bunch of wood tools on it.	a kitchen with a sink and a window.

Figure 3. Uncurated results of the first five images in the COCO test set (Karpathy et al. [17] split).






Ground Truth	A life in photography – in pictures.	Photograph of the sign being repaired by brave person.	Globes : the green 3d person carrying in hands globe.	The player staring intently at a computer screen.	The - bedroom stone cottage can sleep people.
VLP	Actors in a scene from the movie.	The sign at the entrance.	Templates: green cartoon character holding the earth globe.	Person works on a video.	The master bedroom has a king - sized bed with a queen size bed.
Ours; MLP + GPT2 tuning	Actor sits in a hotel room.	The sign at the entrance.	3d render of a man holding a globe.	Person, a student, watches a video on his laptop.	The property is on the market for £ 1.
Ours; Transformer	person sitting on a chair in a room.	a sign is seen at the entrance to the store.	stock image of a man holding the earth.	portrait of a young boy playing video game.	one of the bedrooms in the house has been converted into a living room.

Figure 4. Uncurated results of the first five images in our test set for Conceptual Captions [33].


A person standing in front of a rock formation in the desert.


A man holding a banana in front of a river.


Two horned goats crossing a road in the desert.


A person sitting at a table with a tray of sushi.

Figure 5. Results over smartphone photos. Top: using our Conceptual Captions model. Bottom: COCO model. As demonstrated, our approach generalizes well to newly photographed images.

larly to BERT [11]. Both VLP and Oscar exploit an extensive pre-trained procedure over millions of image-text pairs. Oscar [19] also uses additional supervision compared to our setting, in the form of object tags for each image.

Our default configuration employs the transformer mapping network, without fine-tuning the language model, denoted **Ours; Transformer**. Additionally, we also evaluate our variant that utilizes the MLP mapping network, and fine-tunes the language model, denoted **Ours; MLP + GPT2 tuning**. Other configurations are evaluated in Tab. 1(D).

Evaluation metrics. Similar to Li et al. [19], we validate our results over the COCO dataset using the common metrics BLEU [27], METEOR [10], CIDEr [37] and SPICE [3], and for the nocaps dataset using CIDEr and SPICE. For the Conceptual Captions, we report the ROUGE-L [21], CIDEr, and SPICE, as suggested by the authors [33].

Furthermore, we measure the training time and the number of trainable parameters to validate the applicability of our method. Reducing the training time allows to quickly obtain a new model for new data, create an ensemble of models, and decrease energy consumption. Similar to other works, we report training time in GPU hours, and the GPU model used. The number of trainable parameters is a popular measure to indicate model feasibility.

Quantitative evaluation. Quantitative results for the challenging Conceptual Captions dataset are presented in Tab. 1(A). As can be seen, we surpass the results of VLP, while requiring orders of magnitude less training time. We note that our lightweight model, which does not fine-tune GPT-2, achieves an inferior result for this dataset. We hypothesize that due to the large variety of styles, a more expressive model is required than our light model, which induces a significantly lower parameter count. We compare only to VLP, as the other baselines haven't published results nor trained models for this dataset.

Tab. 1(B) presents results for the nocaps dataset, where we achieve comparable results to the state-of-the-art method Oscar. As can be seen, Oscar achieves a slightly better SPICE score and we attain a slightly better CIDEr score. Still, our method uses a fraction of training time and trainable parameters with no additional object tags required, hence it is much more useful in practice.

						
GPT-2 tuning	Caption	a motorcycle is on display in a showroom.	a group of people sitting around a table.	a living room filled with furniture and a book shelf filled with books.	a fire hydrant is in the middle of a street.	display case filled with lots of different types of donuts.
	Prefix	com showcase motorcycle A ray motorcycle-posed what polished Ink	blond vegetarian dishes dining expects smiling friendships group almost	tt sofa gest chair Bart books modern doorway bedroom	neon street Da alley putistan colorful nighttime	glass bakery dough displays sandwiches2 boxes Prin ten
w/o tuning	Caption	motorcycle that is on display at a show.	a a group of people sitting at a table together.	a living room with a couch and bookshelves.	a fire hydrant in front of a city street.	a display case full of different types of doughnuts.
	Prefix	oover eleph SniperÃÂÃÂ motorcycle synergy undeniably achieving\n	amic Delicious eleph SukActionCode photographers interchangeable undeniably achieving	orianclassic eleph CameroonÃÂÃÂÃÂroom synergy strikingly achieving\n	ockets Pier eleph SniperÃÂÃÂ bicycl synergy undeniably achieving\n	peanuts desserts elephbmÃÂÃÂÃÂ cooking nodd strikingly achieving\n

Figure 6. Prefix Interpretability. We present both the generated caption and our prefix interpretation. Upper: Ours; MLP + GPT2 tuning. Bottom: Ours; Transformer.

Tab. 1(C) present the results for the COCO dataset. Oscar reaches the best results, however, it uses additional input in the form of object tags. Our results are closed to VLP and BUTD which utilize considerably more parameters and training time. Note that the training time of VLP and Oscar does not include the pre-training step. For instance, pre-training of VLP requires training over Conceptual Captions which consumes 1200 GPU hours.

their model has more generalised performance

Both Conceptual Captions and nocaps are designed to model a larger variety of visual concepts than COCO. Therefore, we conclude our method is preferable for generalizing to diverse data using a quick training procedure. This originates from utilizing the already rich semantic representations of both CLIP and GPT-2.

Qualitative evaluation. Visual results of the uncurated first examples in our test sets of both Conceptual Captions and COCO datasets are presented in Figs. 3 and 4 respectively. As can be seen, our generated captions are meaningful and depict the image successfully for both datasets. We present additional examples collected from the web in Fig. 1. As can be seen, our Conceptual Captions model generalizes well to arbitrary unseen images as it was trained over a sizable and diverse set of images. We also present in Fig. 5 results over smartphone images, to further demonstrate generalization to new scenarios. Moreover, our model successfully identifies uncommon objects even when trained only over COCO. For example, our method recognizes the wooden spoons or the cake with a candle better than Oscar in Fig. 3, since CLIP is pre-trained over a diverse set of images. However, our method still fails in some cases, such as recognizing the bicycle next to the train in Fig. 3. This is inherited from the CLIP model, which does not perceive the bicycle in the first place. We conclude that our model would benefit from improving CLIP object detection ability, but leave this direction for future work. For Conceptual Captions, our method mostly produces accurate captions, such as perceiving the green 3d person in Fig. 4. As expected, our method still suffers from data bias. For instance, it depicts the bedroom image in Fig. 4 as "The property is on the market for £ 1" after witnessing such captions of property advertising during training.

Language model fine-tuning. As described in Section. 3, fine-tuning the language model results in a much more expressive model, but that is also more susceptible to overfitting, as the amount of trainable parameters increases. As can be seen in Tab. 1, the two variants — with and without the language model fine-tuning — are comparable. Over the extremely complicated Conceptual Captions dataset, we get superior results with the fine-tuning. While over the popular COCO dataset, avoiding the fine-tuning achieves better results. Regarding nocaps dataset, the results are roughly equal, thus the lighter model would be preferable. We thus hypothesize that extremely elaborated datasets or ones that present a unique style require more expressiveness, and hence the more likely it is to benefit from the fine-tuning.

Prefix Interpretability. To further understand our method and results, we suggest interpreting the generated prefixes as a sequence of words. Since the prefix and word embeddings share the same latent space, they can be treated similarly. We define the interpretation of each of the k prefix embeddings as the *closest* vocabulary token, under cosine similarity. Fig. 6 shows examples of images, the generated captions, and their prefix interpretations. The

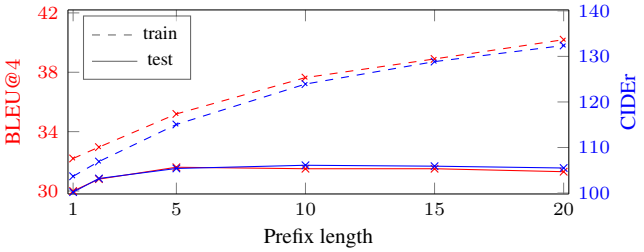
interpretation is meaningful when both the mapping network and GPT-2 are trained. In this case, the interpretation contains salient words that associate with the content of the image. For Instance, motorcycle and showcase in the first example. However, when we only train the mapping network, the interpretation becomes essentially unreadable since the network is also charged with maneuvering the fixed language model. Indeed, a considerable part of the prefix embeddings is shared across different images for the same model, as it performs the same adjustment to GPT-2.

Prefix length. Li and Liang [20] showed that increasing the size of the prefix length, up to a certain value, improves the performance of the model in an underlying task. Moreover, the saturation length might differ between tasks. For the image captioning task, we conduct an ablation over the prefix lengths using the COCO dataset over two configurations of our method: **Ours; Transformer** and **Ours; MLP + GPT2 tuning**. The results are summarized in Fig. 7. For each prefix size and configuration, we train the network for 5 epochs and report the BLEU@4 and CIDEr scores over the test and train sets.

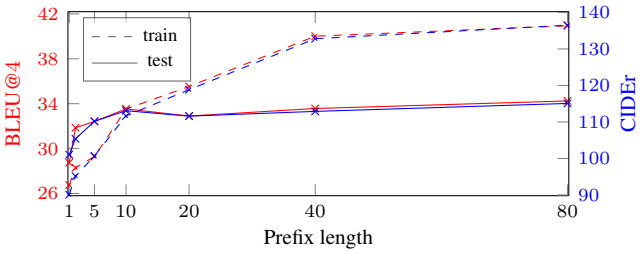
As can be seen in Fig. 7a, increasing the prefix size while allowing tuning of the language model results in overfitting to the training set, due to the large number of trainable parameters. However, when the language model is frozen, we experience improvement for both the training and test evaluations, as can be seen in Fig. 7b. Naturally, extremely small prefix length yields inferior results as the model is not expressive enough. In addition, we point out that the MLP architecture is inherently more limited as it is not scalable for a long prefix. For example, a prefix size of 40 implies a network with over 450M parameters, which is unfeasible for our single GPU setting. The transformer architecture allows increasing the prefix size with only marginal increment to the number of the parameters, but only up to 80 — due to the quadratic memory cost of the attention mechanism.

Mapping network. An ablation study for the mapping network architecture is shown in Tab. 1(C),(D). As can be seen, with language model fine-tuning, the MLP achieves better results. However, the transformer is superior when the language model is frozen. We conclude that when employing the fine-tuning of the language model, the expressive power of the transformer architecture is unnecessary.

Implementation details. We used the prefix length of $K = 10$ for the MLP mapping networks, where the MLP contains a single hidden layer. For the transformer mapping network, we set the CLIP embedding to $K = 10$ constants tokens and use 8 multi-head self-attention layers with 8 heads each. We train for 10 epochs using a batch size of 40. For optimization, we use AdamW [18] with weight decay fix as introduced by Loshchilov et al. [24], with a learning rate of $2e^{-5}$ and 5000 warm-up steps. For GPT-2 we employ the implementation of Wolf et al. [41].



(a) MLP mapping network with fine-tuning of the language model.



(b) Transformer mapping network with frozen language model.

Figure 7. Effect of the prefix length on the captioning performance over the COCO-captions dataset. For each prefix length, we report the BLEU@4 (red) and CIDERr (blue) scores over the test and train (dashed line) sets.

5. Conclusion

Overall, our CLIP-based image-captioning method is simple to use, doesn't require any additional annotations, and is faster to train. Even though we propose a simpler model, it demonstrates more merit as the dataset becomes richer and more diverse. We consider our approach as part of a new image captioning paradigm, concentrating on leveraging existing models, while only training a minimal mapping network. This approach essentially learns to adapt existing semantic understanding of the pre-trained models to the style of the target dataset, instead of learning new semantic entities. We believe the utilization of these powerful pre-trained models would gain traction in the near future. Therefore, the understanding of how to harness these components is of great interest. For future work, we plan to incorporate pre-trained models (e.g., CLIP), to other challenging tasks, such as visual question answering or image to 3D translation, through the utilization of mapping networks.

References

[1] Harsh Agrawal, Karan Desai, Yufei Wang, Xinlei Chen, Rishabh Jain, Mark Johnson, Dhruv Batra, Devi Parikh, Stefan Lee, and Peter Anderson. nocaps: novel object caption-

ing at scale. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8948–8957, 2019. 2, 4

[2] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Guided open vocabulary image captioning with constrained beam search. *arXiv preprint arXiv:1612.00576*, 2016. 4

[3] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Spice: Semantic propositional image caption evaluation. In *European conference on computer vision*, pages 382–398. Springer, 2016. 6

[4] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6077–6086, 2018. 1, 3, 4, 5

[5] David Bau, Alex Andonian, Audrey Cui, YeonHwan Park, Ali Jahanian, Aude Oliva, and Antonio Torralba. Paint by word. *arXiv preprint arXiv:2103.10951*, 2021. 2

[6] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, Wei Liu, and Tat-Seng Chua. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5659–5667, 2017. 3

[7] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015. 2, 3, 4

[8] Xinpeng Chen, Lin Ma, Wenhao Jiang, Jian Yao, and Wei Liu. Regularizing rnns for caption generation by reconstructing the past with the present. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7995–8003, 2018. 3

[9] Xinlei Chen and C Lawrence Zitnick. Learning a recurrent visual representation for image caption generation. *arXiv preprint arXiv:1411.5654*, 2014. 1, 3

[10] Michael Denkowski and Alon Lavie. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the ninth workshop on statistical machine translation*, pages 376–380, 2014. 6

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3, 6

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3

[13] Hao Fang, Saurabh Gupta, Forrest Iandola, Rupesh K Srivastava, Li Deng, Piotr Dollár, Jianfeng Gao, Xiaodong He, Margaret Mitchell, John C Platt, et al. From captions to visual concepts and back. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1473–1482, 2015. 1, 3

[14] Rinon Gal, Or Patashnik, Haggai Maron, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators. *arXiv preprint arXiv:2108.00946*, 2021. 2

[15] Junlong Gao, Shiqi Wang, Shanshe Wang, Siwei Ma, and Wen Gao. Self-critical n-step training for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6300–6308, 2019. 3

[16] Simao Herdade, Armin Kappeler, Kofi Boakye, and Joao Soares. Image captioning: Transforming objects into words. *arXiv preprint arXiv:1906.05963*, 2019. 3

[17] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137, 2015. 4, 5

[18] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015. 8

[19] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision*, pages 121–137. Springer, 2020. 1, 3, 4, 5, 6

[20] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021. 2, 4, 8

[21] Chin-Yew Lin and Franz Josef Och. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 605–612, 2004. 6

[22] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 2, 3, 4

[23] Wei Liu, Sihan Chen, Longteng Guo, Xinxin Zhu, and Jing Liu. Cptr: Full transformer network for image captioning. *arXiv preprint arXiv:2101.10804*, 2021. 3

[24] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 8

[25] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *arXiv preprint arXiv:1908.02265*, 2019. 3

[26] Yunpeng Luo, Jiayi Ji, Xiaoshuai Sun, Liujuan Cao, Yongjian Wu, Feiyue Huang, Chia-Wen Lin, and Rongrong Ji. Dual-level collaborative transformer for image captioning. *arXiv preprint arXiv:2101.06462*, 2021. 3

[27] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002. 6

[28] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2085–2094, 2021. 2

[29] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021. 2, 3

[30] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 2, 3

[31] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28:91–99, 2015. 3, 4

[32] Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7008–7024, 2017. 3

[33] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018. 2, 4, 6

[34] Matteo Stefanini, Marcella Cornia, Lorenzo Baraldi, Silvia Cascianelli, Giuseppe Fiameni, and Rita Cucchiara. From show to tell: A survey on image captioning. *arXiv preprint arXiv:2107.06912*, 2021. 1, 3

[35] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019. 1, 3

[36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 3, 4

[37] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575, 2015. 6

[38] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164, 2015. 3

[39] Yufei Wang, Zhe Lin, Xiaohui Shen, Scott Cohen, and Garrison W Cottrell. Skeleton key: Image captioning by skeleton-attribute decomposition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7272–7281, 2017. 3

[40] Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. Simvlm: Simple visual language model pretraining with weak supervision. *arXiv preprint arXiv:2108.10904*, 2021. 3

[41] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, Oct. 2020. Association for Computational Linguistics. 8

[42] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057. PMLR, 2015. 1, 3

[43] Xu Yang, Hanwang Zhang, and Jianfei Cai. Learning to collocate neural modules for image captioning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4250–4260, 2019. 3

[44] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. Exploring visual relationship for image captioning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 684–699, 2018. 1

[45] Li Zhang, Flood Sung, Feng Liu, Tao Xiang, Shaogang Gong, Yongxin Yang, and Timothy M Hospedales. Actor-critic sequence training for image captioning. *arXiv preprint arXiv:1706.09601*, 2017. 3

[46] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5579–5588, 2021. 3

[47] Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason Corso, and Jianfeng Gao. Unified vision-language pre-training for image captioning and vqa. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13041–13049, 2020. 1, 3, 4, 5