Noise-aware Learning from Web-crawled Image-Text Data for Image Captioning

Wooyoung Kang* Jonghwan Mun* Sungjun Lee* Byungseok Roh
Kakao Brain
{edwin.kang, jason.mun, jun.untitled, peter.roh}@kakaobrain.com

Abstract

*Image captioning is one of the straightforward tasks that can take advantage of large-scale web-crawled data which provides rich knowledge about the visual world for a captioning model. However, since web-crawled data contains image-text pairs that are aligned at different levels, the inherent noises (e.g., misaligned pairs) make it difficult to learn a precise captioning model. While the filtering strategy can effectively remove noisy data, it leads to a decrease in learnable knowledge and sometimes brings about a new problem of data deficiency. To take the best of both worlds, we propose a **No**ise-aware **C**aptioning (NoC) framework, which learns rich knowledge from the whole web-crawled data while being less affected by the noises. This is achieved by the proposed alignment-level-controllable captioner, which is learned using alignment levels of the image-text pairs as a control signal during training. The alignment-level-conditioned training allows the model to generate high-quality captions by simply setting the control signal to the desired alignment level at inference time. An in-depth analysis shows the effectiveness of our framework in handling noise. With two tasks of zero-shot captioning and text-to-image retrieval using generated captions (i.e., self-retrieval), we also demonstrate our model can produce high-quality captions in terms of descriptiveness and distinctiveness. The code is available at https://github.com/kakaobrain/noc.*

1. Introduction

The recent introduction of large-scale data of image-text pairs [5, 42, 20] has brought remarkable advances in computer vision, *e.g.*, CLIP [38] for multi-modal representation learning and DALL·E [40] for the text-to-image generation task. This is mainly thanks to the scalability of the data collection process as well as the rich knowledge described in alt-texts of web-crawled data. Inspired by this, research on image captioning is also moving towards exploiting large-scale web-crawled image-text paired data [49, 47, 23, 54].

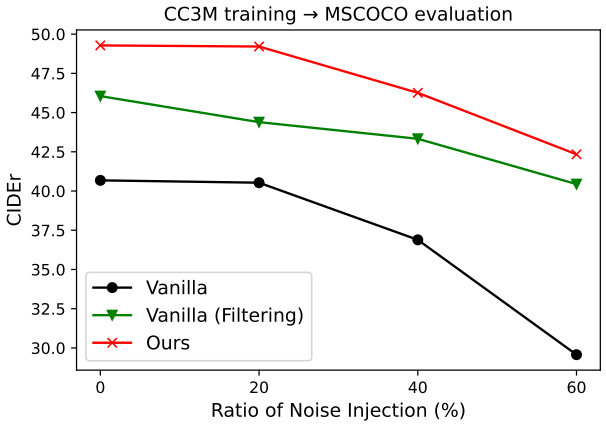

Figure 1: Zero-shot captioning performance curve on MSCOCO when varying the ratio of noise injection to data of CC3M. To deliberately make noise data, we replace captions up to the specified ratio with ones from randomly selected images in the dataset. Models learned without consideration of noises suffer from performance degradation, even with a data filtering scheme.[1] In contrast, our model is more robust to noises and provides more accurate captions, indicating the necessity for noise-aware learning.

While web-crawled data is effective in learning rich knowledge about the visual world, it inherently suffers from *noise issues* as some text may be unrelated to its paired image. According to our observation from Fig. 1, the quality of captions generated by a standard captioning model, when learned without consideration of noises, dramatically deteriorates as more noisy data is included during training.

One straightforward approach to tackle noises in large-scale web-crawled data is the *CLIP-based filtering strategy* [42] where image-text pairs are filtered out according to their CLIP similarity[2]. As shown in Fig. 1, the filtering strategy improves the quality of captions by leaving only relatively well-aligned image-text pairs for training. However, in general, filtering methods without oracle criteria in-

*These authors contributed equally.

[1]For a fair comparison with the filtering scheme reducing the number of training samples, we train all models for the same steps rather than epochs.

[2]Throughout the paper, the term CLIP similarity is used to denote image-text cosine similarity calculated by the CLIP model, indicating the quality of the caption for a paired image as described in [18].

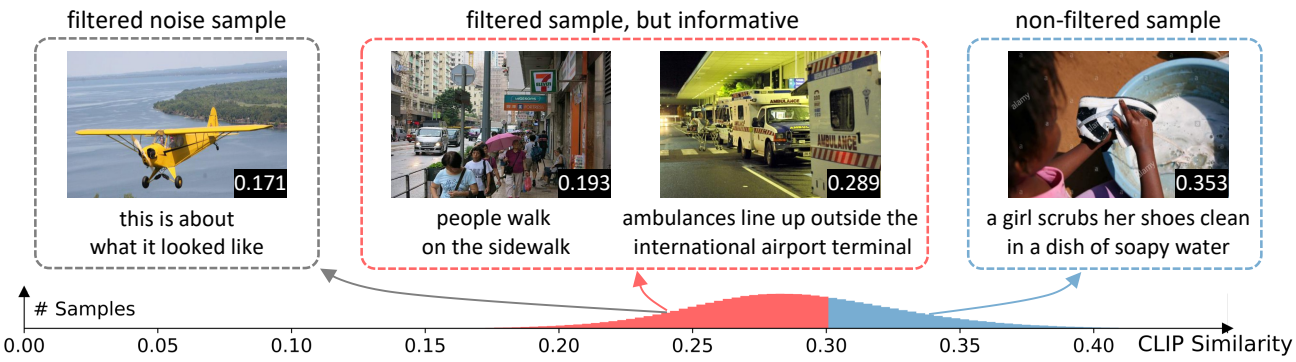

Figure 2: Examples of web-crawled image-text pairs in CC3M, where numbers within each image indicate CLIP similarity. Filtering with a threshold of 0.3, a selected threshold value on [42] after human evaluations, effectively leaves well-aligned samples (right) and removes noise samples (left). However, according to our observation, it often discards informative ones (middle) as well.

evitably discard data informative for training models with CLIP similarity below a certain threshold. Fig. 2 illustrates such unintentionally filtered cases. In addition, the inability to access filtered data reduces learnable knowledge, limiting the power of expression during caption generation. Thus, the filtering strategy may not be optimal for handling noises.

Considering the observation from Fig. 2, we argue the necessity for the noise-robust image captioning model to fully exploit web-crawled image-text data without filtering; note that, despite its importance, it is unexplored yet so far as we know. More specifically, we set our main goal to design a noise-robust model so that the model 1) can generate highly-aligned captions like non-filtered samples in Fig. 2 and 2) also takes advantage of informative knowledge in the data that would be removed with filtering method.

We introduce **No**ise-aware **C**aptioning (NoC) framework based on an alignment-level-controllable captioner. In the framework, we first assign alignment levels to web-crawled data by discretizing CLIP similarities of image-text pairs. Then, the model is trained using the alignment level as an additional control signal, enabling the model to generate captions with the desired alignment level. At inference time, high-quality captions can be generated by feeding a control signal indicating the top level of alignment.

We conduct comprehensive experiments to validate the effectiveness of our model. First, from the experiments on zero-shot image captioning and self-retrieval tasks, our model outperforms comparative methods, indicating the superior quality of the generated captions in both *descriptiveness* and *distinctiveness*. Second, we observe that NoC framework enhances the pre-training→fine-tuning scheme thanks to the more advanced level of visual-language understanding achieved by noise-aware pre-training. Third, we show that NoC framework provides consistent performance gain compared to the filtering strategy when scaling up dataset sizes up to 125M. Finally, we further analyze how the noise issue is addressed in the proposed method by investigating the memorization effect of noisy pairs.

Our main contributions are summarized as follows:

- We propose a novel Noise-aware Captioning (NoC) framework for handling the noise issue, which is underexplored despite its potential importance.
- We propose an alignment-level-controllable captioner that utilizes alignment levels of data as a control signal, thus effectively addressing noise issues and being able to generate highly correlated captions by adjusting the control signal at inference time.
- We show the effectiveness of the proposed noise-aware learning through extensive experiments, where our model outperforms competing methods on both image captioning and self-retrieval tasks with large margins.

2. Related Work

Image captioning. Various captioning algorithms have been proposed within the encoder-decoder framework [9, 46, 13, 32]. In addition, attention mechanism [50, 7, 29, 33] or transformer architecture [36, 11, 17, 16, 30] is incorporated to further boost the performance. Those models are generally trained on top of human-annotated caption data such as MSCOCO [27] and Visual Genome [22]. However, learning captioning models from such data has two limitations. First, scaling up dataset size is extremely difficult due to expensive human annotations. Second, the limited learnable knowledge due to small scales results in a poor generalization to the visual concepts in the wild [1, 45].

Web-crawled datasets [43, 6, 20, 19, 42] have got attention recently because alt-texts of the data describe paired images with diverse visual concepts, and it is much easy to scale up. Indeed, some research [26, 58, 19, 49, 47, 23, 54] on image captioning start to exploit the large-scale web-crawled image-text pairs with various vision-language pre-training objectives and show remarkable performances. However, since the web-crawled data depends on alt-texts, it inevitably includes noisy pairs in a high ratio. Although such noisy data may hamper the learning of normal samples, most large-scale research has not yet studied how to

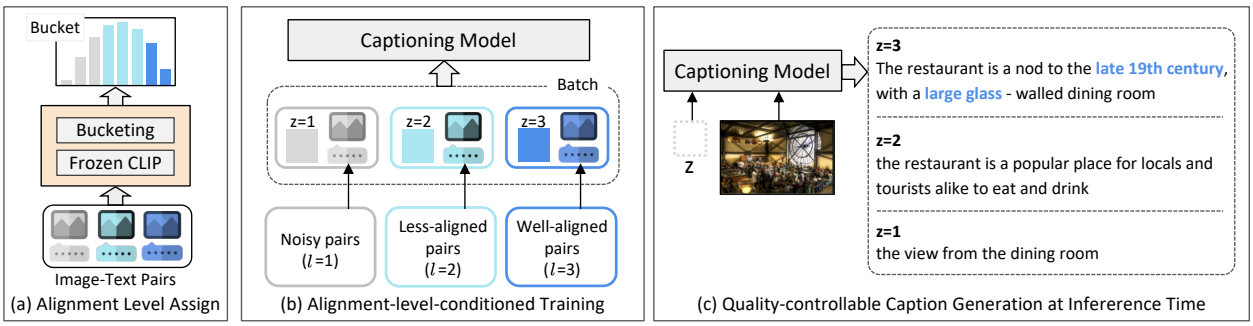

Figure 3: The proposed noise-aware learning framework from web-crawled image-text data. (a) We first identify alignment levels l of individual web-crawled data by discretizing the image-text similarities computed by CLIP. (b) Then, we use the alignment levels as the control signal z to train a captioning model (*i.e.*, $z = l$ during training); through the alignment-level-conditioned training, the model is encouraged to generate well-aligned captions with $z = 3$ while being guided to generate noisy captions with $z = 1$. (c) Finally, at inference time, we can generate highly aligned captions by simply feeding a control signal corresponding to the top level of alignment ($z = 3$).

effectively handle noisy data; existing large-scale learning approaches [49, 54, 19, 47] simply train a captioning model without any consideration for noisy pairs. In contrast, we address the noise issue to maximally exploit all available web-crawled data through noise-aware learning.

Learning from noisy labels. Under the assumption of possible noisy annotations in training data, numerous approaches have been proposed to alleviate the negative effects of the noisy data. Typically, existing works often resort to noise-robust architecture [14, 52, 10], specialized design of loss function [59, 41, 31] or sample selection [34, 15, 21]. However, most methods consider the unimodal classification task, so it is challenging to extend them to multimodal tasks (*e.g.*, image captioning). For example, let us consider one of the most effective approaches, the noise transition matrix [8, 14]. The noise transition matrix is added on the top of the softmax layer and estimates noisy class posterior probability by discovering the underlying label transition pattern. However, defining a transition matrix for image captioning is impractical due to the numerous words and long range of contexts. Moreover, other noise-handling algorithms [15, 21] that employ additional co-trained networks are inefficient and unsuitable for our large-scale training scenario due to expensive computational costs.

Closely related to our motivation, BLIP [24] tackled noisy image-text pairs in web-crawled data with a data bootstrapping technique. However, in addition to web-crawled data, BLIP still relies on supplementary clean human annotations such as MSCOCO to train the captioning and filtering models for bootstrapping. In contrast, without any clean annotations, we tackle noise issues with the proposed noise-aware learning framework given only web-crawled data.

3. Noise-aware Learning for Image Captioning

3.1. Overview

Given a pair of an image I and a caption c consisting of T words $(w_1, w_2, ..., w_T)$, the image captioning models are typically trained by minimizing a negative log-likelihood:

$$\mathcal{L} = -\log p(c|I) = \sum_{t=0}^{T} -\log p(w_{t+1}|w_{\leq t}, I), \quad (1)$$

where two additional words in Eq. (1)—w_0 (<BOS>) and w_{T+1} (<EOS>)—are used to indicate the begin and end of a sentence, respectively. In general, the models are designed with the assumption that the training data is clean enough so all image-text pairs are well-aligned.

However, when using the web-crawled data for training, as presented in Fig. 1, the noisy data hinders the learning of the vanilla captioning models. A filtering strategy can effectively remove noisy data, but it is also highly likely to discard informative data as depicted in Fig. 2. Therefore, to fully benefit from the rich information of web-crawled data, we aim to make a captioning model robust to noise while using the whole dataset.

As illustrated in Fig. 3, we propose our NoC framework with an alignment-level-controllable captioner. During training, we identify an alignment level l of the given image-text pair and use it as a control signal z to make a captioning model be conditioned on the alignment level, which leads to a new objective as follows:

$$\mathcal{L} = -\log p(c|I, z) = \sum_{t=0}^{T} -\log p(w_{t+1}|w_{\leq t}, I, z). \quad (2)$$

This objective encourages the model to learn the capability of generating captions of different alignment levels depending on a control signal. Then, at the inference time, we can steer the captioning model to generate accurate captions by simply feeding a control signal, meaning a top-level of alignment. With this model design, as discussed in our experiments, we can take the following two advantages: (1) it can learn well-aligned data with less distraction by noisy data and (2) it can still take advantage of data containing useful knowledge (*e.g.*, more diverse visual concepts) that might be discarded when filtering is applied.

In the rest of this section, we first explain how we define alignment levels and assign them to image-text pairs. Then, we describe the architecture of our alignment-level-controllable captioner. Finally, we discuss why our method is more effective compared to the filtering strategy.

3.2. Alignment Level Assignment

Our key component is how we can assign an alignment level l to a given image-text pair. For this purpose, we first define the alignment level of a given image-text pair; in a nutshell, as an image-text pair is more correlated, we consider the pair is more aligned. Since there is no ground truth for the correlation, we leverage a pre-trained model (*i.e.*, CLIP) optimized by image-text contrastive learning from web-crawled data; contrastive learning typically drives a model to learn the correlation between images and texts in a data-driven manner, so the similarity scores by CLIP can be used as alignment scores for image-text pairs.

Given the CLIP similarity scores, $s \in \mathbb{R}$, from all training samples, we convert them into K discrete alignment levels $l \in \{1, ..., K\}$ via a bucketing technique:

$$l = f_{\text{bucket}}(s), \tag{3}$$

where f_{bucket} is a bucketing function with K bins of equally spaced boundaries. This bucketing makes more noisy data of low CLIP similarity assigned to a bucket of the lower alignment level (*e.g.*, $l = 1$) and well-aligned data of high CLIP similarity allocated to a bucket of the higher alignment level (*e.g.*, $l = K$).

3.3. Alignment-level-controllable Captioner

Alignment-level controllability in caption generation. Given the image-text data of different alignment levels, our goal is to make a model that generates semantically well-aligned captions (like non-filtered data in Fig. 2) while benefiting from the whole data. However, the vanilla captioning model, which processes all data in the same way, is inevitably infected with the noisy samples and is learned to generate captions of limited quality. Therefore, we introduce a controllable captioning model. By using alignment levels of image-text data as the control signal (*i.e.*, $z = l$ during training), the quality of captions to be generated can be controllable by adjusting the control signal z. That is, our model is learned to generate well-aligned captions with a control signal of $z = K$ (high alignment level) while generating noisy captions with a control signal of $z = 1$ (low alignment level). At inference time, in practice, we feed a control signal corresponding to a higher alignment level to generate highly well-aligned captions from input images.

Architecture. Our method is applicable to any captioning model with a simple modification making the decoder take an additional input of the control signal. We employ a VirTex-like [12] transformer-based encoder-decoder model due to its simplicity. Given a pair of image and text, (I, c), the encoder first extracts a visual feature from the input image. Next, we feed a control signal z—which is an alignment level (*i.e.*, $z = l$) during training calculated as described in Sec. 3.2 and is set to a constant one at inference time—into a learnable control embedding layer and then concatenate the resulting control embedding with the image embedding. Finally, the concatenated vectors are fed into each cross-attention layer of the decoder as key-value, and captions are generated in an auto-regressive manner. A more detailed explanation is given in Appendix B.

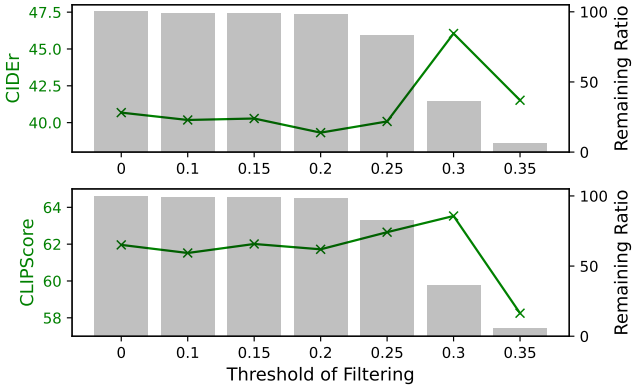

Figure 4: Captioning scores and the ratio of remaining data when varying the threshold of CLIP similarity score for the filtering. Each model is trained on CC3M and evaluated on MSCOCO.

3.4. Discussion

Recent work [48] shows that conventional captioning models are trained to generate the so-called *average* captions that consist of common words and phrases in the training corpus. In other words, with web-crawled data where image and text are aligned at different levels, the captioning models may be trained to generate captions of a common level of alignment (*i.e.*, majority of data), not the highest level. With this observation, the filtering strategy can be interpreted as improving the quality of captions by raising the alignment level of common captions. Accordingly, better performance can be achieved by simply using a higher threshold for filtering. However, filtering with a higher threshold sometimes leads to a data deficiency problem as it significantly decreases dataset size. According to our experiment in Fig. 4, the best performance is achieved at a threshold of 0.3, not 0.35 where almost data is discarded. This implies that the filtering is affected by a trade-off between the quality and the scale of non-filtered data.

In contrast, our NoC framework can address the noise issue in a more principled way. Controllability allows our model to be thought of as an implicit mixture of experts (but sharing parameters); one expert (*e.g.*, our model with

$z = K$) is specialized to a bucket of highly aligned image-text pairs (like filtering), while parameter sharing between experts allows sharing of learned knowledge for rich visual concepts across all experts (*e.g.*, our model with $z \leq K$).

4. Experiment

4.1. Evaluation Setting

Recall that our goal is to learn a captioning model from the web-crawled data, including inherent noises; therefore, we evaluate the quality of models without fine-tuning on clean data to check how effectively to tackle noisy data. To assess the quality of models, we consider two aspects of generated captions: *descriptiveness* (*i.e.*, how well aligned with given images) and *distinctiveness* (*i.e.*, how well describe images with their unique aspects). To be specific, we conduct zero-shot captioning and self-retrieval tasks (*i.e.*, text-to-image retrieval using generated captions) for *descriptiveness* and *distinctiveness* comparison, respectively. Note that the term zero-shot means the model is not further fine-tuned on clean datasets for evaluation.

Metrics. For the zero-shot captioning task, we measure the standard captioning evaluation metrics, *i.e.*, BLEU@4, METEOR, CIDEr, and SPCIE, which compares the generated captions with ground-truth captions. In addition, we use the CLIPScore [18] to measure whether the generated captions are semantically matched with given images using CLIP. For the self-retrieval task, we compute the recall (*i.e.*, R@K with $K = \{1, 5, 10\}$) of the paired images from the generated captions using an external text-to-image retrieval model. Note that we use the pre-trained CLIP ViT-L/14 for CLIPScore calculation and the retrieval task.

Datasets. When training models, we use Conceptual Captions 3M (CC3M) [43] dataset. For the evaluation of zero-shot captioning performance, we exploit MSCOCO [27] and nocaps [1] validation set. For the self-retrieval task, we leverage MSCOCO and Flickr30k [37] test set. Note that, for MSCOCO, we use its Karpathy test split for evaluation.

4.2. Baselines

Since research on noise-robust image captioning has rarely been explored, we evaluate our method against three carefully designed baselines in a controlled setting to measure their performance without any confounding factors. All baseline models and ours have the same backbone architecture for fair comparisons.

Vanilla. The first baseline is a vanilla encoder-decoder model and is trained using whole image-text data without any noise handling technique.

Vanilla (Filtering). The second baseline is a vanilla captioning model but trained with a filtering strategy to tackle the noise issue. Following the previous convention [42], we calculate cosine similarity for each pair of (image, text) using a pre-trained CLIP ViT-B/32 model [38], then leave pairs having similarity larger than 0.3 as training data. The threshold of 0.3 is selected the following [42], which provides the best performance among thresholds from 0.1 to 0.35 with steps of 0.05 as presented in Fig. 4.

Loss weighting. The final baseline is a vanilla captioning model trained with a loss re-weighting strategy to tackle the noise issue. In this baseline, instead of filtering, we use CLIP similarity score to re-weight sample-wise loss as

$$\mathcal{L}_{\text{weighting}} = -\frac{1}{N}\sum_{i=1}^{N} s_i \log p(c_i|I_i), \tag{4}$$

where s_i indicates a cosine similarity computed by CLIP for i^{th} image-text pair in a minibatch of N instances. Further details for this baseline are explained in Appendix D.

4.3. Implementation details

We employ a pre-trained CLIP ViT-L/14 for encoding visual features and computing the alignment level z, and use 6 randomly-initialized transformer blocks as the caption decoder. Except for results in Tabs. 1 to 3, we freeze the visual encoder and take a single CLS token as the visual feature due to its efficiency in all ablation and analytical experiments. More details regarding training settings are described in Appendix A. When collecting alignment levels of training samples, we use a bucket of K (=8) bins. For the data augmentation, we perform the same augmentation strategy of CLIP [38], *i.e.*, resizing the shorter side of an original image to 256, then applying a random square crop of 224x224 scale. We use AdamW optimizer [28] with a linear warm-up strategy for 10% of whole training iterations followed by learning rate decaying with a cosine schedule. We set a base learning rate as 0.0016 with a total batch size of 2048. During the training phase, we train all baselines and our model for the same iteration steps that correspond to 10 epochs when using the whole data.

4.4. Zero-shot Captioning Task

4.4.1 Comparison with baselines

We compare the zero-shot captioning performances of baselines and our method on MSCOCO and nocaps datasets. In Tab. 1, it has been observed that the Vanilla model exhibits the lowest performance on both the MSCOCO and nocaps datasets, compared to other baselines. This suggests that the absence of any noise-handling technique can impede the training of a model due to the presence of noisy samples. Also, the filtering strategy provides considerable performance gain compared to the Vanilla and the Loss weighting,

Models	MSCOCO				nocaps							
					overall		in-domain		near-domain		out-of-domain	
	B@4	M	C	CS	C	CS	C	CS	C	CS	C	CS
Vanilla	10.31	15.48	47.56	62.89	41.58	60.49	38.60	58.64	39.24	59.91	51.22	62.15
Vanilla (Filtering)	12.81	17.30	54.66	64.84	48.96	62.70	46.06	60.74	46.33	62.35	59.50	63.92
Loss weighting	11.16	16.15	50.86	63.87	43.89	61.18	39.30	59.23	41.80	60.50	53.84	63.04
NoC (z=7)	**15.96**	**19.50**	**62.04**	**66.70**	**54.94**	**64.21**	**51.74**	**62.54**	**53.09**	**63.92**	**63.15**	**65.19**

Table 1: Caption generation performance comparison with baselines on MSCOCO and nocaps datasets where all models are trained with CC3M without fine-tuning on the target dataset. B@4, M, C, and CS mean BLEU@4, METEOR, CIDEr, and CLIPScore metrics, respectively. Numbers in **bold** indicates the best method.

Method	Visual Encoder	Text Decoder	Data	MSCOCO			
				BLEU@4	METEOR	CIDEr	SPICE
Inference time optimization or un-paired training							
ZeroCap [44]	CLIP ViT-B/32	GPT2 (345M) [39]	-	2.60	11.50	14.60	5.50
Socratic Models [56]	CLIP ViT-L/14	GPT3 (175B) [4]	-	10.00	16.20	50.10	10.80
DeCAP [25]	CLIP ViT-B/32	Transformer$_{4\text{-}layer}$ (76.5M)	CC3M-text	8.80	16.00	42.10	10.90
Supervised training with image-text paired data							
Re-ViLM [51]	CLIP ViT-L/14	RETRO (410M) [3]	CCS + COYO [5]	**17.90**	-	53.60	-
SimVLM$_{1.4B}$ [49]	-	-	ALIGN 1.8B [20]	11.20	14.70	32.20	8.50
NoC (z=7)†	CLIP ViT-B/32	Transformer$_{4\text{-}layer}$ (76.5M)	CC3M	14.10	18.12	48.66	12.57
NoC (z=7)†	CLIP ViT-L/14	Transformer$_{6\text{-}layer}$ (94.5M)	CC3M	15.96	**19.50**	**62.04**	**14.37**

Table 2: Comparison with other models reporting the zero-shot captioning scores. The CCS is a combination of CC3M, CC12M [6], and SBU [35] datasets. † indicates a method that explicitly handles the problem of noisy samples. Numbers in **bold** indicate the best method.

which indicates learning a model using well-aligned captions is important for *descriptive* caption generation. On the other hand, our model significantly outperforms all baselines on both MSCOCO and nocaps datasets. Especially, the notable gain on the nocaps, which covers more diverse visual concepts, implies the importance of noise-aware learning across the entire dataset for acquiring a broader range of visual knowledge compared to learning from filtered data.

4.4.2 Comparison with other concurrent works

While the primary goal of our experiments is to measure the noise-robustness, we provide additional comparisons with other works that report zero-shot performance on the MSCOCO to better contextualize the effectiveness of our work in Tab. 2. The reported works are divided into two groups: 1) captioning with inference time optimization [44, 56] or leveraging an unpaired training strategy [25], and 2) directly training entire networks [49] or only intermediate modulation network [51] with image-text pairs, but not MSCOCO. Due to differences in architecture, training strategy, and dataset, we also provide comprehensive information about the settings for each method. As shown in Tab. 2, except for the BLEU@4, our algorithm outperforms others in all metrics with a substantial margin, despite having significantly fewer parameters. Specifically, Re-ViLM [51] shows a better BLEU@4 score than our model. We conjecture that Re-VILM would benefit from

Models	MSCOCO			Flickr30k		
	R@1	R@5	R@10	R@1	R@5	R@10
GT Caption	34.57	59.30	69.91	63.08	86.50	92.00
Vanilla	25.44	50.38	61.66	47.10	76.60	85.90
Vanilla (Filtering)	31.64	58.90	70.36	56.50	85.50	92.50
Loss weighting	28.78	54.44	65.44	48.00	78.90	87.50
NoC (z=7)	**40.00**	**66.78**	**77.53**	**65.10**	**92.00**	**96.20**

Table 3: Comparison of self-retrieval capability on MSCOCO and Flickr30k datasets. Numbers in **bold** indicate the best method.

the n-grams in retrieved captions by their retrieval augmentation technique at inference time. While the much higher CIDEr score of our model compared to Re-ViLM indicates NoC generates captions with more diverse expressions considering the TF-IDF weighting of the CIDEr.

4.5. Self-retrieval Task

We compare self-retrieval capability to measure how well each algorithm *distinctively* describes given images. Tab. 3 presents a comparison of self-retrieval performances between baselines and our method on MSCOCO and Flickr30k datasets. From the Tab. 3, our method outperforms all three baselines with large margins. One interesting observation is that generated captions by our method show a higher performance compared to ground-truth captions. We conjecture that the ground-truth captions are semantically accurate but may lack distinctiveness because human annotators would not be explicitly instructed to describe images

Table 4: Performances on MSCOCO and nocaps after fine-tuning on MSCOCO.

Method	MSCOCO test split		nocaps (overall)	
	CIDEr	SPICE	CIDEr	SPICE
Vanilla (Filtering)	126.14	22.37	87.03	12.52
NoC	**129.09**	**23.23**	**93.33**	**13.40**

Figure 5: Distribution of CLIP scores on MSCOCO dataset and examples at different alignment scores. Best viewed in zoom-in.

in a way that they are distinguishable from others. In contrast, our model can generate highly aligned captions with fine-grained details for the given images by adjusting the control signal with a high alignment level. This result implies our model is effective in generating *distinct* captions. The qualitative examples of the retrieval results are illustrated in Fig. 9 and Appendix H.

4.6. In-depth Analysis

4.6.1 Is it effective after fine-tuning?

We present results after fine-tuning CC3M pre-trained models on MSCOCO in Tab. 4. It turns out that our NoC framework can enhance the pre-training→fine-tuning scheme, which is the most common training pipeline for image captioning. We conjecture that there are two reasons for the effectiveness of our method compared to the filtering baseline as a pre-training method: 1) strong noise robustness in our method enables pre-trained models to acquire a more advanced level of visual-language understanding, 2) as even human-annotated data (*e.g.*, MSCOCO) is aligned at different levels in Fig. 5, our NoC framework can be favorably and effectively adapted to fine-tuning with the human-annotated data and leads to performance gains. It is noticeable that this experimental evidence emphasizes the practical usefulness and potential of our method for delivering improved results even in scenarios where human-annotated data is included.

4.6.2 Is it effective when scaling-up data size?

To validate whether our model is effective in larger-scale web-crawled data, we leverage COYO [5] dataset, which consists of 700M web-crawled image-text data without a CLIP-based filtering scheme. From COYO, we create four datasets of 3M, 10M, 23M, and 125M scales where the smaller dataset is a randomly sampled subset of the larger ones. For 10M, 23M, and 125M scales, we use a larger base learning rate (*i.e.*, 0.0032) and a mini-batch size (*i.e.*, 8192), respectively, to train models. Using four datasets, we train two baselines—Vanilla and Vanilla (Filtering)—and our method; due to its limited effectiveness compared to Vanilla (Filtering), we do not compare Loss weighting.

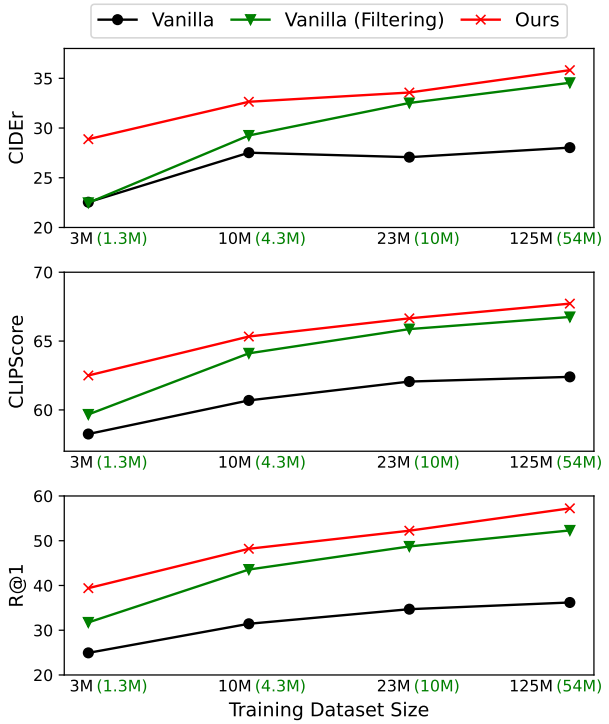

Figure 6: Zero-shot caption generation (CIDEr and CLIPScore) and self-retrieval (R@1) performance on MSCOCO when scaling up the training dataset sizes using COYO. Note that the green-colored numbers in parentheses mean dataset size after filtering. Models of each dataset are trained for the same number of steps.

Fig. 6 summarizes the results where we observe the followings. First, due to noisy data, the performance of Vanilla model is quickly saturated compared to others, thus showing a larger gap in the 125M dataset compared to the 3M one. Second, while Vanilla (Filtering) is considerably enhanced by using larger datasets, the performance gap is consistently kept across larger-scale datasets (23M → 125M). These observations indicate our method is more effective than two baselines in large-scale data as well.

4.6.3 How can our model handle noises?

We examine why our algorithm is robust to noisy data by inspecting the captioning results in different alignment levels. One of the reasons for the degradation of generalization performance when using noisy datasets is the powerful memorization ability of modern DNNs [57, 2]; this results in the over-memorization of noisy data in the training set, hindering the learning of normal data.

We analyze such memorization capability of Vanilla and our models for image-text data of different alignment levels.

Method	Noisy Group				Well-aligned Group			
	EM	B@4	CIDEr	CS	EM	B@4	CIDEr	CS
Vanilla	5350	12.63	122.24	52.49	8092	43.46	407.10	82.67
NoC (z=3)	10527	20.41	210.83	42.87	225	7.43	53.33	56.35
NoC (z=5)	939	5.83	48.47	55.86	4882	33.32	304.18	77.69
NoC (z=7)	47	2.10	18.28	59.57	10562	52.78	504.49	86.57

Table 5: Comparison of memorization capability on CC3M training samples of two different noise levels. Note that higher EM (# Exact Matching), B@4 (BLEU@4), and CIDEr scores in the noisy group mean over-memorization to noisy data. Higher CS (CLIPScore) means better alignment with given images.

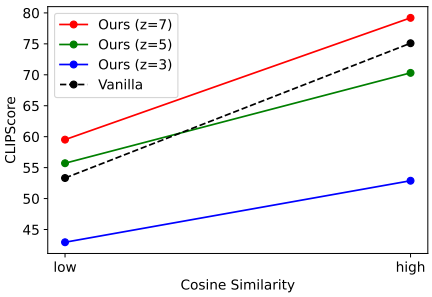

Figure 7: Performances on two groups of different similarities from CC3M validation set.

Options	CIDEr	CLIPScore
Quantile	49.22	65.72
Uniform	49.18	66.65

(a) Bucketing strategy

Options	CIDEr	CLIPScore
Sum	48.99	65.19
MLP	48.70	66.94
Concat.	49.18	66.65

(b) Control fusion method

Options	CIDEr	CLIPScore
4	49.28	65.67
8	49.18	66.65
16	48.58	65.67

(c) The number of bucket bins

Table 6: Ablations on the zero-shot MSCOCO captioning after training on the CC3M dataset. The options, highlighted in bold, are selected as our default model due to their balanced performance considering both CIDEr and CLIPScore.

For this experiment, we train models for longer steps so that the models fully fit and memorize the training data (CC3M). Then, we compare the memorization capability for *training samples split* into two groups with different alignment levels: 1) a noisy group of CLIP similarity between 0 and 0.15, and 2) a well-aligned group of CLIP similarity higher than 0.35. To measure the degree of memorization, we count the number of exact matching (EM) samples, where the generated caption is identical to the paired web-crawled caption of the input image, in addition to captioning metrics.

From Tab. 5, our model with $z = 7$ seems to be trained mainly from the data of high similarity; thus showing a larger number of exact matching examples in the group of high similarity, but an extremely small number of exact matching cases in the group of low similarity. In contrast, our model with $z = 3$ shows the opposite behavior. Based on this observation, we analyze that our controllable model is implicitly specialized in the different groups according to a control signal (z). As a result, our model with $z = 7$ can be less affected by noisy data that is dealt with $z \leq 3$. On the other hand, the Vanilla baseline suffers from over-memorization to noisy data, thus hindering the learning for pairs of high similarity. Consequently, by setting z to 7 at inference time, our model outperforms the Vanilla baseline on the CC3M validation set as presented in Fig. 7.

4.6.4 Ablation study

We also analyze the impact of two options for model design choice—a binning scheme for bucketing and a fusion method for image and control embeddings—and a hyperparameter, the number of buckets for noise levels K.

Bucketing strategy. We compare different discretization strategies for constructing bins of bucketing: 1) *Uniform* bucketing where all bins have identical intervals of CLIP similarity and 2) *Quantile* bucketing where each bin has adaptive widths for containing an equal number of samples. Tab. 6(a) presents results where the two strategies show similar CIDEr scores. Uniform bucketing achieves a higher CLIPScore than Quantile bucketing.

Control fusion method. We compare different operations for fusing the control and image embeddings to make an input embedding for the decoder: 1) element-wise summation, 2) concatenation in sequence direction, and 3) concatenation in channel dimension followed by MLP. Tab. 6(b) shows that concatenation brings the best-balanced performance when considering both CIDEr and CLIPScore. We conjecture that this is partly because the condition information remains using the concatenation operation in the fused embedding and thus is more appropriate to directly control the caption generation, while the condition information is smoothed with visual ones in other operations.

The number of bucket bins. Tab. 6(c) shows the results across the number of bucket bins $K = \{4, 8, 16\}$. While our method seems robust to the number of bucket bins, the lowest performance with higher K implies too fine-level bucketing for alignment levels may slightly hinder the learning.

4.7. Qualitative Analysis

Captioning results. Fig. 8 presents generated captions from Vanilla, Vanilla (Filtering), and our model with $z = \{3, 5, 7\}$. Our model successfully generates captions of different quality by adjusting a control signal (z); when feed-

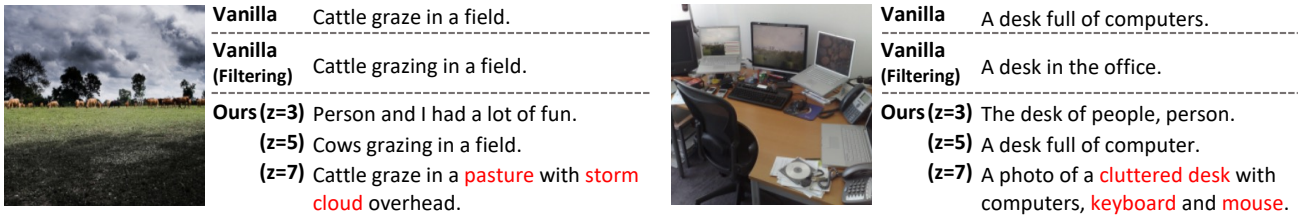

Figure 8: Examples of generated captions. Compared to the baselines, our model can generate captions of different quality by adjusting a control signal (z); as we feed z meaning higher alignment levels ($3 \rightarrow 5 \rightarrow 7$), captions become more descriptive and distinct with expressions (highlighted in red) capturing fine details from images.

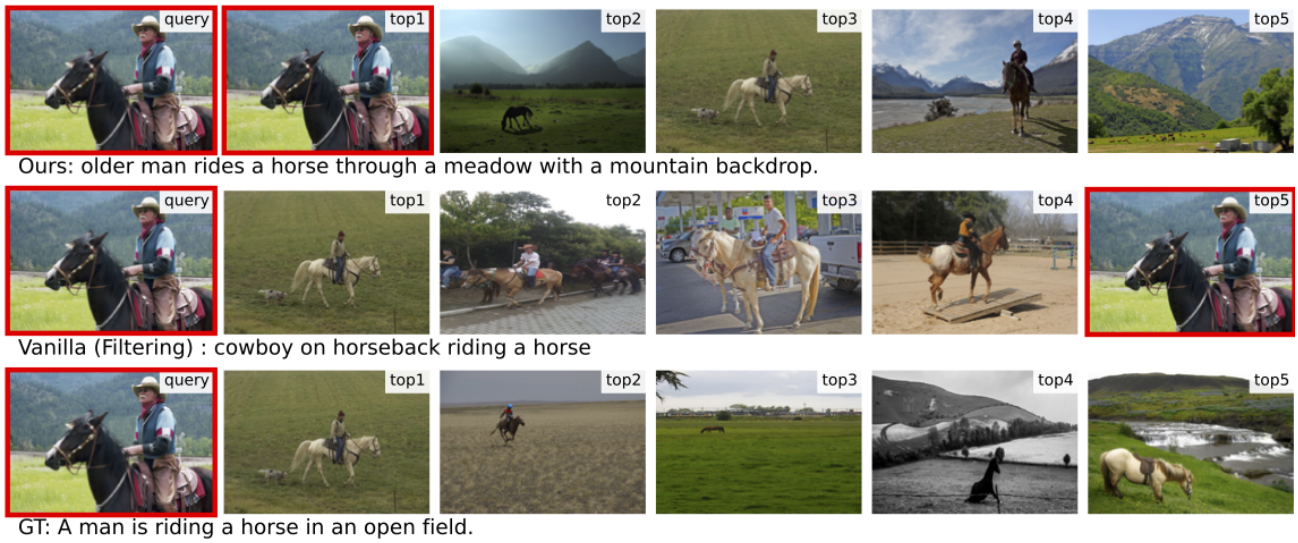

Figure 9: An example of self-retrieval in MSCOCO. In the example, the first column indicates the input image and the generated captions by the specified model, while 2-6th columns show the top-5 retrieved images using the generated captions—by our method and Vanilla (Filtering) baseline—or ground-truth caption. Fine details captured by our model can enhance the search results.

ing z corresponding to higher alignment levels ($3 \rightarrow 5 \rightarrow 7$), captions become more descriptive and distinct by capturing finer-level concepts (*e.g.*, pasture, storm cloud, cluttered desk) from images. In contrast, two baselines generate descriptive but less distinct captions as they typically rely on common words (or phrases) or capture only salient regions. Note that more examples are presented in Appendix H.

Self-retrieval results. We present examples of self-retrieval results on the MSCOCO dataset for Vanilla (Filtering) baseline and our method in Fig. 9. For these models, we generate a caption from an input image (1st column) and retrieve the top-5 images (2-6th columns) using the generated caption. In addition, we also present the retrieval result using ground-truth captions for the given images. Our model successfully captures the main concept (*i.e.*, riding a horse) as well as fine details (e.g., older man, mountain backdrop) that is not captured by the baseline and even in the ground-truth caption. Such captured fine details allow us to search for more similar images for a given image as well as improve the self-retrieval performance. More examples are provided in Appendix H.

5. Conclusion

The recent introduction of large-scale web-crawled data has brought remarkable advances in various computer vision tasks, such as image captioning. However, since the web-crawled data relies on alt-texts, not human annotations, it inevitably includes noisy pairs in a high ratio. Most of the recent works that use large-scale data, however, train models without any consideration for handling the noisy pairs except for simple rule-based filtering strategies. In this paper, we first argue the importance of handling the noise issue in web-crawled image-text data, especially for image captioning. From the comprehensive experiments, our proposed noise-aware learning framework consistently outperforms other carefully designed baselines. We hope our study provides insights to further explore an effective noise-aware learning algorithm for handling inherent noises of a large-scale web-crawled dataset in the future.

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 Captioning at Scale. In *ICCV*, 2019. 2, 5

[2] Devansh Arpit, Stanisław Jastrzebski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A Closer Look at Memorization in Deep Networks. In *ICML*, 2017. 7

[3] Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. Improving language models by retrieving from trillions of tokens. In *ICML*, 2022. 6

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *NIPS*, 2020. 6

[5] Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. COYO-700M: Image-Text Pair Dataset. https://github.com/kakaobrain/coyo-dataset, 2022. 1, 6, 7

[6] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12M: Pushing Web-Scale Image-Text Pre-Training To Recognize Long-Tail Visual Concepts. In *CVPR*, 2021. 2, 6

[7] 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 *CVPR*, 2017. 2

[8] Xinlei Chen and Abhinav Gupta. Webly Supervised Learning of Convolutional Networks. In *ICCV*, 2015. 3

[9] Xinlei Chen and C Lawrence Zitnick. Learning a Recurrent Visual Representation for Image Caption Generation. *arXiv preprint arXiv:1411.5654*, 2014. 2

[10] Lele Cheng, Xiangzeng Zhou, Liming Zhao, Dangwei Li, Hong Shang, Yun Zheng, Pan Pan, and Yinghui Xu. Weakly Supervised Learning with Side Information for Noisy Labeled Images. In *ECCV*, 2020. 3

[11] Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. Meshed-Memory Transformer for Image Captioning. In *CVPR*, 2020. 2

[12] Karan Desai and Justin Johnson. Virtex: Learning visual representations from textual annotations. In *CVPR*, 2021. 4, 12

[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 *CVPR*, 2015. 2

[14] Jacob Goldberger and Ehud Ben-Reuven. Training Deep Neural-Networks using a Noise Adaptation Layer. In *ICLR*, 2017. 3

[15] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust Training of Deep Neural Networks with Extremely Noisy Labels. *NIPS*, 2018. 3

[16] Sen He, Wentong Liao, Hamed R Tavakoli, Michael Yang, Bodo Rosenhahn, and Nicolas Pugeault. Image Captioning through Image Transformer. In *ACCV*, 2020. 2

[17] Simao Herdade, Armin Kappeler, Kofi Boakye, and Joao Soares. Image Captioning: Transforming Objects into Words. *NIPS*, 2019. 2

[18] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. CLIPScore: A Reference-free Evaluation Metric for Image Captioning. In *EMNLP*, 2021. 1, 5

[19] Xiaowei Hu, Zhe Gan, Jianfeng Wang, Zhengyuan Yang, Zicheng Liu, Yumao Lu, and Lijuan Wang. Scaling Up Vision-Language Pre-training for Image Captioning. In *CVPR*, 2022. 2, 3

[20] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision. In *ICML*, 2021. 1, 2, 6

[21] Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Learning Data-driven Curriculum for Very Deep Neural Networks on Corrupted Labels. In *ICML*, 2018. 3

[22] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual Genome: Connecting Language and Vision using Crowdsourced Dense Image Annotations. *IJCV*, 2017. 2

[23] Chenliang Li, Haiyang Xu, Junfeng Tian, Wei Wang, Ming Yan, Bin Bi, Jiabo Ye, Hehong Chen, Guohai Xu, Zheng Cao, et al. mPLUG: Effective and Efficient Vision-Language Learning by Cross-modal Skip-connections. *arXiv preprint arXiv:2205.12005*, 2022. 1, 2

[24] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. 3, 13

[25] Wei Li, Linchao Zhu, Longyin Wen, and Yi Yang. DeCap: Decoding CLIP Latents for Zero-Shot Captioning via Text-Only Training. In *ICLR*, 2023. 6

[26] Xiujun Li, Xi Yin, Chunyuan Li, Xiaowei Hu, Pengchuan Zhang, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks. *ECCV*, 2020. 2

[27] 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 *ECCV*, 2014. 2, 5

[28] Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. In *ICLR*, 2019. 5, 12

[29] Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. Knowing When to Look: Adaptive Attention via a Visual Sentinel for Image Captioning. In *CVPR*, 2017. 2

[30] 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. In *AAAI*, 2021. 2

[31] Xingjun Ma, Hanxun Huang, Yisen Wang, Simone Romano, Sarah Erfani, and James Bailey. Normalized Loss Functions for Deep Learning with Noisy Labels. In *ICML*, 2020. 3

[32] Junhua Mao, Wei Xu, Yi Yang, Jiang Wang, and Alan L. Yuille. Deep Captioning with Multimodal Recurrent Neural Networks (m-RNN). In *ICLR*, 2015. 2

[33] Jonghwan Mun, Minsu Cho, and Bohyung Han. Text-guided Attention Model for Image Captioning. In *AAAI*, 2017. 2

[34] Duc Tam Nguyen, Chaithanya Kumar Mummadi, Thi Phuong Nhung Ngo, Thi Hoai Phuong Nguyen, Laura Beggel, and Thomas Brox. SELF: Learning to Filter Noisy Labels with Self-Ensembling. In *ICLR*, 2020. 3

[35] Vicente Ordonez, Girish Kulkarni, and Tamara L. Berg. Im2Text: Describing Images Using 1 Million Captioned Photographs. In *NIPS*, 2011. 6

[36] Yingwei Pan, Ting Yao, Yehao Li, and Tao Mei. X-Linear Attention Networks for Image Captioning. In *CVPR*, 2020. 2

[37] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k Entities: Collecting Region-to-Phrase Correspondences for Richer Image-to-Sentence Models. In *ICCV*, 2015. 5

[38] 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. In *ICML*, 2021. 1, 5, 15

[39] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners. 2019. 6

[40] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-Shot Text-to-Image Generation. In *ICML*, 2021. 1

[41] Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to Reweight Examples for Robust Deep Learning. In *ICML*, 2018. 3

[42] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. LAION-400M: Open Dataset of CLIP-Filtered 400 Million Image-Text Pairs. *arXiv preprint arXiv:2111.02114*, 2021. 1, 2, 5

[43] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset For Automatic Image Captioning. In *ACL*, 2018. 2, 5

[44] Yoad Tewel, Yoav Shalev, Idan Schwartz, and Lior Wolf. Zerocap: Zero-shot image-to-text generation for visual-semantic arithmetic. In *CVPR*, pages 17918–17928, 2022. 6

[45] Kenneth Tran, Xiaodong He, Lei Zhang, Jian Sun, Cornelia Carapcea, Chris Thrasher, Chris Buehler, and Chris Sienkiewicz. Rich Image Captioning in the Wild. In *CVPRW*, 2016. 2

[46] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and Tell: A Neural Image Caption Generator. In *CVPR*, 2015. 2

[47] Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. GIT: A Generative Image-to-text Transformer for Vision and Language. *arXiv preprint arXiv:2205.14100*, 2022. 1, 2, 3, 12

[48] Qingzhong Wang, Jia Wan, and Antoni B Chan. On Diversity in Image Captioning: Metrics and Methods. *PAMI*, 2020. 4

[49] Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. SimVLM: Simple Visual Language Model Pretraining with Weak Supervision. In *ICLR*, 2022. 1, 2, 3, 6

[50] 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 *ICML*, 2015. 2

[51] Zhuolin Yang, Wei Ping, Zihan Liu, Vijay Korthikanti, Weili Nie, De-An Huang, Linxi Fan, Zhiding Yu, Shiyi Lan, Bo Li, et al. Re-ViLM: Retrieval-Augmented Visual Language Model for Zero and Few-Shot Image Captioning. *arXiv preprint arXiv:2302.04858*, 2023. 6

[52] Jiangchao Yao, Jiajie Wang, Ivor W Tsang, Ya Zhang, Jun Sun, Chengqi Zhang, and Rui Zhang. Deep Learning from Noisy Image Labels with Quality Embedding. *TIP*, 2018. 3

[53] Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. FILIP: Fine-grained Interactive Language-Image Pre-Training. In *ICLR*, 2022. 15

[54] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive Captioners are Image-Text Foundation Models. *arXiv preprint arXiv:2205.01917*, 2022. 1, 2, 3

[55] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A New Foundation Model for Computer Vision. *arXiv preprint arXiv:2111.11432*, 2021. 15

[56] Andy Zeng, Maria Attarian, brian ichter, Krzysztof Marcin Choromanski, Adrian Wong, Stefan Welker, Federico Tombari, Aveek Purohit, Michael S Ryoo, Vikas Sindhwani, Johnny Lee, Vincent Vanhoucke, and Pete Florence. Socratic Models: Composing Zero-Shot Multimodal Reasoning with Language. In *ICLR*, 2023. 6

[57] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding Deep Learning Requires Rethinking Generalization. In *ICLR*, 2017. 7

[58] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. VinVL: Making Visual Representations Matter in Vision-Language Models. *CVPR*, 2021. 2

[59] Zhilu Zhang and Mert Sabuncu. Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels. *NIPS*, 2018. 3