Enhancing Compositional Reasoning in CLIP via Reconstruction and Alignment of Text Descriptions

Jihoon Kwon
Seoul National University
kog0712@snu.ac.kr

Kyle Min*
Oracle
kyle.min@oracle.com

Jy-yong Sohn†
Yonsei University
jysohn1108@yonsei.ac.kr

Abstract

Despite recent advances, vision-language models trained with standard contrastive objectives still struggle with compositional reasoning – the ability to understand structured relationships between visual and linguistic elements. This shortcoming is largely due to the tendency of the text encoder to focus on individual words rather than their relations, a limitation reinforced by contrastive training that primarily aligns words with visual objects. In this paper, we introduce *REconstruction and Alignment of text Descriptions* (READ), a fine-tuning method designed to enhance compositional reasoning by adding two auxiliary objectives to the contrastive learning: (1) a token-level *reconstruction* objective, where a frozen pre-trained decoder reconstructs alternative captions based on the embedding of the original caption; and (2) a sentence-level *alignment* objective, which explicitly aligns paraphrased sentences in the embedding space. We show that READ-CLIP, a model derived by applying the READ method to the pre-trained CLIP model, achieves the state-of-the-art performance across five major compositional reasoning benchmarks, outperforming the strongest conventional fine-tuning baseline by up to 4.1%. Furthermore, applying the READ to existing CLIP variants (including NegCLIP and FSC-CLIP) also improves performance on these benchmarks. Quantitative and qualitative analyses reveal that our proposed objectives – reconstruction and alignment – offer complementary benefits: the former encourages the encoder to capture relationships between words within a caption, while the latter ensures consistent representations for paraphrases expressed with different wording.

1 Introduction

Recent advances in Vision-Language Models (VLMs) have significantly enhanced the ability to align images with text descriptions [8, 67]. A key driver of this progress is contrastive pre-training, such as CLIP [45], which learns to embed images and texts into a shared multi-modal space, in a way that the distance in the embedding space represents the semantic similarity of image-text pairs. VLMs trained with this standard contrastive objective have been widely applied to diverse downstream tasks, including open-vocabulary object detection [17, 72], semantic segmentation [15, 30, 34], cross-modal retrieval [12, 36, 71], and multi-modal generation [1, 47, 53].

Despite their remarkable progress, current VLMs still face challenges with *compositional reasoning* – the ability to understand structured relationships between visual and linguistic elements [37, 56, 64]. Numerous studies have shown that VLMs commonly fail on even simple compositional tasks that humans find straightforward [11, 19, 24, 28, 40, 42, 58, 66, 69]. For instance, when given an image of a horse eating grass, VLMs often assign a higher similarity score to the incorrect caption "the grass is eating the horse" than to the correct caption "the horse is eating the grass", highlighting the limitation of the VLMs in capturing syntactic and relational structures [64]. These failures underscore the need for further research on compositional reasoning to achieve reliable and robust vision-language understanding in real-world applications [4, 9, 29, 33, 39, 56, 57, 60].

*Work partially done while at Intel Labs. †Corresponding author

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

https://doi.org/10.52202/085713-0468

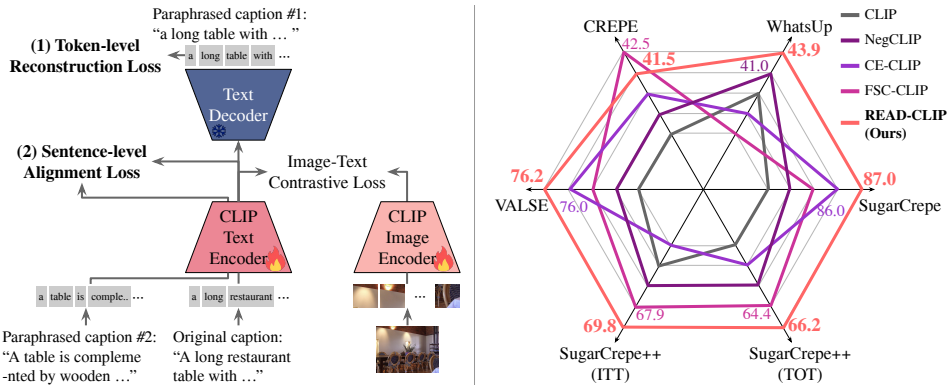

Figure 1: Overview of the training objectives used in READ-CLIP **(left)**, a VLM that applies *REconstruction and Alignment of text Descriptions* (READ) method to the pretrained CLIP model [45]. READ is our proposed fine-tuning method that enhances compositional reasoning in VLMs by augmenting contrastive learning with two auxiliary objectives. The auxiliary objectives consist of two components: *token-level reconstruction* and *sentence-level alignment*. The performance of READ-CLIP **(right)** on compositional reasoning benchmarks demonstrates that it consistently outperforms conventional state-of-the-art methods across diverse aspects of compositional reasoning.

Prior works have identified the *text encoder* as a primary bottleneck for compositional reasoning in VLMs trained with the contrastive objective. Specifically, the text encoder often fails to capture the relationship between words in a sentence [10, 23, 65]. This limitation is largely attributed to the *contrastive* objective, which trains the text encoder to align a caption with its corresponding image, without encouraging the encoder to capture the relationships between words [10, 64]. As a result, the text encoder focuses on words that refer to objects depicted in the image, as they mainly contribute to the image-text alignment, thereby limiting its ability to learn compositional reasoning [7].

Conventional approaches to overcome this limitation can be categorized into two parts. The first approach is to modify the contrastive objective by introducing *hard negatives* [48] – semantically different examples that are nonetheless difficult for the model to distinguish from the positives. Here, the model is trained to bring each sample closer to its positives and push it farther from hard negatives, thus improving the compositional reasoning capability [41, 50, 54, 64]. However, these approaches that rely solely on hard negatives often encourage the model to focus on patterns specific to those negatives, rather than developing genuine compositional reasoning [13, 14, 19]. The second approach is to add auxiliary objectives to the standard contrastive objective. However, these efforts either supervise both image and text encoders jointly [51, 70] or focus solely on the image encoder [2, 18]. Although the text encoder serves as a primary bottleneck for compositional reasoning, limited attention has been paid to adopting auxiliary objectives for the text encoder, aimed at improving the compositional reasoning capability.

In order to enhance the compositional reasoning capability of VLMs more effectively, we tackle the underexplored challenge of improving the text encoder via a targeted training objective. Specifically, our contributions are as follows:

- In Sec. 3, we propose *REconstruction and Alignment of text Descriptions* (READ), a fine-tuning method designed to enhance the text encoder by adding two auxiliary objectives to the standard contrastive objective: (1) token-level reconstruction and (2) sentence-level alignment, as in Fig. 1. First, the *token-level reconstruction* objective trains the text encoder to produce embeddings from the original caption that enable a frozen decoder to reconstruct each token of an alternative caption. Second, the *sentence-level alignment* objective explicitly aligns paraphrased captions in the embedding space to reflect their shared semantics, even when they are expressed differently.

- In Sec. 4, we provide experiments demonstrating the READ method is effective across a wide range of compositional reasoning benchmarks. Specifically, we introduce READ-CLIP, a VLM derived by applying the READ method to the pre-trained CLIP model [45], which achieves the state-of-the-art performance on five compositional reasoning benchmarks. READ-CLIP outperforms the famous baseline NegCLIP [64] by an average of 4.5% across benchmarks, and outperforms the strongest baseline FSC-CLIP [38] by up to 4.1%. Furthermore, applying

https://doi.org/10.52202/085713-0468

the READ method to existing CLIP variants (including NegCLIP and FSC-CLIP) consistently improves performance across these benchmarks, with gains of up to 2.4%.

- Our analysis in Sec. 5 demonstrates that the two objectives in the READ method – reconstruction and alignment – provide complementary benefits for compositional reasoning. The former encourages the encoder to capture relationships between words within a caption, while the latter ensures consistent representations for paraphrases even expressed with different wording. We also find that reconstructing an alternative caption, rather than the original caption, reduces overfitting to exact wording and improves the ability of VLMs to learn relational understanding.

2 Related Work

Compositional Reasoning in Contrastive VLMs. VLMs trained with the contrastive objective often struggle with compositional reasoning [56, 64], as the text encoder tends to overlook relationships between words due to the training objective that prioritizes image-text alignment based on object mentions [7, 10, 23, 64, 65]. To address this limitation, a common approach is to introduce hard negatives by modifying the contrastive objective. These approaches typically generate hard negative captions via rule-based perturbation [6, 64], language models [7, 68], scene graphs [20, 18, 54], or construct hard negative pairs by altering both text and image [3, 41, 51]. These methods have been shown to be effective; for example, NegCLIP improves over CLIP by 23.4% on ARO [64], and CE-CLIP [68] achieves a 7.2% gain on VALSE [40, 68]. Among these, DAC [6] highlights that training with well-aligned captions improves compositional reasoning, while TSLVC [7] finds that using paraphrased captions in analogy loss improves image classification performance.

Beyond contrastive learning, recent work has proposed adding auxiliary objectives to improve the compositional reasoning. Some methods supervise both image and text encoders, such as SF-CLIP [51], which uses masked distillation from pre-trained models, and IL-CLIP [70], which employs codebook alignment and iterative re-initialization. Other approaches target only the image encoder: SDS-CLIP [2] uses distillation from diffusion models [49], and CLIP-SGVL [18] introduces a scene-graph loss. Although the text encoder has been identified as the primary bottleneck [10, 23, 65], approaches that specifically introduce auxiliary objectives for the text encoder for improved compositional reasoning capability of VLMs remain scarce.

Reconstruction Objectives for Training Encoders. For the purpose of language understanding, various recent works have focused on training a text encoder-decoder architecture in a way that the sentence put into the encoder is reconstructed at the output of the decoder. It is reported that such *reconstruction objective* is beneficial for improving the performance of encoders on various language understanding benchmarks [26, 32, 59]. For instance, MASS [55] reconstructs masked fragments of the original sentence, while RetroMAE [62] reconstructs the original sentence from a pooled embedding. These approaches have shown that the auxiliary reconstruction objective can encourage the text encoder to capture both syntactic and semantic relationships among the words in the sentence [55, 62]. However, reconstructing the caption under the encoder-decoder structure has not been explored as a training objective for VLMs. Despite the use of auxiliary objectives in VLMs [31, 63], these approaches do not aim to reconstruct the input caption in the text modality. We introduce a text reconstruction loss during fine-tuning, aiming to enhance the compositional reasoning ability of VLMs.

3 Method

In this section, we formally define our proposed *REconstruction and Alignment of text Descriptions* (READ) method for improving the compositional reasoning performance of VLMs. The READ method is a fine-tuning method using three types of losses: a conventional *contrastive* loss reviewed in Sec. 3.1 and two auxiliary losses proposed in Sec. 3.2 and Sec. 3.3, namely, the token-level *reconstruction* loss and the sentence-level *alignment* loss. The final form of the fine-tuning loss in the READ method is given in Sec. 3.4.

3.1 Contrastive Loss

We consider a batch of B image–text pairs, denoted as $\{(I_i, T_i)\}_{i=1}^{B}$, where I_i and T_i represent the i-th image and its associated caption. The image and text encoders are denoted by f_I and f_T, which produce embeddings $u_i = f_I(I_i)$ and $v_i = f_T(T_i)$, respectively. For convenience, we define the index set $[B] := \{1, 2, \ldots, B\}$.

 https://doi.org/10.52202/085713-0468

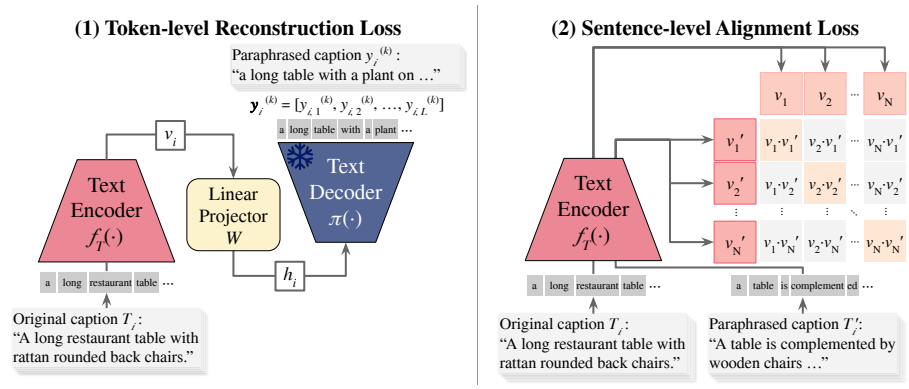

Figure 2: Illustration of our proposed auxiliary objectives of *REconstruction and Alignment of text Description* (READ) method. Given pairs of captions – an original and its paraphrase – that share a common meaning, the *token-level reconstruction* (1) trains the text encoder to produce embeddings from the original caption such that a frozen pre-trained decoder can reconstruct each token of the paraphrased caption. This reconstruction encourages the encoder to capture relationships between words within a caption, which are critical for reconstructing its paraphrase. In contrast, the *sentence-level alignment* (2) aligns the pair of captions in the embedding space. This alignment encourages the encoder to capture underlying semantic relationships across the paraphrased captions.

Suppose we are given a batch of image-text pairs $\{(I_i, T_i)\}_{i=1}^B$. For each $i, j \in [B]$, the similarity between the i-th image I_i with embedding u_i and the j-th text T_j with embedding v_j is defined as $\phi(I_i, T_j) := \exp\left(\cos\left(u_i, v_j\right)/\tau\right)$ where τ is a learnable temperature parameter. The standard contrastive losses used in CLIP [45] is represented as

$$\mathcal{L} = \frac{1}{2}\left(\mathcal{L}_{I\to T} + \mathcal{L}_{T\to I}\right), \tag{1}$$

where each component is defined as

$$\mathcal{L}_{I\to T} = -\frac{1}{B}\sum_{i=1}^{B}\log\frac{\phi(I_i,\ T_i)}{\sum_{j=1}^{B}\phi(I_i,\ T_j)}, \quad \mathcal{L}_{T\to I} = -\frac{1}{B}\sum_{i=1}^{B}\log\frac{\phi(T_i,\ I_i)}{\sum_{j=1}^{B}\phi(T_i,\ I_j)}, \tag{2}$$

which are dubbed as image-to-text loss and the text-to-image loss, respectively. Note that for the standard loss in Eq. 1, each image I_i has one *positive* caption T_i and $B-1$ *negative* captions $\{T_j\}_{j\neq i}$.

The READ method uses a variant of the standard contrastive loss in Eq. 1 to improve the compositional reasoning capability, motivated by the following two observations. First, prior works [10, 23, 65] have identified the *text encoder* as a primary bottleneck for compositional reasoning in VLMs trained with the contrastive objective. Second, recent works [41, 54, 64] showed that compositional reasoning of CLIP model is improved by introducing additional *hard negative* captions – semantically different captions that are nonetheless difficult for the model to distinguish from the positives – into the training loss. Motivated by these observations, the contrastive loss in our proposed READ method uses hard negatives in the text domain, details of which are given as below.

For each sample index i, let $\{\tilde{T}_i^{(m)}\}_{m=1}^M$ be the set of M hard negative captions associated with the positive caption T_i. Incorporating these hard negatives into the denominator, the image-to-text loss in Eq. 2 is modified as

$$\mathcal{L}'_{I\to T} = -\frac{1}{B}\sum_{i=1}^{B}\log\frac{\phi(I_i,\ T_i)}{\sum_{j=1}^{B}[\phi(I_i,\ T_j) + \sum_{m=1}^{M}\phi(I_i,\ \tilde{T}_j^{(m)})]}. \tag{3}$$

Inserting this modified image-to-text loss in Eq. 1, the contrastive loss in the READ method is

$$\mathcal{L}_{\text{Contrastive}} = \frac{1}{2}\left(\mathcal{L}'_{I\to T} + \mathcal{L}_{T\to I}\right). \tag{4}$$

3.2 Token-Level Reconstruction Loss

Here we define our proposed token-level reconstruction loss and discuss how this loss promotes compositional reasoning. Given the i-th image–text pair (I_i, T_i), we consider a set of K captions

$\{\mathbf{y}_i^{(k)}\}_{k=1}^K$ describing the same image I_i, which serve as target sequences for the reconstruction. We refer to this set as the alternative captions of I_i. As shown in the left-hand side of Fig. 2, the token-level reconstruction loss measures how well each token in the alternative caption $\mathbf{y}_i^{(k)}$ is reconstructed at the output of the text decoder, once the original caption T_i is given as the input of the text encoder.

To be specific, let $v_i = f_T(T_i)$ be the text embedding of the original caption T_i for the i-th sample. In order to reconstruct texts from the embedding v_i, we employ a pre-trained frozen decoder π. While the text encoder f_T is initialized from a pre-trained text encoder, one can use any off-the-shelf pre-trained text decoder for π. Thus, the encoder and the decoder may have different embedding dimensions. To address this potential difference, we introduce a learnable projector W that maps the encoder output to the decoder input. Specifically, given the text embedding v_i, we apply a linear projection to obtain $h_i = W^\top v_i$, which is used as the input to the decoder π. Then, for each $k \in [K]$, the decoder predicts the k-th alternative caption $\mathbf{y}_i^{(k)} = [y_{i,1}^{(k)}, \ldots, y_{i,L}^{(k)}]$ composed of L tokens, conditioned on the projected embedding h_i. Thus, the token-level reconstruction loss is defined as

$$\mathcal{L}_{\text{Token Reconstruction}} = -\frac{1}{B \cdot K} \sum_{i=1}^{B} \sum_{k=1}^{K} \log \pi \left(\mathbf{y}_i^{(k)} \mid h_i \right), \tag{5}$$

where the log-likelihood value is

$$\log \pi \left(\mathbf{y}_i^{(k)} \mid h_i \right) = \sum_{t=1}^{L} \log \pi \left(y_{i,t}^{(k)} \mid y_{i,<t}^{(k)},\ h_i \right). \tag{6}$$

Now we discuss how the proposed loss is beneficial for improving the compositional reasoning performance. Since the decoder π is frozen and conditioned solely on the text embedding v_i, the proposed reconstruction loss trains the text encoder f_T to embed a caption T_i such that a frozen pre-trained decoder can reconstruct the tokens in the alternative caption. The encoder f_T is thus encouraged to capture the relationships between words within a caption, as these are necessary for reconstructing its alternatives, which promotes the compositional reasoning.

3.3 Sentence-Level Alignment Loss

Effective compositional reasoning requires not only understanding the relationships between words within a sentence, but also recognizing semantic similarity even when sentences convey the same meaning using different expressions. To this end, as shown in the right-hand side of Fig. 2, we additionally employ a sentence-level alignment loss to explicitly align text embeddings of paraphrased captions that describe the same image. For each image-text pair (I_i, T_i), we generate a paraphrase T_i' of T_i through augmentation, forming a text pair (T_i, T_i'). Here, the pair (T_i, T_i') is treated as positive, while paraphrases (T_i, T_j') from other samples in the batch serve as negatives. The sentence-level alignment loss is then defined as

$$\mathcal{L}_{\text{Sentence Alignment}} = -\frac{1}{B} \sum_{i=1}^{B} \log \frac{\phi(T_i,\ T_i')}{\sum_{j=1}^{B} \phi(T_i,\ T_j')}. \tag{7}$$

where ϕ is the similarity metric defined in Sec. 3.1 with slight abuse of notation[2]. This alignment loss encourages the encoder to embed paraphrased captions (T_i, T_i') close together in the embedding space, thus letting the encoder capture the semantic relationships between sentences that express the same meaning using different wording and phrasing.

3.4 Fine-Tuning Loss of The READ Method

The fine-tuning loss used for the READ method combines the above components:

$$\mathcal{L}_{\text{READ}} = \mathcal{L}_{\text{Contrastive}} + \alpha\, \mathcal{L}_{\text{Token Reconstruction}} + \beta\, \mathcal{L}_{\text{Sentence Alignment}}, \tag{8}$$

where α and β are hyperparameters controlling the relative contribution of the auxiliary losses. Together, these losses operate in a complementary way by capturing relational structure at different levels: the token-level reconstruction loss captures relationships between words within a sentence, while the sentence-level alignment loss captures semantic similarity across paraphrased sentences.

[2]The definition of ϕ in Sec. 3.1 compares an image and a sentence, while here we compare sentences

https://doi.org/10.52202/085713-0468

Table 1: Compositional reasoning performance (%) of the pre-trained CLIP model (ViT-B/32, top row) and its fine-tuned variants (rows 2–7) across five major benchmarks. All models are fine-tuned on 100K samples from the MS-COCO dataset [35]. Among various fine-tuning methods, READ-CLIP achieves the highest average accuracy of 64.1%.

Models	WhatsUp	VALSE	CREPE	SugarCrepe	SugarCrepe++		Avg.
					ITT	TOT	
CLIP [45] (ViT-B/32)	41.0	67.4	23.9	73.2	60.0	46.7	52.0
Fine-tuned: MS-COCO, 100K Samples							
Triplet-CLIP [41]	41.6	64.2	15.0	82.7	61.7	57.4	53.8
GNM-CLIP [50]	41.6	70.7	17.4	77.9	60.2	60.0	54.6
CE-CLIP [68]	40.7	<u>76.0</u>	34.8	<u>86.0</u>	55.7	57.0	58.4
NegCLIP [64]	<u>42.4</u>	73.7	30.5	83.6	65.0	62.5	59.6
FSC-CLIP [38]	39.8	74.4	**42.5**	85.2	<u>67.9</u>	<u>64.4</u>	62.4
READ-CLIP (Ours)	**43.9**	**76.2**	<u>41.5</u>	**87.0**	**69.8**	**66.2**	**64.1**

4 Experiments

In this section, we empirically evaluate the effectiveness of our proposed READ method. To be specific, we fine-tune the pre-trained CLIP model using the READ method, where the fine-tuned model is dubbed as READ-CLIP. We compare READ-CLIP with various baselines on major compositional reasoning benchmarks. We begin in Sec. 4.1 by describing the experimental setup and present the experimental results in Sec. 4.2. Codes are available at this GitHub repository.

4.1 Experimental Setup

Training: We use the MS-COCO dataset [35] for all experiments. We follow training practices established in prior work on compositional reasoning [38, 54, 64, 68], using a 100K subsample with the Karpathy split [25], 5 training epochs, a batch size of 256, and the ViT-B/32 architecture. As defined in Eq. 8, our training loss consists of three components: the standard contrastive loss, the token-level reconstruction loss, and the sentence-level alignment loss. We provide implementation details for each component, including hyperparameters, and other specifics, in Appendix A.1.

Baselines: We compare READ-CLIP against recent state-of-the-art fine-tuning methods designed to improve compositional reasoning in VLMs. To evaluate against a method relying solely on hard negatives, we include NegCLIP [64], which uses rule-based negatives. To compare with methods that construct synthetic image-text negative pairs, we include GNM-CLIP [50] and Triplet-CLIP [41]. We also consider methods that improve the effectiveness of hard negative captions by incorporating multiple contrastive objectives, including CE-CLIP [68] and FSC-CLIP [38].

Evaluation: We evaluate READ-CLIP and the baselines on five benchmarks—WhatsUp [24], CREPE [37], VALSE [40], SugarCrepe [19], and SugarCrepe++ [11]—each designed to assess a different aspect of compositional reasoning. All benchmarks are evaluated using accuracy, which measures whether positive pairs are ranked above all negatives. For each benchmark containing multiple subtasks, we report the accuracy averaged over the subtasks, following prior work [38, 68]. Details of each benchmark are provided in Appendix A.2.

4.2 Results

READ-CLIP outperforms baselines on various compositional benchmarks. Table 1 reports compositional reasoning performance of READ-CLIP and baselines across five benchmarks. READ-CLIP achieves the highest average accuracy of 64.1%, outperforming the pre-trained CLIP by 12.1%, NegCLIP—a strong and widely cited baseline—by 4.5%, and the second-best model FSC-CLIP by 1.7%. Notably, READ-CLIP ranks first on four benchmarks and second on the remaining one, demonstrating consistently strong performance. This consistent outperformance highlights the advantage of enhancing compositional reasoning in the text encoder through the READ method.

Reconstruction and alignment losses provide complementary benefits. We conduct an ablation study to analyze the benefits of the two auxiliary losses introduced in the READ method: token-level reconstruction and sentence-level alignment, summarized in Table 2. Compared to the contrastive-only baseline in row 1, adding token-level reconstruction in row 2 improves average accuracy from

Table 2: Ablation study analyzing how the reconstruction and alignment objectives in our proposed READ method contribute to its performance of READ-CLIP, both individually and jointly. The reconstruction loss improves accuracy on WhatsUp, CREPE, VALSE, and SugarCrepe. The inclusion of the alignment loss improves SugarCrepe++. The combination of both losses results in the highest overall accuracy, indicating their combined contributions to compositional reasoning.

	Reconstruction Loss	Alignment Loss	WhatsUp	VALSE	CREPE	SugarCrepe	SugarCrepe++		Avg.
							ITT	TOT	
(1)			40.5	74.7	38.4	86.4	69.3	66.0	62.2
(2)	✓		<u>43.6</u>	**76.6**	**41.6**	<u>86.9</u>	69.7	64.8	<u>63.9</u>
(3)		✓	43.0	75.5	40.6	86.8	**70.2**	**67.0**	63.8
(4)	✓	✓	**43.9**	<u>76.2</u>	<u>41.5</u>	**87.0**	<u>69.8</u>	<u>66.2</u>	**64.1**

Table 3: Analysis of the impact of key hyperparameter selection on the average accuracy (%) of our proposed READ method: weights for the token reconstruction loss (α) and sentence alignment loss (β), the number of target sequences (K) used in the token reconstruction loss, and the size of the T5 [46] decoder model employed for computing the reconstruction loss. Gray cells indicate the hyperparameter configuration that yields the highest average accuracy.

α (Token Reconst. Loss)		β (Sentence Align. Loss)		K (Num. of Targets)		Decoder Model (T5 [46])	
Value	Avg. Acc.	Value	Avg. Acc.	Value	Avg. Acc.	Size	Avg. Acc.
-	63.8	-	63.9	-	63.8	-	63.8
0.1	64.1	0.1	63.4	1	64.1	Small	64.0
0.2	64.0	0.2	64.0	2	63.5	Base	64.1
0.5	63.4	0.5	64.1	3	63.9	Large	64.1
1.0	62.7	1.0	64.0	4	64.0	XL	63.4
2.0	61.8	2.0	63.8	5	63.7	XXL	63.9

62.2% to 63.9%, notably enhancing WhatsUp by 3.1%, VALSE by 1.9%, and CREPE by 3.2%, while adding sentence-level alignment in row 3 substantially enhances SugarCrepe++ ITT to 70.2% and TOT to 67.0%. Combining both losses in row 4 achieves the highest average accuracy of 64.1%, confirming that each objective offers complementary benefit.

READ-CLIP is robust to hyperparameter selection. To assess the robustness of READ, we analyze its sensitivity to four major hyperparameters used for training READ-CLIP: the weight α and β for the auxiliary objectives in the READ method, the number K of target sequences, and the size of the T5 decoder [46] used for reconstruction. Table 3 summarizes the average accuracy across the five benchmarks for each configuration. The performance of READ-CLIP remains stable over a wide range of hyperparameter values. In particular, varying the loss weights α and β results in only modest performance differences. Also, it achieves strong performance even with a single target sequence ($K = 1$) and a T5-Large decoder, demonstrating robust gains with minimal computational overhead.

READ provides consistent gains when applied to diverse fine-tuning methods. The experiments so far applied READ to a modified CLIP objective (Eq. 4) that incorporates hard negative captions only. To establish broader applicability, it is crucial to verify whether the READ method consistently provides gains when applied on top of diverse fine-tuning methods. Therefore, we evaluate READ alongside three baselines: (1) naive CLIP with standard contrastive loss (Eq. 1); (2) NegCLIP [41], and (3) FSC-CLIP [38], without additional hyperparameter tuning. Table 4 presents the results of applying the READ to these three fine-tuning baselines, illustrating its impact across diverse fine-tuning methods. Across all three settings, augmenting the baseline with READ leads to consistent performance improvements on the majority of benchmarks. When applied to naive CLIP, READ improves the average accuracy from 57.0% to 58.2%, with consistent gains across all five benchmarks. For NegCLIP and FSC-CLIP, the average accuracy increases from 61.5% to 62.5% and from 62.8% to 63.6%, respectively. In both cases, READ leads to clear improvements on WhatsUp, VALSE, CREPE, and SugarCrepe, while maintaining comparable performance on SugarCrepe++.

https://doi.org/10.52202/085713-0468

Table 4: Performance comparison of baseline fine-tuning methods—CLIP [45], NegCLIP [64], and FSC-CLIP [38]—and their READ-augmented counterparts, to assess the effectiveness of READ method. READ consistently improves performance on WhatsUp, VALSE, CREPE, and SugarCrepe, while preserving comparable accuracy on SugarCrepe++.

	WhatsUp	VALSE	CREPE	SugarCrepe	SugarCrepe++		Avg.
Methods					ITT	TOT	
Fine-tuned: MS-COCO, 100K Samples, 5 epoch							
CLIP	41.3	70.4	15.5	81.8	66.2	66.5	57.0
+ *READ*	43.3 (+2.0)	71.3 (+0.9)	17.3 (+1.8)	83.0 (+1.2)	68.2 (+2.0)	66.2 (-0.3)	58.2 (+1.2)
NegCLIP	41.3	75.4	34.4	84.5	68.0	65.6	61.5
+ *READ*	43.7 (+2.4)	76.5 (+1.1)	36.7 (+2.3)	85.2 (+0.7)	68.1 (+0.1)	64.9 (-0.7)	62.5 (+1.0)
FSC-CLIP	41.3	73.9	42.7	85.8	68.1	65.1	62.8
+ *READ*	43.2 (+1.9)	74.4 (+0.5)	45.1 (+2.4)	86.6 (+0.8)	67.1 (-1.0)	64.8 (-0.3)	63.6 (+0.8)

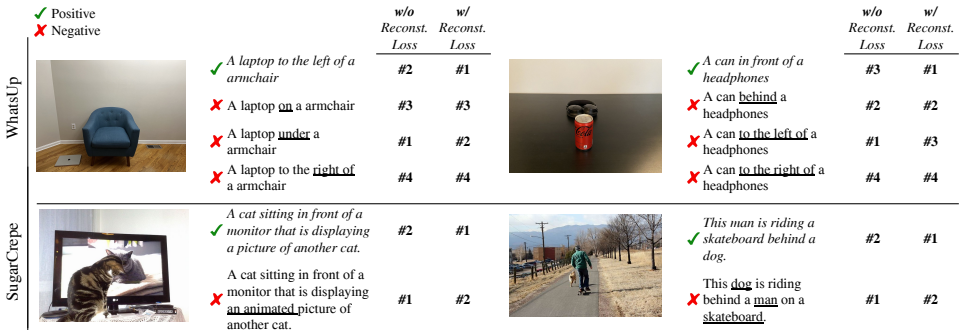

Figure 3: Representative examples illustrating how applying the *reconstruction loss* affects caption rankings based on image-caption cosine similarity on WhatsUp [24] and SugarCrepe [19]. Positive (✓) and negative captions (✗) are shown with their rankings based on image-caption cosine similarity.

5 Analysis

Our results in Sec. 4 show that the reconstruction and alignment losses in the READ method consistently improve compositional reasoning, with each offering complementary benefits. To better understand the role of our proposed loss, we present a qualitative analysis in Sec. 5.1 and Sec. 5.2. We then quantitatively analyze the key components of each loss. In Sec. 5.3, we examine the benefit of reconstructing an alternative caption instead of the original one within the *token-level reconstruction* loss. Subsequently, Sec. 5.4 investigates how the quality and diversity of LLM-generated paraphrases used in the *sentence-level alignment* loss affect compositional reasoning.

5.1 Token-Level Reconstruction Enhances Encoding of Compositional Relationships

Fig. 3 illustrates the effect observed in Table 2 (rows 1 vs. 2), showing that incorporating the reconstruction loss improves compositional reasoning across benchmarks. Specifically, we observe that the reconstruction loss helps lower the ranking of negative captions that differ from positive ones. These negative captions typically involve subtle structural edits—such as swapping, replacing, or inserting single words or short phrases—that preserve most of the original wording while altering the underlying meaning. This improved discrimination between correct captions and their negative counterparts suggests that the reconstruction loss enables the encoder to recognize semantic differences between those captions by capturing the relationships between words.

5.2 Sentence-Level Alignment Promotes Semantic Consistency

Fig. 4 illustrates the effect observed in Table 2 (rows 1 vs. 3), showing how the alignment loss enhances compositional reasoning by affecting the ranking of two positive captions in SugarCrepe++ [11],

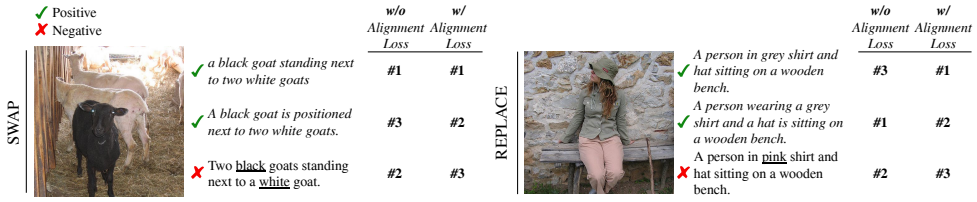

Figure 4: Representative examples from each category (SWAP and REPLACE) of SugarCrepe++ [11] dataset, showing how applying the *alignment loss* improves the ranking of positive captions. In Sugarcrepe++, each image is paired with two positive captions (✓) that are worded differently. In ITT (image-to-text) evaluation, a prediction is considered accurate if both positive captions are ranked higher than all negatives based on image-caption cosine similarity.

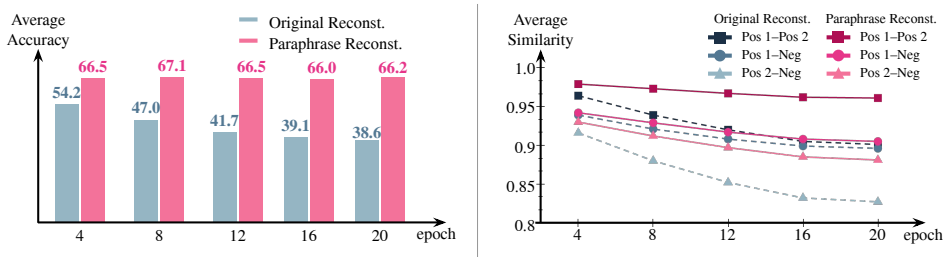

Figure 5: Comparison of reconstructing a paraphrased caption versus the original one, measured by the performance of trained encoder on SugarCrepe++ [11] TOT (text-to-text) benchmark, for various training epochs. In the TOT evaluation, average accuracy **(left)** is computed by checking whether the cosine similarity between the positive caption pair is higher than that of any positive–negative pairs. Average similarity **(right)** measures cosine similarity between caption pairs. Here, `Pos1` and `Pos2` are positive pairs, while others are negative pairs.

where each image is paired with two paraphrases (denoted as `Pos1` and `Pos2`) that convey the same meaning but differ in expression. We observe that applying the alignment loss leads to improved ranking for some positive captions that were previously ranked below negatives. This effect arises because the alignment loss encourages the encoder to embed paraphrased captions closer together in the embedding space, despite differences in wording. Consequently, both captions become embedded closer to the corresponding image than to negative caption, which in turn strengthens the ability in vision-language compositional reasoning.

5.3 Reconstructing Alternative Captions Mitigates Overfitting and Enhances Generalization

Recall that the reconstruction loss in Sec. 3.2 is designed to reconstruct an *alternative* caption rather than the *original* one. In here, we explore the effect of using such alternative caption on the compositional reasoning performance of the trained encoder. To be specific, we conduct experiments on the SugarCrepe++ TOT (text-to-text) benchmark, comparing two variants of READ-CLIP: one that reconstructs the original caption and the other that reconstructs an alternative caption.

The left plot in Fig. 5 shows that reconstructing an alternative caption leads to significantly more stable accuracy across training epochs, whereas using the original caption results in a gradual performance decline. To better understand this effect, in the right plot of Fig. 5, we compare the similarity of positive/negative caption pairs in the trained embedding space. We compare three pairs, where `Pos1`-`Pos2` indicates the positive pair, and other pairs are negative. One can confirm that reconstructing the original caption causes a steady decline in similarity between `Pos1` and `Pos2` over time, which is undesired. This phenomenon can be interpreted as follows: when trained to exactly reconstruct the *original* caption, the encoder increasingly overfits to exact wording and phrasing rather than capturing the underlying relationships between words within a caption. In contrast, reconstructing an *alternative* caption better preserves semantic similarity between positive captions and maintains greater discrimination from negatives. These findings indicate that reconstructing an alternative caption mitigates overfitting to exact wording and enhances generalization in relational understanding.

https://doi.org/10.52202/085713-0468

Table 5: Compositional reasoning performance under varying *quality* of LLM-generated paraphrases in Eq. 7. We randomly replaced 10% or 20% of LLM-generated paraphrases with unrelated captions from the dataset to simulate lower-quality paraphrases during training.

	WhatsUp	VALSE	CREPE	SugarCrepe	SugarCrepe++		Avg.
Model					ITT	TOT	
READ-CLIP	43.9	76.2	41.5	87.0	69.8	66.2	64.1
READ-CLIP (10% Noise)	43.6	76.0	39.0	86.9	67.1	64.7	62.9
READ-CLIP (20% Noise)	43.4	76.0	38.6	86.8	65.1	62.9	62.1

Table 6: Compositional reasoning performance under varying *diversity* of LLM-generated paraphrases in Eq. 7. We generated multiple paraphrases per caption and randomly sampled one during each training step to examine whether increased diversity improves robustness.

	WhatsUp	VALSE	CREPE	SugarCrepe	SugarCrepe++		Avg.
Model					ITT	TOT	
READ-CLIP ($\texttt{num}_p = 1$)	43.9	76.2	41.5	87.0	69.8	66.2	64.1
READ-CLIP ($\texttt{num}_p = 3$)	43.4	76.4	41.1	86.5	70.0	66.4	64.0
READ-CLIP ($\texttt{num}_p = 5$)	43.6	76.0	41.3	86.5	70.8	66.6	64.1

5.4 Sentence-Level Alignment is Robust to Quality and Diversity of Paraphrases

Recall that the sentence-level alignment loss in Sec. 3.3 uses LLM-generated paraphrases to encourage semantic consistency between captions with different wordings. In here, we investigate how the *quality* and *diversity* of these paraphrases affect compositional reasoning performance.

First, we assess whether the performance of compositional reasoning of our proposed READ method is sensitive to lower-quality paraphrases. To this end, we intentionally inject noise by randomly replacing LLM-generated paraphrases with unrelated captions from the dataset at two levels: 10% and 20%. As shown in Table 5, performance drops by only 1.2–2.0% on average when 10–20% of paraphrases are replaced with noise, indicating that the sentence-level alignment loss is reasonably robust to moderate degradation in paraphrase quality.

Next, we examine whether increasing diversity of LLM-generated paraphrases improves the performance of compositional reasoning. We vary the number of LLM-generated paraphrases per caption ($\texttt{num}_p \in \{1, 3, 5\}$) and randomly sample one at each training step. All other components in Eq. 7 remain unchanged, thereby the only difference is the diversity of available paraphrases. Table 6 shows that while increasing $\texttt{num}_p$ slightly improves performance on SugarCrepe++ [11] – where recognizing paraphrased captions as semantically equivalent is critical – the average performance across all benchmarks remains nearly unchanged. This suggests that a single number of LLM-generated paraphrase is already sufficient for effective alignment, and additional diversity provides only marginal benefits.

6 Conclusion

We introduced READ, a fine-tuning method that enhances compositional reasoning in contrastively trained VLMs by integrating token-level reconstruction and sentence-level alignment objectives. READ explicitly captures compositional relationships, enabling READ-CLIP to outperform other fine-tuning baselines across diverse benchmarks. We hope this work provides a practical approach for compositionality-aware fine-tuning of VLMs, and encourages further exploration of auxiliary objectives to strengthen the compositional reasoning ability of text encoders.

Limitation. While our method is designed to leverage multiple captions per image, it can still be applied to the dataset with only a single caption per image by generating additional paraphrases using LLMs, although this introduces additional complexity. In addition, we only used T5 [46] decoder in our reconstruction loss, without exploring the impact of alternative generative architectures [16, 44]. We did not assess the effect of fine-tuning the decoder as well, which may influence the compositional reasoning capability of our proposed method.

Acknowledgements

This work was partially supported by the National Research Foundation of Korea (NRF) grant funded by the Ministry of Science and ICT (MSIT) of the Korean government (RS-2024-00345351, RS-2024-00408003), and Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by MSIT (RS-2023-00259934, RS-2025-02283048).

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

[2] Samyadeep Basu, Shell Xu Hu, Maziar Sanjabi, Daniela Massiceti, and Soheil Feizi. Distilling knowledge from text-to-image generative models improves visio-linguistic reasoning in clip. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6105–6113, 2024.

[3] Paola Cascante-Bonilla, Khaled Shehada, James Seale Smith, Sivan Doveh, Donghyun Kim, Rameswar Panda, Gul Varol, Aude Oliva, Vicente Ordonez, Rogerio Feris, et al. Going beyond nouns with vision & language models using synthetic data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20155–20165, 2023.

[4] Boyuan Chen, Zhuo Xu, Sean Kirmani, Brain Ichter, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14455–14465, 2024.

[5] Mehdi Cherti and Romain Beaumont. Clip benchmark, May 2025.

[6] Sivan Doveh, Assaf Arbelle, Sivan Harary, Roei Herzig, Donghyun Kim, Paola Cascante-Bonilla, Amit Alfassy, Rameswar Panda, Raja Giryes, Rogerio Feris, et al. Dense and aligned captions (dac) promote compositional reasoning in vl models. *Advances in Neural Information Processing Systems*, 36:76137–76150, 2023.

[7] Sivan Doveh, Assaf Arbelle, Sivan Harary, Eli Schwartz, Roei Herzig, Raja Giryes, Rogerio Feris, Rameswar Panda, Shimon Ullman, and Leonid Karlinsky. Teaching structured vision & language concepts to vision & language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2657–2668, 2023.

[8] Yifan Du, Zikang Liu, Junyi Li, and Wayne Xin Zhao. A survey of vision-language pre-trained models. *arXiv preprint arXiv:2202.10936*, 2022.

[9] Jiafei Duan, Wilbert Pumacay, Nishanth Kumar, Yi Ru Wang, Shulin Tian, Wentao Yuan, Ranjay Krishna, Dieter Fox, Ajay Mandlekar, and Yijie Guo. Aha: A vision-language-model for detecting and reasoning over failures in robotic manipulation. *arXiv preprint arXiv:2410.00371*, 2024.

[10] Sri Harsha Dumpala, David Arps, Sageev Oore, Laura Kallmeyer, and Hassan Sajjad. Seeing syntax: Uncovering syntactic learning limitations in vision-language models. *arXiv preprint arXiv:2412.08111*, 2024.

[11] Sri Harsha Dumpala, Aman Jaiswal, Chandramouli Sastry, Evangelos Milios, Sageev Oore, and Hassan Sajjad. Sugarcrepe++ dataset: Vision-language model sensitivity to semantic and lexical alterations. *arXiv preprint arXiv:2406.11171*, 2024.

[12] Han Fang, Pengfei Xiong, Luhui Xu, and Yu Chen. Clip2video: Mastering video-text retrieval via image clip. *arXiv preprint arXiv:2106.11097*, 2021.

 https://doi.org/10.52202/085713-0468

[13] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

[14] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*, 2018.

[15] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *European Conference on Computer Vision*, pages 540–557. Springer, 2022.

[16] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

[17] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. *arXiv preprint arXiv:2104.13921*, 2021.

[18] Roei Herzig, Alon Mendelson, Leonid Karlinsky, Assaf Arbelle, Rogerio Feris, Trevor Darrell, and Amir Globerson. Incorporating structured representations into pretrained vision & language models using scene graphs. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14077–14098, Singapore, December 2023. Association for Computational Linguistics.

[19] Cheng-Yu Hsieh, Jieyu Zhang, Zixian Ma, Aniruddha Kembhavi, and Ranjay Krishna. Sugarcrepe: Fixing hackable benchmarks for vision-language compositionality. *Advances in neural information processing systems*, 36, 2024.

[20] Yufeng Huang, Jiji Tang, Zhuo Chen, Rongsheng Zhang, Xinfeng Zhang, Weijie Chen, Zeng Zhao, Zhou Zhao, Tangjie Lv, Zhipeng Hu, et al. Structure-clip: Towards scene graph knowledge to enhance multi-modal structured representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 2417–2425, 2024.

[21] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709, 2019.

[22] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[23] Amita Kamath, Jack Hessel, and Kai-Wei Chang. Text encoders bottleneck compositionality in contrastive vision-language models. *arXiv preprint arXiv:2305.14897*, 2023.

[24] Amita Kamath, Jack Hessel, and Kai-Wei Chang. What's "up" with vision-language models? investigating their struggle with spatial reasoning. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9161–9175, Singapore, December 2023. Association for Computational Linguistics.

[25] 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.

[26] Ryan Kiros, Yukun Zhu, Russ R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Skip-thought vectors. *Advances in neural information processing systems*, 28, 2015.

[27] 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. *International journal of computer vision*, 123:32–73, 2017.

[28] Benno Krojer, Vaibhav Adlakha, Vibhav Vineet, Yash Goyal, Edoardo Ponti, and Siva Reddy. Image retrieval from contextual descriptions. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3426–3440, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[29] Evelina Leivada, Elliot Murphy, and Gary Marcus. Dall· e 2 fails to reliably capture common syntactic processes. *Social Sciences & Humanities Open*, 8(1):100648, 2023.

[30] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and René Ranftl. Language-driven semantic segmentation. *arXiv preprint arXiv:2201.03546*, 2022.

[31] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR, 2022.

[32] Junyi Li, Tianyi Tang, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. Pre-trained language models for text generation: A survey. *ACM Computing Surveys*, 56(9):1–39, 2024.

[33] Linjie Li, Zhe Gan, and Jingjing Liu. A closer look at the robustness of vision-and-language pre-trained models. *arXiv preprint arXiv:2012.08673*, 2020.

[34] Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-vocabulary semantic segmentation with mask-adapted clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7061–7070, 2023.

[35] 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 *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.

[36] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. Clip4clip: An empirical study of clip for end to end video clip retrieval and captioning. *Neurocomputing*, 508:293–304, 2022.

[37] Zixian Ma, Jerry Hong, Mustafa Omer Gul, Mona Gandhi, Irena Gao, and Ranjay Krishna. Crepe: Can vision-language foundation models reason compositionally? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10910–10921, 2023.

[38] Youngtaek Oh, Jae Won Cho, Dong-Jin Kim, In So Kweon, and Junmo Kim. Preserving multi-modal capabilities of pre-trained vlms for improving vision-linguistic compositionality. *arXiv preprint arXiv:2410.05210*, 2024.

[39] Maya Okawa, Ekdeep S Lubana, Robert Dick, and Hidenori Tanaka. Compositional abilities emerge multiplicatively: Exploring diffusion models on a synthetic task. *Advances in Neural Information Processing Systems*, 36, 2024.

[40] Letitia Parcalabescu, Michele Cafagna, Lilitta Muradjan, Anette Frank, Iacer Calixto, and Albert Gatt. VALSE: A task-independent benchmark for vision and language models centered on linguistic phenomena. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8253–8280, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[41] Maitreya Patel, Abhiram Kusumba, Sheng Cheng, Changhoon Kim, Tejas Gokhale, Chitta Baral, and Yezhou Yang. Tripletclip: Improving compositional reasoning of clip via synthetic vision-language negatives. *Advances in neural information processing systems*, 2024.

[42] Wujian Peng, Sicheng Xie, Zuyao You, Shiyi Lan, and Zuxuan Wu. Synthesize diagnose and optimize: Towards fine-grained vision-language understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13279–13288, 2024.

 https://doi.org/10.52202/085713-0468

[43] Sarah Pratt, Mark Yatskar, Luca Weihs, Ali Farhadi, and Aniruddha Kembhavi. Grounded situation recognition. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*, pages 314–332. Springer, 2020.

[44] Alec Radford. Improving language understanding by generative pre-training, 2018.

[45] 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 *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[46] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

[47] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

[48] Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. Contrastive learning with hard negative samples. *arXiv preprint arXiv:2010.04592*, 2020.

[49] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[50] Ugur Sahin, Hang Li, Qadeer Khan, Daniel Cremers, and Volker Tresp. Enhancing multimodal compositional reasoning of visual language models with generative negative mining. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5563–5573, 2024.

[51] Sepehr Sameni, Kushal Kafle, Hao Tan, and Simon Jenni. Building vision-language models on solid foundations with masked distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14216–14226, 2024.

[52] Ravi Shekhar, Sandro Pezzelle, Yauhen Klimovich, Aurélie Herbelot, Moin Nabi, Enver Sangineto, and Raffaella Bernardi. Foil it! find one mismatch between image and language caption. *arXiv preprint arXiv:1705.01359*, 2017.

[53] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022.

[54] Harman Singh, Pengchuan Zhang, Qifan Wang, Mengjiao Wang, Wenhan Xiong, Jingfei Du, and Yu Chen. Coarse-to-fine contrastive learning in image-text-graph space for improved vision-language compositionality. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 869–893, Singapore, December 2023. Association for Computational Linguistics.

[55] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. Mass: Masked sequence to sequence pre-training for language generation. *arXiv preprint arXiv:1905.02450*, 2019.

[56] Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. Winoground: Probing vision and language models for visio-linguistic compositionality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5238–5248, 2022.

[57] Jiayu Wang, Yifei Ming, Zhenmei Shi, Vibhav Vineet, Xin Wang, Sharon Li, and Neel Joshi. Is a picture worth a thousand words? delving into spatial reasoning for vision language models. *Advances in Neural Information Processing Systems*, 37:75392–75421, 2025.

[58] Tan Wang, Kevin Lin, Linjie Li, Chung-Ching Lin, Zhengyuan Yang, Hanwang Zhang, Zicheng Liu, and Lijuan Wang. Equivariant similarity for vision-language foundation models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11998–12008, 2023.

[59] Thomas Wang, Adam Roberts, Daniel Hesslow, Teven Le Scao, Hyung Won Chung, Iz Beltagy, Julien Launay, and Colin Raffel. What language model architecture and pretraining objective works best for zero-shot generalization? In *International Conference on Machine Learning*, pages 22964–22984. PMLR, 2022.

[60] Yi Ru Wang, Jiafei Duan, Dieter Fox, and Siddhartha Srinivasa. Newton: Are large language models capable of physical reasoning? *arXiv preprint arXiv:2310.07018*, 2023.

[61] 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, October 2020. Association for Computational Linguistics.

[62] Shitao Xiao, Zheng Liu, Yingxia Shao, and Zhao Cao. Retromae: Pre-training retrieval-oriented language models via masked auto-encoder. *arXiv preprint arXiv:2205.12035*, 2022.

[63] 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.

[64] Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? In *International Conference on Learning Representations*, 2023.

[65] Arman Zarei, Keivan Rezaei, Samyadeep Basu, Mehrdad Saberi, Mazda Moayeri, Priyatham Kattakinda, and Soheil Feizi. Mitigating compositional issues in text-to-image generative models via enhanced text embeddings.

[66] Yunan Zeng, Yan Huang, Jinjin Zhang, Zequn Jie, Zhenhua Chai, and Liang Wang. Investigating compositional challenges in vision-language models for visual grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14141–14151, 2024.

[67] Jingyi Zhang, Jiaxing Huang, Sheng Jin, and Shijian Lu. Vision-language models for vision tasks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[68] Le Zhang, Rabiul Awal, and Aishwarya Agrawal. Contrasting intra-modal and ranking cross-modal hard negatives to enhance visio-linguistic compositional understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13774–13784, 2024.

[69] Tiancheng Zhao, Tianqi Zhang, Mingwei Zhu, Haozhan Shen, Kyusong Lee, Xiaopeng Lu, and Jianwei Yin. An explainable toolbox for evaluating pre-trained vision-language models. In Wanxiang Che and Ekaterina Shutova, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 30–37, Abu Dhabi, UAE, December 2022. Association for Computational Linguistics.

[70] Chenhao Zheng, Jieyu Zhang, Aniruddha Kembhavi, and Ranjay Krishna. Iterated learning improves compositionality in large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13785–13795, 2024.

[71] Yiwu Zhong, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, et al. Regionclip: Region-based language-image pretraining. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16793–16803, 2022.

[72] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *European Conference on Computer Vision*, pages 350–368. Springer, 2022.

 https://doi.org/10.52202/085713-0468

[73] Yuke Zhu, Oliver Groth, Michael Bernstein, and Li Fei-Fei. Visual7w: Grounded question answering in images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4995–5004, 2016.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes] .

   Justification: The claims made in the abstract and introduction accurately reflect the paper's contributions. The motivation (limitations of compositional reasoning in contrastive VLMs), method (READ with two auxiliary objectives for the text encoder), and results (consistent improvements across benchmarks and compatibility with other methods) are all clearly stated and supported by empirical evidence throughout the paper.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes] .

   Justification: The paper includes a dedicated Limitations section that discusses the scope of generalization with respect to training data (e.g., dependence on high-quality, multi-caption datasets), architectural choices in the reconstruction objective (e.g., reliance on a T5 decoder), and the absence of exploration into alternative decoder types or objectives. These limitations are clearly acknowledged and contextualized with respect to potential future directions.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover

 https://doi.org/10.52202/085713-0468

limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA] .

   Justification: This work focuses on empirical evaluation and does not contain theoretical assumptions or formal proofs.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes] .

   Justification: We provide all necessary details to reproduce our main experimental results, including architecture, training protocol, hyperparameters, data splits, and implementation specifics in Appendix A.1. All models are trained using publicly available datasets and baselines, and paraphrased captions are generated via an open API with specified parameters.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes] .

   Justification: We provide an anonymous open-access GitHub repository containing all necessary materials to reproduce the main experimental results, including code, model checkpoints, and data. The repository includes detailed instructions on environment setup, data, and exact commands to run each experiment. Scripts are provided to reproduce the result of READ-CLIP.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes] .

   Justification: All training and evaluation details—including dataset splits, optimizer, learning rate, warmup steps, batch size, weight decay, decoder choice, loss weights, and paraphrase generation method—are specified in the main text (Sec. 4.1) and Appendix A.1.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

https://doi.org/10.52202/085713-0468

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No] .

   Justification: None of the existing work, including NegCLIP, CE-CLIP, and FSC-CLIP, report statistical significance. To ensure fair comparison, we follow this convention and report single-run results on standard benchmarks. For training, we fix all random seeds to 2025. Evaluation is deterministic, and thus variability due to randomness does not arise.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes] .

   Justification: We report that all experiments can be conducted on a single NVIDIA A100 40GB GPU.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes] .

   Justification: Our research does not involve human subjects, personal data, or high-risk assets. All datasets and models used are publicly available, properly licensed, and fully credited. We have carefully reviewed the NeurIPS Code of Ethics and confirm that our work

adheres to all its principles regarding safety, fairness, environmental impact, data licensing, and reproducibility.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes] .

Justification: Our paper proposes a fine-tuning framework (READ) to improve compositional reasoning in vision-language models. This may lead to more accurate and interpretable VLMs. We do not foresee direct risks of misuse, as our model does not generate open-ended content or interface with end users.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [NA] .

Justification: The paper does not release any high-risk assets. All datasets used are publicly available and widely adopted compositional reasoning benchmarks with no known safety concerns. Our released model is a fine-tuned version of CLIP (ViT-B/32) on caption-based reasoning tasks, and is intended for academic use only. It does not have generative capabilities and poses minimal risk for misuse.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring

 https://doi.org/10.52202/085713-0468

that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes] .

    Justification: We use five existing datasets—CREPE-Productivity, SugarCrepe, WhatsUp, VALSE, and SugarCrepe++—all of which are publicly released under MIT licenses (with the exception of CREPE-Productivity, for which the license is unspecified). We cite the original papers introducing these benchmarks and describe their licenses and image sources in Appendix A.2 (see Table 7). All datasets were used in accordance with their published terms of use. No scraped or proprietary assets were used.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA] .

    Justification: The paper does not introduce any new assets. All datasets used are publicly available benchmarks, and our method is a fine-tuning approach applied to existing models. No new datasets or models are released.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA] .

Justification: This work does not involve any crowdsourcing or experiments with human participants. All data and evaluations were conducted using publicly available resources without human subject interaction.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA] .

Justification: This study does not involve research with human subjects. All experiments were conducted on publicly available datasets without any interaction with individuals or collection of personal data, and therefore IRB approval was not required.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA] .

Justification: LLMs were used solely for writing assistance (e.g., wording, formatting), and did not contribute to the design, implementation, or evaluation of the proposed methods.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

 https://doi.org/10.52202/085713-0468

```
<task>
   Rewrite the following image description:
</task>
**Input**:
{caption}
**output**:
```

Figure 6: A prompt for generating synthetic paraphrased captions.

Benchmark	License	Image Source
CREPE-Productivity [37]	Unspecified	Visual Genome [27]
SugarCrepe [19]	MIT	COCO [35]
SugarCrepe++ [11]	MIT	COCO [35]
WhatsUp [24]	MIT	Custom-collected, COCO [35], GQA [21]
VALSE [40]	MIT	Visual7W [73], COCO [35], SWiG [43], FOIL-it [52]

Table 7: Benchmarks used in the evaluation, along with license and image source.

A Appendix

A.1 Implementation Details

Batch Sampling for Training. As described in Sec. 4.1, all our experiments are conducted using the COCO [35] dataset. The COCO dataset is an image-caption dataset consisting of images, each associated with a set of captions. During training, we first sample a batch of B images from this dataset. For each $i \in \{1, 2, \ldots, B\}$, let $\mathcal{T}_i = \{T_i^{(n)}\}_{n=1}^{N}$ denote the set of N captions associated with I_i (where I_i denotes the i-th image). From this set, we randomly select one caption $T_i \in \mathcal{T}_i$ to form a positive image-text pair (I_i, T_i). In this way, we obtain a batch of B image-text pairs, denoted as $\{(I_i, T_i)\}_{i=1}^{B}$.

Details of Each Loss Component. Recall that our proposed *REconstruction and Alignment of text Descriptions* (READ) method comprises three components in the final loss (Eq. 8): the standard *contrastive* loss, the token-level *reconstruction* loss, and the sentence-level *alignment* loss. For the *contrastive* loss (Eq. 4), we incorporate $M = 3$ hard negative captions per image, generated via rule-based perturbations as proposed in NegCLIP [64]. For the *reconstruction* loss (Eq. 5), we use a frozen decoder extracted from the pre-trained encoder-decoder language model, `T5-Large` [46]. To obtain $\{y_i^{(k)}\}_{k=1}^{K}$, we randomly sample $K = 1$ element from the caption set $\mathcal{T}_i$ associated with each image-text pair (I_i, T_i) and use it as an alternative caption. For the *alignment* loss (Eq. 7), we generate a paraphrased caption T_i' for each T_i via augmentation using large language models. Specifically, prior to training, we generate one paraphrased caption for every caption in each image's original caption set using the `gpt-4o-mini-2024-07-18` model [22]. This is done by applying a simple prompt as shown in Fig.6, with a temperature of 1.0 and all other parameters set to their default values [22]. From this augmentation, we obtain a synthetic caption set for each image. Given a batch of sampled image-text pairs $\{(I_i, T_i)\}_{i=1}^{B}$ during training, we randomly sample one caption from the union of the original and synthetic caption sets associated to I_i, and use it as T_i'. Finally, the weighting factors in Eq. 8 are set to $\alpha = 0.1$ and $\beta = 0.5$.

Training Details. We fine-tune all models using the Huggingface `transformers` [61] library[3]. The AdamW optimizer is used with a learning rate of 1.0×10^{-5}, cosine annealing schedule, 50 warmup steps, and a weight decay of 0.1. Training is performed with `bf16` mixed precision for computational efficiency. All experiments are conducted using a single A100 40GB GPU.

A.2 Evaluation Details

Description of Benchmarks. WhatsUp [24] evaluates spatial reasoning by testing whether models can interpret relative object positions. **CREPE** [37] measures compositional reasoning at varying

[3] `https://github.com/huggingface/transformers`

Table 8: Detailed results on CREPE [37].

Model	Atom	Negate	Swap	Total
CLIP [45] (Pre-trained)	18.9	35.3	17.3	23.9
Triplet-CLIP [41]	18.5	11.2	15.3	15.0
GNM-CLIP [50]	21.7	13.3	17.3	17.4
CE-CLIP [68]	40.4	21.2	42.7	34.8
NegCLIP [64]	32.1	16.6	42.8	30.5
FSC-CLIP [38]	**40.5**	**41.3**	45.5	**42.5**
READ-CLIP	37.1	26.2	**61.2**	41.5

Table 9: Detailed results on WHATSUP [24].

Model	COCO-Spatial	GQA-Spatial	Whats-up	Total
CLIP [45] (Pre-trained)	44.5	47.8	30.7	41.0
Triplet-CLIP [41]	49.2	47.1	28.5	41.6
GNM-CLIP [50]	44.8	47.4	32.6	41.6
CE-CLIP [68]	43.7	47.8	30.7	40.7
NegCLIP [64]	45.1	47.7	34.4	42.4
FSC-CLIP [38]	47.7	41.9	29.6	39.8
READ-CLIP	**51.6**	**48.1**	**31.8**	**43.9**

complexity levels using logical operations such as conjunction, negation, and attribute swapping. **VALSE** [40] assesses fine-grained linguistic understanding, including object existence, quantity, action semantics, and coreference resolution. **SugarCrepe** [19] focuses on relational reasoning through hard negative captions crafted with natural linguistic variation. **SugarCrepe++** [11] extends SugarCrepe by adding a paraphrased positive caption and introduces two tasks: (1) image-to-text (ITT), which tests whether all paraphrased positives for a given image are ranked above all negatives, and (2) text-to-text (TOT), which evaluates semantic consistency by checking whether each positive paraphrase pair is ranked above all negative pairs in the absence of visual context. Since our study aims to improve the compositional reasoning capability of VLMs such as CLIP, we primarily adopt the ITT metric as a major focus for evaluation, while including TOT as a supplementary measure.

Licensing of the Benchmarks. We conduct our evaluation on five publicly available compositional reasoning benchmarks. Table 7 summarizes their license information and image sources. All datasets used for training and evaluation are either MIT-licensed or publicly released for research use.

A.3 Supplementary Experimental Results

This section provides extended experimental results that complement the main paper. We include both (1) detailed, category-wise results for each benchmark and (2) additional evaluations on zero-shot image classification datasets [5] to further assess the generalization ability of READ-CLIP.

A.3.1 Detailed Version of Experimental Results in the Main Paper

To complement the results presented in Table 1, we report category-wise results on all evaluation benchmarks, including CREPE, VALSE, WhatsUp, SugarCrepe, and SugarCrepe++ (ITT and TOT). Table 8–13 present detailed breakdowns for individual benchmarks. These results provide a finer-grained analysis of compositional reasoning performance supplementing the aggregated scores shown in the main paper. Overall, the detailed results confirm that READ-CLIP consistently improves over baselines across categories. In addition, we provide Fig. 7 and Fig. 8, which present extended qualitative examples that respectively complement Fig. 3 and Fig. 4 in the main paper.

https://doi.org/10.52202/085713-0468

Table 10: Detailed results on VALSE [40].

Model	Actions	Coreference	Counting	Existence	Noun Phrases	Plurals	Relations	**Total**
CLIP [45] (Pre-trained)	74.3	54.4	61.7	69.3	90.4	57.9	66.0	67.4
Triplet-CLIP [41]	72.6	54.8	54.0	59.4	91.4	61.3	60.9	64.2
GNM-CLIP [50]	72.1	**61.1**	66.5	76.0	90.8	68.7	62.6	70.7
CE-CLIP [68]	85.0	59.9	67.6	78.2	94.4	**78.8**	74.0	76.0
NegCLIP [64]	84.1	60.2	65.7	75.3	93.4	70.3	69.6	73.7
FSC-CLIP [38]	82.9	59.4	66.3	77.6	93.5	72.7	**75.3**	74.4
READ-CLIP	**86.3**	55.7	**69.0**	**80.8**	**95.8**	73.8	75.0	**76.2**

Table 11: Detailed results on SugarCrepe [19]. `Att.`, `Obj.`, and `Rel.` denote the targets of transformation: `Attribute`, `Object`, and `Relation`, respectively.

Model	Add Att.	Add Obj.	Replace Att.	Replace Obj.	Replace Rel.	Swap Att.	Swap Obj.	**Total**
CLIP [45] (Pre-trained)	69.5	77.0	80.3	90.7	69.4	64.1	61.2	73.2
Triplet-CLIP [41]	85.5	87.5	86.7	94.5	**83.2**	73.1	68.6	82.7
GNM-CLIP [50]	79.9	88.4	84.9	93.2	67.8	70.0	61.2	77.9
CE-CLIP [68]	**91.9**	**92.3**	90.2	94.4	81.4	76.7	75.1	86.0
NegCLIP [64]	85.3	90.0	88.2	94.0	74.6	77.9	75.5	83.6
FSC-CLIP [38]	86.7	90.2	89.2	94.3	80.4	77.8	77.6	85.2
READ-CLIP	87.7	90.3	**91.0**	**94.9**	80.6	**82.7**	**81.6**	**87.0**

Table 12: Detailed results on *image-to-text* subset of SugarCrepe++ [11]. `Att.`, `Obj.`, and `Rel.` denote the targets of transformation: `Attribute`, `Object`, and `Relation`, respectively.

Model	Replace Att.	Replace Obj.	Replace Rel.	Swap Att.	Swap Obj.	**Total**
CLIP [45] (Pre-trained)	65.7	87.0	56.5	45.0	45.8	60.0
Triplet-CLIP [41]	71.7	87.0	**62.3**	48.5	39.2	61.7
GNM-CLIP [50]	68.9	89.5	52.8	48.6	41.2	60.2
CE-CLIP [68]	62.4	81.9	53.5	40.5	40.0	55.7
NegCLIP [64]	69.7	89.8	52.6	58.1	54.7	65.0
FSC-CLIP [38]	73.5	**90.4**	60.1	60.4	55.1	67.9
READ-CLIP	**72.2**	90.1	57.5	**66.2**	**62.9**	**69.8**

Table 13: Detailed results on *text-to-text* subset of SugarCrepe++ [11]. `Att.`, `Obj.`, and `Rel.` denote the targets of transformation: `Attribute`, `Object`, and `Relation`, respectively.

Model	Replace Att.	Replace Obj.	Replace Rel.	Swap Att.	Swap Obj.	**Total**
CLIP [45] (Pre-trained)	59.3	83.7	38.6	32.7	19.2	46.7
Triplet-CLIP [41]	74.1	92.3	52.3	43.2	24.9	57.4
GNM-CLIP [50]	76.9	95.9	51.9	48.9	26.1	60.0
CE-CLIP [68]	74.2	89.6	52.0	42.8	26.5	57.0
NegCLIP [64]	76.4	94.6	51.7	56.6	33.1	62.5
FSC-CLIP [38]	**83.5**	96.3	56.8	56.3	29.0	64.4
READ-CLIP	77.3	**97.6**	**58.0**	**56.8**	**41.2**	**66.2**

A.3.2 Additional Experimental Results

To further verify the generalization capability of READ-CLIP, we conducted an additional evaluation beyond the main benchmarks. We assessed the model's zero-shot image classification performance across 23 widely used benchmarks [5]. We compared READ-CLIP with the original CLIP [45] pre-trained model and six representative compositional reasoning fine-tuning methods.

In Table 14, The results show that all fine-tuned models, including READ-CLIP, achieve lower average performance than the original CLIP across the 23 datasets. This finding aligns with trends reported in previous study [38], where improvements in compositional understanding often come at the cost of general zero-shot capability. Such trade-offs highlight the inherent difficulty of maintaining broad generalization while adapting models specifically for compositional reasoning.

Figure 7: Extended representative examples for Fig. 3, including additional examples from CREPE [37] and VALSE [40], as well as WhatsUp [24] and SugarCrepe [19]. These extended examples additionally include a broader range of benchmarks where applying the *reconstruction loss* proved effective.

Table 14: Additional results of zero-shot image classification performance across various datasets. Here, CLIP [45] is the pre-trained model, while the other models are fine-tuned versions of CLIP [45].

Model	caltech101	cars	cifar10	cifar100	country211	dtd	eurosat	fer2013	fgvc-aircraft-2013b	flowers	food101	gtsrb	imagenet-o	imagenet-1k	imagenet-sketch	imagenet-v2	kitti-distance	mnist	pcam	rendered-sst2	resisc45-clip	stl10	voc2007classification	Avg.
CLIP [45]	81.5	**59.7**	89.8	64.3	**17.2**	**44.3**	50.5	41.2	**19.7**	**66.4**	**84.0**	**32.5**	**47.6**	**63.4**	**42.3**	**55.7**	27.1	48.3	**62.3**	58.8	53.6	**97.1**	76.5	**55.8**
NegCLIP [64]	**82.6**	53.9	88.9	63.0	15.0	43.0	49.7	46.7	16.8	65.0	79.4	30.2	46.5	60.9	40.4	53.2	27.7	49.7	54.9	58.6	52.9	96.7	**79.6**	54.6
GNM-CLIP [50]	81.5	53.1	88.5	**65.0**	15.2	42.1	50.7	46.0	17.2	63.3	81.8	30.2	47.4	61.4	41.0	54.1	25.3	**54.3**	55.6	58.5	49.8	96.4	77.4	54.6
FSC-CLIP [38]	81.8	51.8	89.1	64.9	14.5	40.7	51.6	49.5	15.8	61.7	78.7	29.8	45.5	59.2	38.9	51.7	29.4	50.4	51.0	**59.8**	52.8	96.1	79.0	54.1
DAC-LLM [6]	77.7	39.4	**90.4**	63.9	14.3	39.0	**52.3**	**50.5**	11.3	54.6	74.2	24.2	45.5	51.0	35.2	45.0	16.6	42.2	50.0	54.4	49.6	**97.1**	77.9	50.3
DAC-SAM [6]	75.7	39.9	89.9	63.7	14.8	40.0	51.2	47.7	9.0	53.9	72.3	24.9	45.5	52.4	35.1	46.8	18.7	45.3	50.0	59.8	51.7	96.1	65.8	50.0
READ-CLIP	78.2	39.6	87.1	57.8	10.2	35.0	39.2	41.0	13.1	52.2	71.6	26.7	44.5	51.5	32.9	45.3	30.5	48.0	47.3	52.3	44.3	95.2	78.9	48.8
CE-CLIP [68]	78.3	35.3	85.9	60.1	9.5	35.2	42.8	39.5	10.0	48.2	70.1	28.0	44.8	49.9	31.5	43.2	**34.6**	40.6	50.0	61.2	47.7	95.8	77.3	48.7
Triplet-CLIP [41]	80.6	23.9	89.1	61.5	7.1	39.3	35.2	47.7	12.7	54.6	76.3	24.7	42.8	54.8	37.0	48.4	15.3	34.3	49.6	51.8	**54.7**	94.6	72.9	48.2

https://doi.org/10.52202/085713-0468

Figure 8: Extended representative examples for Fig. 4, including additional examples each category (SWAP and REPLACE) of SugarCrepe++. These extended examples further illustrate the effectiveness of applying the *alignment loss* across diverse cases.

https://doi.org/10.52202/085713-0468