Fine-tuning CLIP Text Encoders with Two-step Paraphrasing

Hyunjae Kim[1] **Seunghyun Yoon**[2] **Trung Bui**[2]
Handong Zhao[2] **Quan Tran**[2] **Franck Dernoncourt**[2] **Jaewoo Kang**[1]
[1]Korea University [2]Adobe Research
{hyunjae-kim,kangj}@korea.ac.kr
{syoon,bui,hazhao,qtran,dernonco}@adobe.com

Abstract

Contrastive language-image pre-training (CLIP) models have demonstrated considerable success across various vision-language tasks, such as text-to-image retrieval, where the model is required to effectively process natural language input to produce an accurate visual output. However, current models still face limitations in dealing with linguistic variations in input queries, such as paraphrases, making it challenging to handle a broad range of user queries in real-world applications. In this study, we introduce a straightforward fine-tuning approach to enhance the representations of CLIP models for paraphrases. Our approach involves a two-step paraphrase generation process, where we automatically create two categories of paraphrases from web-scale image captions by leveraging large language models. Subsequently, we fine-tune the CLIP text encoder using these generated paraphrases while freezing the image encoder. Our resulting model, which we call ParaCLIP, exhibits significant improvements over baseline CLIP models across various tasks, including paraphrased retrieval (with rank similarity scores improved by up to 2.0% and 5.6%), Visual Genome Relation and Attribution, as well as seven semantic textual similarity tasks.

1 Introduction

Contrastive language-image pre-training (CLIP) models (Radford et al., 2021) have gained significant attention in the fields of computer vision and natural language processing for their remarkable capacity to understand the relationship between text and images. They have been widely used in various vision-language applications, including image classification (Deng et al., 2009), image retrieval (Lin et al., 2014; Plummer et al., 2015), and text-to-image generation (Saharia et al., 2022; Rombach et al., 2022), where the model should return desired visual outputs for a given text, and vice versa.

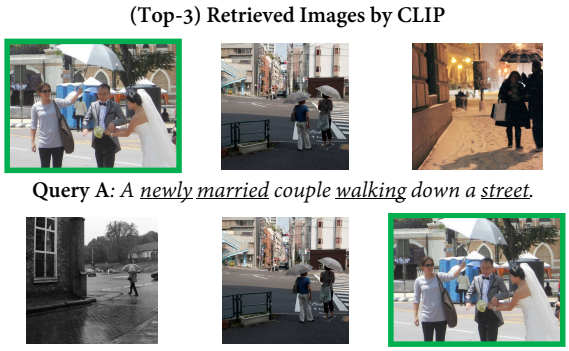

Figure 1: Image retrieval results of CLIP (Radford et al., 2021) for two different queries (the gold image is denoted by a bold border). Despite their comparable meanings, the model yields dissimilar retrieval results, highlighting the model's struggle with linguistic variations.

An inherent challenge in vision-language tasks lies in the variability of text inputs. Even when conveying similar meanings and intentions, they can exhibit variations in vocabulary and structure depending on the particular user. Consequently, it becomes crucial to ensure that CLIP's text encoders are robust enough to handle diverse synonyms and paraphrases in practical scenarios. However, current text encoders exhibit limited proficiency in comprehending linguistic variations, resulting in different retrieval results for user queries with similar meanings (Figure 1).

To address this challenge, we introduce a straightforward method to improve CLIP's text encoders. Specifically, we generated two categories of paraphrases for image captions sourced from the web, leveraging recent large language models (LLM) such as ChatGPT (OpenAI, 2022) and LLaMA (Touvron et al., 2023). Subsequently, we utilized image captions and their corresponding paraphrases to fine-tune the text encoder, which ensures that the representations of captions and paraphrases cluster in a similar vector space.

We validated the effectiveness of our approach

Findings of the Association for Computational Linguistics: EACL 2024, pages 2175–2184

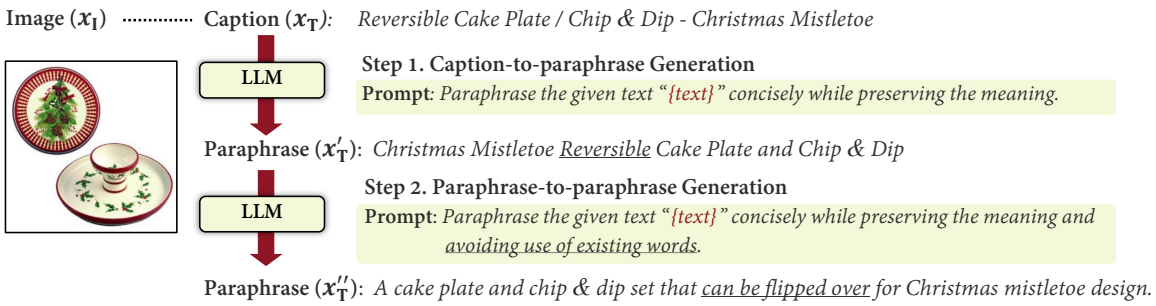

Figure 2: Overview of our two-step paraphrasing process. (1) In caption-to-paraphrase generation, the first paraphrase is generated by removing noise from the original caption and converting it into a more plain language. (2) In paraphrase-to-paraphrase generation, the second paraphrase is generated from the first paraphrase, where the word "reversible" is changed to a semantically similar expression "can be flipped over."

using evaluation tasks that assess models' understanding of language semantics and composition: paraphrased retrieval, Visual Genome Relation (VG-R), Visual Genome Attribution (VG-A) (Yuksekgonul et al., 2023), and semantic textual similarity (STS) tasks (Agirre et al., 2012). Our models, ParaCLIP, significantly outperformed baseline CLIP models, while maintaining or sometimes improving its robust performance on zero-shot image classification (Deng et al., 2009), as well as text and image retrieval (Lin et al., 2014). We emphasize that this is the first study to improve the representations of CLIP's text encoders during the fine-tuning stage using synthetic paraphrases.

2 Method

Our objective is to refine the CLIP model's training process, enabling its text encoder to produce consistent representations for various semantically similar textual inputs that the model might encounter in real-world scenarios. Certain image-captioning datasets provide multiple captions for a single image (Lin et al., 2014; Plummer et al., 2015), which might be utilized as semantically similar text pairs during training. However, the volume of these datasets is limited, which presents a challenge in terms of exposing models to diverse language patterns. Therefore, we automatically generated semantically similar pairs (i.e., paraphrases) for millions of image captions sourced from the web.

2.1 Paraphrase Generation

An image-captioning dataset typically comprises a collection of image-caption pairs $(x_\mathrm{I}, x_\mathrm{T})$, where x_I and x_T represent an image and the corresponding caption, respectively. For each caption x_T, we created two categories of paraphrases through a two-step paraphrasing process, caption-to-paraphrase generation and paraphrase-to-paraphrase generation, as illustrated in Figure 2.

Caption-to-paraphrase generation This process directly rewrites original captions. Image captions on the web often contain considerable noise, such as superfluous punctuation, product codes, and file extensions, which differ from typical queries. This step can be seen as responsible for converting these noisy captions into a more straightforward text format commonly used in everyday language. Using the power of LLMs, we synthesized paraphrases x'_T for each caption with the following prompt: *"Paraphrase the given caption "*`text`*" concisely while preserving the meaning."*, where `text` is substituted with a given caption.

Paraphrase-to-paraphrase generation In this step, additional paraphrases, x''_T, are generated for each generated paraphrase, x'_T. The paraphrasing process is similar to the previous step, but with some differences in the prompt as follows: *"Paraphrase the given text "*`text`*" concisely while preserving the meaning and avoiding use of existing words."*, where the underlined text is used to prompt the model to produce morphologically diverse expressions.

2.2 Training Objectives

Let $\mathbf{X}_\mathrm{I}$, $\mathbf{X}_\mathrm{T}$, $\mathbf{X}'_\mathrm{T}$, and $\mathbf{X}''_\mathrm{T}$ be mini-batches of N examples of an image x_I, caption x_T, and two types of paraphrases, x'_T and x''_T. The final loss is calculated as the summation of three sub-losses as follows: $\mathcal{L}_\mathrm{total} := \mathcal{L}_1(\mathbf{X}_\mathrm{I}, \mathbf{X}''_\mathrm{T}) + \mathcal{L}_2(\mathbf{X}_\mathrm{T}, \mathbf{X}'_\mathrm{T}) + \mathcal{L}_3(\mathbf{X}'_\mathrm{T}, \mathbf{X}''_\mathrm{T})$. The first term, $\mathcal{L}_1$, represents the InfoNCE loss function that operates between images and text (Oord et al., 2018). This loss function is

crucial in the prevention of forgetting CLIP's representations and knowledge acquired during pre-training. We used the paraphrased version of text input $\mathbf{X}''_{\mathrm{T}}$ rather than the original captions $\mathbf{X}_{\mathrm{T}}$ because user queries often resemble plain text rather than the original captions. This choice led to improved performance on the benchmark datasets during our preliminary experiment. If the target domain involves dealing with noisy text inputs, such as in an online shopping mall context, employing the original captions may be more effective.

The second term, $\mathcal{L}_2$, accounts for the relationship between captions and their paraphrases. Conceptually, it serves to establish a connection within the vector space between the representation of noisy captions and the plain text commonly used in everyday language. Lastly, $\mathcal{L}_3$ serves to bring together various semantically similar plain texts within a vector space. For $\mathcal{L}_2$ and $\mathcal{L}_3$, we used the InfoNCE loss. The resulting CLIP model fine-tuned using these three losses is called ParaCLIP.

3 Experimental Setups

We obtained image-caption pairs using LAION-400M (Schuhmann et al., 2021). We initially generated 300K paraphrases using ChatGPT and instruction-tuned an open-sourced LLM named LLaMA (7B) (Touvron et al., 2023) using these 300K data to generate additional paraphrases.[1] Our final dataset comprises 5M examples of x_{I}, x_{T}, x'_{T}, and x''_{T}. More details and hyperparameters are described in Appendix A.

3.1 Baseline Models

We used the following CLIP models as baseline models, all built upon the ViT-B/32 architecture (Dosovitskiy et al., 2021). (1) OpenAI's CLIP (Radford et al., 2021) was trained using a private dataset comprising 400M image-text pairs sourced from the web. (2) OpenCLIP models (Cherti et al., 2023) were trained using the largest open-sourced datasets, LAION-400M and LAION-2B (Schuhmann et al., 2022). (3) OpenCLIP-RoBERTa was pre-trained using LAION-2B. In contrast to the usual practice where text encoders are initialized with random weights and subsequently trained from scratch, its text encoder was initialized with the weights of RoBERTa-base (Liu et al., 2019) for better linguistic comprehension capabilities. (4) LaCLIP (Fan et al., 2023) was pre-trained using the LAION-400m dataset augmented with automatically generated paraphrases.[2] Specifically, a small number of original caption and paraphrase pairs were obtained from COCO text descriptions, or created by ChatGPT, Google BARD, and humans. These seed examples were used to prompt an LLaMA 7B model through a in-context learning approach, which then generated paraphrases for the entire LAION-400m dataset. During pre-training, a standard InfoNCE loss was computed using these paraphrases and corresponding images in combination with original caption and image pairs. While our method shares some similarities with LaCLIP in the use of model-generated paraphrases, it should be noted that ours has unique advantages. First, we enhance CLIP models through fine-tuning the text encoders while freezing the image encoders, which is significantly more efficient compared to pre-training the entire model from scratch. Despite its efficiency, our method is significantly more effective to improve the CLIP's robustness to paraphrases, improving the performance in paraphrased retrieval by a large margin (see Section 4 for details).

3.2 Evaluation

We evaluated models on the following tasks in a zero-shot manner, without fine-tuning them on the target tasks. (1) Paraphrased retrieval (Cheng et al., 2024) involves retrieving identical images for both 4,155 original queries and their corresponding paraphrases from the image set of the COCO 2017 validation set (Lin et al., 2014). Paraphrases were generated using GPT-3 (Brown et al., 2020) and subsequently verified by humans. This task is well-suited for assessing models' ability to effectively handle user queries expressed in diverse forms. For metrics, we used the top-10 average overlap (AO@10) and Jaccard similarity (JS@10) scores, which measure the degree of rank similarity between the top 10 images retrieved for the original query and paraphrased query. Detailed descriptions of the metrics can be found in Appendix B.

(2) VG-R and (3) VG-A (Yuksekgonul et al., 2023) are devised to assess relational and attributive understanding of vision-language models, respectively. They involve determining the correct

[1] We verified that the data generated by LLaMA exhibited comparable quality to that of ChatGPT. Additionally, when training the model using 300K paraphrases from LLaMA and an additional 300K paraphrases from ChatGPT, respectively, we observed similar performance in both cases.

[2] https://github.com/LijieFan/LaCLIP

Model	Paraphrased Rtrv.		VG-R	VG-A	STS	Clsf.	T Rtrv.	I Rtrv.
	AO@10	JS@10	Acc	Acc	Avg.	Acc	R@5	R@5
OpenAI's CLIP (400M)	67.2	57.7	59.7	63.2	65.1	63.4	75.0	54.8
+ ParaCLIP	**72.2**	**63.3**	**60.7**	**64.3**	**72.2**	**63.5**	**77.0**	**58.8**
OpenCLIP (400M)	67.6	58.9	46.4	57.8	67.2	60.2	**76.5**	**59.4**
+ ParaCLIP	**71.3**	**62.9**	**55.4**	**61.7**	**70.1**	**60.8**	76.1	**59.4**
OpenCLIP (2B)	70.6	62.1	45.0	61.8	69.6	**66.5**	80.2	**64.8**
+ ParaCLIP	**73.2**	**65.1**	**58.8**	**65.4**	**71.6**	65.5	**80.4**	63.3
OpenCLIP-RoBERTa (2B)	72.5	64.0	35.6	64.5	71.0	**61.8**	78.8	**62.6**
+ ParaCLIP	**74.5**	**66.2**	**43.2**	**66.5**	**72.5**	61.4	**79.4**	62.0
LaCLIP (400M)	69.9	62.1	50.6	63.6	58.8	**64.5**	68.1	55.5
+ ParaCLIP	**73.5**	**65.8**	**60.6**	**64.6**	**71.4**	**64.5**	**73.6**	**58.0**

Table 1: Zero-shot performance of baseline CLIP models and our ParaCLIP models. The best scores are represented in bold. "Acc": Accuracy. "Avg.": Macro average of Spearman's rank correlations across all STS tasks. "Clsf.": Image classification. "T Rtrv.": Text retrieval. "I Rtrv.": Image retrieval.

caption for a given image from two candidate captions, where negative captions are generated by interchanging objects based on their relational context or interchanging attributes of objects. For instance, given the correct caption "the *dog* is behind the *tree*," a negative counterpart could be formulated as follows: "the *tree* is behind the *dog*." The VG-R and VG-A datasets comprise 23,937 and 28,748 test examples, respectively.

(4) STS has been widely employed to evaluate the text representations of encoders (Conneau et al., 2017; Reimers and Gurevych, 2019; Chuang et al., 2022). This task involves measuring semantic similarity or relatedness between pairs of text. Following Gao et al. (2021), we measured Spearman's correlation for each task in the "all" aggregation setting and reported macro-averaged scores across the seven STS tasks (Agirre et al., 2012, 2013, 2014, 2015, 2016; Cer et al., 2017; Marelli et al., 2014).

Additionally, we assessed whether our models can maintain or even improve their performance on standard vision or vision-language tasks after being fine-tuned, including zero-shot image classification on the ImageNet-1K validation set (Deng et al., 2009), and image-to-text retrieval and text-to-image retrieval on the COCO validation set (Lin et al., 2014). For metrics, top-1 accuracy (Acc) and top-5 recall (R@5) were used in the classification and retrieval tasks, respectively.

4 Results and Discussion

4.1 Main Results

Table 1 shows the zero-shot performance of the baseline and our models in the evaluation tasks.

Effect of fine-tuning using paraphrases Across all CLIP models, our approach consistently demonstrated improved performance in the four primary tasks. Notably, the most significant improvements were observed in the paraphrased retrieval task, where our ParaCLIP model achieved 72.2% and 63.3% in AO@10 and JS@10 scores, increasing the performance of OpenAI's CLIP by 5.0% and 5.6%, respectively.[3] The improvements in the STS tasks are also noticeable, with the macro-average score improving by 7.1%. Although not in all cases, our approach generally enhances performance in the text retrieval task. This is attributed to our model's capability to encode texts that shares semantic similarity with a given input image closely within the vector space.

Effect of initialization with RoBERTa The OpenCLIP-RoBERTa model significantly outperformed the OpenCLIP (2B) model in paraphrased retrieval and STS, highlighting the benefits of leveraging pre-trained language models over randomly initialized text encoders. However, even with these advancements, there is substantial room for improvement in performance on these tasks. Our fine-tuning approach further refined the RoBERTa text encoder, leading to notable achievements across the four primary tasks, with 2.0% (AO@10) and 2.2% (JS@10) scores in paraphrased retrieval.

Comparison with LaCLIP While LaCLIP exhibited superior performance compared to the OpenCLIP (400M) model in image classification, paraphrased retrieval, VG-R, and VG-A, its per-

[3] A case study comparing CLIP and ParaCLIP in the paraphrased retrieval task can be found in Appendix C.

Model	Paraphrased Rtrv.		VG-R	VG-A	STS	Clsf.	T Rtrv.	I Rtrv.
	AO@10	JS@10	Acc	Acc	Avg.	Acc	R@5	R@5
OpenAI's CLIP (400M)	67.2	57.7	59.7	63.2	65.1	63.4	75.0	54.8
+ $\mathcal{L}_1$	68.9	59.9	58.0	62.4	68.7	63.7	75.8	58.0
+ $\mathcal{L}_2 + \mathcal{L}_3$	70.5	61.2	**61.5**	**65.1**	**74.5**	56.7	74.6	51.8
+ $\mathcal{L}_1 + \mathcal{L}'_1$	70.4	61.7	58.2	63.0	69.1	64.0	76.3	58.7
+ $\mathcal{L}_1 + \mathcal{L}'_1 + \mathcal{L}''_1$	71.3	62.8	58.9	63.4	68.8	**64.1**	76.4	**58.8**
+ $\mathcal{L}_1 + \mathcal{L}_2$	69.1	60.0	59.1	63.3	71.8	63.5	76.1	58.2
+ $\mathcal{L}'_1 + \mathcal{L}_2$	70.8	62.0	60.4	64.0	71.6	63.7	76.4	58.6
+ $\mathcal{L}_1 + \mathcal{L}_2 + \mathcal{L}_3$	69.6	60.5	59.2	63.4	72.4	63.1	76.4	58.1
+ $\mathcal{L}''_1 + \mathcal{L}_2 + \mathcal{L}_3$ (**Ours**)	**72.2**	**63.3**	60.4	64.2	72.2	63.5	**77.0**	**58.8**

Table 2: Zero-shot performance of OpenAI's CLIP (400M) with different loss functions applied. The best scores are represented in bold and the second best scores are underlined. "Paraphrased Rtrv.": Paraphrased retrieval. "Acc": Accuracy. "Avg.": Macro average of Spearman's rank correlations across all STS tasks. "Clsf.": Image classification. "T Rtrv.": Text retrieval. "I Rtrv.": Image retrieval.

formance in the text/image retrieval and STS tasks witnessed a decline. This indicates that augmenting paraphrased text data may not consistently yield improvements, without incorporating effective loss functions such as $\mathcal{L}_2$ and $\mathcal{L}_3$. Conversely, our fine-tuning method dramatically enhanced LaCLIP's performance in paraphrased retrieval (+ 3.6% in AO@10 and 3.7% in JS@10), VG-R (+ 10.0%), VG-A (+ 1.0%), STS (+ 12.6%), and even on text retrieval (+ 5.5%) and image retrieval (+ 2.5%), highlighting that our method can complement LaCLIP to achieve optimal performance.

Lack of compositional understanding All CLIP models exhibited significant deficiencies in the VG-R and VG-A tasks. These limitations in compositional understanding can lead to errors in downstream tasks such as text-to-image synthesis, including unintentional attribute interchanges or the omission of objects in generated images (Feng et al., 2023). In future research, we plan to conduct a more in-depth analysis to explore the potential of our approach to mitigate these issues.

4.2 Ablation Study

We conducted an ablation study to closely examine the individual contributions of each loss term (Table 2). In this section, we simplify the notation $\mathcal{L}_1(\mathbf{X}_\mathrm{I}, \mathbf{X}_\mathrm{T})$, $\mathcal{L}_1(\mathbf{X}_\mathrm{I}, \mathbf{X}'_\mathrm{T})$, and $\mathcal{L}_1(\mathbf{X}_\mathrm{I}, \mathbf{X}''_\mathrm{T})$ to $\mathcal{L}_1$, $\mathcal{L}'_1$, and $\mathcal{L}''_1$, respectively. Note that our ParaCLIP model was trained using the combined loss functions, $\mathcal{L}''_1+\mathcal{L}_2+\mathcal{L}_3$, as detailed in Section 2.2.

First, we fine-tuned the OpenAI's CLIP model using the same set of image-caption pairs in LAION-400M as our model, excluding paraphrases (referred to as "$\mathcal{L}_1$"). While there was an overall improvement in performance, it still fell short of our ParaCLIP model's performance. When $\mathcal{L}''_1$ was omitted (i.e., $\mathcal{L}_2 + \mathcal{L}_3$), the model showed the best performance on the VG-R, VG-A, and STS tasks, but the performance on image classification and standard text and image retrieval significantly degraded. This indicates that $\mathcal{L}''_1$ was crucial in preserving the representations of CLIP acquired during pre-training. Although simply augmenting training data with synthetic paraphrases (i.e., $\mathcal{L}_1 + \mathcal{L}'_1$ and $\mathcal{L}_1 + \mathcal{L}'_1 + \mathcal{L}''_1$) generally led to performance improvements, the improvements in the STS tasks were not substantial compared to the models with the $\mathcal{L}_2$ and $\mathcal{L}_3$ losses. Applying $\mathcal{L}_3$ was particularly effective for STS because it involved comparing pairs of semantically similar "plain" text (not pairs of noisy caption and plain text), which aligns well with the goal of STS. Finally, our ParaCLIP model, incorporating three losses (i.e., $\mathcal{L}''_1 + \mathcal{L}_2 + \mathcal{L}_3$), showed the most balanced performance across all tasks among the various models evaluated. In particular, applying $\mathcal{L}''_1$ instead of $\mathcal{L}_1$ proved to be generally effective.

5 Conclusion

In this study, we proposed a two-step paraphrasing approach for enhancing the representations of CLIP for paraphrases that may occur in text inputs in real-world applications. Our ParaCLIP models, fine-tuned using synthetic paraphrases, outperformed baseline models by a large margin on various tasks requiring language semantics and compositional understanding, including paraphrased retrieval.

Limitations

Our method sometimes degrades the performance of CLIP on conventional vision and vision-

language tasks such as zero-shot classification and image retrieval. A significant factor contributing to this performance variation may be the sensitivity of the infoNCE loss to changes in batch size. We observed consistent improvements in the image classification and text/image retrieval tasks by scaling up the batch size from 256 to 3K. Unfortunately, due to constraints in computational resources, we were unable to match the batch size to the scale of CLIP hyperparameters (e.g., OpenAI's CLIP was pre-trained using a batch size of 32K). As a result, the effect of batch size in causing the observed performance degradation has not been thoroughly validated in this study. Although the primary goal of this paper was to showcase the potential improvements in the CLIP model through synthetic paraphrasing and better generalization ability across various input queries, a comprehensive investigation into the factors contributing to performance degradation should be conducted in future research.

Acknowledgements

We thank Fabian Caba Heilbron and Donghee Choi for their help and insightful discussions. This research was supported by (1) National Research Foundation of Korea (NRF-2023R1A2C3004176), (2) ICT Creative Consilience Program through the Institute of Information & Communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT)(IITP-2024-2020-0-01819), and (3) a Korea University Grant.

References

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Iñigo Lopez-Gazpio, Montse Maritxalar, Rada Mihalcea, German Rigau, Larraitz Uria, and Janyce Wiebe. 2015. SemEval-2015 task 2: Semantic textual similarity, English, Spanish and pilot on interpretability. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 252–263, Denver, Colorado. Association for Computational Linguistics.

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2014. SemEval-2014 task 10: Multilingual semantic textual similarity. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 81–91, Dublin, Ireland. Association for Computational Linguistics.

Eneko Agirre, Carmen Banea, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2016. SemEval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 497–511, San Diego, California. Association for Computational Linguistics.

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. SemEval-2012 task 6: A pilot on semantic textual similarity. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 385–393, Montréal, Canada. Association for Computational Linguistics.

Eneko Agirre, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, and Weiwei Guo. 2013. *SEM 2013 shared task: Semantic textual similarity. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, pages 32–43, Atlanta, Georgia, USA. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

Jiacheng Cheng, Hijung Valentina Shin, Nuno Vasconcelos, Bryan Russell, and Fabian Caba Heilbron. 2024. Adapting clip to paraphrased retrieval with pretrained language models.

Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. 2023. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2818–2829.

Yung-Sung Chuang, Rumen Dangovski, Hongyin Luo, Yang Zhang, Shiyu Chang, Marin Soljacic, Shang-Wen Li, Scott Yih, Yoon Kim, and James Glass. 2022. DiffCSE: Difference-based contrastive learning for sentence embeddings. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4207–4218, Seattle,

United States. Association for Computational Linguistics.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark. Association for Computational Linguistics.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*.

Ronald Fagin, Ravi Kumar, and Dakshinamurthi Sivakumar. 2003. Comparing top k lists. *SIAM Journal on discrete mathematics*, 17(1):134–160.

Lijie Fan, Dilip Krishnan, Phillip Isola, Dina Katabi, and Yonglong Tian. 2023. Improving clip training with language rewrites. *Advances in Neural Information Processing Systems*.

Weixi Feng, Xuehai He, Tsu-Jui Fu, Varun Jampani, Arjun Akula, Pradyumna Narayana, Sugato Basu, Xin Eric Wang, and William Yang Wang. 2023. Training-free structured diffusion guidance for compositional text-to-image synthesis. *The Eleventh International Conference on Learning Representations*.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Paul Jaccard. 1912. The distribution of the flora in the alpine zone. 1. *New phytologist*, 11(2):37–50.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. 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.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. 2014. A SICK cure for the evaluation of compositional distributional semantic models. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 216–223, Reykjavik, Iceland. European Language Resources Association (ELRA).

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

OpenAI. 2022. Introducing chatgpt.

Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2015. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

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

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. 2022. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294.

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. 2021. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *NeurIPS Data-Centric AI Workshop 2021*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

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

A Implementation Details

In the data generation process, we used the `gpt-35-turbo-0301` model with the temperature of 1.0 and top-p of 0.1. We paid approximately 130 USD for using ChatGPT to generate 300K paraphrases for captions and 300K additional paraphrases for generated paraphrases.

We used the checkpoints of CLIP models provided in the official OpenCLIP GitHub repository.[4] We used `openai` for OpenAI's CLIP, `laion400m_e32` for OpenCLIP (400M), `laion2b_s34b_b79k` for OpenCLIP (2B), and `laion2b_s12b_b32k` for OpenCLIP-RoBERTa. Our ParaCLIP models were trained for one epoch using the AdamW optimizer (Loshchilov and Hutter, 2019), coupled with a cosine annealing scheduler, on eight A100 80G GPUs. For fine-tuning, a learning rate of 5e-7, a batch size of 3,072, and a weight decay rate of 0.001 were used. All reported scores were measured on a single run.

B Metrics in Paraphrased Retrieval

Average overlap The top-k average overlap (AO@k) (Fagin et al., 2003) quantifies the rank similarity between the top-k elements of the two lists. Let L_a and L_b be ordered lists of retrieved images for two different queries. AO@k between the two lists is calculated based on the weighted sum of intersections of truncated lists as follows:

$$\mathrm{AO@k}(L_\mathrm{a}, L_\mathrm{b}) := \frac{1}{k}\sum_{d=1}^{k}\frac{|L_\mathrm{a}^d \cap L_\mathrm{b}^d|}{d}, \quad (1)$$

where $L_\mathrm{a}^d = L_\mathrm{a}[1:d]$ and $L_\mathrm{b}^d = L_\mathrm{b}[1:d]$ represent the truncated lists at depth d and $|L_\mathrm{a}^d \cap L_\mathrm{b}^d|$ indicates the cardinality of the set intersection between these truncated lists. When AO@k equals 1, it means that the top-k elements of L_a and L_b are exactly the same. Conversely, when AO@k equals 0, it implies that there is no overlap whatsoever between the top-k elements of L_a and L_b. AO@k gives more weight to the higher-ranked retrieval results because they contribute to more terms in the overall summation compared to lower-ranked results.

Jaccard similarity The top-k Jaccard similarity (JS@k) (Jaccard, 1912) is calculated as the ratio of the intersection to the union of the top-k elements in two lists as follows:

$$\mathrm{JS@k}(L_\mathrm{a}, L_\mathrm{b}) := \frac{|L_\mathrm{a}^k \cap L_\mathrm{b}^k|}{|L_\mathrm{a}^k \cup L_\mathrm{b}^k|}, \quad (2)$$

where $|L_\mathrm{a}^k \cup L_\mathrm{b}^k|$ is the cardinality of the set union between L_a^k and L_b^k. JS@k equals 0 when L_a^k and L_b^k are disjoint and equals 1 when L_a^k and L_b^k contain the same retrieval results (although not necessarily in the same order). Unlike the average overlap, the Jaccard similarity does not assign more weight to the higher-ranked retrieval results.

C Case Study

Figure 3 shows several examples where our ParaCLIP model yieled better retrieval results than OpenAI's CLIP for paraphrased queries. In the first example, the paraphrased query (query B) contained several synonyms such as "picture," "guy," "cutting," and "tiny," replacing the words "image," "man," "slicing," and "small," respectively. While the CLIP model output dissimilar results for the given two queries, resulting in a performance drop for query B, ParaCLIP consistently produced identical results for both queries. In the second example, the only difference between the queries was the word "was." Despite this minor variation, CLIP generated different sets of images. On the other hand, ParaCLIP returned the same images for both queries and achieved a better recall for query B, although the recall score for query A was slightly lower than that of CLIP. In the last example, query B was created by expanding the short query A into longer expressions. For instance, the concise phrase "a remote control" was transformed into the more elaborate phrase "a controller for a television that is wirelessly operated." While CLIP exhibited high sensitivity to this long paraphrased query, ParaCLIP demonstrated greater robustness, resulting in more consistent results and superior recall scores.

[4] https://github.com/mlfoundations/open_clip

Figure 3: Examples of retrieved images by the CLIP (Radford et al., 2021) and our ParaCLIP models for two different queries. Note that the queries are obtained from the paraphrased retrieval dataset, and query B is a paraphrase for query A. The gold images are denoted by a bold border.