LLM2CLIP: Powerful Language Model Unlocks Richer Cross-Modality Representation

Weiquan Huang[1*], Aoqi Wu[1*], Yifan Yang[2†‡], Xufang Luo[2], Yuqing Yang[2], Usman Naseem[3], Chunyu Wang[2], Qi Dai[2], Xiyang Dai[2], Dongdong Chen[2], Chong Luo[2], Lili Qiu[2], Liang Hu[1†]

[1] School of Computer Science dnd Technology, Tongji Univeristy, Shanghai, China
[2] Microsoft Corporation
[3] School of Computing, Macquarie University, Sydney, Australia
yifanyang@microsoft.com, lianghu@tongji.edu.cn

Abstract

CLIP is a seminal multimodal model that maps images and text into a shared representation space by contrastive learning on billions of image–caption pairs. Inspired by the rapid progress of large language models (LLMs), we investigate how the superior linguistic understanding and broad world knowledge of LLMs can further strengthen CLIP—particularly in handling long, complex captions. We introduce an efficient fine-tuning framework that embeds an LLM into a pretrained CLIP while incurring almost the same training cost as regular CLIP fine-tuning. Our method first "embedding-izes" the LLM for the CLIP setting, then couples it to the pretrained CLIP vision encoder through a lightweight adaptor trained on only a few million image–caption pairs. With this strategy we achieve large performance gains—without large-scale retraining—over state-of-the-art CLIP variants such as EVA02 and SigLIP-2. The LLM-enhanced CLIP delivers consistent improvements across a wide spectrum of downstream tasks, including linear-probe classification, zero-shot image–text retrieval with both short and long captions (in English and other languages), zero-shot/supervised image segmentation, object detection, and used as tokenizer for multimodal large-model benchmarks.

1 Introduction

CLIP (Radford et al. 2021; Tschannen et al. 2025) has emerged as one of the most influential cross-modal foundation models in recent years. Trained on hundreds of millions to tens of billions of image–text pairs via contrastive pre-training, it embeds vision and language into a shared representation space. As a *retriever*, CLIP underpins zero-shot classification, detection, segmentation, and—most notably—image–text retrieval. As a *feature extractor*, it supports a broad spectrum of cross-modal applications, from image/video understanding to text-to-image and text-to-video generation. For instance, Multi-modality large language models such as LLaVA (Li et al. 2024a) and Qwen-VL (Bai et al. 2023) rely on CLIP's visual features, while image/video generative model like Stable-Diffusion-3 (Esser et al. 2024) and Wan (Kong et al. 2024) leverage its text encoder. Nevertheless, as we push toward broader generality and higher task complexity, the representational capacity inherited from CLIP's original paradigm is gradually becoming inadequate.

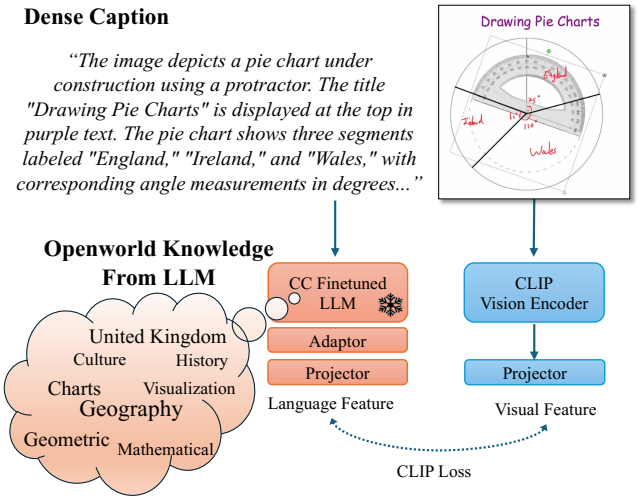

Figure 1: *LLM2CLIP* Overview. After applying caption contrastive fine-tuning to the LLM, the increased textual discriminability enables more effective CLIP training. We leverage the open-world knowledge and general capabilities of the LLM to better process dense captions, addressing the previous limitations of the pretrained CLIP visual encoder and providing richer textual supervision.

The rapid progress of large language models (LLMs) has dramatically advanced textual understanding and generation, motivating us to inject stronger LLMs into the CLIP training pipeline so as to bolster multimodal capability. Prior work such as LLM2Vec (BehnamGhader et al. 2024) and NV-Embed-v2 (Lee et al. 2024) has shown that LLMs can be transformed into competitive text–embedding models, now topping leaderboards like MTEB (Muennighoff et al. 2022). Follow-up efforts (e.g. MM-E5 (Chen et al. 2025),

*These authors contributed equally.
†Corresponding authors.
‡Project Lead.

VLM2Vec (Jiang et al. 2024)) have extended this idea to multi-modality representations. Yet CLIP's light dual-tower architecture remains the de-facto embedding model in many scenarios. For instance, in standard zero-shot image–text retrieval, a 400 M-parameter SigLIP-2 So/14 reaches 85.7 (T→I) and 94.9 (I→T) on Flickr30K at 384×384 resolution, while VLM2Vec—built on a 7B-parameter LLaVA-1.6 and trained at 1344×1344—achieves 79.8 and 91.6, respectively. CLIP's lightweight design also scales more readily to large datasets and batch sizes (SigLIP-2 (Tschannen et al. 2025) is trained on ∼40 B image–text pairs). Consequently, mainstream multimodal retrieval systems—and the multimodal encoders that serve VL-LMs and diffusion models—are increasingly built on CLIP-based architectures. This raises a key question: how can we efficiently harness the strengths of modern LLMs to augment these pretrained CLIP models and further elevate their capabilities?

The potential benefits of incorporating LLMs into CLIP are clear. LLMs' strong textual understanding can fundamentally improve CLIP's ability to handle image captions, drastically enhancing its ability to process long and complex texts—a well-known limitation of vanilla CLIP. Moreover, LLMs are trained on a vast corpus of text, possessing open-world knowledge. This allows them to expand on caption information during training, increasing the efficiency of the learning process.

Bringing LLMs into the CLIP framework, however, raises two key challenges: **1. Feature separability.** Vanilla LLM embeddings are not sufficiently discriminative for contrastive training; existing "LLM-as-embedding" recipes target pure-text tasks. We therefore devise a CLIP-specific *embeddingization* strategy for LLMs that yields far more separable caption features. **2. Training cost.** CLIP pre-training is already expensive; naively fine-tuning an LLM jointly would be prohibitive. We propose a lightweight *fine-tuning* procedure that injects LLM power into a pretrained CLIP at almost no extra cost.

This paper introduces *LLM2CLIP*, an efficient fine–tuning framework that augments the feature space of a *pre-trained* CLIP with an *embedding-tuned* LLM, thereby importing LLM capability at **very low cost**. Injecting a *vanilla* LLM into CLIP is problematic: as shown by the cases in Figure 2, a COCO test reveals that raw LLM embeddings have poor separability for image captions. To remedy this, we perform *caption-contrastive(CC) fine-tuning* on the LLM with a set of high-quality caption datasets, revisiting several design choices—e.g. employing **average pooling** to aggregate token features, enabling **bidirectional attention**, and training with a **supervised SimCSE** contrastive loss.

Another key challenge when introducing LLMs into CLIP is the presence of two text encoders—the original CLIP text encoder and the newly added LLM—which were never aligned during pre-training. We conduct experiments to explore efficient integration of the LLM into the CLIP architecture. As shown in Figure 1, we adopt a cost-effective solution: *freeze all LLM gradients*, treat its sentence embeddings as fixed features, and append a small learnable *adaptor* trained with the CLIP *visual encoder*. The original CLIP text encoder is discarded during both training and inference. Ablation studies indicate that alternative strategies—combining CLIP and LLM embeddings via triplet contrastive losses or concatenating features—provide marginal or negative gains compared to simply replacing the CLIP text encoder. This minimal design yields an LLM-enhanced cross-modal space while keeping compute cost nearly identical to standard CLIP fine-tuning.

Empirical results confirm that *LLM2CLIP* yields substantial improvements—even with only a few million training examples—boosting the performance of the original CLIP across a variety of downstream tasks. Our contributions are as follows:

- We empirically demonstrate that injecting LLM capability into CLIP brings significant performance gains.
- We design (i) a caption-contrastive fine-tuning recipe that turns an LLM into a effective embedding model for CLIP, and (ii) a efficient fine-tuning method that couples this embedding with a pretrained CLIP.
- Extensive experiments reveal that the resulting **LLM-enhanced CLIP** markedly improves several state-of-the-art models—including EVA02 and SigLIP-2—on a wide spectrum of multimodal benchmarks: short/long-text and cross-lingual image retrieval, zero-shot classification, detection, segmentation, and even as the visual encoder inside LLaVA-1.5. Remarkably, *LLM2CLIP* lifts SigLIP-2's short-caption retrieval by +1.0/+1.9, long-caption retrieval by +14.8/+15.8, and multilingual tasks by +11.9/+15.2.

Figure 2: Real examples of top-1 results from the caption-to-caption retrieval experiment in MS COCO 5K test set. Before fine-tuning, Llama3's results were often unrelated.

2 Related Works

CLIP meets Stronger Language Models. Several works have explored the integration of LLMs into CLIP. Jina-CLIP (Koukounas et al. 2024a) employed Jina-embeddings-v2 (Günther et al. 2023) as the text encoder, which is a BERT variant with 137M parameters, supporting longer texts. Though achieving similar visual performance to EVA-CLIP (Sun et al. 2023), its text encoder is far behind ours, limiting the potential benefits from LLMs. MATE (Jang et al. 2024) designed a learnable adaptor to bridge the gap

between CLIP's text encoder and LLMs. They trained the CLIP visual encoder using LoRA on a small dataset focused on long-text image retrieval tasks. However, they did not recognize the critical issue we propose: the poor separability of LLM feature space, which is insufficient for direct support of CLIP training. This paper presents the first comprehensive study of an efficient strategy for incorporating LLMs into CLIP fine-tuning, thereby boosting performance.

CLIP meets Longer Captions. CLIP's text embedding is widely recognized as coarse and limited to 77 tokens. Many works have attempted to extend caption length and retrain CLIP accordingly, including DCI (Urbanek et al. 2024) with human annotation, LaCLIP (Fan et al. 2024) using ChatGPT and Bard, DreamLIP (Zheng et al. 2024) leveraging Share-Captioner (Chen et al. 2023), and Recap-DataComp-1B (Li et al. 2024b) using Llama3-trained LLAVA1.5. To handle longer captions, these methods employ workarounds such as summarization (Urbanek et al. 2024), segmentation (Fan et al. 2024; Zheng et al. 2024), or positional encoding fine-tuning (Zhang et al. 2024a). In comparison, LLM2CLIP leverages LLMs as the text encoder, enabling comprehensive understanding of long and dense captions while utilizing LLMs' open-world knowledge.

3 Methods

3.1 Stage 1: LLM Caption Contrastive Fine-tuning

Traditionally, an LLM's output layer functions as a classification head that produces discrete text tokens. Recent studies—such as LLM2Vec (BehnamGhader et al. 2024) and NV-EMBED-v2 (Lee et al. 2024)—have shown that, with additional design tweaks, an LLM can be repurposed as a text-embedding model. Yet these efforts do not account for CLIP's cross-modal pre-training regime. Here, we advance the discussion from three fronts—*model architecture*, *training methods* and *training data*—explicitly targeting the seamless integration of LLMs with CLIP.

Model Architecture:

- **Sentence Token Representation:** We map the output layer features of the LLM to represent whole sentences. We considered two strategies: using the [EOS] token, which can attend to the entire sentence during pretraining, or employing average pooling across all output tokens. Empirically, average pooling performed better and is our default setting.
- **Bidirectional Attention:** Given that the generative capability is unnecessary for our encoder-focused application, we remove the attention mask from the LLM, enabling bidirectional textual relationship modeling for enhanced comprehension.
- **Fine-tuning Parameters:** To efficiently activate stronger textual comprehension capabilities within the output features of LLMs, we apply parameter-efficient fine-tuning using Low-Rank Adaptation (LoRA).
- **Adaptor Design:** Adding an adaptor following the LLM can potentially enhance feature separability. We experimented with three adaptor variants: (1) a Transformer-based adaptor composed of several latent cross-attention transformer layers following NV-EMBED-v2, (2) a Linear adaptor consisting of multiple linear layers directly attached after extracting sentence tokens from LLM outputs and (3) without using adaptor.

Training Methods:

- **Masked Next Token Prediction (MNTP):** Inspired by masked language modeling in BERT, MNTP improves feature quality by masking and predicting specific tokens. Unlike BERT, predictions are always made at the position preceding the masked token, aligning with LLM's next-token prediction convention. While effective in pure text tasks (as shown by LLM2Vec), we find out that MNTP alone or combined with contrastive learning underperformed relative to direct contrastive fine-tuning in multimodal scenarios.
- **Textual Contrastive Learning:** Following SimCSE, we employ textual contrastive learning to enhance feature separability, pulling positive samples closer and pushing negative samples apart in feature space. Two variants are possible: unsupervised (using dropout augmentation) and supervised (leveraging semantic pairs from annotated data). We default to supervised contrastive learning due to superior performance. Specifically, we generate positive pairs using two distinct captions from the same image, framed by a system prompt: "Given a caption, retrieve a similar relevant caption."

Training Data:

To leverage LLM embeddings for *CLIP fine-tuning*, we fine-tune the LLM on caption corpora extracted from image–text datasets, aligning its embedding space with downstream multimodal tasks. Experiments use the Dreamlip (Zheng et al. 2024) captions, which provide multiple captions per image, suitable for supervised textual contrastive learning. To preserve general language understanding, we also incorporate 1.5M pure-text pairs from Echo Embeddings (Springer et al. 2024) into a mixed contrastive training setup.

3.2 Stage 2: *LLM2CLIP* Post Fine-tuning

We aim to perform post fine-tuning on pretrained CLIP since we believe the fundamental visual-language mapping has been well-established through CLIP's extensive pretraining. By introducing an LLM, we expect to further enrich this representation space, potentially improving CLIP's capabilities through a highly cost-effective fine-tuning step. CLIP inherently consists of a Vision Encoder (typically a ViT) and a relatively small Text Encoder (roughly one-third the parameters of the Vision Encoder), structured as a compact autoregressive model. Integrating a caption-contrastive fine-tuned LLM as an additional text encoder and finding an effective approach to enhance CLIP performance is critical. Here, we also elaborate on *model architecture*, *training methods* and *training data*.

Model Architecture: Training CLIP typically requires large-scale datasets and substantial batch sizes, resulting in considerable computational costs. Therefore, we employ efficient Parameter-Efficient Fine-Tuning (PEFT) methods for this stage. By default, we enable full gradients for the visual

Method	Res	Flickr		COCO		SG4V		Urban		DOCCI		Avg	
		I2T	T2I	I2T	T2I	I2T	T2I	I2T	T2I	I2T	T2I	I2T	T2I
ViT-B/16 (86M)													
ALIGN (Jia et al. 2021)	224	80.6	62.2	52.0	43.2	75.9	80.6	62.2	59.1	59.7	62.1	66.1	61.4
BLIP (Li et al. 2022)	224	80.6	74.1	61.7	48.5	65.8	74.3	45.5	48.5	50.5	53.5	60.8	59.8
Long-CLIP (Zhang et al. 2024a)	224	85.8	70.6	56.9	40.9	94.8	93.5	79.1	79.1	63.1	71.4	75.9	71.1
jina-clip-v2 (Koukounas et al. 2024b)	224	84.4	69.8	57.1	41.9	94.5	91.1	80.4	78.0	77.6	78.2	78.8	71.8
SigLIP (Zhai et al. 2023)	224	89.1	74.7	65.8	47.8	85.8	83.5	62.8	62.4	70.1	70.6	74.7	67.8
MetaCLIP (Xu et al. 2023)	224	85.6	70.8	59.3	41.3	90.4	86.6	68.9	63.3	70.9	71.5	75.0	66.7
CLIP (Radford et al. 2021)	224	82.3	62.2	52.4	33.1	84.5	79.8	67.5	53.1	60.7	57.1	69.5	57.1
+*LLM2CLIP*-15M	224	**91.8**	80.7	64.8	52.4	97.8	97.6	**90.1**	**92.1**	86.1	86.9	**86.1**	81.9
EVA02 (Sun et al. 2023)	224	86.2	71.5	58.7	42.1	90.5	85.5	67.0	60.8	67.7	68.0	74.0	65.6
+*LLM2CLIP*-15M	224	91.2	**81.4**	**66.0**	**53.9**	**98.5**	**98.5**	88.2	91.3	**86.8**	**88.5**	**86.1**	**82.7**
ViT-L/14 (307M)													
Long-CLIP (Zhang et al. 2024a)	224	90.0	76.2	62.8	46.3	97.2	97.3	82.5	86.1	66.5	78.6	79.8	76.9
MetaCLIP (Xu et al. 2023)	224	90.1	76.4	64.5	47.1	86.4	79.5	73.4	70.0	76.5	76.7	78.2	69.9
L-NV2 (Zhang, Yang, and Agrawal 2025)	224	87.9	75.8	62.4	48.7	94.8	94.9	81.5	80.2	76.5	78.9	80.6	75.7
CLIP (Radford et al. 2021)	224	85.2	65.0	56.3	36.5	84.2	83.6	68.3	55.6	63.1	65.8	71.4	61.3
+*LLM2CLIP*-15M	224	93.0	83.5	67.4	56.4	98.7	98.6	91.5	94.1	88.3	90.4	87.8	84.6
EVA02 (Sun et al. 2023)	224	89.7	77.3	63.7	47.5	91.9	89.3	73.3	68.5	73.5	75.0	78.4	71.5
+*LLM2CLIP*-3M	224	94.3	84.3	65.2	56.0	98.0	98.0	91.0	94.7	87.8	90.7	87.3	84.7
+*LLM2CLIP*-15M	224	94.6	85.0	69.5	58.3	98.9	99.1	93.6	95.7	89.8	91.2	89.3	85.8
+*LLM2CLIP*-60M	224	**95.9**	**85.1**	**71.7**	**58.5**	**99.2**	**99.2**	**95.2**	**97.0**	**90.1**	**92.0**	**90.4**	**86.3**
CLIP (Radford et al. 2021)	336	87.7	67.0	58.0	37.1	86.2	84.0	72.8	57.0	67.4	65.7	74.4	62.2
+*LLM2CLIP*-15M	336	91.2	82.1	65.5	53.6	98.1	98.4	90.3	93.2	87.7	89.0	86.6	83.3
+*LLM2CLIP*-60M	336	93.9	84.3	68.5	54.8	**98.9**	99.1	**96.6**	**97.9**	89.6	91.6	**91.1**	84.5
EVA02 (Sun et al. 2023)	336	89.6	78.0	64.2	47.9	91.5	89.4	76.6	70.0	74.7	76.4	79.3	72.3
+*LLM2CLIP*-15M	336	94.7	85.1	69.7	58.5	98.8	**99.2**	93.5	96.2	89.2	91.3	88.0	84.8
+*LLM2CLIP*-60M	336	**95.9**	**85.4**	**72.4**	**58.8**	98.8	**99.2**	95.5	97.3	**90.7**	**92.6**	91.0	**86.8**
SigLIP (Zhai et al. 2023)	384	93.7	81.4	72.0	53.9	90.1	90.4	76.3	74.4	77.8	79.5	82.0	75.9
SO/14 (428M)													
SigLIP (Zhai et al. 2023)	224	91.0	75.2	69.8	51.8	56.1	50.0	31.5	29.1	42.3	44.8	58.1	50.2
SigLIP2 (Tschannen et al. 2025)	224	93.9	82.9	72.0	55.5	90.2	87.2	75.7	74.5	77.0	78.9	81.8	75.8
+*LLM2CLIP*-60M	224	**95.2**	**84.6**	**73.7**	**57.6**	**99.4**	**99.4**	**96.2**	**97.3**	**91.5**	**92.9**	**91.2**	**86.4**
SigLIP (Zhai et al. 2023)	384	94.3	83.0	72.4	54.3	91.6	89.4	74.5	73.6	74.5	75.8	81.5	75.2
InternVL (6B)													
InternVL (Chen et al. 2024)	224	94.2	81.6	69.4	53.2	92.4	90.5	79.6	77.7	79.2	81.6	83.0	76.9
VLM2Vec (7B)													
VLM2Vec* (Jiang et al. 2024)	1344	91.6	79.8	61.3	51.5	93.9	91.0	90.9	92.0	80.5	86.0	83.6	80.1

Table 1: Systematic comparison of model performance on multiple datasets. * means reproduced by ourselves.

encoder to facilitate learning from LLM knowledge. Given the large size of the LLM, we compare two strategies: (1) fine-tuning the LLM with LoRA, and (2) freezing LLM gradients and appending previously introduced Transformer or Linear adaptors after its output layer as learnable modules to obtain sentence token features. Method (2) offers clear advantages: it completely eliminates LLM gradient updates, significantly reducing GPU memory usage. Moreover, since the adaptor is placed after the LLM, text features can be precomputed and stored via offline inference, reducing inference overhead from multiple epochs to a single pass. As a result, this approach avoids loading the LLM into GPU memory during training, accelerating experimentation and reducing training time and memory footprint. By default, we adopt method (2) with an adaptor of four linear blocks. Detailed efficiency analysis is provided in Section 4.4.

Training Methods: After introducing an LLM into CLIP, we are left with two text encoders. How should they be used most effectively? We tested several approaches: (a) Removing the original CLIP Text Encoder and using LLM with CLIP Vision Encoder directly for cross-modal contrastive learning. (b) Retaining both LLM and original Text Encoder, applying separate CLIP losses to both encoders. (c) Adding an additional contrastive learning task between LLM and the original Text Encoder to method (b). (d) Concatenating LLM and Text Encoder output features in method (b), then performing contrastive learning with CLIP Vision Encoder. Our extensive experiments led us to adopt method (a) for its simplicity and effectiveness.

Training Data: With the integration of LLM, our method inherently excels at processing dense and detailed image captions. Datasets such as DreamLIP and Recaption, which leverage Share-Captioner (Chen et al. 2023), InstructBLIP (Dai et al. 2023) and LLaVA-1.5 (Liu et al. 2023a) to generate multiple dense captions per image, can be effectively utilized. We explored multiple caption scenarios, employing multiple positive caption examples during contrastive training and blending real and MLLM-generated

dense captions. Ultimately, we adopted a strategy blending real captions and dense MLLM captions at a 50% ratio.

4 Experiments

4.1 Default Experimental Settings

Default Settings for Stage-1 We choose to use LLaMA 3.1 8B as the LLM integrated into CLIP pre-training, removing the causal attention mask to enable bidirectional attention. No additional adapters are used in caption contrastive fine-tune, and average pooling is employed to obtain the sentence token embedding. MNTP is not used by default.

We utilize 30M DreamLIP caption data to perform caption contrastive fine-tuning on the LLM. Randomly sampling any two captions of an image from DreamLIP as positive pairs, we conduct supervised SimCSE training.

Default Settings for Stage-2 After caption contrastive fine-tuning, we replace the original CLIP text encoder with the LLaMA-3.1-8B to perform cross-modal contrastive learning fine-tuning with the vision encoder. By default, we open the ViT gradients and freeze the LLM gradients, then add an adapter layer composed of four linear layers after the LLM. The design of the Adaptor is the same as FuseMix (Vouitsis et al. 2023), using a simple inverted bottleneck MLP architecture. By default, we use 15M DreamLIP annotated subsets of CC 3M and CC 12M. For the 60M data setting, we use DreamLIP re-annotated CC 3M and CC 12M, YFCC 15M, and a 30M subset of the LAION dataset. For the 3M setting, we only use the CC 3M subset.

4.2 System Comparison

In the system comparison, we used the default settings from Section 4.1. However, we also conducted experiments on different pretrained SOTA CLIP models to verify that applying LLM2CLIP post training can significantly enhance their performance.

Zero-Shot English Text-Image Retrieval For short-text retrieval, we used the MSCOCO (Lin et al. 2014) 5K test set and the Flickr (Young et al. 2014) 1K test set. For long-text retrieval, we employed a 1K subset of ShareGPT4V (Chen et al. 2023) and the Urban1K (Zhang et al. 2024a) dataset from LongCLIP (Zhang et al. 2024a), along with the DOCCI (Onoe et al. 2024) dataset.

The ShareGPT4V-1M dataset consists of captions generated using GPT-4V and ShareCaptioner, covering images from LAION, CC, SBU (Ordonez, Kulkarni, and Berg 2011), and MS COCO. Urban1K includes captions for 1,000 urban scene images, each richly annotated with detailed descriptions. DOCCI contains 1.5K high-resolution images with human-annotated captions and was used for retrieval evaluation. To facilitate observation, we additionally report the average Top-1 image-to-text (I2T) and text-to-image (T2I) accuracy across these five datasets.

***LLM2CLIP* Improves CLIP for Long- and Short-Text Retrieval.** In Table 1, we compared OpenAI's CLIP, EVA02, and SigLIP2, and conducted experiments using three types of visual models: ViT-B/16, ViT-L/14, and So-ViT/14. Regardless of the model size or resolution shown in the table, *LLM2CLIP* yields a significant performance boost for both CLIP and EVA02. For example, under EVA02 at 224 resolution, the LLM2CLIP-60M design achieves an average Top-1 retrieval accuracy improvement of +12 and +14.8. Even for SigLIP2—which was pretrained on 40B data—our method still brings an average performance improvement of +9.4 and +10.6; even on the short-caption datasets Flickr and COCO, where SigLIP2 is already very familiar, improvements of +1 and +1.9 are observed. Comparatively, the improvements for long captions are even larger, as the inherent large window of LLMs fully leverages their advantage in understanding long and complex texts. Notably, *LLM2CLIP* even surpasses *InternVL*—which contains 6 B trainable parameters—and *VLM2Vec*, whose pipeline fine-tunes the entire LLM with LoRA.

Models	Flickr-CN		COCO-CN		XM3600	
	I2T	T2I	I2T	T2I	I2T	T2I
CN-CLIP	80.2	68.0	63.4	64.0	–	–
EVA-L-224	4.4	0.9	2.6	1.0	14.0	8.0
+*LLM2CLIP*	**90.6**	75.6	**72.0**	70.1	68.3	56.0
SigLIP2	79.2	56.9	55.3	51.7	59.7	48.2
+*LLM2CLIP*	90.0	**76.1**	70.8	**70.2**	**69.1**	**56.3**

Table 2: Multi-lingua retrieval results on different datasets.

Methods	Imagenet		
	0-shot*	0-shot	Linear
CLIP L/14-336	**74.9**	**76.6**	84.8
+*LLM2CLIP*	74.6	75.8	**85.2**
CLIP L/14	**73.7**	**75.5**	83.9
+*LLM2CLIP*	73.2	74.3	**84.4**

Table 3: Zero-shot classification and linear probe performance on ImageNet. *0-shot uses the class template *'a photo of the* {*classname*}' only.

Method	Zero-shot Seg. mIOU				OV-COCO Det.			COCO val2017
	COCO-S	ADE	VOC	City	Novel	Base	All	AP^{bb}/AP^{seg}
EVA02	12.9	11.5	21.0	13.5	24.7	53.6	46.0	45.0/38.2
+LLM2CLIP	**15.3**	**15.8**	**29.1**	**20.1**	**28.9**	**54.7**	**48.0**	**45.6/38.7**

Table 4: Zero-shot/supervised segmentation, open-vocabulary detection benchmarks. COCO val2017 is supervised results using CLIP's visual encoder.

Fine-tuning Data Volume Analysis. In Table 1, we experimented with 3M, 15M, and 60M training data on CLIP and EVA02 using ViT-L/14. More data consistently improves retrieval performance for both long and short text tasks at 224 and 336 resolutions. With 3M data, EVA02 shows noticeable improvement in long text retrieval but minimal gain in short text retrieval, suggesting the model first compensates for disrupting the original cross-modal space before achieving broader improvements.

Zero-Shot Multilingual Text-Image Retrieval As in Table 2, We conducted cross-lingual retrieval experiments

Model	VQA					Pope			MM					Seed		
	V2	GQA	Vz	SQA	TV	R	A	P	MME	MB	MC	LB	MV	All	I	V
Llava (Paper)	78.5	62.0	50.0	66.8	58.2	87.3	86.1	84.2	1510.7	64.3	58.3	65.4	31.1	58.6	66.1	37.3
Llava (Rep.)	79.04	62.86	50.57	67.97	57.48	87.7	**84.85**	86.3	1476.69	66.66	60.39	58.0	34.3	59.86	66.95	**39.71**
+LLM2CLIP	**79.80**	**63.15**	**52.37**	**69.92**	**58.35**	**88.55**	82.76	**87.75**	**1505.82**	**68.29**	**60.40**	**62.7**	**34.8**	**60.96**	**68.80**	38.96

Table 5: Performance of Llava 1.5 benchmarks. For + *LLM2CLIP* we replace Llava's CLIP Vit-L/14 with our finetuned version.

where CLIP vision encoders were trained exclusively on English text. We successfully endowed EVA02 with multilingual capabilities, which it previously lacked, and significantly enhanced the already robust multilingual performance of SigLIP2.

Zero-shot & Linear Probe ImageNet Classification We evaluate our model on ImageNet using two standard protocols: (1) **Zero-shot classification**, where we report accuracy using the prompt template "a photo of the {classname}", following the CLIP methodology. We also evaluate the average performance over 80 handcrafted prompt variants, as commonly practiced in prior work. (2) **Linear probing**, where we freeze the visual encoder and train a linear classifier using fixed hyperparameters across all experiments (batch size 1024, learning rate 0.1, momentum 0.9, weight decay 0, 50 epochs with SGD).

As shown in Table 3, we conduct experiments using CLIP ViT-L/14 at both 224 and 336 resolutions. We observe that after applying *LLM2CLIP*, zero-shot performance on ImageNet shows a modest drop. However, the accuracy from linear probing *improves* over the original CLIP baseline. This suggests that while the quality of the learned *visual features* remains strong (as linear probing only tests CLIP visual encoder's representation power using a supervised head), the overall *multimodal alignment* in the shared space may slightly deteriorate since zero-shot classification drops, especially for fine-grained category separation.

We hypothesize that the drop in zero-shot performance could be mitigated by increasing the amount of LLM2CLIP fine-tuning data, which remains to be verified in future work. Nonetheless, this trade-off echoes a common pattern observed in recent CLIP fine-tuning studies, such as LongCLIP (Zhang et al. 2024a) and CLIP-MoE (Zhang et al. 2024b), where performance on head classes improves while long-tail categories suffer due to distribution imbalance—particularly under limited-scale fine-tuning. The contrast with retrieval results is instructive: retrieval benefits greatly because its vocabulary is broad and frequent, whereas classification demands uniform discriminability across all (often obscure) class nouns.

Zero-Shot / Supervised Segmentation and Object Detection As our linear-probe results already suggest, the visual features produced by the encoder become noticeably stronger after LLM2CLIP training. Table 4 further confirms this on both zero-shot and fully supervised object detection and segmentation benchmarks. In the zero-shot setting—which relies on the text encoder—LLM2CLIP improves CLIP's multimodal representations; in the supervised setting—where only the vision encoder is finetuned—it also enhances the purely visual, low-level feature extractor. We conjecture that these gains stem from the LLM's ability to parse dense captions more accurately, especially spatial terms and object-to-object relations, and to inject that knowledge into the joint vision–language space.

Multimodal Large Language Models Performance Following LLAVA1.5 (Liu et al. 2023b), we used OpenAI's CLIP-ViT-L-336 encoder with a simple MLP head to connect to Vicuna-7B. Pretraining included 558K image-caption pairs and 665K visual instruction samples. We finetuned the LLAVA1.5 encoder using *LLM2CLIP* to assess improvements in visual feature extraction. As shown in Table 5, *LLM2CLIP* fine-tuning enhanced multimodal model performance on over 87.5% of benchmarks, with minor losses in only two tasks. This highlights the potential of *LLM2CLIP* for improving CLIP visual encoder's abilities for complex image reasoning and understanding.

Stage1	Avg	
	I2T	T2I
Lora, AvgPool, Bidirectional, Supervise Simcse	**80.4**	**77.9**
Lora, AvgPool, Bidirectional, Un-supervise Simcse	59.2	57.7
Lora, AvgPool, Bidirectional, MNTP	70.1	67.0
Lora, AvgPool, Bidirectional, MNTP, Supervise Simcse	79.7	77.2
Lora, AvgPool, Casual, Supervise Simcse	80.0	77.5
Lora, EOS, Bidirectional, Supervise Simcse	80.0	77.3
Frozen, Linear Adaptor, AvgPool, Bidirectional, Supervise Simcse	74.1	71.3

Table 6: Ablation study on the training methods of LLM caption contrastive finetuning in Stage 1. red text indicates content that differs from the default setting.

4.3 Ablation Study: Stage-1 Caption Contrastive Fine-tuning

For simplicity, we select Llama 3.1 1B as the default LLM to investigate different design choices for caption contrastive fine-tuning in this section. We use a subset of the CC 3M dataset for caption contrastive fine-tuning. Additionally, we explore augmenting the training data with the Wikitext-103 dataset (Merity et al. 2016) for MNTP training and the E5 dataset (Springer et al. 2024) as supplementary pure-text data for caption contrastive fine-tuning.

Architecture Design Ablation. We examine various adaptor architecture in Table 8. Comparing rows 4 and 5, as well as rows 6 and 7, we observe that neither Linear nor Transformer adaptors significantly impact the performance of subsequent CLIP fine-tuning. Therefore, we default to using *no adaptor* in Stage 1.

Training Method Ablation. We conduct an ablation

Method	Avg	
	I2T	T2I
CLIP	74.4	72.0
Directly Finetune (50%)	74.5	72.3
bge-en-icl	78.9	78.2
LLM2Vec-Llama-3-8B	81.4	80.2
NV-Embed-v2	81.4	79.9
VLM2Vec	78.2	77.1
bge-m3-XLM-R	65.0	63.6
jina-v3-XLM-R	73.6	71.0
e5 (XLM-R)	74.0	71.7
Qwen2.5-0.5B-CC	75.6	73.0
Llama-3.2-1B-CC	80.4	77.9
Llama-3-8B-CC	83.4	80.9
DeepSeek-R1-Distill-Llama-8B-CC	83.5	80.5
Llama-3.1-8B-CC	**84.8**	**81.0**
Llama3.1-8B	66.5	62.5

Table 7: Ablation for using different text encoder. "-CC" indicates encoders with caption contrastive fine-tuning.

Stage1	Stage2	Avg	
		I2T	T2I
–	–	78.3	75.5
–	Linear($\times$1)	79.2	76.7
–	Linear($\times$2)	80.1	76.8
–	Linear($\times$4)	80.4	**77.9**
Linear($\times$4)	Linear($\times$4)	**80.5**	77.7
–	Transformer($\times$1)	80.2	77.3
Transformer($\times$1)	Transformer($\times$1)	**80.5**	77.3

Table 8: Ablation experiments for adaptor design on Stage 1 caption contrastive finetune and Stage 2 *LLM2CLIP* post training. The 4-layer Linear Adaptor has 67.1M parameters, while the single layer Transformer Adaptor has 67.6M.

study of Stage-1 LLM caption contrastive fine-tuning methods in Table 6, observing the following key insights: 1. Freezing LLM gradients and relying solely on adaptors yields poor performance, indicating that enabling LoRA is essential. 2. SimCSE is identified as the most critical loss function. MNTP does not significantly enhance SimCSE, and relying solely on MNTP without SimCSE severely deteriorates performance. 3. Supervised SimCSE substantially outperforms unsupervised SimCSE. 4. The performance difference between causal and bidirectional attention is negligible.

Ablation across different LLM backbones. Table 7 compares a variety of text-embedding models plugged into our pipeline. Our approach delivers substantial gains over all contenders. The line highlighted in red shows that *plain* Llama 3.1-8B—without our caption-contrastive (CC) fine-tuning—performs very poorly and even harms the original CLIP, underscoring the necessity of the CC stage. We also evaluated several state-of-the-art embedding models, including `bge`, `jina-v3`, and the VLM2Vec embedding obtained from a multimodal LLM fine-tune. In every case, the *LLM2CLIP* variant still achieves the best results, confirming the effectiveness of our training recipe.

Methods		Training Loss	Testing Text Encoder	Average	
				I2T	T2I
CLIP		CL(CLIP-T, CLIP-V)	CLIP-T	74.4	72.0
Directly Finetune		CL(CLIP-T, CLIP-V)	CLIP-T	74.5	72.3
+LLM2CLIP	a)	CL(LLM, CLIP)	LLM	83.9	82.1
	b)	CL(LLM, CLIP-V)+ CL(CLIP-T, CLIP-V)	CLIP-T	74.4	72.0
			LLM	83.6	81.8
	c)	CL(LLM, CLIP-V)	CLIP-T	74.0	72.1
		+CL(CLIP-T, CLIP-V) +CL(CLIP-T, LLM)	LLM	83.7	81.4
	d)	CL(LLM, CLIP-V)	CLIP-T	74.8	71.3
		+CL(CLIP-T, CLIP-V)+ CL(Cat(CLIP-T, LLM), CLIP-V)	LLM	83.8	82.3
			Cat(CLIP-T, LLM)	**84.7**	**82.8**

Table 9: *LLM2CLIP* Training Method Analysis. We experimented with various possibilities for fine-tuning between the LLM and the pretrained CLIP model's Vision Encoder (CLIP-V) and Text Encoder (CLIP-T). "CL" denotes the contrastive learning loss, and "Cat" means concatenation. Items a–d correspond to the a–d in Section 3.2.

4.4 Ablation Study: Stage-2 *LLM2CLIP*

Adaptor Design. As shown in Table 8, we explore adaptor structures in the Stage-2 LLM2CLIP cross-modal pretraining. Comparing Linear ($\times$4) and Transformer ($\times$1), their performances are relatively similar, and thus, we select the simpler Linear adaptor structure. Furthermore, in our layer-wise ablation of Linear adaptors, we find that increasing the adaptor size indeed improves performance. Specifically, adaptor performance improves progressively when transitioning from no adaptor to 1, 2 and 4-layer.

Training-method ablation. We explored multiple ways to integrate an LLM during CLIP fine-tuning (implementation details appear in Table 9. Comparing experiments (a)–(b) and (b)–(c) shows that neither (i) optimising two separate losses nor (ii) explicitly aligning the two text encoders brings any meaningful benefit. *This observation motivates our choice to replace, rather than reuse, CLIP's original text encoder in the main pipeline.* Experiment (d)—which performs contrastive learning on the *concatenated* outputs of both encoders—does deliver a noticeable uplift; for the sake of simplicity, we do not adopt it as our default, but regard it as a promising future work.

5 Conclusion

We have revisited the problem of upgrading CLIP with large-language-model knowledge and proposed a *lightweight* pipeline that first turns an LLM into a discriminative caption-embedding module and then inserts it into CLIP via a cost-effective fine-tuning stage. With only *million-scale* training pairs and a compute budget essentially identical to standard CLIP fine-tuning, our method delivers sizeable gains—on top of already SOTA baselines—in image–text retrieval, image classification (zero-shot & linear probe), zero-shot/supervised detection/segmentation, and even as the visual encoder inside a multimodal LLM.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (NSFC Granted No. 62276190).

References

Bai, J.; Bai, S.; Yang, S.; Wang, S.; Tan, S.; Wang, P.; Lin, J.; Zhou, C.; and Zhou, J. 2023. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*.

BehnamGhader, P.; Adlakha, V.; Mosbach, M.; Bahdanau, D.; Chapados, N.; and Reddy, S. 2024. LLM2Vec: Large Language Models Are Secretly Powerful Text Encoders. *arXiv preprint*.

Chen, H.; Wang, L.; Yang, N.; Zhu, Y.; Zhao, Z.; Wei, F.; and Dou, Z. 2025. mme5: Improving multimodal multilingual embeddings via high-quality synthetic data. *arXiv preprint arXiv:2502.08468*.

Chen, L.; Li, J.; Dong, X.; Zhang, P.; He, C.; Wang, J.; Zhao, F.; and Lin, D. 2023. Sharegpt4v: Improving large multi-modal models with better captions. *arXiv preprint arXiv:2311.12793*.

Chen, Z.; Wu, J.; Wang, W.; Su, W.; Chen, G.; Xing, S.; Zhong, M.; Zhang, Q.; Zhu, X.; Lu, L.; et al. 2024. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 24185–24198.

Dai, W.; Li, J.; Li, D.; Meng Huat Tiong, A.; Zhao, J.; Wang, W.; Li, B.; Fung, P.; and Hoi, S. 2023. InstructBLIP: Towards General-purpose Vision-Language Models with Instruction Tuning. *arXiv e-prints*, arXiv:2305.06500.

Esser, P.; Kulal, S.; Blattmann, A.; Entezari, R.; Müller, J.; Saini, H.; Levi, Y.; Lorenz, D.; Sauer, A.; Boesel, F.; et al. 2024. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*.

Fan, L.; Krishnan, D.; Isola, P.; Katabi, D.; and Tian, Y. 2024. Improving clip training with language rewrites. *Advances in Neural Information Processing Systems*, 36.

Günther, M.; Ong, J.; Mohr, I.; Abdessalem, A.; Abel, T.; Akram, M. K.; Guzman, S.; Mastrapas, G.; Sturua, S.; Wang, B.; et al. 2023. Jina embeddings 2: 8192-token general-purpose text embeddings for long documents. *arXiv preprint arXiv:2310.19923*.

Jang, Y. K.; Kang, J.; Lee, Y. J.; and Kim, D. 2024. MATE: Meet At The Embedding–Connecting Images with Long Texts. *arXiv preprint arXiv:2407.09541*.

Jia, C.; Yang, Y.; Xia, Y.; Chen, Y.-T.; Parekh, Z.; Pham, H.; Le, Q.; Sung, Y.-H.; Li, Z.; and Duerig, T. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, 4904–4916. PMLR.

Jiang, Z.; Meng, R.; Yang, X.; Yavuz, S.; Zhou, Y.; and Chen, W. 2024. VLM2Vec: Training Vision-Language Models for Massive Multimodal Embedding Tasks. *ArXiv*, abs/2410.05160.

Kong, W.; Tian, Q.; Zhang, Z.; Min, R.; Dai, Z.; Zhou, J.; Xiong, J.; Li, X.; Wu, B.; Zhang, J.; et al. 2024. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*.

Koukounas, A.; Mastrapas, G.; Günther, M.; Wang, B.; Martens, S.; Mohr, I.; Sturua, S.; Akram, M. K.; Martínez, J. F.; Ognawala, S.; et al. 2024a. Jina CLIP: Your CLIP Model Is Also Your Text Retriever. *arXiv preprint arXiv:2405.20204*.

Koukounas, A.; Mastrapas, G.; Wang, B.; Akram, M. K.; Eslami, S.; Günther, M.; Mohr, I.; Sturua, S.; Martens, S.; Wang, N.; et al. 2024b. jina-clip-v2: Multilingual Multimodal Embeddings for Text and Images. *arXiv preprint arXiv:2412.08802*.

Lee, C.; Roy, R.; Xu, M.; Raiman, J.; Shoeybi, M.; Catanzaro, B.; and Ping, W. 2024. Nv-embed: Improved techniques for training llms as generalist embedding models. *arXiv preprint arXiv:2405.17428*.

Li, B.; Zhang, Y.; Guo, D.; Zhang, R.; Li, F.; Zhang, H.; Zhang, K.; Li, Y.; Liu, Z.; and Li, C. 2024a. LLaVA-OneVision: Easy Visual Task Transfer. *arXiv preprint arXiv:2408.03326*.

Li, J.; Li, D.; Xiong, C.; and Hoi, S. 2022. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, 12888–12900. PMLR.

Li, X.; Tu, H.; Hui, M.; Wang, Z.; Zhao, B.; Xiao, J.; Ren, S.; Mei, J.; Liu, Q.; Zheng, H.; et al. 2024b. What If We Recaption Billions of Web Images with LLaMA-3? *arXiv preprint arXiv:2406.08478*.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 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*, 740–755. Springer.

Liu, H.; Li, C.; Li, Y.; and Lee, Y. J. 2023a. Improved Baselines with Visual Instruction Tuning.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2023b. Visual Instruction Tuning.

Merity, S.; Xiong, C.; Bradbury, J.; and Socher, R. 2016. Pointer Sentinel Mixture Models. *arXiv: Computation and Language,arXiv: Computation and Language*.

Muennighoff, N.; Tazi, N.; Magne, L.; and Reimers, N. 2022. Mteb: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*.

Onoe, Y.; Rane, S.; Berger, Z.; Bitton, Y.; Cho, J.; Garg, R.; Ku, A.; Parekh, Z.; Pont-Tuset, J.; Tanzer, G.; Wang, S.; and Baldridge, J. 2024. DOCCI: Descriptions of Connected and Contrasting Images. arXiv:2404.19753.

Ordonez, V.; Kulkarni, G.; and Berg, T. 2011. Im2Text: Describing Images Using 1 Million Captioned Photographs. In Shawe-Taylor, J.; Zemel, R.; Bartlett, P.; Pereira, F.; and Weinberger, K., eds., *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PMLR.

Springer, J. M.; Kotha, S.; Fried, D.; Neubig, G.; and Raghunathan, A. 2024. Repetition Improves Language Model Embeddings. *arXiv preprint arXiv:2402.15449*.

Sun, Q.; Fang, Y.; Wu, L.; Wang, X.; and Cao, Y. 2023. EVA-CLIP: Improved Training Techniques for CLIP at Scale. *arXiv e-prints*, arXiv:2303.15389.

Sun, Q.; Fang, Y.; Wu, L.; Wang, X.; and Cao, Y. 2023. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*.

Tschannen, M.; Gritsenko, A.; Wang, X.; Naeem, M. F.; Alabdulmohsin, I.; Parthasarathy, N.; Evans, T.; Beyer, L.; Xia, Y.; Mustafa, B.; et al. 2025. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*.

Urbanek, J.; Bordes, F.; Astolfi, P.; Williamson, M.; Sharma, V.; and Romero-Soriano, A. 2024. A picture is worth more than 77 text tokens: Evaluating clip-style models on dense captions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 26700–26709.

Vouitsis, N.; Liu, Z.; Gorti, S. K.; Villecroze, V.; Cresswell, J. C.; Yu, G.; Loaiza-Ganem, G.; and Volkovs, M. 2023. Data-Efficient Multimodal Fusion on a Single GPU. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 27229–27241.

Xu, H.; Xie, S.; Tan, X. E.; Huang, P.-Y.; Howes, R.; Sharma, V.; Li, S.-W.; Ghosh, G.; Zettlemoyer, L.; and Feichtenhofer, C. 2023. Demystifying clip data. *arXiv preprint arXiv:2309.16671*.

Young, P.; Lai, A.; Hodosh, M.; and Hockenmaier, J. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2: 67–78.

Zhai, X.; Mustafa, B.; Kolesnikov, A.; and Beyer, L. 2023. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, 11975–11986.

Zhang, B.; Zhang, P.; Dong, X.; Zang, Y.; and Wang, J. 2024a. Long-clip: Unlocking the long-text capability of clip. *arXiv preprint arXiv:2403.15378*.

Zhang, J.; Qu, X.; Zhu, T.; and Cheng, Y. 2024b. Clip-moe: Towards building mixture of experts for clip with diversified multiplet upcycling. *arXiv preprint arXiv:2409.19291*.

Zhang, L.; Yang, Q.; and Agrawal, A. 2025. Assessing and Learning Alignment of Unimodal Vision and Language Models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 14604–14614.

Zheng, K.; Zhang, Y.; Wu, W.; Lu, F.; Ma, S.; Jin, X.; Chen, W.; and Shen, Y. 2024. DreamLIP: Language-Image Pre-training with Long Captions. *ArXiv*, abs/2403.17007.