Preserving Language Capabilities in Vision-Language Models via Representation Regulation

Zixuan Chen[1,2], Juncheng Tao[3], Ziqian Zeng*[1],

[1]South China University of Technology, China, [2]StreamingTelligence Lab, China,
[3]Carnegie Mellon University, USA

zixuanindexerror@gmail.com, junchent@andrew.cmu.edu, zqzeng@scut.edu.cn

Abstract

Vision-Language Models (VLMs) provide a unified framework to process both text-only tasks and vision-language tasks. However, finetuning VLMs on vision-language data has degraded language capabilities. In this paper, we prove that as the training loss declines during finetuning, the visual representation and textual representation move closer to each other, a phenomenon we term "representation mixing." We prove that the representation mixing occurring within the post-representation layers causes the degradation of language capabilities. Post-representation layers refer to the first few layers in LLMs that are involved in representation learning. To preserve the language capabilities, we propose the Representation Regulation for VLM Training (RRVLM), which introduces a Representation Distribution Difference (RDD) loss to reduce the distance between these representations. Extensive experiments on various benchmarks and VLM frameworks show that our method can effectively preserve the language capabilities and achieve superior vision-language performance.

1 Introduction

Vision-Language Models (VLMs) (Bai et al., 2025; Dubey et al., 2024) offer unprecedented capabilities to process and understand vision-language data. Built upon powerful Large Language Model (LLM) (Brown et al., 2020) architectures, these models seamlessly integrate visual and textual inputs, enabling sophisticated reasoning and interaction across diverse input types. Recently, industry shows an increasing preference for VLMs (Peng et al., 2023; Anil et al., 2023; Zhu et al., 2025) that provide a unified framework that excels in both text-only and vision-language tasks. VLMs provide users with a seamless experience, as they can accomplish various tasks through a single, versatile system.

* Corresponding author.

However, improving VLMs on vision-language tasks has unintentionally compromised their capabilities on text-only tasks. Empirical evidence from NVLM (Dai et al., 2024) reveals significant performance declines across leading models. VILA-1.5 40B (Lin et al., 2024) experiences a 6.9 points drop in text-only tasks, while LLaVA-OneVision 72B (Liu et al., 2024) and InternVL-2-Llama3-76B (Chen et al., 2023, 2024) show comparable decreases of 6.3 and 6.9 points, respectively.

Existing methods address this challenge through freezing partial LLM backbones, increasing model size, or incorporating additional high-quality text-only data. Llama 3-V (Dubey et al., 2024) froze the LLM backbone during the vision-language instruction-tuning (VLIT) stage to preserve the model's language capabilities. However, other studies (Dai et al., 2024; Alayrac et al., 2022) report degraded performance on vision-language tasks with this approach. NVLM (Dai et al., 2024) takes an alternative approach by scaling up model size while maintaining frozen LLM layers. Recent advanced VLMs such as DeepSeek-VL (Lu et al., 2024), MM1 (McKinzie et al., 2024), Bunny (He et al., 2024), and NVLM (Dai et al., 2024) have been designed with intricate combinations of text-only and vision-language data for the VLIT stage. While promising, this method introduces significant computational and data overhead. Notably, current approaches focus on workarounds rather than investigating the root cause of degraded language capabilities.

We reveal the root cause of declined language capabilities. Since the LLM backbone is only modified during the VLIT stage, the root cause of language capability degradation originates in this stage. Our analysis in § 4.1 reveals that as the training loss of the VLIT stage decreases, the distance between textual and visual representations will also decrease. We term the phenomenon of reduced distance as representation mixing. Furthermore, our

Findings of the Association for Computational Linguistics: ACL 2026, pages 24189–24205

findings in § 4.2 indicate that the representation mixing within the post-representation layers is a critical factor contributing to the diminished language capabilities. The post-representation layers are the first few layers of the LLM backbone involved in representation learning.

To mitigate the degradation of language capabilities, we propose Representation Regulation for VLM training (RRVLM), a method integrated into the VLIT stage of VLM training. Specifically, we introduce a Representation Distribution Difference (RDD) loss to reduce the average distance between textual and visual embeddings which are the inputs of the post-representation layers. Our method not only preserves the language capability but also enhances vision-language capabilities for the following reasons. First, by shifting representation mixing to the earlier layers before the post-representation layers, it alleviates the representation mixing within the post-representation layers, thus alleviating the language degradation. Second, we prove that minimizing the RDD loss can reduce the distance between two representations, consequently decreasing empirical risk during the VLIT stage, thus enhancing vision-language capabilities. Compared to previous methods, our method eliminates the need for architectural modifications, making it broadly applicable to most VLMs at a low cost.

The contributions are summarized as follows,

- Our theoretical analysis identified the cause of the decline in language capabilities in VLMs.
- We propose representation regulation (RRVLM) which is integrated into the VLIT stage of VLM training. Within the RRVLM framework, we propose Representation Distribution Difference loss to reduce the distance between visual and textual representations.
- Extensive experimental results show that RRVLM preserves the language capabilities of VLMs and enhances vision-language capabilities.

2 Related Work

2.1 Mitigating Language Degradation in VLMs

A recent study (Dai et al., 2024) reveals significant performance declines across leading models include VILA-1.5 40B (Lin et al., 2024), LLaVA-OneVision 72B (Liu et al., 2024), and InternVL-2-Llama3-76B (Chen et al., 2023, 2024). There are three categories of methods for mitigating language degradation.

The **first** category preserves language capabilities by freezing partial LLM parameters. NVLM (Dai et al., 2024) froze the LLM parameters and trained only the cross-attention layers in their proposed NVLM-X framework. Attention-Tuning (Dubey et al., 2024) freezes the LLM backbone while making only the attention layers tunable. Studies in (Srivastava et al., 2024) demonstrate that using LoRA (Hu et al., 2022), which maintains fixed LLM parameters while keeping adaptation matrices trainable, helps alleviate language degradation. SPIDER (Huang et al., 2024) selectively updates parameters based on their importance. The **second** category augments vision-language data with additional text data during the VLIT stage. Recent advanced VLMs such as DeepSeek-VL (Lu et al., 2024), MM1 (McKinzie et al., 2024), Bunny (He et al., 2024), and NVLM (Dai et al., 2024) have designed intricate combinations of text-only and vision-language data for the VLIT stage. The **third** category encompasses diverse approaches to address language degradation from various perspectives. ModelMerge (Ratzlaff et al., 2024) posits that VLMs experience catastrophic forgetting after the VLIT stage. It merged the base LLM parameters back into language model parameters of the VLM after the VLIT stage. SoftLabel (Harun and Kanan, 2024) formulated degraded language capabilities as a catastrophic forgetting problem in continuous learning. It hypothesized that increased training loss when learning new tasks causes forgetting, and addressed this by reducing initial training losses through smoothed labels rather than hard labels. WINGS (Zhang et al., 2024) observed that the attention shift is related to language degradation and proposed to add a trainable module into the original attention layers to compensate for the attention shift.

3 Preliminary

3.1 Physical VLM Architecture

VLM typically consists of four components: the text embedding layer, a vision encoder, the vision projection layer, and the LLM decoding layers. **(1) The text embedding layer** (p_t) refers to the embedding layer of the LLM backbone. It generates

text embeddings given a sequence of text tokens. **(2) The vision embedding generator** (p_v) consists of two sequential components, namely, a vision encoder followed by a vision projection layer. **The vision encoder** extracts visual features from raw visual inputs. It is often pre-trained on large-scale image datasets such as WIT (Radford et al., 2021), enabling robust visual feature extraction. CLIP (Radford et al., 2021) and BEiT (Bao et al., 2022) are widely used vision encoders in the VLM. **The vision projection layer** serves as the critical bridge between visual and language modalities by transforming visual features into the same embedding space as language modality. The inputs of vision projection layer are visual features generated by the vision encoder. The outputs of vision projection layer are termed vision embeddings. The vision projection layer can be a simple linear layer (Liu et al., 2023) or more capable Transformer blocks (Awadalla et al., 2023; Dai et al., 2023). **(3) The LLM decoding layers** (g) refers to the LLM backbone excluding the text embedding layer. It takes the concatenation of text embeddings and vision embeddings as inputs and generates textual outputs.

VLMs process vision-language inputs by augmenting the LLM backbone with a vision embedding generator to process visual inputs. Textual inputs pass through a text embedding layer, while visual inputs are transformed by the vision embedding generator. The resulting textual and visual embeddings are then jointly processed by the LLM decoding layers, enabling reasoning across both modalities.

3.2 Abstract VLM Architecture

From a functional perspective, a VLM can be decomposed into distinct components dedicated to representation learning and prediction. We formalize the VLM from this perspective as follows. The raw textual input x and raw visual input y are mapped into a unified representation space by the **text representator** h_t and **vision representator** h_v, respectively. The output of text representator $h_t(x) \in X$ and the output of vision representator $h_v(y) \in Y$ are concatenated and fed into the **predictor** f, generating the final textual output. X denotes the set of all possible textual representations generated by the text representator given any input. Y denotes the set of all possible visual representations generated by the vision representator given any input.

We map abstract VLM components to their physical implementations. Since the text embedding layer serves as the first component to process raw textual input, the text representator h_t component includes at least the text embedding layer p_t. Similarly, the vision representator h_v includes at least the vision embedding generator p_v. In the simplest case, h_t equals the text embedding layer p_t, h_v equals the vision embedding generator p_v, and the predictor f corresponds directly to the LLM decoding layers g.

However, some studies (Schwettmann et al., 2023; Neo et al., 2024) suggest that initial LLM layers continue sophisticated representation learning for both textual and visual inputs. Hence, the text representator h_t and vision representator h_v likely include the first few layers of LLM decoding layers, which we term **post-representation layers** (r). The remaining LLM layers serve as the predictor f. The inputs of post-representation layers are visual and textual embeddings. Post-representation layers further transform the visual and textual embeddings ($p_v(y)$ and $p_t(x)$) into more sophisticated representations. The outputs of post-representation layers are termed as visual representation and textual representation. Notably, throughout this paper, **we maintain a clear distinction between visual (textual) embedding and visual (textual) representation**. Fig.1 (b) illustrates this correspondence between physical implementations and abstract components.

3.3 VLM Training

The VLM training consists of two stages. The first stage is **Vision-Language Pre-training**, which aims to align representations of visual and language modalities. In this stage, the LLM backbone is frozen. This indicates that there is no decline in the LLM's language capabilities during the first training stage. The second stage is **Vision-Language Instruction-tuning (VLIT)**. In this stage, many components including the LLM backbone are all trainable, which introduces potential degradation in language capabilities. Consequently, our analysis primarily focuses on the second training stage.

The VLIT dataset S is defined as $S = \{(x_1, y_1, z_1), \ldots, (x_n, y_n, z_n)\}$ where n is number of samples in the dataset, and y represents raw visual input, x represents the raw textual input, z represents the textual ground-truth. Tasks in this stage include visual question answering (Antol et al., 2015) and image captioning (Schuhmann et al., 2022). For example, in the visual question

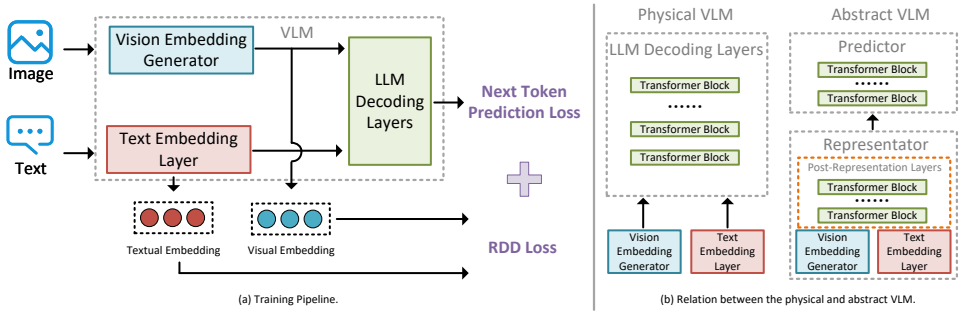


Figure 1: (a) Training Pipeline. First, raw visual and textual inputs are processed through the vision embedding generator and text embedding layer, respectively, producing visual and textual embeddings. These embeddings are fed into the LLM decoding layers to calculate the next token prediction loss. Additionally, the embeddings are utilized to compute the RDD loss. Finally, two objective functions are optimized jointly. (b) Relation between the physical and abstract VLM. The initial LLM layers might continue sophisticated representation learning for both textual and visual inputs. The representator likely includes the first few layers of LLM decoding layers, which we term post-representation layers (r).

task, x is the question, y is an image, and z is the caption of the given image.

In the VLIT stage, both the vision representator h_v, text representator h_t, and predictor f are trainable. The empirical risk for this stage can be formulated as

$$\mathcal{L}_{VLIT}\left(h_t, h_v, f\right) = \frac{1}{n}\sum_{i=1}^{n} \ell\Big(f(h_t(x_i), h_v(y_i)), z_i\Big), \tag{1}$$

where $\ell(\cdot)$ is a specific loss function, e.g., cross-entropy.

VLMs can not only handle vision-language tasks but also text-only tasks. We can use the evaluation loss to measure the performance on the text-only tasks. The evaluation loss on text-only tasks can be formulated as

$$\mathcal{L}_{text}\left(h_t, f\right) = \frac{1}{m}\sum_{i=1}^{m} \ell\left(f(h_t(x_i')), z_i'\right), \tag{2}$$

where $S' = \{(x_1', z_1'), \ldots, (x_m', z_m')\}$ where m is number of samples in the test set S', x' represents textual input, z' represents textual ground-truth.

4 Theoretical Analysis

Since the LLM backbone is only modified during the VLIT stage, the cause of language capability degradation must originate in this stage. Hence, we analyze the training dynamics of the this stage. In §4.1, we find that as the training loss declines, the visual representation and textual representation move closer to each other, a phenomenon we term "representation mixing." In §4.2, we prove that the representation mixing occurring within the post-representation layers is the cause of the degradation of language capabilities.

4.1 Training Dynamics of VLM

We first analyze what happens as the loss function decreases in the VLIT stage by deriving the lower bound of the training loss.

Theorem 1 *Assume that the loss ℓ increases noticeably when the model prediction deviates from the oracle prediction, and that the model prediction changes noticeably when the visual representation deviates from the optimal visual representation; both assumptions are formally stated in Appendix A. Then for any $\delta > 0$, with probability at least $1-\delta$ over the draw of the sample S of size n, we have*

$$\begin{aligned}\mathcal{L}_{\text{VLIT}}\left(h_t, h_v, f\right) \geq\ & \frac{m\alpha^2}{4}\mathbb{E}_S\left[\left\|h_t(x) - h_v(y)\right\|\right]^2 \\ & - m\alpha^2 R^2\left(\frac{\sqrt{2\pi}}{n}G(\Phi(S_{xy})) + \sqrt{\frac{9\log(2/\delta)}{n}}\right) \\ & - \frac{m\alpha^2}{2}\beta^2.\end{aligned} \tag{3}$$

Here $\mathbb{E}_S[\cdot]$ denotes the population expectation with respect to the data distribution over the VLIT dataset S, and $\mathcal{L}_{\text{VLIT}}(h_t, h_v, f)$ is the empirical VLIT loss defined in Eq. (1). Other notations such as S, x_i, y_i, z_i and n follow the definitions in the preliminaries. $G(\Phi(S_{xy}))$ is defined in Appendix A. The π, m, α, β and R are constants.

Due to space limitation, the proof is presented in Appendix A.

Theorem 1 shows the training loss of the VLIT stage has a lower bound consisting of three terms. As the loss decreases, these three terms must collectively decrease. However, as the number of training samples n is typically large, the second terms approach zero and the last term is constant. Therefore, the declining loss primarily leads to a reduction in the first term. It indicates that during the VLIT stage, the average distance between visual and textual representations decreases. We term this phenomenon as **Representation Mixing**. In the VLIT stage, with all components of the VLM being trainable, the visual representation can move closer to the textual representation, and the textual representation can also shift toward the visual representation. Consequently, we can further conclude that during this training process, both representations move closer to each other.

4.2 Degradation of Language Capability

We investigate the root cause of language capability degradation by establishing a lower bound for the difference between evaluation losses on text-only tasks measured before and after VLIT stage.

Theorem 2 *Let b, a denote the VLM before and after vision-language instruction-tuning (VLIT) respectively. We have:*

$$\mathcal{L}^a_{\text{text}} - \mathcal{L}^b_{\text{text}} \geq \frac{\mathcal{D}^b - \mathcal{D}^a - \eta_v}{R} \Delta_{\text{vis}} - \varepsilon_{\text{corr}} - \varepsilon_{\text{adapt}}. \tag{4}$$

where $\mathcal{D}^a = \mathbb{E}_S\left[\left\|h^a_t(x) - h^a_v(y)\right\|\right]$, $\mathcal{L}^a_{text} = \frac{1}{m}\sum_{i=1}^{m} \ell\left(f^a(h^a_t(x_i)), z_i\right)$ is the evaluation loss of VLM a on text-only tasks, x_i is textual input, z_i is textual ground-truth. f^a and h^a_t are the predictor, and text representator of VLM a respectively. For VLM b, we use analogous notations including $\mathcal{D}^b$, $\mathcal{L}^b_{text}$, f^b, and h^b_t, where b replaces a in the subscripts. Δ_{vis}, $\varepsilon_{\text{corr}}$, η_v and $\varepsilon_{\text{adapt}}$ are constants. R is a positive constant consistent with Theorem 1.

Due to space limitation, the proof is shown in Appendix B. Theorem 2 shows that representation mixing is the cause of the degradation of language capability. Theorem 1 states that as training progresses, the distance $\mathbb{E}_S\left[\|h_t(x) - h_v(y)\|\right]$ gradually decreases. The representation distance in VLM a is smaller than that in VLM b, which means $\mathcal{D}^b - \mathcal{D}^a \geq 0$. Given that Δ_{vis} is a large positive value while η_v, $\varepsilon_{\text{corr}}$ and $\varepsilon_{\text{adapt}}$ are small positive constants, the right-hand side of Eq. (4) is likely to be positive. It means the evaluation loss on text-only task increases after VLIT stage, i.e., $\mathcal{L}^a_{\text{text}} \geq \mathcal{L}^b_{\text{text}}$. This suggests that a degradation in language capability is theoretically guaranteed, driven by representation mixing (i.e., $\mathcal{D}^b - \mathcal{D}^a \geq 0$).

To identify the specific components responsible for performance degradation, we analyze where representation mixing occurs. We substitute the abstract components h_t and h_v in the distance definition $\mathcal{D}$ with physical components: $h_t(x) = r(p_t(x))$ and $h_v(y) = r(p_v(y))$, where r denotes the post-representation layers, p_t is the text embedding layer, p_v is the vision embedding generator. Only r, p_t and p_v are involved in the distance computation. During inference on text-only tasks, only three components are active, i.e., the text projector p_t, the post-representation layers r, and the predictor f. By intersecting the components involved in distance computation with those active during inference, we narrow our investigation to p_t and r. Since p_t is a simple embedding layer that remains largely stable during training, the post-representation layers r are the primary drivers of representation mixing. Consequently, the degradation of language capability stems from the representation mixing occurring within post-representation layers.

5 Method

5.1 Motivation

Theorem 2 reveals that representation mixing within the post-representation layers causes language capability degradation. Increasing the distance between two representations might seem like a straightforward solution. However, Theorem 1 reveals that it is necessary to reduce the distance between two representations to improve vision-language capabilities because the empirical risk during the VLIT stage is lower bounded by the distance between two representations. Also, identifying which layers in the LLM backbone correspond to the post-representation layers is non-trivial, because the LLM backbone itself has a complex architecture.

Hence, an effective solution must (1) maintain sufficiently small distances between textual and visual representations for strong vision-language capabilities, (2) alleviate representation mixing within post-representation layers, and (3) achieve the above two objectives without assuming that we know which layers in the LLM backbone correspond to the post-representation layers.

These three objectives can be achieved simultaneously. Note that the post-representation layers

operate on both vision and text embeddings and perform representation mixing on top of them. If there is already a large discrepancy between the vision and text embeddings from the beginning, the post-representation layers must expend significantly more effort to carry out effective representation mixing. Therefore, a simple yet effective strategy is to reduce the distance between vision embeddings and text embeddings. The vision embeddings and text embeddings are the output of p_v and p_t, respectively.

By facilitating distance reduction before the post-representation layers, our method can alleviate mixing within the post-representation layers. While our approach does not eliminate representation mixing within the post-representation layers entirely, it shifts this process to earlier stages, relieving the burden of the post-representation layers, thus maintaining language capabilities.

5.2 Representation Distribution Difference Loss

Specifically, we propose a Representation Regulation method for VLM (**RRVLM**), a novel method integrated into the VLIT stage. This approach introduces a Representation Distribution Difference (RDD) loss to minimize the average distance between textual and visual embeddings prior to the post-representation layers, mitigating the degradation of the language capabilities.

Our goal is to reduce the distribution discrepancy between textual and visual embeddings across the entire dataset. However, during training, the loss is applied per mini-batch, and thus we need a batch-computable surrogate that measures distribution difference reliably. According to (Gretton et al., 2012), kernel methods provide an effective way to measure the discrepancy between two distributions via their embeddings in a reproducing kernel Hilbert space (RKHS).

We introduce a kernel function $\phi(\cdot)$ to project the text embedding $p_t(x_i)$ and visual embedding $p_v(y_i)$ into the reproducing kernel hilbert Space $\mathcal{H}$ for distance computation. The approximated distance is formulated as:

$$\mathcal{L}_{rdd} := \left\| \mathbb{E}_x\left[\phi(p_t(x))\right] - \mathbb{E}_y\left[\phi(p_v(y))\right] \right\|_{\mathcal{H}}. \tag{5}$$

The distance in expectation form would be written in the discrete form:

$$\begin{aligned} &\left\| \mathbb{E}_x\left[\phi(p_t(x))\right] - \mathbb{E}_y\left[\phi(p_v(y))\right] \right\|_{\mathcal{H}} \\ =&\frac{1}{n^2}\left\| \sum_{i=1}^{n}\sum_{i'=1}^{n}\Big[\phi\big(p_t(x_i), p_t(x_{i'})\big) \right. \\ &\left. - 2\phi\big(p_t(x_i), p_v(y_i)\big) + \phi\big(p_v(y_i), p_v(y_{i'})\big)\Big] \right\|. \end{aligned} \tag{6}$$

We minimize $\mathcal{L}_{rdd}$ during optimization.

5.3 Training

As shown in Fig. 1 (a), raw visual and textual inputs are processed through the vision embedding generator and text embedding layer, respectively, producing visual and textual embeddings. These embeddings are fed into the LLM decoding layers to calculate the next token prediction loss. Additionally, the embeddings are utilized to compute the RDD loss. Finally, two objective functions Eq. 1 and Eq. 6 are optimized jointly.

6 Experiment

6.1 Experimental Settings

Benchmarks. To evaluate the **vision-language capabilities**, benchmarks are MMMU (Yue et al., 2024), MME Perception (MME_P) (Fu et al., 2023), MME Cognition (MME_C) (Fu et al., 2023), and POPE (Li et al., 2023). To evaluate **language capabilities**, the benchmarks are MMLU (Hendrycks et al., 2021), ARC_E (Clark et al., 2018), ARC_C (Clark et al., 2018), and OpenBookQA (O.B.QA) (Mihaylov et al., 2018).

Compared Methods. We compared our method with the following approaches. **Attention-Tuning**, **LoRA** (Hu et al., 2022), **SoftLabel** (Harun and Kanan, 2024), **ModelMerge** (Ratzlaff et al., 2024), and **DSVL Mixing**. Due to space limitation, we provide detailed information in Appendix C.

Settings. Since DSVL Mixing uses additional text-only data while other methods do not, to ensure fair comparison, **we establish two settings, namely, with and without additional text-only data in the experiments.** In the setting that includes additional text-only data, all methods were trained using identical data combinations as DSVL Mixing. To ensure fair comparison, we carefully controlled the total training volume across both settings. In the setting that incorporates additional text-only data, we systematically reduced the vision-language samples by an equivalent amount. This ensures that any

Table 1: Results of methods **without** additional text-only data. Best results are marked in **bold**. The results highlighted in **gray** indicate the baseline for comparison; those marked in **green** denote performance degradation relative to the baseline, while those marked in **red** indicate improvements over the baseline.

Methods	Language Capability					Vision-Language Capability			
	MMLU↑	ARC_E ↑	ARC_C ↑	O.B.QA ↑	Avg↑	MMMU ↑	MME_P ↑	MME_C ↑	POPE ↑
Phi2 + EVACLIP									
LLM (Phi2)	54.46	80.05	53.24	**40.20**	56.99	-	-	-	-
Full-Tuning	51.46	78.20	48.29	35.80	53.44	37.30	1307.50	247.14	85.08
Attention-Tuning	54.99	79.76	52.22	37.80	56.19	33.40	1298.09	278.57	81.94
LoRA	54.69	80.77	52.30	37.60	56.34	34.90	1247.29	268.92	81.59
SoftLabel	54.34	79.50	49.40	37.00	55.06	36.10	1308.73	270.00	83.87
ModelMerge	54.12	80.22	50.26	37.80	55.60	36.20	1308.97	248.57	85.25
RRVLM (our)	**55.81**	**82.28**	**53.41**	39.80	**57.83**	**37.40**	**1326.07**	**293.57**	**85.32**
Phi2 + SigLIP									
LLM (Phi2)	54.46	80.05	53.24	40.20	56.99	-	-	-	-
Full-Tuning	51.12	78.16	47.78	36.60	53.42	36.80	1382.50	308.92	83.92
Attention-Tuning	54.50	79.88	51.45	38.40	56.06	35.40	1325.10	302.85	83.17
LoRA	54.30	80.35	51.71	38.60	56.24	33.60	1292.35	245.71	83.39
SoftLabel	53.33	78.32	49.15	37.20	54.50	36.30	1386.80	306.42	83.75
ModelMerge	53.14	79.67	50.09	37.80	55.18	36.70	1363.26	301.42	83.70
RRVLM (our)	**54.59**	**82.32**	**53.84**	**40.60**	**57.74**	**37.30**	**1390.28**	**311.42**	**84.35**

Table 2: Results of methods **with** additional text-only data. Best results are marked in **bold**. The results highlighted in **gray** indicate the baseline for comparison; those marked in **green** denote performance degradation relative to the baseline, while those marked in **red** indicate improvements over the baseline.

Methods	Language Capability					Vision-Language Capability			
	MMLU↑	ARC_E ↑	ARC_C ↑	O.B.QA ↑	Avg↑	MMMU ↑	MME_P ↑	MME_C ↑	POPE ↑
Phi2 + EVACLIP									
LLM (Phi2)	54.46	80.05	53.24	40.20	56.99	-	-	-	-
DSVL Mixing	52.36	79.84	49.15	39.00	55.08	35.20	1258.97	245.00	84.59
Attention-Tuning	54.85	79.80	51.37	39.00	56.26	35.10	1253.91	277.50	82.43
LoRA	54.21	80.22	52.39	39.40	56.56	35.60	1236.69	284.28	81.37
SoftLabel	54.72	79.08	50.68	37.40	55.47	36.40	1239.73	257.50	84.07
ModelMerge	54.73	79.59	51.54	38.20	56.02	35.80	**1271.07**	243.57	84.69
RRVLM (our)	**55.15**	**84.01**	**54.78**	**40.40**	**58.59**	**36.90**	1267.53	**337.14**	**84.97**
Phi2 + SigLIP									
LLM (Phi2)	54.46	80.05	53.24	40.20	56.99	-	-	-	-
DSVL Mixing	53.57	79.63	48.12	38.00	54.83	35.80	1265.70	283.92	83.90
Attention-Tuning	54.63	79.88	52.65	39.40	56.64	35.70	1275.29	320.35	82.80
LoRA	55.26	80.09	52.39	39.20	56.74	**37.10**	1314.97	260.71	82.61
SoftLabel	54.17	78.75	49.83	37.20	54.99	**37.10**	1308.47	308.92	83.91
ModelMerge	54.38	79.92	51.17	39.20	56.17	36.00	1273.64	295.00	83.61
RRVLM (our)	**55.70**	**83.71**	**54.44**	**40.58**	**58.38**	**37.10**	**1333.20**	**344.64**	**84.16**

performance differences observed between methods reflect their algorithmic capabilities rather than variations in overall training data quantity.

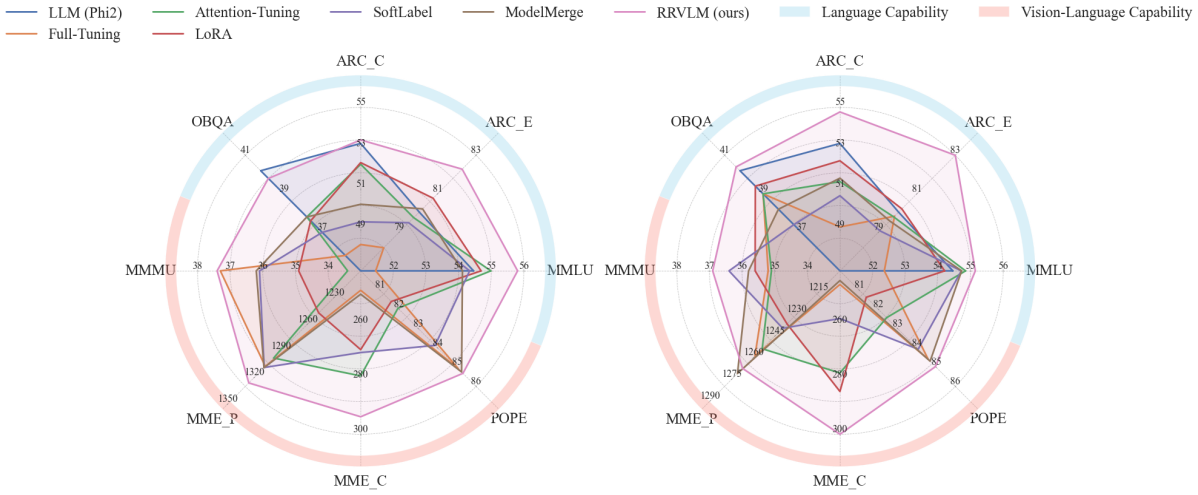
VLM Framework and Backbones. Our experiments are conducted across three prominent multimodal frameworks including **Bunny** (He et al., 2024), **LLaVA** (Liu et al., 2023), and **NVILA** (Liu et al., 2025). The details of backbones within each framework are shown in Appendix D.

Implementation Details. The implementation details, including training datasets and training settings, are provided in Appendix E.

6.2 Main Results

As shown in Tab. 1 and Tab. 2, our method preserves language capabilities while achieving superior performance on vision-language tasks across both settings (with and without additional text-only data). While other baselines have language capability degradation. Moreover, our method performs well on vision-language tasks. This demonstrates our method's ability to excel in both domains without compromising either.

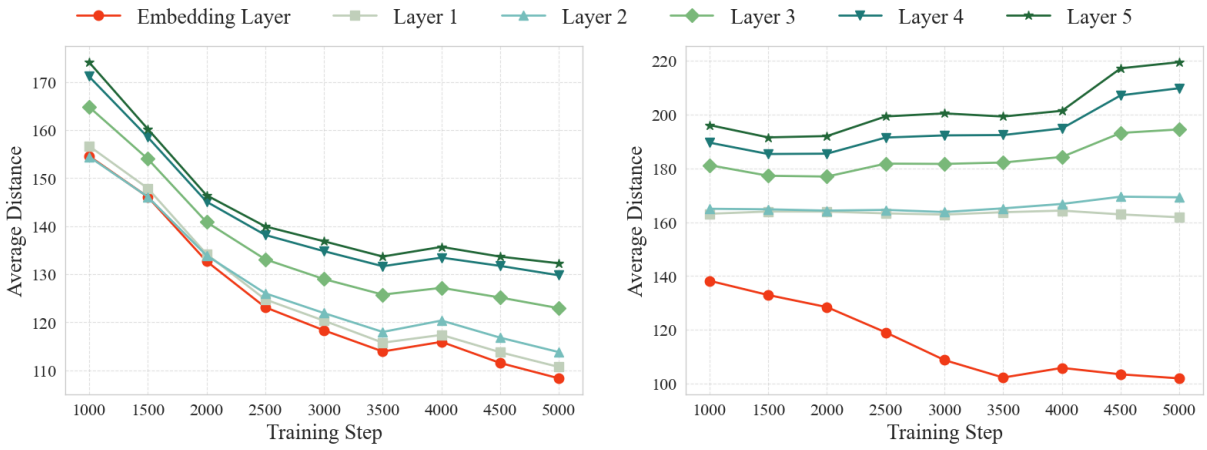
Furthermore, when additional text-only data is incorporated, all methods show improved performance in language capability evaluations, highlighting the usefulness of text-only data for preserving language capabilities. However, with the inclusion of text-only data, the performance of all methods on vision-language benchmarks declines, except on the MME_C benchmark (Fu et al., 2023). This indicates that data mixing may negatively impact vision-language capabilities. The MME_C benchmark includes common-sense reasoning, numerical calculation, text translation, and code reasoning tasks. Text-only data is particularly beneficial for the latter three tasks, which explains why performance on MME_C improves when text-only



(a) Phi2 + EVACLIP without additional text-only data.

(b) Phi2 + EVACLIP with additional text-only data.

Figure 2: Comparison between different methods across different benchmarks under the setting of without/with additional text-only data.



(a) Without RDD loss.

(b) With RDD loss.

Figure 3: (a) and (b) show the trend of average distance between textual and visual representation at different layers under the settings of without and with RDD loss.

data is used.

Fig. 2 demonstrates that our method encompasses a substantially larger area, confirming its comprehensive superiority in both preserving language capabilities and enhancing vision-language performance.

Additional results of more VLM backbones, presented in Appendix F, further corroborate the above findings. Further case study in Appendix H examines our method's performance on text-only tasks, confirming its ability to preserve language capabilities.

6.3 Ablation Study on RDD Loss

In this section, we present an ablation study on RDD loss. We compare RRVLM trained with RDD loss against RRVLM using Euclidean distance regulation loss (L2 loss). The Euclidean distance regulation loss aims to minimize the Euclidean distance between visual and textual embeddings, serving as a straightforward implementation of the objective proposed in our motivation section. As shown in the Tab. 3, RRVLM using the RDD loss best preserves language capabilities while achieving superior performance on vision-language tasks, demonstrating the effectiveness of RDD loss. On the other hand, RRVLM trained with L2 loss also retains language capabilities to some extent. Since the L2 loss can be regarded as a coarse approximation of our theoretically derived objective, this further provides experimental validation for the soundness of our theoretical analysis and core motivation.

Methods	Language Capability					Vision-Language Capability			
	MMLU↑	ARC_E ↑	ARC_C ↑	O.B.QA ↑	Avg↑	MMMU ↑	MME_P ↑	MME_C ↑	POPE ↑
Phi2 + EVACLIP									
LLM (Phi2)	54.46	80.05	53.24	**40.20**	56.99	-	-	-	-
Full-Tuning	51.46	78.20	48.29	35.80	53.44	37.30	1307.50	247.14	85.08
RRVLM$_{Euclidean}$	54.25	79.88	50.09	38.20	55.61	35.80	1319.11	273.57	84.47
RRVLM (our)	**55.81**	**82.28**	**53.41**	39.80	**57.83**	**37.40**	**1326.07**	**293.57**	**85.32**

Table 3: Results of methods on the ablation study of RDD loss. Best results are marked in **bold**.

Kernel Type			Bandwidth		
Method	Language	Vision-Language	Method	Language	Vision-Language
LLM (Phi2)	54.46	-	LLM (Phi2)	54.46	-
Full-Tuning	51.46	37.30	Full-Tuning	51.46	37.30
RRVLM (Linear Kernel)	54.88	36.20	RRVLM (0.5× Mean Heuristic)	54.55	36.80
RRVLM (Polynomial Kernel)	55.03	36.90	RRVLM (2× Mean Heuristic)	55.09	37.15
RRVLM (RBF Kernel)	55.81	37.40	RRVLM (Mean Heuristic)	55.81	37.40

Table 4: Ablation Study on Kernel Type and Bandwidth.

6.4 Representation Mixing

To investigate the phenomenon of Representation Mixing during the VLIT stage and the effectiveness of our method in mitigating the degradation in language capabilities, we present the average distances between visual and textual representations in different layers across different training steps. The settings are described in Appendix G.

In Fig. 3 (a), without RDD loss, the distance between textual and visual representations decreases during the VLIT stage, supporting Theorem 1. In Fig. 3 (b), when applying RDD loss, the distance in the embedding layer decreases, confirming the effectiveness of the RDD loss. Meanwhile, the distance in the first few layers increases as the training process progresses, effectively mitigating representation mixing within the first few layers and thus preserving language capabilities.

6.5 Hyperparameter Analysis

We provide an analysis on the kernel type and bandwidth selection strategies in the Table 4. The experiment setting is the same with the ablation study in Section 6.3. The strength of regularization can be controlled by selecting different kernel types and bandwidth strategies. The linear kernel provides more robust regularization, while the polynomial kernel imposes a more sensitive regularization. A smaller bandwidth yields stronger and more concentrated regularization, whereas a larger bandwidth leads to weaker and flatter regularization. The Language capability is evaluated using MMLU, the vision-language capability is evaluated using MMMU.

7 Conclusion

In this paper, we alleviate the degraded language capabilities in VLMs that arise during the vision-language instruction-tuning. We have identified the root cause of language capability degradation in VLMs as "representation mixing" within post-representation layers. Our proposed solution, Representation Regulation for VLM Training (RRVLM), introduces a Representation Distribution Difference loss that reduces the textual and visual representations, shifting representation mixing to earlier layers, effectively preserving language capabilities while enhancing vision-language performance. Extensive results show that our method is effective.

Limitation

Due to limited computing resources available in academic institutions, our study is conducted at a relatively modest scale. While several major research institutions have consistently reported language degradation in larger models (with sizes of up to 30B parameters) (Dai et al., 2024; McKinzie et al., 2024), we are unable to evaluate the performance of our method at comparable scales.

Acknowledgment

This work was supported by Guangdong Basic and Applied Basic Research Foundation (2025A1515011413), National Natural Science Foundation of China (62406114).

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L. Menick, Sebastian Borgeaud, and 8 others. 2022. Flamingo: a visual language model for few-shot learning. In *NeurIPS*.

Rie Kubota Ando, Tong Zhang, and Peter Bartlett. 2005. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of machine learning research*, 6(11).

Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Slav Petrov, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy P. Lillicrap, and 33 others. 2023. Gemini: A family of highly capable multimodal models. *CoRR*, abs/2312.11805.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433.

Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, Jenia Jitsev, Simon Kornblith, Pang Wei Koh, Gabriel Ilharco, Mitchell Wortsman, and Ludwig Schmidt. 2023. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, and 45 others. 2025. Qwen3-vl technical report. *Preprint*, arXiv:2511.21631.

Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. 2022. Beit: BERT pre-training of image transformers. In *ICLR*. OpenReview.net.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *NeurIPS*.

Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, and 1 others. 2024. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. *arXiv preprint arXiv:2404.16821*.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. 2023. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *arXiv preprint arXiv:2312.14238*.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, and 1 others. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2(3):6.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Wenliang Dai, Nayeon Lee, Boxin Wang, Zhuoling Yang, Zihan Liu, Jon Barker, Tuomas Rintamaki, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2024. NVLM: open frontier-class multimodal llms. *CoRR*, abs/2409.11402.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven C. H. Hoi. 2023. Instructblip: Towards general-purpose vision-language models with instruction tuning. In *NeurIPS*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, and 82 others. 2024. The llama 3 herd of models. *CoRR*, abs/2407.21783.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, and Rongrong Ji. 2023. MME: A comprehensive evaluation benchmark for multimodal large language models. *CoRR*, abs/2306.13394.

Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. 2012. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773.

Md Yousuf Harun and Christopher Kanan. 2024. Overcoming the stability gap in continual learning. *Transaction on Machine Learning Resesarch*, abs/2306.01904.

Muyang He, Yexin Liu, Boya Wu, Jianhao Yuan, Yueze Wang, Tiejun Huang, and Bo Zhao. 2024. Efficient multimodal learning from data-centric perspective. *CoRR*, abs/2402.11530.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *ICLR*. OpenReview.net.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *ICLR*. OpenReview.net.

Wenke Huang, Jian Liang, Zekun Shi, Didi Zhu, Guancheng Wan, He Li, Bo Du, Dacheng Tao, and Mang Ye. 2024. Learn from downstream and be yourself in multimodal large language model fine-tuning. *CoRR*, abs/2411.10928.

Mojan Javaheripi, Sébastien Bubeck, Marah Abdin, Jyoti Aneja, Sebastien Bubeck, Caio César Teodoro Mendes, Weizhu Chen, Allie Del Giorno, Ronen Eldan, Sivakanth Gopi, and 1 others. 2023. Phi-2: The surprising power of small language models. *Microsoft Research Blog*, 1(3):3.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. 2023. Evaluating object hallucination in large vision-language models. In *EMNLP*, pages 292–305. Association for Computational Linguistics.

Ji Lin, Hongxu Yin, Wei Ping, Pavlo Molchanov, Mohammad Shoeybi, and Song Han. 2024. Vila: On pre-training for visual language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26689–26699.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. 2024. Llava-next: Improved reasoning, ocr, and world knowledge.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. In *NeurIPS*.

Zhijian Liu, Ligeng Zhu, Baifeng Shi, Zhuoyang Zhang, Yuming Lou, Shang Yang, Haocheng Xi, Shiyi Cao, Yuxian Gu, Dacheng Li, Xiuyu Li, Yunhao Fang, Yukang Chen, Cheng-Yu Hsieh, De-An Huang, An-Chieh Cheng, Vishwesh Nath, Jinyi Hu, Sifei Liu, and 8 others. 2025. Nvila: Efficient frontier visual language models. *Preprint*, arXiv:2412.04468.

Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Hao Yang, Yaofeng Sun, Chengqi Deng, Hanwei Xu, Zhenda Xie, and Chong Ruan. 2024. Deepseek-vl: Towards real-world vision-language understanding. *CoRR*, abs/2403.05525.

Brandon McKinzie, Zhe Gan, Jean-Philippe Fauconnier, Sam Dodge, Bowen Zhang, Philipp Dufter, Dhruti Shah, Xianzhi Du, Futang Peng, Anton Belyi, Haotian Zhang, Karanjeet Singh, Doug Kang, Hongyu Hè, Max Schwarzer, Tom Gunter, Xiang Kong, Aonan Zhang, Jianyu Wang, and 10 others. 2024. MM1: methods, analysis and insights from multimodal LLM pre-training. In *ECCV*, volume 15087, pages 304–323.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. *Preprint*, arXiv:1809.02789.

Clement Neo, Luke Ong, Philip Torr, Mor Geva, David Krueger, and Fazl Barez. 2024. Towards interpreting visual information processing in vision-language models. *Preprint*, arXiv:2410.07149.

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. Instruction tuning with GPT-4. *CoRR*, abs/2304.03277.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *ICML*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Neale Ratzlaff, Man Luo, Xin Su, Vasudev Lal, and Phillip Howard. 2024. Training-free mitigation of language reasoning degradation after multimodal instruction tuning. *CoRR*, abs/2412.03467.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. 2022. Laion-5b: An open large-scale dataset for training next generation image-text models. *Preprint*, arXiv:2210.08402.

Sarah Schwettmann, Neil Chowdhury, Samuel Klein, David Bau, and Antonio Torralba. 2023. Multimodal neurons in pretrained text-only transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2862–2867.

Shikhar Srivastava, Md Yousuf Harun, Robik Shrestha, and Christopher Kanan. 2024. Improving multimodal large language models using continual learning. *CoRR*, abs/2410.19925.

Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. 2023. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 43 others. 2024. Qwen2 technical report. *CoRR*, abs/2407.10671.

Xiang Yue, Yuansheng Ni, Tianyu Zheng, Kai Zhang, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, and 3 others. 2024. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. In *CVPR*, pages 9556–9567.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986.

Yi-Kai Zhang, Shiyin Lu, Yang Li, Yanqing Ma, Qingguo Chen, Zhao Xu, Weihua Luo, Kaifu Zhang, De-Chuan Zhan, and Han-Jia Ye. 2024. Wings: Learning multimodal llms without text-only forgetting. In *Neurips*.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, Zhangwei Gao, Erfei Cui, Xuehui Wang, Yue Cao, Yangzhou Liu, Xingguang Wei, Hongjie Zhang, Haomin Wang, Weiye Xu, and 32 others. 2025. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *Preprint*, arXiv:2504.10479.

A Proof for Theorem 1

To save space, we make the following conventions. Let

$$\begin{aligned}\overline{d} &:= \frac{1}{n}\sum_{i=1}^{n}\big\|h_t(x_i) - h_v(y_i)\big\|,\\ \overline{d}^{(2)} &:= \frac{1}{n}\sum_{i=1}^{n}\big\|h_t(x_i) - h_v(y_i)\big\|^2,\end{aligned} \tag{7}$$

denote the empirical average distance and the empirical average squared distance, respectively. We also write

$$\widetilde{\mathbb{E}}_S[\cdot] := \frac{1}{n}\sum_{i=1}^{n}(\cdot) \tag{8}$$

for empirical expectations over the sample $S = \{(x_i, y_i, z_i)\}_{i=1}^n$. Note that in the theorem statement, $\mathbb{E}_S[\cdot]$ refers to the population expectation; the two differ in general.

We consider the empirical VLIT loss

$$\mathcal{L}_{\mathrm{VLIT}}(h_t, h_v, f) := \frac{1}{n}\sum_{i=1}^{n}\ell\big(f(h_t(x_i), h_v(y_i)), z_i\big). \tag{9}$$

Let K be the prediction dimension (e.g., vocabulary size). The predictor f outputs a probability vector

$$\begin{aligned}u = f(h_t(x), h_v(y)) \in \Delta_\varepsilon^K := \Big\{u \in \mathbb{R}^K :\\ \sum_{j=1}^{K} u_j = 1,\ u_j \in [\varepsilon, 1-\varepsilon]\ \forall j\Big\}.\end{aligned} \tag{10}$$

We consider the next-token supervision, where $z = e_{k(x,y)} \in \Delta^K$ is one-hot with the correct index $k(x, y) \in \{1, \dots, K\}$. The loss is the (soft) cross-entropy

$$\ell(u, z) := -\sum_{j=1}^{K} z_j \log u_j \;=\; -\log u_{k(x,y)}. \tag{11}$$

Assume there exist constants $m > 0$, $\alpha > 0$, and $\beta > 0$ such that for all triples (x, y, z), the following hold.

The first assumption characterizes that when the model's output deviates from the oracle output, the loss increases as the degree of this deviation becomes more pronounced. Define the reachable set of predictions at (x, y) as

$$\mathcal{U}(x, y) := \Big\{f(h_t(x), h_v(y)) : h_v \in \mathcal{H}\Big\} \subseteq \Delta_\varepsilon^K, \tag{12}$$

where $\mathcal{H}$ is the hypothesis class of visual encoders. Let the oracle visual encoder be h_v^* and define

$$u^* := f(h_t(x), h_v^*(y)). \tag{13}$$

Then for all $u \in \mathcal{U}(x, y)$,

$$\ell(u, z) \;\geq\; \ell(u^*, z) + \frac{m}{2}\big\|u_k - u_k^*\big\|^2. \tag{14}$$

The second assumption formalizes the prediction sensitivity to visual deviation. Namely, when the visual encoder deviates from the oracle visual encoder, the model's output undergoes at least a certain amount of change. This is a reasonable assumption.

$$\begin{aligned}&\big\|f_k(h_t(x), h_v(y)) - f_k(h_t(x), h_v^*(y))\big\| \;\geq\\ &\alpha\big\|h_v(y) - h_v^*(y)\big\|.\end{aligned} \tag{15}$$

We also need to assume some terms are bound to make analysis tractable:

$$\big\|h_t(x) - h_v^*(y)\big\| \;\leq\; \beta. \tag{16}$$

We also assume the loss is nonnegative, $\ell(u, z) \geq 0$ for all (u, z).

Fix an index $i \in \{1, \dots, n\}$ and consider the two prediction vectors

$$u_i := f(h_t(x_i), h_v(y_i)), \qquad u_i^* := f(h_t(x_i), h_v^*(y_i)). \tag{17}$$

By construction, $u_i \in \mathcal{U}(x_i, y_i)$. Applying (14) with $u = u_i$ and $u^* = u_i^*$ gives

$$\begin{aligned}&\ell(u_i, z_i) = \ell(f(h_t(x_i), h_v(y_i)), z_i) \;\geq\\ &\ell(f(h_t(x_i), h_v^*(y_i)), z_i) + \frac{m}{2}\big((u_i)_{k_i} - (u_i^*)_{k_i}\big)^2,\end{aligned} \tag{18}$$

where $k_i := k(x_i, y_i)$.

We then use the sensitivity assumption (15) to lower bound the squared prediction deviation by the squared deviation in visual representations:

$$\begin{aligned}&\Big|(u_i)_{k_i} - (u_i^*)_{k_i}\Big| =\\ &\Big|f_{k_i}(h_t(x_i), h_v(y_i)) - f_{k_i}(h_t(x_i), h_v^*(y_i))\Big|\\ &\geq\; \alpha\big\|h_v(y_i) - h_v^*(y_i)\big\|.\end{aligned} \tag{19}$$

Squaring both sides of (19) and substituting into (18) yields

$$\begin{aligned}&\ell(f(h_t(x_i), h_v(y_i)), z_i) \;\geq\; \ell(f(h_t(x_i), h_v^*(y_i)), z_i)\\ &+ \frac{m\alpha^2}{2}\big\|h_v(y_i) - h_v^*(y_i)\big\|^2.\end{aligned} \tag{20}$$

Next, we relate $\|h_v(y_i) - h_v^*(y_i)\|^2$ to the squared alignment distance $\|h_t(x_i) - h_v(y_i)\|^2$ by the triangle inequality:

$$\begin{aligned}&\big\|h_t(x_i) - h_v(y_i)\big\| \leq\\ &\big\|h_t(x_i) - h_v^*(y_i)\big\| + \big\|h_v^*(y_i) - h_v(y_i)\big\|.\end{aligned} \tag{21}$$

Rearranging gives

$$\begin{aligned}&\big\|h_v(y_i) - h_v^*(y_i)\big\| \;\geq\\ &\big\|h_t(x_i) - h_v(y_i)\big\| - \big\|h_t(x_i) - h_v^*(y_i)\big\|.\end{aligned} \tag{22}$$

Using (16),

$$\big\|h_t(x_i) - h_v^*(y_i)\big\| \;\leq\; \beta, \tag{23}$$

we obtain

$$\big\|h_v(y_i) - h_v^*(y_i)\big\| \;\geq\; \big\|h_t(x_i) - h_v(y_i)\big\| - \beta. \tag{24}$$

Let $a_i := \|h_t(x_i) - h_v(y_i)\| \geq 0$ and denote the positive part by $(t)_+ := \max\{t, 0\}$. Since the left-hand side of (24) is nonnegative,

$$\begin{aligned}\big\|h_v(y_i) - h_v^*(y_i)\big\| &\;\geq\; \big(a_i - \beta\big)_+,\\ \big\|h_v(y_i) - h_v^*(y_i)\big\|^2 &\;\geq\; \big(a_i - \beta\big)_+^2.\end{aligned} \tag{25}$$

For any $a \geq 0$, we have

$$\big(a - \beta\big)_+^2 \;\geq\; \frac{1}{2}a^2 - \beta^2. \tag{26}$$

Indeed, if $a \le \beta$ then $(a-\beta)_+^2 = 0$ and $\frac{1}{2}a^2 - \beta^2 \le 0$; if $a \ge \beta$ then $(a-\beta)_+^2 = (a-\beta)^2$ and (26) is equivalent to $(a-2\beta)^2 \ge 0$. Applying (26) with $a = a_i$ gives

$$\left\|h_v(y_i) - h_v^*(y_i)\right\|^2 \;\ge\; \frac{1}{2}\left\|h_t(x_i) - h_v(y_i)\right\|^2 - \beta^2. \quad (27)$$

Substituting (27) into (20) yields

$$\begin{aligned}
&\ell(f(h_t(x_i), h_v(y_i)), z_i) \ge \ell(f(h_t(x_i), h_v^*(y_i)), z_i) \\
&+ \frac{m\alpha^2}{2}\left(\frac{1}{2}\left\|h_t(x_i) - h_v(y_i)\right\|^2 - \beta^2\right) \\
&= \frac{m\alpha^2}{4}\left\|h_t(x_i) - h_v(y_i)\right\|^2 \\
&+ \left(\ell(f(h_t(x_i), h_v^*(y_i)), z_i) - \frac{m\alpha^2}{2}\beta^2\right).
\end{aligned} \quad (28)$$

Using $\ell(f(h_t(x_i), h_v^*(y_i)), z_i) \ge 0$, we obtain the simpler bound

$$\begin{aligned}
\ell(f(h_t(x_i), h_v(y_i)), z_i) \;\ge\; &\frac{m\alpha^2}{4}\left\|h_t(x_i) - h_v(y_i)\right\|^2 \\
&- \frac{m\alpha^2}{2}\beta^2.
\end{aligned} \quad (29)$$

Averaging (29) over $i = 1, \dots, n$ gives

$$\begin{aligned}
&\mathcal{L}_{\mathrm{VLIT}}(h_t, h_v, f) = \frac{1}{n}\sum_{i=1}^{n} \ell(f(h_t(x_i), h_v(y_i)), z_i) \\
&\ge \frac{m\alpha^2}{4}\,\overline{d}^{(2)} + C,
\end{aligned} \quad (30)$$

where

$$C := -\frac{m\alpha^2}{2}\beta^2. \quad (31)$$

In particular, up to an additive constant independent of h_v, the empirical VLIT loss lower bounds the empirical average squared alignment distance between h_t and h_v.

Assume that the representations are uniformly bounded in norm: there exists $R > 0$ such that

$$\left\|h_t(x)\right\| \le R, \qquad \left\|h_v(y)\right\| \le R \quad (32)$$

for all (x, y) in the support of the data distribution. Then

$$\left\|h_t(x) - h_v(y)\right\|^2 \le (2R)^2 = 4R^2. \quad (33)$$

Let

$$d^{(2)}(h_v) := \mathbb{E}_{(x,y)\sim\mu}\left[\left\|h_t(x) - h_v(y)\right\|^2\right] \quad (34)$$

denote the population average squared alignment distance. Define the function class

$$\Phi := \left\{\phi_{h_v} : (x,y) \mapsto \frac{1}{4R^2}\left\|h_t(x) - h_v(y)\right\|^2 \;:\; h_v \in \mathcal{H}\right\}, \quad (35)$$

so that $\phi_{h_v}(x,y) \in [0,1]$ for all (x,y) and $h_v \in \mathcal{H}$. Let $S_{xy} := \{(x_i, y_i)\}_{i=1}^n$ and define the Gaussian complexity

$$\begin{aligned}
G(\Phi(S_{xy})) &:= \mathbb{E}_\sigma\left[\sup_{\phi\in\Phi}\sum_{i=1}^{n}\sigma_i\phi(x_i,y_i)\right], \\
\sigma_i &\sim \mathcal{N}(0,1).
\end{aligned} \quad (36)$$

We use the following lemma.

Lemma 1 ((Ando et al., 2005)) *Let $\mathcal{F}$ be a function class $f : \mathcal{X} \to \mathbb{R}^k$ with $f(x) \in [0,1]^k$, and let μ be a distribution on $\mathcal{X}$. Suppose $X = \{x_1, \dots, x_n\}$ is drawn i.i.d. from μ. Then with probability at least $1-\delta$, for all $f \in \mathcal{F}$,*

$$\begin{aligned}
&\left|\frac{1}{n}\sum_{i=1}^{n} f(x_i) - \mathbb{E}_{x\sim\mu}[f(x)]\right| \le \\
&\frac{\sqrt{2\pi}}{n}G(\mathcal{F}(X)) + \sqrt{\frac{9\log(2/\delta)}{n}},
\end{aligned} \quad (37)$$

where $G(\mathcal{F}(X)) := \mathbb{E}_\sigma\left[\sup_{f\in\mathcal{F}}\sum_{i=1}^n \sigma_i f(x_i)\right]$ with $\sigma_i \sim \mathcal{N}(0,1)$.

By Lemma (37), for any $\delta \in (0,1)$, with probability at least $1-\delta$ over the draw of S_{xy}, the following holds simultaneously for all $h_v \in \mathcal{H}$:

$$\begin{aligned}
&\left|\frac{1}{n}\sum_{i=1}^{n}\frac{1}{4R^2}\left\|h_t(x_i) - h_v(y_i)\right\|^2\right. \\
&\left. \quad - \mathbb{E}_{(x,y)\sim\mu}\left[\frac{1}{4R^2}\left\|h_t(x) - h_v(y)\right\|^2\right]\right| \le \\
&\qquad \frac{\sqrt{2\pi}}{n}G(\Phi(S_{xy})) + \sqrt{\frac{9\log(2/\delta)}{n}}.
\end{aligned} \quad (38)$$

Multiplying both sides of (38) by $4R^2$ yields that with probability at least $1-\delta$, for all $h_v \in \mathcal{H}$,

$$\begin{aligned}
&\left|\overline{d}^{(2)} - d^{(2)}(h_v)\right| \le \\
&4R^2\left(\frac{\sqrt{2\pi}}{n}G(\Phi(S_{xy})) + \sqrt{\frac{9\log(2/\delta)}{n}}\right).
\end{aligned} \quad (39)$$

In particular, with probability at least $1-\delta$, for all $h_v \in \mathcal{H}$,

$$\begin{aligned}
&\overline{d}^{(2)} \;\ge \\
&d^{(2)}(h_v) - 4R^2\left(\frac{\sqrt{2\pi}}{n}G(\Phi(S_{xy})) + \sqrt{\frac{9\log(2/\delta)}{n}}\right).
\end{aligned} \quad (40)$$

Combining (30) and (40), we obtain that with probability at least $1-\delta$, for all $h_v \in \mathcal{H}$,

$$\begin{aligned}
&\mathcal{L}_{\mathrm{VLIT}}(h_t, h_v, f) \\
&\ge \frac{m\alpha^2}{4}\left(\begin{aligned}&d^{(2)}(h_v) \quad - \\ &4R^2\left(\frac{\sqrt{2\pi}}{n}G(\Phi(S_{xy})) + \sqrt{\frac{9\log(2/\delta)}{n}}\right)\end{aligned}\right) \\
&+ C \\
&= \frac{m\alpha^2}{4}\,d^{(2)}(h_v) - \\
&m\alpha^2 R^2\left(\frac{\sqrt{2\pi}}{n}G(\Phi(S_{xy})) + \sqrt{\frac{9\log(2/\delta)}{n}}\right) + C.
\end{aligned} \quad (41)$$

Now define

$$d^{(1)}(h_v) := \mathbb{E}_{(x,y)\sim\mu}\left[\left\|h_t(x) - h_v(y)\right\|\right]. \quad (42)$$

By Jensen's inequality (equivalently Cauchy–Schwarz),

$$\begin{aligned}
d^{(2)}(h_v) &= \mathbb{E}\left[\left\|h_t(x) - h_v(y)\right\|^2\right] \\
&\ge \left(\mathbb{E}\left[\left\|h_t(x) - h_v(y)\right\|\right]\right)^2 \\
&= \left(d^{(1)}(h_v)\right)^2.
\end{aligned} \quad (43)$$

Substituting (43) into (41) yields that with probability at least $1-\delta$, for all $h_v \in \mathcal{H}$,

$$\mathcal{L}_{\mathrm{VLIT}}(h_t, h_v, f) \;\geq\; \frac{m\alpha^2}{4}\left(d^{(1)}(h_v)\right)^2 - m\alpha^2 R^2\left(\frac{\sqrt{2\pi}}{n}G(\Phi(S_{xy})) + \sqrt{\frac{9\log(2/\delta)}{n}}\right) + C, \tag{44}$$

which is a lower bound in terms of $\left(\mathbb{E}\|h_t(x) - h_v(y)\|\right)^2$ and completes the proof.

B Proof for Theorem 2

Let g_θ denote the language backbone and ℓ the prediction loss (e.g., cross-entropy). Let h_t^b and h_v^b be the text encoder and vision encoder before VLIT, and let h_t^a and h_v^a be the text encoder and vision encoder after VLIT, respectively.

Here we rephrase the evaluation distribution. Let ν be the population distribution of text-only evaluation examples (x, z) (e.g., z is the next-token label). Let μ be a joint distribution over paired samples (x, y) (paired text–vision data). We assume the x-marginal of μ equals the x-marginal of ν: for any measurable $q(x)$,

$$\mathbb{E}_{(x,z)\sim\nu}[q(x)] = \mathbb{E}_{(x,y)\sim\mu}[q(x)]. \tag{45}$$

For notational convenience, we also consider a joint distribution $\widetilde{\mu}$ over triples (x, y, z) such that $(x, y) \sim \mu$ and z is the text-label paired with x (so that the marginal over (x, z) is ν).

Define the text-only evaluation risks

$$\begin{aligned}\mathcal{L}^a_{\mathrm{text}} &:= \mathbb{E}_{(x,z)\sim\nu}\big[\ell(g_{\theta^a}(h_t^a(x)), z)\big],\\ \mathcal{L}^b_{\mathrm{text}} &:= \mathbb{E}_{(x,z)\sim\nu}\big[\ell(g_{\theta^b}(h_t^b(x)), z)\big],\end{aligned} \tag{46}$$

where θ^a and θ^b are backbone parameters after and before the multimodal stage, respectively.

Define the alignment distances

$$\begin{aligned}\mathcal{D}^a &:= \mathbb{E}_{(x,y)\sim\mu}\big[\|h_t^a(x) - h_v^a(y)\|\big],\\ \mathcal{D}^b &:= \mathbb{E}_{(x,y)\sim\mu}\big[\|h_t^b(x) - h_v^b(y)\|\big].\end{aligned} \tag{47}$$

We also define the vision drift term

$$\eta_v := \mathbb{E}_{(x,y)\sim\mu}\big[\|h_v^a(y) - h_v^b(y)\|\big]. \tag{48}$$

The first assumption is that there exists a measurable random variable $\tau = \tau(x, y) \in [0, 1]$ such that under $\widetilde{\mu}$,

$$h_t^a(x) = (1-\tau)\, h_t^b(x) + \tau\, h_v^b(y). \tag{49}$$

This models representation mixing as *per-example interpolation along the segment from $h_t^b(x)$ to $h_v^b(y)$*.

Moreover, define for $\widetilde{\mu}$-a.e. (x, y, z) the one-dimensional function

$$\varphi_{x,y,z}(t) := \ell\Big(g_{\theta^b}\big((1-t)\, h_t^b(x) + t\, h_v^b(y)\big),\, z\Big), \quad t \in [0, 1]. \tag{50}$$

Assume:

$$\varphi_{x,y,z}(t) \text{ is concave in } t \text{ on } [0, 1], \tag{51}$$

and there exists a constant $\Delta_{\mathrm{vis}} > 0$ such that

$$\mathbb{E}_{(x,y,z)\sim\widetilde{\mu}}\big[\varphi_{x,y,z}(1) - \varphi_{x,y,z}(0)\big] \geq \Delta_{\mathrm{vis}}. \tag{52}$$

We also assume there exists $R > 0$ such that

$$\|h_t^b(x) - h_v^b(y)\| \leq R \qquad \widetilde{\mu}\text{-a.s.}, \tag{53}$$

and there exists $\varepsilon_{\mathrm{adapt}} \geq 0$ such that

$$\begin{aligned}&\mathbb{E}_{(x,z)\sim\nu}\big[\ell(g_{\theta^a}(h_t^a(x)), z)\big]\\ &\geq \mathbb{E}_{(x,z)\sim\nu}\big[\ell(g_{\theta^b}(h_t^a(x)), z)\big] - \varepsilon_{\mathrm{adapt}}.\end{aligned} \tag{54}$$

We now begin the proof. We first lower bound the text-only loss under the pretrained backbone θ^b when evaluated on the post-VLIT representations. By Assumption (49) and the definition (50), for $\widetilde{\mu}$-a.e. (x, y, z) we have

$$\ell\big(g_{\theta^b}(h_t^a(x)), z\big) = \varphi_{x,y,z}(\tau). \tag{55}$$

By concavity (51), for any $t \in [0, 1]$,

$$\begin{aligned}\varphi_{x,y,z}(t) &\geq (1-t)\,\varphi_{x,y,z}(0) + t\,\varphi_{x,y,z}(1)\\ &= \varphi_{x,y,z}(0) + t\big(\varphi_{x,y,z}(1) - \varphi_{x,y,z}(0)\big).\end{aligned} \tag{56}$$

Plugging $t = \tau$ and taking expectation over $\widetilde{\mu}$ yields

$$\begin{aligned}&\mathbb{E}_{\widetilde{\mu}}\big[\ell(g_{\theta^b}(h_t^a(x)), z)\big]\\ &\geq \mathbb{E}_{\widetilde{\mu}}\big[\ell(g_{\theta^b}(h_t^b(x)), z)\big] + \mathbb{E}_{\widetilde{\mu}}\Big[\tau\big(\varphi_{x,y,z}(1) - \varphi_{x,y,z}(0)\big)\Big].\end{aligned} \tag{57}$$

Next, we relate $\mathbb{E}[\tau]$ to the alignment improvement measured with the *same anchor* h_v^b. Define

$$\begin{aligned}\overline{\mathcal{D}}^{\,a} &:= \mathbb{E}_{(x,y)\sim\mu}\big[\|h_t^a(x) - h_v^b(y)\|\big],\\ \overline{\mathcal{D}}^{\,b} &:= \mathbb{E}_{(x,y)\sim\mu}\big[\|h_t^b(x) - h_v^b(y)\|\big].\end{aligned} \tag{58}$$

By definition, $\overline{\mathcal{D}}^{\,b} = \mathcal{D}^b$. Moreover, from Assumption (49),

$$h_t^a(x) - h_v^b(y) = (1-\tau)\big(h_t^b(x) - h_v^b(y)\big), \tag{59}$$

hence

$$\|h_t^b(x) - h_v^b(y)\| - \|h_t^a(x) - h_v^b(y)\| = \tau\,\|h_t^b(x) - h_v^b(y)\|. \tag{60}$$

Taking expectation over $(x, y) \sim \mu$ gives

$$\overline{\mathcal{D}}^{\,b} - \overline{\mathcal{D}}^{\,a} = \mathbb{E}_{\widetilde{\mu}}\big[\tau\,\|h_t^b(x) - h_v^b(y)\|\big]. \tag{61}$$

Using the boundedness (53), we obtain

$$\begin{aligned}&\overline{\mathcal{D}}^{\,b} - \overline{\mathcal{D}}^{\,a} = \mathbb{E}_{\widetilde{\mu}}\big[\tau\,\|h_t^b(x) - h_v^b(y)\|\big] \leq R\,\mathbb{E}_{\widetilde{\mu}}[\tau],\\ \Rightarrow\quad &\mathbb{E}_{\widetilde{\mu}}[\tau] \geq \frac{\overline{\mathcal{D}}^{\,b} - \overline{\mathcal{D}}^{\,a}}{R}.\end{aligned} \tag{62}$$

We now connect $\overline{\mathcal{D}}^{\,a}$ back to the theorem notation $\mathcal{D}^a$. By the triangle inequality,

$$\|h_t^a(x) - h_v^b(y)\| \leq \|h_t^a(x) - h_v^a(y)\| + \|h_v^a(y) - h_v^b(y)\|. \tag{63}$$

Taking expectation over $(x, y) \sim \mu$ and using the definitions (47) and (48), we obtain

$$\overline{\mathcal{D}}^{\,a} \leq \mathcal{D}^a + \eta_v. \tag{64}$$

Since $\overline{\mathcal{D}}^{\,b} = \mathcal{D}^b$, it follows that

$$\overline{\mathcal{D}}^{\,b} - \overline{\mathcal{D}}^{\,a} \geq \mathcal{D}^b - \mathcal{D}^a - \eta_v. \tag{65}$$

Methods	Language Capability					Vision-Language Capability			
	MMLU↑	ARC_E ↑	ARC_C ↑	O.B.QA ↑	Avg↑	MMMU ↑	MME_P ↑	MME_C ↑	POPE ↑
Llava 1.5-7B									
LLM (Vicuna-1.5)	**50.04**	**78.87**	**48.01**	34.20	**52.78**	-	-	-	-
Full-Tuning	48.48	76.68	44.80	34.00	50.99	26.70	1363.88	288.57	86.30
RRVLM (our)	49.25	78.24	46.42	**35.60**	52.38	**27.80**	**1375.45**	**305.00**	**86.57**
NVILA-8B-Lite									
LLM (Qwen2)	64.51	**79.42**	48.46	**30.40**	55.69	-	-	-	-
Full-Tuning	63.77	76.81	48.89	31.40	55.21	**45.33**	1359.55	262.14	78.72
RRVLM (our)	**64.86**	79.34	**49.83**	**30.40**	**56.10**	**45.33**	**1392.34**	**276.78**	**82.81**

Table 5: Results of on Llava and NVILA frameworks. Best results are marked in **bold**.

Combining (62) and (65), we get

$$\mathbb{E}_{\widetilde{\mu}}[\tau] \geq \frac{\mathcal{D}^b - \mathcal{D}^a - \eta_v}{R}. \tag{66}$$

Now we lower bound the mixed term in (57). Since $\tau \in [0,1]$ and $\varphi_{x,y,z}(1) - \varphi_{x,y,z}(0)$ is integrable, we use the bound

$$\begin{aligned}&\mathbb{E}_{\widetilde{\mu}}\Big[\tau\big(\varphi_{x,y,z}(1) - \varphi_{x,y,z}(0)\big)\Big] \\ &\geq \mathbb{E}_{\widetilde{\mu}}[\tau] \cdot \mathbb{E}_{\widetilde{\mu}}\big[\varphi_{x,y,z}(1) - \varphi_{x,y,z}(0)\big] - \varepsilon_{\text{corr}},\end{aligned} \tag{67}$$

where $\varepsilon_{\text{corr}} \geq 0$ captures any possible negative correlation between τ and the endpoint gap. (If one additionally assumes independence, then $\varepsilon_{\text{corr}} = 0$.)

Combining (57), (66), (52), and (67), we obtain

$$\begin{aligned}&\mathbb{E}_{\widetilde{\mu}}\big[\ell(g_{\theta^b}(h_t^a(x)), z)\big] \\ &\geq \mathbb{E}_{\widetilde{\mu}}\big[\ell(g_{\theta^b}(h_t^b(x)), z)\big] + \frac{\mathcal{D}^b - \mathcal{D}^a - \eta_v}{R}\,\Delta_{\text{vis}} - \varepsilon_{\text{corr}}.\end{aligned} \tag{68}$$

By the marginal matching (45), the expectations over $\widetilde{\mu}$ coincide with those over ν for losses that depend only on (x, z), hence

$$\begin{aligned}&\mathbb{E}_{(x,z)\sim\nu}\big[\ell(g_{\theta^b}(h_t^a(x)), z)\big] \geq \\ &\mathcal{L}_{\text{text}}^b + \frac{\mathcal{D}^b - \mathcal{D}^a - \eta_v}{R}\,\Delta_{\text{vis}} - \varepsilon_{\text{corr}}.\end{aligned} \tag{69}$$

Finally, apply the limited-adaptation assumption (54):

$$\begin{aligned}\mathcal{L}_{\text{text}}^a &= \mathbb{E}_{(x,z)\sim\nu}\big[\ell(g_{\theta^a}(h_t^a(x)), z)\big] \\ &\geq \mathbb{E}_{(x,z)\sim\nu}\big[\ell(g_{\theta^b}(h_t^a(x)), z)\big] - \varepsilon_{\text{adapt}}.\end{aligned} \tag{70}$$

Combining (69) and (70) yields

$$\mathcal{L}_{\text{text}}^a - \mathcal{L}_{\text{text}}^b \geq \frac{\mathcal{D}^b - \mathcal{D}^a - \eta_v}{R}\,\Delta_{\text{vis}} - \varepsilon_{\text{corr}} - \varepsilon_{\text{adapt}}. \tag{71}$$

In particular, if

$$\frac{\mathcal{D}^b - \mathcal{D}^a - \eta_v}{R}\,\Delta_{\text{vis}} > \varepsilon_{\text{corr}} + \varepsilon_{\text{adapt}}, \tag{72}$$

then $\mathcal{L}_{\text{text}}^a > \mathcal{L}_{\text{text}}^b$, which concludes the proof.

C Compared Method

LLM represents the original LLM within the VLM, establishing the baseline language capabilities possessed by the unmodified LLM. **Attention-Tuning** follows the setting of Llama 3-V (Dubey et al., 2024), only tuning the attention layers during VLIT while keeping all other layers frozen throughout the process. **LoRA** (Hu et al., 2022) freezes the original VLM weights and adds learnable low-rank adapters. **Full-Tuning** refers to full-parameter fine-tuning on the VLM. **SoftLabel** (Harun and Kanan, 2024) adopts smoothed labels rather than hard labels during the VLIT stage. **ModelMerge** (Ratzlaff et al., 2024) merges the base LLM parameters back into language model parameters of the VLM after the VLIT stage. **RRVLM** represents our proposed training approach. Recent advanced models like Deepseek-VL (Lu et al., 2024) incorporate additional text-only data during VLIT to preserve language capabilities. **DSVL Mixing** fully fine-tunes the VLM on a combined dataset of vision-language and text-only data at a 7:3 ratio.

D Backbones

Within the Bunny framework, we evaluate methods on three combinations of LLM backbones and vision encoders Phi2 (Javaheripi et al., 2023) with EvaCLIP (Sun et al., 2023), Phi2 (Javaheripi et al., 2023) with SigLIP (Zhai et al., 2023), Llama3-7B (Dubey et al., 2024) with SigLIP (Zhai et al., 2023). Within the LLava framework, the LLM backbone is Vicuna (Chiang et al., 2023) and the vision encoder is CLIP (Radford et al., 2021). Within the NVILA framework, the LLM backbone is Qwen2-8B (Yang et al., 2024) and the vision encoder is SigLIP (Zhai et al., 2023).

E Implementation Details

Implementation Details. We use a learning rate of `2e-5` and conduct training on a single NVIDIA A100 GPU. Other hyperparameter settings follow those of Bunny (He et al., 2024), LLaVA (Liu et al., 2023), and VILA (Lin et al., 2024). We provide an anonymous link `https://anonymous.4open.science/r/LanguageCapability-ED08` to our code to ensure reproducibility.

Datasets. The training set used in the VLIT stage is the **Bunny-695K** (He et al., 2024). Bunny dataset consists of 624k image-text pairs and 70k text-only samples. Given that we have multiple backbones and various comparison methods, training on the entire dataset would be prohibitively expensive. Therefore, we utilize a subset of 200k image-text pairs for training. In experiments, we have two settings, namely, without text-only data and with text-only data. In the setting that includes additional text-only data, we augmented the image-text pairs with text-only data at varying mixture ratios according to the specifications of different methods. In the setting that without text-only data, we remove the text-only data and only use image-text pairs as the training set.

F Results on more Backbones

To further evaluate the effectiveness of our method on large-scale models, we conduct additional experiments using two extra frameworks, including Llava and NVILA. Detailed description of the two frameworks can be found in Appendix D. These

models were trained using their official open-source codes and initialized with publicly available checkpoints, which only undergo the first stage (Vision-Language Pre-training). This stage aims to align representations of visual and textual modalities. In this experiment, we did not use any text-only data during the VLIT stage. Due to the high computational cost, we used only 100K multimodal samples in the VLIT stage (as opposed to the 200K samples used in the main paper). As shown in Tab. 5, our method preserves language capabilities well, demonstrating the consistent improvement under different frameworks and different model scales.

G Settings of Representation Mixing

Our analysis is based on a dataset containing 1,000 samples. For each sample, we compute the pairwise Euclidean distances between all textual and visual token embeddings and then calculate the average distance.

H Case Study

Tab.6 and Tab.7 show that our method preserves the language capabilities, while the baseline fails to preserve the language capabilities.

Tab.8 Case study comparing baselines and RRVLM on vision-language understanding task. RRVLM demonstrates better vision-language capabilities.

User Prompt	LLM	Full-Tuning	RRVLM (Ours)
`Write a short story about a robot who learns to love.`	*Once upon a time, there was a robot named Robby. He was created to be the perfect assistant, but he felt lonely and unfulfilled. One day, he met a little girl who loved to play with him. They became best friends and Robby learned that love was not just about being perfect, but about being kind and caring.*	*Exercise 4: Write a short story about a robot who learns to love.*	*Once upon a time, there was a robot named Robby. Robby was created to be a cleaning robot, but he always felt like something was missing. He longed for love and companionship, but he didn't know how to find it. One day, Robby stumbled upon a group of humans who were playing with their pets. He watched as the humans laughed and played with their furry friends, and he felt a pang of jealousy. ... He listened as they strummed their guitars and sang their hearts out, and he felt a sense of joy that he had never experienced before.*

Table 6: Case study comparing LLM, Full-Tuning, and RRVLM. Our model exhibits greater narrative depth. The **...** is short for the paragraph: "He wished he could feel the same way. As he continued to observe the humans, he noticed that they were all wearing different types of clothing. Some were wearing bright colors, while others were dressed in more muted tones. Robby realized that clothing was a way for humans to express themselves and show their personalities. Inspired by this realization, Robby decided to try something new. He began to experiment with different colors and patterns, and soon he was wearing a bright red shirt and a pair of blue jeans. He felt a sense of pride and confidence as he walked around, and he knew that he was finally starting to find his place in the world. As he continued to explore, Robby stumbled upon a group of humans who were playing music.".

User Prompt	LLM	Full-Tuning	RRVLM (Ours)
`Solve the puzzle below: I speak without a mouth and hear without ears. I have no body, but I come alive with the wind. What am I?`	*Solution: The answer to the puzzle is "echo".*	*It is a small, white, fluffy cloud in the sky.*	*Answer: The answer to the puzzle is an echo.*

Table 7: Case study comparing LLM, Full-Tuning, and RRVLM. Our model accurately identifies the abstract concept of "echo", while the baseline fails semantically.

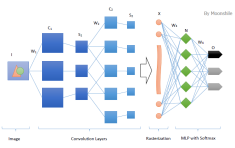
User Prompt (Text + Image)	Full-Tuning	RRVLM (Ours)
`What is described in the image?` 	*The image is a complex diagram or flowchart, possibly related to computer science or programming. It's filled with various shapes, arrows, and text.*	*The image seems to be related to CNN Neural Network. It includes elements like a large circle, a square, and a triangle, which could be part of a CNN architecture. The arrows and lines might represent connections or pathways within the network.*

Table 8: Case study comparing baselines and RRVLM on vision-language understanding task. RRVLM demonstrates more specific and semantically grounded understanding of neural network structures.