Permute-then-Adapt: Weak-to-Strong Contrastive Image–Text Adaptation

Jinhao Li[1] jinhao.li2@unimelb.edu.au
Sarah Erfani[1] sarah.erfani@unimelb.edu.au
Lei Feng[2] fenglei@seu.edu.cn
Guangrui Li[3]* guangrli@amazon.com
James Bailey[4] james.a.bailey@monash.edu
Feng Liu[1]† feng.liu1@unimelb.edu.au

[1]School of Computing and Information Systems, The University of Melbourne, Australia
[2]School of Computer Science and Engineering, Southeast University, China
[3]AWS AI Labs, USA
[4]Department of Data Science and Artificial Intelligence, Monash University, Australia

Abstract

Image–text alignment models such as CLIP are typically trained with contrastive learning, where unpaired examples are treated as strict negatives. While this assumption introduces noise by penalising semantically related pairs, existing solutions often rely on heuristic similarity thresholds that lack statistical grounding and sensitivity to dataset-specific noise distributions. In this paper, we propose *Permute-then-Adapt (PTA)*, a framework that leverages efficient *weak models* to identify missed positives while controlling Type-I error relative to a conditional permutation reference, up to calibration approximation error. Unlike standard distillation or arbitrary thresholding, PTA estimates a random-pairing reference distribution from the weak teacher's similarity scores and uses its upper tail, determined by α, to select auxiliary positives. The calibrated positive set is then trained against by a single multi-positive contrastive objective: each anchor maximises the total probability mass it assigns to its positive set. Offline calibration with the weak model adds negligible training overhead. Experiments show gains over heuristic soft-label approaches in object recognition, cross-modal retrieval, and data efficiency. Project repository: https://github.com/JinhaoLee/Permute-then-Adapt.

1 Introduction

Recent advances in multimodal learning [37, 19, 32, 48, 7, 51, 42] have demonstrated the immense potential of image-text alignment models for understanding and connecting visual and textual information. Models like CLIP [37] have shown remarkable capabilities in zero-shot recognition [6], cross-modal retrieval [30], and serving as foundational components for larger multimodal systems [44]. These models achieve their performance through contrastive learning on large-scale image-text pairs, where the core training objective is to maximise similarity between matched pairs while minimising similarity between unmatched ones.

However, the assumption that all unmatched pairs are completely unrelated is a fundamental limitation in contrastive learning, as it forces the model to treat semantically valid overlaps as false negatives [4, 47]. Prior work uses similarity-based filtering to curate image–text training data [7], or learns soft alignment targets through teacher supervision and momentum distillation [18, 28]. These strategies motivate dataset-adaptive calibration of auxiliary-positive selection. A fixed threshold is

*Work done before Guangrui joined AWS.
†Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

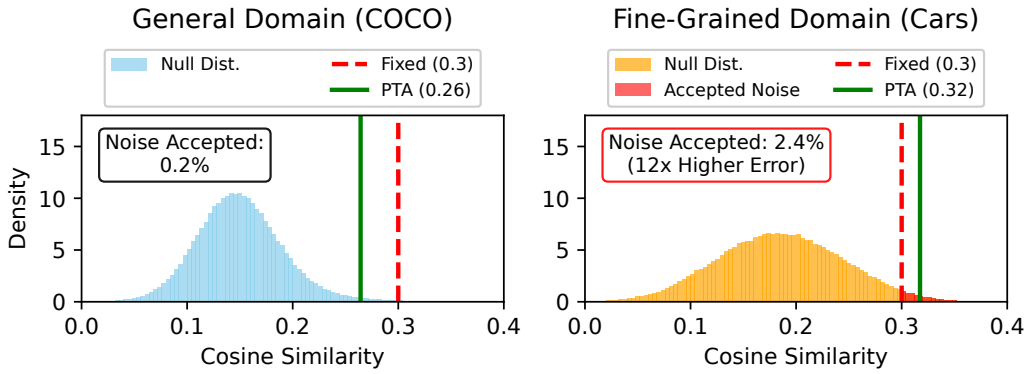


Figure 1: **The risk of uncalibrated heuristics.** While a fixed threshold (red dashed) works on COCO (**Left**), it fails on fine-grained Stanford Cars (**Right**), where the random-pairing null distribution shifts rightward. The fixed threshold accepts **12× more false positives** from the null on Cars. In contrast, **PTA** (green solid) estimates a dataset-specific permutation null and adapts its cutoff to control Type-I error relative to that reference at level α, up to calibration approximation error.

inherently brittle: it fails to adapt to the varying signal-to-noise ratios across different datasets or model architectures, leading to either missed positives or the introduction of noise. This reveals a critical gap: the lack of a mechanism to distinguish potential signals from the null distribution of random pairings without manual tuning. As shown in Figure 1, a fixed threshold that works for a general domain like COCO fails catastrophically on fine-grained tasks (e.g., Stanford Cars), accepting 12× more noise due to the domain's naturally higher baseline similarities. We address this by proposing a *statistically grounded* framework. Our method uses permutation testing to estimate the conditional random-pairing null and select auxiliary positives with Type-I error control relative to that reference, up to calibration approximation error.

In this paper, we propose a framework for enhancing image-text alignment by leveraging the semantic knowledge of *efficient weak models*. Our motivation stems from the insight that weak-to-strong generalisation [3] (e.g., smaller CLIP variants), despite their limitations in zero-shot accuracy, encode rich signals about semantic relatedness that are computationally cheap to access. However, raw similarity scores are noisy; relying on a single fixed threshold to extract them is brittle and fails to generalise across domains. To rigorously separate this signal from noise, we introduce **Permute-then-Adapt (PTA)**, a weak-to-strong framework that replaces heuristic thresholding with *statistical hypothesis testing*. Inspired by the permutation test [15], PTA estimates a dataset-specific reference distribution of weak-teacher similarities under random image–text pairings and selects auxiliary positives whose scores exceed its calibrated upper-tail threshold. This approach requires no additional validation data and adapts naturally to the intrinsic noise floor of different datasets. The resulting calibrated positive set is then trained against by a single multi-positive contrastive objective: for each anchor we maximise the total probability mass assigned to its positive set, where the positive set is the ground-truth diagonal augmented with the calibrated off-diagonal pairs. We validate our approach through extensive experiments across image classification and retrieval benchmarks, demonstrating that our statistically grounded guidance consistently outperforms heuristic baselines.

Our main contributions can be summarised as follows: (i) *Permutation-calibrated auxiliary positive selection.* We use semantically irrelevant permutations to calibrate a dataset-adaptive cutoff, providing Type-I error control relative to the conditional random-pairing reference at significance level α, up to Monte Carlo and percentile approximation error. (ii) *PTA multi-positive objective.* We train against the calibrated set with a single positive-set marginal-likelihood loss. (iii) Through extensive experiments on multiple zero-shot image classification benchmarks and two cross-modal retrieval datasets, we show that PTA consistently outperforms existing weakly supervised baselines and tolerates moderate inference-time teacher dropout.

2 Related work

Vision-language models. Since the introduction of CLIP [37], numerous models building upon its multimodal representation learning framework have emerged. Subsequent work has explored

architectural improvements and training-recipe refinements, such as ALIGN [19] which scales up the training data and model size, EVA-CLIP [41] which improves the training stability and efficiency of large-scale CLIP through optimised initialisations and learning rates, and Florence [49] which incorporates additional visual reasoning capabilities. BLIP [29] uses caption bootstrapping for unified vision–language understanding and generation, while LiT [50] trains a text encoder against a locked pretrained image encoder. FG-CLIP [46] explicitly targets fine-grained visual–textual alignment. PTA is orthogonal to this line of work: rather than scaling the model or replacing the backbone, it provides a post-hoc refinement that exploits a calibrated, dataset-adaptive selection of auxiliary positives and could in principle be applied on top of any of these architectures.

CLIP fine-tuning. Recent advancements in fine-tuning CLIP models have led to various parameter-efficient methods aimed at enhancing performance on downstream tasks. The Context Optimization (CoOp) [55] approach introduces learnable text prompts, enabling the model to adapt to specific tasks by optimizing these prompts during training. Building upon this, Conditional Prompt Learning (CoCoOp) [54] extends CoOp by conditioning the prompts on image instances, thereby improving generalisation across diverse visual concepts. Alternatively, CLIP-Adapter [9] incorporates lightweight residual-style adapters into CLIP's architecture, facilitating the learning of task-specific features without extensive retraining. Tip-Adapter [53] further streamlines this process by constructing a key-value cache model from the few-shot training set, enabling efficient adaptation with minimal or no training. Additionally, methods like Visual Prompt Tuning (VPT) [20] add learnable prompt tokens in the input space, fine-tuning the model effectively for various tasks. These strategies aim to improve CLIP's adaptability and performance while maintaining computational efficiency.

Bootstrapping and pseudo-labelling. Several works attempt to mitigate noise in image-text alignment through bootstrapping or pseudo-labelling. BLIP [29] employs caption bootstrapping and multi-stage training to refine noisy web-crawled supervision. More recent studies propose pseudo-labelling strategies to relax one-hot supervision, such as Cross-Modal Similarity Regulation [21], multi-label pseudo-labelling [47], and soft-label alignment [18]. These methods highlight the importance of leveraging softer or multiple alignment signals. Our approach shares this motivation but differs by introducing a statistically grounded mechanism (permutation testing) to identify semantically meaningful extra positives with controlled false alarm rates.

Knowledge distillation and weak-to-strong supervision. Knowledge distillation techniques, such as ALBEF [28], employ momentum encoders to refine alignment signals from auxiliary models. Weak-to-strong transfer has also been studied more broadly [3], emphasising how weaker models guide stronger ones. For CLIP classification, CPL [27] learns class prototypes under weak-model supervision. PTA builds upon this weak-to-strong philosophy but departs from direct distillation: instead of transferring continuous similarity scores, it adaptively thresholds weak-model signals using a permutation-derived null distribution, producing statistically validated supervision.

Mitigating false negatives. False negatives, i.e. unpaired image–text instances that are semantically related, have recently received direct attention in vision–language pretraining. For instance, MAFA [4] proposes an efficient connection mining process to convert some hard false negatives into positives, and smooths the contrastive ITC loss to reduce over-penalisation. Empirically, MAFA shows that doing this yields substantial downstream gains, especially under hard-negative sampling strategies. PTA distinguishes itself by employing a permutation-based null distribution to statistically validate and promote high-scoring unpaired tuples as auxiliary positives for weak-to-strong supervision, and by reusing the same calibrated mask to clean false negatives from the contrastive loss. Related approaches mitigate noisy supervision by modifying contrastive objectives or learning softer alignment targets. PyramidCLIP [10], SoftCLIP [11], and CLIP-PSD [1] follow this direction. PTA calibrates auxiliary-positive selection against a permutation reference. AdaCL [34] focuses on *clone negatives* captions that are semantically close but not exact matches by adaptively reshaping the positive logit, but still treats all non-matching pairs as negatives. NCU [14] studies post-hoc unlearning of noisy correspondences in pretrained CLIP, addressing both false positives and false negatives, and can operate with a subset of the original pretraining data. We instead study post-hoc weak-to-strong refinement on a small, clean downstream dataset with correct positives, where in-batch sampling introduces hard false negatives into the contrastive loss.

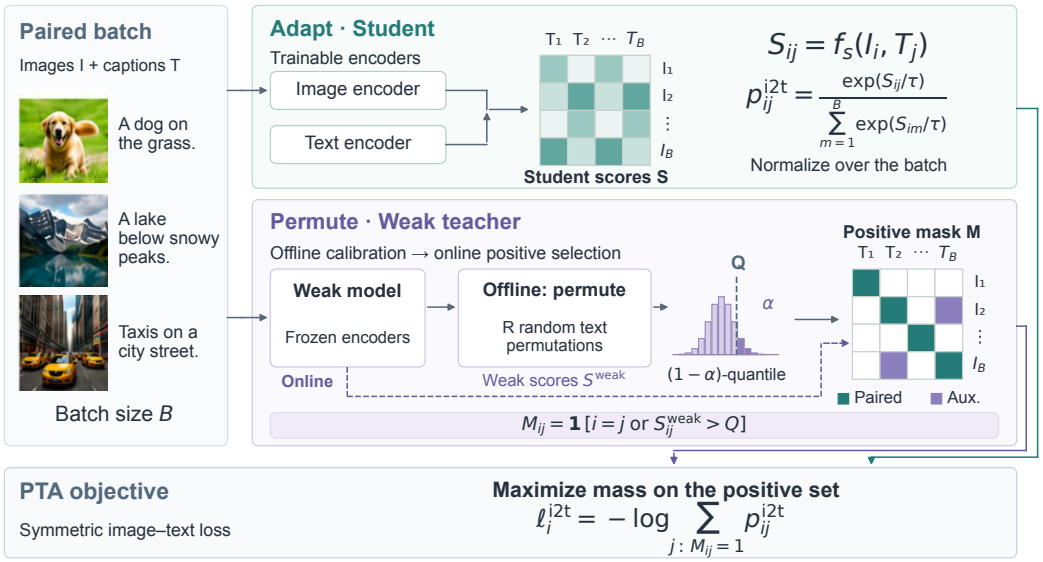


Figure 2: **Overview of Permute-then-Adapt (PTA).** Given paired image–text samples, PTA uses a frozen weak teacher to estimate a permutation null distribution and dataset-adaptive threshold. This threshold selects calibrated off-diagonal auxiliary positives, which join ground-truth diagonal pairs to form positive sets. The student is optimized with a positive-set marginal likelihood objective.

3 Preliminaries

Dual encoder models. Dual encoder models [37] belong to a class of neural architectures designed to learn representations of two different modalities in a shared embedding space. These models consist of two separate encoders that process different input types (e.g., images and text) and project them into a common semantic space where similarity can be directly computed. Formally, we define: an image encoder $f_{\text{image}} : \mathcal{X} \to \mathbb{R}^d$ that maps images to a d-dimensional embedding, and a text encoder $f_{\text{text}} : \mathcal{Z} \to \mathbb{R}^d$ that maps text to the same d-dimensional space.

Image-text alignment. Image-text alignment aims to capture semantic correspondences between visual and textual content [37, 26]. Given a dataset $\mathcal{D} = \{(I_i, T_i)\}_{i=1}^N$ of image-text pairs, where $I_i \in \mathcal{X}$ represents an image and $T_i \in \mathcal{Z}$ represents its corresponding text description, the goal is to learn a mapping that reflects semantic similarity in a shared embedding space. For notational brevity, we denote the similarity score assigned by a model f to an image-text pair (I_i, T_j) as the inner product of their normalised embeddings: $f(I_i, T_j) = \langle f_{\text{image}}(I_i), f_{\text{text}}(T_j) \rangle$.

We formulate the alignment task using contrastive learning on the full dataset $\mathcal{D}$. The objective aligns embeddings by maximising the similarity score $f(I_i, T_i)$ for matched pairs while minimising it for mismatched pairs (e.g., I_i and T_j where $i \neq j$).

To ensure proper gradient scaling, a learnable temperature parameter τ is introduced. For each anchor i, the image-to-text and text-to-image contrastive losses are defined as:

$$\ell_{\text{i2t}}^{(i)} = -\log \frac{\exp\left(f(I_i, T_i)/\tau\right)}{\sum_{j=1}^{N} \exp\left(f(I_i, T_j)/\tau\right)}, \ell_{\text{t2i}}^{(i)} = -\log \frac{\exp\left(f(I_i, T_i)/\tau\right)}{\sum_{j=1}^{N} \exp\left(f(I_j, T_i)/\tau\right)}. \tag{1}$$

The total loss

$$\mathcal{L}^{\text{clip}} = \frac{1}{2N} \sum_{i=1}^{N} \left(\ell_{\text{i2t}}^{(i)} + \ell_{\text{t2i}}^{(i)} \right) \tag{2}$$

is minimised over $\mathcal{D}$ to align embeddings, with negative pairs enforcing discriminative separation.

4 Image-text alignment with weak models

We introduce our method to improve a strong model's image-text alignment using weak models.

4.1 Our method: Permute-then-Adapt

In this section, we will discuss the problem setup and our method. Figure 2 demonstrates our method visually. The overall algorithm can be found in Algorithm 1.

A key limitation of standard contrastive learning (Equation 2) is its hard-negative assumption: all non-paired tuples (I_i, T_j) with $i \neq j$ are treated as negatives, even though some may be semantically related. A natural idea is to threshold teacher f similarity scores to convert such tuples into auxiliary positives; however, fixed cutoffs [7] are brittle across datasets and model architectures.

Problem setup. We consider post-hoc refinement of a pretrained dual-encoder (the *student* f^s, also denoted the *strong model*) on a paired dataset $\mathcal{D}$. Given a frozen, lightweight *teacher* f (also denoted the *weak model*[3]), our goal is to identify a subset of non-paired tuples (I_i, T_j) with $i \neq j$ as *auxiliary positives* and use them to enhance the image-text alignment of the student f_s.

Weak teacher similarities. A data-calibrated threshold Q defines pseudo-labels for non-paired tuples. Standard contrastive learning encodes supervision via the dataset pairing indicator

$$l_D(I_i, T_j) := \mathbb{I}[i = j], \tag{3}$$

Unpaired tuples receive the hard label $l_D = 0$, which we replace with a weak pseudo-label $l_f(\cdot, \cdot; Q)$:

$$l_f(I, T; Q) = \begin{cases} 1 & \text{if } f(I, T) > Q, \\ 0 & \text{otherwise,} \end{cases} \tag{4}$$

where f is a frozen weak teacher that scores arbitrary image–text pairs, and Q selects high-scoring tuples as auxiliary positives. Assigning the weak pseudo-label $l_f(I, T; Q)$ gives the adaptation set

$$\mathcal{D}^f_{\text{adapt}}(Q) = \{(I_i, T_j, l_f(I_i, T_j; Q))\}_{i,j=1}^N. \tag{5}$$

We can then fine-tune the student using $\mathcal{D}^f_{\text{adapt}}(Q)$[4]. A key difficulty is that $\mathcal{D}^f_{\text{adapt}}(Q)$ is highly sensitive to the choice of Q. Fixed similarity thresholds [7] are not comparable across datasets or model architectures: the same numeric value of Q may correspond to very different signal-to-noise ratios, leading to either missed auxiliary positives or excessive noise (as shown in Figure 1). To make selection principled and dataset-adaptive, we calibrate Q via permutation testing at significance level α, providing Type-I error control relative to the conditional permutation reference, up to Monte Carlo and percentile approximation error [15].

4.1.1 Permute: estimating the null distribution

Permutation-based null. Selecting Q requires a reference distribution of similarity scores for *irrelevant* image–text pairs. Under the independence null $H_0 : I \perp T$, text permutation preserves the observed marginals but breaks associations, yielding a valid (conditional) permutation null [13, 15].

Interpretation. The permutation null is intentionally *not* a generative model of hard semantic negatives, and our framework does not require that real images and texts are independent. Instead, the null serves as an *empirical reference distribution of weak-similarity scores under random mismatching*, evaluated on the same dataset and the same weak teacher used downstream. PTA then promotes off-diagonal pairs whose weak similarity is implausibly large relative to this reference; semantically related hard negatives, which would otherwise be penalised by the standard contrastive loss, are precisely the right-tail deviations we wish to rescue from strict negation. The null thus provides a conservative, dataset-adaptive baseline without assuming that semantic clustering is absent.

Semantically irrelevant perturbations. Shuffling text in the paired dataset $\mathcal{D} = \{(I_i, T_i)\}_{i=1}^N$ yields an empirical permutation null *sample* of weak similarity scores. Let $\mathfrak{G}_N$ denote the permutation group on $[N] = \{1, \ldots, N\}$, with elements $\sigma : [N] \to [N]$ acting on the text tuple $\mathcal{T} = (T_1, \ldots, T_N)$ via $\sigma\mathcal{T} := (T_{\sigma(1)}, \ldots, T_{\sigma(N)})$. Under semantically irrelevant image–text permutations, we draw R random permutations $\{\sigma_r\}_{r=1}^R \sim \text{Unif}(\mathfrak{G}_N)$ and collect the resulting null similarity scores:

$$\mathcal{S}_{\text{perm}} := \left\{ f(I_i, T_{\sigma_r(i)}) \;:\; i \in [N],\ r \in [R] \right\}. \tag{6}$$

[3]We use "teacher"/"student" and "weak"/"strong" interchangeably throughout the paper. The framework only requires that the teacher is a frozen dual-encoder whose similarity scores can serve as the calibration signal; nothing in our derivation relies on it being smaller than the student. We choose a smaller teacher in our experiments for computational practicality.

[4]Although Eq. 5 defines the full candidate set of N^2 image–text pairs, in practice we do not materialise all possible pairs during training. Instead, training operates on mini-batches

This empirical permutation null sample retains repeated scores; we threshold auxiliary positives with Type-I error control relative to that reference, up to calibration approximation error.

Proposition 4.1 (Validity of the permutation null [12, 24, 15])**.** *Under the independence null $H_0 : I \perp T$, suppose text assignments are exchangeable conditional on the observed image and text marginals $\mathcal{M}$. All matchings $\{(I_i, T_{\sigma(i)})\}_{i=1}^N$ are then equally likely under H_0. For a fixed teacher f, a uniformly sampled matching and image index yield the conditional random-pairing score CDF*

$$F_0(t) = \frac{1}{N^2} \sum_{i,j=1}^{N} \mathbf{1}\{f(I_i, T_j) \le t\}.$$

Independent uniform text permutations provide a Monte Carlo approximation to this reference distribution. When calibration uses a subset, apply the statements below with $N = N_{\text{cal}}$ and $\mathcal{M}$ denoting its observed marginals.

Conditional permutation testing provides the theoretical background [24]; the bound below explicitly accounts for calibration approximation error.

Definition 4.2 (Permutation-Calibrated Selection)**.** Given a significance level $\alpha \in (0, 1)$, define

$$Q_{\text{PTA}} := \text{percentile}_{1-\alpha}(\mathcal{S}_{\text{perm}}).$$

The selection rule assigns a pseudo-label according to

$$l_f^p(I, T) := \begin{cases} 1 & \text{if } f(I, T) > Q_{\text{PTA}}, \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

Corollary 4.3 (Type-I error control)**.** *Condition on $\mathcal{M}$, the fixed teacher f, and the sampled permutations Π. Let J, K be independent uniform indices in $[N]$, independent of Π, so that (I_J, T_K) is a random pair from the same observed marginals. Write $Q = Q_{\text{PTA}}$, and let $\widehat{F}_R$ be the empirical CDF of $\mathcal{S}_{\text{perm}}$. Define the finite-calibration approximation error*

$$\varepsilon_{\text{cal}} := \|\widehat{F}_R - F_0\|_\infty + [1 - \widehat{F}_R(Q) - \alpha]_+,$$

where $[x]_+ = \max\{x, 0\}$. Then the Type-I error relative to the conditional random-pairing reference satisfies

$$\Pr_{J,K}\Big(l_f^p(I_J, T_K) = 1 \mid \mathcal{M}, f, \Pi\Big) \le \min\{1, \alpha + \varepsilon_{\text{cal}}\}. \tag{8}$$

Indeed, the left-hand side equals $1 - F_0(Q)$, which is at most $1 - \widehat{F}_R(Q) + \|\widehat{F}_R - F_0\|_\infty$. The first error term is the Monte Carlo reference approximation; the second accounts for percentile discretisation or interpolation and is zero for a generalised-inverse empirical quantile.

Thus, α specifies the tolerated Type-I error relative to the conditional permutation reference, up to the stated calibration approximation error. This reference control does not assert semantic precision of the selected auxiliary positives.

4.1.2 Adapt: enhance image-text alignment

Following this selection procedure, we collect the permutation-calibrated pseudo-labels into a training set of additional positives for fine-tuning the strong model. Let $\mathcal{K} \subset [N]$ be a set of indices with $|\mathcal{K}| = B$ defining a mini-batch. We use paired samples from $\mathcal{D}$ together with off-diagonal PTA-labelled pairs mined within the batch:

$$\mathcal{D}_{\text{batch}} := \{(I_k, T_k) : k \in \mathcal{K}\}, \mathcal{D}_{\text{batch}}^{\text{PTA}} := \{(I_k, T_m, l_f^p(I_k, T_m)) : k, m \in \mathcal{K},\ k \ne m\}. \tag{9}$$

By construction, $\mathcal{D}_{\text{batch}}^{\text{PTA}} \subset \mathcal{D}_{\text{adapt}}^f(Q)$. Define the PTA indicator for an in-batch pair as

$$l_{km}^{\text{PTA}} := l_f^p(I_k, T_m) \in \{0, 1\}, \qquad (k \ne m), \tag{10}$$

and set $l_{kk}^{\text{PTA}} := 0$ for completeness. We then form the final target mask

$$\tilde{l}_{km}^{\text{PTA}} \ = \ \mathbb{I}[k = m] \ \vee \ l_{km}^{\text{PTA}}, \quad \tilde{l}_{km}^{\text{PTA}} \in \{0, 1\}. \tag{11}$$

Algorithm 1 Permute-then-Adapt (PTA) Training Framework

Input: Teacher f, Student f^s, Dataset $\mathcal{D}$, significance level α.
Stage 1: Calibration (Offline)
Sample N_{cal} image-text pairs from $\mathcal{D}$ and apply R random text permutations to obtain the empirical null sample $\mathcal{S}_{\text{perm}}$.
Determine threshold $Q_{\text{PTA}} \leftarrow \text{percentile}_{1-\alpha}(\mathcal{S}_{\text{perm}})$.
Stage 2: Adaptation (Online)
for mini-batch $\mathcal{B} = \{(I_k, T_k)\}_{k=1}^{B}$ **do**
    // Calibrated positive set
    Compute teacher similarity matrix $\mathbf{S}_{\text{weak}} \in \mathbb{R}^{B\times B}$.
    For each anchor k, form $P_k \leftarrow \{k\} \cup \{m \neq k : S_{\text{weak}}(I_k, T_m) > Q_{\text{PTA}}\}$ (Eq. 12).
    // Single-equation multi-positive contrastive loss
    Compute $\mathcal{L}^{\text{PTA}}$: for each anchor, the negative log of the student's total probability mass on P_k (Eq. 13); standard CLIP InfoNCE is recovered when $P_k = \{k\}$.
    Update Student: $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}^{\text{PTA}}$.
end for

That is, each anchor always includes its ground-truth match ($k = m$) and includes additional in-batch positives selected by PTA via the permutation-calibrated threshold. In contrast to standard CLIP supervision, which treats only the diagonal pair as positive, PTA promotes a selected subset of off-diagonal in-batch pairs to positives.

Positive-set marginal likelihood objective. PTA trains the student with a *positive-set marginal likelihood* [8, 52]: each anchor maximises the total probability mass it assigns to its calibrated positive set. For an image anchor I_k with positive set

$$P_k \;=\; \{k\} \cup \{\, m \neq k \,:\, l_{km}^{\text{PTA}} = 1 \,\}, \tag{12}$$

the image-to-text loss is

$$\ell_{\text{i2t}}^{(k)} \;=\; -\log \frac{\sum_{m\in P_k} \exp(f^{\text{s}}(I_k, T_m)/\tau)}{\sum_{m'=1}^{B} \exp(f^{\text{s}}(I_k, T_{m'})/\tau)}, \tag{13}$$

with the symmetric text-to-image counterpart $\ell_{\text{t2i}}^{(m)}$. Averaging both directions over the batch gives

$$\mathcal{L}^{\text{PTA}} \;=\; \frac{1}{2}\left(\frac{1}{B}\sum_{k=1}^{B} \ell_{\text{i2t}}^{(k)} \;+\; \frac{1}{B}\sum_{m=1}^{B} \ell_{\text{t2i}}^{(m)}\right). \tag{14}$$

When $P_k = \{k\}$ this objective reduces to the standard CLIP InfoNCE loss; the only hyperparameter is the significance level α controlling which off-diagonal pairs join P_k.

5 Experiments

We evaluate PTA on zero-shot classification and Karpathy-split cross-modal retrieval, compare it with heuristic weak supervision, and examine how calibrated positives enter the objective. Table 1 reports five-seed mean±SD; additional ablations examine teacher quality and calibration.

Datasets. Our experiments leverage the Microsoft COCO [30] (Common Objects in Context) 2014 dataset. To evaluate generalisation, we test on natural-image object-recognition tasks spanning CIFAR-10 [23], CIFAR-100 [23], Food-101 [2], ImageNet [6] together with its ImageNet-V2/A/R variants [38, 17, 16], Describable Textures (DTD) [5], StanfordCars [22], CUB-200 [43], FGVC Aircraft [31], Flowers-102 [33], and Oxford Pets [35], which cover diverse domains. For cross-modal retrieval, we use Flickr30K [36] and MS-COCO [30], standard benchmarks for image-text matching.

Baselines. We compare our method against a comprehensive set of baselines: (1) CLIP supervised fine-tuning (SFT) baselines. These provide reference points for direct adaptation of student. (2) Weak-to-Strong Augmentation (WSA). Proposed by Burns et al. [3], WSA incrementally augments positives discovered by weaker models in multi-stage training. This is the closest prior to our weak-to-strong framework. (3) Soft Align (SA). Following recent work on soft-label alignment [18], we use similarity

Table 1: **Zero-shot classification accuracy (%).** CLIP ViT-B/16 students are fine-tuned on COCO train2014 with a CLIP ViT-B/32 teacher. Upper and lower rows show the mean and sample SD over seeds 41–45. PTA uses finite-rank calibration (implementation details). Best means are bold.

Method	ImageNet	IN-V2	IN-A	IN-R	Cars	DTD	C-100	C-10	Food	CUB	Aircraft	Flowers	Pets	Avg.
SA	66.23 ±0.05	59.49 ±0.13	37.15 ±0.27	71.82 ±0.22	59.38 ±0.23	**43.36** ±0.23	67.46 ±0.41	90.40 ±0.40	84.55 ±0.13	**53.30** ±0.24	19.86 ±0.48	63.27 ±0.12	87.11 ±0.17	61.80 ±0.06
VPL	65.78 ±0.05	58.74 ±0.07	40.89 ±0.31	72.55 ±0.14	58.29 ±0.15	43.17 ±0.29	65.88 ±0.51	89.80 ±0.22	82.58 ±0.14	49.88 ±0.31	20.15 ±0.17	62.07 ±0.41	87.63 ±0.20	61.34 ±0.08
FT	65.02 ±0.05	57.76 ±0.06	40.49 ±0.30	72.15 ±0.16	56.87 ±0.33	42.19 ±0.27	65.05 ±0.57	89.63 ±0.25	81.80 ±0.08	48.05 ±0.22	18.83 ±0.43	61.39 ±0.51	87.31 ±0.21	60.50 ±0.06
MD	65.68 ±0.10	58.53 ±0.21	40.80 ±0.25	72.57 ±0.03	59.28 ±0.23	42.75 ±0.36	65.34 ±0.37	88.89 ±0.23	82.23 ±0.08	50.42 ±0.15	19.66 ±0.35	62.96 ±0.24	87.53 ±0.17	61.28 ±0.07
WSA	65.77 ±0.07	58.61 ±0.20	**41.28** ±0.28	72.97 ±0.08	59.28 ±0.20	42.71 ±0.21	65.36 ±0.49	89.37 ±0.19	82.38 ±0.11	50.04 ±0.35	19.58 ±0.47	62.03 ±0.59	87.65 ±0.17	61.31 ±0.08
PTA (ours)	**66.63** ±0.07	**59.84** ±0.20	40.34 ±0.27	**73.57** ±0.13	**60.54** ±0.27	43.32 ±0.26	**67.64** ±0.38	**90.90** ±0.41	**84.64** ±0.10	52.58 ±0.20	**22.21** ±0.52	**63.51** ±0.37	**88.00** ±0.28	**62.59** ±0.06

SA: Soft Align; VPL: VL PseudoLabel; FT: Fixed Threshold; MD: Momentum Distillation;
WSA: Weak-to-Strong Augmentation. Avg. is the 13-benchmark mean; its SD uses five seed-wise averages.

Table 2: **Cross-modal retrieval (recall@1/5).** All systems are fine-tuned on COCO train2014; best scores are bold. SFT: supervised fine-tuning. Other abbreviations follow Table 1.

Method	Flickr30k				MS-COCO			
	Image to Text		Text to Image		Image to Text		Text to Image	
	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5
SFT	81.00	95.60	79.70	95.30	22.69	41.69	21.94	41.29
SA	81.90	96.40	81.70	96.00	23.65	43.69	22.26	42.06
VPL	81.30	96.10	81.50	95.80	24.91	45.35	24.24	44.81
FT	83.00	96.20	81.30	**96.40**	23.91	43.72	23.84	43.69
MD	82.70	96.20	81.20	96.30	23.89	43.73	23.69	43.58
WSA	**83.40**	96.20	81.40	96.30	23.97	43.67	23.85	43.72
PTA (ours)	82.40	**96.60**	**83.20**	96.30	**25.24**	**45.93**	**25.06**	**45.73**

scores from a weaker model to generate soft positive labels, replacing the binary supervision in the contrastive loss. (4) VL PseudoLabel (VPL). Inspired by multi-label pseudo-labeling approaches for vision-language models [47], this baseline allows each image to have multiple captions marked positive if predicted above a confidence threshold. (5) Fixed Threshold (FT). As a naive baseline, we apply a constant threshold (e.g., 0.3) on weak model similarity scores to decide additional positives [7]. This tests whether our permutation-based statistical control is necessary. (6) Momentum Distillation (MD). Following ALBEF [28], we maintain a momentum-updated copy of the weak model as a teacher, and minimize KL divergence between the teacher's and student's similarity distributions.

Implementation details. Unless otherwise specified, a frozen CLIP ViT-B/32 teacher guides a CLIP ViT-B/16 student on COCO train2014. We train for up to 20 epochs using AdamW (learning rate $10^{-4}B/4096$, weight decay 10^{-4}, 10% warmup, cosine decay), BF16 mixed precision, and gradient clipping on one GPU (Appendix A.10.1). Legacy single-seed runs tune the last four vision blocks. The Table 1 runs use the full vision tower and last two text blocks, $B = 512$, seeds $\{41, 42, 43, 44, 45\}$, and the checkpoint with the best in-domain validation recall. For PTA in Table 1, calibration uses 5,000 records in batches of 1,000 at $\alpha = 0.01$, discarding fixed-point scores and retaining repetitions. Given M retained scores, we use the k-th order statistic with $k = \lceil (M+1)(1-\alpha) \rceil$ and strict selection $s > Q$. This finite-rank variant retains the positive-set objective in Eq. 13. Other experiments retain the original empirical 99th-percentile calibration unless stated otherwise.

Main results. Table 1 reports five-seed zero-shot classification on 13 benchmarks after fine-tuning on COCO. PTA achieves the highest average accuracy, 62.59%, and the best mean on 10 of the 13 tasks. SA is strongest on DTD and CUB-200, while WSA is strongest on ImageNet-A.

PTA exceeds SA, the strongest baseline by average accuracy, by 0.80 percentage points (62.59% versus 61.80%). Relative to SA, PTA gains 3.19 points on ImageNet-A and 2.35 on FGVC Aircraft,

while trailing by 0.04 on DTD and 0.72 on CUB-200. These differences are computed from unrounded five-seed means.

Table 2 reports cross-modal retrieval on Flickr30k and MS-COCO. On MS-COCO, PTA raises image-to-text recall@1 from 22.69% to 25.24% and text-to-image recall@1 from 21.94% to 25.06%, gains of 2.55 and 3.12 percentage points over SFT. Recall@5 gains are 4.24 and 4.44 points. On Flickr30k, recall@1 gains are 1.40 points for image-to-text and 3.50 for text-to-image. PTA achieves the highest text-to-image recall@1, while WSA remains stronger on image-to-text recall@1, showing that gains vary with retrieval direction.

How weak can the teacher be? We progressively corrupt the CLIP-B/32 teacher at inference time with dropout rates $\{0, 0.1, 0.3, 0.5, 0.8\}$, keeping the CLIP-B/16 student architecture and training hyperparameters fixed. Figure 3 compares the teacher's standalone accuracy with the PTA-trained student's benchmark average. At dropout 0.3, teacher accuracy falls from 61.00% to 39.00%, whereas student accuracy falls from 65.60% to 64.63% (0.97 percentage points). Student accuracy is less sensitive to this moderate corruption than the teacher's standalone accuracy. Severe corruption still matters: at dropout 0.8, teacher accuracy is 3.54% and student accuracy falls to 59.79%, 5.81 points below its uncorrupted-teacher counterpart. Thus, the student tolerates moderate degradation but remains dependent on useful teacher information. Inference-time dropout is a controlled corruption proxy; these results do not establish robustness to every source of teacher error. Full per-task results and a separate calibration-only diagnostic are in Appendix A.8.

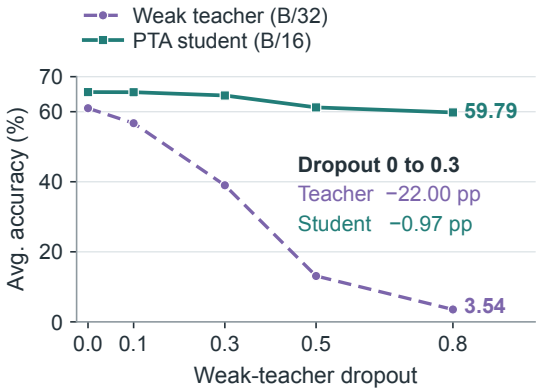


Figure 3: **Sensitivity to weak-teacher corruption.** Teacher and PTA student accuracy under inference-time teacher dropout, with the student setup fixed. Points reproduce reported benchmark averages; per-task results are in Appendix A.8.

Mechanism decomposition. Table 3 reports independent ablation runs with matched SFT controls, varying how the calibrated mask is consumed: (i) a *CN-only* variant that uses the mask only to remove false negatives from the contrastive denominator, and (ii) an *asymmetric* variant that keeps the full multi-positive numerator only on the image→text direction. CN-only accounts for roughly 71% of the full PTA gain ($+1.08 \pm 0.06$); admitting the calibrated pairs into the numerator (full PTA) buys the remaining ∼0.45.

Table 3: **PTA mechanism decomposition.** Independent ablations of Eq. 13: paired accuracy gains (pp) over matched SFT controls on 13 benchmarks, mean ± SD over five seeds (three for asymmetric).

Variant	Avg. gain (pp)
SFT (ceiling)	–
CN-only (App. A.1.1)	+1.076 ±0.061
Asymmetric (t→i: CE)	+1.147 ±0.035
PTA (full)	**+1.524** ±0.108

CN-only retains the observed match as the sole numerator term while removing selected off-diagonal pairs from the denominator. Full PTA retains the full normalization and maximizes probability mass over the calibrated positive set, without requiring equal probability for each candidate. The asymmetric result is consistent with a benefit from applying the positive-set objective in both directions.

Empirical calibration across datasets and teachers. Across 100 resampling seeds on COCO and Flickr30k with CLIP ViT-B/32 and ViT-B/16 teachers, held-out random-pairing exceedance at $\alpha = 0.01$ is 1.005–1.023% using $M = 5{,}000$ calibration scores. Larger budgets reduce resampling variability. These global-permutation diagnostics are separate from training-time minibatch calibration; dependent pooled scores and finite cached pools limit their interpretation. These empirical rates do not establish exact α control or semantic precision. Full protocols and results are in Appendix A.6; additional permutation-stability results are in Appendix A.5.

Visual grounding. Figure 4 compares Grad-CAM [39] maps for SA and PTA using the same three queries. PTA can reduce conflicting supervision on potentially related pairs through calibrated auxiliary positives, encouraging the student to focus on query-relevant visual regions. Full queries and retrieval rankings are in Appendix A.11.

Training efficiency. Figure 5 compares per-step throughput relative to SFT. Under fixed teacher preprocessing, PTA-Cache reuses frozen embeddings and scores candidates by matrix multiplication, avoiding repeated teacher encoding. It retains the thresholding and positive-set training rules while achieving 95–99% of SFT throughput. The trade-off is preprocessing and embedding storage; cache construction is excluded. Details are in Appendix A.10.

Sensitivity to α. The highest reported ImageNet accuracy occurs at $\alpha = 0.01$ (Figure 6).

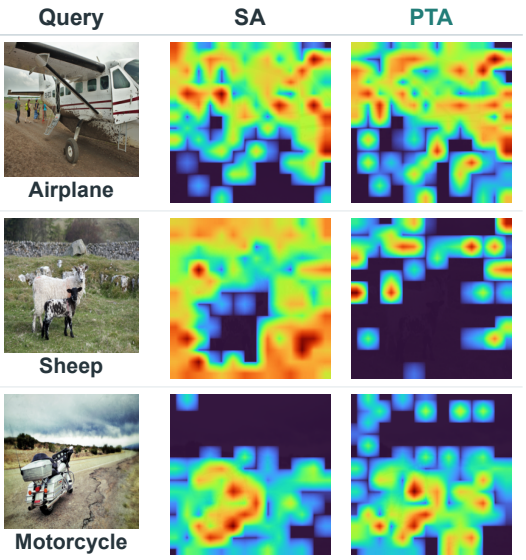


Figure 4: **Grad-CAM visualisations.** Query images and SA/PTA maps for three selected examples. Full query texts are in Appendix A.11.

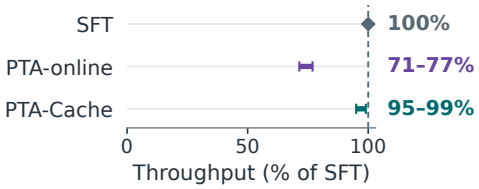


Figure 5: **Training throughput.** Ranges are not confidence intervals. Online values use inverse slowdown; cache construction excluded.

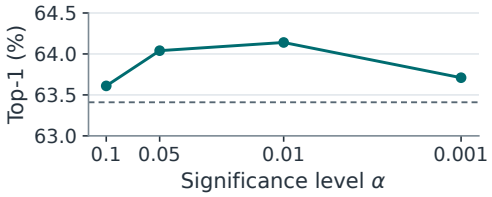


Figure 6: **Sensitivity to α.** ImageNet top-1 accuracy at the tested α values. Points are reported estimates; dashed: no PTA.

Additional experiments. The appendix covers clean-negative ablations (Appendix A.1.1); calibration budgets and α (Appendices A.2–A.3); low-scoring matched pairs (Appendix A.4); permutation stability and held-out calibration (Appendices A.5–A.6); threshold rules (Appendix A.7); teacher-quality diagnostics (Appendix A.8); data scaling and cached training (Appendices A.9–A.10); and retrieval examples with full queries (Appendix A.11).

Limitations. Experiments cover CLIP-based dual encoders, natural-image classification, and image–text retrieval, with scaling up to three million CC3M pairs. Multilingual corpora, specialised domains, and larger pretraining regimes remain to be evaluated. PTA retains observed matches as positives and selects auxiliary positives within each minibatch. Repairing incorrect original pairings or searching broader candidate pools would require extending the reference construction and evaluating the associated computational trade-offs.

6 Conclusion

PTA uses a weak teacher to select auxiliary positives for image–text adaptation. Permutation calibration provides Type-I error control relative to the conditional permutation reference, up to calibration approximation error. A frozen dual-encoder teacher enables offline calibration. A single multi-positive objective maximises probability mass over the calibrated positive set and recovers CLIP InfoNCE when auxiliary positives are absent. PTA achieves the highest average classification accuracy among the evaluated weak-supervision methods and improves retrieval over supervised fine-tuning. Under inference-time dropout, student accuracy is less sensitive to moderate corruption than teacher accuracy, while severe corruption still degrades student performance.

Acknowledgments and Disclosure of Funding

We thank The University of Melbourne's Research Computing Services and the Petascale Campus for providing computational resources.

Funding. Jinhao Li is supported by the Melbourne Research Scholarship. Feng Liu is supported by the ARC with grant numbers DP230101540, DE240101089 and DP260103996.

The authors declare no competing interests.

References

[1] Alex Andonian, Shixing Chen, and Raffay Hamid. Robust cross-modal representation learning with progressive self-distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16430–16441, 2022.

[2] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part VI 13*, pages 446–461. Springer, 2014.

[3] Collin Burns, Pavel Izmailov, Jan Hendrik Kirchner, Bowen Baker, Leo Gao, Leopold Aschenbrenner, Yining Chen, Adrien Ecoffet, Manas Joglekar, Jan Leike, et al. Weak-to-strong generalization: Eliciting strong capabilities with weak supervision. *arXiv preprint arXiv:2312.09390*, 2023.

[4] Jaeseok Byun, Dohoon Kim, and Taesup Moon. Mafa: Managing false negatives for vision-language pretraining. In *CVPR*, 2024.

[5] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613, 2014.

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

[7] Alex Fang, Albin Madappally Jose, Amit Jain, Ludwig Schmidt, Alexander Toshev, and Vaishaal Shankar. Data filtering networks. *arXiv preprint arXiv:2309.17425*, 2023.

[8] Lei Feng, Jiaqi Lv, Bo Han, Miao Xu, Gang Niu, Xin Geng, Bo An, and Masashi Sugiyama. Provably consistent partial-label learning. *Advances in neural information processing systems*, 33:10948–10960, 2020.

[9] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision*, 132(2):581–595, 2024.

[10] Yuting Gao, Jinfeng Liu, Zihan Xu, Jun Zhang, Ke Li, Rongrong Ji, and Chunhua Shen. Pyramidclip: Hierarchical feature alignment for vision-language model pretraining. *Advances in neural information processing systems*, 35:35959–35970, 2022.

[11] Yuting Gao, Jinfeng Liu, Zihan Xu, Tong Wu, Enwei Zhang, Ke Li, Jie Yang, Wei Liu, and Xing Sun. Softclip: Softer cross-modal alignment makes clip stronger. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 1860–1868, 2024.

[12] Phillip Good. *Permutation, parametric and bootstrap tests of hypotheses*. Springer, 2005.

[13] Arthur Gretton, Kenji Fukumizu, Choon Teo, Le Song, Bernhard Schölkopf, and Alex Smola. A kernel statistical test of independence. In *Advances in neural information processing systems*, 2007.

[14] Haochen Han, Alex Jinpeng Wang, Peijun Ye, and Fangming Liu. Unlearning the noisy correspondence makes clip more robust. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4518–4528, 2025.

[15] Jesse Hemerik and Jelle Goeman. Exact testing with random permutations. *Test*, 27(4):811–825, 2018.

[16] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. *ICCV*, 2021.

[17] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. *CVPR*, 2021.

[18] Hailang Huang, Zhijie Nie, Ziqiao Wang, and Ziyu Shang. Cross-modal and uni-modal soft-label alignment for image-text retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18298–18306, 2024.

[19] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021.

[20] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, pages 709–727. Springer, 2022.

[21] Chaoya Jiang, Wei Ye, Haiyang Xu, Songfang Huang, Fei Huang, and Shikun Zhang. Vision language pre-training by contrastive learning with cross-modal similarity regulation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14660–14679. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.acl-long.819. URL https://aclanthology.org/2023.acl-long.819/.

[22] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops*, June 2013.

[23] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009. URL https://cave.cs.toronto.edu/kriz/learning-features-2009-TR.pdf.

[24] Erich Leo Lehmann and Joseph P Romano. *Testing statistical hypotheses*. Springer, 2005.

[25] Fei-Fei Li, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. In *2004 Conference on Computer Vision and Pattern Recognition Workshop*, page 178. IEEE, 2004. doi: 10.1109/CVPR.2004.383. URL https://doi.org/10.1109/CVPR.2004.383.

[26] Jinhao Li, Haopeng Li, Sarah Monazam Erfani, Lei Feng, James Bailey, and Feng Liu. Visual-text cross alignment: Refining the similarity score in vision-language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 28018–28039. PMLR, 2024. URL https://proceedings.mlr.press/v235/li24ag.html.

[27] Jinhao Li, Sarah Monazam Erfani, Lei Feng, James Bailey, and Feng Liu. Exploring weak-to-strong generalization for CLIP-based classification. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=quE8gDDegf.

[28] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705, 2021.

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

[30] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.

[31] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

[32] Norman Mu, Alexander Kirillov, David Wagner, and Saining Xie. Slip: Self-supervision meets language-image pre-training. In *European conference on computer vision*, pages 529–544. Springer, 2022.

[33] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics and Image Processing*, Dec 2008.

[34] Renjie Pan, Jihao Dong, and Hua Yang. Discovering clone negatives via adaptive contrastive learning for image-text matching. In *The Thirteenth International Conference on Learning Representations*, 2025.

[35] Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

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

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

[38] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, pages 5389–5400. PMLR, 2019.

[39] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[40] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565. Association for Computational Linguistics, 2018. doi: 10.18653/v1/P18-1238. URL https://aclanthology.org/P18-1238/.

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

[42] Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, et al. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

[43] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The Caltech-UCSD Birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

[44] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

[45] Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. SUN database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3485–3492. IEEE, 2010. doi: 10.1109/CVPR.2010.5539970. URL https://doi.org/10.1109/CVPR.2010.5539970.

[46] Chunyu Xie, Bin Wang, Fanjing Kong, Jincheng Li, Dawei Liang, Gengshen Zhang, Dawei Leng, and Yuhui Yin. Fg-clip: Fine-grained visual and textual alignment. *arXiv preprint arXiv:2505.05071*, 2025.

[47] Xin Xing, Zhexiao Xiong, Abby Stylianou, Srikumar Sastry, Liyu Gong, and Nathan Jacobs. Vision-language pseudo-labels for single-positive multi-label learning. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 7799–7808. IEEE, 2024.

[48] Hu Xu, Saining Xie, Xiaoqing Ellen Tan, Po-Yao Huang, Russell Howes, Vasu Sharma, Shang-Wen Li, Gargi Ghosh, Luke Zettlemoyer, and Christoph Feichtenhofer. Demystifying clip data. *arXiv preprint arXiv:2309.16671*, 2023.

[49] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021.

[50] Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18123–18133, 2022.

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

[52] Jiahan Zhang, Qi Wei, Feng Liu, and Lei Feng. Candidate pseudolabel learning: Enhancing vision-language models by prompt tuning with unlabeled data. *arXiv preprint arXiv:2406.10502*, 2024.

[53] Renrui Zhang, Rongyao Fang, Wei Zhang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free clip-adapter for better vision-language modeling. *arXiv preprint arXiv:2111.03930*, 2021.

[54] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16816–16825, 2022.

[55] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022.

A Appendix

Contents

A.1 Additional experiments . . . 15
A.1.1 Clean-negatives ablation . . . 15
A.2 Ablation on the number of permutations . . . 15
A.3 Ablation on sensitivity to α . . . 15
A.4 What happens to ground-truth pairs below Q? . . . 16
A.5 Monte Carlo stability of global permutation calibration . . . 16
A.6 Empirical calibration across datasets and teachers . . . 17
A.7 Ablation on thresholding strategies . . . 18
A.8 Additional details on weak-teacher quality . . . 18
A.8.1 Calibration diagnostics under teacher corruption . . . 18
A.9 Ablation on data scaling . . . 19
A.10 PTA-Cache for efficient training . . . 19
A.10.1 Compute resources and recorded runtimes . . . 20
A.11 Visualisation . . . 21
A.11.1 Qualitative retrieval . . . 21
A.11.2 Queries for the main-text Grad-CAM visualisations . . . 21

A.1 Additional experiments

A.1.1 Clean-negatives ablation

The mechanism-decomposition study in Table 3 compares the full PTA objective (Eq. 13) against a *clean-negatives only* variant. The CN-only variant uses the calibrated mask solely to remove false negatives from the contrastive denominator, restricting the numerator to the ground-truth diagonal:

$$\ell_{\text{i2t}}^{\text{CN},(k)} = -\log \frac{\exp(f^s(I_k, T_k)/\tau)}{\exp(f^s(I_k, T_k)/\tau) + \sum_{m \neq k}(1 - l_{km}^{\text{PTA}}) \exp(f^s(I_k, T_m)/\tau)}, \tag{15}$$

with the symmetric text-to-image counterpart. CN-only therefore isolates the contribution of *not penalising* calibrated positives from the contribution of *pulling* them into the numerator. Empirically, CN-only recovers $\sim$71% of the full PTA gain in classification (Table 3); the remaining $\sim$0.45 comes from admitting calibrated pairs into the numerator, with no additional hyperparameters.

A.2 Ablation on the number of permutations

The permutation null is a Monte-Carlo estimate, so the calibrated threshold Q inherits sampling variability from the size of $\mathcal{S}_{\text{perm}}$. Table 4 reports the across-seed variability of Q as we vary the number of permuted scores from $1{,}000$ to $20{,}000$. With only $1{,}000$ scores the threshold is noticeably noisy across seeds, but at our default of $5{,}000$ scores (R=5 permutations of $1{,}000$ pairs) the across-seed range has dropped below 0.3 in scaled-similarity units, and further increases to $10{,}000$ or $20{,}000$ produce only marginal stabilisation. We therefore use $5{,}000$ permuted scores throughout the main results; the calibration cost is negligible relative to a single training step.

A.3 Ablation on sensitivity to α

We study how the permutation-test significance level α (used to accept weakly-matched pairs as additional positives) affects downstream performance. Figure 6 reports ImageNet top-1 accuracy after fine-tuning with different α. Overall, performance exhibits a clear inverted-U trend: overly permissive thresholds (large α) admit noisy positives and reduce accuracy, while overly conservative thresholds (small α) accept too few positives and diminish the benefit of PTA. In our setting, $\alpha = 10^{-2}$ achieves

Table 4: **Threshold stability.** Range and standard deviation of Q across seeds $\{41, 42, 43\}$ for different numbers of permuted scores in $\mathcal{S}_{\text{perm}}$. Threshold units are cosine similarity $\cdot$ $\exp(\text{logit_scale})$. The default is $5{,}000$ scores from $R{=}5$ random permutations of $1{,}000$ pairs.

# permuted scores	Across-seed range of Q	Std of Q
1,000	8.50 – 9.00	0.25
5,000 (default)	8.44 – 8.69	0.13
10,000	8.49 – 8.62	0.07
20,000	8.51 – 8.57	0.03

Table 5: Thresholding ablation. Average accuracy (%) on a legacy benchmark suite at $B{=}4096$ with the last four vision blocks unfrozen (single seed); included to mirror prior protocol.

Method	Avg. accuracy (%)	Δ vs. SFT
SFT	62.42	0.00
FT	63.08	0.66
Naive percentile	64.59	2.17
PTA	65.60	3.18

the best accuracy, and $\alpha \in \{5 \times 10^{-2}, 10^{-2}\}$ remains competitive, indicating that PTA is not overly sensitive within a moderate range. We therefore set $\alpha = 10^{-2}$ as the default in all main experiments.

A.4 What happens to ground-truth pairs below Q?

A natural question is whether the calibration mistakenly excludes some true matched pairs whose weak similarity falls below the threshold. On a diagnostic over $10{,}000$ COCO training pairs evaluated with the learned threshold Q, around 5.6% of the diagonal pairs fall below Q under the weak model. The PTA selection rule preserves these pairs by construction. Specifically, each anchor's positive set always includes its original matched example; the threshold Q determines which off-diagonal pairs are added as auxiliary positives. Low teacher scores therefore do not remove the original image–text pairs from supervision during student adaptation.

A.5 Monte Carlo stability of global permutation calibration

We fix $n = 5{,}000$ COCO train2014 image–caption records and their frozen CLIP B/32 features, using the first caption per image and fixed preprocessing. This calibration-only diagnostic evaluates global permutations separately from the batch-wise training implementation. For each $R \in \{1, 10, 100\}$, we pool scores from R uniform permutations of the complete calibration set, discard fixed-point positions, and retain repeated scores with equal weight. We use the linear empirical $(1 - \alpha)$ quantile and the strict decision $s > Q$. We use 30 permutation seeds (41–70); per repetition, the R settings share permutation prefixes. The calibration features, evaluation pairs, and probe batches remain fixed. Reference exceedance is measured on 10,000 pairs whose image and text sources come from separate held-out record blocks. Auxiliary selection is measured on off-diagonal pairs in ten disjoint probe batches of 512 records. The calibration, probe, and reference record sets are disjoint.

Table 6 shows that increasing R reduces conditional Monte Carlo variability. At $\alpha = 0.01$, the threshold standard deviation is approximately 8.1 times smaller for $R = 100$ than for $R = 1$, while mean reference exceedance remains near 1%. This experiment assesses calibration stability, not student accuracy or semantic precision. Reusing the same records does not create nR independent calibration samples or establish exact α control.

Table 6: Global-permutation calibration on fixed features ($n = 5{,}000$). Values are mean $\pm$ sample standard deviation across 30 permutation seeds. Q is in scaled-similarity units. Rates are percentages, with their standard deviations in percentage points. Variability is conditional on the fixed records and features; reference exceedance measures random-pair selection, not semantic error.

α	R	Q	Reference exceedance (%)	Auxiliary selection (%)
0.005	1	28.262 ± 0.359	0.462 ± 0.067	0.451 ± 0.077
0.005	10	28.152 ± 0.142	0.473 ± 0.024	0.471 ± 0.031
0.005	100	28.151 ± 0.048	0.468 ± 0.008	0.471 ± 0.011
0.010	1	26.520 ± 0.294	0.969 ± 0.094	0.932 ± 0.106
0.010	10	26.384 ± 0.126	1.007 ± 0.036	0.977 ± 0.046
0.010	100	26.387 ± 0.036	1.008 ± 0.011	0.975 ± 0.013

Table 7: Held-out random-pairing exceedance rate (%) for the linear-quantile calibration rule. Entries are mean $\pm$ sample standard deviation over 100 resampling seeds; standard deviations are in percentage points. M is the number of retained permuted calibration scores. These rates concern random pairing, rather than semantic false positives.

α	M	COCO ViT-B/32	COCO ViT-B/16	Flickr30k ViT-B/32	Flickr30k ViT-B/16
0.005	100	1.258 ± 1.166	1.170 ± 0.869	1.384 ± 1.248	1.176 ± 0.945
0.005	200	0.909 ± 0.742	0.904 ± 0.684	0.951 ± 0.817	0.942 ± 0.707
0.005	500	0.643 ± 0.330	0.631 ± 0.342	0.595 ± 0.321	0.605 ± 0.309
0.005	1,000	0.560 ± 0.250	0.548 ± 0.222	0.581 ± 0.293	0.572 ± 0.220
0.005	5,000	0.518 ± 0.109	0.509 ± 0.101	0.509 ± 0.136	0.525 ± 0.121
0.005	10,000	0.515 ± 0.089	0.508 ± 0.094	0.496 ± 0.108	0.519 ± 0.094
0.005	20,000	0.503 ± 0.070	0.503 ± 0.074	0.490 ± 0.095	0.502 ± 0.084
0.010	100	1.766 ± 1.368	1.675 ± 1.076	2.071 ± 1.584	1.846 ± 1.285
0.010	200	1.416 ± 0.907	1.388 ± 0.824	1.480 ± 1.067	1.460 ± 0.869
0.010	500	1.097 ± 0.489	1.123 ± 0.478	1.152 ± 0.478	1.146 ± 0.512
0.010	1,000	1.027 ± 0.326	1.027 ± 0.296	1.087 ± 0.383	1.087 ± 0.320
0.010	5,000	1.005 ± 0.159	1.006 ± 0.153	1.023 ± 0.174	1.009 ± 0.173
0.010	10,000	1.009 ± 0.140	1.002 ± 0.139	1.006 ± 0.140	0.995 ± 0.135
0.010	20,000	1.007 ± 0.097	1.005 ± 0.110	1.005 ± 0.129	0.985 ± 0.124

A.6 Empirical calibration across datasets and teachers

We evaluate the random-pairing reference calibration on the COCO Captions and Flickr30k training splits using pretrained CLIP ViT-B/32 and ViT-B/16 teachers in evaluation mode. We retain one caption per image. For each of 100 seeds (41–140), we randomly repartition a fixed cached feature pool into image-disjoint calibration and held-out pools, and independently draw one uniform caption permutation within each pool. Fixed-point positions are removed without modifying the permutation. The calibration pool contains 20,064 image–caption records; M denotes the number of retained permuted scores used for the threshold, rather than the number of independent observations. We take the first M retained scores and set Q to their linearly interpolated empirical $(1 - \alpha)$-quantile. We measure the fraction of held-out random-pairing scores satisfying $S > Q$, using 20,000 scores per seed on COCO and 8,000 on Flickr30k.

Table 7 shows larger mean exceedance and variability at small calibration budgets. With $M = 20{,}000$, the mean exceedance ranges from 0.490% to 0.503% at $\alpha = 0.005$ and from 0.985% to 1.007% at $\alpha = 0.01$ across the four settings. Increasing M also reduces the observed resampling variability. These measurements provide empirical calibration diagnostics within the tested feature pools; they do not establish exact α control, conditional guarantees for an observed calibration sample, or semantic precision. In particular, the permuted scores share records, and the resampling repetitions reuse a finite cached pool. This diagnostic uses a global permutation within each pool and is separate from the minibatch calibration used in the training experiments.

A.7 Ablation on thresholding strategies

PTA uses a calibrated rule to select additional in-batch positives based on weak similarities. Here we compare several thresholding strategies under an identical weak-to-strong setting, and report their average zero-shot accuracy (Table 5). The variants are: (i) *SFT.* Standard supervised fine-tuning on COCO without any weak-similarity filtering, (ii) *FT.* A single global cosine-similarity cutoff is tuned on COCO validation and reused for all batches, (iii) *Naive percentile.* The cutoff is set to the 99th percentile of weak similarities on ground-truth COCO pairs and kept fixed during training, and (iv) *PTA*. Our method derives the cutoff from the permuted null distribution of weak similarities at a chosen significance level α, yielding a dataset- and teacher-adaptive threshold with explicit control over the tolerated false-alarm rate. Table 5 summarises the seven-dataset averages for these strategies. Both FT and naive percentile improve over the SFT baseline, showing that even simple heuristics can benefit from weak similarities. However, PTA achieves the highest average accuracy among the tested thresholding strategies in this setting.

A.8 Additional details on weak-teacher quality

Figure 3 in the main text summarises the teacher dropout study. Tables 9 and 10 give the per-task teacher and student accuracies, including Caltech101 [25] and SUN397 [45]. We vary inference-time dropout in CLIP-B/32 over $\{0, 0.1, 0.3, 0.5, 0.8\}$ while holding the CLIP-B/16 student architecture and training hyperparameters fixed. The calibration-only diagnostic below uses a separate protocol and does not attribute the student accuracy changes to one calibration metric.

A.8.1 Calibration diagnostics under teacher corruption

To complement the weak-teacher accuracy study, we conduct a separate calibration-only diagnostic with the pretrained CLIP-B/32 teacher. We vary its inference-time dropout rate over $\{0, 0.1, 0.3, 0.5, 0.8\}$, keeping the weights fixed. The teacher is first set to evaluation mode; modules with dropout are then selectively activated at the requested rate. Scores are 100 times the cosine similarity of normalized image and text embeddings.

We fix the first $5{,}064$ eligible COCO train2014 images in image-ID order, using the first annotated caption per image. For each of five paired seeds (41–45), we encode this fixed sample and draw one uniform global text permutation. We discard fixed points and retain the first $M = 5{,}000$ mismatched scores. This diagnostic uses the k-th order statistic as Q, where $k = \lceil (M+1)(1-\alpha) \rceil$; the two α settings reuse the same score vector. The order-statistic rule specifies the diagnostic threshold and does not establish an exact finite-sample guarantee for these dependent pooled scores.

We evaluate selection on a separate, fixed set of $5{,}120$ COCO val2014 images, arranged into ten seed-shuffled batches of size $B = 512$. The first caption per image defines each batch score matrix. The auxiliary selection rate is the fraction of its $B(B-1)$ off-diagonal scores strictly above Q, aggregated over the ten batches. Known-positive recall uses separate encodings of all $25{,}610$ annotated captions for these images to measure the fraction of corresponding image–caption scores strictly above Q. This raw-score recall is distinct from the unconditional inclusion of observed paired positives in the training mask. Across dropout conditions, images, captions, mismatch permutations, and batch orders are paired by seed. The fixed images are not resampled across seeds.

Table 8 shows that the threshold changes with teacher corruption, but not monotonically. At $\alpha = 0.01$, known-positive recall falls from 91.51% without dropout to 1.16% at dropout 0.8, while the off-diagonal selection rate remains 1.12%. Thus, recalibration does not restore matching information lost under severe teacher corruption. The auxiliary selection rate measures selection yield, not the semantic precision of the selected pairs. These calibration-only results complement the student accuracy study; they do not attribute its accuracy changes to a single calibration metric.

Table 8: Calibration-only weak-teacher corruption diagnostic. Entries are mean ± sample standard deviation across five paired seeds. Auxiliary selection and known-positive recall are percentages; their standard deviations are in percentage points. No student is trained in this experiment.

α	Dropout	Q	Auxiliary selection (%)	Known-positive recall (%)
0.005	0.0	27.698 ± 0.623	0.589 ± 0.145	80.72 ± 5.44
0.005	0.1	27.509 ± 0.537	0.581 ± 0.139	74.49 ± 4.96
0.005	0.3	26.827 ± 0.376	0.540 ± 0.122	46.70 ± 5.12
0.005	0.5	27.756 ± 0.157	0.510 ± 0.127	4.88 ± 0.88
0.005	0.8	25.418 ± 0.178	0.633 ± 0.146	0.65 ± 0.16
0.010	0.0	25.990 ± 0.599	1.143 ± 0.255	91.51 ± 2.49
0.010	0.1	26.110 ± 0.567	1.056 ± 0.238	85.68 ± 3.64
0.010	0.3	25.741 ± 0.290	1.059 ± 0.186	60.36 ± 3.36
0.010	0.5	27.269 ± 0.133	1.041 ± 0.193	7.63 ± 0.98
0.010	0.8	24.909 ± 0.101	1.119 ± 0.162	1.16 ± 0.19

A.9 Ablation on data scaling

To evaluate whether PTA acts as a data-efficient refinement mechanism rather than merely adding computation, we conduct a scaling experiment on CC3M [40]. We conducted a pretraining on a CLIP-B/16 student for one epoch under identical hyperparameters, varying only the number of CC3M image–text pairs used: 1.0M, 2.0M, and 3.0M. All baselines and PTA use the same weak teacher (CLIP-B/32) and the same training setup; the only difference is how additional positive signals are extracted. Figure 7 shows the average accuracy over downstream benchmarks as a function of the number of CC3M pairs. The supervised baseline exhibits clear diminishing returns: increasing the dataset from 1.0M to 2.0M pairs yields a $+2.91$ point gain, whereas the next million (from 2.0M to 3.0M) adds only $+1.40$ points. On the same 3M examples, PTA improves performance by a further $+1.25$ points over the supervised baseline. These results highlight that PTA yields non-trivial improvements even when the training data is fixed and relatively small. Rather than requiring additional multimodal pairs, PTA provides a statistically calibrated way to reuse in-batch negatives as auxiliary positives, improving alignment in a data-efficient manner.

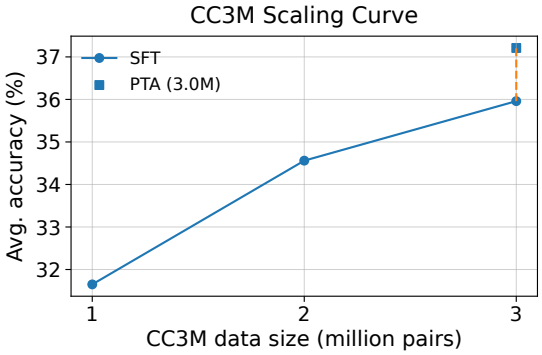


Figure 7: PTA data efficiency on CC3M.

A.10 PTA-Cache for efficient training

To eliminate the dominant weak-model overhead, we implement a cached variant of PTA that precomputes weak embeddings once and reuses them throughout training, as the weak model is fixed and their embedding should be fixed as well. Concretely, before fine-tuning we run the weak CLIP encoders over the entire training set to compute and store embeddings. During training, for each mini-batch we gather the corresponding cached vectors by index, transfer them to GPU, and compute weak similarities via a single matrix multiplication; all permutation-based thresholding and positive selection is then applied directly without any additional encoder forward pass. In practice this removes the $\sim$1.3–1.4$\times$ slowdown of PTA-online (which adds one extra CLIP forward per step) and yields per-step throughput within 1–5% of the SFT ceiling, since the remaining costs are lightweight.

Table 9: Zero-shot classification accuracy (%) of CLIP-B/32 with different dropout rates during inference. Higher dropout rates simulate progressively weaker models.

Dropout	Caltech101	CIFAR10	CIFAR100	DTD	Flowers102	Food101	SUN397	ImageNet	Average
0.0	59.58	58.38	42.66	36.52	60.60	88.71	80.55	61.00	61.00
0.1	52.77	50.22	36.06	39.74	51.66	89.26	77.36	56.72	56.72
0.3	28.31	21.74	26.17	26.36	38.93	81.08	50.43	39.00	39.00
0.5	7.32	4.44	10.96	12.51	10.02	33.37	13.09	13.10	13.10
0.8	0.11	0.50	2.07	9.77	1.06	10.32	0.97	3.54	3.54

Table 10: Classification accuracy (%) after fine-tuning CLIP-B/16 using PTA with CLIP-B/32 weak models at different dropout rates. All use identical hyperparameters on COCO Captions.

Weak Dropout	Caltech101	CIFAR10	CIFAR100	DTD	Flowers102	Food101	SUN397	ImageNet	Average
0.0	64.14	60.40	44.57	50.51	66.45	89.38	83.77	65.60	65.60
0.1	64.13	59.47	44.52	50.94	66.58	89.36	83.98	65.57	65.57
0.3	62.80	56.06	44.31	51.17	65.56	89.47	83.04	64.63	64.63
0.5	58.74	47.10	44.73	49.75	62.43	88.72	77.12	61.23	61.23
0.8	56.84	47.78	40.53	46.51	60.81	87.74	78.30	59.79	59.79

Table 11: **Recorded training allocations for Table 1.** Each job uses one GPU. Counts in parentheses indicate how many of the five seeds ran on each GPU model. Elapsed time is the per-job range; GPU-hours sum the five allocations, using unrounded times.

Method	Runs	GPU mix across runs	Elapsed/run (h)	GPU-hours
SA	5	L40S (5)	1.08–1.35	5.99
VPL	5	L40S (5)	1.34–1.98	8.15
FT	5	L40S (4), H100 (1)	0.99–1.62	6.81
MD	5	L40S (5)	1.43–1.74	7.99
WSA	5	L40S (5)	1.44–1.67	7.78
PTA	5	A100 (2), H100 (3)	1.00–2.14	7.24
Total	30	Mixed hardware	–	**43.97**

A.10.1 Compute resources and recorded runtimes

We recover hardware and elapsed times from allocation records and the corresponding training logs and environment records. The 30 recorded training jobs underlying Table 1 each use one GPU, eight CPU cores, and an 80 GB host-memory allocation. The GPU models are NVIDIA A100 80GB PCIe, H100 80GB HBM3, and L40S 48GB; these memory values denote nominal device capacities. The five seeds per method follow the COCO protocol in the main text, with batch size 512, up to 20 epochs, and validation-based early stopping.

Timing scope. Job time is the allocation duration, including setup, calibration where applicable, training, scheduled validation, and saving; queueing and separate downstream evaluation are excluded. GPU-hours equal allocated GPUs multiplied by elapsed hours. The 30 jobs total 43.97 GPU-hours (4.06 A100, 4.17 H100, and 35.74 L40S). PTA's Trainer runtime, excluding initialization and calibration but including validation, is 53.93–87.80 minutes; its five allocations sum to 7.24 GPU-hours. Hardware and stopping epochs differ, so these are cost records rather than controlled speed comparisons.

Additional cost examples. Two separate one-epoch CC3M runs use one A100 80GB GPU, eight CPU cores, 160 GB host memory, batch size 4096, and seed 42, updating the full vision tower and last two text blocks; PTA uses $\alpha = 0.005$. SFT and PTA allocations are 1:38:14 and 2:06:13, respectively (3.74 A100 GPU-hours combined). This configuration differs from the scaling runs in Figure 7.

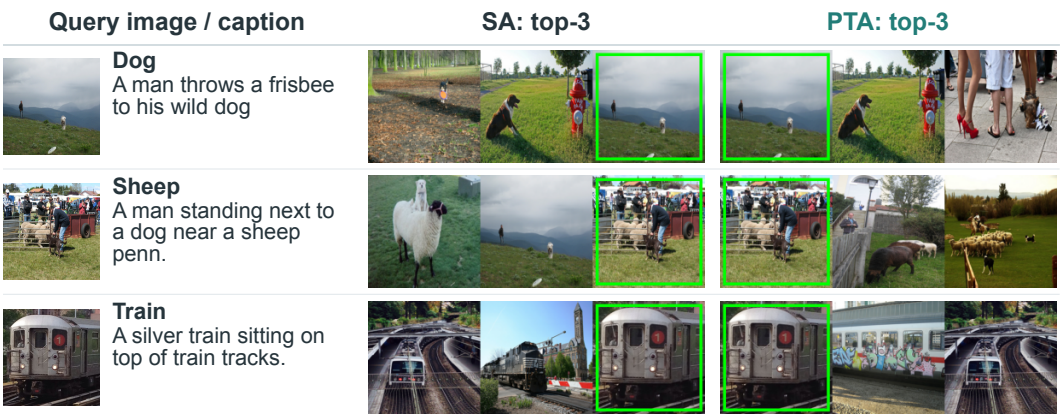


Figure 8: **Qualitative retrieval.** Three selected examples. Within each top-3 strip, ranks run left to right; green outlines mark the matching image, ranked third by SA and first by PTA.

A.11 Visualisation

The main text presents Grad-CAM visualisations in Figure 4. Here we provide complementary retrieval examples and the full text queries used in the visualisations.

A.11.1 Qualitative retrieval

Figure 8 compares the top-three retrieval results for three selected queries. Green outlines identify the matching image, ranked third by SA and first by PTA in each example.

A.11.2 Queries for the main-text Grad-CAM visualisations

The full text queries in Figure 4, from top to bottom, are:

1. **Airplane:** A small white plane sitting on top of an airport runway.
2. **Sheep:** Some very cute sheep in a big field.
3. **Motorcycle:** A parked motorcycle sitting on the side of the road.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction accurately reflect the contributions.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: Limitation is discussed in the paper.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

   Answer: [Yes]

Justification: Yes, we have included it.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification:We have included detailed implementation details.

   Guidelines:

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

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

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

Answer: [Yes]

Justification: We will release code upon publication.

Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: We have included it.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: It has been reported.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Appendix A.10.1 reports verified hardware, GPU and host-memory allocations, and recorded times for the 30 training jobs underlying Table 1, plus two historical CC3M cost examples.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: We reviewed the NeurIPS Code of Ethics and confirm this research complies.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: By providing a computationally efficient and statistically robust mechanism for weak-to-strong alignment, our framework lowers resource barriers to training advanced multimodal systems, though care is needed to ensure promoting uncurated auxiliary pairings does not inadvertently amplify existing societal biases present in the training data.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [N/A]

    Justification: This work studies contrastive adaptation of existing image–text encoders using existing datasets and does not release data or models that pose a high risk for misuse.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: Asset sources, versions, and available licenses and usage terms are documented in the project repository's third-party notices.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [N/A]

    Justification: We do not introduce new datasets or models.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: We do not involve it.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: We do not.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [Yes]

    Justification: Polish scripts.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.