CLIP Models Generalize Less Than Compositional Benchmarks Suggest

Shuman Peng [1] **Arnas Uselis** [2] **Darina Koishigarina** [2] **Martin Ester** [1] **Seong Joon Oh** [3]

Abstract

Compositional benchmarks track progress on CLIP-based compositional reasoning. Each new method reports higher scores than the last, but it is unclear whether the improvements reflect generalization to novel bindings or memorization of bindings already seen during training. To find out, we run two analyses: a synthetic ground-truth study with curated fully-seen, partially-unseen, and fully-unseen binding splits; and extension to three real compositional benchmarks (ARO VG-A, BiVLC, VisMin) using binding-overlap with COCO as a proxy for the alignment-training distribution. On the synthetic dataset, accuracy drops monotonically from fully-seen to fully-unseen across nine CLIP backbones. On ARO VG-A, positive captions overlap COCO bindings nearly twice as often as their attribute-swapped negatives (79.8% vs. 41.8%); only 1.2% of samples have no COCO-overlapping bindings. Restricting evaluation to the shortcut-free *seen* split, where positive and negative captions are equally COCO-familiar, reorders the top of the leaderboard relative to the full benchmark. The accuracy drop from seen to unseen bindings broadly replicates on BiVLC and VisMin, though with greater noise. Compositional benchmarks should report performance on these shortcut-free splits; otherwise reported improvements likely overstate how much CLIP has learned to bind.

1. Introduction

CLIP (Radford et al., 2021) and similar dual-encoder vision-language models are widely used for image-text retrieval, zero-shot image recognition, and the visual backbone of MLLMs (Liu et al., 2023). However, they struggle with compositional generalization – the ability to understand new compositions of familiar parts. CLIP recognizes parts of a caption but fails to bind attributes to the right objects. Prior work calls this a "bag of words" behavior (Yuksekgonul et al., 2023).

Several recent methods improve CLIP on compositional benchmarks (ARO, Winoground, SugarCrepe) through hard-negative training objectives, caption augmentation, or architectural changes (Yuksekgonul et al., 2023; Vani et al., 2024; Patel et al., 2024; Peleg et al., 2025).

But what does a higher benchmark score actually mean? When a model picks "red car next to wooden table" over the attribute-swapped "wooden car next to red table," it could be *generalizing* (composing attributes and objects, including pairings it has never seen) or *memorizing* (preferring "red car" and "wooden table" because they appeared in training, while "wooden car" did not). Prior work has examined related forms of training-data familiarity, but each at a coarser unit than what these compositional benchmarks actually measure. For example, how often a *single* concept or a pair of concepts appears in pretraining (Udandarao et al., 2024; Wiedemer et al., 2025; Qu & Xie, 2025). But whether the specific *(attribute, object) bindings* in a given evaluation sample (e.g., "red car," "wooden table") appear in the model's image-text training pairs remains unmeasured.

In this work, we ask: *are compositional benchmark scores reflecting **generalization** to unseen attribute and object bindings or **memorization** of familiar bindings?*

We investigate this question with a controlled synthetic dataset and three real-world compositional benchmarks. First, we construct fully-seen, partially-unseen, and fully-unseen splits on the synthetic PUG:SPARE dataset (Koishigarina et al., 2026) based on whether the bindings are seen in the training set. Second, we introduce a per-sample, paired *binding-overlap* measurement on real compositional benchmarks (ARO VG-A, BiVLC, VisMin) against actual image-text training pairs (COCO). This binding-overlap measurement allows us to create evaluation splits consisting of seen or unseen bindings. On our PUG:SPARE splits, model performance drops monotonically as the number of unseen bindings per sample increases. On real benchmarks, the same seen-to-unseen performance drop broadly appears, and on ARO VG-A, restricting to the seen split (free from the binding-familiarity shortcut) reorders the top-performing

[1]School of Computing Science, Simon Fraser University [2]Tübingen AI Center, University of Tübingen [3]KAIST AI. Correspondence to: Shuman Peng <shumanp@sfu.ca>.

Accepted to the 1st Workshop on Combining Theory and Benchmarks, CTB@ICML 2026, Seoul, South Korea. Copyright 2026 by the author(s).

models relative to the full benchmark (Figure 1).

Our contributions:

- **BindSplit protocol**: an evaluation protocol that partitions a compositional benchmark into seen, partially-unseen (mixed), and unseen splits based on whether each evaluation sample's (attribute, object) bindings appear in the model's training data. We apply BindSplit in two settings: a synthetic dataset (PUG:SPARE), where we control exactly which bindings appear in alignment training or fine-tuning, and three real-world benchmarks (ARO VG-A, BiVLC, VisMin), where we measure binding overlap against COCO as a proxy for the training distribution. The resulting splits isolate compositional generalization from training-set familiarity.
- **Benchmark audit**: on ARO VG-A, the (attribute, object) bindings in positive captions are more familiar than those in negative captions. Bindings in positive captions appear in training data nearly twice as often as those in their attribute-swapped negative captions (79.8% vs. 41.8%); we name this asymmetry the *binding-familiarity shortcut* and show it is distinct from the plausibility shortcut identified in prior work (Hsieh et al., 2023). Across all three real benchmarks (ARO VG-A, BiVLC, VisMin), fewer than 2.5% of evaluation samples consist entirely of bindings that are unseen.
- **Empirical findings**: performance drops from seen to unseen splits on PUG:SPARE for every backbone, and broadly replicates on the three real benchmarks. On ARO VG-A, restricting to the binding-familiarity shortcut-free seen split (where bindings in positive and negative captions are equally familiar) reorders the top-performing models relative to the full-benchmark leaderboard.
- **Diagnosis**: on ARO VG-A, shortcut exploitation is related to pretraining scale. The three large-pretraining backbones (CLIP OpenAI, OpenCLIP LAION-2B, NegCLIP COCO-ft) lose the most accuracy when the asymmetry is removed (a ∼4–7 pp full-to-seen performance drop), whereas the smaller CC12M-pretrained backbones lose little or nothing. Current leaderboards therefore reward exploiting familiar bindings over generalizing to novel ones.

Section 2 surveys the three relevant threads (methods for CLIP compositionality, benchmark audits, and training-data studies) and identifies the gap our work fills. Section 3 formalizes BindSplit: the definitions, per-sample paired overlap measurement, and split construction. We first validate the protocol on a controlled synthetic dataset (Section 4) where unseen bindings are held out by construction, then replicate the analysis on three real-world benchmarks using COCO as a proxy for the training distribution (Section 5). Section 6 discusses findings and recommends benchmark practices.

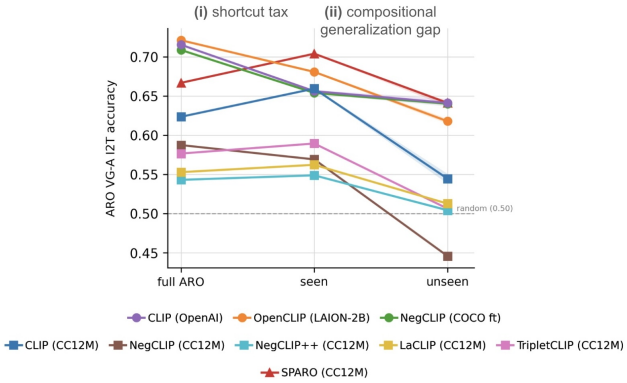

Figure 1. **Compositional accuracy falls in two steps across BindSplit, and the full-benchmark leaderboard reorders once the binding-familiarity shortcut is removed.** Image-to-text accuracy on the full ARO VG-A set and its shortcut-free *seen* and *unseen* splits, for nine COCO-finetuned backbones. **(i) Shortcut tax:** removing the positive caption's binding-familiarity advantage (full → seen, where both captions are equally COCO-familiar) costs the large-pretraining backbones 4.0–5.9 pp, while smaller CC12M backbones, which rely less on the shortcut, stay flat or score slightly higher on seen. **(ii) Compositional generalization gap:** moving from seen to unseen bindings drops accuracy further for every backbone. Because the drop is uneven, model ranking reorders between the full benchmark and the shortcut-free seen split. Full-benchmark scores therefore overstate performance on shortcut-free (seen and unseen) bindings. Lines: mean across 3 finetuning seeds; ribbons: [min, max]. Same data as Figure 4 (§5.3).

2. Related Work

Three research threads are relevant. Methods improve CLIP's compositional understanding, but benchmark scores conflate generalization with training-set familiarity, motivating our analysis of what these scores actually measure. Existing audits of compositional benchmarks identify linguistic artifacts (plausibility of negative captions, annotation ambiguity) without examining training-distribution familiarity at the binding level in multi-object settings. Training-data studies connect concept frequency and word co-occurrence to CLIP zero-shot generalization, but not at the (attribute, object) binding unit on multi-object compositional benchmarks. Our per-sample, paired binding-overlap closes that gap.

Methods improving VLM compositionality. Several methods improve CLIP's compositional understanding. The most common approach introduces hard negative training examples to sharpen attribute-object discrimination; for in-

stance, by swapping syntactic elements within captions, synthetically generating both negative captions and images, or constructing hard negatives through word-level caption recombination (Yuksekgonul et al., 2023; Patel et al., 2024; Peleg et al., 2025; Doveh et al., 2023). A separate line of work diversifies training captions through language model rewrites to provide richer supervision (Fan et al., 2023). Architectural and inference-time methods take yet another approach: adding a slot-based read-out mechanism for concept disentanglement, incorporating component-level contrastive objectives, or refining token representations at inference time (Vani et al., 2024; Kargi et al., 2026; Zhang et al., 2025). Each reports improvements on compositional benchmarks. However, benchmark scores conflate compositional generalization with training-set familiarity: standard evaluations do not separate seen bindings from unseen ones.

Evaluating compositional reasoning in vision-language models. Benchmarks for compositional understanding in vision-language models follow a standard paradigm: a model must match an image to its caption when the distractor differs by compositional attributes, such as attribute-object binding, relations, or word order (Yuksekgonul et al., 2023; Hsieh et al., 2023; Burapacheep et al., 2024; Miranda et al., 2024; Awal et al., 2024). Several recent studies identify flaws in some of these benchmarks, such as visual difficulty, annotation ambiguity (Diwan et al., 2022), and implausibility of negative captions (Hsieh et al., 2023). However, these audits focus on linguistic or annotation artifacts. They miss a more fundamental issue: whether the specific (attribute, object) pairs in evaluation captions appeared in the model's image-text pre-training data. Existing evaluations do not account for this training-set familiarity, nor do prior audits check for it. In this work, we address both. We evaluate compositional methods under the BindSplit protocol, which partitions benchmarks by binding familiarity to measure compositional generalization.

Training-data quality and familiarity influence CLIP's generalization behavior. CLIP's behavior is shaped by its training data composition at multiple levels of granularity. At the individual concept level, how often a concept appears in training determines recognition quality, and pre-training datasets exhibit long-tailed concept distributions that concentrate mass on a small set of frequent concepts (Fang et al., 2022; Nguyen et al., 2022; Mayilvahanan et al., 2024; Udandarao et al., 2024; Parashar et al., 2024). Beyond individual concepts, co-occurrence and combinatorial coverage govern compositional behavior: accuracy on novel object combinations is predicted by individual object frequency, concept pair co-occurrence PMI strongly predicts zero-shot accuracy, and the number of distinct concept combinations covered in training matters more for generalization than total data quantity (Wiedemer et al., 2025; Qu & Xie, 2025; Uselis et al., 2025). Closest to our work, Qu & Xie (2025) compute per-sample word-co-occurrence PMI and correlate it with CLIP accuracy on standard zero-shot recognition tasks. We share their per-sample, text-side stance but apply it to compositional evaluation in multi-object settings, and narrow the unit of analysis from arbitrary word co-occurrence to (attribute, object) bindings — the unit that swap-attribute benchmarks (§3.1) test. To do so, we need a protocol that measures per-sample, paired binding-overlap with the training distribution and partitions evaluation data accordingly. We introduce this protocol, BindSplit, in the next section.

3. Setup: the BindSplit protocol

To separate familiarity-driven performance from true compositional generalization, we need to measure which parts of an evaluation sample the model has already encountered during training. We therefore introduce *BindSplit*, a protocol that measures per-sample, paired binding-overlap with the training distribution and partitions evaluation data into seen, mixed, and unseen splits. This section defines the core concepts, gives a conceptual overview of BindSplit, describes the overlap measurement, and lists the models evaluated.

3.1. Definitions

Bindings. A *binding* is an ordered pair `(attribute, object)` describing an attribute applied to an object in a caption. We denote a binding by $b = (\text{attribute}, \text{object})$. For example, "the red car" yields the binding `(red, car)`; "the blue sky and the white wall" yields `(blue, sky)` and `(white, wall)`.

Vision-language compositional task. A specific task within vision-language compositional reasoning is identifying the correct attribute-object bindings. In compositional benchmarks (e.g., ARO (Yuksekgonul et al., 2023), SugarCrepe (Hsieh et al., 2023)), this is called `swap-attribute`. Each sample consists of an image and two captions: a *positive* caption describing the image and a *negative* caption formed by swapping the attributes between two object slots. The captions share the same objects and attributes, differing only in their bindings. A model solves the task by scoring the positive caption higher than the negative one, which requires correctly binding each attribute to its object. While sample structure varies across benchmarks, the two most common formats are the single-image version described above and a two-image variant: the same setup but with a second image generated from the negative caption (making the negative caption correct for that image). The task is then to identify the correct caption for each image and vice versa (Thrush et al., 2022).

Evaluation metrics. We report three metrics. **R@1 retrieval** (Recall@1) is the fraction of samples where the

correct caption is the top-ranked result among all candidate captions for a given image. **Binary accuracy** is the fraction of samples where the model assigns a higher similarity score to the positive caption than to the negative caption; this applies to the single-image two-caption task used in ARO VG-A. **Group accuracy** applies to the two-image two-caption task (BiVLC (Miranda et al., 2024), VisMin (Awal et al., 2024)) and follows the Winoground formulation (Thrush et al., 2022). Let $s_{ij} \triangleq \mathrm{sim}(\mathbf{I}_i, \mathbf{C}_j)$, where $(\mathbf{I}_0, \mathbf{C}_0)$ and $(\mathbf{I}_1, \mathbf{C}_1)$ are the matched image and caption pairs. A sample is correct if and only if both image-to-text retrieval and text-to-image retrieval are correct:

$$\underbrace{s_{00} > s_{01} \wedge s_{11} > s_{10}}_{\text{image-to-text}} \wedge \underbrace{s_{00} > s_{10} \wedge s_{11} > s_{01}}_{\text{text-to-image}}.$$

3.2. BindSplit at a glance

BindSplit partitions evaluation samples by how many of their bindings appear in the training set. This separates samples where a model can rely on training-set familiarity from those where it must generalize to unseen bindings. Consider a sample with positive caption "blue sky and white wall" and negative caption "white sky and blue wall". If all four bindings from its two captions – `(blue, sky)`, `(white, wall)`, `(white, sky)`, `(blue, wall)` – appear in training, the sample is seen. If none appear, it is unseen. If some appear and others do not, it is partially-unseen (or mixed).

Figure 6 shows the idea on a grid of colors and animals. Later sections in this paper apply BindSplit to two concrete settings: synthetic PUG:SPARE with a controlled held-out block (§4) and the real-world ARO (VG-A), BiVLC, and VisMin benchmarks, whose bindings we check against a proxy-training set, MS-COCO (§5). We formalize this measurement in §3.3.

3.3. Binding-overlap measurement

Per-sample, paired binding-overlap. Let T be a reference training set and r a matching rule; together they determine, for any binding b, an *overlap label* $\ell_{T,r}(b) \in \{\text{SEEN}, \text{UNSEEN}\}$. For a swap-attribute evaluation sample with positive-caption bindings (b_1^+, b_2^+) and attribute-swapped negative-caption bindings (b_1^-, b_2^-), the *per-sample, paired binding-overlap* is the ordered tuple

$$\Lambda_{T,r}(s) = \big(\ell_{T,r}(b_1^+), \ell_{T,r}(b_2^+), \ell_{T,r}(b_1^-), \ell_{T,r}(b_2^-)\big).$$

Concretely, on the example above, subscripts index the object slot: b_1^+ = `(blue, sky)` and b_1^- = `(white, sky)` share slot 1 (`sky`) with swapped attributes, and b_2^+ = `(white, wall)` and b_2^- = `(blue, wall)` share slot 2 (`wall`). The unit of measurement is the individual sample s (*per-sample*). A sample's positive and negative binding labels (each SEEN or UNSEEN, computed independently per binding) stay bundled in one tuple (*paired*); for every sample, we can therefore tell which of its bindings are seen or unseen. We use T = COCO (train) for real benchmarks and T = PUG:SPARE (train) for the controlled setting; the matching rule r is the lemmatized rule defined below.

Matching rule. For each binding extracted from an evaluation caption, we look for an exact match in a training caption under *lemmatized match*: we singularize the object slot (e.g., "cars" → "car") and leave the attribute slot unchanged. A binding is *seen* if it matches any training caption under this rule, and *unseen* otherwise. Appendix B.3 gives the full extraction and matching procedure, including how multi-attribute phrases are handled.

Why per-sample pairing is the key structure. Computing overlap separately for the positive and the attribute-swapped negative caption exposes a *familiarity asymmetry* between the two. When a positive caption has high overlap while its swapped negative has low overlap, a model can select the more familiar caption without performing compositional reasoning. Aggregate measures, such as concept-frequency analyses (Udandarao et al., 2024; Wiedemer et al., 2025), average over samples and wash out this per-sample pairing, making the asymmetry invisible to those measures. §5.2 quantifies this asymmetry on the ARO benchmark.

3.4. Models evaluated

We evaluate the BindSplit partitions on nine CLIP ViT-B/32 backbones spanning two pretraining scales. **Large-scale**: OpenAI CLIP (Radford et al., 2021), OpenCLIP (LAION-2B) (Ilharco et al., 2021), and NegCLIP COCO-ft (Yuksekgonul et al., 2023) (fine-tuned from OpenAI CLIP on COCO). **CC12M-scale (smaller):** vanilla CLIP, SPARO (Vani et al., 2024), LaCLIP (Fan et al., 2023), NegCLIP (Yuksekgonul et al., 2023), NegCLIP++ (Patel et al., 2024), and TripletCLIP (Patel et al., 2024). We evaluate the backbones in two settings: (1) freezing the backbone and training a linear image-text alignment layer using the LABCLIP protocol (Koishigarina et al., 2026), and (2) fine-tuning the backbones.

LABCLIP linear alignment protocol. LABCLIP (Koishigarina et al., 2026) trains a linear layer on frozen text embeddings with the same contrastive loss as in the original CLIP. We train one linear layer on the text embeddings. Because the backbone is frozen, the only thing that differs between models is their pretrained representations. We train this alignment layer on paired image-caption data. For synthetic experiments, that data is the training set of PUG:SPARE (§4). For real-benchmark experiments (ARO

VG-A, BiVLC, VisMin), it is MS-COCO (training set)[1]. The same alignment training dataset defines which bindings count as *seen* in our overlap measurement.

Backbone fine-tuning. We fine-tune each backbone following a similar protocol to NegCLIP (Yuksekgonul et al., 2023), using paired image-caption data with hard-negative captions and the corresponding hard-negative augmented contrastive loss, but without hard-negative images. The training sets are the same as those used in the LABCLIP setting.

4. Controlled study on PUG:SPARE

4.1. PUG:SPARE setup: controlled 3×3 holdout

We first validate BindSplit in a controlled setting where we have control over the ground-truth training distribution and unseen bindings are held out from alignment training and fine-tuning by construction. We use the synthetic PUG:SPARE dataset (Koishigarina et al., 2026), which provides photorealistic images of animals in different backgrounds and positions with controlled color–animal bindings. Each image shows two animals (e.g., "a red elephant and a blue zebra") with caption describing which color binds to which animal. The full color–animal grid is 8 colors $\times$ 12 animals $=$ 96 bindings, and the dataset is large enough that every binding appears many times in the training set by default.

To create a controlled testbed for compositional generalization, we hold out a 3×3 block of (color, animal) bindings from the alignment training set.[2]

The block is designed around two principles for a shortcut-free evaluation. First, the positive and negative captions must be symmetric in familiarity — neither should be systematically more or less familiar than the other. The model should not be able to pick the correct caption by rejecting a more familiar (or less familiar) foil. Within the 3×3 block, both the positive caption and its attribute-swapped negative contain only unseen bindings, satisfying this principle. Second, when captions mix seen and unseen bindings, the positive and attribute-swapped negative caption must contain the same number of seen bindings, otherwise one caption is more familiar than the other. Our held-out block design draws on the same combinatorial logic as the (n, k) framework of (Uselis et al., 2025), who study how training-set diversity and coverage of concept combinations affect compositional generalization. We extend this logic to construct evaluation splits that isolate compositional reasoning from binding familiarity.

[1]The 2014 version of COCO.

[2]We use 3 different seeds to generate the hold-out bindings in §4.2.

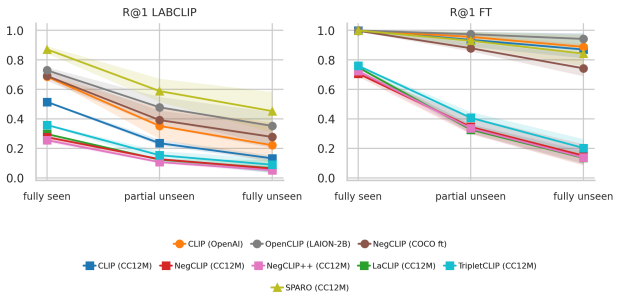

Figure 2. **Model performance degrades with unseen bindings.** Across all nine backbones in the LABCLIP alignment (left) and full backbone fine-tuning (FT) (right) settings, R@1 retrieval accuracy drops monotonically with the number of held-out attribute–object bindings per sample (0 = fully-seen, 1 = partially-unseen, 2 = fully-unseen). Lines: mean across 3 data-split seeds; ribbons: [min, max]. Random-chance order-invariant R@1 is 0.0271% $(2/7392)$, so all plotted values lie above chance.

The held-out bindings are fully absent from alignment training: no image–caption pair in the alignment (or fine-tuning) set contains any of them. This produces three disjoint evaluation splits based on how many held-out bindings appear in each sample:

1. **Fully-seen (0 held-out bindings):** both color–animal bindings in the sample are present in the training set.
2. **Partially-unseen (1 held-out binding):** one of the two bindings is held out from training, the other is seen.
3. **Fully-unseen (2 held-out bindings):** neither binding appears in the training set.

Because the held-out bindings are never observed during alignment training by construction, PUG:SPARE provides a controlled benchmark for compositional generalization to unseen bindings. Every individual color and animal in the held-out bindings appears in other (non-held-out) bindings elsewhere in the training set; only the specific *combination* of color and animal is unobserved.

4.2. Performance drops monotonically as unseen bindings increase

We evaluate nine LABCLIP-aligned CLIP backbones (Koishigarina et al., 2026) on samples containing zero, one, or two held-out bindings. For example, a sample showing a red elephant and a blue zebra where both (red, elephant) and (blue, zebra) are held out contains 2 unseen bindings (fully-unseen); if only (red, elephant) is held out, the sample has 1 unseen binding (partially-unseen). Figure 2 reports R@1 in two training settings: LABCLIP alignment (left) and full backbone finetuning (right). Under LABCLIP alignment, R@1 degrades monotonically with the number of held-out bindings per sample for all nine

backbones; the large-scale backbones lose 38–43 R@1 points from fully-seen to fully-unseen (e.g., CLIP OpenAI 68.2 → 25.3). Prior work (Koishigarina et al., 2026) reported that LABCLIP alignment improves R@1 on PUG:SPARE but evaluated only on seen bindings; our BindSplit evaluation disentangles the splits and shows that the alignment layer alone does not guarantee compositional generalization to unfamiliar binding combinations.

Full backbone finetuning raises R@1 on every split (fully-seen, partially-unseen, and fully-unseen) for all nine backbones relative to LABCLIP, but it does not narrow the seen-to-unseen gap uniformly. For the three large-scale backbones together with SPARO and vanilla CLIP-CC12M, finetuning brings fully-unseen R@1 to 74–94% and shrinks the seen-to-unseen gap to 6–25 points, allowing these backbones to generalize better to held-out bindings. For the four remaining CC12M backbones (LaCLIP, NegCLIP, NegCLIP++, TripletCLIP), finetuning raises fully-seen R@1 to 71–76% but lifts fully-unseen R@1 only to 14–20%, so the seen-to-unseen gap widens to 56–61 points. Even for these four backbones, fully-unseen R@1 stays far above the 0.0271% random-chance level (order-invariant R@1 over 7392 total captions): 5–9% under LABCLIP and 14–20% under finetuning. This gap reflects degraded rather than chance-level retrieval. Per-backbone R@1 across all three splits are reported in Table 2, with pairwise R@1 gaps in Table 3, and per-backbone R@1 under full finetuning in Table 4 (Appendix). Because every held-out color and animal also appears in non-held-out combinations during training (§4.1), this seen-to-unseen gap isolates generalization to the novel *combination* rather than to unfamiliar individual concepts.

5. Replication on real compositional benchmarks

5.1. Real-benchmark setup: binding-overlap with COCO

Having validated BindSplit on the controlled PUG:SPARE setting, we now apply it to real-world compositional benchmarks. We first apply BindSplit to ARO Visual Genome-Attribution (VG-A) (Yuksekgonul et al., 2023), a compositional benchmark for object-attribute binding. Each VG-A sample pairs an image with a correct caption and an attribute-swapped negative caption. The two captions share the same objects and attributes; only the binding differs. This two-binding, paired structure is unique to VG-A among ARO subsets and makes our per-sample, paired binding-overlap (§3.3) well-defined.

The synthetic PUG:SPARE setting holds out bindings from alignment training by construction; real benchmarks require a proxy for the training distribution. We use the training set of MS-COCO (2014) (Chen et al., 2015) as the alignment-training data for LABCLIP (Koishigarina et al., 2026) and as the backbone fine-tuning data; bindings appearing in COCO training captions are therefore seen during these alignment and fine-tuning stages. For each of the four bindings across the positive and negative captions (two per caption), we check for an exact lemmatized match in any COCO training caption. This partitions the $26{,}508$ single-attribute-filtered evaluation samples[3] into three buckets (counts reported in Table 1):

1. **Seen:** all four bindings appear in COCO.
2. **Mixed:** some bindings appear, others do not.
3. **Unseen:** none of the four bindings across the positive and negative captions appears in COCO.

These splits isolate two distinct kinds of familiarity difference. On the full evaluation set, positive captions are far more COCO-familiar than their attribute-swapped negatives; the *seen* split removes this asymmetry, leaving both captions equally familiar, while the *unseen* split removes binding familiarity altogether. We quantify the asymmetry in §5.2 and test whether models exploit it in §5.3.

Table 1. **Only less than 2.3% of samples in compositional benchmarks consist of unseen bindings.** Split composition for ARO VG-A, BiVLC, and VisMin based on binding overlap with COCO (train).

Split	ARO VG-A		BiVLC		VisMin	
	Count	%	Count	%	Count	%
full	26,508	100.0%	2,933	100.0%	2,084	100.0%
seen	4,329	16.3%	925	31.5%	822	39.4%
mixed	21,865	82.5%	1,361	46.4%	774	37.1%
unseen	314	1.2%	54	1.8%	48	2.3%
excluded	2,240	7.8%	593	20.2%	440	21.1%

COCO is the data we use for alignment and fine-tuning, not the backbones' pretraining corpus. Binding-overlap with COCO therefore captures familiarity from alignment or fine-tuning, not from pretraining. This proxy is imperfect. We discuss the implications for interpreting the unseen-split results below.

5.2. ARO VG-A is confounded by a binding-familiarity asymmetry

On the full ARO (VG-A) evaluation set (Figure 3, left), 79.8% of positive-caption bindings perfectly overlap the COCO training set, compared to only 41.8% of attribute-swapped negatives. 45% of negative caption bindings have no COCO overlap in the training set. On the mixed subset (samples where some but not all four bindings appear in

[3] $2,240$ samples were excluded because they did not contain single attributes so the matching is not as clean.

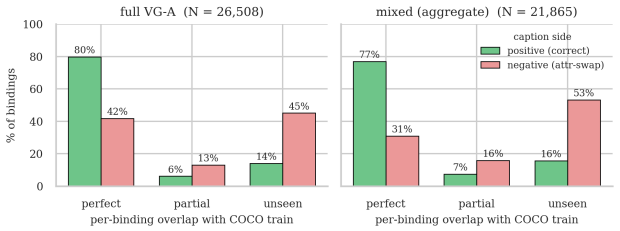

Figure 3. **Positive-caption bindings overlap COCO (train) far more than negative (attribute-swapped) bindings.** On the full VG-A evaluation set (left, N=26,508), 79.8% of positive-caption bindings perfectly overlap COCO, vs. only 41.8% of negative-caption bindings; conversely, 45% of negatives have no COCO overlap. The asymmetry is even larger on the mixed subset (right, N=21,865): 76.9% positive vs. 30.9% negative perfect overlap, with 53% of negatives unseen. The mixed subset contains samples whose captions mix seen and unseen bindings. A model can therefore select the correct caption from binding-familiarity alone, without compositional understanding.

COCO) (Figure 3, right), the gap widens: 76.9% positive vs. 30.9% negative are seen, with 53% of negatives unseen.

This asymmetry makes ARO VG-A a confounded test of composition: because the positive caption's bindings are systematically more familiar than the negative's, a model can score correctly from binding-familiarity alone, without compositional understanding. The benchmark therefore cannot, by construction, separate compositionality from familiarity. Of the 26,508 evaluation samples, 82.5% are mixed, where the positive caption has a strong familiarity advantage over the negative. Whether models actually exploit this confound as a shortcut is what we examine in §5.3.

This binding-familiarity asymmetry is distinct from the language-plausibility artifact identified by (Hsieh et al., 2023): plausibility is a text-side artifact (the negative caption is linguistically improbable), whereas binding-familiarity is a data-side artifact (the positive caption's bindings appear more often in training data).

5.3. Accuracy drops unevenly across BindSplit splits

Restricting evaluation to the splits without the asymmetry (seen and unseen) reveals three findings: (1) accuracy drops relative to the full set in two steps, when the familiarity asymmetry is removed (seen) and again on novel bindings (unseen); (2) the full benchmark reorders the top models relative to the shortcut-free seen split (statistically significant under fine-tuning), whereas the seen and unseen rankings do not differ within our statistical resolution; and (3) the drop magnitude varies with pretraining scale (per-backbone, per-split accuracies in Table 5).

Accuracy drops when the binding-familiarity asymmetry is removed. When both captions have equal COCO overlap (the seen split), the positive caption loses the $\sim 2\times$ binding-familiarity advantage it holds on the full split. For about half the backbones, this lowers accuracy relative to the full ARO set (Figure 4). We call this drop the *shortcut tax*: accuracy falls even though every binding remains familiar. Moving from seen to unseen, where the bindings are no longer familiar, produces a further drop, consistent with the PUG:SPARE results(§4.2). We call this second drop the *compositional generalization gap*. This two-step decline varies in magnitude across backbones, raising the question of whether model rankings are stable across splits.

Testing on the full benchmark rewards the familiarity shortcut: the leaderboard reorders on the shortcut-free seen split. The accuracy drop is uneven, so model ordering changes between splits (Figure 4). To test whether a reordering is statistically significant, we run pairwise McNemar tests—the standard test for whether two models differ on the same set of examples—on the per-sample image-to-text correctness of the nine *fine-tuned* backbones, with a Benjamini–Hochberg correction (FDR $q = 0.05$) across model pairs to control the false-discovery rate (the expected fraction of false positives among the pairs we flag as different). We count a pair's order as a *significant flip* only when its lead is significant on *both* splits with *opposite* directions, and summarize each flip by its combined p-value $p_{\text{flip}} = \max(p_{\text{A}}, p_{\text{B}})$ (the larger of the two single-split p-values).[4] *Between the two shortcut-free splits (seen vs. unseen), we find no statistically significant change in model ranking.* Only two adjacent pairs reverse order in the fine-tune setting (i.e., OpenCLIP LAION-2B vs. CLIP OpenAI, and TripletCLIP vs. LaCLIP) and both are significant on seen ($p = 0.004$, 0.007) yet far from significant on unseen ($p = 0.37$, 0.87, N=314). With only 314 unseen samples, the test cannot resolve the $\sim$1–3 pp gaps these flips depend on, so for these two splits we treat the reordering as illustrative.

The standard full benchmark, however, reorders model ranking relative to the shortcut-free seen split. Here the flip *is* statistically significant, on two pairs, both involving SPARO: CLIP OpenAI and NegCLIP (COCO-ft) outrank SPARO on full, but SPARO outperforms both on seen ($p_{\text{flip}} = 2.9 \times 10^{-7}$ and 4.3×10^{-8}, three-seed agreement). This is the shortcut at work: full is $\sim$82.5% ambiguous samples, where positive captions hold a $\sim 2\times$ binding-familiarity advantage (§5.2), so shortcut-exploiting backbones are rewarded and the non-exploiting SPARO (relatively flat across splits) ranks below them; removing the

[4]This significance analysis is run on the fine-tuned backbones, not the LABCLIP-aligned models. We use the two-sided mid-p variant of McNemar's test (null hypothesis: the two models are equally accurate); "mid-p" is a standard, slightly less conservative adjustment for discrete test statistics.

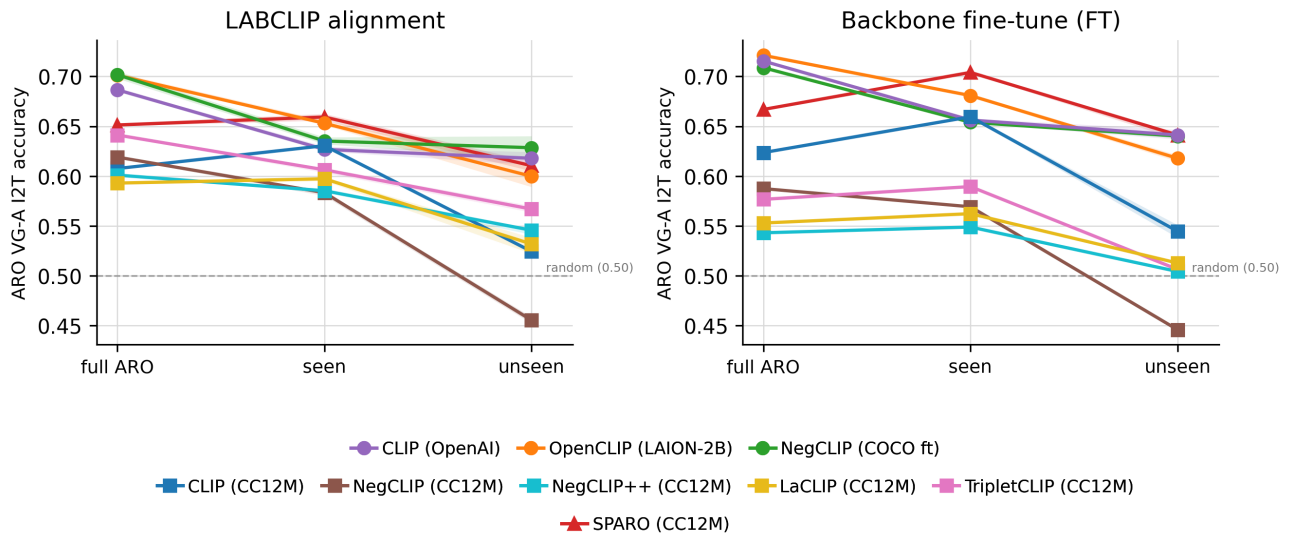

Figure 4. **Accuracy drops from seen to unseen bindings and rank reorders from the full ARO evaluation set to the seen split.** Accuracy across full ARO evaluation, seen, and unseen splits for each LABCLIP-aligned backbone (left) and fine-tuned backbone (right) (lines and ribbons show mean and [min, max] across 3 seeds). Accuracy drops from seen to unseen bindings for every backbone under both regimes. The full-to-seen drop, where the binding-familiarity shortcut is removed, holds for the large-pretraining backbones (CLIP OpenAI, OpenCLIP LAION-2B, NegCLIP COCO-ft) but not uniformly for the CC12M backbones, several of which are flat or slightly higher on seen, especially under finetuning. Model ordering also shifts between splits: moving from the full set to the shortcut-free seen split reorders the top models (statistically supported, §5.3), whereas the close adjacent flips between seen and unseen are not significant at the small unseen sample size and are shown only as illustrative.

asymmetry on seen lets SPARO outperform them. The full-versus-unseen transition shows no significant flip (smallest $p_{\text{flip}} = 0.094$).

Shortcut exploitation varies with pretraining scale. The three backbones with the largest pretraining corpora (OpenCLIP LAION-2B, CLIP OpenAI, NegCLIP COCO-ft) drop 4.8–6.6 percentage points from full to seen when the shortcut advantage vanishes (Table 6; 4.0–5.9 pp under full COCO finetuning, Table 8). CC12M-pretrained backbones with hard-negative augmentations (NegCLIP, NegCLIP++, TripletCLIP) follow a similar but weaker pattern under LABCLIP; under full COCO finetuning only NegCLIP retains a small full-to-seen drop, while NegCLIP++ and TripletCLIP show no full-to-seen drop (if anything a slight increase). CC12M-pretrained backbones without such augmentations (SPARO, vanilla CLIP CC12M, LaCLIP) stay roughly flat across full and seen splits, suggesting they rely less on the familiarity shortcut; under full COCO finetuning SPARO and CLIP CC12M score slightly *higher* on seen than full, the opposite of a shortcut tax. This scale-dependent pattern explains why the accuracy drop is uneven: larger-pretraining models rely more on the shortcut and suffer a larger drop when it is removed, consistent with the reordering illustrated above.

Implications of COCO proxy. This analysis uses COCO as a proxy for the training distribution, which is far smaller than the backbones' pretraining corpora. The unseen split is defined by absence from COCO, not from the larger pretraining corpora (CC12M, LAION-2B). Bindings labeled "unseen" by COCO may still appear in those larger corpora, so the unseen accuracies in Figure 4 can be *optimistic*: the true generalization gap is likely wider than what we observe.

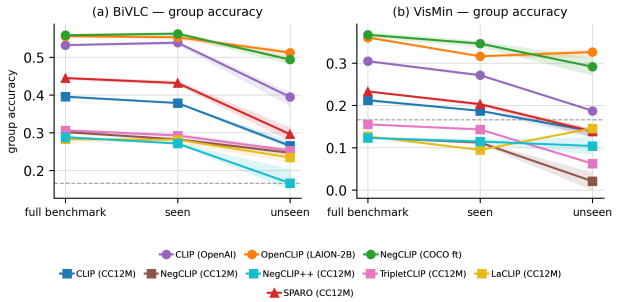

Figure 5. **BindSplit accuracy on BiVLC and VisMin across seen, mixed, and unseen splits.** For each benchmark (left: BiVLC, right: VisMin), grouped bars show accuracy by split across nine LABCLIP-aligned backbones. The seen-to-unseen accuracy drop—first observed on PUG:SPARE and ARO VG-A—replicates on both benchmarks. The dashed line shows random chance group accuracy (∼16.67%).

5.4. BindSplit gap replicates on BiVLC and VisMin

We extend the BindSplit analysis to the BiVLC (Miranda et al., 2024) and VisMin (Awal et al., 2024) vision-language compositional benchmarks. As with ARO VG-A, we measure whether each object-attribute pair in the captions appears in any COCO training caption and partition samples into seen (all four bindings appear in COCO), mixed (some appear), and unseen (none appear). Table 1 shows the split composition. The unseen fraction is again tiny: 1.8% on BiVLC and 2.3% on VisMin).

Figure 5 shows that the accuracy pattern broadly replicates: accuracy drops from seen to unseen for most backbones on both benchmarks. These results are noisier than ARO VG-A. Both benchmarks combine multiple task types (attribute swap, relations, counting), and restricting to the attribute-swap subset alone would leave fewer than five samples per split. For VisMin in particular, individual backbones deviate from the seen-to-unseen drop: OpenCLIP LAION-2B increases slightly from seen (31.6) to unseen (32.6), and LaCLIP-CC12M performs at or below random on *most* splits (its mixed-split group accuracy, 17.2, is marginally above the 16.67% chance line), making it uninformative. Despite these caveats, the majority of backbones follow the expected pattern, broadly replicating the ARO VG-A findings.

6. Discussion and Conclusion

The BindSplit gap is not an artifact of a single benchmark. In PUG:SPARE, where held-out bindings are absent from alignment training by construction, performance drops monotonically as the number of unseen bindings per sample increases, across all backbones. On the three real-world benchmarks, the unseen split is small (under 2.5% of evaluation samples) and the seen-to-unseen accuracy drop broadly replicates. ARO VG-A's single-image two-caption structure additionally exposes the binding-familiarity shortcut directly: positive captions overlap COCO bindings nearly twice as often as their attribute-swapped negatives (79.8% vs. 41.8%), and aggregate ARO scores are dominated by the mixed split where this asymmetry operates. Together, these results suggest that reported compositional benchmark scores likely overstate models' ability to generalize to novel bindings, and raise the question: are current benchmarks measuring compositional reasoning, or are they measuring a model's ability to exploit binding familiarity in the evaluation set? We recommend three practices to isolate this confound.

When constructing new benchmarks (or auditing existing ones): **(a)** measure per-sample, paired *binding-overlap* (or any compositional-unit overlap quantity natural to the benchmark) between evaluation captions and the model's alignment-training distribution; **(b)** partition the benchmark into seen, partially-unseen (mixed), and unseen splits by that overlap; and **(c)** report and compare performance on the asymmetry-shortcut-free sets, where positive and negative captions match in binding-familiarity by construction, in addition to the aggregate score. As §5.3 shows, on ARO VG-A the model ordering on the shortcut-free *seen* set shifts from the full-benchmark (aggregate-score) ordering.

Limitations. (1) Our binding-overlap measurement uses a *lemmatized match* procedure (object-side singularization, attribute slot untouched). We do not account for synonym or semantic overlap between bindings in real benchmarks. (2) Our seen versus unseen splits on the real compositional benchmarks are constructed based on overlap with the COCO training set, which we use for LABCLIP alignment. This is only a proxy for pretraining data for the evaluated CLIP backbones. Extending this to the CC12M pretraining set is left to future work. (3) Our results are at the object-attribute binding granularity ; scene-level relational compositionality (ordered relations, complex scenes with more than two objects) is beyond our current scope. (4) All evaluated backbones use the ViT-B/32 architecture. How these findings generalize to other architectures (ViT-L/14, ViT-g/14, SigLIP) remains open.

To answer the question we posed at the beginning: our findings suggest current compositional benchmark scores conflate *generalization* to novel bindings with *memorization* of familiar ones. The strongest evidence is coverage: only 1.2–2.3% of samples across the three benchmarks consist entirely of unseen bindings, so the benchmarks barely test novel-binding generalization. On ARO VG-A the binding-familiarity shortcut additionally inflates the scores of the models that exploit it—a single-digit drop ($\sim$6 pp full-to-seen, $\sim$9 pp mixed-to-seen) for those models, near zero for the rest—and restricting to the shortcut-free seen split reorders the top models relative to the full benchmark. The central implication is that reported progress in compositional reasoning likely overstates how much CLIP has learned to bind. Going forward, compositional evaluation should routinely report performance on splits that control for binding familiarity. This focuses the community's attention on what matters: composing novel combinations of familiar concepts.

Acknowledgements

The authors thank Hyeonbin Hwang, the STAI group members, and the anonymous reviewers for their valuable feedback. Shuman Peng was supported by the Mitacs Globalink Research Award (GRA) program. Arnas Uselis and Darina Koishigarina were supported by the Tübingen AI Center and International Max Planck Research School for Intelli-

gent Systems (IMPRS-IS). Seong Joon Oh was supported by the Institute for Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (RS-2019-II190075, Artificial Intelligence Graduate School Program (KAIST)).

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Awal, R., Ahmadi, S., Zhang, L., and Agrawal, A. VisMin: Visual minimal-change understanding. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2407.16772. Added: 2026-05-07.

Burapacheep, J., Gaur, I., Bhatia, A., and Thrush, T. ColorSwap: A color and word order dataset for multimodal evaluation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 1716–1726, 2024. URL https://arxiv.org/abs/2402.04492. Added: 2026-05-07.

Chen, X., Fang, H., Lin, T.-Y., Vedantam, R., Gupta, S., Dollar, P., and Zitnick, C. L. Microsoft COCO captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015. URL https://arxiv.org/abs/1504.00325. Added: 2026-05-07.

Diwan, A., Berry, L., Choi, E., Harwath, D., and Mahowald, K. Why is Winoground hard? investigating failures in visuolinguistic compositionality. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 2236–2250, 2022.

Doveh, S., Arbelle, A., Harary, S., Herzig, R., Kim, D., Cascante-Bonilla, P., Alfassy, A., Panda, R., Giryes, R., Feris, R., Ullman, S., and Karlinsky, L. Dense and aligned captions (DAC) promote compositional reasoning in VL models. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://arxiv.org/abs/2305.19595. Added: 2026-05-07.

Fan, L., Krishnan, D., Isola, P., Katabi, D., and Tian, Y. Improving CLIP training with language rewrites. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=SVjDiiVySh.

Fang, A., Ilharco, G., Wortsman, M., Wan, Y., Shankar, V., Dave, A., and Schmidt, L. Data determines distributional robustness in contrastive language image pre-training (CLIP). In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, pp. 6216–6234. PMLR, 2022.

Hsieh, C.-Y., Zhang, J., Ma, Z., Kembhavi, A., and Krishna, R. SugarCrepe: Fixing hackable benchmarks for vision-language compositionality. In *Advances in Neural Information Processing Systems*, 2023.

Ilharco, G., Wortsman, M., Wightman, R., Gordon, C., Carlini, N., Taori, R., Dave, A., Shankar, V., Namkoong, H., Miller, J., Hajishirzi, H., Farhadi, A., and Schmidt, L. OpenCLIP, 2021. URL https://doi.org/10.5281/zenodo.5143773. Added: 2026-05-07.

Kargi, B., Uselis, A., and Oh, S. J. Half-truths break similarity-based retrieval. *arXiv preprint arXiv:2602.23906*, 2026. URL https://arxiv.org/abs/2602.23906. Added: 2026-05-07. Method referred to as CS-CLIP (Component-Supervised CLIP).

Koishigarina, D., Uselis, A., and Oh, S. J. CLIP behaves like a bag-of-words model cross-modally but not uni-modally. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=DldwXCCP25.

Liu, H., Li, C., Wu, Q., and Lee, Y. J. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 34892–34916, 2023. URL https://arxiv.org/abs/2304.08485. Added: 2026-05-07. NeurIPS 2023 Oral. Model referred to as LLaVA.

Mayilvahanan, P., Wiedemer, T., Rusak, E., Bethge, M., and Brendel, W. Does CLIP's generalization performance mainly stem from high train-test similarity? In *The Twelfth International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.09562.

Miranda, I., Salaberria, A., Agirre, E., and Azkune, G. BiVLC: Extending vision-language compositionality evaluation with text-to-image retrieval. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2406.09952. Datasets and Benchmarks Track.

Nguyen, T., Ilharco, G., Wortsman, M., Oh, S., and Schmidt, L. Quality not quantity: On the interaction between dataset design and robustness of CLIP. *Advances in Neural Information Processing Systems*, 35:21455–21469, 2022.

Parashar, S., Lin, Z., Liu, T., Dong, X., Li, Y., Ramanan, D., Caverlee, J., and Kong, S. The neglected tails in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12988–12997, 2024.

Patel, M., Kusumba, A., Cheng, S., Kim, C., Gokhale, T., Baral, C., and Yang, Y. TripletCLIP: Improving compositional reasoning of CLIP via synthetic vision-language negatives. In *Advances in Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=ZfRGRK5Kxl.

Peleg, A., Singh, N. D., and Hein, M. Advancing compositional awareness in CLIP with efficient fine-tuning. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2505.24424. Added: 2026-05-07. Method referred to as CLiC.

Qu, H. and Xie, S. M. Impact of pretraining word co-occurrence on compositional generalization in multimodal models. *arXiv preprint arXiv:2507.08000*, 2025.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 8748–8763. PMLR, 2021. URL https://proceedings.mlr.press/v139/radford21a.html. Added: 2026-05-07.

Thrush, T., Jiang, R., Bartolo, M., Singh, A., Williams, A., Kiela, D., and Ross, C. Winoground: Probing vision and language models for visio-linguistic compositionality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5238–5248, 2022.

Udandarao, V., Prabhu, A., Ghosh, A., Sharma, Y., Torr, P. H., Bibi, A., Albanie, S., and Bethge, M. No "Zero-Shot" without exponential data: Pretraining concept frequency determines multimodal model performance. *Advances in Neural Information Processing Systems*, 37: 61735–61792, 2024.

Uselis, A., Dittadi, A., and Oh, S. J. Does data scaling lead to visual compositional generalization? In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=M2WMUuwoh5.

Vani, A., Nguyen, B., Lavoie, S., Krishna, R., and Courville, A. SPARO: Selective attention for robust and compositional transformer encodings for vision. In *European Conference on Computer Vision*, pp. 233–251, 2024. URL https://arxiv.org/abs/2404.15721. Added: 2026-05-07.

Wiedemer, T., Sharma, Y., Prabhu, A., Bethge, M., and Brendel, W. Pretraining frequency predicts compositional generalization of CLIP on real-world tasks. *arXiv preprint arXiv:2502.18326*, 2025.

Yuksekgonul, M., Bianchi, F., Kalluri, P., Jurafsky, D., and Zou, J. When and why vision-language models behave like bags-of-words, and what to do about it? In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=KRLUvxh8uaX. Added: 2026-05-07. Introduces ARO benchmark and NegCLIP.

Zhang, Q., Chen, Y., Deng, L., and Shen, L. ABE-CLIP: Training-free attribute binding enhancement for compositional image-text matching. *arXiv preprint arXiv:2512.17178*, 2025.

A. Technical appendices and supplementary material

A.1. PUG:SPARE additional results

Table 2 reports per-backbone binary accuracy and top-1 retrieval (R@1) on the fully-seen, partially-unseen (mixed), and fully-unseen splits for all nine LABCLIP-aligned backbones, with mean and standard deviation across 3 seen–unseen data-split seeds. Across all backbones, both metrics drop monotonically from seen to unseen, with R@1 dropping more sharply than binary accuracy.

Table 3 summarizes the pairwise R@1 gaps between splits. The Seen → Unseen column quantifies the compositional generalization gap: every backbone loses at least 20 R@1 points when moving from seen to fully-unseen bindings, with the largest gaps ($\sim$42 points) on CLIP OpenAI (42.9), SPARO (41.8, a CC12M backbone), and NegCLIP COCO-ft (41.4); the absolute PUG:SPARE gap does not track pretraining scale cleanly, since OpenCLIP LAION-2B (a large backbone) has a smaller 37.7-point gap. The smaller-pretrained CC12M backbones with hard-negative augmentation (LaCLIP, NegCLIP, NegCLIP++) show smaller absolute gaps, but this reflects already-low seen-split R@1 rather than better generalization.

Table 2. PUG:SPARE binary accuracy (Acc) and top-1 retrieval (R@1) across evaluation splits. Results show the mean and standard deviation across 3 seen-unseen data split seeds.

Backbone	Seen		Partially Unseen		Fully Unseen	
	Acc	R@1	Acc	R@1	Acc	R@1
CLIP (OpenAI)	93.1 ± 0.9	68.2 ± 1.2	76.4 ± 2.2	36.7 ± 11.2	63.4 ± 2.8	25.3 ± 14.8
OpenCLIP (LAION-2B)	93.4 ± 0.5	73.0 ± 2.2	80.7 ± 2.2	47.9 ± 7.3	71.2 ± 2.9	35.3 ± 11.5
NegCLIP (COCO ft)	95.9 ± 0.5	69.2 ± 3.4	80.0 ± 2.5	39.2 ± 5.7	63.4 ± 1.0	27.8 ± 8.0
SPARO (CC12M)	99.2 ± 0.1	87.1 ± 2.6	93.7 ± 2.7	58.9 ± 7.9	84.6 ± 7.9	45.3 ± 13.7
CLIP (CC12M)	92.1 ± 0.8	51.5 ± 0.4	80.2 ± 3.0	23.5 ± 2.1	71.2 ± 4.8	13.3 ± 1.9
LaCLIP (CC12M)	89.4 ± 0.3	29.7 ± 0.7	76.0 ± 3.2	12.4 ± 0.5	64.2 ± 6.6	6.6 ± 4.9
NegCLIP (CC12M)	88.2 ± 0.8	27.6 ± 2.3	78.1 ± 3.7	12.7 ± 0.6	71.0 ± 6.1	6.3 ± 1.8
NegCLIP++ (CC12M)	85.7 ± 0.4	25.6 ± 1.5	74.8 ± 3.6	10.7 ± 1.3	65.2 ± 6.9	5.2 ± 2.0
TripletCLIP (CC12M)	90.7 ± 0.2	35.9 ± 0.5	79.5 ± 1.9	15.4 ± 3.3	68.4 ± 2.2	9.0 ± 5.1

Table 3. PUG:SPARE pairwise gaps in top-1 retrieval (R@1) across evaluation splits. Each entry is computed as R@1(first split) − R@1(second split), so a positive value indicates a drop in retrieval performance when moving to the second (typically harder) split, with **larger positive values reflecting a wider generalization gap**. Negative values indicate that the model performs better on the second split than the first.

Backbone	Seen → Partial Unseen	Seen → Fully Unseen	Partial Unseen → Fully Unseen
CLIP (OpenAI)	+31.5	+42.9	+11.4
OpenCLIP (LAION-2B)	+25.1	+37.7	+12.6
NegCLIP (COCO ft)	+30.0	+41.4	+11.4
SPARO (CC12M)	+28.3	+41.8	+13.6
CLIP (CC12M)	+28.0	+38.3	+10.3
LaCLIP (CC12M)	+17.4	+23.1	+5.7
NegCLIP (CC12M)	+14.9	+21.3	+6.4
NegCLIP++ (CC12M)	+14.9	+20.4	+5.6
TripletCLIP (CC12M)	+20.5	+26.9	+6.4

Table 4 reports the full-finetuning (FT) counterpart to the LABCLIP results above (the right panel of Figure 2): instead of training a frozen-backbone alignment layer, each backbone is fully finetuned. Finetuning raises R@1 on every split (fully-seen, partially-unseen, and fully-unseen) for all nine backbones relative to LABCLIP, and R@1 still drops monotonically from seen to fully-unseen, but the size of that drop splits into two regimes. The three large-scale backbones together with SPARO and vanilla CLIP-CC12M reach 74–94% R@1 on fully-unseen bindings, shrinking their seen-to-unseen gap to 6–25 points relative to the LABCLIP setting. The four remaining CC12M backbones (LaCLIP, NegCLIP, NegCLIP++, TripletCLIP) instead rise only to 14–20% R@1 on fully-unseen while reaching 71–76% on fully-seen, so finetuning *widens* their gap to 56–61 points. Full finetuning therefore narrows the compositional generalization gap for the stronger backbones

but widens it for the four remaining CC12M backbones.

Table 4. PUG:SPARE top-1 retrieval (R@1) across evaluation splits under full backbone finetuning (FT). Results show the mean and standard deviation across 3 seen–unseen data split seeds.

Backbone	Seen	Partially Unseen	Unseen
CLIP (OpenAI)	100.0 ± 0.0	95.6 ± 1.6	88.8 ± 1.9
OpenCLIP (LAION-2B)	100.0 ± 0.0	97.4 ± 2.0	94.2 ± 4.3
NegCLIP (COCO ft)	99.7 ± 0.5	87.9 ± 3.5	74.3 ± 8.0
SPARO (CC12M)	99.9 ± 0.1	93.0 ± 5.3	84.3 ± 11.2
CLIP (CC12M)	100.0 ± 0.0	93.6 ± 5.1	87.0 ± 9.6
LaCLIP (CC12M)	74.9 ± 0.7	32.7 ± 3.2	13.6 ± 4.3
NegCLIP (CC12M)	70.7 ± 1.8	34.6 ± 6.0	15.1 ± 7.3
NegCLIP++ (CC12M)	72.4 ± 1.4	33.6 ± 3.2	13.9 ± 4.8
TripletCLIP (CC12M)	76.1 ± 0.2	40.7 ± 6.5	20.3 ± 7.1

A.2. ARO VG-A additional results

Table 5 reports per-backbone image-to-text accuracy on the full ARO VG-A evaluation set and on the seen, mixed, and unseen splits for all nine LABCLIP-aligned backbones, with mean and standard deviation across 3 LABCLIP alignment seeds. The larger-pretrained backbones (CLIP OpenAI, OpenCLIP LAION-2B, NegCLIP COCO-ft) peak on the mixed split (where the binding-familiarity shortcut is present) and drop on both the seen and unseen splits. CC12M backbones without hard-negative augmentation (SPARO, CLIP CC12M, LaCLIP) stay roughly flat across the full, seen, and mixed splits and drop only on unseen, consistent with relying less on the shortcut.

Table 6 summarizes the pairwise accuracy gaps between splits. The Mixed $\rightarrow$ Seen column quantifies the *shortcut tax*—how much accuracy each backbone loses when the asymmetric familiarity advantage is removed: the larger-pretrained backbones lose $+6.1$ to $+8.6$ points (OpenCLIP LAION-2B $+6.1$, CLIP OpenAI $+7.7$, NegCLIP COCO-ft $+8.6$), while CC12M backbones without hard-negative augmentation register near-zero or slightly negative shortcut taxes (-2.4 to -0.2). The Seen $\rightarrow$ Unseen column quantifies the compositional generalization gap and is smaller and noisier than the shortcut tax: gaps range from under 1 point on CLIP OpenAI and NegCLIP COCO-ft to $+12.8$ on NegCLIP (CC12M), with the unseen split's small sample size ($n = 314$) inflating the noise.

Table 5. ARO VG-A image-to-text accuracy across evaluation splits. Results show the mean and standard deviation across 3 LABCLIP alignment seeds.

Backbone	Full	Seen	Mixed	Unseen
CLIP (OpenAI)	68.7 ± 0.2	62.7 ± 0.3	70.4 ± 0.2	61.8 ± 0.6
OpenCLIP (LAION-2B)	70.1 ± 0.2	65.3 ± 0.2	71.4 ± 0.2	60.0 ± 1.1
NegCLIP (COCO ft)	70.2 ± 0.4	63.5 ± 0.4	72.1 ± 0.4	62.8 ± 1.1
SPARO (CC12M)	65.1 ± 0.2	65.9 ± 0.3	65.6 ± 0.2	61.0 ± 0.5
CLIP (CC12M)	60.8 ± 0.1	63.0 ± 0.1	60.7 ± 0.1	52.4 ± 0.2
LaCLIP (CC12M)	59.3 ± 0.1	59.7 ± 0.2	59.6 ± 0.1	53.2 ± 0.8
NegCLIP (CC12M)	61.9 ± 0.1	58.3 ± 0.3	63.4 ± 0.1	45.5 ± 0.3
NegCLIP++ (CC12M)	60.1 ± 0.1	58.5 ± 0.2	60.8 ± 0.2	54.6 ± 0.4
TripletCLIP (CC12M)	64.1 ± 0.0	60.6 ± 0.3	65.3 ± 0.0	56.7 ± 0.3

Tables 7 and 8 report the same per-split analysis under full COCO finetuning instead of LABCLIP alignment: each backbone is fully finetuned on COCO, except NegCLIP (COCO ft), which is used unchanged because it is already finetuned on COCO. The BindSplit pattern persists under full finetuning.

A.3. BiVLC and VisMin additional results

Table 9 reports per-backbone group accuracy, image-to-text (I2T), and text-to-image (T2I) accuracy on the full BiVLC evaluation set and on the seen, mixed, and unseen splits for all nine LABCLIP-aligned backbones, with mean and standard

Table 6. ARO VG-A pairwise gaps in image-to-text accuracy across evaluation splits. Each entry is computed as acc(first split) − acc(second split), so a positive value indicates a drop in accuracy when moving to the second (typically harder) split, with **larger positive values reflecting a wider generalization gap**. Negative values indicate that the model performs better on the second split than the first.

Backbone	Full → Seen	Mixed → Seen	Seen → Unseen	Full → Unseen
CLIP (OpenAI)	$+6.0$	$+7.7$	$+0.9$	$+6.9$
OpenCLIP (LAION-2B)	$+4.8$	$+6.1$	$+5.3$	$+10.1$
NegCLIP (COCO ft)	$+6.6$	$+8.6$	$+0.7$	$+7.3$
SPARO (CC12M)	-0.8	-0.3	$+4.9$	$+4.1$
CLIP (CC12M)	-2.3	-2.4	$+10.6$	$+8.3$
LaCLIP (CC12M)	-0.4	-0.2	$+6.5$	$+6.1$
NegCLIP (CC12M)	$+3.6$	$+5.1$	$+12.8$	$+16.4$
NegCLIP++ (CC12M)	$+1.6$	$+2.3$	$+4.0$	$+5.5$
TripletCLIP (CC12M)	$+3.5$	$+4.7$	$+3.9$	$+7.4$

Table 7. ARO VG-A image-to-text accuracy across evaluation splits under full COCO finetuning. Each backbone is fully finetuned on COCO, except NegCLIP (COCO ft), which is already finetuned on COCO and used unchanged. Results show the mean and standard deviation across 3 finetuning seeds.

Backbone	Full	Seen	Mixed	Unseen
CLIP (OpenAI)	71.5 ± 0.1	65.6 ± 0.3	73.2 ± 0.1	64.1 ± 0.5
OpenCLIP (LAION-2B)	72.1 ± 0.0	68.1 ± 0.1	73.6 ± 0.0	61.8 ± 0.3
NegCLIP (COCO ft)	70.9	65.4	72.5	64.0
SPARO (CC12M)	66.7 ± 0.1	70.4 ± 0.1	66.4 ± 0.0	64.1 ± 0.4
CLIP (CC12M)	62.4 ± 0.0	65.9 ± 0.1	61.9 ± 0.0	54.5 ± 0.6
LaCLIP (CC12M)	55.3 ± 0.0	56.2 ± 0.0	55.3 ± 0.0	51.3 ± 0.0
NegCLIP (CC12M)	58.7 ± 0.0	56.9 ± 0.0	59.6 ± 0.0	44.6 ± 0.0
NegCLIP++ (CC12M)	54.3 ± 0.0	54.9 ± 0.0	54.6 ± 0.0	50.4 ± 0.2
TripletCLIP (CC12M)	57.7 ± 0.0	58.9 ± 0.0	57.8 ± 0.0	50.6 ± 0.0

deviation across 3 LABCLIP alignment seeds. Group accuracy is the headline metric per §3.1. Every backbone drops from seen to unseen. The within-shortcut pattern from ARO VG-A does not replicate cleanly: on BiVLC, the three larger-pretrained backbones (CLIP OpenAI, OpenCLIP LAION-2B, NegCLIP COCO-ft) perform comparably on full and seen and *better* on seen than on mixed, opposite to the ARO VG-A shortcut tax.

Table 10 summarizes the pairwise group-accuracy gaps. The Seen → Unseen column quantifies the compositional generalization gap and is the dominant signal: gaps range from $+3.6$ on NegCLIP (CC12M) to $+14.4$ on CLIP OpenAI. The Mixed → Seen column is negative for the larger-pretrained backbones (CLIP OpenAI -4.4, OpenCLIP LAION-2B -2.4, NegCLIP COCO-ft -3.9) and small but positive for most CC12M backbones—an inverted pattern relative to ARO VG-A. The unseen split's small sample size ($n = 54$) inflates per-backbone noise.

Tables 11 and 12 report the same BiVLC analysis under full COCO finetuning instead of LABCLIP alignment: each backbone is fully finetuned on COCO, except NegCLIP (COCO ft), which is used unchanged because it is already finetuned on COCO.

Table 13 reports per-backbone group accuracy, text retrieval, and image retrieval accuracy on the full VisMin evaluation set and on the seen, mixed, and unseen splits, with mean and standard deviation across 3 LABCLIP alignment seeds. Group accuracy is the headline metric. Most backbones drop monotonically from seen to unseen, with two notable exceptions: OpenCLIP LAION-2B increases slightly from seen (31.6) to unseen (32.6), and LaCLIP-CC12M sits at or below random group accuracy ($\sim 16.67\%$) across most splits, rendering its results uninformative.

Table 14 summarizes the pairwise group-accuracy gaps. The Mixed → Seen column is positive across every backbone (range $+4.1$ to $+11.0$), recovering a conventional shortcut-tax pattern. The Seen → Unseen column ranges from -5.1 on the uninformative LaCLIP-CC12M backbone to $+9.7$ on CLIP OpenAI. The unseen split's small sample size ($n = 48$) inflates per-backbone noise.

Table 8. ARO VG-A pairwise gaps in image-to-text accuracy across evaluation splits under full COCO finetuning. Each entry is computed as acc(first split) − acc(second split), so a positive value indicates a drop in accuracy when moving to the second (typically harder) split, with **larger positive values reflecting a wider generalization gap**. Negative values indicate that the model performs better on the second split than the first.

Backbone	Full → Seen	Mixed → Seen	Seen → Unseen	Full → Unseen
CLIP (OpenAI)	+5.9	+7.6	+1.5	+7.4
OpenCLIP (LAION-2B)	+4.0	+5.5	+6.3	+10.3
NegCLIP (COCO ft)	+5.5	+7.1	+1.4	+6.9
SPARO (CC12M)	−3.7	−4.0	+6.3	+2.6
CLIP (CC12M)	−3.6	−4.1	+11.5	+7.9
LaCLIP (CC12M)	−0.9	−0.9	+5.0	+4.0
NegCLIP (CC12M)	+1.8	+2.7	+12.3	+14.2
NegCLIP++ (CC12M)	−0.6	−0.3	+4.5	+3.9
TripletCLIP (CC12M)	−1.3	−1.2	+8.3	+7.0

Table 9. BiVLC group accuracy, image-to-text (I2T), and text-to-image (T2I) accuracy across evaluation splits. Results show the mean and standard deviation across 3 LABCLIP alignment seeds.

Backbone	Full			Seen			Mixed			Unseen		
	Group	I2T	T2I	Group	I2T	T2I	Group	I2T	T2I	Group	I2T	T2I
CLIP (OpenAI)	53.2 ± 0.3	76.2 ± 0.3	56.8 ± 0.5	53.9 ± 0.5	75.6 ± 0.5	57.4 ± 0.8	49.5 ± 0.3	73.7 ± 0.4	53.4 ± 0.3	39.5 ± 2.1	69.8 ± 1.1	42.0 ± 2.8
OpenCLIP (LAION-2B)	55.6 ± 0.1	79.6 ± 0.1	58.6 ± 0.1	55.3 ± 0.5	79.5 ± 0.2	58.2 ± 0.5	53.0 ± 0.3	77.7 ± 0.3	56.2 ± 0.3	51.2 ± 1.1	77.2 ± 1.1	55.6 ± 1.9
NegCLIP (COCO ft)	55.8 ± 0.2	78.3 ± 0.3	59.3 ± 0.2	56.3 ± 0.5	79.1 ± 0.3	59.7 ± 0.6	52.3 ± 0.0	75.3 ± 0.5	55.6 ± 0.2	49.4 ± 1.1	81.5 ± 0.0	54.9 ± 1.1
SPARO (CC12M)	44.5 ± 0.4	70.9 ± 0.3	49.3 ± 0.2	43.2 ± 0.6	70.5 ± 0.5	48.3 ± 0.3	44.3 ± 0.2	70.1 ± 0.4	48.7 ± 0.2	29.6 ± 1.9	61.7 ± 2.8	35.8 ± 2.8
CLIP (CC12M)	39.6 ± 0.2	67.2 ± 0.2	44.3 ± 0.3	37.9 ± 0.2	65.7 ± 0.5	42.3 ± 0.3	39.6 ± 0.4	67.1 ± 0.3	44.4 ± 0.6	26.5 ± 1.1	59.3 ± 1.9	32.7 ± 2.8
LaCLIP (CC12M)	28.3 ± 0.5	54.4 ± 0.2	33.2 ± 0.2	28.1 ± 0.9	54.8 ± 0.3	32.4 ± 0.5	27.1 ± 0.3	52.2 ± 0.3	32.0 ± 0.1	23.5 ± 1.1	49.4 ± 1.1	32.1 ± 1.1
NegCLIP (CC12M)	30.2 ± 0.4	57.7 ± 0.1	34.9 ± 0.2	28.3 ± 0.2	57.7 ± 0.4	32.6 ± 0.2	30.7 ± 0.7	57.8 ± 0.4	35.7 ± 0.4	24.7 ± 1.1	42.0 ± 1.1	27.2 ± 2.1
NegCLIP++ (CC12M)	28.8 ± 0.3	53.8 ± 0.3	34.5 ± 0.2	27.1 ± 0.3	52.9 ± 0.2	32.1 ± 0.2	29.7 ± 0.4	54.4 ± 0.3	35.4 ± 0.5	16.7 ± 3.2	46.3 ± 0.0	22.8 ± 1.1
TripletCLIP (CC12M)	30.6 ± 0.3	63.2 ± 0.2	35.1 ± 0.5	29.3 ± 0.6	63.7 ± 0.4	32.9 ± 0.9	30.9 ± 0.5	61.5 ± 0.0	35.4 ± 0.5	25.3 ± 1.1	62.3 ± 1.1	28.4 ± 1.1

Tables 15 and 16 report the same VisMin analysis under full COCO finetuning instead of LABCLIP alignment: each backbone is fully finetuned on COCO, except NegCLIP (COCO ft), which is used unchanged because it is already finetuned on COCO.

A.4. LABCLIP alignment is necessary for compositional binding on PUG:SPARE

A.5. Full-finetuning control: LABCLIP and COCO finetuning produce similar BindSplit patterns

A natural concern is whether the BindSplit gap reflects a limitation of LABCLIP's frozen-backbone setup rather than a property of CLIP's pretrained representations: perhaps full finetuning would close the gap. We rule this out with a head-to-head comparison on the same base model and the same training data. We compare three conditions on ARO VG-A, BiVLC, and VisMin: (i) **OpenAI CLIP raw**, the unaligned baseline with no COCO exposure; (ii) **OpenAI CLIP + LABCLIP-on-COCO**, frozen backbone with a linear alignment layer trained on COCO; and (iii) **NegCLIP (COCO-ft)**, a full finetune of OpenAI CLIP on COCO with hard-negative training. Conditions (ii) and (iii) share the same base model and training set, so any difference between them isolates the alignment method.

Figure 7 shows the three conditions across all three benchmarks. Raw CLIP establishes the no-COCO-exposure anchor; both LABCLIP-on-COCO and NegCLIP COCO-ft show the BindSplit pattern (mixed-split shortcut tax and seen-to-unseen drop), and LABCLIP-on-COCO moves the OpenAI CLIP pattern toward NegCLIP COCO-ft. The match between LABCLIP and full finetuning is not exact, but the LABCLIP-to-full-finetune gap is small relative to the raw-CLIP-to-aligned gap, indicating that COCO exposure, not alignment method, is the dominant driver of the BindSplit pattern.

The BindSplit gap is therefore not an artifact of LABCLIP's frozen-backbone setup. Full finetuning on COCO produces the same shortcut tax and compositional generalization gap that LABCLIP-on-COCO produces, on the same base model and across all three real benchmarks.

Table 10. BiVLC pairwise gaps in group accuracy across evaluation splits. Each entry is computed as acc(first split) − acc(second split), so a positive value indicates a drop in accuracy when moving to the second (typically harder) split, with **larger positive values reflecting a wider generalization gap**. Negative values indicate that the model performs better on the second split than the first.

Backbone	Full → Seen	Mixed → Seen	Seen → Unseen	Full → Unseen
CLIP (OpenAI)	−0.7	−4.4	+14.4	+13.7
OpenCLIP (LAION-2B)	+0.2	−2.4	+4.1	+4.3
NegCLIP (COCO ft)	−0.4	−3.9	+6.9	+6.5
SPARO (CC12M)	+1.3	+1.1	+13.5	+14.9
CLIP (CC12M)	+1.7	+1.7	+11.3	+13.0
LaCLIP (CC12M)	+0.2	−1.1	+4.7	+4.8
NegCLIP (CC12M)	+1.9	+2.5	+3.6	+5.5
NegCLIP++ (CC12M)	+1.7	+2.6	+10.5	+12.2
TripletCLIP (CC12M)	+1.4	+1.6	+4.0	+5.3

Table 11. BiVLC group accuracy, image-to-text (I2T), and text-to-image (T2I) accuracy across evaluation splits under full COCO finetuning. Each backbone is fully finetuned on COCO, except NegCLIP (COCO ft), which is already finetuned on COCO and used unchanged. Results show the mean and standard deviation across finetuning seeds.

Backbone	Full			Seen			Mixed			Unseen		
	Group	I2T	T2I	Group	I2T	T2I	Group	I2T	T2I	Group	I2T	T2I
CLIP (OpenAI)	57.5 ± 0.1	79.7 ± 0.1	60.7 ± 0.2	58.5 ± 0.7	79.2 ± 0.4	61.8 ± 0.7	53.7 ± 0.2	77.8 ± 0.1	56.7 ± 0.2	49.4 ± 1.1	75.9 ± 0.0	51.9 ± 1.9
OpenCLIP (LAION-2B)	57.3 ± 0.2	81.3 ± 0.1	60.0 ± 0.3	58.3 ± 0.2	81.0 ± 0.3	61.3 ± 0.2	53.9 ± 0.3	79.6 ± 0.3	56.8 ± 0.4	43.8 ± 1.1	77.2 ± 1.1	43.8 ± 1.1
NegCLIP (COCO ft)	55.3	77.5	59.2	57.0	79.0	60.6	50.6	74.1	54.7	44.4	77.8	51.9
SPARO (CC12M)	43.0 ± 0.1	74.1 ± 0.1	46.6 ± 0.1	44.4 ± 0.1	72.8 ± 0.2	48.7 ± 0.2	40.9 ± 0.2	74.1 ± 0.0	43.9 ± 0.3	33.3 ± 0.0	74.1 ± 0.0	33.3 ± 0.0
CLIP (CC12M)	42.0 ± 0.1	69.8 ± 0.1	46.4 ± 0.1	42.9 ± 0.3	69.9 ± 0.1	47.2 ± 0.4	40.8 ± 0.0	68.6 ± 0.1	45.1 ± 0.1	29.6 ± 0.0	56.8 ± 1.1	33.3 ± 0.0
LaCLIP (CC12M)	26.3 ± 0.0	56.2 ± 0.0	31.2 ± 0.0	25.8 ± 0.0	55.7 ± 0.0	30.6 ± 0.0	25.6 ± 0.0	56.0 ± 0.0	30.2 ± 0.0	24.1 ± 0.0	48.1 ± 0.0	33.3 ± 0.0
NegCLIP (CC12M)	26.2 ± 0.1	55.6 ± 0.0	31.4 ± 0.1	23.2 ± 0.1	54.9 ± 0.0	28.7 ± 0.1	27.5 ± 0.1	56.0 ± 0.1	33.0 ± 0.1	20.4 ± 0.0	42.6 ± 0.0	24.1 ± 0.0
NegCLIP++ (CC12M)	26.6 ± 0.0	51.9 ± 0.0	32.3 ± 0.0	22.5 ± 0.1	49.5 ± 0.1	27.2 ± 0.1	27.6 ± 0.0	52.7 ± 0.0	33.6 ± 0.0	14.8 ± 0.0	42.6 ± 0.0	18.5 ± 0.0
TripletCLIP (CC12M)	28.0 ± 0.0	62.4 ± 0.0	31.8 ± 0.0	26.5 ± 0.0	63.0 ± 0.1	29.7 ± 0.1	27.6 ± 0.0	60.4 ± 0.0	31.3 ± 0.0	29.6 ± 0.0	61.1 ± 0.0	35.2 ± 0.0

B. ARO–COCO binding-overlap pipeline

B.1. Overview

This appendix documents the pipeline that, for every sample in the ARO VG-Attribution benchmark (Yuksekgonul et al., 2023), decides whether the attribute–object bindings used in the positive and the attribute-swapped negative caption appear in COCO's training set (2014 version) captions (Chen et al., 2015), the dataset most commonly used as a training-distribution proxy for CLIP-style models (Radford et al., 2021). The pipeline produces (i) a per-sample overlap label for each of the four bindings in every ARO VG-A sample, (ii) granular and collapsed partitions of the benchmark into seen, mixed, and unseen subsets, and (iii) aggregate caption-level statistics used in the main paper. The two headline numbers reported in the abstract — positive captions overlap COCO bindings $1.9\times$ as often than their attribute-swapped negatives, and only 1.2% of samples are fully unseen — are produced by this pipeline. The pipeline is intentionally permissive in what it counts as overlap, so the reported contamination of ARO VG-A by COCO is best read as an upper bound on the seen fraction and a lower bound on the unseen fraction in terms of exact matching attribute–object bindings (no synonym matches).

B.2. Datasets

ARO VG-Attribution (VG-A). The Attribution, Relations and Order (ARO) benchmark (Yuksekgonul et al., 2023) probes whether a vision-language model distinguishes a true caption from a minimally edited distractor that reuses the same content words in a different syntactic role. Its Visual-Genome Attribution split (VG-A) consists of 28,748 image–text samples; each sample comes with a positive caption of the form "the a_1 o_1 and the a_2 o_2" (where a_i are attributes and o_i are objects) and an attribute-swapped negative "the a_2 o_1 and the a_1 o_2". By construction, both captions reuse the same four content words; only the attribute–object pairing changes.

COCO captions (training split). The COCO dataset (Chen et al., 2015) pairs each image with five free-form English captions. We use only the *training* split as the training-distribution proxy. Free-form captions do not directly expose attribute–object bindings, so we first extract noun-phrase components from each caption with the extraction pipeline of (Kargi et al., 2026) an open-weights vision-language model. The extractor returns, for each caption, a list of head-noun-anchored

Table 12. BiVLC pairwise gaps in group accuracy across evaluation splits under full COCO finetuning. Each entry is computed as acc(first split) − acc(second split), so a positive value indicates a drop in accuracy when moving to the second (typically harder) split, with **larger positive values reflecting a wider generalization gap**. Negative values indicate that the model performs better on the second split than the first.

Backbone	Full → Seen	Mixed → Seen	Seen → Unseen	Full → Unseen
CLIP (OpenAI)	−1.0	−4.8	+9.1	+8.1
OpenCLIP (LAION-2B)	−1.0	−4.4	+14.4	+13.4
NegCLIP (COCO ft)	−1.7	−6.3	+12.5	+10.8
SPARO (CC12M)	−1.4	−3.5	+11.1	+9.7
CLIP (CC12M)	−0.9	−2.1	+13.3	+12.4
LaCLIP (CC12M)	+0.4	−0.2	+1.8	+2.2
NegCLIP (CC12M)	+3.0	+4.3	+2.8	+5.8
NegCLIP++ (CC12M)	+4.1	+5.2	+7.6	+11.8
TripletCLIP (CC12M)	+1.5	+1.1	−3.1	−1.6

Table 13. VisMin group accuracy, text retrieval, and image retrieval accuracy across evaluation splits. Results show the mean and standard deviation across 3 LABCLIP alignment seeds.

Backbone	Full			Seen			Mixed			Unseen		
	Group	Text	Image	Group	Text	Image	Group	Text	Image	Group	Text	Image
CLIP (OpenAI)	30.5 ± 0.3	44.8 ± 0.4	38.7 ± 0.4	27.2 ± 0.4	41.9 ± 0.5	35.5 ± 0.6	36.9 ± 0.2	51.7 ± 0.5	44.8 ± 0.5	18.8 ± 0.0	37.5 ± 0.0	27.1 ± 2.1
OpenCLIP (LAION-2B)	36.1 ± 0.3	51.2 ± 0.3	45.0 ± 0.2	31.6 ± 0.2	48.3 ± 0.4	41.1 ± 0.4	42.7 ± 0.5	57.2 ± 0.4	52.4 ± 0.3	32.6 ± 1.2	53.5 ± 1.2	36.8 ± 2.4
NegCLIP (COCO ft)	36.7 ± 0.7	49.7 ± 0.3	46.1 ± 0.2	34.6 ± 0.7	48.7 ± 0.2	43.1 ± 0.3	41.9 ± 0.6	54.8 ± 0.5	52.2 ± 0.3	29.2 ± 2.1	51.4 ± 3.2	36.8 ± 1.2
SPARO (CC12M)	23.3 ± 0.2	35.9 ± 0.9	33.2 ± 0.4	20.3 ± 0.4	33.8 ± 0.4	30.0 ± 0.5	29.0 ± 0.7	42.0 ± 1.1	39.9 ± 0.2	13.9 ± 1.2	30.6 ± 4.3	19.4 ± 1.2
CLIP (CC12M)	21.2 ± 0.4	34.5 ± 0.2	31.3 ± 0.5	18.7 ± 0.1	32.2 ± 0.2	28.5 ± 0.1	27.4 ± 1.0	42.0 ± 0.6	37.9 ± 0.6	13.9 ± 1.2	28.5 ± 1.2	27.1 ± 2.1
LaCLIP (CC12M)	12.7 ± 0.1	25.5 ± 0.2	20.2 ± 0.3	9.5 ± 0.0	24.0 ± 0.4	16.7 ± 0.4	17.2 ± 0.2	30.5 ± 0.3	25.6 ± 0.3	14.6 ± 0.0	22.2 ± 1.2	18.1 ± 1.2
NegCLIP (CC12M)	12.4 ± 0.1	25.8 ± 0.2	20.5 ± 0.3	11.2 ± 0.4	24.2 ± 0.9	18.8 ± 0.1	16.4 ± 0.3	31.2 ± 0.3	25.5 ± 0.8	2.1 ± 2.1	17.4 ± 1.2	19.4 ± 1.2
NegCLIP++ (CC12M)	12.3 ± 0.1	23.5 ± 0.2	22.0 ± 0.1	11.5 ± 0.5	23.1 ± 0.2	20.6 ± 0.2	15.6 ± 0.2	27.7 ± 0.4	25.8 ± 0.2	10.4 ± 2.1	23.6 ± 2.4	20.1 ± 1.2
TripletCLIP (CC12M)	15.5 ± 0.2	29.2 ± 0.3	23.4 ± 0.2	14.3 ± 0.4	29.3 ± 0.6	20.9 ± 0.2	20.7 ± 0.2	34.5 ± 0.3	30.0 ± 0.2	6.2 ± 0.0	25.7 ± 1.2	11.8 ± 1.2

components with their attached attributes (e.g. from "a small white dog on a red couch" it returns ["small white dog", "red couch"]). We do not modify the captions themselves. The extraction yields 1,121,471 noun-phrase components across 410,340 training captions. These component counts are *measured* from running the extractor end-to-end; the downstream binding extraction in §B.5 discards a sizable fraction of them.

B.3. Defining a binding

We define a *binding* as an ordered pair $(attr, obj)$ of strings, both lowercased and ASCII-normalized. The two sides of a binding are processed asymmetrically:

- **Object side (obj) is lemmatized to singular.** We use a lemmatizer with two manual overrides: an *irregular plural* list (e.g. `children`→`child`, `teeth`→`tooth`, `geese`→`goose`) and a small *keep-as-is* list of nouns whose "singular" form is not a real referent in everyday English (e.g. `glasses`, `scissors`, `pants`, `jeans`). Without the overrides the lemmatizer produces non-words or wrong roots on these items.

- **Attribute side ($attr$) is left untouched.** Many attributes in both ARO and COCO are participles ("striped", "smiling", "frosted") or compound colour terms; lemmatizing them collapses meaningful distinctions and inflates the apparent overlap. We verified by spot-check that lemmatizing the attribute side merges, for example, the ARO attribute "smiling" into the COCO bare verb "smile", which is not the same descriptor. We therefore only case-fold and trim whitespace on the attribute side.

A binding match is therefore a *lemmatized match* on the object string and a case-folded exact match on the attribute string. This is more permissive than literal exact match (singular and plural forms of the same noun are unified) and stricter than approximate string matching. The asymmetry is the most important caveat to keep in mind when reading the overlap numbers; we revisit it in §B.8.

Table 14. VisMin pairwise gaps in group accuracy across evaluation splits. Each entry is computed as acc(first split) − acc(second split), so a positive value indicates a drop in accuracy when moving to the second (typically harder) split, with **larger positive values reflecting a wider generalization gap**. Negative values indicate that the model performs better on the second split than the first.

Backbone	Full → Seen	Mixed → Seen	Seen → Unseen	Full → Unseen
CLIP (OpenAI)	+3.3	+9.7	+8.4	+11.7
OpenCLIP (LAION-2B)	+4.4	+11.0	−1.0	+3.4
NegCLIP (COCO ft)	+2.1	+7.3	+5.5	+7.5
SPARO (CC12M)	+3.0	+8.7	+6.4	+9.4
CLIP (CC12M)	+2.5	+8.7	+4.8	+7.4
LaCLIP (CC12M)	+3.2	+7.7	−5.1	−1.9
NegCLIP (CC12M)	+1.2	+5.1	+9.1	+10.3
NegCLIP++ (CC12M)	+0.8	+4.1	+1.1	+1.9
TripletCLIP (CC12M)	+1.2	+6.4	+8.1	+9.3

Table 15. VisMin group accuracy, text retrieval, and image retrieval accuracy across evaluation splits under full COCO finetuning. Each backbone is fully finetuned on COCO, except NegCLIP (COCO ft), which is already finetuned on COCO and used unchanged. Results show the mean and standard deviation across finetuning seeds.

Backbone	Full			Seen			Mixed			Unseen		
	Group	Text	Image	Group	Text	Image	Group	Text	Image	Group	Text	Image
CLIP (OpenAI)	35.3 ± 0.2	49.0 ± 0.2	44.6 ± 0.5	32.2 ± 0.2	47.1 ± 0.3	41.3 ± 0.5	41.5 ± 0.3	55.5 ± 0.1	51.1 ± 0.6	22.9 ± 3.6	47.2 ± 2.4	31.2 ± 4.2
OpenCLIP (LAION-2B)	42.6 ± 0.1	54.8 ± 0.2	50.8 ± 0.1	40.6 ± 0.4	52.4 ± 0.6	48.4 ± 0.6	48.4 ± 0.2	60.9 ± 0.2	56.9 ± 0.4	41.0 ± 1.2	55.6 ± 1.2	45.8 ± 2.1
NegCLIP (COCO ft)	37.5	51.0	46.6	35.0	49.4	43.9	43.5	57.4	53.7	25.0	54.2	35.4
SPARO (CC12M)	26.8 ± 0.1	40.7 ± 0.1	36.1 ± 0.1	23.9 ± 0.2	37.2 ± 0.1	33.9 ± 0.4	34.3 ± 0.1	49.1 ± 0.1	42.0 ± 0.3	19.4 ± 1.2	36.8 ± 1.2	25.7 ± 1.2
CLIP (CC12M)	23.7 ± 0.1	37.7 ± 0.2	33.5 ± 0.1	21.7 ± 0.1	36.1 ± 0.4	31.1 ± 0.3	29.5 ± 0.3	43.5 ± 0.3	40.3 ± 0.2	18.8 ± 2.1	35.4 ± 0.0	29.9 ± 1.2
LaCLIP (CC12M)	12.5 ± 0.0	25.6 ± 0.0	21.0 ± 0.0	9.2 ± 0.0	22.9 ± 0.1	17.2 ± 0.0	16.6 ± 0.1	30.8 ± 0.1	26.7 ± 0.0	12.5 ± 0.0	27.1 ± 0.0	16.7 ± 0.0
NegCLIP (CC12M)	12.9 ± 0.0	27.0 ± 0.0	20.6 ± 0.0	11.2 ± 0.0	25.5 ± 0.0	18.5 ± 0.0	17.1 ± 0.0	32.6 ± 0.0	25.2 ± 0.0	6.2 ± 0.0	20.8 ± 0.0	14.6 ± 0.0
NegCLIP++ (CC12M)	12.0 ± 0.0	23.0 ± 0.0	20.3 ± 0.0	10.1 ± 0.1	22.6 ± 0.0	17.5 ± 0.0	15.8 ± 0.0	26.7 ± 0.0	25.1 ± 0.0	10.4 ± 0.0	16.7 ± 0.0	17.4 ± 1.2
TripletCLIP (CC12M)	14.2 ± 0.0	28.1 ± 0.0	23.6 ± 0.0	12.4 ± 0.0	26.5 ± 0.0	20.9 ± 0.0	18.6 ± 0.0	32.0 ± 0.0	29.3 ± 0.0	6.2 ± 0.0	14.6 ± 0.0	20.8 ± 0.0

B.4. Parsing VG-A captions

VG-A captions follow a near-deterministic surface form, which we parse with the strict regular expression

```
^the (\S+) (\S+) and the (\S+) (\S+)$
```

applied to the lowercased positive caption. A caption matches if and only if it has exactly two single-token attributes and two single-token objects joined by "`and the`". We retain only samples for which both the positive and the negative caption match this regex. This drops 2,240 of 28,748 samples (7.8%); 26,508 samples are retained. Manual inspection of a 100-sample random subset of the dropped rows shows the drops are dominated by multi-word object phrases ("long sleeved shirt", "coffee table", "dining table") and hyphen-free colour compounds ("light blue", "dark green"). The strict parser therefore biases the retained subset toward simpler bindings, but does not introduce a systematic bias between the positive and negative caption within a sample, since both are parsed by the same regex and a sample is retained only if both pass.

B.5. Extracting bindings from COCO components

Each Qwen-extracted component is a whitespace-tokenized sequence of one or more lowercase tokens. We treat the last token as the head object and all preceding tokens as attributes:

- A component with exactly one preceding token ("red couch") yields one binding $(\textit{red}, \textit{couch})$ and is labelled `perfect`.
- A component with $k > 1$ preceding tokens ("small white dog") decomposes into k bindings, all sharing the head object: $(\textit{small}, \textit{dog})$ and $(\textit{white}, \textit{dog})$. These bindings are labelled `close`, indicating that they witness the attribute–object pairing but in the context of additional attributes on the same head noun.
- A component with zero preceding tokens (a bare noun such as "chairs" or "person") yields no binding and is dropped.

The bare-noun drop discards 582,134 of 1,121,471 components (51.9%). This is a large and asymmetric loss — a substantial

Table 16. VisMin pairwise gaps in group accuracy across evaluation splits under full COCO finetuning. Each entry is computed as acc(first split) − acc(second split), so a positive value indicates a drop in accuracy when moving to the second (typically harder) split, with **larger positive values reflecting a wider generalization gap**. Negative values indicate that the model performs better on the second split than the first.

Backbone	Full → Seen	Mixed → Seen	Seen → Unseen	Full → Unseen
CLIP (OpenAI)	+3.1	+9.3	+9.2	+12.4
OpenCLIP (LAION-2B)	+2.1	+7.8	−0.4	+1.7
NegCLIP (COCO ft)	+2.4	+8.5	+10.0	+12.5
SPARO (CC12M)	+2.9	+10.4	+4.5	+7.4
CLIP (CC12M)	+2.0	+7.8	+2.9	+4.9
LaCLIP (CC12M)	+3.2	+7.4	−3.3	−0.0
NegCLIP (CC12M)	+1.7	+5.9	+4.9	+6.6
NegCLIP++ (CC12M)	+1.8	+5.6	−0.3	+1.5
TripletCLIP (CC12M)	+1.8	+6.2	+6.2	+8.0

Table 17. Full-finetuning control accuracies (%) across ARO VG-A, BiVLC, and VisMin, organized by condition. The LABCLIP row shows mean ± SD over 3 alignment seeds; the other two rows are deterministic single evaluations. BiVLC and VisMin report group accuracy.

Condition	ARO VG-A				BiVLC (group)				VisMin (group)			
	Full	Seen	Mixed	Unseen	Full	Seen	Mixed	Unseen	Full	Seen	Mixed	Unseen
CLIP (OpenAI) + LABCLIP	68.7 ± 0.2	62.7 ± 0.3	70.4 ± 0.2	61.8 ± 0.6	53.2 ± 0.3	53.9 ± 0.5	49.5 ± 0.3	39.5 ± 2.1	30.5 ± 0.3	27.2 ± 0.4	36.9 ± 0.2	18.8 ± 0.0
NegCLIP (COCO ft) raw	70.9	65.4	72.5	64.0	55.3	57.0	50.8	44.4	37.5	35.0	43.5	25.0
CLIP (OpenAI) raw	61.3	59.1	62.5	55.7	49.1	47.1	47.5	44.4	29.5	25.7	35.8	16.7

portion of the COCO training distribution simply does not expose any attribute–object binding under our convention — and it pushes the COCO-side support down. We discuss the directional consequences in §B.8.

After lemmatizing the object side and aggregating across all retained components, the COCO-side artifact is a table of 131,556 unique $(attr, obj)$ pairs, each with a `perfect` count and a `close` count (number of COCO components that witnessed the pair as the sole attribute or as one of several attributes, respectively).

B.6. Per-sample overlap labelling

For each VG-A sample we extract two bindings from the positive caption, (a_1, o_1) and (a_2, o_2), and two bindings from the attribute-swapped negative caption, (a_2, o_1) and (a_1, o_2) — four bindings per sample in total. For each binding b we look up $(\textit{attr}, \textit{lemma}(\textit{obj}))$ in the COCO pair table and assign one of three labels:

- `perfect`: the pair has nonzero `perfect` count in COCO, i.e. at least one COCO component consists exactly of this attribute and this object;
- `close-only`: the pair has zero `perfect` count but nonzero `close` count, i.e. COCO witnesses the binding only as part of a longer attributive phrase;
- `none`: neither count is nonzero.

The same labelling is applied independently to the two negative-side bindings. A single VG-A sample therefore carries four binding-level labels: two from the positive and two from the negative caption.

B.7. Partitioning the benchmark

We partition the 26,508 retained samples by the multiset of the four per-sample binding labels (two from the positive caption, two from the negative caption). For each sample we count $n_{\text{perfect}}, n_{\text{close}}, n_{\text{none}} \in \{0, \ldots, 4\}$ with $n_{\text{perfect}} + n_{\text{close}} + n_{\text{none}} = 4$, and assign the sample to one of seven buckets according to which of the three labels are present. Table 19 gives the buckets, their defining rules, and their counts. The buckets are mutually exclusive and exhaustive, so they sum to 26,508 by construction.

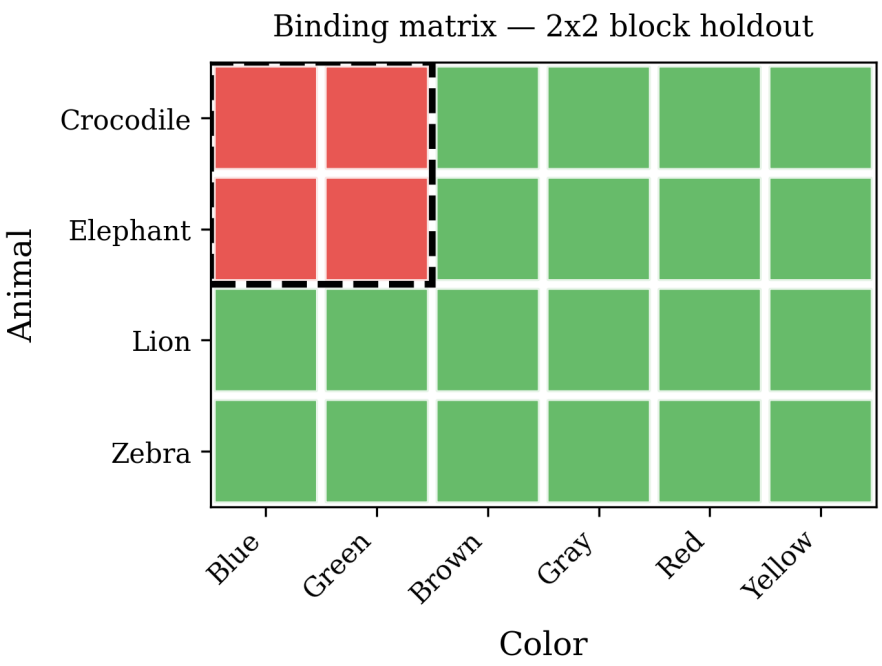

Figure 6. **Toy illustration of the BindSplit holdout on PUG:SPARE.** Green cells: (color, animal) bindings seen during alignment training. Red 2×2 block (Blue/Green $\times$ Crocodile/Elephant): held out. Every color and animal also appears outside the block—e.g., blue with Lion and Zebra—so individual concepts remain recognizable. The actual experiment uses a 3×3 holdout on the full 8×12 grid. Samples are classified as fully-seen, partially-unseen, or fully-unseen.

Table 18. Same data as Table 17, transposed: rows index splits, columns index (benchmark, condition). The LABCLIP column shows mean $\pm$ SD over 3 alignment seeds.

	ARO VG-A			BiVLC (group)			VisMin (group)		
Split	+LABCLIP	NegCLIP	CLIP	+LABCLIP	NegCLIP	CLIP	+LABCLIP	NegCLIP	CLIP
Full	68.7 ± 0.2	70.9	61.3	53.2 ± 0.3	55.3	49.1	30.5 ± 0.3	37.5	29.5
Seen	62.7 ± 0.3	65.4	59.1	53.9 ± 0.5	57.0	47.1	27.2 ± 0.4	35.0	25.7
Mixed	70.4 ± 0.2	72.5	62.5	49.5 ± 0.3	50.8	47.5	36.9 ± 0.2	43.5	35.8
Unseen	61.8 ± 0.6	64.0	55.7	39.5 ± 2.1	44.4	44.4	18.8 ± 0.0	25.0	16.7

Three-bucket collapse. The main paper uses the collapsed three-way partition $\{\texttt{seen}, \texttt{mixed}, \texttt{unseen}\}$ with counts $4{,}329/21{,}865/314$. The collapse is motivated by sample size: the inner ambiguous buckets are too small individually to support per-bucket numerical comparisons, but their union is large enough to serve as the contrast class for both the seen and the unseen partition. The `amb_close_only` bucket ($n = 5$) is excluded from any downstream evaluation as numerical noise; it is retained in Table 19 only so the granular partition remains exhaustive.

Strict vs. loose binary headlines. We additionally report the all-seen / all-unseen rates under two binarisations of the per-binding label (Table 20). Under the `STRICT` rule, only `perfect` counts as overlap, so a `close-only` match is treated as "not seen"; under the `LOOSE` rule, both `perfect` and `close-only` count as overlap. `LOOSE` is the rule used to define the canonical `definitely_unseen` subset ($n = 314$).

The `STRICT` all-unseen number (2.8%, 752 samples) is *misleading as a contamination floor* and we do not use it in the main paper. Of those 752 `STRICT`-unseen samples, 438 contain at least one `close-only` match in COCO, i.e. the pair appears in COCO as part of a longer attributive phrase. Reporting them as "unseen" would conflate "never appears in COCO" with "never appears as the head pair of a COCO component". The `definitely_unseen` subset ($n = 314$) is the intersection of `STRICT`-unseen with `LOOSE`-unseen and is the binding-overlap-conservative "no contamination of either

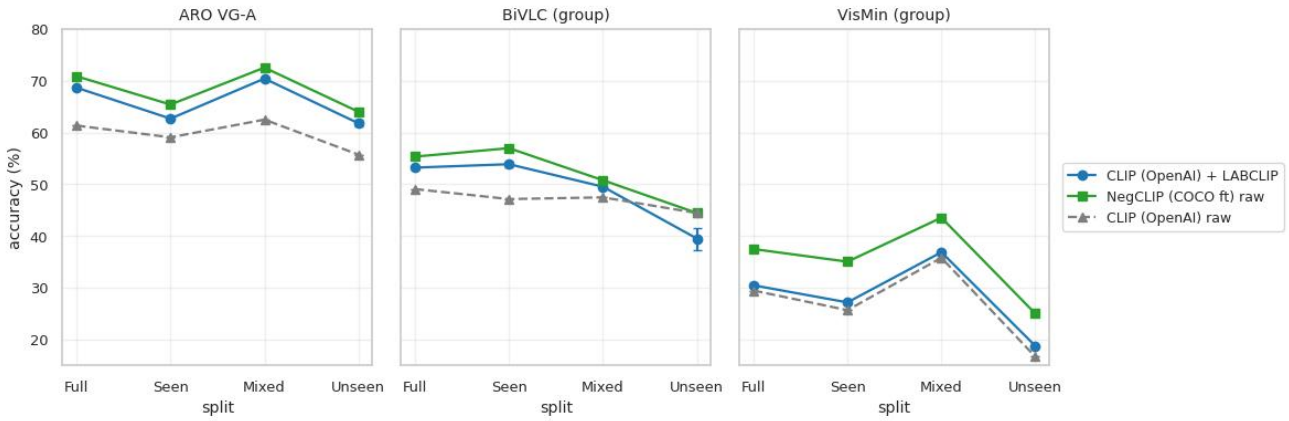

Figure 7. **LABCLIP vs. full finetuning control across ARO VG-A, BiVLC, and VisMin.** Three conditions on the same base model (OpenAI CLIP): *Raw* (no COCO exposure), + *LABCLIP-on-COCO* (frozen backbone, linear alignment trained on COCO), and *NegCLIP COCO-ft* (full finetune on COCO with hard-negative training). LABCLIP-on-COCO moves the raw CLIP pattern toward the NegCLIP COCO-ft pattern across all three benchmarks, indicating that COCO exposure—not alignment method—is the dominant driver of the BindSplit pattern. The LABCLIP condition shows mean ± SD over 3 alignment seeds; NegCLIP COCO-ft and raw CLIP are single evaluations from publicly downloaded checkpoints with no LABCLIP alignment trained on top.

Table 19. Granular 7-bucket partition of the 26,508 retained ARO VG-A samples by the multiset of their four per-sample binding labels. Each sample contributes counts $n_{\text{perfect}} + n_{\text{close}} + n_{\text{none}} = 4$. Counts are measured directly from the pipeline output. The `amb_close_only` bucket is so small (5 samples) that it is excluded from downstream evaluation; we list it for partition completeness. Collapsed labels (used in the main paper) are shown in the rightmost column.

Bucket	Defining rule on $(n_{\text{perfect}}, n_{\text{close}}, n_{\text{none}})$	Collapsed label	Count
`definitely_seen`	$n_{\text{perfect}} = 4$	`seen`	4,329
`amb_perfect_close`	$n_{\text{perfect}} > 0,\ n_{\text{close}} > 0,\ n_{\text{none}} = 0$	`mixed`	2,971
`amb_mixed`	$n_{\text{perfect}} > 0,\ n_{\text{close}} > 0,\ n_{\text{none}} > 0$	`mixed`	5,270
`amb_perfect_none`	$n_{\text{perfect}} > 0,\ n_{\text{close}} = 0,\ n_{\text{none}} > 0$	`mixed`	13,186
`amb_close_only`	$n_{\text{close}} = 4$	`mixed`	5
`amb_close_none`	$n_{\text{perfect}} = 0,\ n_{\text{close}} > 0,\ n_{\text{none}} > 0$	`mixed`	433
`definitely_unseen`	$n_{\text{none}} = 4$	`unseen`	314
All `mixed` *buckets combined*			21,865
Total			26,508

kind" subset. It is also the subset on which the rank-flip diagnostic in the main paper is computed.

B.8. Caveats and threats to validity

The pipeline is conservative about calling overlap, but it is not free of bias. We list the threats we are aware of, in roughly decreasing order of how much they should temper the headline numbers.

- **Lemmatize-then-exact-match, not exact match.** The object side of every binding is lemmatized before lookup, so "red cars" in ARO and "red car" in COCO are matched. The attribute side is case-folded only. Calling this "exact match" would be wrong; we expect a small upward bias on the seen rate from the lemmatization step. The bias is small because the canonical ARO and COCO surface forms are already singular for most head nouns.
- **Parser asymmetry between the two sides.** The VG-A regex rejects multi-word attributes and multi-word objects, while the COCO extractor decomposes a multi-word component into a fan-out of single-attribute bindings. Both directions push the headline "% seen" upward: the VG-A side is restricted to bindings that are easier to look up, and the COCO side multiplies the support of the COCO binding table. The directional claim that ARO VG-A has substantial COCO contamination is robust to this asymmetry, but the precise magnitudes (16.3% `STRICT`, 27.6% `LOOSE`) should be read as an upper bound on contamination if extrapolated naively to the full 28,748-sample VG-A.
- **Asymmetric loss from the bare-noun drop.** The 51.9% single-token drop on the COCO side is a one-sided loss:

Table 20. Binary all-seen / all-unseen rates on the 26,508 retained VG-A samples under two binarisations of the per-binding label.

Rule	% all-seen (both `perfect`)	% all-unseen
`STRICT` (only `perfect` counts as overlap)	16.3%	2.8%
`LOOSE` (both `perfect` and `close-only` count)	27.6%	1.2%

COCO captions that contain bare-noun phrases ("two chairs", "a person") do not contribute any binding to the COCO pair table even though they witness the head noun. Concretely, the COCO contribution is undercounted by phrases that contain no attributive token. This pushes the headline "% unseen" upward.

- **Implausible negatives, not just novel positives.** The attribute-swap construction produces negative captions that natural text would rarely contain ("crouched door", "melted apple"). A non-trivial share of the all-`none` samples in `definitely_unseen` are driven by negative-side novelty rather than positive-side novelty. We therefore use `definitely_unseen` only for the rank-flip diagnostic — which compares model orderings across subsets where the asymmetry between positive-side and negative-side overlap vanishes by construction — and not for claims about generalization to genuinely novel positive-side compositions.
- **Function-word artefacts on the COCO side.** Qwen-extracted components occasionally include preposition tokens as the leading attribute, e.g. "of birds" from a parse of "a flock of birds", producing bindings such as $(of, person)$ with non-trivial counts in the COCO pair table. These do not contaminate VG-A lookups, because every VG-A attribute is a real adjective by construction, but readers who inspect the released COCO-side pair counts will see them. We retain such bindings rather than apply a stoplist, both because there is no principled cutoff and because we prefer transparency over a hand-tuned filter.
- **No deduplication of repeated VG-A triples.** Approximately 695 of the 26,508 rows are exact duplicates of an earlier row at the level of (image id, positive caption, negative caption). We do not deduplicate, because the canonical ARO score reported in the original benchmark also evaluates at the row granularity, and deduplicating would silently change the reference score.

B.9. Reproducibility and released artefacts

The pipeline is deterministic given the input captions and the choice of lemmatizer and Qwen extractor. Re-running it from the released code produces, in the cache directory, the following files:

- `cache/aro_coco_overlap/coco_train_pair_counts.csv` — 131,556 rows, columns `(attr, obj_lemma, perfect_count, close_count)`.
- `cache/aro_coco_overlap/vga_per_sample_overlap.tsv` — 26,508 rows, one per retained VG-A sample, with the positive-caption and negative-caption per-binding labels and the joint bucket assignment.
- Nine split files (one per partition) in `cache/aro_coco_overlap/splits/`: the seven granular buckets `definitely_seen`, `amb_perfect_close`, `amb_mixed`, `amb_perfect_none`, `amb_close_only`, `amb_close_none`, `definitely_unseen`, plus the two aggregates `ambiguous` (union of all five `amb_*` buckets) and `any_overlap` (`definitely_seen` ∪ `ambiguous`).

Each split file is a list of VG-A row indices into the canonical VG-A ordering. All headline numbers in the main paper, in the abstract, and in this appendix can be regenerated from these three artefacts plus the ARO VG-A captions and the COCO training captions.