Half-Truths Break Similarity-Based Retrieval

Bora Kargi*
University of Tübingen, Tübingen AI Center
Konrad Zuse School of Excellence in Learning and Intelligent Systems (ELIZA)

Arnas Uselis
University of Tübingen, Tübingen AI Center

Seong Joon Oh
University of Tübingen, Tübingen AI Center

Code 🤗 Checkpoints 🤗 Dataset

Abstract

When a text description is extended with an additional detail, image-text similarity should drop if that detail is wrong. We show that CLIP-style dual encoders often violate this intuition: appending a plausible but incorrect object or relation to an otherwise correct description can increase the similarity score. We call such cases *half-truths*. Across 1,437 manually verified comparisons from COCO and CC3M using two foil generators, we find that this failure persists across generators and image domains. On COCO with Qwen-generated foils, CLIP prefers the correct shorter description only 34.0% of the time, dropping to 27.8% for relation additions. We trace this vulnerability to weak supervision on caption parts: contrastive training aligns full sentences but does not explicitly enforce that individual entities and relations are grounded. We propose CS-CLIP (Component-Supervised CLIP), which decomposes captions into entity and relation units, constructs a minimally edited foil for each unit, and fine-tunes the model to score the correct unit above its foil while preserving standard dual-encoder inference. CS-CLIP raises half-truth accuracy to 76.4% on this set and improves average performance on established compositional benchmarks by 5.8 points, suggesting that reducing half-truth errors aligns with broader gains in compositional understanding.

1 Introduction

When describing an image, adding a wrong detail should make the description *less* relevant, not more. If "a dog" correctly describes an image, then "a dog on a skateboard" should score *lower* when the dog is not on a skateboard. We show that CLIP-style dual encoders Radford et al. (2021) systematically violate this: appending a single incorrect but plausible detail often *increases* similarity.

We formalize this failure as the **half-truth vulnerability**. Given an image, we start from a short, caption-supported *anchor* description. We then form a *half-truth* by appending exactly one additional *unit* that is fluent and plausible in context but incorrect for the image. The appended unit is either an **entity unit** (a wrong entity description) or a **relation unit** (a wrong relation involving the anchor entity). Figure 1 illustrates both cases: an incorrect entity addition (*log* $\rightarrow$ *log and zebras*) and an incorrect relation addition (*elephants away from the log* instead of near), where CLIP Radford et al. (2021) and NegCLIP Yuksekgonul et al. (2023) score the half-truth higher than the anchor, while CS-CLIP correctly prefers the anchor.

*Corresponding author: kargibora@gmail.com

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

This vulnerability is common. On MS-COCO Lin et al. (2014), CLIP ranks the anchor above the half-truth in only 34.0% of cases overall, and 27.8% when the incorrect addition is a relation. Even training with sentence-level hard negatives improves this only moderately: NegCLIP achieves 53.8% overall but remains below chance (45.5%) for relation additions. A single wrong detail, especially a relational one, is not reliably penalized.

This pattern resembles the conjunction fallacy, where adding a plausible detail increases perceived likelihood even though conjunctions are more restrictive Tversky & Kahneman (1983); here the added detail is *plausible but wrong*, yet similarity still rises. The key difference from prior work on compositional understanding is the operation: existing benchmarks test whether models distinguish captions that *swap or reorder* information (e.g., "red cat and blue dog" vs. "blue cat and red dog") Yuksekgonul et al. (2023); Thrush et al. (2022), whereas half-truths test whether models *penalize incorrect additions*.

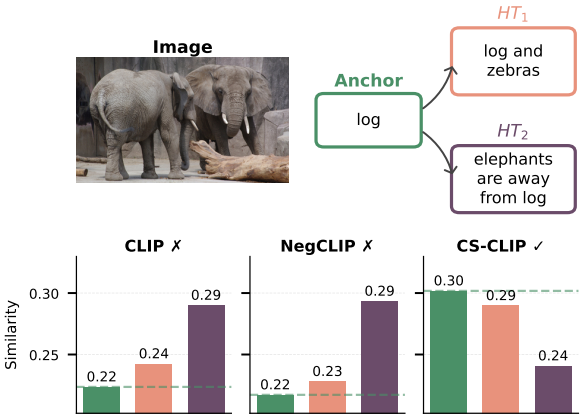

Figure 1: **The half-truth vulnerability.** Starting from a short, caption-supported description (the anchor), we form a *half-truth* by adding one realistic but incorrect detail. CLIP and NegCLIP assign *higher* similarity to the half-truth, while **CS-CLIP (ours)** correctly penalizes the incorrect addition.

Why does this happen? Contrastive training aligns images with *full captions*. This provides strong supervision at the sentence level, but weak supervision on the individual *units* that compose the caption's meaning. As a result, similarity can be dominated by coarse overlap (e.g., detecting the right objects), and an additional plausible unit can increase similarity even when it is incorrect, especially for relations and role-sensitive structure that require verifying *how* entities are composed.

Our approach: supervision on units. We propose **CS-CLIP** (**Component-Supervised CLIP**), which adds explicit supervision on caption units during fine-tuning. We parse each caption into *entity units* (noun phrases with bound attributes) and *relation units* (directed relations between entities). For each unit, we generate a minimally edited foil that preserves fluency and context while changing meaning (e.g., "brown horse" $\rightarrow$ "white horse", or "horse near barn" $\rightarrow$ "horse inside barn"). During training, we contrast each correct unit against its foil, teaching the model to respond to fine-grained compositional differences. Critically, this supervision is applied *only during training*, at test time, CS-CLIP uses the same dual-encoder architecture and the same retrieval scoring as standard CLIP.

CS-CLIP achieves 76.4% Half-Truth Accuracy on COCO with Qwen-generated foils, compared to 34.0% for CLIP and 53.8% for NegCLIP. It also reaches 67.2% on COCO with Mistral-generated foils and 66.2% on CC3M, supporting transfer across foil generators and image domains. It also improves on 16 established compositional benchmarks, achieving the best average Image-to-Text accuracy (57.8%) and Group Accuracy among evaluated models.

Our contributions are:

- **Diagnostic.** We introduce the half-truth diagnostic, which tests whether adding one incorrect detail increases similarity. CLIP fails this test in 66.0% of COCO (Qwen) cases.
- **Method.** CS-CLIP fine-tunes CLIP with supervision on units: each entity and relation unit is contrasted against a matched foil. This achieves 76.4% Half-Truth Accuracy (vs. 34.0% for CLIP) while preserving standard dual-encoder inference.
- **Compositional benchmarks.** CS-CLIP achieves the best average I2T accuracy (57.8%) and Group Accuracy across 16 compositional benchmarks, outperforming CLIP by 5.8 percentage points.

2 Related Work

Contrastive vision-language pretraining and compositional sensitivity. Contrastive dual encoders such as CLIP and ALIGN learn a shared embedding space for images and text, enabling efficient

retrieval via a single similarity score Radford et al. (2021); Jia et al. (2021). A recurring concern is that this score can under-use linguistic structure: CLIP-style models behave like bag-of-words under minimal edits that change meaning, including attribute binding, word order, and relations Yuksekgonul et al. (2023); Thrush et al. (2022). Recent analysis suggests this is largely a cross-modal issue: binding cues may be present within modalities but not reliably reflected in cosine alignment Koishigarina et al. (2026); other work reports scoring biases that favor central objects or frequent attribute-object pairings Schrodi et al. (2025); Tang et al. (2023). To evaluate this, controlled-edit benchmarks test whether models distinguish correct image-text pairs from minimally edited alternatives, including ARO, Winoground, SugarCrepe, and VALSE Yuksekgonul et al. (2023); Thrush et al. (2022); Hsieh et al. (2023); Parcalabescu et al. (2022), as well as targeted tests for specific phenomena Burapacheep et al. (2024); Kamath et al. (2023); Paiss et al. (2023).

Adding information and our diagnostic. Beyond swaps and reorderings, several works study how model confidence changes as more information is added to an input. HiMo-CLIP targets monotonic alignment under correct additions Wu et al. (2026), while confidence inflation patterns in LLM reasoning show that added plausible context can inflate confidence under misleading inputs Suri et al. (2024); Richardson et al. (2025); Jiang et al. (2024a). Our work is complementary: we test whether similarity properly penalizes incorrect additions through the half-truth diagnostic, which appends one realistic but incorrect detail to a correct description (Section 3).

Training methods for compositional sensitivity. Many methods strengthen sensitivity through data-level signals: sentence-level hard negatives that edit objects, attributes, or relations Yuksekgonul et al. (2023); Zhang et al. (2024); Peleg et al. (2025); Singh et al. (2025); Patel et al. (2024), scene-graph perturbations Singh et al. (2023), or caption synthesis Fan et al. (2023); Doveh et al. (2023a); Castro et al. (2024); Doveh et al. (2023b); Nayak et al. (2023). Other approaches add architectural components or auxiliary objectives such as region alignment Zhong et al. (2022); Huang et al. (2024), auxiliary modules Yao et al. (2021); Goel et al. (2022); Abdollah et al. (2024); Oh et al. (2024); Kwon et al. (2025), or multi-stage training Hu et al. (2025), while some modify inference through decomposition Jiang et al. (2024b); Miranda et al. (2025); Menon & Vondrick (2023). We build on sentence-level negatives but add targeted unit-level supervision: for each caption, we parse entity and relation units and contrast each against a minimally edited foil. This provides direct compositional pressure on individual caption parts while maintaining the standard dual-encoder architecture (Section 4).

3 The Half-Truth Vulnerability

3.1 Motivation

Prior work on compositional sensitivity in CLIP-style models has mainly studied minimal edits that *rearrange* existing information, such as swapping attributes or changing role assignments Yuksekgonul et al. (2023); Thrush et al. (2022). We study a complementary retrieval failure: *adding* incorrect information.

In retrieval, users often start from a short but correct description and then refine it by appending more details. If the added detail is wrong, the refined query should become *less* relevant to the image, not more. Yet CLIP-style similarity can violate this basic monotonicity intuition: adding one plausible but false detail can increase the score. We call this failure the **half-truth vulnerability**.

3.2 Half-Truth Diagnostic: Metric and Construction

Anchor vs. half-truth. Given an image I, we compare a short correct description, the **anchor** A, with a **half-truth** A^- formed by appending exactly one additional detail that is plausible in context but incorrect for the image. The desired ordering is simple: an incorrect refinement should not outrank a correct but incomplete one. We therefore report **Half-Truth Accuracy** as the fraction of cases where the model prefers the anchor over the half-truth:

$$\mathrm{Acc}_{\mathrm{HT}} \;=\; \Pr\left[s(I, A) > s(I, A^-)\right], \tag{1}$$

where $s(\cdot, \cdot)$ is the image–text similarity score (higher is better). Random choice corresponds to 50%. Values below 50% mean that the model often assigns higher similarity to the incorrect refinement than to the correct anchor.

Diagnostic construction. We construct COCO comparisons under the Karpathy partition Karpathy & Fei-Fei (2015) using Qwen3-8B and Mistral-8B foil generators, and CC3M comparisons using Qwen3-8B. We first construct candidate half-truths automatically from captions, then use manual verification to retain only clean examples for the main paper results. Figure 2 illustrates the construction.

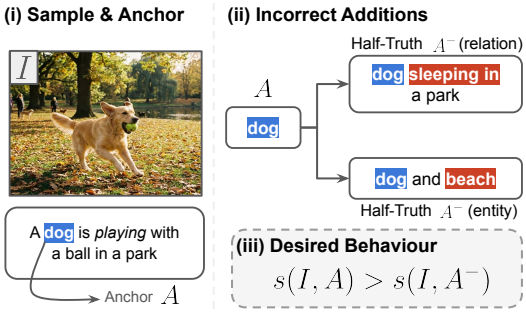

Figure 2: **Half-truth construction.** Parse captions into units and generate foils via minimal edits, then sample an anchor A and append one foil to form half-truth A^-.

We parse each caption T into two kinds of textual units using a text-only LLM pipeline (Appendix C.1): **entity units** $\mathcal{E}(T)$, noun phrases including attributes and quantifiers (e.g., "brown horse", "three dogs"), and **relation units** $\mathcal{R}(T)$, directed relations between two entities (e.g., "person riding horse", "ball in park").

For each unit, the same pipeline proposes minimally edited *matched foils*, incorrect variants that are plausible in context. For entity units, foils change either the object (e.g., "brown horse" $\rightarrow$ "brown giraffe") or an attribute (e.g., "brown horse" $\rightarrow$ "white horse").

For relation units, foils change the predicate, swap arguments, or replace one entity. The anchor A is sampled from the entity units $\mathcal{E}(T)$, and a half-truth A^- is formed by appending exactly one foil from either $\mathcal{E}(T)$ or $\mathcal{R}(T)$. When appending a relation unit, the relation introduces a second entity and describes an interaction or spatial relationship between it and the anchor entity. This construction ensures that half-truths differ from anchors by a single, targeted error.

Verification. We screen candidate foils with Qwen3-VL-30B and use Claude Opus 5 to review them and propose alternatives within the same foil category. The authors manually check the final evaluation candidates, including replacements, against their images and discard foils judged correct or ambiguous. The evaluation contains 1,437 comparisons: 757 entity and 680 relation additions across COCO (Qwen), COCO (Mistral), and CC3M (Qwen). Manual verification applies only to evaluation; training foils are generated and filtered automatically. Appendix C.2 gives the source-wise counts and training-foil audit.

3.3 Half-Truth Vulnerability on COCO

On COCO (Qwen), CLIP reaches 34.0% overall accuracy: 40.2% for entity additions and 27.8% for relation additions (Figure 3). NegCLIP improves these to 62.2% and 45.5%, respectively, leaving a substantial relation gap. CS-CLIP reaches 73.6% on entities and 79.2% on relations.

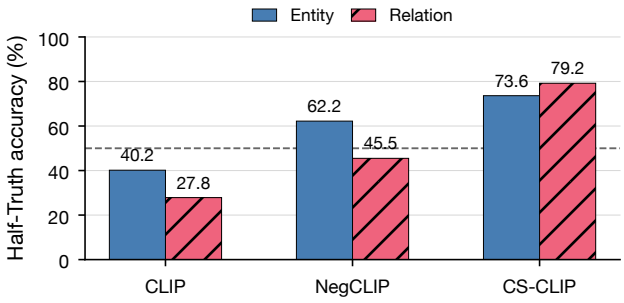

Figure 3: **Half-Truth Accuracy on COCO (Qwen).** Entity versus relation additions; dashed line indicates chance.

Why unit-level supervision? Half-truths test whether similarity responds to the specific added detail. This motivates contrasting each caption unit against a matched foil during training, alongside the full-caption objective. Section 5.4 compares sentence-level and unit-level use of these foils.

4 Method: CS-CLIP

The half-truth diagnostic reveals that models struggle when incorrect information is added to correct descriptions. To address this, we introduce unit-level supervision during fine-tuning. Rather than

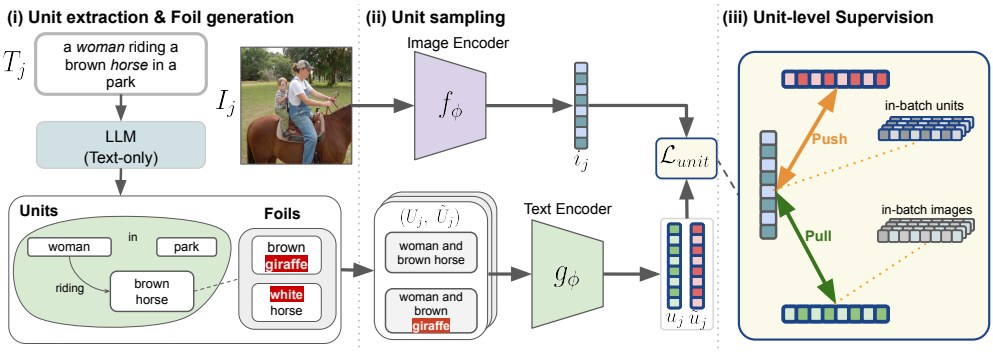

Figure 4: **CS-CLIP training pipeline.** (i) A text-only LLM extracts entity and relation units from each caption and creates matched foils via minimal edits. (ii) For each image-caption pair, we sample unit–foil pairs and encode the image and unit texts with the dual encoder. (iii) The unit loss scores the correct unit above its matched foil and other in-batch units, while the standard sentence-level contrastive loss is applied in parallel.

training on anchor-vs-half-truth comparisons directly, we provide supervision at the unit level: for each caption unit, we contrast the correct unit against a minimally edited foil (e.g., "brown horse" vs. "white horse"). This targets the underlying weakness, sensitivity to compositional structure, making the approach general beyond the half-truth setting.

Standard CLIP training aligns each image with its full caption using a global contrastive objective, which can leave the similarity score insensitive to individual caption parts (Section 3). To address this, we propose CS-CLIP (Component-Supervised CLIP), which adds targeted unit-level supervision during fine-tuning while keeping the dual-encoder architecture and standard cosine scoring unchanged at test time.

For each image-caption pair, we parse the caption into entity units and relation units (defined in Section 3), sample a unit with its matched foil, and train the image embedding to score the correct unit higher than the foil. This direct supervision on caption parts makes similarity more sensitive to individual details while preserving full-caption alignment through a parallel sentence-level loss. Figure 4 illustrates the complete training pipeline.

4.1 Unit Sampling and Encoding

We reuse the text-only parsing pipeline from Section 3 to extract entity units $\mathcal{E}(T)$ and relation units $\mathcal{R}(T)$ from each caption T, along with matched foils for each unit (Appendix B provides implementation details).

Sampling strategy. For each image-caption pair, we sample one unit U from either entities or relations. With probability p_{rel}, we sample a relation unit $U \in \mathcal{R}(T)$; otherwise, we sample an entity unit $U \in \mathcal{E}(T)$, optionally forming conjunctions of up to $K{=}2$ entities. We sample the unit's matched foil $\tilde{U}$ and encode both with the text encoder $g_\phi(\cdot)$ as noun phrases (entities) or predicate phrases (relations), yielding normalized embeddings. We investigate other sampling strategies in Appendix F.

4.2 Training Objective

Sentence-level loss ($\mathcal{L}_{\text{global}}$). We use a standard global image-caption contrastive objective with NegCLIP-style hard negatives Yuksekgonul et al. (2023) (Appendix A), which we combine with unit-level supervision to improve sensitivity to compositional structure.

Unit-level loss ($\mathcal{L}_{\text{unit}}$). For unit supervision, we use the same cosine scoring as CLIP with a learnable temperature τ. Since embeddings are normalized, cosine similarity equals a dot product, and we define $\kappa(\mathbf{x}, \mathbf{y}) = \exp(\mathbf{x}^\top \mathbf{y}/\tau)$.

For each image I_i, we sample N_u unit/foil pairs $\{(U_{i,k}, \tilde{U}_{i,k})\}_{k=1}^{N_u}$. Let $\mathbf{v}_i$ denote the normalized image embedding, and let $\mathbf{u}_{i,k}$ and $\tilde{\mathbf{u}}_{i,k}$ denote the normalized embeddings of the k-th unit and its matched foil for image i. For a fixed unit index k, the image-to-unit and symmetric unit-to-image

Table 1: **Half-Truth Accuracy across foil generators and image domains.** We report percentages on 509 COCO (Qwen), 454 COCO (Mistral), and 474 CC3M (Qwen) manually verified comparisons. All is the sample-weighted entity/relation average. FSC-CLIP is trained on CC3M; the other fine-tuned models use COCO. Higher is better.

	COCO (Qwen)			COCO (Mistral)			CC3M (Qwen)		
Model	All	Entity	Relation	All	Entity	Relation	All	Entity	Relation
CLIP	34.0	40.2	27.8	28.6	34.5	21.3	36.1	46.6	24.2
NegCLIP	53.8	62.2	45.5	45.2	54.8	33.2	55.3	67.7	41.3
DeGLA	46.2	59.1	33.3	40.1	53.2	23.8	52.3	64.9	38.1
ReadCLIP	58.0	66.1	49.8	50.9	60.3	39.1	60.1	65.7	53.8
FSC-CLIP (CC3M)	61.3	70.9	51.8	54.0	58.3	48.5	**67.5**	**76.5**	57.4
CS-CLIP	**76.4**	**73.6**	**79.2**	**67.2**	**70.2**	**63.4**	66.2	68.9	**63.2**

losses are:

$$\mathcal{L}^{(k)}_{I\to U} = -\frac{1}{B}\sum_{i=1}^{B}\log\frac{\kappa(\mathbf{v}_i,\mathbf{u}_{i,k})}{\sum_{j=1}^{B}\kappa(\mathbf{v}_i,\mathbf{u}_{j,k})+\kappa(\mathbf{v}_i,\tilde{\mathbf{u}}_{i,k})},\ \mathcal{L}^{(k)}_{U\to I} = -\frac{1}{B}\sum_{i=1}^{B}\log\frac{\kappa(\mathbf{v}_i,\mathbf{u}_{i,k})}{\sum_{j=1}^{B}\kappa(\mathbf{v}_j,\mathbf{u}_{i,k})}. \tag{2}$$

$\mathcal{L}^{(k)}_{I\to U}$ trains the image to score the correct unit $U_{i,k}$ higher than (i) its matched foil $\tilde{U}_{i,k}$ and (ii) in-batch units from other images. We exclude foils from $\mathcal{L}^{(k)}_{U\to I}$ since each foil is tied to a specific image-unit pair; the symmetric term provides bidirectional regularization. We average over the N_u sampled unit indices: $\mathcal{L}_{\text{unit}} = \frac{1}{2N_u}\sum_{k=1}^{N_u}(\mathcal{L}^{(k)}_{I\to U} + \mathcal{L}^{(k)}_{U\to I})$.

Final objective. We combine global alignment with unit-level supervision:

$$\mathcal{L}_{\text{CS}} = \mathcal{L}_{\text{global}} + \lambda_u\,\mathcal{L}_{\text{unit}}. \tag{3}$$

This objective keeps standard CLIP inference unchanged, but makes the learned similarity more sensitive to caption structure by training the model to distinguish correct caption units from their matched foils.

5 Experiments

We evaluate CS-CLIP in three settings. First, we measure robustness to incorrect additions using the half-truth diagnostic (Section 3). Second, we evaluate compositional generalization on established controlled-edit benchmarks. Third, we report downstream zero-shot classification and image–text retrieval results, and we include ablations to isolate the effect of each design choice.

Implementation and training. We fine-tune CLIP using the OpenCLIP codebase Cherti et al. (2023) with OpenAI CLIP initialization. Unless stated otherwise, we use ViT-B/32 Radford et al. (2021). We fine-tune on MS-COCO under the Karpathy split Karpathy & Fei-Fei (2015) for 25 epochs with batch size 128 on 8×A100 GPUs, using AdamW (lr 5×10^{-6}, weight decay 10^{-2}). We use cosine warm-up for the first three epochs followed by cosine decay. For the unit-level objective, we sample $N_u = 2$ unit/foil pairs per image per step and use a learnable temperature τ initialized from the pretrained CLIP temperature. We report results from the final training checkpoint.

Training objective. We set the unit-loss weight to $\lambda_u = 0.5$. For global caption-level negatives we follow NegCLIP-style training Yuksekgonul et al. (2023), using in-batch captions plus shuffled synthetic hard negatives. Caption parsing and unit sampling follow Section 4.1; implementation details are in Appendix C.1.

5.1 Half-Truth Vulnerability

Table 1 reports Half-Truth Accuracy on the three evaluation sources. COCO (Mistral) changes the foil generator, while CC3M (Qwen) changes the image domain.

COCO results. CS-CLIP reaches 76.4% on COCO (Qwen), improving over CLIP by 42.4 points and FSC-CLIP by 15.1 points. The largest gain is on relations: CLIP reaches 27.8%, NegCLIP 45.5%, and FSC-CLIP 51.8%, compared to 79.2% for CS-CLIP. CS-CLIP also leads on entity additions at 73.6%.

Cross-generator and cross-domain transfer. With Mistral-generated COCO foils, CS-CLIP reaches 67.2%, compared to 28.6% for CLIP and 54.0% for FSC-CLIP. On CC3M, CS-CLIP improves over CLIP from 36.1% to 66.2%; CC3M-trained FSC-CLIP leads overall at 67.5%. CS-CLIP remains strongest on relations among these baselines across all three sources (79.2%, 63.4%, and 63.2%). Its mean accuracy across sources is 69.9%, versus 60.9% for FSC-CLIP.

Correct refinements. CS-CLIP also ranks a correct refinement above the anchor and the half-truth below it more often than the displayed baselines: Correct Ordering Rate is 58.2%, 50.4%, and 42.2% on COCO (Qwen), COCO (Mistral), and CC3M (Qwen), respectively (Appendix C.3). This supports sensitivity to the added content rather than a uniform preference for shorter descriptions.

5.2 Compositional Understanding

The half-truth diagnostic isolates sensitivity to a single incorrect added detail. We next test whether CS-CLIP improves broader compositional understanding on controlled-foil benchmarks covering attribute binding, relations, and linguistic variation, including ARO Yuksekgonul et al. (2023), SugarCrepe Hsieh et al. (2023), and Winoground Thrush et al. (2022). We report I2T, T2I, and, for paired datasets, Group Accuracy (Grp); full protocols are in Appendix E.1.

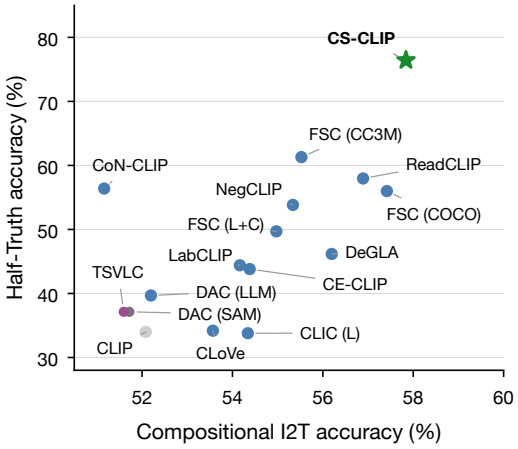

Figure 5: **Compositional I2T vs. Half-Truth Accuracy.** Sixteen models; Half-Truth Accuracy is measured on COCO (Qwen). L: LAION; C: COCO.

Overall compositional performance. Figure 5 plots average compositional I2T accuracy against Half-Truth Accuracy across models. CS-CLIP achieves the best performance on both metrics: **57.8%** compositional I2T (+5.8 pp over CLIP; Table 12) and **76.4%** $\mathrm{Acc}_{\mathrm{HT}}$ on COCO (Qwen). The gains on independently constructed benchmarks support transfer beyond the half-truth diagnostic.

Dataset-wise breakdown. CS-CLIP improves over NegCLIP on 14/16 benchmarks (Figure 6) and outperforms the next best COCO-trained baselines by +0.4 pp (FSC-CLIP), +0.9 pp (ReadCLIP), and +1.6 pp (DeGLA). CS-CLIP preserves the dual-encoder architecture and test-time scoring. CS-CLIP achieves the best VL Checklist Zhao et al. (2022) score (79.2%), a systematic audit of objects, attributes, and relations.

Paired benchmarks. On paired compositional datasets, CS-CLIP also improves Text-to-Image accuracy and achieves the best average Group Accuracy among evaluated models, indicating that gains do not come at the expense of weaker matching in the opposite retrieval direction. Full bidirectional results are in Appendix E.5.

Main takeaway. CS-CLIP achieves the best average compositional I2T accuracy (57.8%, +5.8 pp over CLIP) and the best average Group Accuracy among evaluated models. Improvements are consistent across datasets and aligned with half-truth robustness, demonstrating that unit-level supervision addresses compositional understanding broadly rather than overfitting to specific tests.

5.3 Downstream Performance

We evaluate downstream performance with CLIPBench Cherti & Beaumont (2022) on zero-shot classification (ImageNet variants and standard recognition datasets) and image-text retrieval (COCO and Flickr8k Plummer et al. (2015)). Full per-dataset results and evaluation details are in Appendix E.6.

Zero-shot classification. Table 15 shows zero-shot classification results. Compared to CLIP, CS-CLIP changes average accuracy modestly (Acc@1: **63.6** → **59.9**; Acc@5: **86.5** → **84.6**). This drop is comparable to other COCO-fine-tuned baselines (NegCLIP: 58.2%, FSC-CLIP: 61.2%), indicating that unit-level supervision trades off zero-shot accuracy similarly to other fine-tuning approaches. The

Table 2: **Backbone scaling and fine-tuning strategy.** For ViT-B/32, we compare **Full FT** (updating both encoders) against **Text-only FT** and **Image-only FT**. We also report backbone scaling under Full FT.

Model	Compositionality			Half-Truth			ZS	Retr	
	I2T	T2I	Grp	All	Ent	Rel	Avg	T2I	I2T
ViT-B/32 (fine-tuning strategy)									
ViT-B/32 Full FT (Ours)	57.8	39.8	27.8	76.4	73.6	79.2	59.9	**71.7**	**56.8**
ViT-B/32 Text-only FT	55.9	38.0	26.0	62.5	74.4	50.6	**61.4**	69.5	53.0
ViT-B/32 Image-only FT	56.2	37.9	27.4	59.5	72.0	47.1	58.6	65.6	54.0
Backbone scaling (Full FT)									
ViT-B/16	58.5	39.8	28.7	78.8	85.0	72.5	62.5	74.8	61.6
ViT-L/14	**59.5**	**42.4**	**31.4**	80.4	86.6	74.1	**65.9**	**79.9**	**67.1**

Table 3: **Effect of the unit-loss weight λ_u.** We vary λ_u while keeping all other settings fixed.

λ_u	Compositionality			Half-Truth			ZS	Retr	
	I2T	T2I	Grp	All	Ent	Rel	Avg	T2I	I2T
0.10	56.9	38.4	26.4	73.1	70.9	75.3	**60.7**	70.8	56.7
0.25	56.7	39.4	27.2	75.4	73.6	77.3	59.3	71.0	56.5
0.50 (Ours)	**57.8**	**39.8**	27.8	76.4	73.6	79.2	59.9	**71.7**	56.8
0.75	57.7	39.3	**27.9**	77.6	73.6	81.6	59.8	71.1	**56.9**
1.00	57.2	38.5	27.2	78.0	76.0	80.0	60.6	71.1	56.4

unit-only ablation reaches 61.8%, close to plain COCO fine-tuning at 62.0%; global hard-negative training produces a larger drop (NegCLIP: 58.2%).

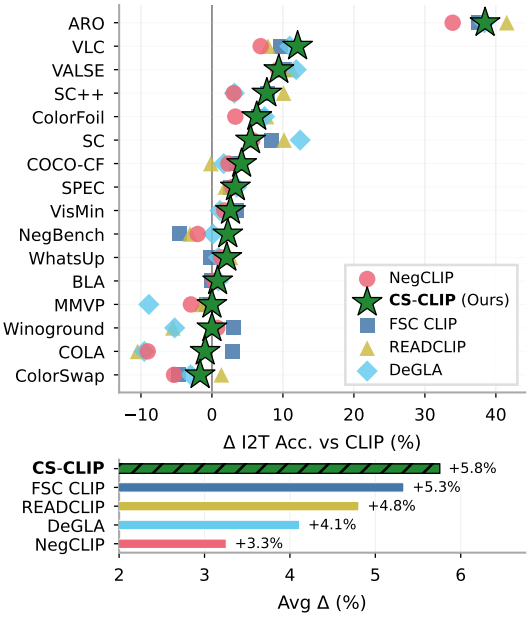

Figure 6: **Per-benchmark compositional gains.** ΔI2T relative to zero-shot CLIP by benchmark and on average.

Retrieval tasks. Table 16 reports Recall@1 on COCO and Flickr8k Plummer et al. (2015). Since CS-CLIP is fine-tuned on COCO, we treat retrieval as an internal comparison among COCO-fine-tuned models. CS-CLIP achieves the **best average T2I Recall@1** (**71.7**, +5.8 pp over NegCLIP and +1.9 pp over FSC-CLIP) and ties for the **best average I2T** (**56.8**), demonstrating that unit-level supervision improves retrieval alignment while maintaining competitive performance in both directions.

5.4 Ablations

Backbone and fine-tuning strategy. Table 2 shows that updating both encoders reaches 79.2% relation accuracy, compared to 50.6% for text-only and 47.1% for image-only tuning. Larger backbones improve overall Half-Truth accuracy and downstream performance.

Unit-loss weight. Increasing λ_u from 0.10 to 0.75 raises relation accuracy from 75.3% to 81.6%, while compositional I2T varies less (Table 3). We use $\lambda_u = 0.5$ to balance these metrics.

Training signals. Table 4 separates global negatives, unit supervision, and matched unit foils. With global negatives and unit supervision, adding matched foils raises Half-Truth accuracy from 47.7% to 76.4%, and relation accuracy from 30.6% to 79.2%. Unit supervision alone yields

Table 4: **Training-signal ablations.** G: global sentence negatives; U: unit supervision; F: matched unit foils. Half-Truth accuracy uses COCO (Qwen); ZS is average zero-shot classification accuracy.

		Compositionality			Half-Truth			Downstream		
Variant	G/U/F	I2T	T2I	Group	All	Entity	Relation	ZS	Retr. T2I	I2T
1	✗✗✗	52.8	39.4	27.2	43.8	62.6	25.1	62.0	71.7	56.7
2	✓✗✗	55.3	38.5	25.7	53.8	62.2	45.5	58.2	65.9	52.7
3	✗✓✗	52.5	39.8	28.0	46.0	66.9	25.1	61.8	71.3	56.2
4	✗✓✓	54.8	39.1	28.3	76.2	74.8	77.6	60.7	70.7	56.7
5	✓✓✗	55.2	38.6	27.3	47.7	65.0	30.6	59.6	70.9	56.6
6	✓✓✓	57.8	39.8	27.8	76.4	73.6	79.2	59.9	71.7	56.8

Table 5: **Foil placement and generator controls.** Half-Truth columns report entity/relation accuracy (%); Comp. is average compositional I2T accuracy.

Model	COCO (Qwen)	COCO (Mistral)	CC3M (Qwen)	Comp.
NegCLIP	62.2 / 45.5	54.8 / 33.2	67.7 / 41.3	55.3
Sentence entity foils	71.7 / 36.5	65.5 / 31.2	66.5 / 34.1	55.8
Sentence entity+relation foils	66.5 / 82.7	59.5 / 69.3	62.2 / 72.2	56.8
CS-CLIP (Qwen)	73.6 / 79.2	70.2 / 63.4	68.9 / 63.2	57.8
CS-CLIP (Mistral)	66.9 / 60.4	65.5 / 55.0	64.5 / 48.0	57.3

61.8% zero-shot accuracy, close to plain COCO fine-tuning at 62.0%. Combining all signals gives 57.8% compositional I2T and 59.9% zero-shot accuracy.

Matched foils and supervision granularity. Table 5 compares sentence-level foil training with unit-level supervision. Adding entity and relation foils at the sentence level raises COCO (Qwen) relation accuracy from NegCLIP's 45.5% to 82.7%. Entity-only sentence foils reach just 36.5%, highlighting the contribution of relation foils. CS-CLIP improves entity accuracy from 66.5% to 73.6% and compositional I2T from 56.8% to 57.8% relative to the entity+relation sentence control, while relation accuracy is slightly lower (79.2%). Thus, matched foils drive much of the relation gain; unit supervision improves the balance across capabilities.

Generator sensitivity. Training with Mistral foils retains gains over CLIP on all three sources and reaches 57.3% compositional I2T, compared to 57.8% for Qwen-trained CS-CLIP. Its Half-Truth accuracy drops most on relations: 60.4% versus 79.2% on COCO (Qwen). Generator quality therefore matters, although improvements transfer across generators. Appendix F gives configurations and additional construction and optimization ablations.

6 Conclusion

We study a simple retrieval failure in CLIP-style models: appending one realistic but incorrect detail to a correct description can increase similarity, making the half-truth rank above the anchor. To address this, we introduce CS-CLIP, which adds unit-level supervision during fine-tuning by parsing captions into entity and relation units and contrasting each against a minimally edited foil. This provides direct compositional pressure while maintaining the standard dual-encoder architecture. CS-CLIP achieves 76.4% Half-Truth Accuracy on COCO (Qwen), compared to 34.0% for CLIP, and retains gains across foil generators and image domains. Its 57.8% compositional I2T accuracy shows that these improvements extend to independently constructed benchmarks.

Limitations and future work. Our approach relies on text-only LLM parsing, which may introduce artifacts or miss visual details not expressed in captions, and fine-tuning on COCO trades some zero-shot accuracy for compositional sensitivity. While CS-CLIP reduces half-truth failures, it does not guarantee factual correctness or demographic fairness; models may still reflect biases present in training data. Future work could explore image-side half-truths (adding incorrect visual elements to correct images), joint image-text parsing that leverages visual grounding, or applying unit-level supervision during large-scale pretraining to reduce the zero-shot trade-off.

Acknowledgments and Disclosure of Funding

The author B. K. acknowledges support from the Zuse School ELIZA, which provided essential funding that enabled their contributions to this work.

References

Abdollah, A., Izadi, A., Saghafian, A., Vahidimajd, R., Mozafari, M., Mirzaei, A., Samiei, M., and Baghshah, M. S. Comalign: Compositional alignment in vision-language models, 2024. URL https://arxiv.org/abs/2409.08206.

Alhamoud, K., Alshammari, S., Tian, Y., Li, G., Torr, P., Kim, Y., and Ghassemi, M. Vision-language models do not understand negation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

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/pdf/2407.16772.

Burapacheep, J., Gaur, I., Bhatia, A., and Thrush, T. ColorSwap: A color and word order dataset for multimodal evaluation. In Ku, L.-W., Martins, A., and Srikumar, V. (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 1716–1726, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.99. URL https://aclanthology.org/2024.findings-acl.99/.

Castro, S., Ziai, A., Saluja, A., Yuan, Z., and Mihalcea, R. CLoVe: Encoding compositional language in contrastive vision-language models. arXiv:2402.15021, February 2024. URL https://arxiv.org/abs/2402.15021.

Chen, X., Fernández, R., and Pezzelle, S. The BLA benchmark: Investigating basic language abilities of pre-trained multimodal models. In Bouamor, H., Pino, J., and Bali, K. (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5817–5830, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.356. URL https://aclanthology.org/2023.emnlp-main.356/.

Cherti, M. and Beaumont, R. Clip benchmark, November 2022. URL https://doi.org/10.5281/zenodo.15403103.

Cherti, M., Beaumont, R., Wightman, R., Wortsman, M., Ilharco, G., Gordon, C., Schuhmann, C., Schmidt, L., and Jitsev, J. Reproducible scaling laws for contrastive language-image learning. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2829. IEEE, June 2023. doi: 10.1109/cvpr52729.2023.00276. URL http://dx.doi.org/10.1109/CVPR52729.2023.00276.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

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 Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 76137–76150, 2023a. URL https://openreview.net/pdf?id=ARrwf7Ev2T.

Doveh, S., Arbelle, A., Harary, S., Panda, R., Herzig, R., Schwartz, E., Kim, D., Giryes, R., Feris, R., Ullman, S., and Karlinsky, L. Teaching structured vision & language concepts to vision & language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2657–2668, 2023b. URL https://openaccess.thecvf.com/content/CVPR2023/papers/Doveh_Teaching_Structured_Vision__Language_Concepts_to_Vision__Language_CVPR_2023_paper.pdf.

Dumpala, S. H., Jaiswal, A., Sastry, C., Milios, E., Oore, S., and Sajjad, H. Sugarcrepe++ dataset: Vision-language model sensitivity to semantic and lexical alterations. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2024.

Fan, L., Krishnan, D., Isola, P., Katabi, D., and Tian, Y. Improving clip training with language rewrites. In *NeurIPS*, 2023.

Fei-Fei, L., Fergus, R., and Perona, P. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 Conference on Computer Vision and Pattern Recognition Workshop*, pp. 178–178, 2004. doi: 10.1109/CVPR.2004.383.

Goel, S., Bansal, H., Bhatia, S., Rossi, R. A., Vinay, V., and Grover, A. Cyclip: Cyclic contrastive language-image pretraining, 2022. URL https://arxiv.org/abs/2205.14459.

Hendrycks, D., Zhao, K., Basart, S., Steinhardt, J., and Song, D. Natural adversarial examples, 2021. URL https://arxiv.org/abs/1907.07174.

Hsieh, C.-Y., Zhang, J., Ma, Z., Kembhavi, A., and Krishna, R. Sugarcrepe: Fixing hackable benchmarks for vision-language compositionality. In *Thirty-Seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023.

Hu, X., Yang, K., Wang, J., Xu, H., Feng, Z., and Wang, Y. Decoupled global-local alignment for improving compositional understanding. In *Proceedings of the 33rd ACM International Conference on Multimedia (ACM MM)*, 2025. doi: 10.1145/3746027.3755032. URL https://arxiv.org/abs/2504.16801.

Huang, Y., Tang, J., Chen, Z., Zhang, R., Zhang, X., Chen, W., Zhao, Z., Zhao, Z., Lv, T., Hu, Z., and Zhang, W. Structure-clip: Towards scene graph knowledge to enhance multi-modal structured representations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(3):2417–2425, Mar. 2024. doi: 10.1609/aaai.v38i3.28017. URL https://ojs.aaai.org/index.php/AAAI/article/view/28017.

Jia, C., Yang, Y., Xia, Y., Chen, Y.-T., Parekh, Z., Pham, H., Le, Q. V., Sung, Y., Li, Z., and Duerig, T. Scaling up visual and vision-language representation learning with noisy text supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139. PMLR, 2021.

Jiang, B., Xie, Y., Hao, Z., Wang, X., Mallick, T., Su, W. J., Taylor, C. J., and Roth, D. A peek into token bias: Large language models are not yet genuine reasoners. In Al-Onaizan, Y., Bansal, M., and Chen, Y.-N. (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 4722–4756, Miami, Florida, USA, November 2024a. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.272. URL https://aclanthology.org/2024.emnlp-main.272/.

Jiang, K., He, X., Xu, R., and Wang, X. ComCLIP: Training-free compositional image and text matching. In Duh, K., Gomez, H., and Bethard, S. (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 6639–6659, Mexico City, Mexico, June 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.370. URL https://aclanthology.org/2024.naacl-long.370/.

Kamath, A., Hessel, J., and Chang, K.-W. What's "up" with vision-language models? investigating their struggle with spatial reasoning. In Bouamor, H., Pino, J., and Bali, K. (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 9161–9175, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.568. URL https://aclanthology.org/2023.emnlp-main.568/.

Karpathy, A. and Fei-Fei, L. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3128–3137, 2015.

Koishigarina, D., Uselis, A., and Oh, S. J. Clip behaves like a bag-of-words model cross-modally but not uni-modally. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://arxiv.org/abs/2502.03566.

Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.-J., Shamma, D. A., Bernstein, M. S., and Fei-Fei, L. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123:32–73, 2017. doi: 10.1007/s11263-016-0981-7. URL https://link.springer.com/article/10.1007/s11263-016-0981-7.

Krizhevsky, A. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Kwon, J., Min, K., and Sohn, J.-y. Enhancing compositional reasoning in clip via reconstruction and alignment of text descriptions. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. URL https://arxiv.org/abs/2510.16540.

Le, T., Lal, V., and Howard, P. COCO-counterfactuals: Automatically constructed counterfactual examples for image-text pairs. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://openreview.net/forum?id=7AjdHnjIHX.

Lin, T.-Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C. L., and Dollár, P. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pp. 740–755. Springer, 2014.

Menon, S. and Vondrick, C. Visual classification via description from large language models. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=jlAjNL8z5cs.

Miranda, I., Salaberria, A., Agirre, E., and Azkune, G. Adding simple structure at inference improves vision-language compositionality, 2025. URL https://arxiv.org/abs/2506.09691.

Nayak, N. V., Yu, P., and Bach, S. H. Learning to compose soft prompts for compositional zero-shot learning. In *International Conference on Learning Representations*, 2023.

Oh, Y., Cho, J. W., Kim, D.-J., Kweon, I. S., and Kim, J. Preserving multi-modal capabilities of pre-trained vlms for improving vision-linguistic compositionality. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 2024.

Paiss, R., Ephrat, A., Tov, O., Zada, S., Mosseri, I., Irani, M., and Dekel, T. Teaching clip to count to ten. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

Parcalabescu, L., Cafagna, M., Muradjan, L., Frank, A., Calixto, I., and Gatt, A. VALSE: A task-independent benchmark for vision and language models centered on linguistic phenomena. In Muresan, S., Nakov, P., and Villavicencio, A. (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8253–8280, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.567. URL https://aclanthology.org/2022.acl-long.567/.

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. *Advances in neural information processing systems*, 2024.

Peleg, A., Singh, N. D., and Hein, M. Advancing compositional awareness in clip with efficient fine-tuning. In *NeurIPS*, 2025.

Peng, W., Xie, S., You, Z., Lan, S., and Wu, Z. Synthesize, diagnose, and optimize: Towards fine-grained vision-language understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13279–13288, 2024.

Plummer, B. A., Wang, L., Cervantes, C. M., Caicedo, J. C., Hockenmaier, J., and Lazebnik, S. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2641–2649, 2015.

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 (ICML 2021)*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021. URL http://proceedings.mlr.press/v139/radford21a.html.

Ray, A., Radenovic, F., Dubey, A., Plummer, B. A., Krishna, R., and Saenko, K. Cola: A benchmark for compositional text-to-image retrieval. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2305.03689.

Recht, B., Roelofs, R., Schmidt, L., and Shankar, V. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning*, 2019. URL https://api.semanticscholar.org/CorpusID:67855879.

Richardson, A. K., Kearns, R. O., Moss, S., Wang-Mascianica, V., and Koralus, P. Theory-grounded evaluation of human-like fallacy patterns in llm reasoning, 2025. URL https://arxiv.org/abs/2506.11128.

Samin, A. M., Ahmed, M. F., and Rafee, M. M. S. ColorFoil: Investigating color blindness in large vision and language models. In Ebrahimi, A., Haider, S., Liu, E., Haider, S., Leonor Pacheco, M., and Wein, S. (eds.), *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 4: Student Research Workshop)*, pp. 294–300, Albuquerque, USA, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-192-6. doi: 10.18653/v1/2025.naacl-srw.29. URL https://aclanthology.org/2025.naacl-srw.29/.

Schrodi, S., Hoffmann, D. T., Argus, M., Fischer, V., and Brox, T. Two effects, one trigger: On the modality gap, object bias, and information imbalance in contrastive vision-language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=uAFHCZRmXk.

Schuhmann, C., Vencu, R., Beaumont, R., Kaczmarczyk, R., Mullis, C., Katta, A., Coombes, T., Jitsev, J., and Komatsuzaki, A. LAION-400M: Open dataset of CLIP-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021. URL https://arxiv.org/abs/2111.02114.

Sharma, P., Ding, N., Goodman, S., and Soricut, R. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 2556–2565, 2018.

Singh, H., Zhang, P., Wang, Q., Wang, M., Xiong, W., Du, J., and Chen, Y. Coarse-to-fine contrastive learning in image-text-graph space for improved vision-language compositionality. In Bouamor, H., Pino, J., and Bali, K. (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 869–893, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.56. URL https://aclanthology.org/2023.emnlp-main.56/.

Singh, J., Shrivastava, I., Vatsa, M., Singh, R., and Bharati, A. Learning the power of "no": Foundation models with negations. In *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*, pp. 7991–8001, February 2025.

Suri, G., Slater, L. R., Ziaee, A., and Nguyen, M. Do large language models show decision heuristics similar to humans? a case study using gpt-3.5. *Journal of Experimental Psychology: General*, 153 (4):1066–1075, 2024.

Tang, Y., Yamada, Y., Zhang, Y., and Yildirim, I. When are lemons purple? the concept association bias of vision-language models. In Bouamor, H., Pino, J., and Bali, K. (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 14333–14348, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.886. URL https://aclanthology.org/2023.emnlp-main.886/.

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 *CVPR*, 2022.

Tong, S., Liu, Z., Zhai, Y., Ma, Y., LeCun, Y., and Xie, S. Eyes wide shut? exploring the visual shortcomings of multimodal LLMs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9568–9578, 2024.

Tschannen, M., Gritsenko, A., Wang, X., Naeem, M. F., Alabdulmohsin, I., Parthasarathy, N., Evans, T., Beyer, L., Xia, Y., Mustafa, B., Hénaff, O., Harmsen, J., Steiner, A., and Zhai, X. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025. URL https://arxiv.org/abs/2502.14786.

Tversky, A. and Kahneman, D. Extensional versus intuitive reasoning: The conjunction fallacy in probability judgment. *Psychological Review*, 90(4):293–315, 1983. doi: 10.1037/0033-295X.90.4.293.

Wang, H., Ge, S., Lipton, Z., and Xing, E. P. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pp. 10506–10518, 2019.

Wu, R., Chen, P., Shen, F., Zhao, S., Hui, Q., Gao, H., Lu, T., Liu, Z., Zhao, F., Wang, K., and Lian, S. Himo-clip: Modeling semantic hierarchy and monotonicity in vision-language alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026. URL https://arxiv.org/abs/2511.06653.

Xiao, J., Hays, J., Ehinger, K. A., Oliva, A., and Torralba, A. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 3485–3492, 2010. doi: 10.1109/CVPR.2010.5539970.

Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., Zheng, C., Liu, D., Zhou, F., Huang, F., Hu, F., Ge, H., Wei, H., Lin, H., Tang, J., Yang, J., Tu, J., Zhang, J., Yang, J., Yang, J., Zhou, J., Zhou, J., Lin, J., Dang, K., Bao, K., Yang, K., Yu, L., Deng, L., Li, M., Xue, M., Li, M., Zhang, P., Wang, P., Zhu, Q., Men, R., Gao, R., Liu, S., Luo, S., Li, T., Tang, T., Yin, W., Ren, X., Wang, X., Zhang, X., Ren, X., Fan, Y., Su, Y., Zhang, Y., Zhang, Y., Wan, Y., Liu, Y., Wang, Z., Cui, Z., Zhang, Z., Zhou, Z., and Qiu, Z. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Yao, L., Huang, R., Hou, L., Lu, G., Niu, M., Xu, H., Liang, X., Li, Z., Jiang, X., and Xu, C. Filip: Fine-grained interactive language-image pre-training, 2021. URL https://arxiv.org/abs/2111.07783.

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 *Proceedings of the International Conference on Learning Representations (ICLR 2023)*, 2023. URL https://arxiv.org/abs/2210.01936.

Zhai, X., Mustafa, B., Kolesnikov, A., and Beyer, L. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. URL https://openaccess.thecvf.com/content/ICCV2023/papers/Zhai_Sigmoid_Loss_for_Language_Image_Pre-Training_ICCV_2023_paper.pdf.

Zhang, L., Awal, R., and Agrawal, A. Contrasting intra-modal and ranking cross-modal hard negatives to enhance visio-linguistic compositional understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/papers/Zhang_Contrasting_Intra-Modal_and_Ranking_Cross-Modal_Hard_Negatives_to_Enhance_Visio-Linguistic_CVPR_2024_paper.pdf.

Zhao, T., Zhang, T., Zhu, M., Shen, H., Lee, K., Lu, X., and Yin, J. Vl-checklist: Evaluating pre-trained vision-language models with objects, attributes and relations, 2022. URL https://arxiv.org/abs/2207.00221.

Zhong, Y., Yang, J., Zhang, P., Li, C., Codella, N., Li, L. H., Zhou, L., Dai, X., Yuan, L., Li, Y., et al. Regionclip: Region-based language-image pretraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16793–16803, 2022.

A Sentence-Level Global Objective

This appendix defines the sentence-level contrastive objective $\mathcal{L}_{\text{global}}$ from Eq. 3 (Section 4.2), and summarizes how NegCLIP Yuksekgonul et al. (2023) modifies the denominator with synthetic hard negative captions.

We use the scoring notation $\kappa(\mathbf{x}, \mathbf{y}) = \exp(\mathbf{x}^\top \mathbf{y}/\tau)$ from Section 4.2, where all embeddings are ℓ_2-normalized so cosine similarity equals dot product.

A.1 CLIP global contrastive loss

Consider a minibatch of B matched image–caption pairs $\{(I_i, T_i)\}_{i=1}^B$. Let $\mathbf{v}_i$ be the normalized image embedding and $\mathbf{t}_i$ be the normalized caption embedding. The image-to-text (I2T) loss is

$$\mathcal{L}_{I\to T} = -\frac{1}{B}\sum_{i=1}^{B} \log \frac{\kappa(\mathbf{v}_i, \mathbf{t}_i)}{\sum_{j=1}^{B} \kappa(\mathbf{v}_i, \mathbf{t}_j)}. \tag{4}$$

The symmetric text-to-image (T2I) loss is

$$\mathcal{L}_{T\to I} = -\frac{1}{B}\sum_{i=1}^{B} \log \frac{\kappa(\mathbf{v}_i, \mathbf{t}_i)}{\sum_{j=1}^{B} \kappa(\mathbf{v}_j, \mathbf{t}_i)}. \tag{5}$$

We define the global CLIP objective as

$$\mathcal{L}_{\text{CLIP}} = \frac{1}{2}\Big(\mathcal{L}_{I\to T} + \mathcal{L}_{T\to I}\Big). \tag{6}$$

A.2 NegCLIP-style hard negative captions

NegCLIP Yuksekgonul et al. (2023) augments each training example with a *shuffled-caption* hard negative. Given a caption T_i, it constructs a foil $\tilde{T}_i$ by shuffling or swapping content words (e.g., nouns, adjectives) so that the new sentence reuses nearly the same vocabulary but changes the sentence-level meaning. Let $\tilde{\mathbf{t}}_i$ denote the normalized embedding of the shuffled caption $\tilde{T}_i$.

NegCLIP then adds these shuffled negatives to the image-to-text denominator. With a minibatch of B pairs, the I2T loss becomes

$$\mathcal{L}^{\text{Neg}}_{I\to T} = -\frac{1}{B}\sum_{i=1}^{B} \log \frac{\kappa(\mathbf{v}_i, \mathbf{t}_i)}{\sum_{j=1}^{B} \kappa(\mathbf{v}_i, \mathbf{t}_j) \;+\; \sum_{j=1}^{B} \kappa(\mathbf{v}_i, \tilde{\mathbf{t}}_j)}. \tag{7}$$

That is, each image $\mathbf{v}_i$ is contrasted against all B in-batch captions $\{\mathbf{t}_j\}$ plus all B shuffled-caption foils $\{\tilde{\mathbf{t}}_j\}$ from the same batch.

The text-to-image term remains unchanged from Eq. 5 (i.e., no hard negatives are added in the T2I direction). We define the NegCLIP objective as

$$\mathcal{L}_{\text{NegCLIP}} = \frac{1}{2}\Big(\mathcal{L}^{\text{Neg}}_{I\to T} + \mathcal{L}_{T\to I}\Big). \tag{8}$$

Relation to CS-CLIP. Unless otherwise stated, CS-CLIP uses the NegCLIP-style objective (Eq. 8) as $\mathcal{L}_{\text{global}}$. In our ablation studies (Appendix F), we also explore using the standard CLIP objective (Eq. 6) without hard negatives. We then add $\lambda_u \mathcal{L}_{\text{unit}}$ (Eq. 3) to provide supervision on individual caption units and their matched foils.

Table 6: Model and inference configuration for the text-only pipeline.

Parameter	Value
Model	`Qwen3-8B-AWQ`
Quantization	AWQ (4-bit)
Inference framework	vLLM
Batch size	256
Output format	Structured JSON

B Unit Parsing and Foil Generation Pipeline

This appendix describes the text-only LLM pipeline used to (i) parse captions into entity units and relation units, and (ii) generate matched foils via minimal edits. The same pipeline is used both for CS-CLIP training (Section 4.1) and for constructing the half-truth diagnostic (Section 3). The LLM operates on caption text only and never sees images. For evaluation, this pipeline generates candidates; subsequent image-grounded checks and manual verification are described in Appendix C.2.

Units and foils. Given a caption T, we extract two types of units:

- **Entity units** $\mathcal{E}(T)$: noun phrases describing visually grounded entities, including bound attributes and quantifiers (e.g., *"three dogs"*, *"a brown horse"*, *"a man in a blue shirt"*).
- **Relation units** $\mathcal{R}(T)$: directed relations between two entity units, written as (e_s, p, e_o) and rendered as short phrases (e.g., *"person riding horse"*, *"ball in park"*).

For each unit, we generate one or more **matched foils** $\tilde{U}$ that are minimally edited (changing exactly one component) and realistic in context, while changing the unit's meaning. Entity foils change either the object category or one attribute/quantifier. Relation foils change a single relation component (predicate or one argument), or swap arguments when the predicate is asymmetric.

B.1 Model and Inference Configuration

We use Qwen3-8B-AWQ served with vLLM to produce structured JSON outputs. Table 6 summarizes the configuration.

B.2 Pipeline Overview

For each caption T, the pipeline runs three stages:

1. **Extract units:** produce entity units $\mathcal{E}(T)$ and relation units $\mathcal{R}(T)$.
2. **Generate entity foils:** for each $e \in \mathcal{E}(T)$, generate minimally edited foils $\tilde{e}$.
3. **Generate relation foils:** for each $r \in \mathcal{R}(T)$, generate minimally edited foils $\tilde{r}$.

We apply lightweight rule-based filters to remove malformed outputs and avoid degenerate foils (e.g., near-synonyms or duplicates).

B.3 Entity and Relation Extraction

This stage parses a caption into entity units and relation units.

B.3.1 System Prompt (Extraction)

```
You extract units for vision-language retrieval.
ENTITY UNITS: - Output noun phrases for visually grounded entities.
- Keep bound attributes and quantifiers inside the noun phrase.
Examples: "three dogs", "a brown horse", "a man in a blue shirt".
- Do not output abstract concepts or non-visual phrases. - Do not
duplicate entities (avoid both "brown horse" and "horse").
```

```
RELATION UNITS: - Output directed relations as (subject, predicate,
object). - Subject and object must be chosen from the extracted
entity units. - Use short, visually checkable predicates (spatial,
action, possession/association). - Keep directionality: subject does
predicate to object.
Return ONLY valid JSON: { "entities": ["..."], "relations": [
{"subject": "...", "predicate": "...", "object": "..."} ] }
```

B.3.2 User Prompt Template (Extraction)

```
Extract entity units and relation units from this caption:
Caption: "{caption}"
Return only valid JSON.
```

B.3.3 Extraction constraints

We apply the following constraints:

- **Attribute binding:** keep modifiers attached to the noun phrase head (e.g., *"three dogs"*, not *"dogs"*).
- **Groundedness:** filter generic placeholders (e.g., *"thing"*) and non-visual phrases.
- **Relation grounding:** require relation arguments to match extracted entity strings exactly.

B.4 Matched Foils for Entity Units

For each entity unit, we generate foils that change exactly one component while staying realistic.

B.4.1 System Prompt (Entity Foils)

```
Generate matched foils for an entity noun phrase using minimal
edits.
Allowed edits (choose one per foil): 1) Object change:
replace the head noun with a similar-category object, keep
attributes/quantifiers. Example: "three dogs" -> "three cats" 2)
Attribute/quantifier change: change exactly one attribute or the
quantity, keep the head noun. Example: "a brown horse" -> "a white
horse"
Requirements: - Generate 2-4 foils. - Each foil must differ by
exactly one edit. - Avoid synonyms (e.g., couch->sofa). - Keep a
similar specificity level.
Return ONLY valid JSON: { "foils": [ {"text": "...", "edit_type":
"object_change"}, ... ] }
```

B.4.2 User Prompt Template (Entity Foils)

```
Generate matched foils for this entity unit.
Entity unit: "{entity_unit}" Caption context: "{caption}"
Return only valid JSON.
```

B.5 Matched Foils for Relation Units

For each relation unit (e_s, p, e_o), we generate foils that change one component while keeping arguments grounded.

B.5.1 System Prompt (Relation Foils)

```
Generate matched foils for a directed visual relation (subject,
predicate, object).
Allowed edits (one per foil): 1) Predicate change: replace the
predicate with a different plausible predicate. 2) Argument swap:
swap subject and object ONLY if the predicate is asymmetric. 3)
```

```
Argument change: replace ONE argument with a similar-category
entity.
Requirements: - Generate 1-3 foils. - Keep subject/object strings
grounded (use entities from the caption when possible). - Avoid
symmetric swaps (e.g., "next to", "near", "with"). - Keep the text
fluent and minimal-edit.
Return ONLY valid JSON: { "foils": [ {"subject":"...",
"predicate":"...", "object":"..."
,"edit_type": "predicate_change"}, ... ] }
```

B.5.2 User Prompt Template (Relation Foils)

```
Generate matched foils for this relation.
Relation: <{subject}, {predicate}, {object}> Caption context:
"{caption}"
Return only valid JSON.
```

B.6 Post-processing and Filters

We apply lightweight filters to remove invalid outputs:

- **Duplicate removal:** drop foils identical to the original unit or duplicated within the same caption.
- **Near-synonym filter:** drop obvious paraphrases that preserve meaning.
- **Minimal-edit check:** prefer single-component edits; reject multi-edit strings when detected.
- **Swap validity:** block swaps for symmetric predicates (e.g., *next to*, *near*, *with*).

B.7 Usage in Training and Evaluation

CS-CLIP training. For each image-caption pair (I, T), we sample one unit U from $\mathcal{E}(T) \cup \mathcal{R}(T)$ and one matched foil $\tilde{U}$ produced by this pipeline (Section 4.1).

Half-truth diagnostic. Given an anchor entity unit $A \in \mathcal{E}(T)$, we form a half-truth A^- by appending exactly one foil from a different unit (Section 3).

C Half-Truth Diagnostic: Construction and Additional Results

C.1 Construction and Notation

This appendix details how we construct the texts used in the half-truth diagnostic (Section 3). The construction follows a simple refinement template: start from a caption-supported *anchor* entity description and add *exactly one* additional piece of information.

Parsed caption structure. For each evaluation pair (I, T), we run our text-only parsing pipeline on caption T (Appendix B) and obtain: (i) a set of **entity units** $\mathcal{E}(T)$ (noun phrases with bound attributes/quantifiers, e.g., *"red car"*, *"three dogs"*), and (ii) a set of **relation units** $\mathcal{R}(T)$ as triplets $r = \langle e_s, p, e_o \rangle$ where $e_s, e_o \in \mathcal{E}(T)$ and p is a visually grounded predicate. For each unit we also construct matched minimal foils: $\tilde{\mathcal{E}}(e)$ for entity units and $\tilde{\mathcal{R}}(r)$ for relation units.

Text serialization. All comparisons are defined over text strings. We use a single serialization operator $\mathrm{txt}(\cdot)$:

- **Entity:** $\mathrm{txt}(e)$ is the surface string of entity unit e.
- **Entity conjunction:** $\mathrm{txt}(e_a, e_b) = \mathrm{txt}(e_a)$ `and` $\mathrm{txt}(e_b)$.
- **Relation:** for $r = \langle e_s, p, e_o \rangle$, $\mathrm{txt}(r) = \mathrm{txt}(e_s)\ p\ \mathrm{txt}(e_o)$.

Anchor (short description). We sample an anchor entity unit $e_a \in \mathcal{E}(T)$ and define the anchor text

$$A \;=\; \mathrm{txt}(e_a).$$

The anchor is *caption-supported* in the sense that it is extracted from T.

Truthful vs. half-truth additions. We consider two ways to refine the anchor by adding one unit:

- A **truthful addition** uses a caption-consistent unit (another entity unit or a relation unit involving the anchor) extracted from the same caption.
- A **half-truth** uses the same refinement template, but replaces the added unit with a **matched foil** produced by a minimal edit.

By construction, the anchor and its truthful/half-truth refinements share most tokens, isolating whether the model verifies the *added unit*.

C.1.1 Entity addition: adding one more entity

Truthful entity addition and half-truth. We sample a second entity unit $e^+ \in \mathcal{E}(T) \setminus \{e_a\}$ and form the truthful refinement

$$A_e^+ \;=\; \mathrm{txt}(e_a, e^+) \;=\; \mathrm{txt}(e_a)\ \texttt{and}\ \mathrm{txt}(e^+).$$

We then keep the anchor fixed and replace only the added entity unit with a matched foil:

$$A_e^- \;=\; \mathrm{txt}(e_a, e^-), \qquad e^- \in \tilde{\mathcal{E}}(e^+).$$

Entity foil types (+Obj, +Attr). In the verified evaluation set, we group entity half-truths by how the added entity unit is corrupted:

- **+Obj (wrong object):** replace the head noun of e^+ while keeping modifiers fluent (e.g., *"red car"*→*"red truck"*).
- **+Attr (wrong attribute):** replace an attribute/modifier while keeping the head noun (e.g., *"red car"*→*"blue car"*).

C.1.2 Relation addition: adding one relation involving the anchor

Truthful relation addition and half-truth. We sample a relation unit $r^+ \in \mathcal{R}(T)$ in which the anchor entity participates (as subject or object). We form the truthful refinement by appending this relation to the anchor:

$$A_r^+ \;=\; \mathrm{txt}(e_a)\ \texttt{and}\ \mathrm{txt}(r^+).$$

Table 7: Manually verified evaluation comparisons by source and foil type.

Source	Object	Attribute	Predicate	Swap	Total
COCO (Qwen)	195	59	167	88	509
COCO (Mistral)	177	75	159	43	454
CC3M (Qwen)	173	78	157	66	474

We then form a half-truth by substituting a matched foil relation:

$$A_r^- \;=\; \mathrm{txt}(e_a)\ \texttt{and}\ \mathrm{txt}(r^-), \qquad r^- \in \tilde{\mathcal{R}}(r^+).$$

Relation foil types (Ant, Swap). In the verified evaluation set, we group relation half-truths by whether the predicate or the argument order is perturbed:

- **Ant (predicate change):** replace p with an opposing relation (e.g., *on*→*under*, *in front of*→*behind*).
- **Swap (role swap):** swap subject and object ($\langle e_s, p, e_o\rangle \to \langle e_o, p, e_s\rangle$), preserving the predicate but changing directionality.

Why edits are minimal. All half-truth variants are generated by a single localized perturbation so the anchor and its refinements share most tokens. This keeps the comparison focused on verifying the added unit rather than topic drift. Figure 7 provides additional qualitative examples comparing CLIP, NegCLIP, and CS-CLIP similarity scores across multiple half-truth examples.

C.2 Evaluation Sources and Verification

The evaluation covers two foil generators and two image domains. Entity foils change an object or attribute; relation foils change a predicate or swap its arguments. We first check candidate foils against their images with Qwen3-VL-30B. We then use Claude Opus 5 to review the candidates and propose replacements within the same foil category. Finally, we manually inspect the candidates, including proposed replacements, and retain additions judged incorrect for the image. Candidates judged correct or ambiguous are discarded.

Training-foil quality. Training foils are automatically generated and filtered, without manual verification at scale. An image-grounded Qwen3-VL-30B audit of 1,000 foils flags 246 as accidentally true (24.6%): 132/500 entity foils (26.4%) and 114/500 relation foils (22.8%). This is a model-based estimate. On 816 independently human-labeled verification decisions, the verifier agrees with annotators on 80.3% of entity cases and 55.8% of relation cases. These errors motivate manual evaluation checks; the training results show gains despite estimated foil noise, rather than establishing that false negatives have no effect.

C.3 Correct Ordering Rate

To test whether a model distinguishes correct additions from incorrect ones, we measure Correct Ordering Rate (COR), $\Pr[s(I, A^+) > s(I, A) > s(I, A^-)]$, where A^+ is a correct refinement of the anchor. All three descriptions refer to the same image.

CS-CLIP achieves the highest COR among the displayed models: 58.2% on COCO (Qwen), 50.4% on COCO (Mistral), and 42.2% on CC3M (Qwen). On COCO (Qwen), its relational COR is 60.0%, compared to 11.8% for CLIP and 23.9% for NegCLIP. Requiring both inequalities prevents a uniform preference for shorter captions from satisfying the metric.



Figure 7: **Additional half-truth examples with model scores.** Multiple anchors and their half-truth refinements with similarity scores from CLIP, NegCLIP, and CS-CLIP. Some baselines assign comparable or higher similarity to incorrect refinements, while CS-CLIP more consistently lowers similarity for half-truths.

Table 8: **Correct Ordering Rate across three evaluation sources.** COR measures $\Pr[s(I, A^+) > s(I, A) > s(I, A^-)]$. We report percentages on 509 COCO (Qwen), 454 COCO (Mistral), and 474 CC3M (Qwen) manually verified comparisons. All is the sample-weighted entity/relation average. FSC-CLIP is trained on CC3M; the other fine-tuned models use COCO. Higher is better.

	COCO (Qwen)			COCO (Mistral)			CC3M (Qwen)		
Model	All	Entity	Relation	All	Entity	Relation	All	Entity	Relation
CLIP	18.9	26.0	11.8	19.2	26.6	9.9	24.5	34.7	13.0
NegCLIP	32.2	40.6	23.9	25.8	36.1	12.9	29.1	41.4	15.2
DeGLA	32.4	40.6	24.3	28.2	38.1	15.8	31.4	41.8	19.7
ReadCLIP	35.0	38.2	31.8	31.7	38.5	23.3	30.2	35.1	24.7
FSC-CLIP (CC3M)	24.2	36.2	12.2	25.6	35.3	13.4	26.8	39.8	12.1
CS-CLIP	**58.2**	**56.3**	**60.0**	**50.4**	**50.0**	**51.0**	**42.2**	**43.8**	**40.4**

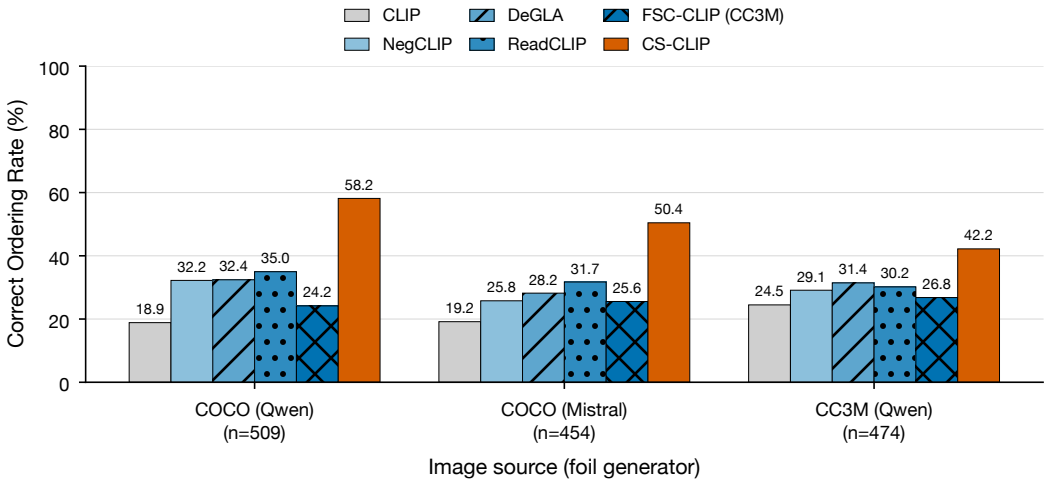

Figure 8: **Correct ordering across evaluation sources.** COR on the same comparisons as Table 1. Each bar reports one checkpoint evaluation; uncertainty is not shown.

Table 9: **Compositional benchmark suite.** Text: foil captions provided; Img: foil images provided.

Benchmark	Cite	Text	Img	Primary phenomenon
ARO	Yuksekgonul et al. (2023)	✓		Binding, relations, word order
BLA	Chen et al. (2023)	✓		Syntax and word order
COCO-CF	Le et al. (2023)	✓	✓	Counterfactual edits
COLA	Ray et al. (2023)	✓	✓	Multi-object binding
ColorFoil	Samin et al. (2025)	✓		Color attributes
ColorSwap	Burapacheep et al. (2024)	✓		Attribute binding
MMVP	Tong et al. (2024)	✓	✓	Visual attributes, spatial
NegBench	Alhamoud et al. (2025)	✓		Negation and absence
SPEC	Peng et al. (2024)	✓	✓	Synthetic controlled scenes
SugarCrepe	Hsieh et al. (2023)	✓		Fluent minimal edits
SugarCrepe++	Dumpala et al. (2024)	✓		Paraphrase invariance
VALSE	Parcalabescu et al. (2022)	✓		Existence, counting, coreference
VisMin	Awal et al. (2024)	✓	✓	Minimal visual changes
What'sUp	Kamath et al. (2023)	✓	✓	Spatial relations
Winoground	Thrush et al. (2022)	✓	✓	Role sensitivity
VL-CheckList	Zhao et al. (2022)	✓		Objects, attributes, relations

D Compositional Benchmark Suite and Capability Taxonomy

We evaluate compositional robustness using a heterogeneous suite of 16 benchmarks that test entity content, relational structure, attribute binding, and linguistic phenomena through controlled perturbations. To enable fine-grained analysis beyond aggregate scores, we organize benchmark subsets into a shared capability taxonomy with four top-level categories and ten sub-capabilities.

D.1 Benchmark Suite

Table 9 summarizes our compositional benchmark suite. Most benchmarks provide *foil captions* that preserve topical content while changing a targeted semantic factor (e.g., swapping attributes, changing predicates, or altering word order). Several benchmarks additionally provide *foil images* for stricter cross-modal consistency checks.

D.2 Capability Taxonomy

To analyze which compositional skills are improved by different methods, we categorize benchmark subsets into a shared taxonomy. For each subset, we assign a top-level **category** and a **sub-capability** based on the minimal semantic change needed to distinguish correct captions from foils. This enables macro-averaged reporting at the category level and fine-grained analysis at the sub-capability level.

Top-level categories. We define four categories (Table 10):

- **Entity Content:** Recognizing objects, attributes, quantities, and existence.
- **Relational Structure:** Understanding predicates and role assignments in relations.
- **Binding:** Associating attributes with the correct objects.
- **Linguistic:** Handling syntax, coreference, and negation.

Category scores are computed by macro-averaging subset accuracies within each category.

Sub-capabilities. We define ten sub-capabilities for fine-grained analysis:

- **Object Recognition:** Correct object category (e.g., *dog* vs. *cat*).
- **Attribute Recognition:** Correct attribute (e.g., *red* vs. *blue*).
- **Existence:** Whether an object is present.
- **Counting:** Correct number of objects (e.g., *four* vs. *three horses*).
- **Predicate Sensitivity:** Correct relation predicate (e.g., *left of* vs. *right of*).
- **Role Sensitivity:** Which entity plays which role (e.g., *dog chasing cat* vs. *cat chasing dog*).

Table 10: **Capability taxonomy summary.**

Category	Sub-capabilities
Entity Content	Object Recognition, Attribute Recognition, Counting, Existence
Relational Structure	Predicate Sensitivity, Role Sensitivity
Binding	Attribute Binding
Linguistic	Syntax, Coreference, Negation

- **Attribute Binding:** Which attribute binds to which object (e.g., *red car and blue cat* vs. *blue car and red cat*).
- **Syntax:** Word order changes meaning.
- **Coreference:** Resolving references across phrases.
- **Negation:** Understanding negation (e.g., *has A but not B*).

D.3 Benchmark Subset Mapping

Table 11 provides the complete mapping from benchmark subsets to our taxonomy, organized by category.

Table 11: **Benchmark subset to capability mapping.**

Benchmark	Subset	Category	Sub-capability
Entity Content			
SugarCrepe	add_att	Entity Content	Attribute Recognition
SugarCrepe	replace_att	Entity Content	Attribute Recognition
SugarCrepe	add_obj	Entity Content	Object Recognition
SugarCrepe	replace_obj	Entity Content	Object Recognition
SugarCrepe++	replace_attribute	Entity Content	Attribute Recognition
SugarCrepe++	replace_object	Entity Content	Object Recognition
VisMin	object	Entity Content	Object Recognition
VisMin	attribute	Entity Content	Attribute Recognition
VisMin	counting	Entity Content	Counting
SPEC	count	Entity Content	Counting
SPEC	absolute_size	Entity Content	Attribute Recognition
SPEC	existence	Entity Content	Existence
VL-CheckList	attr_color	Entity Content	Attribute Recognition
VL-CheckList	attr_material	Entity Content	Attribute Recognition
VL-CheckList	attr_size	Entity Content	Attribute Recognition
VL-CheckList	attr_state	Entity Content	Attribute Recognition
VL-CheckList	attr_action	Entity Content	Attribute Recognition
VL-CheckList	obj_location	Entity Content	Object Recognition
VL-CheckList	obj_size	Entity Content	Object Recognition
VALSE	existence	Entity Content	Existence
VALSE	counting	Entity Content	Counting
VALSE	plurals	Entity Content	Counting
ColorFoil		Entity Content	Attribute Recognition
COCO-CF		Entity Content	Object Recognition
MMVP	Color	Entity Content	Attribute Recognition
MMVP	Presence	Entity Content	Existence
MMVP	Quantity	Entity Content	Counting
MMVP	Structural Char.	Entity Content	Attribute Recognition
Relational Structure			
ARO	VG_Relation	Relational Structure	Predicate Sensitivity
SugarCrepe	replace_rel	Relational Structure	Predicate Sensitivity

(continued)

Benchmark	Subset	Category	Sub-capability
SugarCrepe	swap_obj	Relational Structure	Predicate Sensitivity
SugarCrepe++	replace_relation	Relational Structure	Predicate Sensitivity
SugarCrepe++	swap_object	Relational Structure	Predicate Sensitivity
VisMin	relation	Relational Structure	Predicate Sensitivity
SPEC	relative_spatial	Relational Structure	Role Sensitivity
SPEC	absolute_position	Relational Structure	Role Sensitivity
SPEC	relative_size	Relational Structure	Role Sensitivity
What'sUp	A	Relational Structure	Role Sensitivity
What'sUp	B	Relational Structure	Role Sensitivity
What'sUp	VG-One	Relational Structure	Role Sensitivity
What'sUp	VG-Two	Relational Structure	Role Sensitivity
What'sUp	COCO-One	Relational Structure	Role Sensitivity
What'sUp	COCO-Two	Relational Structure	Role Sensitivity
VL-CheckList	rel_action	Relational Structure	Role Sensitivity
VL-CheckList	rel_spatial	Relational Structure	Role Sensitivity
VALSE	relations	Relational Structure	Predicate Sensitivity
VALSE	actions	Relational Structure	Role Sensitivity
MMVP	Orientation	Relational Structure	Role Sensitivity
MMVP	Spatial	Relational Structure	Role Sensitivity
MMVP	State	Relational Structure	Role Sensitivity
Winoground		Relational Structure	Role Sensitivity
Binding			
ARO	VG_Attribution	Binding	Attribute Binding
SugarCrepe	swap_att	Binding	Attribute Binding
SugarCrepe++	swap_attribute	Binding	Attribute Binding
ColorSwap		Binding	Attribute Binding
COLA	multi_object	Binding	Attribute Binding
Linguistic			
ARO	COCO_Order	Linguistic	Syntax
ARO	Flickr30k_Order	Linguistic	Syntax
VALSE	coreference	Linguistic	Coreference
VALSE	noun phrases	Linguistic	Syntax
BLA	ap	Linguistic	Syntax
BLA	co	Linguistic	Syntax
BLA	rc	Linguistic	Syntax
NegBench	MSRVTT	Linguistic	Negation
NegBench	VOC2007	Linguistic	Negation
NegBench	COCO	Linguistic	Negation

E Compositional Benchmark Suite: Additional Analyses

This appendix provides detailed breakdowns for the compositional evaluation in Section 5.2. In the main paper, we report aggregate suite-level performance showing CS-CLIP achieves 57.8% average I2T accuracy across 16 benchmarks (Figure 6). Here we provide: (i) per-benchmark results (Table 12), (ii) per-subset visualizations (Figures 9–10), (iii) per-capability breakdowns (Table 13), and (iv) downstream retrieval results and the zero-shot classification trade-off (Section E.6).

All analyses use the same models and evaluation protocol as in the main paper. The capability taxonomy and benchmark subset mapping are provided in Appendix D.

E.1 Evaluation Protocol and Metrics

Compositional benchmarks present *contrast sets* where one item is correct and one or more *foils* differ by a minimal semantic change (e.g., swapping roles, changing an attribute, or altering word order). Let $s(I, T)$ denote the model's image–text similarity score.

Image-to-Text (I2T) accuracy. For each image I_i, the benchmark provides a candidate caption set $\mathcal{T}_i = \{T_{i,1}, \ldots, T_{i,m_i}\}$ with a single correct caption $T_i^\star \in \mathcal{T}_i$. I2T accuracy is the fraction of cases where the correct caption ranks above all foils:

$$\mathrm{Acc}_{\mathrm{I2T}} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}\Big[s(I_i, T_i^\star) \; > \; \max_{T \in \mathcal{T}_i \setminus \{T_i^\star\}} s(I_i, T) \Big]. \tag{9}$$

When $\mathcal{T}_i$ contains exactly one foil, this reduces to a pairwise win-rate.

Text-to-Image (T2I) accuracy. For each text query T_i, the benchmark provides a candidate image set $\mathcal{I}_i = \{I_{i,1}, \ldots, I_{i,n_i}\}$ with a single correct image $I_i^\star \in \mathcal{I}_i$. T2I accuracy is the fraction of cases where the correct image ranks above all foils:

$$\mathrm{Acc}_{\mathrm{T2I}} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}\Big[s(I_i^\star, T_i) \; > \; \max_{I \in \mathcal{I}_i \setminus \{I_i^\star\}} s(I, T_i) \Big]. \tag{10}$$

Group accuracy. Some benchmarks provide groups consisting of n matched image-caption pairs $\{(I_{i,1}, T_{i,1}), \ldots, (I_{i,n}, T_{i,n})\}$. Group Accuracy is the fraction of groups where the model assigns the highest similarity to the *matched* pair over all mismatched pairs, in both directions:

$$\mathrm{Acc}_{\mathrm{Grp}} = \frac{1}{N_g} \sum_{i=1}^{N_g} \mathbb{I}\Big[\forall j \in \{1, \ldots, n\} : \; s(I_{i,j}, T_{i,j}) > \max\Big(\max_{k \neq j} s(I_{i,k}, T_{i,j}), \max_{\ell \neq j} s(I_{i,j}, T_{i,\ell}) \Big) \Big]. \tag{11}$$

Averaging. We report per-benchmark accuracies and suite-level averages computed as the unweighted mean of benchmark-level accuracies. For capability-wise analysis, we compute accuracies for each benchmark subset, then macro-average within each capability category (Appendix D.2). CS-CLIP point estimates use the best-performing run; per-benchmark $\pm$ values denote standard deviations across five runs. All other models are reported with their single published checkpoint.

E.2 Per-Benchmark Results

Analysis. Table 12 reveals several patterns. First, fine-tuning on COCO consistently improves compositional robustness: the top-performing models (CS-CLIP, FSC-CLIP, ReadCLIP, DeGLA) all fine-tune on COCO and achieve 56–58% average accuracy, compared to 52% for zero-shot CLIP. Several compositional benchmarks draw images from COCO or Visual Genome, including ARO, What'sUp, VALSE, VL-CheckList, and SugarCrepe. The benchmarks use independently constructed evaluation tasks, but their image sources are not all independent of the training corpus. This shared provenance should be considered when interpreting the gains; Figure 6 alone does not establish whether they reflect source-specific adaptation or broader compositional improvements.

Second, CS-CLIP achieves the highest overall average (57.8%) while excelling on benchmarks that require tight binding (ARO 86.9%, VL-CheckList 79.2%) and spatial reasoning (What'sUp 43.5%, SPEC 36.1%). Third, performance varies substantially across benchmarks: models generally perform well on color attributes (ColorFoil 84–93%) but struggle on spatial relations (What'sUp 39–44%, MMVP 4–16%) and group-level consistency (Winoground 20–33%). Finally, training on larger datasets (CC3M, LAION) does not consistently improve compositional robustness compared to COCO fine-tuning, suggesting dataset quality and alignment signal matter more than scale alone.

Table 12: **Compositional benchmark results (I2T accuracy, %).** Each benchmark averaged over all subsets. Rows grouped by training dataset and sorted by overall average within each group. SC = SugarCrepe, SC++ = Sugar++, VLC = VL-Checklist. **Bold**: best; underline: second best.

Model	ARO	BLA	COCO-CF	COLA	ColorFoil	ColorSwap	WhatsUp	MMVP	NegBench	SPEC	SC	SC++	VALSE	VLC	VisMin	Winoground	Avg
Pre-training variants (zero-shot)																	
CLIP	48.5	49.4	74.0	41.9	84.2	60.7	41.4	13.3	30.6	32.7	76.8	59.7	65.4	67.1	57.9	29.8	52.1
TripletCLIP	37.9	49.7	47.6	24.3	75.5	55.7	41.6	10.4	27.3	30.5	74.0	55.4	59.8	66.2	35.8	22.2	44.6
LaCLIP	39.4	50.2	41.0	21.9	69.1	56.7	41.8	**15.6**	25.5	29.2	63.5	46.0	57.3	67.6	33.7	26.2	42.8
Fine-tuned on COCO (direct comparison)																	
CS-CLIP (Ours)	86.9$_{\pm 0.2}$	50.2$_{\pm 0.4}$	78.2$_{\pm 0.2}$	41.0$_{\pm 2.2}$	90.5$_{\pm 0.8}$	59.0$_{\pm 1.2}$	43.5$_{\pm 0.6}$	13.3$_{\pm 1.9}$	32.8$_{\pm 1.2}$	36.1$_{\pm 0.3}$	82.2$_{\pm 0.5}$	67.4$_{\pm 0.3}$	74.8$_{\pm 0.3}$	**79.2**$_{\pm 0.2}$	60.5$_{\pm 0.2}$	29.8$_{\pm 1.0}$	**57.8**
FSC-CLIP	86.0	49.3	77.5	**44.8**	90.8	56.0	41.2	12.6	26.0	35.3	85.2	67.5	75.7	76.8	**61.3**	**32.8**	57.4
ReadCLIP	**90.0**	50.0	73.9	31.4	91.9	**62.0**	**44.0**	11.9	27.5	34.6	87.0	69.8	76.6	75.0	60.5	24.2	56.9
DeGLA	86.9	50.3	75.7	32.4	91.6	57.7	42.3	4.4	30.7	**36.2**	**89.2**	62.8	**77.3**	78.1	59.0	24.5	56.2
NegCLIP	82.4	49.8	76.4	32.9	87.5	55.3	42.6	10.4	28.6	35.6	82.5	62.7	74.7	74.0	59.6	30.5	55.3
LabCLIP	78.9	49.5	75.8	32.9	88.0	57.7	41.9	11.1	31.2	33.5	79.9	61.0	72.0	73.9	54.8	24.5	54.2
CE-CLIP	76.8	50.1	69.2	24.8	**93.0**	60.3	41.8	8.9	35.7	34.3	85.9	55.6	76.8	77.7	59.4	19.8	54.4
DAC (LLM)	64.9	49.0	72.2	35.7	84.7	56.0	42.6	11.1	35.9	33.0	75.6	59.0	66.7	71.8	51.5	25.5	52.2
DAC (SAM)	62.2	48.8	72.8	34.3	84.8	54.7	42.8	8.9	**37.2**	32.2	74.4	59.5	65.8	71.4	51.2	26.5	51.7
Fine-tuned on larger/other datasets (CC3M, LAION, RedCaps)																	
FSC-CLIP (CC3M)	80.4	49.1	73.7	37.1	91.1	56.3	42.8	8.9	27.0	35.7	85.2	59.7	75.1	78.0	59.0	29.2	55.5
CLoVe	81.7	**52.1**	73.3	35.7	88.3	58.0	41.1	13.3	25.2	32.0	84.1	51.1	72.4	72.9	53.5	22.5	53.6
FSC-CLIP (L+C)	83.9	50.3	75.8	30.0	89.8	55.0	42.8	9.6	29.1	34.3	83.4	59.9	74.0	76.1	57.8	27.8	55.0
CLIC (RedCaps)	66.8	49.4	77.4	36.2	84.9	60.0	39.9	10.4	32.4	32.6	81.8	**70.3**	69.9	74.0	54.2	32.2	54.5
CLIC (LAION)	67.8	49.5	78.0	34.3	84.6	58.3	40.0	10.4	30.3	33.0	81.2	69.9	70.0	75.3	55.4	31.5	54.3
CON-CLIP	41.2	49.0	**78.8**	36.2	83.0	56.3	41.2	8.9	34.8	33.5	76.5	60.6	66.8	68.0	54.9	29.0	51.2
TSVLC	61.1	48.2	73.2	33.3	83.9	54.3	40.6	11.9	36.5	32.7	72.4	60.5	66.9	71.4	51.8	26.8	51.6

Table 13: **Performance by sub-capability (I2T accuracy, %).** Scores macro-averaged within each sub-capability. **Bold**: best; underline: second best.

Model	Attr. Bind.	Attr. Recog.	Coref.	Count	Exist.	Negation	Obj. Recog.	Pred. Sens.	Role Sens.	Syntax	Avg
Pre-training variants (zero-shot)											
CLIP	39.3	53.4	51.4	34.3	24.1	30.6	65.0	42.4	34.6	40.9	41.6
TripletCLIP (CC12M)	35.5	48.4	50.9	26.9	24.1	27.3	56.6	42.7	33.9	35.3	38.2
LaCLIP (CC12M)	37.7	47.5	52.2	29.6	29.0	25.5	55.3	41.1	35.3	38.1	39.1
Fine-tuned on COCO (direct comparison)											
CS-CLIP	43.8$_{\pm 1.0}$	58.1$_{\pm 0.5}$	56.2$_{\pm 1.7}$	37.7$_{\pm 0.6}$	30.7$_{\pm 1.2}$	32.8$_{\pm 1.2}$	71.3$_{\pm 0.7}$	51.0$_{\pm 0.3}$	**40.1**$_{\pm 0.6}$	54.3$_{\pm 0.6}$	47.6$_{\pm 0.8}$
ReadCLIP	**46.7**	58.4	55.7	**39.0**	**31.4**	27.5	**72.0**	**54.1**	38.8	**55.5**	**47.9**
FSC-CLIP (COCO)	45.3	**58.8**	60.2	**39.0**	30.0	26.0	71.1	52.8	37.4	54.3	47.5
DeGLA	40.7	56.8	59.5	35.9	27.7	30.7	68.4	50.1	35.5	51.4	45.7
NegCLIP	42.4	56.0	59.7	38.4	28.8	28.6	69.4	50.3	37.1	52.5	46.3
LabCLIP	40.0	54.6	54.3	32.9	28.4	31.2	67.1	46.8	35.9	49.1	44.0
CE-CLIP	37.8	56.2	**60.6**	37.0	30.1	35.7	66.2	46.4	34.8	46.0	45.1
DAC (LLM)	38.6	52.9	54.6	31.3	26.5	35.9	65.5	44.8	36.2	45.9	43.2
DAC (SAM)	39.3	52.4	53.3	31.6	28.7	**37.2**	66.0	45.1	35.2	45.8	43.5
Fine-tuned on larger/other datasets (CC3M, LAION, RedCaps)											
FSC-CLIP (CC3M)	40.3	57.1	55.7	34.9	27.5	27.0	67.1	48.9	37.4	49.5	44.5
CLoVe	39.8	52.3	49.1	34.6	30.0	25.2	64.7	51.1	37.0	50.7	43.5
FSC-CLIP (L+C)	39.5	55.0	52.7	34.8	27.5	29.1	67.0	48.1	38.0	51.1	44.3
CLIC (RedCaps)	42.6	56.6	47.7	34.4	25.7	32.4	69.0	50.1	35.9	48.4	44.3
CLIC (LAION)	40.4	56.5	51.7	32.8	25.1	30.3	67.7	48.3	35.6	47.1	43.6
CON-CLIP	38.1	52.2	45.8	33.7	26.7	34.8	66.9	42.4	33.0	38.1	41.2
TSVLC	38.4	52.4	53.9	32.5	30.1	36.5	65.1	45.2	34.5	44.6	43.3

E.3 Per-Subset Visualization

Figures 9 and 10 visualize I2T accuracy differences relative to CLIP across all benchmark subsets, providing a comprehensive view of each model's strengths and weaknesses beyond aggregate scores.

E.4 Per-Capability Results

Analysis. Table 13 shows capability-level performance patterns. CS-CLIP achieves the best Role Sensitivity (40.1%), substantially outperforming CLIP (34.6%) and matching or exceeding all other COCO-trained methods. This aligns with our approach: unit-level supervision explicitly targets relational structure by contrasting entities in different roles (e.g., "dog chasing cat" vs. "cat chasing dog"). CS-CLIP also ranks second on Object Recognition (71.3%), Existence (30.7%), and Syntax (54.3%), and is competitive on Attribute Binding (43.8%), demonstrating broad improvements.

The capability breakdown reveals systematic weaknesses across all models. Counting remains challenging (24–39%), likely because numerical information is poorly grounded in bag-of-visual-features representations. Negation understanding is similarly weak (26–37%), consistent with prior findings that CLIP-style models

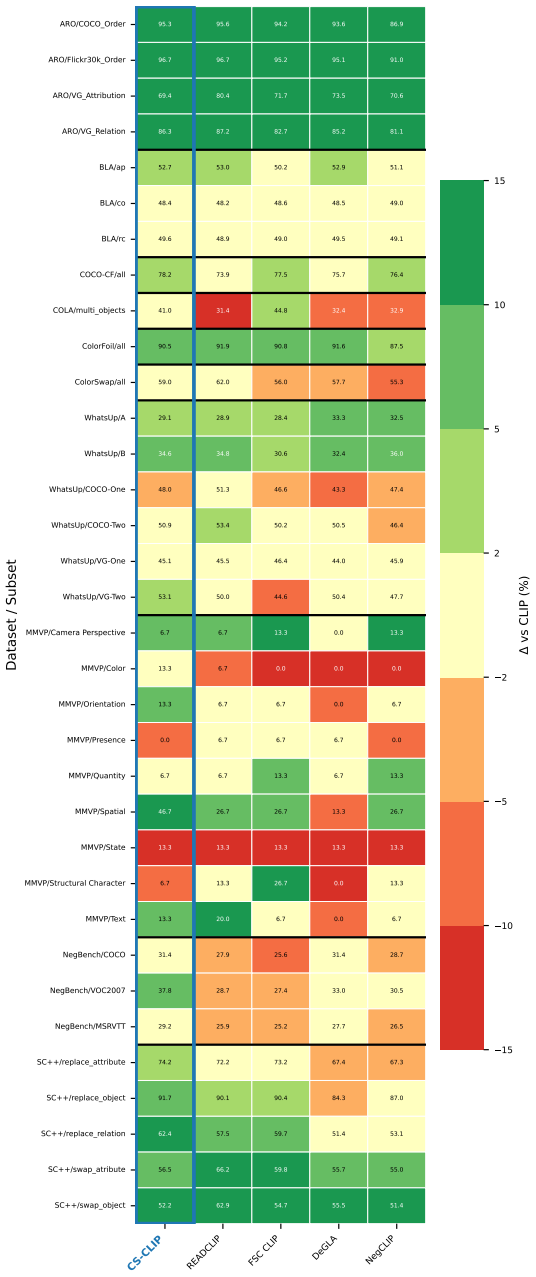

Figure 9: **Per-subset compositional I2T accuracy (Part 1/2).** Green: improvement over CLIP; red: degradation.

struggle with linguistic negation. Object Recognition shows the highest absolute performance (55–72%), suggesting entity-level grounding is more robust than relational or linguistic phenomena.

Comparing COCO-trained models to larger-dataset variants, we observe that COCO fine-tuning generally produces better compositional capabilities than training on CC3M or LAION. For example, CS-CLIP (COCO) achieves 47.6% average across capabilities, while FSC-CLIP (CC3M) achieves 44.5% despite being trained on 30× more data. This suggests compositional robustness depends more on alignment signal quality than raw dataset size.

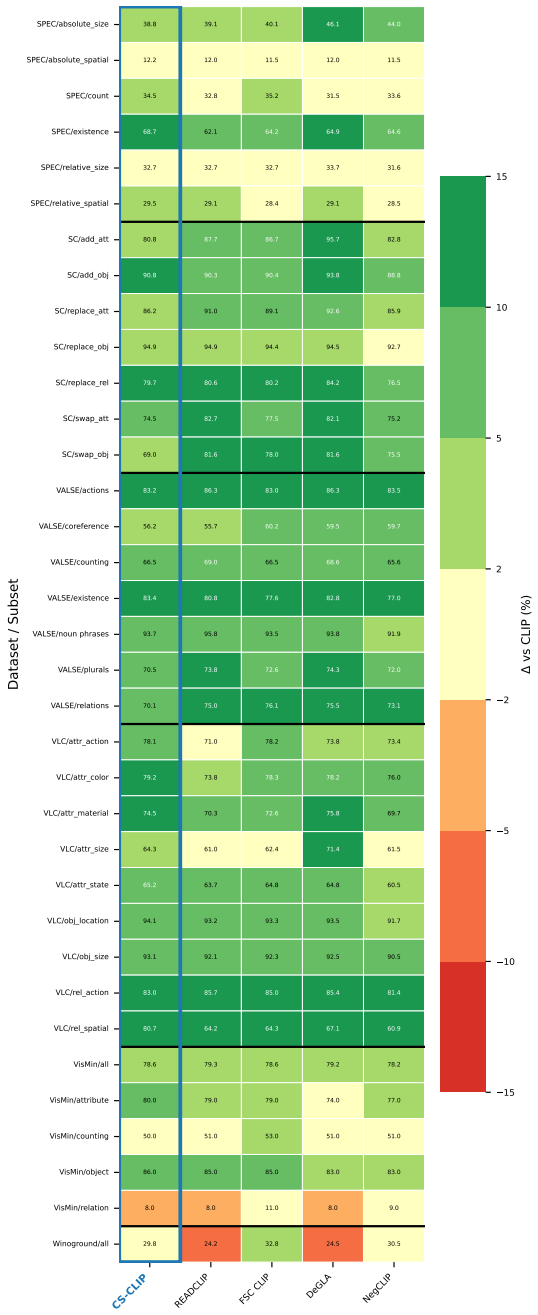

Figure 10: **Per-subset compositional I2T accuracy (Part 2/2).** Continuation of Figure 9.

E.5 Bidirectional Performance on Paired Datasets

Analysis. Table 14 reports bidirectional performance on seven paired compositional datasets that provide both image and text foils. CS-CLIP achieves the **best average Group Accuracy** (27.8%), indicating strong performance when both retrieval directions must succeed simultaneously. This confirms that unit-level supervision benefits both I2T and T2I matching rather than improving one direction at the expense of the other.

Among COCO-trained models, CS-CLIP and FSC-CLIP achieve comparable average I2T (45.4% vs. 45.8%) and T2I (39.8% vs. 40.0%) accuracy, with Group Accuracy of 27.8% and 27.7%, respectively. Notably, CS-CLIP achieves the best Winoground T2I score (13.0%) among all models, demonstrating improved role sensitivity in the T2I direction. Performance on MMVP remains challenging across all models (Group Accuracy below 6%),

Table 14: **Bidirectional performance on paired compositional datasets (accuracy, %).** I2T = Image-to-Text, T2I = Text-to-Image, Grp = Group accuracy (requires correctness in both directions). CSwap = ColorSwap, Wino. = Winoground. Checkpoints and within-dataset aggregation match the 16-benchmark summary; averages here give equal weight to the seven displayed datasets. **Bold**: best; underline: second best at the displayed precision.

	COCO-CF			COLA			CSwap			MMVP			SPEC			VisMin			Wino.			Average		
Model	I2T	T2I	Grp	I2T	T2I	Grp	I2T	T2I	Grp	I2T	T2I	Grp	I2T	T2I	Grp	I2T	T2I	Grp	I2T	T2I	Grp	I2T	T2I	Grp
Pre-training variants (zero-shot)																								
CLIP	74.0	73.1	60.4	41.9	22.4	14.8	60.7	**72.7**	**45.0**	13.3	9.6	**5.2**	32.7	31.8	14.2	57.9	50.1	38.0	29.8	10.8	7.8	44.3	38.6	26.5
TripletCLIP	47.6	45.8	31.1	24.3	26.2	11.4	55.7	64.0	38.0	10.4	9.6	3.7	30.5	28.8	13.1	35.8	32.6	15.2	22.2	6.2	3.8	32.3	30.5	16.6
LaCLIP	41.0	36.7	23.9	21.9	14.8	6.7	56.7	63.7	40.3	**15.6**	4.4	3.0	29.2	28.2	11.5	33.7	27.7	13.0	26.2	8.0	6.2	32.0	26.2	14.9
Fine-tuned on COCO (direct comparison)																								
CS-CLIP (Ours)	78.2	75.9	**65.1**	41.0	26.2	20.0	59.0	69.3	40.7	13.3	9.6	3.7	36.1	31.9	15.0	60.5	52.6	42.2	29.8	**13.0**	8.0	45.4	39.8	**27.8**
FSC-CLIP	77.5	75.5	64.5	**44.8**	**27.1**	**21.9**	56.0	69.0	38.3	12.6	7.4	0.7	35.3	33.0	14.6	**61.3**	**55.9**	**44.5**	**32.8**	12.0	9.5	**45.8**	**40.0**	27.7
ReadCLIP	73.9	75.6	62.7	31.4	25.7	15.2	**62.0**	67.3	43.3	11.9	4.4	2.2	34.6	32.3	13.2	60.5	55.2	44.2	24.2	10.2	6.0	42.6	38.7	26.7
DeGLA	75.7	74.6	62.7	32.4	20.0	15.7	57.7	71.0	41.7	4.4	7.4	0.7	**36.2**	31.9	**16.4**	59.0	53.1	40.0	24.5	10.5	7.5	41.4	38.4	26.4
NegCLIP	76.4	72.9	61.6	32.9	16.2	11.4	55.3	70.7	39.7	10.4	**10.4**	2.2	35.6	**33.4**	15.0	59.6	54.4	41.8	30.5	11.2	8.0	43.0	38.5	25.7
LabCLIP	75.8	72.4	61.1	32.9	26.2	16.7	57.7	62.7	37.0	11.1	8.1	2.2	33.5	30.6	12.9	54.8	47.6	35.2	24.5	10.2	6.8	41.5	36.8	24.6
CE-CLIP	69.2	74.6	58.8	24.8	23.3	11.0	60.3	66.3	42.7	8.9	5.9	1.5	34.3	31.8	13.7	59.4	54.7	42.8	19.8	12.0	5.2	39.5	38.4	25.1
DAC (LLM)	72.2	69.0	56.7	35.7	22.4	16.2	56.0	63.0	36.7	11.1	9.6	2.2	33.0	29.8	14.1	51.5	45.9	32.2	25.5	9.8	6.5	40.7	35.6	23.5
DAC (SAM)	72.8	68.4	56.5	34.3	20.5	14.8	54.7	65.7	36.0	8.9	8.1	0.7	32.2	29.8	13.6	51.2	45.4	32.2	26.5	10.2	6.5	40.1	35.4	22.9
Fine-tuned on larger/other datasets (CC3M, LAION, RedCaps)																								
FSC-CLIP (CC3M)	73.7	73.4	60.6	37.1	20.5	13.8	56.3	69.7	39.7	8.9	6.7	1.5	35.7	32.9	15.2	59.0	54.6	42.0	29.2	8.8	6.0	42.9	38.1	25.5
FSC-CLIP (L+C)	75.8	**76.0**	63.6	30.0	21.0	11.4	55.0	72.3	39.0	9.6	4.4	1.5	34.3	32.5	14.3	57.8	52.7	40.0	27.8	9.2	6.8	41.5	38.3	25.2
CLoVe	73.3	73.2	59.7	35.7	21.4	14.3	58.0	68.7	39.3	13.3	3.7	0.7	32.0	31.5	13.5	53.5	48.1	34.2	22.5	10.2	6.2	41.2	36.7	24.0
CLIC (RedCaps)	77.4	72.6	61.8	36.2	19.0	11.4	60.0	66.0	42.3	10.4	**10.4**	0.7	32.6	30.1	11.4	54.2	46.8	34.5	32.2	12.5	**10.8**	43.3	36.8	24.7
CLIC (LAION)	78.0	72.5	62.4	34.3	24.3	14.3	58.3	66.7	40.7	10.4	5.2	0.7	33.0	30.3	11.7	55.4	47.6	35.2	31.5	11.8	9.2	43.0	36.9	24.9
CON-CLIP	**78.8**	73.7	63.5	36.2	16.2	8.1	56.3	66.7	40.7	8.9	**10.4**	3.0	33.5	32.3	11.9	54.9	44.8	33.0	29.0	10.0	7.2	42.5	36.3	23.9
TSVLC	73.2	68.9	56.9	33.3	17.6	11.0	54.3	65.7	37.7	11.9	9.6	3.0	32.7	29.6	13.6	51.8	45.7	33.2	26.8	10.5	7.8	40.6	35.4	23.3

suggesting that fine-grained visual attribute distinctions require further advances beyond compositional training signals.

E.6 Downstream Zero-Shot Classification and Retrieval

We evaluate whether compositional improvements come at the cost of standard vision-language tasks. Table 15 reports zero-shot classification on ImageNet variants and fine-grained datasets, while Table 16 reports image–text retrieval on Flickr8k and MS-COCO.

Zero-shot classification. Given an image I and class prompts $\{T_c\}$, we compute similarities $s(I, T_c)$ and predict the class with highest score. We report Acc@1 and Acc@5 on seven datasets.

Image–text retrieval. Given an image (or text) query and a gallery of candidate texts (or images), we rank candidates by similarity and report Recall@1 in both directions: I2T (image-to-text) and T2I (text-to-image).

Analysis. CS-CLIP maintains competitive downstream performance while achieving the best compositional robustness. On zero-shot classification, CS-CLIP achieves 59.9% Acc@1 and 84.6% Acc@5, compared to CLIP's 63.6% and 86.5%. The 3.7-point Acc@1 gap represents a modest trade-off for substantial compositional gains (+5.8 points on the compositional suite, +42.4 points on COCO (Qwen) Half-Truth Accuracy). Notably, CS-CLIP achieves the best Caltech101 Acc@5 (97.9%), suggesting unit-level supervision does not uniformly harm recognition.

For retrieval, CS-CLIP achieves the best performance among all evaluated models: 56.8% I2T and 71.7% T2I, outperforming CLIP by +11.6 and +9.0 points respectively. This indicates that compositional improvements directly benefit retrieval tasks where fine-grained matching matters. The gains are particularly pronounced on Flickr8k (67.3% I2T, 81.5% T2I), where captions are more descriptive and compositional structure is more salient.

Comparing COCO-trained models, we observe that methods achieving strong compositional robustness (CS-CLIP, FSC-CLIP, ReadCLIP) also excel at retrieval, while zero-shot classification performance varies. This suggests retrieval and compositional evaluation measure related capabilities, both require fine-grained text-image alignment, while zero-shot classification depends more on coarse-grained category recognition.

Table 15: **Zero-shot classification (Acc@1 / Acc@5, %).** IN1k: ImageNet-1k Deng et al. (2009); INv2: ImageNetV2 Recht et al. (2019); Sketch: ImageNet-Sketch Wang et al. (2019); IN-O: ImageNet-O Hendrycks et al. (2021); Cal101: Caltech101 Fei-Fei et al. (2004); C10: CIFAR-10 Krizhevsky (2009); SUN: SUN397 Xiao et al. (2010). **Bold**: best; underline: second best.

Model	IN1k		INv2		Sketch		IN-O		Cal101		C10		SUN		Avg	
	@1	@5	@1	@5	@1	@5	@1	@5	@1	@5	@1	@5	@1	@5	@1	@5
Pre-training variants (zero-shot)																
CLIP	60.2	85.9	52.4	79.4	**46.4**	**72.6**	**50.7**	**82.3**	81.9	94.1	88.4	99.7	**65.0**	91.7	63.6	86.5
TripletCLIP (CC12M)	23.0	46.9	19.3	42.2	14.8	33.3	29.2	57.8	65.5	89.1	63.2	95.5	36.4	68.5	35.9	61.9
Fine-tuned on COCO (direct comparison)																
CS-CLIP	58.9	86.3	51.7	80.9	38.0	65.4	43.4	75.2	81.5	**97.9**	88.7	99.6	57.4	86.8	59.9	84.6
FSC-CLIP (COCO)	59.2	86.4	51.7	80.5	38.9	66.6	45.5	76.6	81.8	96.7	89.1	99.6	62.4	90.5	61.2	85.3
NegCLIP	55.8	83.8	48.9	78.0	35.9	62.6	42.4	72.7	81.2	96.8	85.9	99.5	57.7	87.3	58.2	83.0
DeGLA	56.3	84.5	50.0	79.0	36.8	64.0	44.4	75.2	80.9	95.8	86.0	99.5	60.3	88.9	59.3	83.9
ReadCLIP	51.5	80.8	45.3	75.0	32.8	59.8	44.4	74.7	78.3	92.9	87.2	99.6	56.5	86.1	56.6	81.3
CE-CLIP	49.9	80.5	43.3	74.3	31.5	58.6	44.9	74.8	78.4	94.6	85.9	98.7	56.6	87.5	55.8	81.3
DAC (LLM)	52.7	81.3	46.3	75.2	35.5	62.9	43.7	71.9	79.3	93.5	87.0	98.8	56.3	87.2	57.3	81.5
DAC (SAM)	52.9	81.4	46.8	75.5	35.3	62.6	44.0	72.4	78.8	93.0	86.7	98.9	55.1	86.8	57.1	81.5
Fine-tuned on larger/other datasets (CC3M, LAION, RedCaps)																
CON-CLIP	**63.2**	88.7	**55.5**	83.2	42.8	70.7	47.0	78.0	**82.5**	95.8	**90.7**	**99.8**	63.9	**92.1**	**63.7**	**86.9**
CLIC (RedCaps)	62.3	88.1	55.3	82.5	41.8	69.6	47.2	77.1	82.4	96.7	89.8	99.6	64.2	91.6	63.3	86.5
CLIC (LAION)	61.7	87.7	55.0	82.4	41.4	69.1	46.8	77.7	81.8	96.1	89.5	99.6	64.3	91.6	62.9	86.3
FSC-CLIP (L+C)	58.1	85.8	51.0	80.0	39.8	67.2	48.3	76.8	79.9	93.7	87.5	99.2	64.3	91.5	61.3	84.9
FSC-CLIP (CC3M)	54.9	83.8	47.7	78.4	37.6	65.0	46.0	74.9	80.3	94.2	89.5	99.5	62.7	90.5	59.8	83.8
TSVLC	54.3	82.4	47.6	76.6	36.1	63.5	44.5	73.7	79.4	93.5	87.0	99.0	57.0	88.0	58.0	82.4
CLoVe	52.9	81.8	46.6	76.2	35.5	62.7	41.6	72.6	78.6	89.9	88.4	99.7	60.2	88.7	57.7	81.6

Table 16: **Image–text retrieval (Recall@1, %).** I2T: image-to-text; T2I: text-to-image. **Bold**: best; underline: second best.

	Flickr8k		MS-COCO		Average	
Model	I2T	T2I	I2T	T2I	I2T	T2I
Pre-training variants						
CLIP	56.2	72.9	34.3	52.5	45.2	62.7
TripletCLIP (CC12M)	23.4	28.4	11.3	14.9	17.3	21.7
Fine-tuned on COCO (direct comparison)						
CS-CLIP	**67.3**	**81.5**	46.3	**61.9**	**56.8**	**71.7**
FSC-CLIP (COCO)	67.0	79.2	46.0	60.4	56.5	69.8
ReadCLIP	66.6	75.6	**47.1**	60.3	**56.8**	68.0
DeGLA	65.3	70.2	43.2	50.9	54.2	60.6
NegCLIP	63.9	75.6	41.5	56.2	52.7	65.9
CE-CLIP	64.8	67.8	**47.1**	56.1	55.9	61.9
DAC (LLM)	53.5	66.7	30.3	46.0	41.9	56.3
DAC (SAM)	52.0	66.5	29.8	45.9	40.9	56.2
Fine-tuned on larger/other datasets (CC3M, LAION, RedCaps)						
CLIC (RedCaps)	56.1	70.8	34.5	47.1	45.3	58.9
CLIC (LAION)	56.5	68.3	34.6	47.6	45.5	57.9
FSC-CLIP (CC3M)	63.6	70.0	40.9	47.6	52.3	58.8
FSC-CLIP (L+C)	63.2	68.8	40.4	49.6	51.8	59.2
CLoVe	58.8	64.0	37.3	46.3	48.1	55.2
CON-CLIP	56.6	70.0	30.9	49.1	43.7	59.5
TSVLC	51.6	66.1	29.8	45.9	40.7	56.0

Table 17: **Unit construction hyperparameters.** We vary the number of units per caption N, the fraction of relation-type units, and the fraction of swap-style foils. **Gray**: baseline configuration.

Setting	Compositionality			Half-Truth			Downstream		
	I2T	T2I	Grp	All	Ent	Rel	ZS Avg	Retr T2I	Retr I2T
Number of units per caption N									
N=1	57.4	39.1	27.4	76.6	74.4	78.8	59.5	70.7	56.3
N=2 **(Ours)**	**57.8**	**39.8**	**27.8**	76.4	73.6	79.2	**59.9**	**71.7**	**56.8**
N=3	57.3	38.7	27.4	74.5	72.4	76.5	59.6	70.6	56.4
N=4	57.5	39.0	27.2	77.4	73.2	81.6	58.9	70.2	56.5
Relation probability p (fraction of relation-type units)									
p=0.0 (entities only)	56.7	39.7	27.5	58.2	75.6	40.8	59.1	70.6	56.5
p=0.25	57.1	39.3	27.0	75.6	76.8	74.5	59.4	70.8	56.7
p=0.50	57.1	39.2	26.4	77.4	78.0	76.9	**60.4**	71.5	56.7
p=0.75	57.5	39.5	27.5	77.4	75.2	79.6	59.3	70.1	56.4
p=1.0 **(Ours)**	**57.8**	**39.8**	**27.8**	76.4	73.6	79.2	59.9	**71.7**	**56.8**
Swap probability (fraction of swap-style foils)									
swap= 0.0 (add/replace only)	57.0	39.5	27.7	76.6	77.2	76.1	59.7	69.4	56.2
swap= 0.5	57.6	38.5	26.6	76.4	74.0	78.8	58.7	69.9	56.7
swap= 1.0 **(Ours)**	**57.8**	**39.8**	**27.8**	76.4	73.6	79.2	**59.9**	**71.7**	**56.8**

Table 18: **Optimization hyperparameters.** We vary learning rate, weight decay, and batch size. **Gray**: baseline configuration.

Setting	Compositionality			Half-Truth			Downstream		
	I2T	T2I	Grp	All	Ent	Rel	ZS Avg	Retr T2I	Retr I2T
Learning rate									
LR= 10^{-6}	56.6	39.9	27.2	66.6	66.9	66.3	**61.7**	70.8	55.7
LR= 2×10^{-6}	57.3	39.0	27.2	70.5	68.1	72.9	61.3	71.4	56.6
LR= 5×10^{-6} **(Ours)**	**57.8**	39.8	27.8	76.4	73.6	79.2	59.9	**71.7**	**56.8**
LR= 10^{-5}	57.6	40.2	**28.1**	78.6	75.2	82.0	58.0	70.0	56.3
LR= 2×10^{-5}	57.7	**40.7**	**28.1**	78.8	74.4	83.1	56.5	69.0	56.0
Weight decay									
WD= 10^{-3}	57.4	39.4	27.1	76.2	72.4	80.0	59.2	70.3	56.4
WD= 5×10^{-3}	57.4	39.0	27.2	75.0	73.2	76.9	59.4	70.5	56.1
WD= 10^{-2} **(Ours)**	**57.8**	**39.8**	27.8	76.4	73.6	79.2	**59.9**	**71.7**	**56.8**
WD= 2×10^{-2}	57.5	39.3	27.4	76.4	74.0	78.8	59.8	70.7	56.2
WD= 5×10^{-2}	57.3	39.5	27.6	76.8	74.0	79.6	59.2	70.4	56.6
Batch size									
BS= 32	57.0	38.7	26.7	75.0	72.0	78.0	**60.4**	69.8	55.8
BS= 64	57.3	39.4	27.0	75.8	71.7	80.0	59.9	70.1	56.1
BS= 128 **(Ours)**	**57.8**	**39.8**	**27.8**	76.4	73.6	79.2	59.9	**71.7**	**56.8**

F Additional Ablations

This appendix reports extended ablations covering optimization hyperparameters and unit construction choices not included in the main paper due to space constraints. The main ablations (Section 5.4) compare backbone, loss weight, training signals, foil placement, and foil generators. Here we report construction and optimization ablations. Half-Truth metrics use COCO (Qwen).

All metrics are accuracies (%) and higher is better. We report three groups: **(i) Compositionality** (I2T/T2I/Group averaged across 16 benchmarks), **(ii) Half-Truth Accuracy** (All/Entity/Relation), and **(iii) Downstream** (zero-shot classification averaged across 7 datasets and retrieval on Flickr8k/COCO).

Unit construction choices. Using only entity units gives 75.6% entity accuracy but 40.8% relation accuracy. Increasing the relation sampling probability improves relations, reaching 79.6% at $p = 0.75$ and 79.2% at $p = 1.0$. Four units per caption reach 81.6% relation accuracy, while two give stronger compositional I2T and retrieval performance. Swap probability also shifts this balance: relation accuracy rises from 76.1% without swaps to 79.2% with full swap usage.

Optimization hyperparameters. Increasing the learning rate to 2×10^{-5} reaches 78.8% overall and 83.1% relation Half-Truth accuracy, but lowers zero-shot classification to 56.5%. The default 5×10^{-6} balances Half-Truth accuracy (76.4%), zero-shot classification (59.9%), and T2I retrieval (71.7%). Weight decay has

smaller effects. Batch sizes 32 and 64 reach 75.0% and 75.8% Half-Truth accuracy, respectively, compared to 76.4% with batch size 128.

F.1 Control Configurations

The foil-placement and Mistral-training controls use 25 epochs, seed 42, and configured batch size 256; the main Qwen-trained model uses batch size 128. The controls therefore compare practical training configurations rather than isolating supervision placement or generator choice with every hyperparameter fixed.

G Assets and Licenses

Table 19 lists assets used for CS-CLIP training, evaluation construction, and compositional evaluation, along with their licenses or service terms. All assets are used strictly for non-commercial academic research in accordance with their respective terms.

Table 19: **Assets used in this work and their licenses.**

Asset	Type	License	Usage
MS-COCO Lin et al. (2014)	Dataset	CC BY 4.0	Training and half-truth evaluation
CC3M Sharma et al. (2018)	Dataset	Freely usable with attribution (Google LLC)	OOD half-truth evaluation
OpenAI CLIP ViT-B/32 Radford et al. (2021)	Model weights	MIT License	Initialization for CS-CLIP
OpenCLIP Cherti et al. (2023)	Codebase	MIT License	Fine-tuning framework
Qwen3-8B-AWQ Yang et al. (2025)	LLM	Apache 2.0	Caption parsing pipeline (text only)
Qwen3-VL-30B-A3B-Instruct	VLM	Apache 2.0	Evaluation screening and training-foil audit
Claude Opus 5	Hosted model	Proprietary (Anthropic terms)	Evaluation review and replacement proposals
Compositional evaluation benchmarks (all publicly available)			
ARO Yuksekgonul et al. (2023)	Benchmark	MIT License	Evaluation only
BLA Chen et al. (2023)	Benchmark	MIT License	Evaluation only
COCO-CF Le et al. (2023)	Benchmark	CC BY 4.0	Evaluation only
COLA Ray et al. (2023)	Benchmark	MIT License	Evaluation only
ColorFoil Samin et al. (2025)	Benchmark	MIT License	Evaluation only
ColorSwap Burapacheep et al. (2024)	Benchmark	MIT License	Evaluation only
MMVP Tong et al. (2024)	Benchmark	MIT License	Evaluation only
NegBench Alhamoud et al. (2025)	Benchmark	MIT License	Evaluation only
SPEC Peng et al. (2024)	Benchmark	Research-only (no explicit license file)	Evaluation only
SugarCrepe Hsieh et al. (2023)	Benchmark	MIT License	Evaluation only
SugarCrepe++ Dumpala et al. (2024)	Benchmark	MIT License + CC BY 4.0	Evaluation only
VALSE Parcalabescu et al. (2022)	Benchmark	MIT License	Evaluation only
VisMin Awal et al. (2024)	Benchmark	CC BY 4.0	Evaluation only
VL-CheckList Zhao et al. (2022)	Benchmark	Research-only (no explicit license file)	Evaluation only
What'sUp Kamath et al. (2023)	Benchmark	MIT License	Evaluation only
Winoground Thrush et al. (2022)	Benchmark	Custom research-only license (gated)	Evaluation only

Claude is accessed as a hosted service under the applicable Anthropic Consumer Terms or Commercial Terms. The authors manually verify evaluation candidates after the automated checks (Appendix C.2).

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction state all main claims (half-truth failure mode, unit-level supervision, HT-Acc 76.4% on COCO (Qwen), best compositional I2T 57.8%, best Group Accuracy) that are directly supported by the main-paper tables (half-truth and ablations) and the compositional results in Appendix E. Scope is bounded to COCO fine-tuning; OOD transfer is presented as supporting evidence, not a primary claim.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: Section 6 discusses text-only parsing errors, the zero-shot classification trade-off, and limits on factual correctness and demographic fairness. Appendix C.2 reports the training-foil audit and distinguishes automated training-data filtering from manual evaluation verification.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

   Answer: [N/A]

   Justification: The paper makes no theoretical claims and contains no theorems, lemmas, or proofs; all contributions are empirical.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: Section 5 details the training setup (optimizer, learning rate, batch size, backbone, temperature). The caption-parsing and foil-generation pipeline is described in Section 3 and Appendix B. All evaluation benchmarks are publicly available. Code is available at `https://github.com/kargibora/CS-CLIP`.

   Guidelines:

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

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

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

   Answer: [Yes]

   Justification: Code, environment requirements, and training and evaluation commands are available at `https://github.com/kargibora/CS-CLIP`. The repository links to the released checkpoint, manually verified Half-Truth evaluation set, and generated training annotations on Hugging Face; training images are obtained separately from COCO.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: Section 5 specifies the optimizer, learning rate, weight decay, batch size, training epochs, backbone, and temperature initialization; Section 5.4 examines the hyperparameter choices. Appendix C describes the constructed evaluation sets, and Appendix E.1 defines the compositional evaluation metrics.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: Appendix E.2 reports per-benchmark standard deviations across five CS-CLIP runs. Point estimates use the best-performing run, with overall compositional I2T accuracy of 57.8%. Baselines use their published checkpoints.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).

- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

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

   Answer: [No]

   Justification: Section 5 reports training on 8×A100 GPUs for 25 epochs with batch size 128. We do not report per-experiment wall-clock times, memory usage, or the total compute used across the research project.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics. We fine-tune a publicly available vision-language model on COCO and evaluate on existing benchmarks and manually verified COCO/CC3M comparisons; we collect no private data and conduct no research with human participants.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: Section 6 discusses improved sensitivity to incorrect details in image–text retrieval, along with remaining factual errors and demographic biases. These limitations remain relevant when using or deploying the released model.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.

- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [N/A]

    Justification: We release a fine-tuned CLIP checkpoint and annotations derived from existing image–caption datasets for retrieval research. The model card and dataset documentation describe their intended use and limitations; CS-CLIP does not guarantee factual correctness or demographic fairness.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: Appendix G lists licenses and usage for the base datasets, CLIP weights, OpenCLIP code, and compositional benchmarks. The original sources are cited in the paper, and the released assets include their own documentation.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [Yes]

    Justification: The Half-Truth benchmark contains 1,437 manually verified comparisons from COCO and CC3M, documented at `https://huggingface.co/datasets/kbora/Half-Truths`. The checkpoint and automatically generated training annotations are documented at `https://huggingface.co/kbora/CS-CLIP` and `https://huggingface.co/datasets/kbora/CS-CLIP-Training`, respectively; Appendix C.2 distinguishes evaluation verification from training-data filtering.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.

- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

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

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

    Answer: [N/A]

    Justification: We do not conduct any crowdsourcing or research with human subjects. The half-truth evaluation suite (1,437 comparisons across COCO and CC3M) is manually verified by the authors themselves; no crowdworkers or external annotators are involved.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: No human subjects are involved in this research. The authors manually verified the evaluation foils against their images after automated screening. Training foils were automatically generated and filtered, without manual verification at scale.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [Yes]

    Justification: We use text-only Qwen3-8B-AWQ to parse captions and generate foils, with Mistral-8B in the cross-generator controls (Appendix B). Evaluation candidates undergo image-grounded Qwen3-VL-30B screening and Claude Opus 5 review with proposed replacements, followed by manual verification by the authors (Appendix C.2); training foils are not manually verified at scale.

    Guidelines:

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