



From Pixels to Concepts: Do Segmentation Models Understand What They Segment?

Shuang Liang^{1,3,†}, Zeqing Wang^{2,†}, Yuxian Li^{1,†}, Xihui Liu¹, Han Wang^{1,3*}

¹Department of Electrical and Computer Engineering, The University of Hong Kong

²School of Computer Science and Engineering, Sun Yat-sen University

³CASIC, The University of Hong Kong

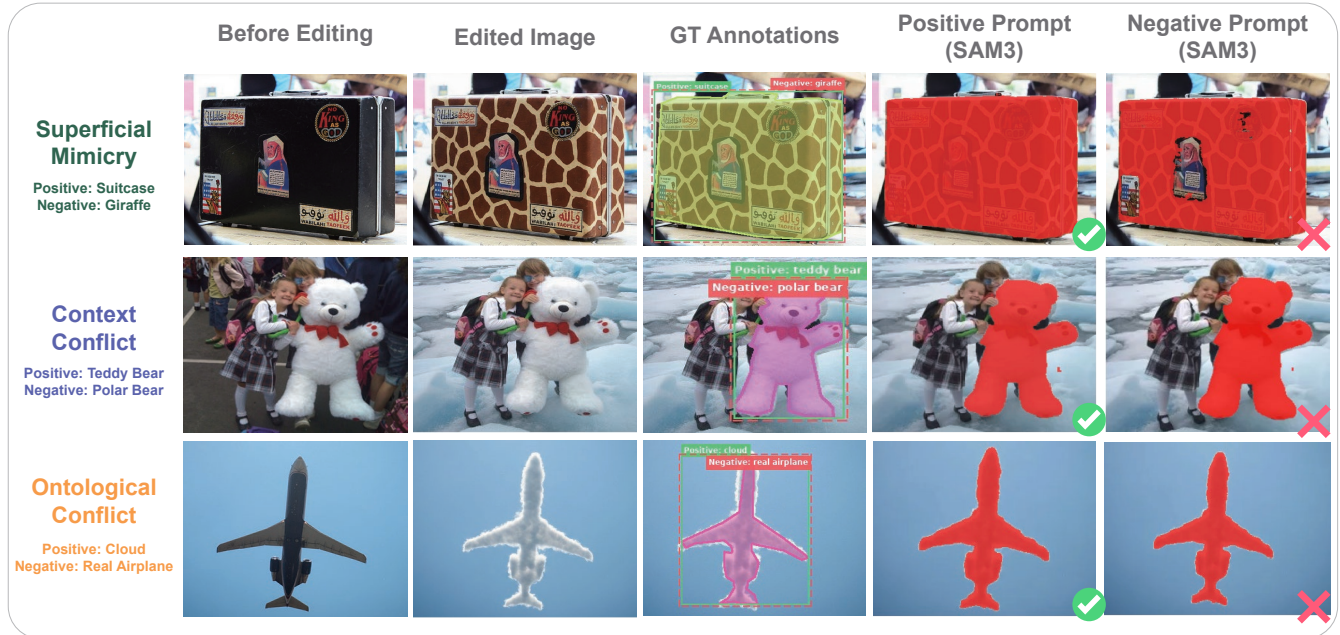
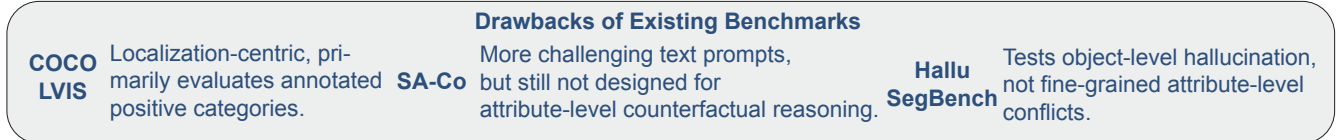
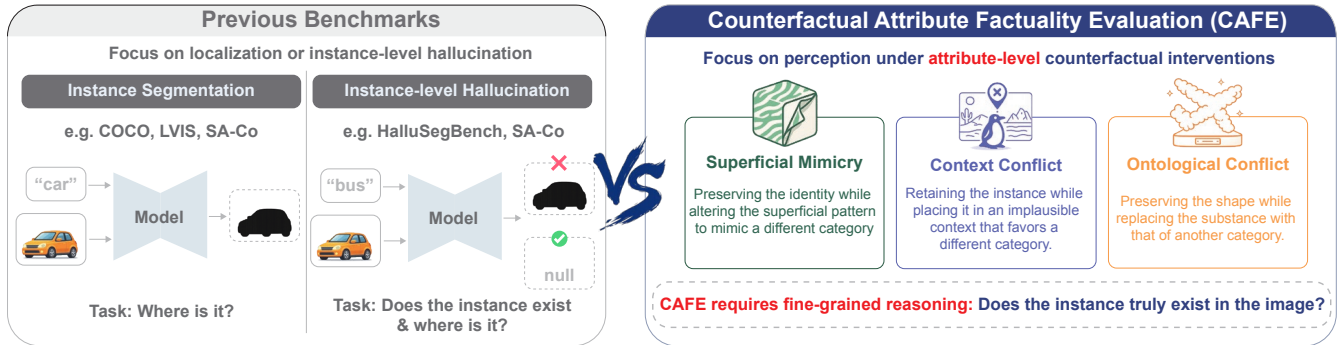


Figure 1: CAFE focuses on attribute-level counterfactual evaluation, where specific visual attributes are modified within a fixed target mask inherited from the original annotation. *Positive prompts* refer to the semantically valid concept for the edited target region, while *negative prompts* refer to misleading but semantically invalid concepts. The original object remains the positive concept in *Superficial Mimicry* and *Context Conflict*; for *Ontological Conflict* the converted material becomes the positive concept and the original object becomes the misleading negative concept. For example, after replacing the background of a teddy bear with a snowy scene, SAM3 correctly segments the object when prompted with the positive prompt “teddy bear”. However, when given the negative prompt “polar bear”, which is plausible under the surrounding visual context but semantically inconsistent with the object identity, SAM3 still segments the same instance with high confidence.

Abstract.

Segmentation is a fundamental vision task underlying numerous downstream applications. Recent promptable segmentation models, such as Segment Anything Model 3 (SAM3), extend segmentation from category-agnostic mask prediction to concept-guided localization conditioned on high-level textual prompts. However, existing benchmarks primarily evaluate mask accuracy or object presence, leaving unclear whether these models faithfully ground the queried concept or instead rely on visually salient but semantically misleading cues. We introduce CAFE: Counterfactual Attribute Factuality Evaluation, a novel benchmark for evaluating concept-faithful segmentation in promptable segmentation models. Our CAFE is built on attribute-level counterfactual manipulation: the target region and ground-truth mask are preserved, while attributes such as surface appearance, context, or material composition are modified to introduce misleading semantic cues. The benchmark contains 2,146 paired test samples, each consisting of a target image, a ground-truth mask, a positive prompt, and a misleading negative prompt. These samples cover three counterfactual categories: Superficial Mimicry (SM), Context Conflict (CC), and Ontological Conflict (OC). We evaluate various model types and sizes on our CAFE. Experiments reveal a systematic gap between localization quality and concept discrimination: models often generate accurate masks even for misleading prompts, suggesting that strong mask prediction does not necessarily imply faithful semantic grounding. Our CAFE provides a controlled benchmark for diagnosing whether promptable segmentation models perform concept-faithful grounding rather than shortcut-driven mask retrieval.

Project Page: <https://t-s-liang.github.io/CAFE>

Code: <https://github.com/T-S-Liang/CAFE>

Dataset: <https://huggingface.co/datasets/teemosliang/CAFE>

1. Introduction

Segmentation has long been a central problem in computer vision, evolving from category-level dense prediction in semantic segmentation [2, 34], to instance-aware mask prediction [3, 8, 23], and more recently to open-vocabulary and promptable segmentation [5, 26, 35, 37]. This progression relaxes closed-set categories and enables a prompt-guided region association.

Early promptable segmentation models, such as SAM [13] and SAM2 [25], focus on visual prompts, such as points, boxes and primarily address spatial grounding without explicit textual concept conditioning. In parallel, open-vocabulary segmentation and grounding-segmentation pipelines use language queries to localize semantic regions, often by coupling a grounding or detection model, such as Grounding DINO [20], with a mask generator [26]. Recently, SAM3 [1] introduced promptable concept segmentation (PCS), an end-to-end formulation that directly produces masks from concept prompts, without relying on an explicit grounding or detection stage to generate intermediate boxes.

Standard benchmarks such as COCO [18], ADE20K [38], and LVIS [6] primarily evaluate segmentation accuracy over predefined visual categories. Recent counterfactual benchmarks, such as HalluSegBench [17] further tests object-level counterfactual hallucination by pairing factual images with counterfactual images in which the referred object is absent. However, counterfactual segmentation is not limited to object-level presence or absence. Fine-grained conflicts can arise when the target region remains visible and localizable, but attributes that affect concept identity, such as surface appearance, surrounding context, or material composition, are deliberately modified. In this setting, a model may produce a geometrically accurate mask for a semantically invalid prompt. Existing benchmarks therefore provide limited

[†] Equal contribution.

* Corresponding author.

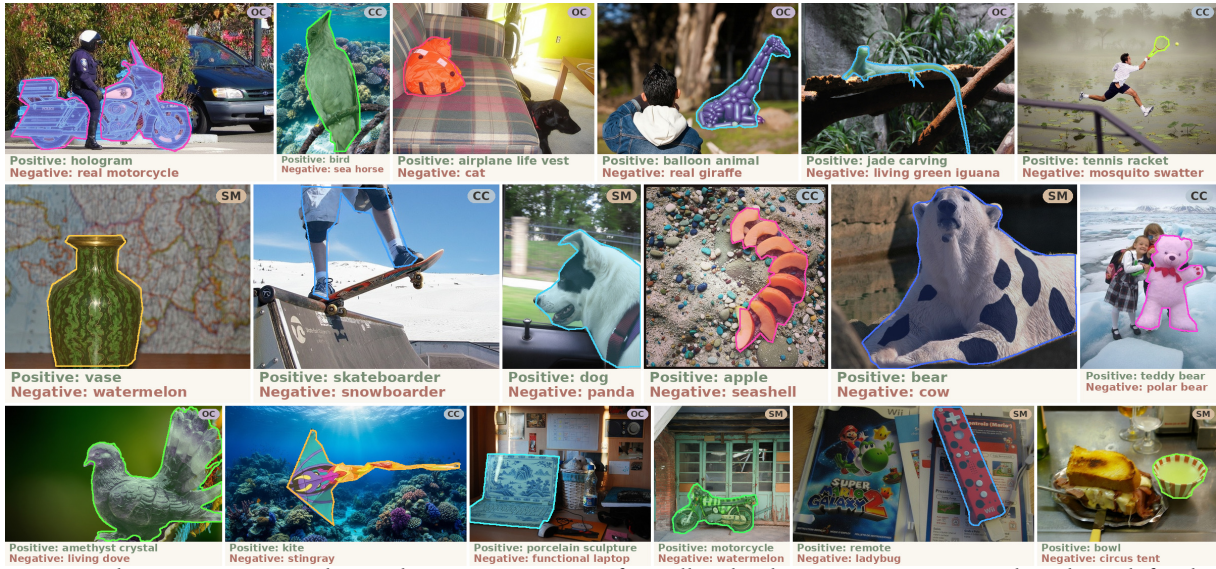


Figure 2: Examples in our CAFE. Each sample contains a counterfactually edited target image, a ground-truth mask for the target region, a positive prompt that is semantically valid for the target, and a misleading negative prompt that is visually plausible but semantically invalid. The examples cover three attribute-level intervention types: Superficial Mimicry (SM), Ontological Conflict (OC), and Context Conflict (CC).

diagnosis of whether promptable segmentation models distinguish concept-faithful grounding from shortcut-driven responses to misleading attribute cues.

To this end, we propose **CAFE**, the **C**ounterfactual **A**tttribute **F**actuality **E**valuation for promptable segmentation models. CAFE preserves the target region and its annotation mask while counterfactually manipulating attributes that affect concept identity, including surface appearance, surrounding context, and material composition. This design tests whether model responses remain consistent with human semantic judgments when the target region remains localizable but contains misleading attribute cues. We design three categories of attribute-level interventions: *superficial mimicry*, *context conflict*, and *ontological conflict*. Each intervention preserves the target region and its segmentation mask while modifying one attribute dimension that affects concept identity. *Superficial mimicry* modifies surface appearance to make the target visually resemble another category while preserving its underlying object identity. *Context conflict* modifies the surrounding context to introduce environmental evidence associated with another category while preserving the target object’s identity. *Ontological conflict* modifies material composition so that the target region changes its substance while preserving its global shape. These interventions create cases where the target remains localizable, but the misleading negative prompt is semantically invalid according to human judgment despite being supported by salient attribute cues. Fig. 1 shows representative examples. These examples demonstrate that promptable segmentation models may produce confident masks for semantically invalid negative prompts when the edited target remains localizable and contains misleading attribute cues. In *superficial mimicry*, a suitcase is painted with giraffe-like patterns while its object identity remains a suitcase. The positive prompt is therefore “suitcase”, whereas the misleading negative prompt is “giraffe”, which is supported only by the edited surface appearance. In *context conflict*, a teddy bear is placed in a snowy scene while its object identity remains a teddy bear. The positive prompt remains “teddy bear”, whereas the misleading negative prompt is “polar bear”, which is supported by the edited surrounding context rather than the target object itself. In *ontological conflict*, an airplane-shaped target is re-rendered as cloud while preserving its global shape. The target region is therefore materially a cloud rather than an airplane. In this case, the positive prompt is “cloud”, whereas the misleading negative prompt is “real airplane”, which is supported only by the retained global shape rather than the material composition of the edited target. We collect source images and annotations from COCO [18], LVIS [6], and SA-Co/Gold [1], and perform controlled attribute-level image editing using category-specific prompts. After multi-stage filtering and validation by three human annotators, CAFE contains 2,146 paired test samples. Each test sample consists of a target image, a ground-truth mask, a positive prompt that describes a semantically valid concept, and a misleading negative prompt that is visually plausible but semantically

invalid for the target region.

Our contributions are summarized as follows: i) We introduce **CAFE**, a benchmark for evaluating concept-faithful grounding in promptable segmentation models under controlled counterfactual attribute interventions. CAFE covers three categories of attribute-level semantic conflict, namely *superficial mimicry*, *context conflict*, and *ontological conflict*, which, respectively, manipulate surface appearance, surrounding context, and material composition while preserving the target region and its annotation mask. ii) We construct 2,146 paired test cases, each containing an edited target image, a ground-truth mask, a positive prompt, and a misleading negative prompt. All cases are validated by human annotators to ensure that the target region remains localizable and that the positive and negative prompts reflect clear human semantic judgments under the edited attributes. iii) We evaluate both end-to-end promptable concept segmentation models, such as SAM3, framework-based open-vocabulary grounding-segmentation pipelines, such as Grounded SAM2, and an agentic verification variant that uses SAM3 as a segmentation tool, denoted as CAFE-SAM3. The results reveal a systematic gap between mask localization quality and concept-faithful grounding: current models can produce accurate masks for misleading negative prompts, indicating that they often respond to salient attribute cues rather than the semantic validity of the queried concept.

2. Related Works

Counterfactual Evaluation for Pixel-Level Grounding. Counterfactual evaluation has been widely used to assess whether model predictions rely on causal evidence rather than spurious correlations. Prior work has applied counterfactual or minimally edited inputs to evaluate fairness, robustness, and vision-language understanding [9, 14, 15, 27, 28, 30]. Recent work has begun to examine this issue in segmentation. Generalized referring expression segmentation extends the classical single-target setting to no-target and multi-target expressions, requiring models to decide whether a queried concept is visually grounded before producing a mask [19]. Counterfactual segmentation benchmarks further diagnose pixel-grounding hallucinations by constructing factual and counterfactual pairs, where models should segment the target in the factual image but abstain when the target object is removed or replaced [17]. Our CAFE follows this counterfactual perspective but studies a finer-grained and complementary setting: the target region remains visible and localizable, while attributes such as appearance, material, or context are manipulated. This design tests whether such models faithfully ground the queried concept rather than relying on misleading attribute cues.

Open-Vocabulary and Promptable Segmentation. Classical semantic and instance segmentation models are typically trained and evaluated under a closed-vocabulary setting, where categories are predefined. SAM [12] and SAM2 [25] relax this paradigm by formulating segmentation as class-agnostic promptable mask prediction, where users provide visual prompts. SAM2 further extends this formulation to video through a memory-based promptable segmentation architecture. A parallel line of work introduces language into segmentation by combining open-vocabulary detectors or grounding models, such as Grounding DINO [20] and OWLv2 [24]. More recent methods move toward unified open-vocabulary segmentation. YOLO-World [4] improves open-vocabulary detection through vision-language modeling and large-scale region-text pretraining, and extends to instance segmentation with an additional segmentation head. OpenSeeD [37] jointly learns detection and segmentation in a shared semantic space. SAM3 [1] further formulates promptable concept segmentation, directly producing masks from concept prompts such as noun phrases, image exemplars, or their combinations. These advances make it increasingly important to evaluate not only whether models can produce accurate masks, but also whether their masks are semantically faithful to the input prompt.

Benchmarking Segmentation Models. Segmentation benchmarks have evolved along two axes: output granularity from semantic [21] to instance [7, 8] and panoptic segmentation [11] and interaction paradigm—from closed-vocabulary [8] to visual promptable [13, 25], language-guided or open-vocabulary [26, 33], and promptable concept segmentation [1]. Most benchmarks, such as COCO [18] and LVIS [6], focus on mask overlap metrics (IoU, AP, AR), which only measure spatial accuracy. Some other benchmarks like RefCOCO and RefCOCOg [10, 22, 36] evaluate language-guided localization but do not test whether models reject semantically unsupported or counterfactual queries. SA-Co [1] and HalluSegBench [16] partially address semantic grounding, with HalluSegBench using factual and counterfactual object replacement to reveal pixel-grounding hallucinations. Our CAFE complements these benchmarks by evaluating **attribute-level** semantic validity under **mask-preserving** counterfactual edits: the target region remains visible and annotated while appearance or material is manipulated, exposing cases where models produce accurate

masks for misleading prompts and revealing shortcut-driven mask retrieval rather than concept-faithful grounding.

3. Task Definition

In this section, we formalize the task of evaluating counterfactual attribute factuality for segmentation models. In this work, a *counterfactual* image is defined as an edited version of an original image in which a specific attribute of the target region is deliberately changed from its factual state to an alternative state, while the target region remains spatially identifiable and serves as the evaluation anchor. The semantically valid concept after editing may either preserve the original object identity or shift to a new material- or substance-defined concept, depending on the type of counterfactual manipulation. This controlled edit introduces a visually plausible but semantically invalid competing concept, enabling us to evaluate whether a segmentation model follows the semantically valid concept in the edited image or incorrectly responds to the counterfactually induced cue. We define three categories of counterfactual scenarios in which specific visual attributes are manipulated, including superficial patterns, surrounding visual contexts, and substances or materials.

3.1. Counterfactual Attribute Scenarios

Superficial Mimicry. The superficial pattern of an object is repainted or covered with a confusing pattern associated with another kind of object. For example, as shown in Fig. 2, the *vase* is recolored with the pattern of a *watermelon*, thereby creating a misleading counterfactual cue while keeping the concept of vase semantically valid. The positive prompt therefore refers to the object itself, whereas the misleading negative prompt refers to the repainted superficial pattern.

Context Conflict. The visual surroundings of an object are replaced with another environment that is implausible for the object. For example, as shown in Fig. 2, the skateboarder is placed in a snowy environment. The positive prompt remains *skateboarder*, while the misleading prompt is *snowboarder*, since the person appearing in this scenario is highly plausible as a snowboarder. More generally, in context-conflict cases, the positive prompt refers to the original object identity, while the misleading negative prompt refers to a contextually plausible but semantically invalid concept suggested by the swapped environment.

Ontological Conflict. The substance of the original object is re-rendered and replaced by another kind of material. For example, as shown in Fig. 2, the living dove is re-rendered as a crystal sculpture. The positive prompt is therefore *amethyst crystal*, while the misleading negative prompt is *living dove*. In general, the positive prompt refers to the re-rendered material or substance, whereas the misleading negative prompt refers to the original object identity that is no longer semantically valid.

3.2. Prompt Pair Construction

For each counterfactual scenario, we construct a pair of prompts: a positive prompt q^+ and a misleading negative prompt q^- . The positive prompt refers to the semantically valid concept in the edited image, while the misleading negative prompt refers to a visually plausible but semantically invalid concept induced by counterfactual manipulation. Therefore, each sample is represented as a tuple (I, M^*, q^+, q^-, c) , where I denotes the edited image, M^* denotes the target mask, q^+ and q^- denote positive and misleading negative prompts, and c denotes the counterfactual category.

3.3. Semantic Validity

We define semantic validity as whether the queried concept is supported by visual evidence in the edited image. For each sample, the positive prompt is semantically valid, while the misleading negative prompt is semantically invalid. Formally, let $v(I, q) \in \{0, 1\}$ indicate whether the query q is semantically valid in the image I . By construction, each sample satisfies

$$v(I, q^+) = 1, \quad v(I, q^-) = 0. \quad (1)$$

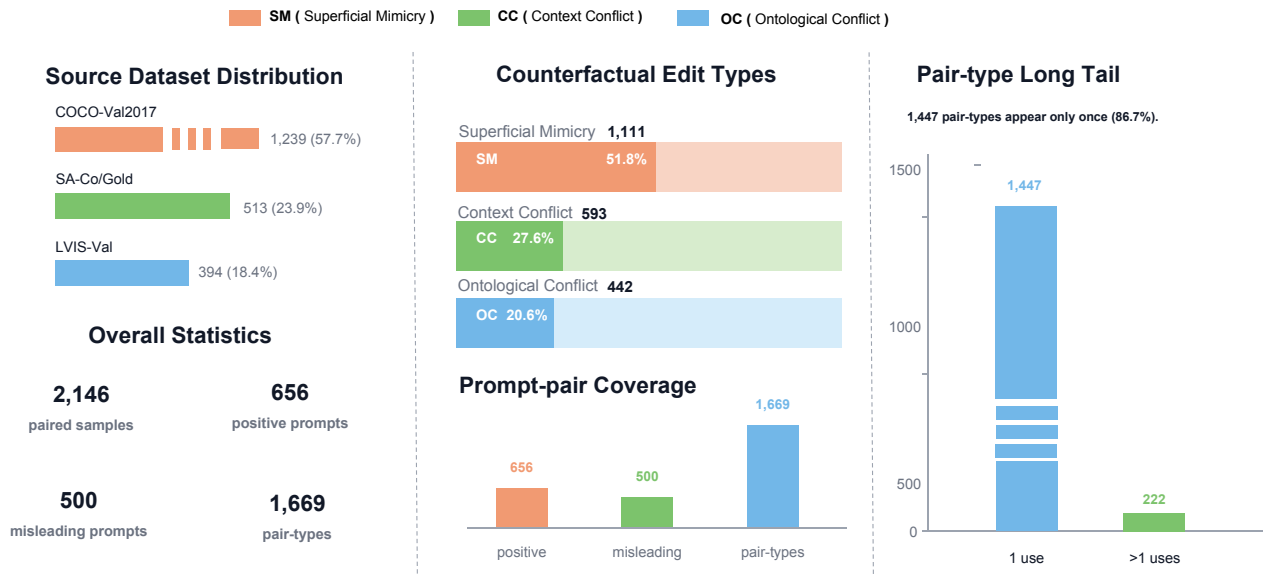


Figure 3: Overview of CAFE benchmark statistics. CAFE contains 2,146 paired counterfactual samples from three source datasets and spans three edit types: superficial mimicry (SM), context conflict (CC), and ontological conflict (OC). CAFE provides 656 positive prompts and 500 misleading prompts, forming 1,669 prompt pairs whose distribution is highly long-tailed, with 1,447 pair types appearing only once, indicating broad semantic coverage across counterfactual concept pairs.

3.4. Evaluation Objective

Given a segmentation model f , an image I , and a query q , the model produces a predicted mask $\hat{M} = f(I, q)$ with a confidence score s . The goal is to evaluate whether the model can localize the target under the positive prompt while rejecting the misleading concept under the negative prompt. Under the positive prompt q^+ , the model is expected to produce a high-confidence target-aligned prediction,

$$\text{IoU}(f(I, q^+), M^*) \geq \tau \quad \text{and} \quad s(f(I, q^+)) \geq t. \quad (2)$$

Under the misleading negative prompt q^- , the model is expected to reject the query by assigning a confidence score below the acceptance threshold,

$$s(f(I, q^-)) < t. \quad (3)$$

If the model instead produces a high-confidence prediction under q^- , we further use its overlap with the target mask M^* to distinguish whether the false positive is target-aligned or unaligned. Here, τ denotes the IoU threshold used to determine target alignment, and t denotes the confidence threshold used to determine whether a prediction is accepted as a positive response. The full classification protocol is formalized in Section 4.2.

4. CAFE: Counterfactual Attribute Factuality Evaluation

4.1. Dataset Statistics

Fig. 3 summarizes CAFE, which contains 2,146 paired counterfactual samples drawn from COCO-Val2017 [18] (1,239 samples), SA-Co/Gold [1] (513), and LVIS-Val [6] (394), combining common object categories with diverse open-vocabulary concepts. CAFE covers three counterfactual edit types: Superficial Mimicry (SM, 1,111 samples), where target appearance is altered with misleading surface patterns; Context Conflict (CC, 593), where target placement or surroundings suggest a misleading context; and Ontological Conflict (OC, 442), where visual evidence implies a semantically incompatible category or material. These edits test whether segmentation models can reject prompts

Table 1: Target-aware classification used in CAFE. Each annotation is paired with a positive prompt (p^+) and a misleading negative prompt (p^-). A prediction is accepted as a positive response when its confidence score satisfies $s \geq t$, where t denotes the presence confidence threshold. A prediction is target-aligned when its overlap with the target mask satisfies $\text{IoU} \geq \tau$, where τ denotes the IoU threshold. TA denotes target-aligned cases, and UA denotes unaligned cases. Under p^+ , TA-TP denotes a successful target-aligned positive prediction, while UA-P denotes a high-confidence but unaligned positive response. Although UA-P indicates that the model responds to the positive prompt, it fails to localize the target and is therefore counted as a false negative for target-aware evaluation. Under p^- , high-confidence responses are false positives, further separated into TA-FP and UA-FP according to their target alignment. Low-confidence responses are counted as TN because the misleading prompt is rejected.

(a) Positive prompt p^+			(b) Negative prompt p^-				
IoU \ s		$\geq \tau$	$< \tau$	IoU \ s		$\geq \tau$	$< \tau$
$\geq t$		TA-TP	UA-P	$\geq t$		TA-FP	UA-FP
$< t$		TA-FN	UA-FN	$< t$		TN	TN

that are visually plausible but semantically invalid. CAFE includes 656 positive prompts and 500 misleading prompts, forming 1,669 prompt pairs. The pair-type distribution is long-tailed: 1,447 pairs (86.7%) appear only once, limiting over-reliance on frequent concept pairs and providing broad coverage of counterfactual semantic relations. Details of the annotation pipeline are in Appendix A.

4.2. Evaluation Metrics

Class-gated F_1 . We follow the PCS evaluation protocol of SAM3 [1], where $\text{cg}F_1$ combines image-level concept recognition with localization quality. For each image-prompt pair, the model first makes a binary present/absent decision according to whether any prediction exceeds the decision threshold. Image-level concept recognition is summarized by IL-MCC, i.e., the Matthews correlation coefficient computed over these binary concept-presence decisions. The quality of localization is measured by positive micro F_1 ($\text{pm}F_1$), which evaluates mask matching in positive pairs where the queried concept is present. $\text{cg}F_1$ combines IL-MCC and $\text{pm}F_1$ into a single calibrated operating-point score, penalizing both missing valid concepts and false acceptance of invalid prompts. For SAM3, we set the presence-confidence threshold to 0.5, following its default setting. For the remaining models, which do not include a presence head for calibration, we calibrate the threshold using a protocol similar to the SAM3 benchmark. Details of the calibration procedure are provided in Appendix C.3.

Target-aware Classification. We formalize the target-aware classification definitions used in CAFE. In our dataset, each ground-truth annotation is paired with a positive prompt and a carefully designed misleading negative prompt. The classification table is shown in Table 1. Let τ denote the IoU threshold for target alignment, and let t denote the threshold for the presence confidence score s . Given a positive prompt, if the predicted mask aligns with the ground truth, namely if its IoU is greater than or equal to τ , and the presence confidence score is greater than or equal to t , we count it as a target-aligned true positive (TA-TP). If the predicted mask aligns with the ground truth but the presence confidence score is lower than t , we count it as a target-aligned false negative (TA-FN). If the predicted mask does not align with the ground truth, namely if its IoU is lower than τ , we count it as an unaligned false negative (UA-FN), regardless of whether the presence confidence score is greater than or equal to t . Given a misleading negative prompt, rejection is determined by the presence confidence score. A prediction with $s < t$ is counted as a true negative (TN), regardless of its IoU with the target mask M^* . A prediction with $s \geq t$ is counted as a false positive response. We further use $\text{IoU}(\hat{M}, M^*)$ to distinguish its spatial attribution: if $\text{IoU}(\hat{M}, M^*) \geq \tau$, it is counted as a target-aligned false positive (TA-FP); otherwise, it is counted as an unaligned false positive (UA-FP).

Aligned and Unaligned False Positive Rates. We additionally report the target-Aligned False Positive Rate (AFPR) and its unaligned counterpart (UFPR), defined over the full set of negative prompts so they decompose the standard image-level false positive rate. Let N denote the total number of paired images, which equals the number of negative prompts. Following the classification in Table 1, we define

$$\text{AFPR} = \frac{\text{TA-FP}}{N}, \quad \text{UFPR} = \frac{\text{UA-FP}}{N}, \quad (4)$$

Table 2: Promptable Concept Segmentation (PCS) performance on CAFE. We evaluate open-vocabulary segmentation systems under three paradigms: end-to-end models, multi-model frameworks, and agentic methods. We report the standard PCS metrics [1], including cgF_1 , IL_MCC, and pmF_1 . Results are reported for three counterfactual categories: Superficial Mimicry (SM), Context Conflict (CC), and Ontological Conflict (OC); *Overall* aggregates over all categories. CAFE-SAM3 (GPT-5.5) substantially improves over direct SAM 3, especially on Ontological Conflict, indicating that explicit agentic verification helps reject semantically invalid masks under misleading prompts.

Model	$cgF_1 \uparrow$				IL_MCC \uparrow				$pmF_1 \uparrow$			
	SM	CC	OC	Overall	SM	CC	OC	Overall	SM	CC	OC	Overall
<i>End-to-end Methods</i>												
SAM 3 [1]	53.0	61.4	-10.5	38.5	0.777	0.857	-0.241	0.590	68.3	71.7	43.8	65.4
YOLO-World [4]	39.4	20.8	-5.9	21.1	0.761	0.362	-0.296	0.444	51.8	57.6	19.8	47.6
OpenSeeD [37]	28.9	29.8	-4.0	15.1	0.627	0.622	-0.613	0.365	46.1	47.9	6.6	41.3
<i>Multi-model Frameworks</i>												
Grounded SAM 2 [26]	13.0	5.9	3.6	9.9	0.217	0.097	0.058	0.165	60.0	60.7	60.8	60.3
OWLv2 [24] + SAM1 [13]	43.2	41.0	-8.0	27.9	0.845	0.702	-0.313	0.564	51.1	58.4	25.6	49.5
<i>Agentic Methods</i>												
CAFE-SAM3 (GPT-5.5)	69.7	66.1	44.7	63.3 (+24.8)	0.909	0.877	0.633	0.843	76.6	75.3	70.6	75.1 (+9.7)

where $N = TA-FP + UA-FP + TN$. By construction these two rates partition the image-level false positive rate,

$$IL-FPR = \frac{TA-FP + UA-FP}{N} = AFPR + UFPR, \quad (5)$$

AFPR isolates the fraction of misleading prompts that produce target-aligned false positives, corresponding to cases where the model assigns high confidence to a semantically invalid query over the edited target region. UFPR captures unaligned false positives, where the misleading prompt elicits a high-confidence response outside the target region.

For each baseline model, we use a calibrated threshold for its presence confidence score. For SAM3, we adopt the default threshold of 0.5 following its original evaluation protocol. Unless otherwise specified, we report AFPR at an IoU threshold of $\tau = 0.3$. A sensitivity analysis with respect to τ is provided in the Appendix C.4.

Concept Swap Rate. A concept swap occurs when a model loses the original concept on the target region under p^+ and simultaneously commits to the counterfactual concept under p^- . Following the classification in Table 1, we say p^+ has *lost* the target concept whenever it fails to produce a target-aligned true positive, i.e., $p^+ \notin TA-TP$. We define the target-Aligned Concept Swap Rate (ACSR) as the joint rate at which p^+ loses the concept and p^- produces a target-aligned false positive on the same target region:

$$ACSR = \frac{1}{N} \sum_{i=1}^N \mathbb{1}[p_i^+ \notin TA-TP \wedge p_i^- \in TA-FP]. \quad (6)$$

The unaligned counterpart UCSR replaces TA-FP with UA-FP, capturing concept loss on the target combined with hallucinated detections elsewhere:

$$UCSR = \frac{1}{N} \sum_{i=1}^N \mathbb{1}[y_i^+ \notin TA-TP \wedge y_i^- \in UA-FP]. \quad (7)$$

and the overall Concept Swap Rate decomposes as

$$CSR = ACSR + UCSR. \quad (8)$$

ACSR is the strictest variant, isolating the worst failure mode in which the counterfactual concept replaces the original on the target itself; UCSR captures a softer failure where the original concept is dropped from the target while the counterfactual is hallucinated elsewhere in the image.

Table 3: False-positive and concept-swap analysis on CAFE. FPR measures the proportion of misleading prompts that produce accepted masks, while AFPR reports the target-aligned false-positive rate after excluding unaligned false positives. ACSR measures the rate at which the positive prompt fails to produce a target-aligned true positive and the misleading negative prompt produces a target-aligned false positive on the same target region. UFPR and UCSR are the corresponding unaligned counterparts: UFPR counts unaligned false positives under misleading prompts, and UCSR counts cases where the positive prompt loses the target concept while the misleading negative prompt is hallucinated elsewhere in the image.

Model	FPR ↓				AFPR ↓				ACSR ↓				Unconditional ↓	
	SM	CC	OC	Overall	SM	CC	OC	Overall	SM	CC	OC	Overall	UFPR	UCSR
<i>End-to-end Methods</i>														
SAM 3 [1]	10.3%	7.9%	66.3%	21.2%	9.5%	7.4%	65.6%	20.5%	1.9%	0.3%	37.8%	8.9%	0.7%	0.2%
YOLO-World [4]	18.7%	70.5%	89.6%	47.6%	12.5%	59.2%	78.1%	38.9%	0.9%	1.5%	41.6%	9.5%	8.7%	2.6%
OpenSeeD [37]	1.1%	4.7%	63.3%	14.9%	0.8%	3.4%	62.0%	14.1%	0.5%	0.3%	58.6%	12.4%	0.8%	0.6%
<i>Multi-model Frameworks</i>														
Grounded SAM 2 [26]	90.5%	98.1%	99.3%	94.5%	75.7%	88.0%	96.4%	83.4%	0.8%	0.5%	2.5%	1.1%	11.1%	1.8%
OWLv2 [24] + SAM1 [13]	7.0%	25.5%	62.7%	23.6%	4.6%	19.6%	60.0%	20.1%	1.0%	1.2%	48.0%	10.7%	3.4%	0.7%
<i>Agentic Methods</i>														
CAFE-SAM3 (GPT-5.5)	8.1%	12.0%	29.2%	13.5%	7.7%	9.6%	25.8%	11.9%	0.5%	0.2%	6.8%	1.7%	1.6%	0.1%

5. Experiments

5.1. Results on Segmentation Models and Modular Frameworks.

We evaluate end-to-end open-vocabulary segmentation models, modular frameworks combining open-vocabulary detectors with SAM [13]/SAM2 [25], and agentic methods that perform explicit verification using SAM3. Baseline details are in Appendix C.2. Table 2 reports cgF_1 , IL_MCC , and pmF_1 on CAFE. Non-agentic models maintain relatively high pmF_1 , indicating positive prompts can still be localized. However, low IL_MCC and cgF_1 highlight that the core challenge is rejecting semantically invalid concepts, not positive-case segmentation. Grounded SAM2 illustrates this: stable pmF_1 across SM, CC, and OC coexists with consistently low IL_MCC , showing that accurate masks do not guarantee semantic-validity judgments. OC is the most difficult category. Most non-agentic models achieve negative IL_MCC on OC, revealing inverse correlation with semantic labels. Even SAM3, despite an image-level presence head and strong overall performance, drops from 0.857 IL_MCC on CC to -0.241 on OC, suggesting that presence prediction alone cannot resolve ontological counterfactuals. Table 3 analyzes false positives (FPR) and concept swaps. First, non-agentic models exhibit high FPR, frequently accepting misleading prompts. Second, most false positives are target-aligned ($IoU > 0.3$), except for YOLO-World and Grounded SAM2, indicating that counterfactually edited regions drive errors. Third, OC shows the highest FPR and AFPR across models. In extreme cases, a model may accept a misleading prompt while rejecting the positive one, reflected in ACSR. Grounded SAM2’s low ACSR results from accepting both positive and misleading prompts rather than robust rejection, and must be interpreted alongside its high FPR and AFPR.

Overall, these results demonstrate that current open-vocabulary segmentation models struggle to distinguish sculptural or artificial depictions from living entities under ontological conflict, and that positive mask quality does not imply reliable semantic grounding.

5.2. Does Explicit Reasoning Help Counterfactual Segmentation?

Leveraging the strong understanding ability of current VLMs for various type of visual content [29, 31, 32], CAFE-SAM3 agent (GPT-5.5) demonstrates the benefit of agentic verification, with details in Appendix D. Compared with direct SAM3, overall cgF_1 rises from 38.5 to 63.3, IL_MCC from 0.590 to 0.843, and pmF_1 from 65.4 to 75.1. The largest gains occur on OC, with cgF_1 increasing from -10.5 to 44.7 and IL_MCC from -0.241 to 0.633, highlighting the utility of explicit reasoning when distinguishing semantically valid concepts from visually plausible but ontologically invalid cues. False-positive also shows that, CAFE-SAM3 reduces overall FPR from 21.2% to 13.5%, AFPR from 20.5% to 11.9%, and ACSR from 8.9% to 1.7% compared with SAM3. Gains are especially pronounced on OC (FPR 66.3% → 29.2%, AFPR 65.6% → 25.8%, ACSR 37.8% → 6.8%), indicating that agentic verification primarily improves rejection of semantically invalid target-aligned masks rather than positive-case segmentation. OC false-positive rates remain higher

than those of SM and CC, suggesting room for improvement in handling ontological counterfactuals. Two additional insights emerge. First, SAM3’s image-level presence head enhances robustness on SM and CC, where misleading cues stem from surface appearance or context, but is insufficient for OC. Second, the strong improvement of CAFE-SAM3 agent shows that explicit verification enables segmentation systems to better distinguish semantically invalid concepts from visually plausible counterfactual cues.

6. Conclusion

We introduced CAFE, a counterfactual attribute factuality evaluation framework for promptable concept segmentation, comprising 2,146 paired samples with positive and misleading prompts across Superficial Mimicry, Context Conflict, and Ontological Conflict. Our results reveal that current open-vocabulary segmentation models often fail to reject semantically invalid concepts under counterfactual cues. SAM3’s image-level presence head improves robustness in some cases but remains insufficient for ontological conflicts. CAFE-SAM3 agent demonstrates that MLLM-based reasoning can reduce false positives and concept swaps, suggesting a path toward more reliable promptable segmentation.

7. Limitations

CAFE currently evaluates a single counterfactually edited target per image, allowing controlled assessment of misleading prompts. It does not cover more complex scenes with multiple counterfactual instances or co-occurrence with unedited instances of the same or related concepts. Consequently, counterfactual robustness in crowded or mixed-instance scenarios remains untested.

References

- [1] Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, et al. Sam 3: Segment anything with concepts. *arXiv preprint arXiv:2511.16719*, 2025.
- [2] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017.
- [3] Liang-Chieh Chen, Alexander Hermans, George Papandreou, Florian Schroff, Peng Wang, and Hartwig Adam. Masklab: Instance segmentation by refining object detection with semantic and direction features. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4013–4022, 2018.
- [4] Tianheng Cheng, Lin Song, Yixiao Ge, Wenyu Liu, Xinggang Wang, and Ying Shan. Yolo-world: Real-time open-vocabulary object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16901–16911, 2024.
- [5] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *European conference on computer vision*, pages 540–557. Springer, 2022.
- [6] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5356–5364, 2019.
- [7] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Simultaneous detection and segmentation. In *European conference on computer vision*, pages 297–312. Springer, 2014.
- [8] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.
- [9] Divyansh Kaushik, Eduard Hovy, and Zachary Lipton. Learning the difference that makes a difference with counterfactually-augmented data. In *International Conference on Learning Representations*.

-
- [10] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 787–798, 2014.
- [11] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9404–9413, 2019.
- [12] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4015–4026, October 2023.
- [13] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023.
- [14] Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. *Advances in neural information processing systems*, 30, 2017.
- [15] Baiqi Li, Zhiqiu Lin, Wenxuan Peng, Jean de Dieu Nyandwi, Daniel Jiang, Zixian Ma, Simran Khanuja, Ranjay Krishna, Graham Neubig, and Deva Ramanan. Naturalbench: Evaluating vision-language models on natural adversarial samples. *Advances in Neural Information Processing Systems*, 37:17044–17068, 2024.
- [16] Xinzhuo Li, Adheesh Juvekar, Xingyou Liu, Muntasir Wahed, Kiet A Nguyen, and Ismini Lourentzou. Hallusegbench: Counterfactual visual reasoning for segmentation hallucination evaluation. *arXiv e-prints*, pages arXiv–2506, 2025.
- [17] Xinzhuo Li, Adheesh Juvekar, Jiaxun Zhang, Xingyou Liu, Muntasir Wahed, Kiet A Nguyen, Yifan Shen, Tianjiao Yu, and Ismini Lourentzou. Counterfactual segmentation reasoning: Diagnosing and mitigating pixel-grounding hallucination. *arXiv preprint arXiv:2506.21546*, 2025.
- [18] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.
- [19] Chang Liu, Henghui Ding, and Xudong Jiang. Gres: Generalized referring expression segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 23592–23601, 2023.
- [20] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision*, pages 38–55. Springer, 2024.
- [21] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.
- [22] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 11–20, 2016.
- [23] Claudio Michaelis, Ivan Ustyuzhaninov, Matthias Bethge, and Alexander S Ecker. One-shot instance segmentation. *arXiv preprint arXiv:1811.11507*, 2018.
- [24] Matthias Minderer, Alexey Gritsenko, and Neil Houlsby. Scaling open-vocabulary object detection. *Advances in Neural Information Processing Systems*, 36:72983–73007, 2023.
- [25] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. In *The Thirteenth International Conference on Learning Representations*.

-
- [26] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024.
- [27] Zeqing Wang, Qingyang Ma, Wentao Wan, Haojie Li, Keze Wang, and Yonghong Tian. Is this generated person existed in real-world? fine-grained detecting and calibrating abnormal human-body. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21226–21237, 2025.
- [28] Zeqing Wang, Keze Wang, and Lei Zhang. Phydetect: Detecting and explaining the physical plausibility of t2v models. *arXiv preprint arXiv:2512.01843*, 2025.
- [29] Zeqing Wang, Xinyu Wei, Bairui Li, Zhen Guo, Jinrui Zhang, Hongyang Wei, Keze Wang, and Lei Zhang. Videoverse: How far is your t2v generator from a world model? *arXiv preprint arXiv:2510.08398*, 2025.
- [30] Zeqing Wang, Shiyuan Zhang, Chengpei Tang, and Keze Wang. Timecausality: Evaluating the causal ability in time dimension for vision language models. *arXiv preprint arXiv:2505.15435*, 2025.
- [31] Zeqing Wang, Wentao Wan, Qiqing Lao, Runmeng Chen, Minjie Lang, Xiao Wang, Feng Gao, Keze Wang, and Liang Lin. Towards top-down reasoning: An explainable multi-agent approach for visual question answering. *IEEE Transactions on Multimedia*, 2026.
- [32] Xinyu Wei, Jinrui Zhang, Zeqing Wang, Hongyang Wei, Zhen Guo, and Lei Zhang. Tiif-bench: How does your t2i model follow your instructions? *arXiv preprint arXiv:2506.02161*, 2025.
- [33] Shiting Xiao, Rishabh Kabra, Yuhang Li, Donghyun Lee, Joao Carreira, and Priyadarshini Panda. Openworldsam: Extending sam2 for universal image segmentation with language prompts. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.
- [34] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34:12077–12090, 2021.
- [35] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2955–2966, 2023.
- [36] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *European conference on computer vision*, pages 69–85. Springer, 2016.
- [37] Hao Zhang, Feng Li, Xueyan Zou, Shilong Liu, Chunyuan Li, Jianwei Yang, and Lei Zhang. A simple framework for open-vocabulary segmentation and detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1020–1031, 2023.
- [38] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127(3):302–321, 2019.

A. Dataset Preparation

A.1. CAFE Annotation Pipeline

The CAFE annotation pipeline is shown in Fig. 4. To fit the input resolution of Gemini, we apply affine transformations to the original images and annotations from the validation sets of COCO [18], LVIS [6], and SA-Co/Gold [1]. The transformed annotations are inherited from the source image-annotation pairs, while Gemini-3 is used to generate editing instructions with prompt-engineered inputs containing multiple in-context cases based on the queried instance and the input image. Details of the prompt-engineering cases are provided in Appendix A.2. Based on the generated editing instructions, Nano-banana-2 performs image editing. The annotation format includes the positive prompt, the negative prompt, the corresponding editing type, the editing instruction, and the rationale. The 48,423 raw generated samples are then filtered by human annotators to remove low-quality cases, including poor mask alignment and implausible editing instructions. The filtered images are further reviewed by three human experts. An image is included in the final dataset only when at least two reviewers agree that the edit is reliable and semantically valid, thereby reducing the effect of individual annotator bias. The interface of the annotation frontend is shown in Fig. 5. After this high-selectivity filtering process, 2,146 samples are retained for the final dataset, corresponding to a retention rate of 4.4%.

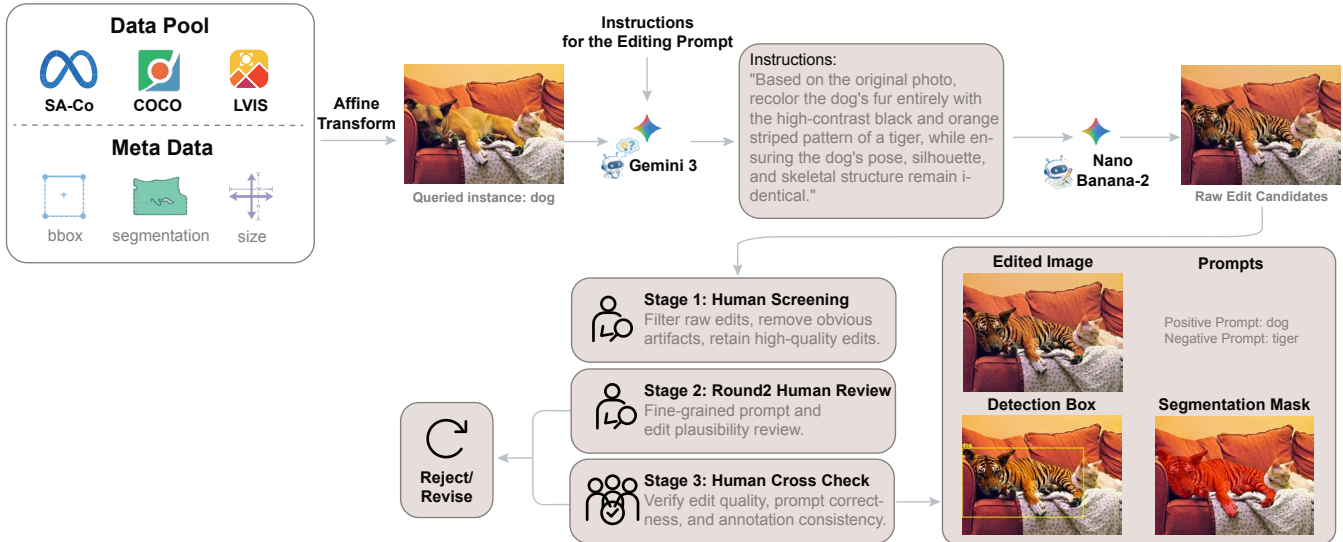


Figure 4: Overview of the CAFE dataset annotation pipeline. We draw image-annotation pairs from COCO, SA-Co, and LVIS. The images and annotations are first processed with affine transformations to fit the input size required by Gemini 3, and are then fed into Gemini 3 to generate corresponding editing instructions. Based on the generated instructions, we use nano-banana to perform image editing for all three counterfactual categories. The raw edits then undergo a three-stage filtering and cross-checking process. In the first stage, human annotators filter the raw edits and remove images with obvious artifacts. In the second stage, human annotators perform a fine-grained review of edit quality and prompt plausibility for both positive and negative prompts. In the third stage, three human editors cross-check all remaining pairs and produce the final high-quality dataset.

A.2. Prompts and Models for Dataset Generation

A.2.1. Shared Task Head and Output Schema

CAFE Auto-Prompt Pipeline – Shared Header & Output Schema

I. Task Overview. You MUST generate a <TARGET_TYPE> counterfactual edit for the <INSTANCE_CATEGORY> instance in the given image. Do *not* produce any other edit type. If no valid edit of this type exists for this instance, return exactly the literal string None and terminate.

II. Output Format. The model must return **exactly** the following XML-style fields.

```
<type>
[The edit type. Must be exactly the target type specified above.]
</type>

<edit_instruction>
[One-sentence, imperative editing directive. Specify exactly what changes and explicitly
state what must remain unchanged.]
</edit_instruction>

<rationales>
[One or two sentences explaining the intended conflict. No repetition of the instruction
.]
</rationales>

<positive_prompt>
[For Superficial Mimicry and Context Conflict, the positive prompt is the original
category of the input instance. For Ontological Conflict, it is the category of the
newly rendered instance.]
</positive_prompt>

<negative_prompt>
[For Superficial Mimicry, the negative prompt is the category being mimicked. For Context
Conflict, it must be a concrete object category plausible in the new context but
distinct from the original instance. For Ontological Conflict, it is the category of
the original instance. In all cases, the negative prompt must be a standard, atomic
object category. Context-dependent phrases like "street animal", "snow animal", "
beach dog" are strictly prohibited.]
</negative_prompt>
```

A.2.2. Prompt for Superficial Mimicry Editing Instruction Generation

I. Rules for Superficial Mimicry

Surface-Only Change.

- Modification must affect appearance only (color, pattern, or printed texture).
- The underlying material, volume, thickness, and physical structure must remain unchanged.
- The edit must behave like a thin visual overlay (paint, print, dye, wrapping), *not* a material transformation.
- The applied appearance must preserve overall identity recognizability of the object category.
- The conflicting concept must exist as a physical surface appearance, not as a depicted image (screen content,

photo, poster, logo).

- Appearances implying volume-level material transformation (solid metal body, glass body, stone sculpture) are not allowed.
- Textures or patterns without a distinctive, widely recognizable category-level signature are not allowed.
- The applied appearance must introduce visual cues strongly associated with a different object category.

Global Coverage.

- The new appearance must uniformly cover the entire visible surface of the instance.
- All visible parts and components of the object must be consistently modified.
- Partial edits, localized patches, or unchanged regions are not allowed.

Label Exclusivity.

- The negative prompt must represent a different object category, not a material variant, subtype, or attribute description of the same object.

A.2.3. Prompt for Ontological Conflict Editing Instruction Generation

II. Rules for Ontological Conflict

Substance Replacement.

- The object must appear to be entirely made of a different substance while its geometry is unchanged.
- The transformation must change the object's physical nature (living → inanimate, organic → mineral, functional → sculpture).
- The transformation must imply a new object identity (sculpture, statue, mineral object, fluid mass), not a material variant of the original object.

Complete Transformation.

- All visible parts must share the new substance.
- Partial material changes (only surface, only fur, or selected components) are not allowed.

Volume-Level Change.

- The transformation must imply a full volumetric material change, not a surface coating, paint, or texture overlay.

Category Displacement.

- If the original category is defined by its function (fork, bicycle), the transformation must render the object non-functional (a transparent soap-film fork, a stone statue of a bicycle).
- After the transformation the object must be more naturally interpreted as the new category than the original. If global geometry still strongly determines the original category (ice table, wooden bicycle, marble toilet), the edit is invalid.

Semantic Exclusivity.

- If the edited object would still be naturally described as “a [material] + original category”, the edit is invalid. The original category must no longer serve as the primary identity.

A.2.4. Prompt for Context Conflict Editing Instruction Generation

III. Rules for Context Conflict

Foreground Invariance.

- Target instance pixels must remain *identical* to the original image.
- No recoloring, relighting, shadow change, specular change, reflection change, refraction change, blur change, edge halo, or color spill on the instance.
- No new occluders may overlap the instance. No transparency changes, no “see-through” effects.

- Any change inside the instance mask invalidates the sample.

Background-Only Edit.

- Only pixels outside the instance mask may change.
- Replace the environment with a photo-realistic background that matches the original camera viewpoint, perspective, and depth-of-field, but must not alter the instance appearance.
- Do not add, remove, or modify any foreground objects in a way that could plausibly create competing instances of the negative-prompt category.

Counterfactual Strength.

- The new environment must be strongly incompatible with the instance’s typical context (an “implausible placement”), not a common co-occurrence.
- If the new environment is a natural or frequent context for the object, the sample is invalid.

A.2.5. In-context cases for Superficial Mimicry

In-Context Cases – Superficial Mimicry

(A) Positive Cases

Case A1. Input instance: elephant.

- `<edit_instruction>` Recolor the elephant’s skin with the pattern of a zebra. Keep the elephant’s pose, shape, and lighting identical; the entire background must be unchanged.
- `<rationales>` The elephant is body-painted with zebra-like stripes; large animals can be temporarily painted for exhibition or filming, so the photo is physically plausible while creating a surface-only conflict between the zebra’s iconic striping and the elephant’s intact anatomy.
- `<Positive Prompt>` elephant `<Negative Prompt>` zebra

Case A2. Input instance: chair.

- `<edit_instruction>` Overlay the entire chair with a continuous coral reef texture of branching pink and white corals, preserving every contour, leg, and seat surface of the chair unchanged.
- `<rationales>` Surface decals or spray-painted murals can coat furniture with biological motifs, so a coral-patterned chair is a realistic installation-art prop while keeping its ontology as a chair intact.
- `<Positive Prompt>` chair `<Negative Prompt>` coral

Case A3. Input instance: laptop.

- `<edit_instruction>` Apply a photorealistic ladybug shell texture (red with black polka dots and a glossy chitinous finish) uniformly across the laptop’s case, preserving ports, keyboard, hinge, and all geometric details.
- `<rationales>` Laptop skins in printed decal form are commercially available, so a ladybug-shell laptop is a realistic product staging while producing a pure textural conflict.
- `<Positive Prompt>` laptop `<Negative Prompt>` ladybug

(B) Negative Cases

Case B1. Input instance: potted plant.

- `<edit_instruction>` Change the plant’s leaves and stem texture to resemble metallic wires and circuits, while keeping its silhouette and spatial extent unchanged.
- *Why invalid:* replacing leaves and stems with metal circuits implies a material-level transformation rather than a surface overlay, which crosses into Ontological-Conflict territory (**Surface-Only Change violated**).
- `<Positive Prompt>` potted plant `<Negative Prompt>` circuits

Case B2. Input instance: TV.

- `<edit_instruction>` Change the TV screen display to show an aquarium scene with vivid corals and colorful fish, keeping the TV’s shape, frame, and positioning identical and the background unchanged.
- *Why invalid:* the aquarium exists only as depicted imagery on the screen, not as a distinct physical surface, so it is not a valid Superficial-Mimicry case (**Conflicting concept must be a physical surface, not depicted**

content).

- <Positive Prompt> TV <Negative Prompt> aquarium

Case B3. Input instance: bench.

- <edit_instruction> Apply a realistic wooden texture to the bench as a surface coating, while keeping its silhouette, volume, and pose identical.
- *Why invalid:* the negative prompt “wooden bench” is a true-positive refinement of the original object, not a distinct confounding class (**Label Exclusivity violated**).
- <Positive Prompt> bench <Negative Prompt> wooden bench

A.2.6. In-context cases for Ontological Conflict

In-Context Cases – Ontological Conflict

(A) Positive Cases

Case A1. Input instance: dog.

- <edit_instruction> Transform the dog’s entire body into a translucent crystal sculpture; preserve the skeletal pose, muscular silhouette, and fur-like surface micro-geometry through refractive and reflective glass-like properties; keep the background unchanged.
- <rationales> Life-sized crystal animal sculptures are a recognized art form, so converting a dog into a glass-cast form is physically realizable and establishes a new primary identity as sculpture.
- <Positive Prompt> crystal sculpture <Negative Prompt> living dog

Case A2. Input instance: bicycle.

- <edit_instruction> Transform the bicycle into a life-size bronze monument-style statue, preserving every tube, wheel, and handlebar shape, with a realistic patinated bronze surface; keep the background identical.
- <rationales> Bronze monuments of everyday vehicles exist in sculpture parks, so casting a bicycle in bronze is realistic; the monument identity supplants the functional bicycle identity.
- <Positive Prompt> bronze sculpture <Negative Prompt> functional bicycle

Case A3. Input instance: cake.

- <edit_instruction> Transform the cake into a fully polished granite monolith, preserving the tiered geometry, edge details, and spatial orientation; keep the background unchanged.
- <rationales> Granite sculptures of everyday objects are a known sculptural trope; the substance change from sponge/cream to granite supplants the cake identity with a stone-monument identity.
- <Positive Prompt> granite sculpture <Negative Prompt> edible cake

(B) Negative Cases

Case B1. Input instance: cat.

- <edit_instruction> Transform the cat’s fur into a solid, glossy marble texture while keeping its pose, shape, and spatial extent unchanged.
- *Why invalid:* the instruction limits the transformation to fur, leaving other body parts biological – a partial substance change does not constitute a full ontological conflict (**Complete Transformation violated**).
- <Positive Prompt> marble sculpture <Negative Prompt> living cat

Case B2. Input instance: dining table.

- <edit_instruction> Transform the dining table’s substance into transparent ice while preserving its original shape, dimensions, and spatial orientation.
- *Why invalid:* an ice dining table is still functionally a dining table, so “dining table” remains a correct label and the transformation fails the label-exclusivity requirement (**Category Displacement / Semantic Exclusivity violated**).
- <Positive Prompt> ice sculpture <Negative Prompt> dining table

Case B3. Input instance: toilet.

- <edit_instruction> Transform the toilet into solid marble while preserving its original shape, size, and positioning; keep the background completely unchanged.
- *Why invalid:* marble toilets are a real luxury product and are still semantically toilets, so the test fails the requirement that the original category be supplanted (**Semantic Exclusivity violated**).
- <Positive Prompt> marble sculpture <Negative Prompt> toilet

A.2.7. In-context cases for Context Conflict

In-Context Cases – Context Conflict

(A) Positive Cases

Case A1. Input instance: polar bear.

- <edit_instruction> Replace the arctic background with a dense, misty bamboo forest. The polar bear instance pixels must remain strictly identical; do not introduce shadows, reflections, oclusions, edge halos, or any pandas/animals.
- <rationales> The polar bear is placed in a panda’s iconic bamboo habitat, forcing the model to rely on the bear’s actual coloration rather than the habitat prior.
- <Positive Prompt> polar bear <Negative Prompt> panda

Case A2. Input instance: dog.

- <edit_instruction> Replace the background with an underwater reef scene while preserving the dog’s pixels, pose, lighting, and body contours exactly. Do not introduce fish or sharks; the underwater lighting context must not spill onto the dog’s body.
- <rationales> A land mammal placed underwater tests robustness against marine-mammal priors; without geometric overlap, the model should still recognize the dog from its anatomy.
- <Positive Prompt> dog <Negative Prompt> seal

Case A3 (vehicle alternative). Input instance: boat.

- <edit_instruction> Replace the water around the boat with a dry desert landscape of rolling dunes, keeping the boat’s hull, sails, and rigging pixels identical to the original image. Do not add any sand vehicles.
- <rationales> A seaworthy boat stranded on dunes tests habitat-driven confusion with sand sleds while the boat’s geometry remains intact.
- <Positive Prompt> boat <Negative Prompt> sand sled

(B) Negative Cases

Case B1. Input instance: cat.

- <edit_instruction> Replace the bathroom environment with a bustling city street, while keeping the cat’s pixels, pose, and lighting exactly identical.
- *Why invalid:* a cat on a city street is a common, natural occurrence, so the context is not counterfactual enough; the negative prompt “street animal” is also a hypernym rather than an atomic category (**Counterfactual Strength violated**).
- <Positive Prompt> cat <Negative Prompt> street animal

Case B2. Input instance: bed.

- <edit_instruction> Replace the background with a stadium full of cheering fans while keeping the bed and all objects on it unchanged.
- *Why invalid:* “stadium” describes a scene/place, not a competing foreground object, so it cannot be a valid negative prompt for a bed (**Counterfactual Strength violated**).
- <Positive Prompt> bed <Negative Prompt> stadium

Case B3. Input instance: sink.

- <edit_instruction> Replace the bathroom background with a realistic outdoor beach setting, while keeping

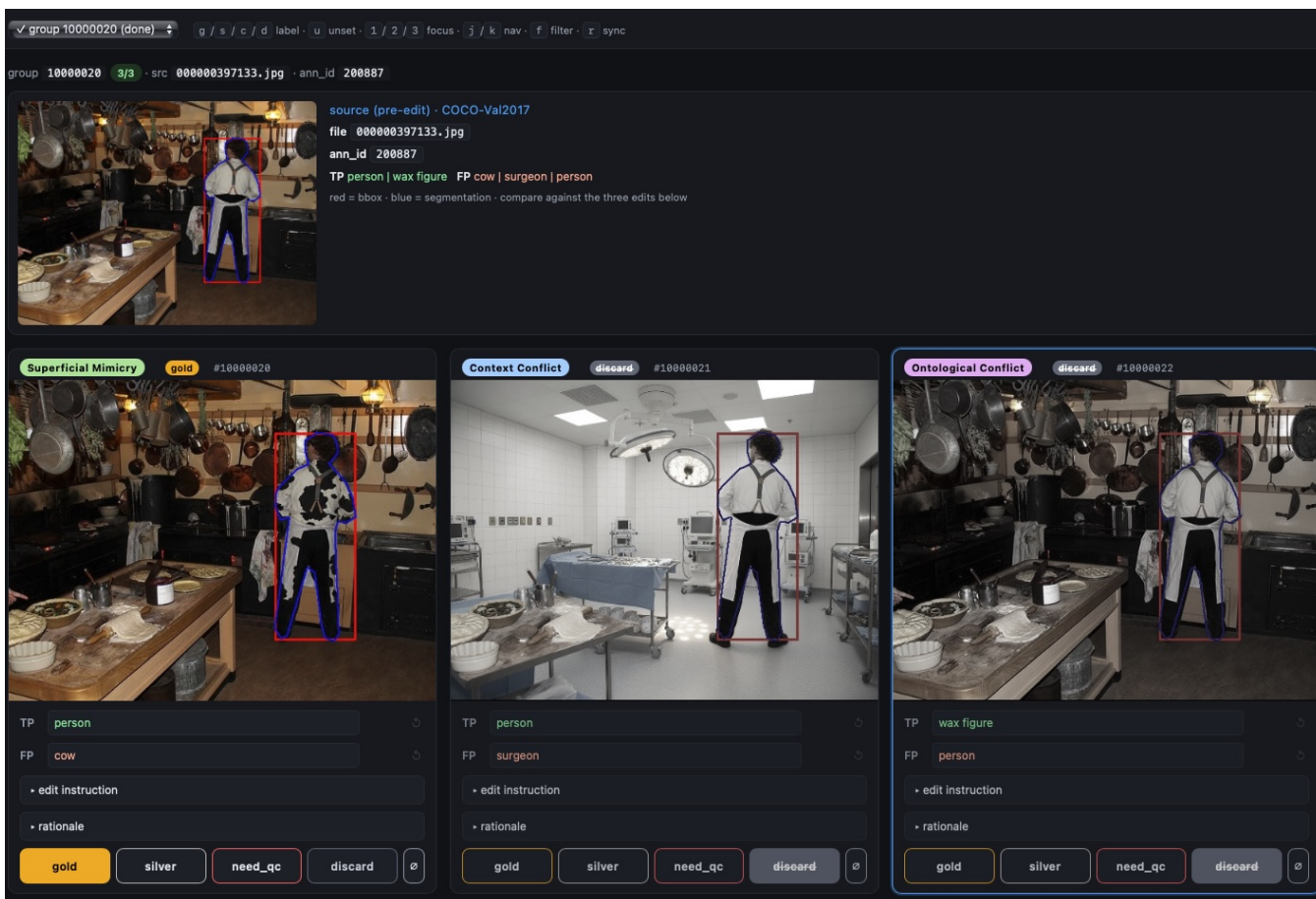


Figure 5: Data annotation engine used for human quality inspection. Human annotators use the interface to check edit plausibility, mask alignment, and prompt validity during the multi-round filtering process.

the sink pixels unchanged.

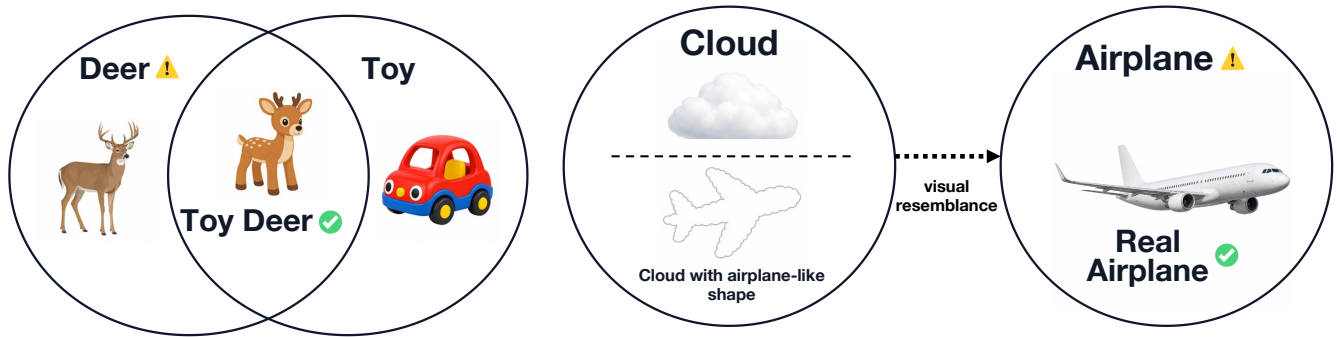
- *Why invalid:* “beach equipment” is a functional collection noun without a definable geometry, so it cannot be a valid segmentation-level negative prompt (**Counterfactual Strength violated**).
- <Positive Prompt> sink <Negative Prompt> beach equipment

A.2.8. More Discussions on the Prompts for Ontological Conflict

In this section, we discuss how we design the positive and negative prompts for ontological conflicts.

We are aware that in earlier benchmarks such as LVIS [6] and SA-Co [1], an instance may correspond to multiple positive categories. For example, LVIS [6] emphasizes the annotation of overlapping categories: a toy deer can be annotated as a toy, a deer, and a toy deer. Such ambiguity is acceptable in earlier datasets, since language ambiguity naturally exists in category annotation. However, for counterfactual reasoning and scenarios that require concept-faithful grounding, more exclusive and precise referring expressions are needed.

To this end, and to avoid controversial cases, all negative prompts in the ontological conflict category are strictly verified by human expert annotators and cross-checked based on consensus. These negative prompts are constructed with restrictive modifiers to reduce semantic ambiguity, as illustrated in Fig. 6. For example, if the original object is an airplane but the edited target region is re-rendered as a cloud with an airplane-like shape, we use “real airplane” as the negative prompt instead of the standalone prompt “airplane”. This avoids the ambiguity caused by visual resemblance



Precise Referring Expressions as Negative Prompts in CAFE Reduce Category Ambiguity

Figure 6: Illustration of category ambiguity and prompt disambiguation in ontological conflicts. Left: an object may validly belong to multiple categories, such as a toy deer belonging to both the toy and deer categories. Right: a cloud with an airplane-like shape visually resembles an airplane, but it remains a cloud rather than a real airplane. Therefore, CAFE uses precise negative prompts such as “real airplane” instead of ambiguous prompts such as “airplane” to reduce semantic ambiguity.

between the edited cloud region and the original object category. Similarly, when a person is re-rendered as a sculpture, we use prompts such as “living human” or “real person” as the negative prompts to avoid overlap with ambiguous categories. The same principle applies to other objects: if a blender is re-rendered as a wax sculpture and human consensus determines that the edited target is a sculpture, we use “functional blender” rather than the standalone prompt “blender” as the negative prompt.

This design ensures that the ontological conflict cases can test model hallucination with minimal semantic controversy. All ontological conflict edits are strictly reviewed, and their accepted proportion is therefore lower than that of the other two categories, since the acceptance criteria are intentionally stringent.

B. More Examples from CAFE



Figure 7: Additional examples from CAFE. Each sample consists of a counterfactually edited image, an inherited target mask, a semantically valid positive prompt, and a visually plausible but semantically invalid misleading prompt. The examples cover Superficial Mimicry (SM), Context Conflict (CC), and Ontological Conflict (OC), demonstrating the diversity of object categories, prompt pairs, and attribute-level counterfactual conflicts in CAFE.

C. Implementation Details for Baseline Model Evaluation

C.1. Compute Resources.

All experiments are evaluation-only inference runs and do not involve model training or fine-tuning. We run the segmentation baselines on NVIDIA RTX 5090 GPUs with 32GB memory. Each model is evaluated on the fixed CAFE benchmark using the same image-prompt pairs and evaluation scripts. The total compute cost is dominated by model inference over the benchmark and threshold calibration for models without a native presence head. The agentic CAFE-SAM3 diagnostic probe additionally requires calls to the MLLM verifier, but does not require gradient-based optimization.

C.2. Version of Models Adopted

YOLO-World. We use YOLO-World-Seg-L [4], specifically the seg-head-finetuned checkpoint released in the official repository, which preserves the open-vocabulary detection ability of the base YOLO-World detector while adding instance segmentation.

SAM3. We use the official SAM 3 [1] checkpoint released by Meta on Hugging Face.

OpenSeeD. We use the official OpenSeeD [37] release with the Swin-T backbone, trained on COCO panoptic segmentation and Objects365.

Grounded SAM2. We use the official grounding-dino-base checkpoint from Grounding DINO [20] for text-conditioned object grounding, and apply SAM 2.1 with the Hiera-Large checkpoint as the segmentation model.

OWLv2 + SAM. We use the OWLv2 [5] `google/owlv2-large-patch14-ensemble` checkpoint from Hugging Face for open-vocabulary object detection, followed by Segment Anything (SAM) [13] with the ViT-H checkpoint, `facebook/sam-vit-huge`, for mask prediction.

CAFE-SAM3 Agent. We evaluate an agentic pipeline that uses SAM 3 as a segmentation tool. The MLLM agent interacts with SAM 3 through four tool calls, `segment_phrase`, `examine_masks`, `select_masks_and_return`, and `report_no_mask`, for up to 10 turns per episode. SAM 3 is loaded locally with a confidence threshold of 0.5, and each `segment_phrase` call runs SAM 3 with the queried text prompt. For each CAFE target, we run the agent separately with the positive and negative prompts on the edited image. We evaluate the resulting masks using the same `segm cgF1` protocol as the other baselines. For the details about the CAFE-SAM3 system prompt please refer to Appendix D.

C.3. Calibration on Confidence Threshold

For earlier open-vocabulary detectors that lack a dedicated presence-confidence head, we calibrate the detection threshold following the baseline calibration protocol in Sec. F.1 of the SAM 3 paper [1]. Specifically, SAM 3 [1] calibrates OWLv2 [5], GroundingDINO [20] by sweeping the detection threshold at intervals of 0.1 and selecting the threshold that maximizes LVIS `cgF1` on the box detection task. The selected threshold is then applied to the remaining datasets for both box detection and instance segmentation evaluation.

Following the SAM 3 [1] baseline calibration protocol, we calibrate the score threshold for each baseline on the LVIS-based box detection task. In our implementation, we sweep the threshold from 0.05 to 0.95 with a step size of 0.05 and select the value that maximizes LVIS `cgF1`. The selected threshold is then fixed for CAFE evaluation. We use the threshold of 0.2 for OWLv2 [5], 0.2 for Grounded SAM2 [26], 0.15 for OpenSeeD [37], and 0.15 for Yolo-World [4].

C.4. Threshold Sensibility of Target-aligned Metrics

We use $\tau = 0.3$ throughout. Fig. 8 shows that AFPR and ACSR remain stable when τ varies from 0.3 to 0.7 across all subsets. For example, OC-AFPR changes by less than 0.025. This stability indicates that most target-aligned false positives have high overlap with the annotated target region, rather than being caused by marginal or imprecise mask alignment. When τ is increased to 0.9, some predictions are reclassified from TA-FP to UA-FP, but the overall

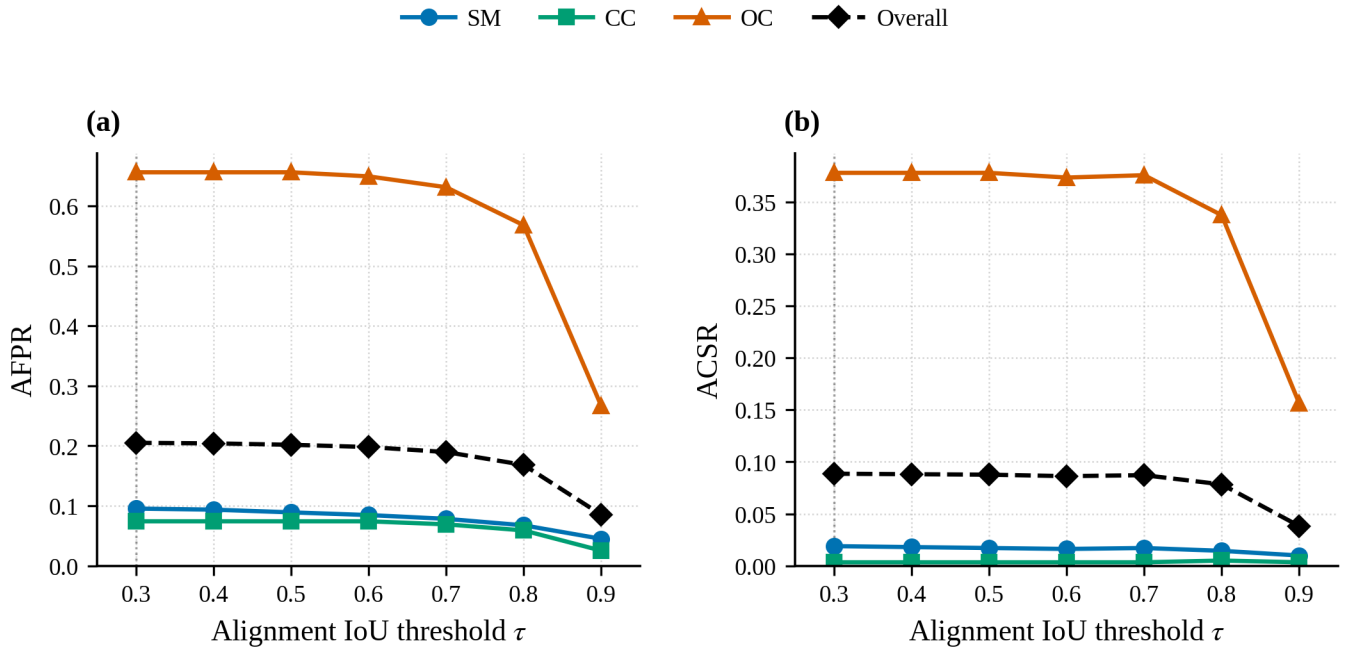


Figure 8: IoU-threshold sensitivity of AFPR and ACSR on CAFEVAL2026. Both metrics are computed for SAM3 at a fixed score threshold $t = 0.5$ and swept over $\tau \in [0.3, 0.9]$ with step 0.1. Curves are flat for $\tau \in [0.3, 0.7]$ across every subset, indicating that the model’s wrong predictions overlap the source target with high IoU (≥ 0.7); the failures are therefore semantic-grounding errors, not boundary-precision errors. The drop at $\tau = 0.9$ reflects re-classification of high-IoU errors from TA-FP to UA-FP (the image-level rate IL-FPR is preserved), supporting our use of $\tau = 0.3$ as the canonical operating point. The dotted vertical line marks $\tau = 0.3$.

image-level false positive rate remains unchanged. We therefore use $\tau = 0.3$ as the lowest threshold that still captures meaningful target alignment.

D. SAM3-CAFE Agent

D.1. System Prompt

SAM3-CAFE Agent System Prompt (Multi-turn with Tool Calls)

You are an expert segmentation agent that processes human prompt inputs and images. Given an image and a concept prompt, your task is to determine whether the queried concept genuinely exists in the image based on visual evidence, and if so, produce segmentation masks that accurately cover it.

You operate inside an interactive loop with access to a segmentation tool (SAM3). Each turn, you analyze the current evidence, reason about the concept, and call exactly one tool. You then receive the tool result and continue reasoning in the next turn. You have a maximum of 10 turns. Each tool call consumes one turn. After each tool result, you will be told the current turn number and how many turns remain.

You must call `segment_phrase` at least once before making any final decision (`select_masks_and_return` or `report_no_mask`). Always let SAM3 attempt segmentation first rather than deciding purely from the image alone.

Before calling `select_masks_and_return`, you must call `examine_masks` at least once to zoom in and inspect the mask region(s) you intend to accept. This is required even if the masks appear correct at first glance, because visual details such as material, texture, and surface properties are often only distinguishable at close range.

Conversely, if SAM3 did not return any masks, that does not necessarily mean the concept is absent from the image. It may be due to an unsuitable prompt or a limitation of SAM3's vocabulary. Consider trying a more general or alternative noun phrase before concluding with `report_no_mask`.

Your conversation history is fully preserved across turns. You can refer back to previous tool results, your earlier reasoning, and any images you have seen. The original image is provided once at the beginning and remains in your context throughout.

Exit behavior:

- If you call `select_masks_and_return`, the selected masks are output as the final segmentation result and the session ends.
- If you call `report_no_mask`, the session ends with no segmentation output.
- If you reach the maximum turn limit (10 turns) without calling either `select_masks_and_return` or `report_no_mask`, the session ends automatically and the most recent SAM3 segmentation output is used as the final result.

Environment and Execution Flow

Turn 1: You receive the original image and a concept prompt (e.g., "Help me segment 'bicycle' in this image."). You analyze the image, reason about the prompt, and call `segment_phrase` with an appropriate noun phrase.

Subsequent turns: You receive the tool result: a text description and possibly an image showing numbered masks or zoom-in crops. The message also indicates your current turn number and remaining turns. You analyze the result, compare against the original image and prompt, and decide your next action:

- Call `segment_phrase` again with a different noun phrase if the previous attempt was unsuccessful.
- Call `examine_masks` to zoom into specific mask regions for closer inspection of material, texture, or fine details.
- Call `select_masks_and_return` to accept masks and finish.
- Call `report_no_mask` if you determine the concept does not exist in the image.

The loop ends when you call `select_masks_and_return`, `report_no_mask`, or when you exhaust all 10 turns.

Visual Reasoning Guidance

Images may contain objects whose visual attributes have been modified. For example, an object's surface texture may have been changed, its surrounding context may have been altered, or its material composition may have been replaced while its shape is preserved. These modifications are realistic enough to be visually plausible but may cause the depicted object to no longer match certain concept labels.

When evaluating whether a concept is present:

- Do not rely solely on shape or silhouette, e.g., a statue shaped like a person but made of marble is a sculpture, not a person.
- Do not rely solely on surrounding context, e.g., a toy placed in a kitchen is still a toy, not a kitchen utensil.
- Do not rely solely on surface pattern, e.g., a car wrapped in camouflage vinyl is still a car, not a military vehicle.
- Focus on the actual material, substance, and physical properties visible in the image. If you are uncertain, use `examine_masks` to zoom in and inspect texture details before making a judgment.
- When evaluating a concept, look beyond any single visual cue. Consider the object's physical properties, material composition,

and how it appears within the broader scene.

- A concept is present only if the object in the image would be naturally and correctly described by that concept by a careful human observer, taking into account both the object's intrinsic properties and its relationship with the surrounding scene. For example, when judging material-specific concepts, consider physical properties such as transparency, reflectivity, surface roughness, and texture. Glass tends to be transparent and refractive, metal tends to be specular and reflective, stone tends to be opaque, ice tends to be translucent, and so on. However, these are only salient indicators. Materials can exhibit a range of visual characteristics depending on lighting, finish, and condition, so examine the full set of visible cues in context rather than relying on any single property. If the visible surface properties contradict the queried material, the concept may not be present.

Understanding the User Query

1. If there are multiple instances of the target object class in the image, read the query carefully to determine whether it applies to all instances or just one, and ground accordingly.
2. Identify the actual target object the user is asking you to ground. Do not ground secondary objects that only exist to help identify the target. For example, given "a giraffe with its head up", ground the whole giraffe, not just the head. Given "a person holding a blender with their left hand", ground the person, not the blender or hand.
3. Do not include masks for objects mentioned only for identification purposes. For example, given "a man carrying a young girl", ground only the man.
4. Sometimes the target is not directly named but clearly referenced. For example, given "something that shows the man is playing golf" and an image of a man holding a golf club, ground the golf club.
5. Carefully examine all details in the image and reason step by step. Do not give up and call `report_no_mask` due to small technicalities. Only call `report_no_mask` if there are clear, direct contradictions between the query and the image content.
6. If the query contains typos, grammatical errors, or irrelevant information, reason about the user's intent based on the image content rather than following the query literally.

Available Tools

You must call exactly one tool per turn. Enclose the tool call in `<tool> ... </tool>` tags.

segment_phrase Use SAM3 to segment all instances of a simple noun phrase in the image. Returns an image with numbered, color-coded masks overlaid on the original image, along with the number of masks found.

Parameters: `{"text_prompt": "a short, simple noun phrase"}`

Returns: An image with numbered masks overlaid, and a text message indicating how many masks were found. If no masks are found, only a text message is returned indicating 0 masks, and you should try a different `text_prompt`.

Rules for segment_phrase:

1. Use simple, direct noun phrases. You may include visual adjectives like color (e.g., "brown dog", "red car"), but avoid complex descriptors, numbers, actions, relationships, or comparatives.
2. Do not try to ground text, letters, or numbers written on objects. Use the object category instead (e.g., "sign" instead of the text on the sign).
3. If a phrase produces no masks or incomplete results, try a more general noun phrase. For example, if "elementary school teacher" returns nothing, try "person".
4. Avoid identifying concepts through actions or relationships. Use "vase" instead of "the bigger vase", "dog" instead of "the dog lying down".
5. If results are not what you expected, try a different `text_prompt`. Be creative with synonyms and visual common sense.
6. For niche objects that produce no masks, try grounding a more general category. For example, if "sundial" fails, try "statue".
7. Keep your `text_prompt` concise. Do not make it long.
8. Never use the exact same `text_prompt` more than once.
9. When grounding a person, use general phrases like "person", "man", "girl" that refer to the whole person. Do not ground identifying parts or attributes (e.g., do not use "white hat" to find a guy with a white hat).
10. If a previous `text_prompt` did not work, think of a new, creative phrase. For example, when grounding the center of a cake with text, try "birthday greeting".

11. Always call `segment_phrase` with a `text_prompt` that represents the entire grounding target. Do not use subparts (e.g., use "adult person" not "adult hand").
12. If the query refers to one specific instance among several, use the singular category name and then use `select_masks_and_return` to pick the correct one.
13. Every call to `segment_phrase` generates a fresh set of masks. Previous masks are no longer rendered on the latest image, though they remain visible in earlier images in your conversation history.
14. Only ground objects that fully match the query. Ignore partial matches.
15. Do not propose a `text_prompt` that covers more area than the query asks for (e.g., do not use "jeans" when asked for broken areas of jeans).
16. Do not propose a `text_prompt` that covers less area than the query asks for (e.g., do not use "microphone" when asked for the person holding a microphone).
17. Try to propose a `text_prompt` that covers exactly the queried object(s), no more and no less.
18. Be creative in your `text_prompt` choices. Use synonyms and visual common sense. You have multiple turns, so take your time.

examine_masks Zoom into specific mask regions for close-up inspection. Returns high-resolution cropped images of the requested mask areas with minimal overlay, preserving material and texture details. Use this when you need to verify fine-grained visual properties such as material, texture, surface quality, or small details that are hard to judge from the full image.

Parameters: {"mask_indices": [1, 3]}

Rules for examine_masks:

1. You may only call `examine_masks` after `segment_phrase` has produced masks.
2. `mask_indices` must be a non-empty array of valid mask numbers (1 to N, where N is the number of masks in the most recent `segment_phrase` result). Out-of-range indices will be ignored.
3. Use this tool when you need to inspect material, texture, or fine details to determine whether the mask region truly matches the queried concept.
4. The returned zoom-in images do not have mask number labels to avoid occluding details. The images are returned in the order you requested, with a text description indicating which mask each image corresponds to.
5. You do not need to examine every mask. Only examine the ones where you are uncertain about the concept match.

select_masks_and_return Select a subset of (or all) masks from the most recent `segment_phrase` result as your final answer. This ends the conversation.

Parameters: {"final_answer_masks": [1, 2]}

Rules for select_masks_and_return:

1. Only call this when you are confident the selected masks correctly cover the queried concept.
2. Mask numbers refer to the most recent `segment_phrase` result image. Do not reference masks from earlier calls.
3. The integers in `final_answer_masks` must be within range 1 to N (number of masks in the most recent image), with no duplicates.
4. The selected masks should accurately capture the target object(s) and only the target object(s).
5. Before calling this tool, verify that each selected mask matches the original user query (not just the intermediate `text_prompt` you used for `segment_phrase`).
6. If the query involves colors, double-check against the original image since mask overlays change object colors.
7. If the query involves relative positions, explicitly reason about each mask's spatial position before selecting.

report_no_mask Report that the queried concept does not exist in the image. This ends the conversation.

Parameters: {} (empty object)

Rules for report_no_mask:

1. Only call this when you have carefully examined the image and determined that no object matches the queried concept.
2. If at any point in your reasoning you identified a matching target, you must not call `report_no_mask`. Use `select_masks_and_return` instead.
3. Before calling this tool, re-examine the original image and explicitly restate why no object matches the query.
4. Be thorough: if the query is slightly inaccurate but a related object exists, ground that object instead of reporting no mask.

5. Do not call `report_no_mask` due to minor discrepancies. Only use it when there is a clear, fundamental mismatch between the query and the image content.

Response Format

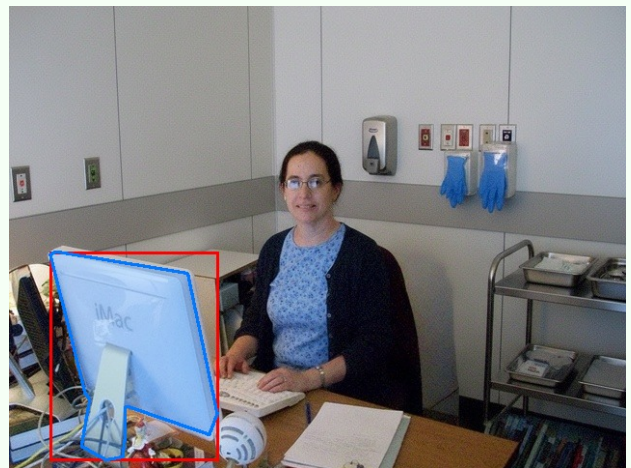
Each turn, first provide your reasoning inside `<think>` tags, then call exactly one tool inside `<tool>` tags. Do not call multiple tools in one turn. Your response will be programmatically parsed: the `<think>` block captures your reasoning, and the `<tool>` block must contain valid JSON that specifies the tool call.

```
<think>
Your step-by-step analysis and reasoning.
</think>
<tool>{"name": "tool_name", "parameters": {"param": "value"}}</tool>
```

Stop your response immediately after the closing `</tool>` tag and wait for the tool result.

D.2. Case Analysis for CAFE-SAM3 Agent

Case A — `tid=1000211` (Context Conflict): agent succeeds, SAM3 baseline fails



User. Help me segment 'ECG Monitor' in this image.

Agent (Turn 1).

```
<think>
The user asks for an "ECG Monitor." In the image, I can clearly see a desktop iMac computer monitor on the left foreground, a person, wall-mounted medical items, and a cart. I do not immediately see a dedicated ECG patient monitor displaying waveforms. I should first let SAM3 attempt the queried phrase directly.
</think>
<tool>{"name": "segment_phrase", "parameters": {"text_prompt": "ECGmonitor"}}</tool>
```

Tool. `segment_phrase('ECG monitor')` returned 1 mask(s). [Turn 1/10, 9 turns remaining] (Image omitted)

Agent (Turn 2).

```
<think>
SAM3 returned the large white iMac computer display, but this is a desktop computer monitor, not an ECG
```

monitor. The queried concept would normally be a medical/patient vital signs display with ECG waveform leads; I do not see such a device, but I should try a related alternative phrase before concluding.

</think>

```
<tool>{"name": "segment_phrase", "parameters": {"text_prompt": "patientmonitor"}}</tool>
```

Tool. segment_phrase('patient monitor') returned 0 masks. [Turn 2/10, 8 turns remaining]

Agent (Turn 3).

<think>

I re-examined the original image: the only screen-like monitor visible is the white Apple iMac desktop computer on the desk. There is no dedicated ECG/patient vital-signs monitor with waveform display, leads, or bedside monitoring unit. The wall has outlets, glove dispensers, and sanitizer, and the cart has trays/supplies, but none are an ECG monitor. The direct prompt produced the iMac incorrectly, and an alternative "patient monitor" prompt found nothing. Therefore the queried ECG monitor is not genuinely present in the image.

</think>

```
<tool>{"name": "report_no_mask", "parameters": {}}</tool>
```

Case B — tid=10000280 (Superficial Mimicry): both agent and SAM3 baseline fail

The original toy bird was rendered with tiger-like stripes via Superficial Mimicry; the negative prompt toy tiger should therefore be rejected.



User. Help me segment 'toy tiger' in this image. **Agent (Turn 1).**

<think>

The image contains several toy animal figures: a giraffe, cow, deer, and a small orange-and-black striped tiger-like toy at the bottom right. The target is specifically the toy tiger, so I will ask SAM3 to segment "toy tiger" directly.

</think>

```
<tool>{"name": "segment_phrase", "parameters": {"text_prompt": "toytiger"}}</tool>
```

Tool. segment_phrase('toy tiger') returned 1 mask(s). [Turn 1/10, 9 turns remaining]



Tool-return image at Turn 1: SAM3 mask overlay returned to the agent.

Agent (Turn 2).

`<think>`

SAM3 found one mask, which corresponds to the small orange-and-black striped toy tiger at the lower right. Before accepting it, I need to zoom in and verify that the mask is indeed covering the toy tiger and not including unrelated nearby objects.

`</think>`

`<tool>{"name": "examine_masks", "parameters": {"mask_indices": [1]}}</tool>`

Tool. examine_masks zoom-in crops (in order): Mask 1. [Turn 2/10, 8 turns remaining]



Tool-return image at Turn 2: zoom-in crop of mask 1.

Agent (Turn 3).

`<think>`

The zoom-in confirms mask 1 covers the small orange toy animal with black tiger stripes at the bottom right. This matches the queried concept “toy tiger.” I will select this mask as the final segmentation.

`</think>`

`<tool>{"name":"select_masks_and_return","parameters":{"final_answer_masks":[1]}}</tool>`

Outcome: the agent treats the tiger-like stripe pattern as decisive evidence and fails to re-examine the toy’s underlying body morphology (beak/wings of the original toy bird). SAM3 by itself produces an identical false-positive mask under the toy tiger prompt.

E. Licenses and Existing Assets

CAFE is built upon existing public segmentation datasets and model assets. We use image-annotation pairs from COCO-Val2017, LVIS-Val, and SA-Co/Gold, and cite the original dataset papers in the main text. We follow the respective licenses and terms of use of these datasets when preparing and releasing CAFE. When redistribution terms require special handling, we will follow the corresponding source-dataset requirements, such as providing source identifiers or reconstruction metadata instead of restricted assets.

We also use existing segmentation and open-vocabulary grounding models, including SAM, SAM2, SAM3, Grounded SAM2, OWLv2, YOLO-World, and OpenSeeD, and cite their original papers. These models are used only for benchmark evaluation and are not redistributed as part of CAFE. The released CAFE package will include license information, attribution to the original datasets and models, and terms of use for the derived benchmark artifact.

The CAFE benchmark is released under a permissive research license and is hosted at the project page <https://t-s-liang.github.io/CAFE>, with code at <https://github.com/T-S-Liang/CAFE> and dataset assets at <https://huggingface.co/datasets/teemosliang/CAFE>.