Not All Attributes Compose Equally: Family-Aware Structured Calibration for Compositional Zero-Shot Learning

Anonymous ACL submission

Abstract

Compositional zero-shot learning (CZSL) aims to recognize unseen attribute-object compositions by recombining known primitives. Methods built on CLIP improve CZSL via prompt learning, branch alignment, logic constraints, and visual adaptation, yet usually treat attributes as a flat vocabulary. This leaves regularities across semantic families underused: color and material often depend on local surface cues, whereas state, pose, interaction, and affect may require evidence from the object or the scene. We propose FASC (Family-Aware Structured Calibration), a compact calibrator for pair scores that uses attribute families as intermediate structure while preserving the original candidate pair prediction space. FASC averages source logits selected by validation, obtains family evidence for each image from source family posteriors or by marginalizing pair posteriors, and learns structured corrections from labeled training pairs. Across C-GQA, MIT-States, and UT-Zappos, FASC achieves the best performance on most reported metrics, showing that evidence at the family level provides a lightweight calibration signal. Code is available at https://anonymous.4open.science/r/FASC-05CB.

1 Introduction

Humans can recognize unfamiliar visual compositions by recombining known concepts, such as inferring a red tomato or a sliced apple from red apples and sliced tomatoes. Compositional zero-shot learning (CZSL) formalizes this ability by training on a subset of attribute-object pairs and testing on novel compositions of known primitives (Misra et al., 2017; Nagarajan and Grauman, 2018; Purushwalkam et al., 2019).

Large vision-language models (VLMs), especially CLIP (Radford et al., 2021), provide a strong foundation for CZSL because candidate labels can be expressed as natural language. Recent CZSL methods improve pair scoring with prompt learning and decomposition (Nayak et al., 2023; Lu et al., 2023), branch alignment (Huang et al., 2024), semantic relations constrained by logic (Wu et al., 2025b), or visual adaptation (Zhang et al., 2025). However, after pair logits are produced, attributes are often treated as flat primitives, leaving broader regularities underused for final calibration.

Attribute evidence has coarse semantic regularities: color and material cues tend to be local surface evidence (Christoff and Tonchev, 2024; Balestra et al., 2025), whereas physical state, pose, interaction, and facial affect often require object geometry or context (Stavrothanasopoulos et al., 2025; Yongsiriwit et al., 2025). Families can therefore calibrate heterogeneous candidate pairs (Kim and Choi, 2025), but should not replace scoring at the pair level, since *open* can look different for a door, laptop, or mouth. We use families as structured calibration over pair scores, not as standalone visual templates.

Figure 1 illustrates this motivation. The upper part shows why attribute families are useful but should not be treated as rigid templates: the same family may manifest differently across object categories, while attributes from the same family can share more consistent evidence for a fixed object. The lower part shows the resulting scoring ambiguity. Flat pair scores can confuse candidates with similar logits, whereas marginalization at the family level exposes a stronger cue for the image and allows FASC to calibrate the ranking (Fan et al., 2022; Kim and Choi, 2025).

Following this calibration view, we propose FASC (Family-Aware Structured Calibration) for CZSL candidate pair logits. FASC preserves the original prediction space: a CZSL backbone produces pair logits, source checkpoints selected by validation are uniformly averaged, and a compact calibrator reranks the same candidates. Families enter only as intermediate calibration structure. Given

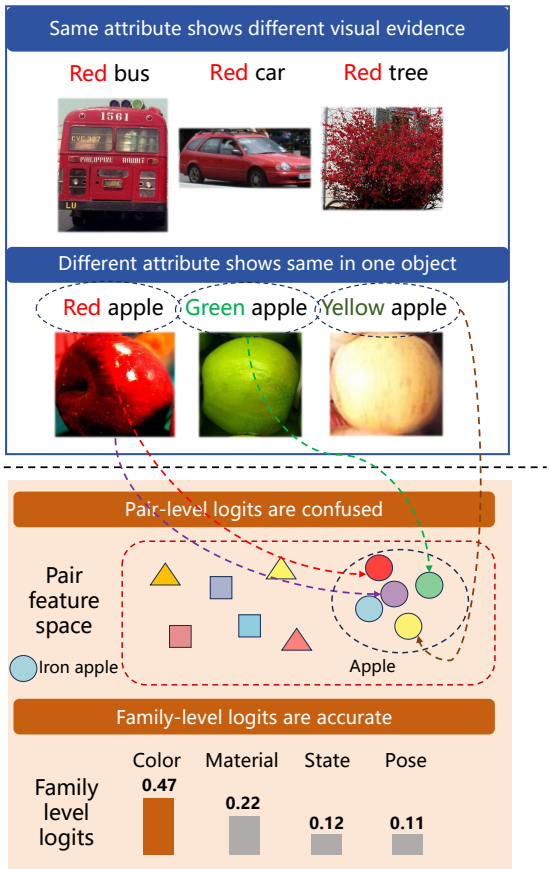

Figure 1

a fixed semantic family taxonomy, FASC obtains family evidence for each image from available source family posteriors or, when unavailable, by marginalizing pair posterior mass over the families of candidate attributes (Balestra et al., 2025; Kim and Choi, 2025). It then calibrates pair scores by combining the source pair logits with lightweight structural priors and corrections conditioned on families (Fan et al., 2022; Liu et al., 2024). The calibrator is fitted on labeled training pairs with a ranking objective, aligning corrections with generalized CZSL ranking rather than family classification.

Our contributions are threefold:

(1) We identify semantic attribute family structure as an underused prior for CZSL pair logit calibration while preserving the original candidate pair prediction space.

(2) We propose FASC, a compact calibrator that retains pair scoring through source logit averaging selected by validation and uses fixed families only as intermediate calibration structure.

(3) We train the calibrator with structured priors, family evidence, and a pseudo unseen ranking objective to improve calibrated candidate pair ranking.

Experiments on C-GQA, MIT-States, and UT-Zappos demonstrate improved calibrated pair ranking, supporting the value of evidence at the family level for pair score calibration.

2 Related Work

CZSL studies how to recognize unseen attribute-object pairs by composing known primitives. Early work composes classifiers with attribute transformations (Misra et al., 2017), models attributes as operators on object representations (Nagarajan and Grauman, 2018), assembles modular classifiers for new compositions (Purushwalkam et al., 2019), or regularizes attribute-object transformations with symmetry (Li et al., 2020). Subsequent methods introduce graph propagation among attributes, objects, and compositions (Naeem et al., 2021), open world scoring with feasibility estimates (Mancini et al., 2021), and primitive modeling guided by knowledge (Karthik et al., 2022). Recent work further shows that attributes should not always be treated as fixed visual tokens: CANet learns attributes conditioned on the image and object (Wang et al., 2023), CDS-CZSL studies attribute specificity across object contexts (Li et al., 2024), and CPF decomposes composition probability into object likelihood and conditional attribute likelihood (Wu et al., 2025a).

Vision-language pretraining has shifted CZSL toward composition scoring with text prompts. CLIP (Radford et al., 2021) enables candidate pairs to be represented as prompts written in natural language, while CSP (Nayak et al., 2023), DFSP (Lu et al., 2023), and ADE (Hao et al., 2023) adapt prompt or concept representations for compositional recognition. Troika jointly models state, object, and composition branches with visual adaptation (Huang et al., 2024); LOGICZSL injects semantic relations among primitives and compositions through objectives induced by logic (Wu et al., 2025b); and recent visual proxy methods further align textual prototypes with composition evidence on the image side (Zhang et al., 2025).

These methods mainly improve representations or scoring before candidate pair logits are finalized. Our work is complementary: FASC uses a fixed semantic attribute family taxonomy to calibrate such logits while keeping the original candidate pair prediction space unchanged.

3 Method

FASC is a structured calibrator that uses family information for CZSL candidate pair logits, as shown in Fig. 2. Given an image, a CZSL backbone first scores all candidate attribute-object pairs. FASC keeps this pair score as the base signal, while a fixed taxonomy maps each candidate attribute to a semantic family. It then derives family evidence for each image from source family posteriors or by marginalizing pair posteriors, and combines this evidence with training frequency priors over pairs, objects, attributes, and families. The resulting lightweight calibrator rescores the same candidate pairs, so the prediction space is unchanged while the final seen/unseen ranking is adjusted by family structure.

3.1 Problem Formulation

Let $\mathcal{A}$ be the attribute set, $\mathcal{O}$ the object set, and $\mathcal{C} \subseteq \mathcal{A} \times \mathcal{O}$ the candidate pair set. In closed world evaluation, $\mathcal{C}$ is the benchmark candidate composition set; in open world evaluation, $\mathcal{C}$ can be the full Cartesian product, optionally constrained by the benchmark feasibility protocol. Each image is labeled with an attribute-object pair $c^\star = (a^\star, o^\star)$. The final prediction ranks all candidate pairs:

$$\hat{c} = \arg\max_{c \in \mathcal{C}} S(x, c). \tag{1}$$

FASC does not change this prediction space. Attribute families are used only as intermediate structure for calibration.

3.2 Source Logits and Attribute Families

Let $s_m(x, c)$ be the logit from source checkpoint m for image x and candidate pair $c = (a, o)$. When a pool of checkpoints or seeds is available, we select a subset $\mathcal{S}$ by validation performance and combine the selected sources with uniform averaging:

$$\bar{s}(x, c) = \frac{1}{|\mathcal{S}|} \sum_{m \in \mathcal{S}} s_m(x, c). \tag{2}$$

Let $\mathcal{F} = \{1, \ldots, K\}$ be a fixed set of semantic attribute families, and let

$$\tau : \mathcal{A} \to \mathcal{F} \tag{3}$$

map each attribute to its family. We construct this taxonomy offline with GPT-4o (OpenAI, 2024): given the dataset attribute vocabulary, the model assigns each attribute to a concise semantic family, after which the names are canonicalized and the mapping is fixed. The taxonomy is not a prediction target; it defines structured statistics and family evidence for calibrating pair scores.

3.3 Family Evidence

FASC uses family evidence in two places. First, a selected source may provide family posteriors for each sample. We denote the averaged main and auxiliary family posteriors by $q^m(f \mid x)$ and $q^a(f \mid x)$. If such posteriors are unavailable, they are obtained from the pair logits by marginalization. Specifically, from the averaged logits we compute

$$p(c \mid x) = \frac{\exp(\bar{s}(x, c))}{\sum_{c' \in \mathcal{C}} \exp(\bar{s}(x, c'))}, \tag{4}$$

and the family posterior from pair marginalization is

$$q^p(f \mid x) = \sum_{(a,o) \in \mathcal{C} : \tau(a) = f} p((a, o) \mid x). \tag{5}$$

Equivalently, if $M_{cf} = \mathbb{I}[\tau(a_c) = f]$, then $\mathbf{q}^p(x) = \mathbf{M}^\top \mathbf{p}(x)$. This posterior is also used to form a family prior correction for each sample. The marginalization step is important because it does not force a hard family decision. Instead, it summarizes how much probability mass the current image assigns to each attribute family after considering all candidate objects. Thus, family evidence can be informative even when the top pair is ambiguous: several uncertain pairs may still agree that the relevant visual cue is, for example, color, material, or physical state.

3.4 Family-Aware Structured Calibration

Given $\bar{s}(x, c)$ and the family evidence above, FASC starts from the backbone score and adds several lightweight calibration terms. For a candidate pair $c = (a, o)$, $\bar{s}(x, c)$ is the backbone evidence for this exact attribute-object composition. The prior term adjusts the score according to training set statistics, including how often the pair, the attribute, the object, the attribute family, and the relation between family and object appear in training. The family term uses the family posterior for each image to adjust candidates whose attributes belong to highly ranked families for the current image. The final family prior term checks whether the image assigns more evidence to the candidate's family than expected from the training distribution. These additions only rescore the existing candidate

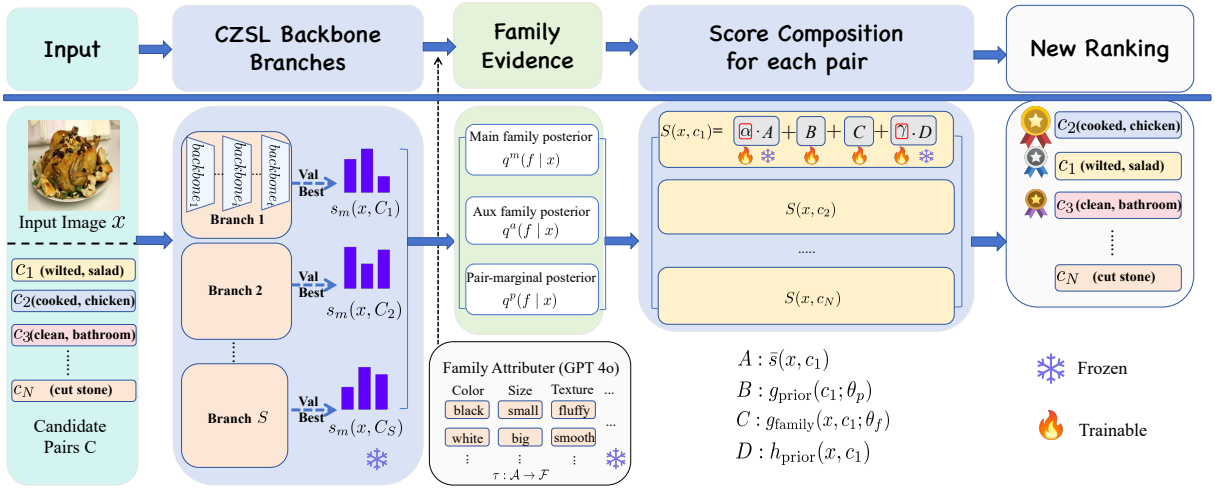

Figure 2: Overview of FASC. A CZSL backbone first produces candidate pair logits over attribute-object compositions. FASC maps attributes to semantic families, obtains family evidence for each image from source posteriors or by marginalizing pair posteriors, combines it with training frequency structural priors, and learns a lightweight calibrator that rescores the same candidate pairs for final ranking.

pairs: FASC does not create new labels or change the candidate set. Formally, FASC computes

$$S(x, c) = \alpha\, \bar{s}(x, c) + g_{\text{prior}}(c; \theta_p) + g_{\text{family}}(x, c; \theta_f) + \gamma\, h_{\text{prior}}(x, c). \tag{6}$$

Here, α is a learnable base logit scale, g_{prior} is a training frequency structural prior, g_{family} is a ranked family bias, and h_{prior} is a family prior correction for each sample.

Training frequency prior. Let $\hat{P}_{\text{tr}}$ denote smoothed empirical frequencies from the training split. For $c = (a, o)$, we use five statistics:

$$\begin{aligned} \ell_c^P &= \log \hat{P}_{\text{tr}}(a, o), & \ell_c^A &= \log \hat{P}_{\text{tr}}(a), \\ \ell_c^O &= \log \hat{P}_{\text{tr}}(o), & \ell_c^F &= \log \hat{P}_{\text{tr}}(\tau(a)), \\ \ell_c^{FO} &= \log \hat{P}_{\text{tr}}(\tau(a), o). \end{aligned} \tag{7}$$

The prior term is

$$g_{\text{prior}}(c; \theta_p) = \beta_P \ell_c^P + \beta_A \ell_c^A + \beta_O \ell_c^O + \beta_F \ell_c^F + \beta_{FO} \ell_c^{FO}. \tag{8}$$

Thus, statistics over pairs, attributes, objects, families, and family object combinations are all part of the shared architecture. The support coefficients for pair, family, and family object terms are constrained to be nonnegative, while the attribute and object tail coefficients are constrained to be nonpositive.

Ranked family bias. The main and auxiliary family posteriors are converted into a small ranked bias over candidate families. Let $T_r^m(x)$ and $T_r^a(x)$ denote the family at rank r under $q^m(\cdot \mid x)$ and $q^a(\cdot \mid x)$, respectively. Let $\kappa(x) = \max_f q^m(f \mid x)$ and $\rho(x) = \mathbb{I}[\kappa(x) \geq \theta]$. For $f_c = \tau(a)$, the bias is

$$g_{\text{family}}(x, c; \theta_f) = \rho(x) \sum_{r=1}^{R_m} \eta_r^m \mathbf{1}[f_c = T_r^m(x)] + \sum_{r=1}^{R_a} \eta_r^a \mathbf{1}[f_c = T_r^a(x)]. \tag{9}$$

In implementation, $R_m = 3$ and $R_a = 2$; the lower ranked main and auxiliary positions share a rest coefficient. During optimization, the hard confidence gate is replaced by a sigmoid relaxation. This term shifts candidate groups according to family evidence, while differences between pairs remain governed by $\bar{s}(x, c)$ and g_{prior}.

Family prior correction for each sample. We also compare the family posterior from pair marginalization with the training family prior. For $c = (a, o)$, define

$$z(x, c) = \log q^p(\tau(a) \mid x) - \log \hat{P}_{\text{tr}}(\tau(a)). \tag{10}$$

We center this quantity over candidate pairs for the same image:

$$h_{\text{prior}}(x, c) = z(x, c) - \frac{1}{|\mathcal{C}|} \sum_{c' \in \mathcal{C}} z(x, c'). \tag{11}$$

The scalar γ is learned with the rest of the calibrator.

3.5 Learning Objective

For each source subset, we fit the parameters $\Theta = \{\alpha, \theta_p, \theta_f, \gamma\}$ from labeled training pairs. For each training image, we take the top-K candidates under $\bar{s}(x, c)$ and keep the sample when the true pair is included in this candidate set. The primary objective is cross entropy over the calibrated top-K scores:

$$\mathcal{L}_{\text{CE}} = -\log \frac{\exp(S(x, c^\star))}{\sum_{c \in \mathcal{T}_K(x)} \exp(S(x, c))}. \quad (12)$$

To align learning with generalized CZSL evaluation, we form a deterministic pseudo unseen partition of training pairs and compute a differentiable AUC surrogate over the pseudo seen and pseudo unseen domains. This surrogate uses soft seen/unseen maxima, a smooth harmonic mean term, and a pairwise margin term. The final objective is

$$\begin{aligned} \mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda_{\text{auc}} \mathcal{L}_{\text{pAUC}} \\ + \lambda_{\text{anchor}} \mathcal{L}_{\text{anchor}} + \lambda_1 \|\Theta\|_1. \end{aligned} \quad (13)$$

The anchor term limits deviation from initialization, and the L_1 term shrinks weak calibration coefficients. After fitting each source subset, its validation AUC is used to choose the final source subset.

3.6 Training and Inference Procedure

Algorithm 1 summarizes the full procedure. During training, for each source subset $\mathcal{S}$, FASC performs three steps: it averages the logits from the sources in $\mathcal{S}$, computes family evidence from source posteriors or by marginalizing pair posteriors, and learns the calibrator parameters on the training images. Each subset therefore has its own calibrated scorer. After training, the validation AUC selects the final subset and its parameters.

At inference time, no parameters are updated. FASC averages the selected source logits, computes the same family evidence, adds the learned calibration terms to every candidate pair, and ranks candidates by Eq. 6. Closed world prediction and the generalized CZSL seen/unseen bias sweep both use this same calibrated score.

Algorithm 1 FASC Calibration and Inference

Require: Source logits $\{s_m(x, c)\}$, optional family posteriors, training set $\mathcal{D}_{tr}$, validation set $\mathcal{D}_{val}$, candidate set $\mathcal{C}$, GPT-4o family map τ, maximum subset size B.

Ensure: Calibrated ranking score $S(x, c)$.

Compute smoothed training frequency statistics $\ell^P, \ell^A, \ell^O, \ell^F, \ell^{FO}$ from $\mathcal{D}_{tr}$.
Initialize best validation AUC $v^\star \leftarrow -\infty$ and best subset $\mathcal{S}^\star \leftarrow \emptyset$.
for all source subsets $\mathcal{S}$ with $1 \leq |\mathcal{S}| \leq B$ **do**
  Average selected source logits: $\bar{s}(x, c) \leftarrow |\mathcal{S}|^{-1} \sum_{m \in \mathcal{S}} s_m(x, c)$.
  Obtain family evidence q^m, q^a from source posteriors; if unavailable, use q^p from pair marginalization.
  Initialize $\Theta = \{\alpha, \theta_p, \theta_f, \gamma\}$.
  for epoch $= 1, \ldots, E$ **do**
    Build top-K candidate set $\mathcal{T}_K(x)$ under $\bar{s}(x, c)$ for each $x \in \mathcal{D}_{tr}$.
    Compute calibrated scores $S(x, c)$ by Eq. 6 on $\mathcal{T}_K(x)$.
    Update Θ using $\mathcal{L}_{CE} + \lambda_{auc}\mathcal{L}_{pAUC} + \lambda_{anchor}\mathcal{L}_{anchor} + \lambda_1\|\Theta\|_1$.
  end for
  Evaluate $S(x, c)$ on $\mathcal{D}_{val}$ and obtain validation AUC $v_{\mathcal{S}}$.
  if $v_{\mathcal{S}} > v^\star$ **then**
    $v^\star \leftarrow v_{\mathcal{S}}$; $\mathcal{S}^\star \leftarrow \mathcal{S}$; store $\Theta^\star \leftarrow \Theta$.
  end if
end for
Inference: average logits from $\mathcal{S}^\star$, compute family evidence, score every $c \in \mathcal{C}$ with Eq. 6, and rank pairs by $S(x, c)$.

4 Experiments

4.1 Experimental Setting

Datasets. Following standard CZSL evaluation protocols, we evaluate FASC on three widely used benchmarks: MIT-States (Isola et al., 2015), UT-Zappos (Yu and Grauman, 2014), and C-GQA (Naeem et al., 2021). Detailed dataset statistics are reported in Appendix A.

Metrics. Following the standard closed world setting (Huang et al., 2024; Zhang et al., 2025), we evaluate using Seen (S)/Unseen (U) composition accuracy, Harmonic Mean (HM), and Area Under Curve (AUC).

Implementation details. We use VP-CMJL (Zhang et al., 2025) as the CZSL backbone and apply FASC to its candidate pair logits. The backbone uses CLIP ViT-L/14 (Radford et al., 2021) as the image/text encoder. For each dataset, we build a source pool from independent checkpoints of the same backbone. The pool consists of checkpoint logits obtained from different random seeds and op-

timizer settings, including the released checkpoint when available. We enumerate all nonempty source subsets from a pool of four sources, train the same calibrator on each averaged logit subset, and select the final subset by validation AUC. This gives 15 candidate subsets per dataset and evaluation setting. The cap of four sources keeps the search compact while reducing variance from a single checkpoint, giving FASC a more stable base logit distribution for calibration with family structure; the selected subset can contain fewer than four sources when this gives better validation AUC. The complete source pools and selected subsets are reported in Appendix B. The FASC calibrator is optimized for 4 epochs with AdamW, learning rate 5×10^{-3}, and batch size 512. We use top-K training candidates with $K = 50$, pseudo unseen fraction 0.25, AUC auxiliary weight 0.02, anchor weight 0.001, and L_1 shrinkage weight 0.05. All experiments are conducted on NVIDIA A100 GPUs.

4.2 Main Results

Table 1 reports the results under both closed and open world settings, comparing FASC with recent CZSL methods based on CLIP. Since FASC is applied to the same VP-CMJL candidate pair logits used by the backbone row, the direct comparison with VP-CMJL isolates the effect of structured calibration with families. In the closed world setting, FASC achieves the best HM and AUC on all three datasets and obtains the best result in 11 out of 12 reported metrics. Compared with VP-CMJL, FASC improves AUC by +0.8, +0.6, and +1.0 on C-GQA, MIT-States, and UT-Zappos, respectively, while also improving HM by +0.1, +0.5, and +1.1. These improvements are concentrated on HM and AUC, the metrics that directly summarize the tradeoff between seen and unseen pairs in CZSL, showing that FASC improves calibrated composition ranking rather than only increasing seen accuracy.

In the open world setting, FASC remains competitive under the enlarged candidate space. It achieves the best AUC on C-GQA and MIT-States and improves MIT-States HM/AUC from 22.0/8.3 to 22.5/8.8. UT-Zappos shows a more challenging pattern: its attributes are largely material or state cues, and the open world candidate expansion introduces many plausible but incorrect compositions, such as *leather boot*, *suede shoe*, and *rubber sandal*. Calibration with family structure can strengthen material family consistency, but it does not by itself fully resolve object composition ambiguity among such plausible pairs. Overall, the gains indicate that structured calibration with families improves the tradeoff between seen and unseen pairs while preserving the discriminative strength of the underlying pair scorer.

4.3 Ablation Study

To verify the effectiveness of the proposed calibration modules, we conduct ablation studies on UT-Zappos. This dataset provides a clear testbed for examining whether corrections based on family structure improve the final seen/unseen ranking. Each variant removes one component while keeping the remaining setting fixed.

Table 2 shows that the family terms improve the closed world ranking beyond pair/object/attribute calibration without family information: removing them lowers HM by 0.6 points and AUC by 1.3 points. Removing the learnable base logit scale causes a comparable test AUC drop, supporting its role in matching the magnitude of the VP-CMJL logits and the structured calibration terms.

Parameter sensitivity. We also report a compact sensitivity study for the hyperparameters that determine candidate coverage, ranking regularization, and calibrator capacity. These results summarize the representative factors that most directly affect the calibrated ranking behavior.

Table 3 shows that the method is stable around the main configuration. Using $K = 20$ and $K = 50$ gives essentially the same AUC, while $K = 100$ hurts the selected ranking; thus $K = 50$ provides a practical balance between candidate coverage and ranking stability. A nonzero auxiliary AUC loss is important: removing it lowers test AUC by 1.4 points, while increasing the weight from 0.02 to 0.05 gives no meaningful gain. Moderate L_1 shrinkage gives the best AUC among the tested capacities; removing it or shrinking the coefficients too strongly both reduce performance.

4.4 Efficiency Analysis

We measure the scoring cost after the backbone for FASC on C-GQA, which has the largest candidate space among our benchmarks. Once candidate pair logits are available, FASC only applies vectorized score scaling, family marginalization, offsets conditioned on family, and final ranking. On C-GQA with 5,098 test images and 7,767 candidate pairs, a single thread CPU benchmark adds only 0.1044 ms per image over base argmax ranking. This indicates

Method	Venue	C-GQA				MIT-States				UT-Zappos			
		S	U	HM	AUC	S	U	HM	AUC	S	U	HM	AUC
Closed World Results													
CLIP (2021)	ICML'21	7.5	25.0	8.6	1.4	30.2	46.0	26.1	11.0	15.8	49.1	15.6	5.0
CoOp (2022)	IJCV'22	20.5	26.8	17.1	4.4	34.4	47.6	29.8	13.5	52.1	49.3	34.6	18.8
CSP (2023)	ICLR'23	28.8	26.8	20.5	6.2	46.6	49.9	36.3	19.4	64.2	66.2	46.6	33.0
DFSP(i2t) (2023)	CVPR'23	35.6	29.3	24.3	8.7	47.4	52.4	37.2	20.7	64.2	66.4	45.1	32.1
DFSP(BiF) (2023)	CVPR'23	36.5	32.0	26.2	9.9	47.1	52.8	37.7	20.8	63.3	69.2	47.1	33.5
DFSP(t2i) (2023)	CVPR'23	38.2	32.0	27.1	10.5	46.9	52.0	37.3	20.6	66.7	71.7	47.2	36.0
DLM (2024)	AAAI'24	32.4	28.5	21.9	7.3	46.3	49.8	37.4	20.0	67.1	72.5	52.0	39.6
ProLT (2024)	AAAI'24	39.5	32.9	27.7	11.0	49.1	51.0	38.2	21.1	66.0	70.1	49.4	36.1
PLID (2024)	ECCV'24	38.8	33.0	27.9	11.0	49.7	52.4	39.0	22.1	67.3	68.8	52.4	38.7
CDS-CZSL (2024)	CVPR'24	38.3	34.2	28.1	11.1	50.3	52.9	39.2	22.4	63.9	74.8	52.7	39.5
Troika (2024)	CVPR'24	41.0	35.7	29.4	12.4	49.0	53.0	39.3	22.1	66.8	73.8	54.6	41.7
IMAX (2025)	TPAMI'25	39.7	35.8	29.8	12.8	48.7	53.8	39.1	21.9	69.3	70.7	54.2	40.6
VP-CMJL (2025)	ICCV'25	46.0	40.2	34.9	16.3	**51.8**	52.6	40.4	23.3	71.9	76.3	58.5	47.9
FASC (Ours)	–	**46.9**	**41.0**	**35.0**	**17.1**	51.5	**53.9**	**40.9**	**23.9**	**72.1**	**77.7**	**59.6**	**48.9**
Open World Results													
CLIP (2021)	ICML'21	7.5	4.6	4.0	0.3	30.1	14.3	12.8	3.0	15.7	20.6	11.2	2.2
CoOp (2022)	IJCV'22	21.0	4.6	5.5	0.7	34.6	9.3	12.3	2.8	52.1	31.5	28.9	13.2
CSP (2023)	ICLR'23	28.7	5.2	6.9	1.2	46.3	15.7	17.4	5.7	64.1	44.1	38.9	22.7
DFSP(i2t) (2023)	CVPR'23	35.6	5.6	9.0	1.9	47.2	18.2	19.1	6.7	64.3	53.8	41.2	26.4
DFSP(BiF) (2023)	CVPR'23	36.5	7.6	10.6	2.4	47.1	18.1	19.2	6.7	63.5	57.2	42.7	27.6
DFSP(t2i) (2023)	CVPR'23	38.2	7.2	10.4	2.4	47.5	18.5	19.3	6.8	66.8	60.0	44.0	30.3
PLID (2024)	ECCV'24	39.1	7.5	10.6	2.5	49.1	18.7	20.4	7.3	67.6	55.5	46.6	30.8
CDS-CZSL (2024)	CVPR'24	37.6	8.2	11.6	2.7	49.4	**21.8**	22.1	8.5	64.7	61.3	48.2	32.3
Troika (2024)	CVPR'24	40.8	7.9	10.9	2.7	48.8	18.4	20.1	7.2	66.4	61.2	47.8	33.0
IMAX (2025)	TPAMI'25	38.7	7.9	11.2	2.5	50.2	18.6	21.4	7.6	68.4	57.3	47.5	32.3
VP-CMJL (2025)	ICCV'25	46.0	**11.5**	**15.5**	4.6	**51.8**	19.9	22.0	8.3	**71.9**	**66.6**	**54.5**	**41.4**
FASC (Ours)	–	**47.0**	11.2	15.3	**4.7**	51.6	20.8	**22.5**	**8.8**	68.6	66.5	52.7	38.3

Table 1: Experimental results for both closed and open world settings. The best and second best performance are highlighted in bold and underlined, respectively.

Variant	Val AUC	S	U	HM	AUC
FASC	53.72	72.1	77.7	59.6	48.9
w/o family terms	53.68	70.8	77.2	59.0	47.6
w/o base logit scale	53.80	70.7	77.4	58.7	47.3

Table 2: Component ablation on UT-Zappos. "w/o family terms" removes the family prior, the prior over family and object combinations, the ranked family bias, and the sample level correction to the family log prior while keeping pair/object/attribute calibration.

Hyperparameter	Setting	Val AUC	HM	AUC
Top-K candidates	20	54.00	59.64	48.93
	50	53.72	59.60	48.91
	100	53.70	59.07	47.37
AUC auxiliary weight λ_{auc}	0.00	53.78	58.89	47.47
	0.02	53.72	59.60	48.91
	0.05	53.80	59.60	48.92
L_1 shrinkage weight λ_1	0.00	53.89	59.50	48.70
	0.05	53.72	59.60	48.91
	0.10	54.02	59.23	47.62

Table 3: Parameter sensitivity on UT-Zappos. Bold settings are used in the main experiments.

that the inference cost is dominated by the backbone scorer, while the proposed calibration with family structure introduces negligible additional overhead.

4.5 Quality Analysis

Case analysis. We visualize representative cases of FASC and its variant without family terms in Fig. 3. In successful cases, FASC corrects predictions that are visually plausible but supported by a less appropriate attribute family. For example, flat pair scoring can favor salient color or size cues, predicting *white tie* instead of *plaid tie*, *wet dog* instead of *brown dog*, or *large window* instead of *closed window*. By aggregating evidence at the attribute family level, FASC increases the consistency between the image cue and the candidate attribute family while preserving composition ranking between pairs. This helps the model select the intended composition rather than a visually related distractor from another family.

The MIT-States cases show a similar pattern.

Figure 3: Qualitative results on C-GQA and MIT-States. The term "w/o family" denotes FASC with the family calibration terms removed, while the green font indicates correct labels and the red font indicates incorrect labels. Family labels are shown to highlight whether each prediction follows the true attribute family.

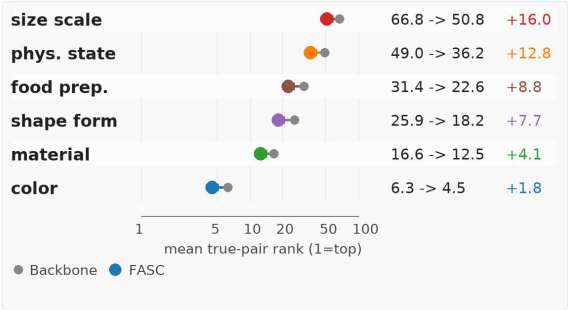

Figure 4: Visualization by family in score space on C-GQA. Rows group test images by true attribute family and report the mean true pair rank before and after FASC; leftward arrows indicate rank gains.

Calibration with family structure suppresses distractors from other families such as *wilted salad* for *cooked chicken*, and it also helps among visually close alternatives within the same family such as *crumpled plastic* versus *crinkled paper*. In failure cases, both variants may still be distracted when multiple plausible primitives are present or when object identity dominates the attribute family cue. These examples suggest that family structure mainly improves the calibration of ambiguous pair rankings.

Score space visualization. We further conduct a visualization by family in score space in Fig. 4. For each image, we rank all candidate pairs and record the true composition rank; smaller ranks indicate better ordering. We group selected C-GQA test images by true attribute family and compare the mean rank before and after FASC. Gray markers denote the backbone rank, colored markers denote the calibrated rank, and leftward arrows indicate that the correct pair moves closer to the top. For example, "66.8 $\rightarrow$ 50.8" means that, within the size family, the correct composition moves from an average rank of 66.8 under the backbone to 50.8 after calibration; the reported "+16.0" is the number of rank positions gained. The visualization shows consistent improvements across different families, including size, physical state, food preparation, shape, material, and color; physical state also improves from 49.0 to 36.2. These shifts illustrate how calibration with family structure converts evidence at the family level into better ordering of candidate pairs, complementing the qualitative cases in Fig. 3.

5 Conclusion

We presented FASC, a lightweight structured calibrator that uses family information for compositional zero-shot learning. FASC keeps the original candidate pair prediction space, but uses semantic attribute families as intermediate structure for calibrating pair logits. By combining averaged source logits, image level family evidence, training frequency priors, and a ranking objective, FASC calibrates pair logits without replacing the backbone scorer. Experiments on C-GQA, MIT-States, and UT-Zappos show stronger seen/unseen trade-offs and the best results on most reported metrics. Qualitative and score space analyses further show that family evidence helps correct ambiguous pair rankings. Overall, semantic family structure provides a simple and effective signal for CZSL score calibration.

Limitations

FASC is a calibration method built on top of candidate pair logits, so it cannot recover information that is absent from the underlying scorer or from the candidate set. This limitation is most visible in the UT-Zappos open world setting in Table 1. UT-Zappos attributes are largely material or state cues, and the open world candidate space contains many semantically plausible but incorrect combinations, such as different material–footwear pairings. Calibration with family structure can strengthen family consistency, but it cannot by itself decide which plausible object composition is the true pair when several candidates share similar family evidence. Thus, FASC does not fully solve the object ambiguity introduced by open world expansion on this benchmark.

The method also relies on a fixed attribute family taxonomy constructed offline with GPT-4o. Although we canonicalize family names and use the taxonomy only as intermediate calibration structure, ambiguous attributes or attributes specific to a dataset can still be assigned imperfectly. Future work could model uncertainty over family assignments or learn the taxonomy jointly with calibration. Finally, our main experiments use a strong VP-CMJL backbone. Evaluating the same calibration design across a broader set of CZSL backbones would further clarify how broadly the family signal transfers.

References

Mattia Balestra, Marina Paolanti, and Roberto Pierdicca. 2025. Whu-rs19 abzsl: An attribute-based dataset for remote sensing image understanding. *Remote Sensing*.

Wentao Bao, Lichang Chen, Heng Huang, and Yu Kong. 2024. Prompting language-informed distribution for compositional zero-shot learning. In *European Conference on Computer Vision*.

Nicole Christoff and Krasimir Tonchev. 2024. Material attribute estimation as part of telecommunication augmented reality, virtual reality, and mixed reality system: Systematic review. *Electronics*.

Xinyue Fan, Teng Liu, Hong Bao, Weiguo Pan, Tianjiao Liang, and Han Li. 2022. Long-tail instance segmentation based on memory bank and confidence calibration. *Applied Sciences*.

Shaozhe Hao, Kai Han, and Kwan-Yee Kenneth Wong. 2023. Learning attention as disentangler for compositional zero-shot learning. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15315–15324.

Xiaoming Hu and Zilei Wang. 2024. A dynamic learning method towards realistic compositional zero-shot learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 2265–2273.

Siteng Huang, Biao Gong, Yutong Feng, Min Zhang, Yiliang Lv, and Donglin Wang. 2024. Troika: Multi-path cross-modal traction for compositional zero-shot learning. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24005–24014.

Phillip Isola, Joseph J. Lim, and Edward H. Adelson. 2015. Discovering states and transformations in image collections. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1383–1391.

Chenyi Jiang, Shidong Wang, Yang Long, Zechao Li, Haofeng Zhang, and Ling Shao. 2025. Imaginary-connected embedding in complex space for unseen attribute-object discrimination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(3):1395–1413.

Chenyi Jiang and Haofeng Zhang. 2024. Revealing the proximate long-tail distribution in compositional zero-shot learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 2498–2506.

Shyamgopal Karthik, Massimiliano Mancini, and Zeynep Akata. 2022. Kg-sp: Knowledge guided simple primitives for open world compositional zero-shot learning. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9326–9335.

Soohyeong Kim and Yong Suk Choi. 2025. Contextual feature expansion with superordinate concept for compositional zero-shot learning. *Applied Sciences*.

Yong-Lu Li, Yue Xu, Xiaohan Mao, and Cewu Lu. 2020. Symmetry and group in attribute-object compositions. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11313–11322.

Yun Li, Zhe Liu, Hang Chen, and Lina Yao. 2024. Context-based and diversity-driven specificity in compositional zero-shot learning. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17037–17046.

Xiaodong Liu, Weixing Luo, Jiale Du, Xinshuo Wang, Yuhao Dang, and Yang Liu. 2024. A robust generalized zero-shot learning method with attribute prototype and discriminative attention mechanism. *Electronics*.

Xiaocheng Lu, Song Guo, Ziming Liu, and Jingcai Guo. 2023. Decomposed soft prompt guided fusion enhancing for compositional zero-shot learning. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23560–23569.

Massimiliano Mancini, Muhammad Ferjad Naeem, Yongqin Xian, and Zeynep Akata. 2021. Open world compositional zero-shot learning. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5218–5226.

Ishan Misra, Abhinav Kumar Gupta, and Martial Hebert. 2017. From red wine to red tomato: Composition with context. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1160–1169.

Muhammad Ferjad Naeem, Yongqin Xian, Federico Tombari, and Zeynep Akata. 2021. Learning graph embeddings for compositional zero-shot learning. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 953–962.

Tushar Nagarajan and Kristen Grauman. 2018. Attributes as operators: Factorizing unseen attribute-object compositions. *Proceedings of the European Conference on Computer Vision (ECCV)*, abs/1803.09851.

Nihal V. Nayak, Peilin Yu, and Stephen H. Bach. 2023. Learning to compose soft prompts for compositional zero-shot learning. *International Conference on Learning Representations*, abs/2204.03574.

OpenAI. 2024. Gpt-4o system card. Accessed: 2026-05-26.

Senthil Purushwalkam, Maximilian Nickel, Abhinav Kumar Gupta, and Marc'Aurelio Ranzato. 2019. Task-driven modular networks for zero-shot compositional learning. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3592–3601.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. *Proceedings of the 38th International Conference on Machine Learning*, abs/2103.00020:8748–8763.

Klearchos Stavrothanasopoulos, Konstantinos Gkountakos, Konstantinos Ioannidis, Theodora Tsikrika, Stefanos Vrochidis, and Ioannis Kompatsiaris. 2025. Cylinders: A benchmark visual dataset for robust gas cylinder detection and attribute classification in real-world scenes. *Sensors (Basel, Switzerland)*, 25.

Qingsheng Wang, Lingqiao Liu, Chenchen Jing, Hao Chen, Guoqiang Liang, Peng Wang, and Chunhua Shen. 2023. Learning conditional attributes for compositional zero-shot learning. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11197–11206.

Peng Wu, Qiuxia Lai, Hao Fang, Guo-Sen Xie, Yilong Yin, Xiankai Lu, and Wenguan Wang. 2025a. A conditional probability framework for compositional zero-shot learning. *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3673–3683.

Peng Wu, Xiankai Lu, Hao Hu, Yongqin Xian, Jianbing Shen, and Wenguan Wang. 2025b. Logiczsl: Exploring logic-induced representation for compositional zero-shot learning. *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 30301–30311.

Karn Yongsiriwit, Parkpoom Chaisiriprasert, Thannob Aribarg, and Sokliv Kork. 2025. Context-aware visual emotion recognition through hierarchical fusion of facial micro-features and scene semantics. *Applied Sciences*.

Aron Yu and Kristen Grauman. 2014. Fine-grained visual comparisons with local learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 192–199.

Shiyu Zhang, Cheng Yan, Yang Liu, Chenchen Jing, Lei Zhou, and Wenjun Wang. 2025. Learning visual proxy for compositional zero-shot learning. *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2793–2802.

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

A Dataset Details

Table 4 summarizes the dataset statistics used by prior CZSL work (Lu et al., 2023; Huang et al., 2024; Zhang et al., 2025). Here, s and o denote the number of attribute/state and object concepts, c_s and c_u denote seen and unseen composition pairs for each split, and i denotes images.

B Source Pool Details

Table 5 reports the fixed checkpoint pool with four sources used to generate candidate pair logits for each benchmark. We construct three source pools in total, one for each benchmark. Each pool is a deterministic checkpoint grid over backbone learning rates and random seeds, and each source uses the retained validation checkpoint snapshot from the corresponding backbone run. Here, a seed denotes an independently trained backbone checkpoint and is not a random branch sampled during subset search. After the pool is fixed, every benchmark follows the same procedure: enumerate all nonempty source subsets with size at most four, uniformly average their logits, fit the same calibrator, and select the final subset by validation AUC. Concretely, for a pool $\mathcal{M} = \{S1, S2, S3, S4\}$, the candidate subset set is $\mathcal{Q} = \{\mathcal{S} : \emptyset \neq \mathcal{S} \subseteq \mathcal{M}\}$,

Dataset	s	o	Train		Validation			Test		
			c_s	i	c_s	c_u	i	c_s	c_u	i
MIT-States	115	245	1262	30338	300	300	10420	400	400	12995
UT-Zappos	16	12	83	22998	15	15	3214	18	18	2914
C-GQA	453	870	6963	26920	1173	1368	7280	1022	1047	5098

Table 4: Summary statistics of the three CZSL datasets.

which contains $2^4 - 1 = 15$ subsets: four subsets of size one, six of size two, four of size three, and one of size four. Equivalently, the exhaustive list consists of $\{S1\}$, $\{S2\}$, $\{S3\}$, $\{S4\}$; all source pairs $\{S1, S2\}$, $\{S1, S3\}$, $\{S1, S4\}$, $\{S2, S3\}$, $\{S2, S4\}$, $\{S3, S4\}$; all source triples $\{S1, S2, S3\}$, $\{S1, S2, S4\}$, $\{S1, S3, S4\}$, $\{S2, S3, S4\}$; and the full pool $\{S1, S2, S3, S4\}$. For each candidate $\mathcal{S} \in \mathcal{Q}$, we first average its source logits and then train FASC with the same objective and hyperparameters.

All sources use the same VP-CMJL backbone architecture. The optimizer is Adam; batch size and training epochs follow the corresponding backbone recipe. C-GQA and MIT-States use the same seed/lr grid to keep source pool construction aligned across the two benchmarks.

Table 6 lists the subsets selected by validation AUC under the closed and open world settings. Closed and open world candidate spaces are evaluated separately, so the validation selected subset can also differ between the two settings. In all cases, the selection rule and the calibrator are unchanged; a smaller subset is selected when adding more sources does not improve validation AUC.

Prompt for Attribute Family Taxonomy Construction

Role. You are a careful visual attribute taxonomy annotator for compositional zero-shot learning.
Task. Given a list of English attribute words, assign each attribute to three fields: `family`, a short and reusable semantic family name in snake case; `strength`, one of `core`, `aux`, or `noise`; and `mask_mode`, one of `local`, `global`, `hybrid`, or `none`.
Definitions. `local` means that the main visual evidence is usually on the foreground object; `global` means that it is mainly in the scene or background; `hybrid` means that object and context are both important; and `none` is used for noisy or unstable attributes. `core` families include color, material, pattern, shape, size, age, surface texture, physical state, pose, action, expression, cleanliness, and illumination. `aux` families include weather, environment, style, configuration, and food preparation. `noise` is used for object like modifiers, suspicious labels, or attributes without stable shared visual evidence.
Constraints. The family set is not closed: create a new family when needed, but prefer reusing semantically equivalent names from `[SEED_FAMILIES]` and `[EXISTING_FAMILIES]`. Use concise, stable, and general family names; do not use uninformative names such as `other`, `misc`, or `unknown`. Cover every attribute in `[ATTRIBUTE_LIST]` exactly once. Return JSON only, with no Markdown or extra explanation.
Output schema. Return a JSON object with a `results` list. Each item must contain `attr`, `family`, `strength`, and `mask_mode`, e.g., `red` → `color/core/local`, `urban` → `environment_context/aux/global`, and `banana` → `object_like_modifier/noise/none`.

C Attribute Family Prompt

We construct the attribute family taxonomy offline from the dataset attribute vocabulary. For each dataset, the placeholders in the prompt card below are replaced with the dataset attribute list, a small set of seed family names, and family names already produced in previous batches. The resulting model outputs are canonicalized before being fixed as the mapping τ used by FASC.

Dataset	S1	S2	S3	S4
C-GQA	lr=1.0×10^{-4}, seed 51	lr=1.5×10^{-4}, seed 52	lr=2.0×10^{-4}, seed 53	lr=3.0×10^{-4}, seed 54
MIT-States	lr=1.0×10^{-4}, seed 51	lr=1.5×10^{-4}, seed 52	lr=2.0×10^{-4}, seed 53	lr=3.0×10^{-4}, seed 54
UT-Zappos	lr=5.0×10^{-4}, seed 0	lr=5.0×10^{-4}, seed 0	lr=5.0×10^{-4}, seed 0	lr=5.0×10^{-4}, seed 6

Table 5: Backbone source pool used by FASC for source logit averaging.

Dataset	Closed world selected sources	Open world selected sources
C-GQA	S1, S2, S3, S4	S1, S3, S4
MIT-States	S1, S4	S1, S4
UT-Zappos	S1, S2, S4	S1, S2, S3

Table 6: Validation selected source subsets used in the main experiments.