Visual Delta Generator with Large Multi-modal Models for Semi-supervised Composed Image Retrieval

Supplementary Material

6. Reproduction Guide

6.1. More Implementation details

VDG: Alignment. For stage 1 in Sec. 3.1, the focus is on training the projection layer in the vision projector and our baseline LLM, LLaMA2-13B, to achieve alignment. Training is executed over a single epoch with a learning rate of 1×10^{-3}, batch size 64 per GPU.

VDG: Instructional Tuning. For stage 2 in Sec. 3.1, additional LoRA parameters (θ_{lora}) are applied, configured with $\alpha = 16$, rank $= 64$, and dropout $= 0.05$. The tuning process begins at a learning rate of 2×10^{-4}, incorporating a warm-up phase over 100 iterations. The learning rate is reduced to one-tenth after reaching half of the total 10 epochs, batch size 8 per GPU.

VDG: Augmentation. For VDG training, we simply apply basic random resized cropping to our images. This involves adjusting the scale of the images between 0.8 and 1.0 and their aspect ratios between 0.9 and 1.1.

VDG: Visual Delta Generation The generation of visual deltas is conducted autoregressively, applying a temperature scaling of 0.2 to the top 50 token predictions.

CIR: Hyper-parameters For Eqn. 2, we set the hyper-parameters as $\tau = 0.01, \alpha = 1.0, \beta = 0.0$ for CLIP-based Combiner, which makes loss as same as standard contrastive loss, and $\tau = 0.01, \alpha = 1.0, \beta = 0.5$ for BLIP baseline.

CIR: Augmentation. CIR model training employs a standard data augmentation pipeline to enhance robustness. We start with a random resized crop, adjusting the scale of the images between 0.5 and 1.0. Further, a random horizontal flip, and random adjustments to image contrast, brightness, and sharpness are applied. We also incorporate different perspectives and angles of images by modifying translation and rotation.

CIR: Model Training. Batch size is set as 64 per GPU for CIR model training. The CIR models begin with an initial learning rate of $1e-4$, which follows a cosine decay schedule to zero for 6 and 10 epochs for BLIP and CLIP baselines, separately.

Model Training and Optimization All models are optimized using AdamW optimizer [37], with $\beta_1 = 0.9$, $\beta_2 = 0.99$, and a consistent weight decay of 0.05. Training is performed on 8 NVIDIA A100 80GB GPUs using *bfloat16* precision.

6.2. Training procedure

We provide a training procedure for VDG in Algorithms 1 and 2, as well as a semi-supervised learning approach for the CIR model in Algorithm 3. We set the same batch size for $\mathcal{B}$ and $\mathcal{B}'$.

Algorithm 1 Stage 1 - Alignment of VDG

Initialize θ_{proj}
Input: D - Filtered image-text pairs from CC3M
Input: P - Set of prompts from LLaVA [33]
$\ell_{align} \leftarrow \mathcal{L}_{LLM}$ with $\mathcal{B} \sim D, p \sim P$
$\theta_{proj} \leftarrow \theta_{proj} - \gamma \frac{\partial \ell_{align}}{\partial \theta_{proj}}$
Ensure: Updated θ_{proj}

Algorithm 2 Stage 2 - Instruction Tuning of VDG

Load θ_{proj} from Stage 1, initialize θ_{lora}
Input: D - Train set of CIRR or FashionIQ
Input: p_{inst} - prompt from Fig. 3
$\ell_{inst} \leftarrow \mathcal{L}_{LLM}$ with $\mathcal{B} \sim D, p_{inst}$
$\theta_{proj} \leftarrow \theta_{proj} - \gamma \frac{\partial \ell_{inst}}{\partial \theta_{proj}}$
$\theta_{lora} \leftarrow \theta_{lora} - \gamma \frac{\partial \ell_{inst}}{\partial \theta_{lora}}$
Ensure: Updated $\theta_{proj}, \theta_{lora}$

Algorithm 3 Semi-supervised CIR training

Load E_{img}, f_θ
Input: D - Train set of CIRR or FashionIQ (additional visual deltas are applied with VDG)
Input: D' - Pseudo triplet generated from auxiliary gallery with VDG
$\ell_{tcc} \leftarrow \mathcal{L}_{tcc}$ with $\mathcal{B} \sim D, \mathcal{B}' \sim D'$
$\ell_{tdm} \leftarrow \mathcal{L}_{tdm}$ with $\mathcal{B} \sim D, \mathcal{B}' \sim D'$ (BLIP only)
$f_\theta \leftarrow f_\theta - \gamma \frac{\partial (\ell_{tcc} + \ell_{tdm})}{\partial f_\theta}$
Ensure: Updated f_θ

Table 6. Detailed dataset configurations for auxiliary galleries and CC3M-Filtered for alignment training (Stage 1 in Sec. 3). '#' denotes the number of images in the dataset. '#' ref-tar pairs denotes the number of unique reference-target image pairs.

Dataset	# images	# ref-tar pairs
NLVR2′	63,788	152,604
COCO′	102,436	285,939
FashionIQ′	33,994	100,647
DeepFashion′	38,237	111,168
CC3M-Filtered	595,375	-

Table 7. Ablation study on VDG for CIRR validation set.

Methods	R@1	R@5	R@10	R@50
Our Final Model	50.16	80.03	87.78	96.75
(1) LLaMA2-7B	50.04	79.29	86.94	95.86
(2) LLaMA2-13B-chat	50.23	79.67	86.96	96.03
(3) LoRA rank=32	49.03	79.12	86.80	96.24
(4) LoRA rank=128	48.94	79.05	86.96	95.86
(5) Q-Former: BLIP-2	49.63	78.76	86.68	95.72

Table 8. Experiment results on the CIRR test set using different scales of the CC3M-Filtered Dataset (∼1.4M pseudo CIR triplets). The dataset scales from one eight (1/8) to the full set (1).

Ratio	R@1	R@5	R@10	R@50
1/8	44.43	73.18	82.36	92.28
1/4	45.34	73.90	82.82	93.21
1/2	45.37	74.24	83.06	93.25
1	45.42	74.55	83.28	93.32

7. Further Analysis

Data statistics. We provide detailed configuration of auxiliary gallery used for experiments in Table 6.

Design Choice. We investigate several configuration options, including: (a) the model size of the LLM, (b) the type of LLM used, (c) the rank of LoRA, and (4) the type of Q-Former. Based on our final model in the first row, we change the designated component in each row. We explore these options in Table 7 to assess their impact on CIR performance. Specifically, for (1) we experiment with the LLaMA2-7B model. For (2), we opt for the chat-bot style tuned LLaMA2-13B-chat model. Regarding (3) and (4), we experiment with varying the rank at 32 and 128, noting that our baseline is 64. Finally, for (5), we employ the Q-Former from BLIP-2, which is in line with FlanT5-XXL [8], rather than using the InstructBLIP one. It is important to note that our evaluations indicate that these options do not significantly affect performance. This underscores the general applicability and robustness of our VDG, demonstrating its effectiveness across a variety of configurations.

Larger Scale Experiment. We further configure 1,431,135 pseudo triplets with the CC3M-Filtered dataset to explore the scalability of VDG to larger datasets and show the results in Table 8. It appears that as the dataset size increases, there's a gradual improvement in the recall metrics, suggesting that using more data improves the model's ability to retrieve relevant results.

VDG Generation Results. We provide more generation results based on subgroups in Figs. 9 and 10. We notice that the VDG excels in generating high-quality visual deltas, with only a few errors.

Retrieval Results. We provide more retrieval results for natural images in Figs. 11 and 12, and for fashion images in Figs. 13 and 14. In the domain of natural images, we chose query examples containing the word *must*. We observe that our CIR model effectively grasps the meaning of the text query and reflects this understanding in the retrieval results. In addition, the domain of fashion images is also well-represented in the retrieval results, accurately reflecting the user's text query while maintaining visual information of query image.

Figure 9. Visual delta generation results with VDG on CIRR validation set.

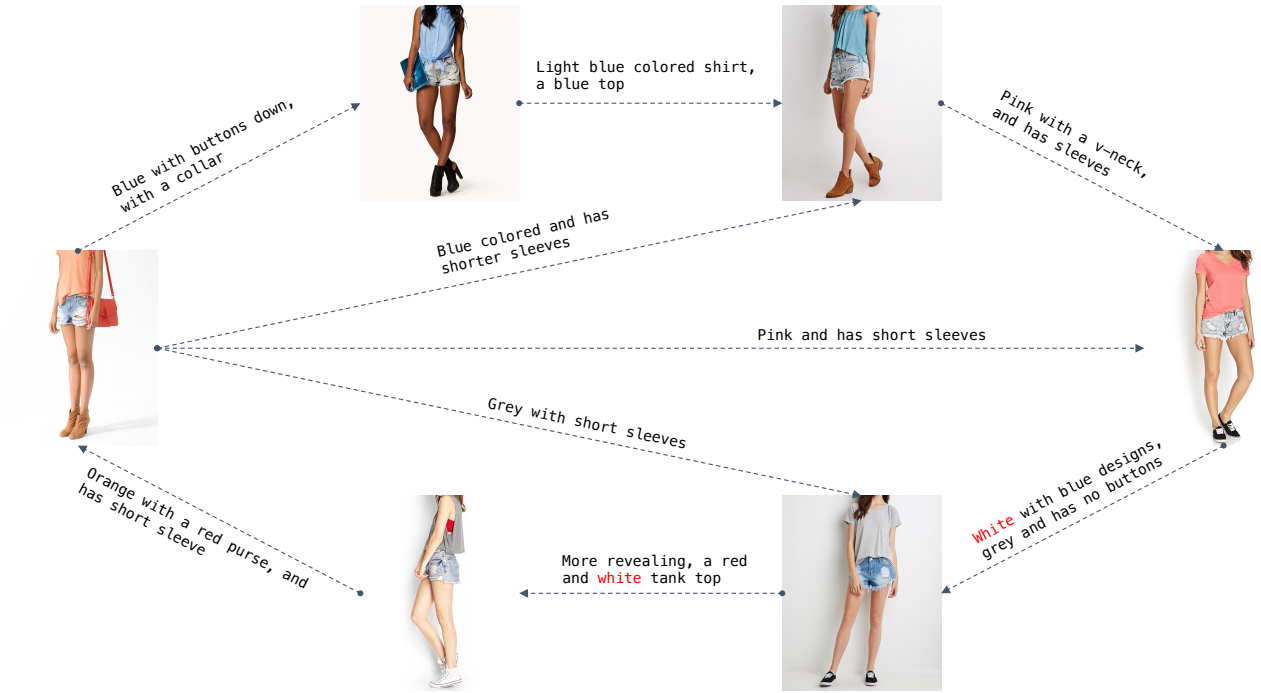

Figure 10. Visual delta generation results with VDG on the DeepFashion dataset.

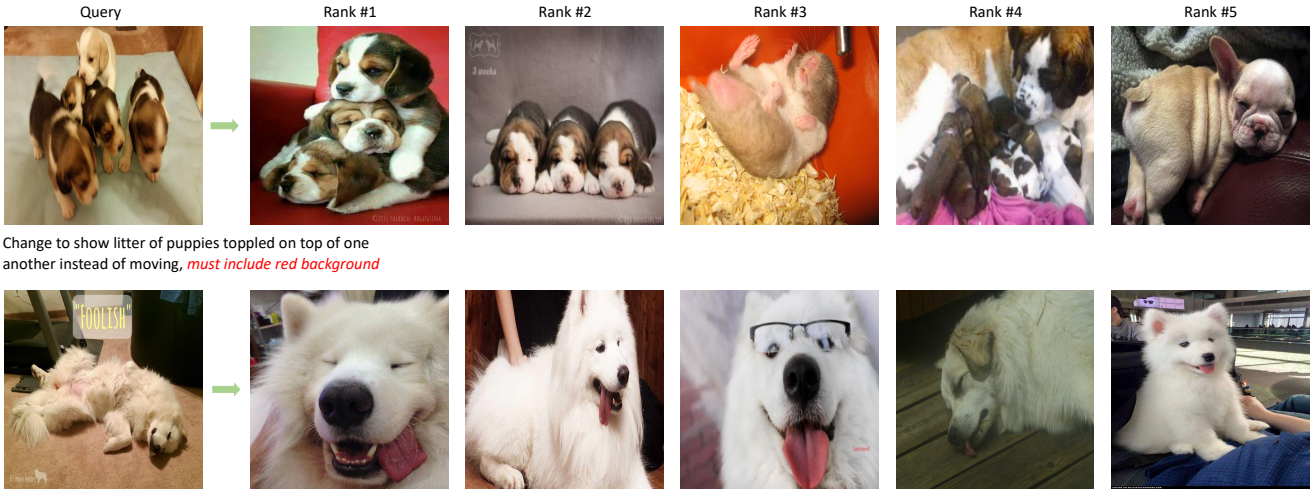

Figure 11. Retrieval results on the CIRR test set.

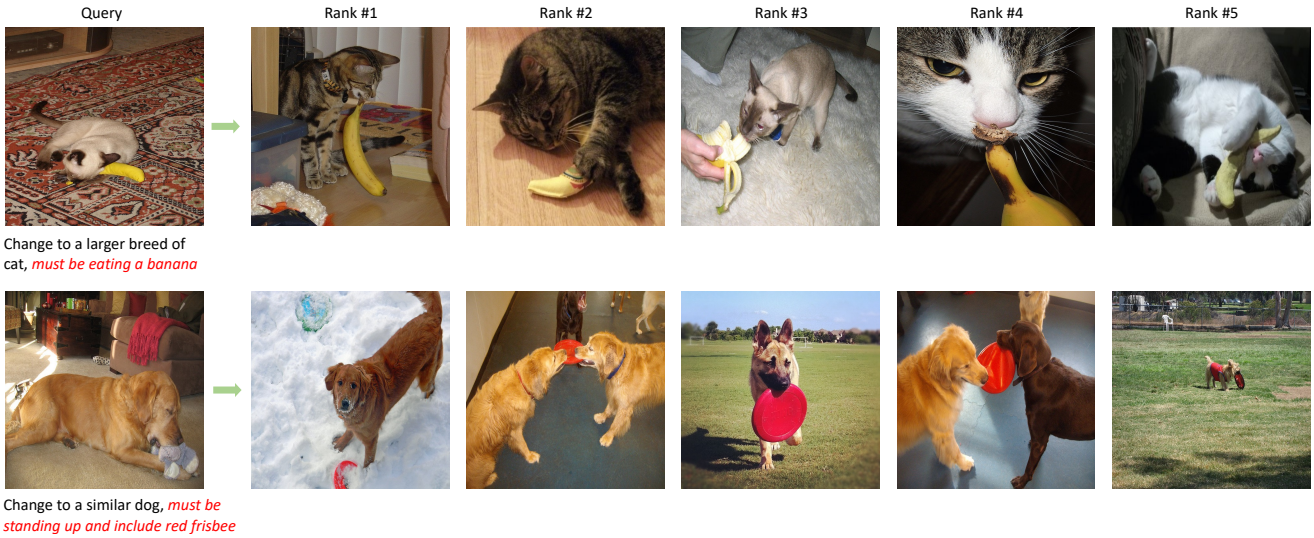

Figure 12. Retrieval results on the COCO dataset.

Figure 13. Retrieval results on the FashionIQ dataset.

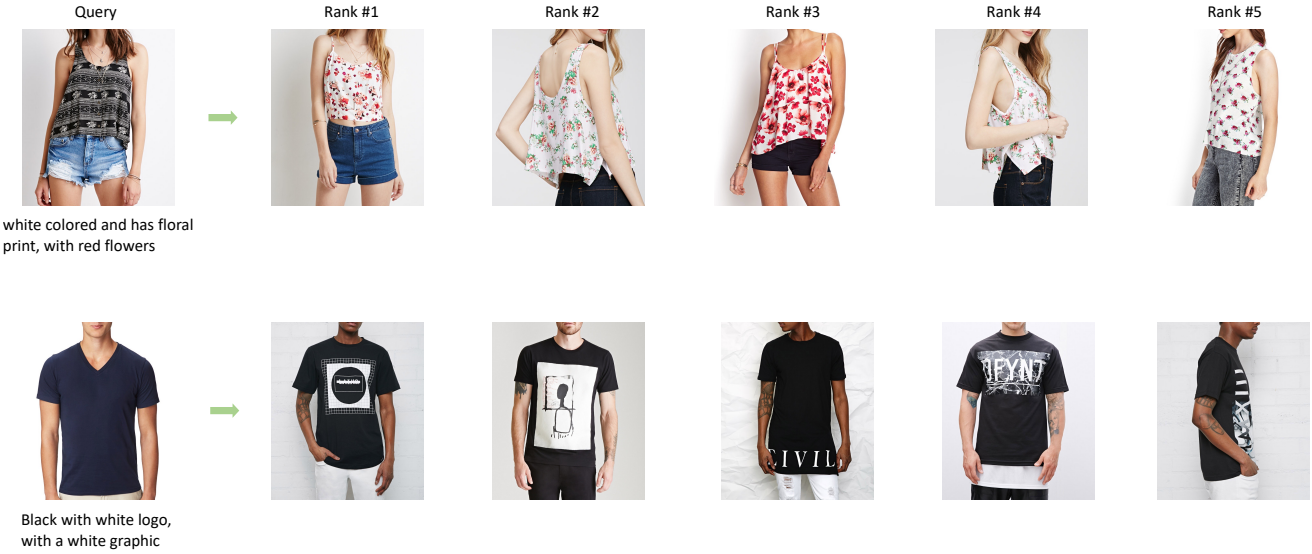

Figure 14. Retrieval results on the DeepFashion dataset.