Multimodal RAG Enhanced Visual Description

Amit Kumar Jaiswal[1] Haiming Liu[2] Ingo Frommholz[3]

PROBLEM FORMULATION AND CONTRIBUTION

- **Goal** To generate high-quality textual descriptions for images by developing a lightweight, computationally efficient method that effectively bridges the 'modality gap' (the misalignment between visual and textual representations) found in pre-trained Large Multimodal Models (LMMs).
- **Problem Settings:** Pre-trained LMMs like CLIP suffer from a modality gap, where visual and text embeddings are not perfectly aligned in the common embedding space.
  - This misalignment harms the performance of retrieval-based methods for tasks like image captioning.
  - Existing solutions such as full fine-tuning or end-to-end training, are computationally expensive, impractical, and require extensive domain-specific data.

Contributions:

- mRAG-gim, a lightweight, training-free approach that uses Retrieval-Augmented Generation (RAG) and a simple linear mapping to bridge the modality gap.
- A novel iterative technique (Algorithm 2) that uses synthetic descriptions generated by an LLM to augment the training data and progressively optimize the linear mapping.
- Demonstration that the linear mapping retains semantic meaning by evaluating it with a user-behavior-driven metric (nDCG) alongside standard captioning metrics.

METHODOLOGY

The mRAG-gim approach is divided into two main stages, with an optional refinement loop.

Stage I: Aligning Visual & Textual Space (Mapping)

- **Objective:** To create a training-free mapping L_M that projects the visual embedding space onto the textual embedding space.
  1. Use a pre-trained CLIP model to extract image embeddings ($\mathbf{v}_i$) and text embeddings ($\mathbf{e}_i$) from the training dataset S_{train}.
  2. Compute the linear mapping L_M using Ordinary Least Squares (OLS) by finding a closed-form solution that minimizes the objective:

$$\min_{L_M} \sum_{i \in S_{train}} ||L_M \mathbf{v}_i - \mathbf{e}_i||^2$$

Stage II: Retrieval-Augmented Generation (Inference)

- **Objective:** To generate a new textual description for a given input image X.
  1. Embed the input image using the image encoder: $\mathbf{v} = \Phi_{IE}(X)$.
  2. Apply the mapping to project the image embedding into the text space: $L_M \mathbf{v}$.
  3. Use this mapped embedding $L_M \mathbf{v}$ to retrieve the **Top-k** most similar textual descriptions $\mathcal{D} = \{d_1, d_2, ..., d_k\}$ from a vector database (using cosine similarity).
  4. Feed these retrieved descriptions $\mathcal{D}$ as a **context prompt**, along with an instruction (e.g., "Show similar images: The image describes:"), into an instruction-tuned LLM (FLAN-T5) to generate the final description.

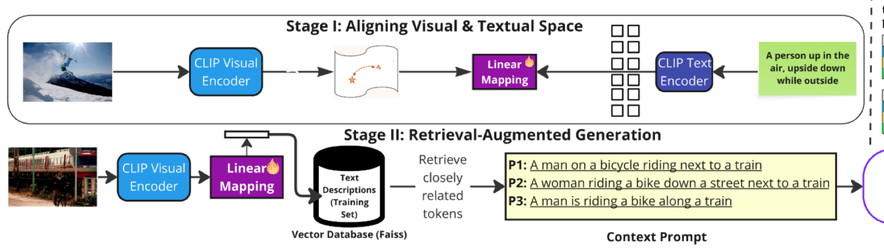

Table 3: Tokens ranking from the visual feature space to the input of FLAN-T5 textual feature spaces for mappings (OLS) evaluated on MSCOCO external validation data. Cyan bold text indicates optimal captioning performance across CLIP backbones.

CLIP Encoder (Backbone)	NDCG@1	NDCG@10	NDCG@100
RN50x64	$0.413_{\pm 0.062}$	$0.572_{\pm 0.091}$	$0.592_{\pm 0.094}$
ViT-B/32	$0.472_{\pm 0.0101}$	$0.573_{\pm 0.103}$	$0.581_{\pm 0.104}$
BeiT-B/16	$0.384_{\pm 0.060}$	$0.371_{\pm 0.061}$	$0.376_{\pm 0.066}$

Captioning Performance

B@1	B@4	R-L	C-D
$64.2_{\pm 0.2}$	$17.8_{\pm 0.3}$	$48.1_{\pm 0.3}$	$66.1_{\pm 0.6}$
$59.4_{\pm 0.3}$	$16.2_{\pm 0.3}$	$46.7_{\pm 0.3}$	$61.7_{\pm 0.7}$
$22.1_{\pm 0.2}$	$0.9_{\pm 0.0}$	$22.8_{\pm 0.2}$	$1.9_{\pm 0.1}$

Continuous Refinement (Algorithm 2): This is an iterative process to further improve the mapping L_M.

1. **Generate:** Use the current mRAG-gim model to generate synthetic descriptions for the training set.
2. **Filter:** Evaluate the synthetic descriptions against reference captions using a quality metric (e.g., CIDEr-D) and discard any that fall below a computed average score.
3. **Augment:** Add the high-quality synthetic descriptions to the training set S_{train} and the vector database.
4. **Retrain:** Re-compute the linear mapping L_M on the newly augmented dataset. This loop is repeated until performance on a validation set plateaus.

Algorithm 1 mRAG-gim: RAG-based Visual Descriptions

Input: Image encoder Φ_{IE}, text encoder Φ_{TE}, training data $S_{train}=(X_i, T_i)$, test data $S_{test} = (X_j)$, LLM$(\cdot)$ as generative model, hyperparameter k, prompt $\mathcal{P}$

$\{v_i, e_i\}_{i=1}^{|S_{train}|} \leftarrow \Phi_{IE}(X_i), \Phi_{TE}(T_i)$ for $(X_i, T_i) \in S_{train}$ ▷ Incorporate training data
$L_M \leftarrow$ fit linear mapping$(\{v_i, e_i\})$ ▷ pre-compute mapping scheme
VD $\leftarrow \{e_i\}$ ▷ Initialise vector database with training text descriptions
$\{L_M v_i\}_{i=1}^{|S_{test}|} \leftarrow \Phi(X_j)$ for $(X_j, T_j) \in S_{test}$ ▷ Incorporate unseen images from test set in CLIP space
$\{\mathcal{D}_j\} \leftarrow$ topk$(\{L_M v_j\}, \text{VD}, k)$ ▷ top-k descriptions to retrieve from vector database
$\{\mathcal{G}_j\} \leftarrow$ LLM(concatenate$(\mathcal{P} + \mathcal{D}_j)$) ▷ Generate new textual descriptions

Output: $\{\mathcal{G}_j\}$

- *1: Indian Institute of Technology (BHU) Varanasi, India amit.chr@iitbhu.ac.in*
- *2:University of Southampton, United Kingdom h.liu@soton.ac.uk*
- *3: Modul University Vienna, Austria ifrommholz@acm.org*

METHODOLOGY

Algorithm 2 Continuous Refinement for Retrieval Augmentation

Input: Image encoder Φ_{IE}, text encoder Φ_{TE}, training data $S_{train}=(X_i, T_i)$, validation data $S_{val} = (X_j)$, LLM$(\cdot)$ as generative model, hyperparameter k, prompt $\mathcal{P}$

$\{v_i, e_i\}_{i=1}^{|S_{train}|} \leftarrow \Phi_{IE}(X_i), \Phi_{TE}(T_i)$ for $(X_i, T_i) \in S_{train}$ ▷ Incorporate training data
$L_M \leftarrow$ fit linear mapping$(\{v_i, e_i\})$ ▷ Pre-compute mapping scheme
VD $\leftarrow \{e_i\}$ ▷ Initialise vector database with training text descriptions
$\bar{\tau} \leftarrow$ eval$(S_{val}, \Phi_{IE}, L_M, \text{LLM}, \text{VD})$ ▷ Evaluation on the validation data
for _ in range(n) **do**
  $\{\mathcal{D}_i\} \leftarrow$ topk$(\{L_M v_i\}, \text{VD}, k)$ ▷ Top-k descriptions to retrieve from vector database
  $\{\mathcal{G}_j\} \leftarrow$ LLM(concatenate$(\mathcal{P} + \mathcal{D}_j)$) ▷ Generate new textual descriptions
  $\{\mathcal{G}_j\} \leftarrow$ filter$(\mathcal{G}_j, \bar{\tau})$ ▷ Remove de-duplicate descriptions based on $\bar{\tau}$
  $\{e_l\}_{l=1}^{|S_{train}|} \leftarrow \Phi_{IE}(u_l)$ for $(u_l) \in \mathcal{D}$ ▷ Incorporate new generated descriptions
  VD $\leftarrow$ VD $\cup \{e_l\}$ ▷ Add generated descriptions to database
  $\{v_i, e_i, e_l\}_{i,l=1}^{|S_{train}|} \leftarrow e_l$ for $(e_l) \in \{e_l\}$ ▷ Augment training data
  $L_M \leftarrow$ fit linear mapping$(\{v_i, e_i, e_l\})$ ▷ Recompute mapping scheme
  $\bar{\tau} \leftarrow$ eval$(S_{val}, \Phi_{IE}, L_M, \text{LLM}, \text{VD})$ ▷ Update average metric
end for

EXPERIMENT & RESULT

Table 1: Quantitative comparison against state-of-the-art methods on two multimodal datasets. † depicts the training time on CPU, and †† is the training time in multiple of n iterations to run the Algorithm 2. The metric scores on MSCOCO and Flickr30k test sets are in-domain and out-domain data (↑ is better). The CLIP-score (C-S) and RefClip (RC) [10] is for the ablation analysis. Bold (in cyan) indicates the best results among baseline methods. The green shaded cells are for mRAG-gim without linear mapping, whereas the yellow shaded cells are with linear mapping.

Method \ Datasets	MSCOCO B@4	MSCOCO S	MSCOCO C-D	Flickr30k S	Flickr30k C-D	Training (Params,Time)	Ablation (CIDEr-D) in-domain	Ablation (CIDEr-D) out-domain	Ablation Test (CLIP-score/RefClip) C-D	Ablation Test (CLIP-score/RefClip) S	Ablation Test (CLIP-score/RefClip) C-S/RC
ClipCap [23]	33.5	21.1	113.1	15.8	57.9	43M,1.4hr(L4)	-	-	$15.8_{\pm 0.1}$	$5.8_{\pm 0.1}$	75.6, 79.1
SmallCap [26]	36	21	117.4	-	-	1.8M,13hr(L4)	55.4	52.2	$17.8_{\pm 0.2}$	$6.6_{\pm 0.2}$	75.9, 79.4
Llama-AdapterV2 [7]	36.2	-	122.2	-	-	14M, -	-	-	$80.0_{\pm 0.7}$	$18.4_{\pm 0.1}$	79.3, 80.2
mRAG-gim	$31.3_{\pm 0.2}$	$21.1_{\pm 0.2}$	$107.8_{\pm 0.4}$	$16.1_{\pm 0.3}$	$64.5_{\pm 1.7}$	1M,8s-10s†	$56.9_{\pm 1.4}$	$43.2_{\pm 1.2}$	$47.0_{\pm 0.3}$	$14.1_{\pm 0.2}$	73.7, 78.1
mRAG-gim+Alg. 2	$29.0_{\pm 0.4}$	$21.6_{\pm 0.1}$	$104.3_{\pm 0.7}$	$17.3_{\pm 0.1}$	$66.8_{\pm 2.0}$	1M,8s-10s††	$60.2_{\pm 1.4}$	$45.1_{\pm 1.5}$	$41.4_{\pm 0.2}$	$12.9_{\pm 0.1}$	72.8, 77.2

Table 2: LLM evaluations on the MSCOCO validation set provided standard errors for all models.

LLMs	B@1	B@2	R-L	C-D	S	Params
FLAN-T5-small	$57.6_{\pm 0.2}$	$21.2_{\pm 0.2}$	$54.2_{\pm 0.2}$	$90.2_{\pm 0.6}$	$20.6_{\pm 0.1}$	60M
FLAN-T5-base	$60.4_{\pm 0.2}$	$22.5_{\pm 0.4}$	$54.9_{\pm 0.2}$	$92.3_{\pm 0.4}$	$20.6_{\pm 0.1}$	220M
FLAN-T5-large	$77.7_{\pm 0.2}$	$31.0_{\pm 0.5}$	$57.9_{\pm 0.3}$	$106.1_{\pm 0.7}$	$21.0_{\pm 0.1}$	770M
FLAN-T5-xl	$76.0_{\pm 0.1}$	$29.8_{\pm 0.3}$	$57.0_{\pm 0.3}$	$103.4_{\pm 1.0}$	$20.6_{\pm 0.2}$	3B
FLAN-T5-xxl	$64.2_{\pm 0.1}$	$23.5_{\pm 0.2}$	$54.6_{\pm 0.1}$	$94.7_{\pm 0.2}$	- $_{\pm 0.1}$	11B
DeBERTaV3-base [9]	$59.2_{\pm 0.3}$	$22.1_{\pm 0.4}$	$54.7_{\pm 0.3}$	$91.2_{\pm 0.2}$	$20.9_{\pm 0.3}$	86M
LLaMa [31]	$63.8_{\pm 0.5}$	$27.6_{\pm 0.5}$	$50.2_{\pm 0.3}$	$89.2_{\pm 0.9}$	$19.4_{\pm 0.3}$	7B
Results for the most similar-to-dissimilar (S2d) descriptions ordering in the Prompt						
Most S2d	$77.4_{\pm 0.1}$	$30.4_{\pm 0.1}$	$58.0_{\pm 0.3}$	$105.6_{\pm 0.6}$	$21.0_{\pm 0.3}$	
d2S	$77.8_{\pm 0.1}$	$30.5_{\pm 0.1}$	$58.1_{\pm 0.3}$	$106.3_{\pm 0.9}$	$21.4_{\pm 0.3}$	

OUR RESULT

Figure 1:Our method (*mRAG-gim*) generates textual descriptions using MSCOCO validation images, employing our computed mapping derived from the MSCOCO training set.

- **Extreme Efficiency:** The linear mapping (with 1M parameters) trains in 8-10 seconds on a CPU. This is orders of magnitude faster than the SmallCap baseline, which requires 13 hours on an L4 GPU.
- **Competitive Performance:** mRAG-gim achieves performance close to the SOTA lightweight SmallCap method on the SPICE metric.
- **Refinement Boost:** The continuous refinement (Alg. 2) significantly improves out-of-domain generalization, achieving a CIDEr-D score of 60.2 on the Flickr30k ablation, surpassing SmallCap (52.2).
- **Metric Mismatch:** The study confirms that unsupervised metrics like CLIP-score correlate poorly with supervised, reference-based metrics (like CIDEr-D), suggesting they can be misleading.
- **Recency Bias:** Altering the order of retrieved descriptions in the prompt (e.g., similar-to-dissimilar vs. dissimilar-to-similar) impacts LLM output, confirming

CONCLUSIONS

mRAG-gim is a RAG-based approach that successfully generates high-quality visual descriptions by using a simple, training-free OLS linear mapping to bridge the LMM modality gap. By retrieving captions to use as context prompts for an LLM, the method achieves competitive performance against more complex lightweight methods.
The proposed continuous refinement process iteratively enhances the mapping using high-quality synthetic captions. Most importantly, the approach is exceptionally computationally efficient, enabling the use of smaller LLMs (like FLAN-T5) and democratizing high-performance image captioning for users with limited computational resources.