UNIMORE
UNIVERSITÀ DEGLI STUDI DI
MODENA E REGGIO EMILIA

University of Modena and Reggio Emilia
Department of Engineering "Enzo Ferrari"

XXXVIII cycle of the International Doctorate School in
Information and Communication Technologies (ICT)

Retrieval-Augmented Multimodal Understanding

From Models to Evaluation

Sara Sarto

Advisor: Prof. Rita Cucchiara
Director of the School: Prof. Luigi Rovati

Advisor:

Prof. Rita Cucchiara — University of Modena and Reggio Emilia

Director of the School:

Prof. Luigi Rovati — University of Modena and Reggio Emilia

Review Committee:

Prof. Nicu Sebe — University of Trento
Prof. Mohamed Elhoseiny — King Abdullah University of Science and Technology

Tesi di dottorato finanziata dall'Unione europea – Next Generation EU, Missione 4, componente 2 "Dalla Ricerca all'Impresa" – Investimento 3.3 "Introduzione di dottorati innovativi che rispondono ai fabbisogni di innovazione delle imprese e promuovono l'assunzione dei ricercatori dalle imprese".

To my mum and dad.

Retrieval-Augmented Multimodal Understanding From Models to Evaluation

Abstract

In Artificial Intelligence, the introduction of the attention mechanism and the Transformer architecture has enabled models capable of processing multiple modalities at unprecedented scale. This breakthrough is largely due to the flexibility of the attention operator and the adaptability of the architecture, which have given rise to a new generation of vision-language systems. Among the tasks at the intersection of Computer Vision, Natural Language Processing, and Multimedia, image captioning (i.e., the task of generating natural language descriptions of visual content) has played a pivotal role. In the era of modern Multimodal Large Language Models (MLLMs), captioning remains a fundamental component, now coexisting with multimodal tasks such as Visual Question Answering (VQA).

To further enhance the capabilities of such models, retrieval augmentation has emerged as a key strategy. Enriching models with relevant external knowledge improves factual grounding and adaptability, enabling more accurate and context-aware responses, particularly in knowledge-intensive or domain-specific scenarios. This thesis represents the natural evolution of retrieval augmentation, moving from its early application in image captioning to its integration within modern MLLMs. Each stage builds upon the insights and challenges encountered along the way, addressing open problems in evaluation and retrieval effectiveness.

The first part of the thesis establishes the foundations of retrieval-augmented vision–language models. It analyzes classical cross-modal retrieval and extends beyond standard settings to address more complex scenarios, including multimodal queries and heterogeneous documents. A central insight is that retrieval quality critically determines performance, particularly as multimodal applications increasingly involve both multimodal queries and document collections. In response, this work we present new retrievers (ReT and ReT-2) tailored for multimodal scenarios.

Building on this, this work investigates retrieval-augmented architectures for captioning through the introduction of the RA-Transformer, in which external knowledge is integrated into the caption generation, providing cues to generate richer and more precise descriptions.

The thesis then extends retrieval augmentation to MLLMs, motivated by the fact that even large-scale pretraining struggles with domain-specific or knowledge-intensive queries. Specifically, WikiLLaVA introduces retrieval-augmented MLLM architectures for knowledge-based VQA, where retrieval mechanisms are used to enhance reasoning capabilities and adaptability to complex multimodal queries.

Throughout this research, it becomes evident that the advancement of captioning models is constrained by the lack of robust and reliable evaluation metrics. Traditional metrics, while widely adopted, often fail to capture semantic adequacy, factual grounding, and fluency. To address this, a major contribution of this thesis is the design and analysis of new evaluation metrics for image captioning – i.e., PAC-S, BRIDGE, and an improved version of PAC-S. These metrics are specifically designed to align with human judgment and capture the multifaceted quality of captions. Beyond their introduction, the thesis investigates their application across different benchmarks and domains, including their ability to evaluate captions generated by MLLMs, reflecting the shift of captioning from a standalone task to an integral component of broader multimodal reasoning systems.

Overall, this thesis, through novel retrieval-augmented captioning architectures, improved evaluation metrics, and specialized multimodal retrievers, contributes new methodologies, tools, and insights that advance the field of multimodal AI.

Multimodal Understanding tramite Retrieval-Augmentation Dai Modelli alla Valutazione

Sommario

Nel campo dell'Intelligenza Artificiale (IA), l'introduzione del meccanismo di attention e del Transformer ha reso possibili modelli in grado di elaborare più modalità su scala senza precedenti. Questa svolta è dovuta in gran parte alla flessibilità dell'operatore di attention e all'adattabilità dell'architettura, che hanno dato origine a una nuova generazione di sistemi visione-linguaggio. Tra i task all'intersezione tra Computer Vision, Natural Language Processing e Multimedia, l'image captioning (ovvero la generazione di descrizioni in linguaggio naturale a partire da contenuti visivi) ha svolto un ruolo centrale. Nell'era dei moderni Multimodal Large Language Models (MLLMs), il captioning resta un elemento fondamentale, oggi affiancato da task multimodali come il Visual Question Answering (VQA).

Per potenziare le capacità di questi modelli, la retrieval augmentation è emersa come una strategia chiave. L'arricchimento dei modelli con rilevante conoscenza esterna migliora l'adattabilità, consentendo risposte più accurate e sensibili al contesto, soprattutto in scenari complessi o specialistici. Questa tesi rappresenta l'evoluzione naturale della retrieval augmentation, passando dalle sue prime applicazioni nell'image captioning alla sua integrazione nei moderni MLLMs. Ogni fase si basa sulle intuizioni e sulle sfide incontrate, affrontando problemi aperti legati alla valutazione e all'efficacia del retrieval.

La prima parte della tesi stabilisce le basi dei modelli visione–linguaggio con retrieval augmentation. Si analizzano le tecniche classiche di cross-modal retrieval e si va oltre le casistiche standard per affrontare scenari più complessi, inclusi query multimodali e collezioni di documenti eterogenee. Un'intuizione centrale è che la qualità del retrieval determina in modo critico le prestazioni, soprattutto in applicazioni multimodali con query multimodali e grandi collezioni documentali. In risposta a questa osservazione, il lavoro presenta nuovi retriever, ReT e ReT-2, progettati specificamente per scenari multimodali.

La tesi indaga anche architetture di captioning con retrieval augmentation attraverso l'introduzione del RA-Transformer, in cui la conoscenza esterna viene integrata direttamente nel processo di generazione, fornendo segnali utili a produrre caption più ricche e precise.

Successivamente, il lavoro estende la retrieval augmentation ai MLLMs, motivato dal fatto che anche il pretraining su larga scala mostra limiti nell'affrontare query knowledge-intensive o specifiche di dominio. In particolare, WikiLLaVA introduce architetture MLLM con retrieval augmentation per il knowledge-based VQA, in cui i meccanismi di retrieval potenziano le capacità di ragionamento e l'adattabilità a query multimodali complesse.

Nel corso della ricerca emerge come il progresso dei modelli di captioning sia limitato dalla mancanza di metriche di valutazione robuste e affidabili. Le metriche tradizionali, sebbene ampiamente utilizzate, spesso non riescono a catturare adeguatezza semantica, grounding fattuale e fluidità linguistica. Quindi, un contributo fondamentale di questa tesi è la progettazione e l'analisi di nuove metriche di valutazione per l'image captioning, ovvero PAC-S, BRIDGE e una versione migliorata di PAC-S. Tali metriche sono progettate per allinearsi al giudizio umano e per catturare la qualità delle descrizioni. La tesi ne analizza anche l'applicazione su diversi benchmark e domini, inclusa la loro capacità di valutare caption generate da MLLMs, riflettendo il passaggio del captioning da compito autonomo a

componente di sistemi di ragionamento multimodale più ampi.

Nel complesso, attraverso nuove architetture di captioning con retrieval augmentation, retriever multimodali specializzati e metriche di valutazione, questa tesi fornisce metodologie, strumenti e contributi che avanzano lo stato dell'arte nell'ambito dell'Intelligenza Artificiale multimodale.

Contents

1

Introduction

Vision and language constitute two of the most fundamental modalities through which humans perceive, interpret, and communicate about the world. Visual input provides rich, structured information about objects, scenes, and spatial relationships, while language enables abstraction, reasoning, and the expression of meaning. The interaction between these modalities is central to human intelligence and underpins our ability to describe, explain, and reason about what we see.

The growing interest in generative models has naturally led to the development of systems designed to describe visual content in natural language. For a human observer, this task feels effortless: we instinctively recognize objects, infer their relationships and actions, and choose words that convey what is happening in a scene. Depending on the context and

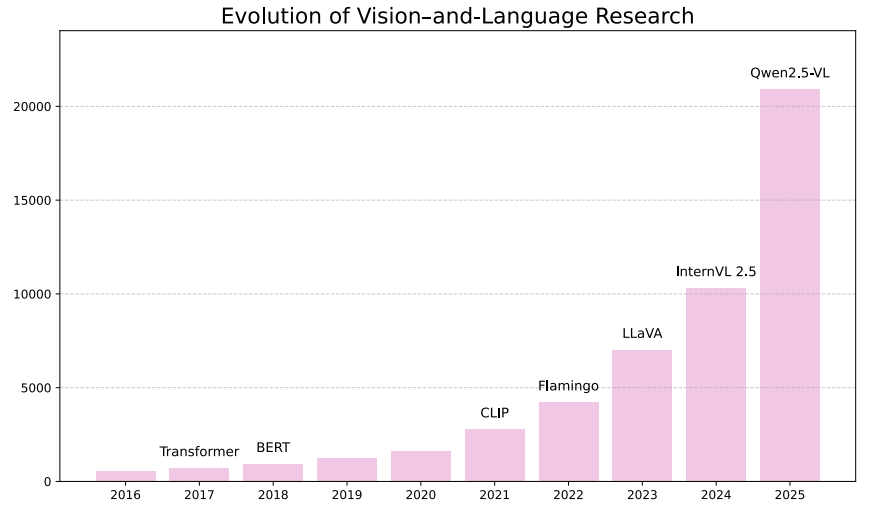


Figure 1.1: Evolution of vision–and-language research over time*. Key milestones in model development are annotated, highlighting major advances in the field.

intent, we may produce a concise factual description or articulate implications that go beyond what is explicitly visible. Beneath this apparent simplicity lies a complex cognitive process, requiring the tight integration of visual perception, semantic understanding, and language generation, with attention dynamically focused on the most relevant elements.

Enabling machines to achieve comparable capabilities poses a fundamental challenge in artificial intelligence. This has given rise to the task of **image captioning**, which serves as a canonical problem for studying vision–language integration and generative modeling.

More recently, the advent of Large Language Models (LLMs) with strong reasoning and generative abilities has reshaped the multimodal landscape. By extending these models to process visual inputs, **Multimodal Large Language Models (MLLMs)** have moved beyond standalone captioning toward more general multimodal reasoning tasks, such as visual

*The number of papers per year was estimated using Google Scholar by searching for publications containing "vision and language."

question answering and grounded dialogue. In this setting, captioning becomes one of several ways in which models express visual understanding, often producing longer, more descriptive, or more explanatory outputs. While these models exhibit impressive capabilities, they also introduce new challenges in terms of evaluation, controllability, and factual grounding. As illustrated in Figure 1.1, the number of vision-and-language papers has grown rapidly over the last decade.Early work in image captioning primarily relied on attentive architectures, followed by approaches leveraging the CLIP model. The introduction of multimodal large language models (MLLMs), such as LLaVA, has further accelerated research, leading to a significant increase in publications in this field.

Despite their rapid development and impressive capabilities, both image captioning models and modern MLLMs share a fundamental limitation: their understanding of the world is ultimately bounded by the data seen during pretraining. Captioning models, in particular, are often trained on relatively narrow or domain-specific datasets, which restricts their ability to generalize beyond familiar visual concepts. Even large-scale MLLMs, although trained on vast multimodal corpora, can struggle when faced with fine-grained queries, specialized domains, or information that is rare, implicit, or simply absent from their training data. In these situations, fluent generation and reasoning alone are not sufficient.

Retrieval augmentation naturally emerges as a response to this gap. By allowing models to dynamically retrieve and incorporate external information at inference time, retrieval-augmented architectures offer a way to ground multimodal generation in up-to-date, domain-specific, and contextually relevant knowledge. Rather than relying solely on what has been implicitly memorized during training, models can extend their effective knowledge, producing outputs that are not only more informative but also more reliable and adaptable.

Crucially, the effectiveness of retrieval-augmented systems critically de-

pends on the quality of the retrieval mechanism itself. As multimodal applications increasingly involve complex queries and large-scale, heterogeneous corpora retrieval evolves from a simple matching problem into a challenging **multimodal reasoning task** in its own right. This progression motivates the study of increasingly sophisticated retrieval models, moving from standard vision–language alignment toward advanced multimodal retrieval systems.

At the same time, evaluating the outputs of these systems remains a persistent challenge. While substantial progress has been made in generating fluent and descriptive captions, assessing their quality is inherently difficult due to the subjectivity and variability of natural language. This challenge has motivated the development of specialized **evaluation metrics** aimed at capturing semantic adequacy, factual correctness, and alignment with human judgment.

1.1 Contributions & Organization

During my doctoral research, I explored the multimodal world from several complementary perspectives, following the natural evolution from foundational tasks to complex reasoning and generation.

- **Chapter 2** presents the background on vision-and-language models, tracing the evolution from traditional image captioning approaches to modern multimodal large language models (MLLMs), and introduces the most relevant benchmarks for evaluation.
- **Chapter 3** reviews retrieval-augmented models, covering their application to both image captioning and MLLMs, including knowledge-based visual question answering and associated benchmarks.
- **Chapter 4** focuses on learning to retrieve, discussing standard retrieval foundations, complex multimodal scenarios, and the develop-

ment of the ReT and ReT-2 retrieval models, including training procedures, analyses, and experimental results.

- **Chapter 5** describes retrieval-augmented image captioning models, detailing the RA-Transformer (RA-T) and Prototypical Memory Networks for Image Captioning (PMA-Net), along with their corresponding experiments.
- **Chapter 6** covers retrieval-augmented MLLMs, including the WikiLLaVA and ReAG architectures, their retrieval strategies, filtering mechanisms, generator training procedures, and experimental evaluations.
- **Chapter 7** introduces image captioning evaluation metrics, including popular metrics, taxonomy, benchmarks, and correlations with human judgment.
- **Chapter 8** presents the evolution of image captioning evaluation, describing advanced metrics (PAC-S, PAC-S++, BRIDGE), comparative analyses, and applications in multimodal LLMs.
- **Chapter 9** concludes the thesis, summarizing the contributions and outlining future directions and open problems in vision-and-language research.

2

Vision & Language Models: Foundation and Evolution

The integration of visual and linguistic understanding has long been a central pursuit in multimodal AI, driven by the goal of enabling machines to interpret, describe, and reason about the world in a human-like manner. Early research focused on tightly scoped tasks—such as aligning images with textual labels or retrieving descriptive sentences—laying the groundwork for more structured approaches to connecting vision and language. Among these, **image captioning** emerged as a pivotal problem setting, offering a concrete way to study how vi-

This chapter discusses topics from the following papers: D. Caffagni *et al.*, "The Revolution of Multimodal Large Language Models: A Survey", ACL Findings 2024 [27].

sual stimuli can be transformed into coherent natural language. Beyond its practical applications, image captioning has served as a conceptual bridge toward more general multimodal reasoning, influencing the development of vision–language pretraining, cross-modal alignment strategies, and, ultimately, the **multimodal large language models** that now underlie state-of-the-art systems.

2.1 From Image Captioning to Multimodal Large Language Models

This section provides a brief overview of the evolution from traditional image captioning to modern multimodal large language models (MLLMs). While image captioning focuses on generating descriptive text for images, MLLMs extend this capability to more complex reasoning tasks, including visual question answering and grounded dialogue. The following subsections first revisit the foundations of captioning before highlighting how these models have expanded toward general multimodal understanding and reasoning.

2.1.1 The Emergence of Vision–Language Learning

Because of the important role it can play in connecting vision and language in multimedia systems [117, 7, 4], image captioning has emerged as a fundamental task at the intersection of Computer Vision, Natural Language Processing, and Multimedia. Specifically, image captioning is the task of describing the visual content of an image in natural language, employing a visual understanding system and a language model capable of generating meaningful and syntactically correct sentences [251].

Standard Captioning Architecture. Image captioning architectures consist of an image encoding component and a language model that produces a coherent sentence in natural language describing the visual content of the input image. Therefore, it is important to focus on developing appropriate connections between the visual and textual modality [251].

Early image captioning approaches relied on detecting relevant objects in an image and filling pre-defined linguistic templates accordingly [249, 297]. More recent methods frame image captioning as a deep learning–based generative problem. In the standard setting, the task is formulated as an image-to-sequence mapping, where the input consists of raw pixels. These inputs are first transformed during a *visual encoding* stage into one or more feature vectors that represent the visual content of the image. This representation is then passed to a generative *language model*, which produces a sequence of words or subwords drawn from a fixed vocabulary. An effective representation of visual information is therefore a central challenge in image captioning pipelines. Complementarily, the language model is responsible for modeling natural language as a stochastic process by predicting the probability of word sequences. This makes it a crucial component of image captioning systems. Formally, image captioning models aim at modeling a distribution $p(y|I)$ over possible natural language descriptions y given an input image I.

During generation, the language model operates in an auto-regressive manner: each word is predicted conditioned on the previously generated words. Caption generation typically terminates when the model emits a special end-of-sequence token.

Building on this formulation, the advent of deep learning led to the widespread adoption of recurrent neural networks (RNN) and long short-term memory (LSTM) for the captioning task. In analogy to the sequence modeling used in machine translation [258], the basic RNN-based encoder-decoder scheme was employed in conjunction with CNNs for encoding

the visual content [272, 222, 131]. Nowadays, attentive and Transformer-based architectures [270] are often employed both in the visual encoding stage [61, 172], either applied to image patches directly [70, 265] or to refine features from a visual backbone, and as language models [97, 60, 32]. Regarding language models, in the last few years, large performance improvements have come from increasing the amount of training data, model size, and performing large-scale training [216, 23].

The introduction of Transformer-based models in image captioning has also brought to the development of effective variants of the self-attention operator [108, 97, 206, 64, 90, 164] and to that of vision-and-language early-fusion approaches [150, 106, 325] based on BERT-like architectures [68]. On the image encoding side, a recent paradigm is that of employing visual features extracted from large-scale multi-modal architectures [245, 62, 16, 18] like CLIP [215].

Convolutional [8] and fully-attentive language models [185, 296, 320] based on the Transformer paradigm have been used due to the limited representation power and sequential nature of RNN-based language models and thanks to their success in NLP tasks such as machine translation and language understandings [270, 68, 252].

Training Strategy. During the training stage, an image captioning model is commonly expected to generate a caption word by word by taking into account the previous words and the image. At each step, the output word is sampled from a learned distribution over the vocabulary words.

Specifically, the model is optimized with a time-wise language modeling objective, usually expressed with a cross-entropy loss. Given an image I and a ground-truth caption $\hat{y}$, in the form of a sequence of tokens, the objective at each time-step t is to predict a probability density over the token dictionary given previous ground-truth tokens $\{\hat{y}_\tau\}_{\tau<t}$ [64, 63]. Depending on the tokenization algorithm of choice, tokens might correspond

to entire words or to sub-words, and the cross-entropy loss encourages the predicted probability distribution to match the ground-truth token $\hat{y}_t$.

An additional fine-tuning stage based on reinforcement learning might also be carried out – in this case, the model is usually asked to generate an entire caption y by relying on its own prediction of previous tokens. The generated caption is then usually matched with ground-truth captions to obtain a reward signal [222].

Training Datasets. Early image captioning architectures were commonly trained and tested on the Flickr30K [307] and Flickr8K [100] datasets, consisting of pictures from the Flickr website, containing everyday activities, events, and scenes, paired with five captions each. Currently, the most commonly used dataset is Microsoft COCO [158], which consists of images of complex scenes with people, animals, and common everyday objects.

Beyond supervised training on curated benchmarks, recent work has demonstrated the effectiveness of large-scale pre-training. These datasets often consist of weakly associated or noisy image–text pairs collected from the web or from related tasks such as visual question answering, image–text retrieval, and text-to-image generation. Examples include SBU Captions [205], YFCC100M [261], Conceptual Captions [241, 36], and WIT [250], as well as even larger proprietary datasets such as ALIGN [113, 278] and those used to train CLIP [215] and DALL-E [220].

In addition to domain-generic resources, domain-specific datasets play an important role in addressing specialized challenges in captioning. These datasets target particular visual or semantic domains, such as assistive technologies (VizWiz [92] , fine-grained object categories (CUB-200 [221], Oxford-102 [221]), fashion, news imagery, or images containing embedded text. These datasets help extend captioning models beyond generic scenarios and support specialized applications. In Figure 2.1, some qualitative examples from most of the most common captioning datasets.

Figure 2.1: Qualitatives examples of different captioning datasets [251].

2.1.2 Towards General Multimodal Reasoning

The introduction of the attention operation and the Transformer architecture [270] not only has been useful to develop more powerful image captioning models, but it has enabled the creation of models capable of handling various modalities on an increasingly large scale. Initially, this breakthrough was leveraged for language-specific models [68, 23] but quickly extended to support diverse modalities [178] and facilitate their integration within unified embedding spaces [215]. The surge in sophisticated Large Language Models (LLMs), particularly their capacity for in-context learning, has encouraged researchers to broaden the scope of these models to encompass multiple modalities, both as inputs and outputs.

Building on these developments, the role of vision–language models has progressively shifted from narrowly defined generation tasks toward more general forms of multimodal reasoning. While early image captioning models were explicitly designed to produce short, descriptive sentences grounded in visual content, contemporary Multimodal Large Language Models are primarily optimized for reasoning-intensive tasks such as visual question answering, multimodal dialogue, and instruction following. Within this broader setting, image captioning is no longer treated as a

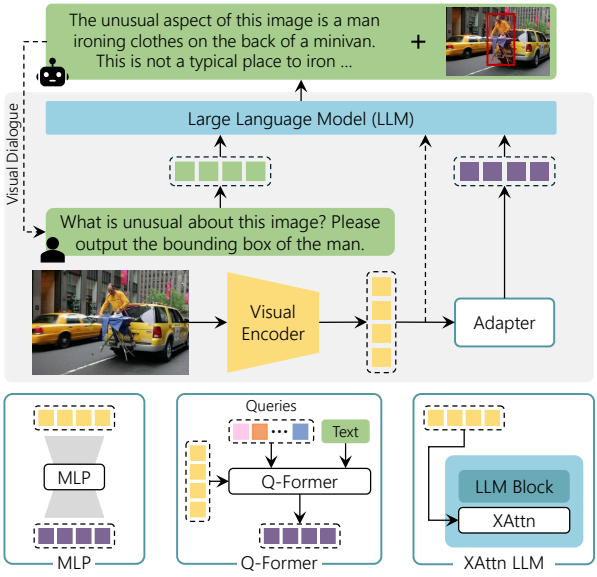


Figure 2.2: General architecture of Multimodal Large Language Models (MLLMs), composed of a visual encoder, a language model, and an adapter module that connects visual inputs to the textual space.

standalone task, but rather as one of several possible multimodal outputs that can be elicited through appropriate prompts or instructions.

The advanced reasoning capabilities of these models have also reshaped the nature of captions, which are now typically longer, more expressive, and stylistically diverse. While this shift enables richer descriptions, it simultaneously raises new challenges related to visual grounding, controllability, and evaluation, redefining the role of image captioning in contemporary multimodal systems.

Standard Multimodal LLMs Architecture. Any MLLM contains at least three components: an LLM backbone serving as an interface with the user, one (or more) visual encoders, and one or more vision-to-language adapter modules. Figure 2.2 illustrates the general architecture and the interaction among these components. Popular choices for the LLM backbone often fall into the LLaMA family [266, 267], given that their weights are freely ac-

cessible, they have been trained on public data, and they boast different sizes to accommodate various use cases. In addition, their derivative versions are popular as well, such as Alpaca [259] and Vicuna [54]. The former fine-tunes LLaMA on instructions written using GPT-3, while the latter exploits user-shared conversations with ChatGPT [203]. Alternatives are OPT [317], Magneto [275], MPT [198], and the instruction-tuned [56] or multilingual [292] flavors of T5 [217], an encoder-decoder language model pre-trained for multiple tasks.

The visual encoder is designed to provide LLMs with visual information and the most used ones are CLIP-based architectures [215, 282] whose pre-training objective is the alignment between CLIP embeddings, obtained thanks to a contrastive loss that aligns the correct image-text pairs.

The simultaneous presence of inputs from different modalities emphasizes the need to incorporate a module capable of delineating latent correspondences within these unimodal domains. These modules, termed as "adapters", are intended to facilitate interoperability between the visual and textual domains. A spectrum of different adapters are used in common MLLMs, ranging from elementary architectures such as linear layers or MLP to advanced methodologies such as Transformer-based solutions, exemplified by the Q-Former model, and conditioned cross-attention layers added to the LLM.

Training Startegy. Starting from a pre-trained LLM, the training of an MLLM undergoes a single-stage or a two-stage process. In both cases, a standard cross-entropy loss is utilized for predicting the next token, serving as an auto-regressive objective. The most popular approach is the two-stage process (see Figure 2.3), where in the first of the two training stages, the objective is to align the image features with the text embedding space. After this stage, the outputs tend to be fragmented and not coherent. Therefore, a second step is done to improve multimodal conversational capabilities.

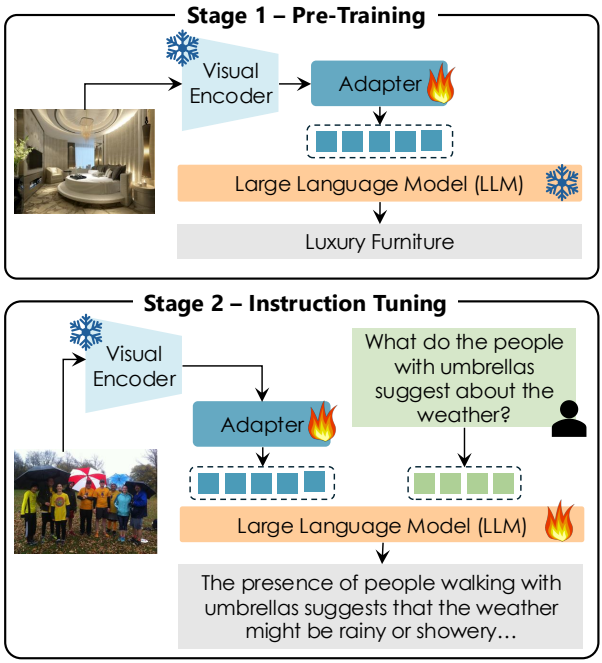


Figure 2.3: Standard LLaVA framework with a two-stage training process. The first stage aligns the visual features to the underlying LLM, ensuring effective cross-modal representation. The second stage enhances the MLLM conversational capabilities through visual instruction tuning. Following this paradigm, we systematically compare different LLM and visual encoder choices to evaluate their impact on various multimodal tasks.

LLaVA [168, 166] is among the first to introduce a visual instruction-following training scheme, which is performed as a second training stage updating the parameters of both the multimodal adapter and LLM. During the first stage, instead, only the multimodal adapter is trainable.

Training Datasets. During the first (or single) training stage, the datasets predominantly consist of large-scale, publicly available, and uncurated data. For instance, the Conceptual Captions 3M (CC3M) dataset [241] is composed of 3M images paired with textual captions specifically designed for image captioning systems. Unlike the widely-used and curated MS-COCO [158] dataset, which serves similar purposes, images and cap-

tions in CC3M are gathered from the web, showcasing a broader spectrum of styles and content. Similarly, the LAION family [236, 235] represents an extended collection of non-curated image-text pairs sourced from web pages, providing a rich resource for pre-training multimodal language models. Additionally, the COYO-700M [26] dataset stands out as a significant resource, containing 747M image-text pairs. Notably, each alt-text in COYO-700M is linked to an image within HTML documents. Furthermore, DataComp [78] presents an extensive pool of 12.8B filtered image-text pairs sourced from common crawl.

It is important to highlight the distinction between datasets used in the initial phase of training, which typically comprise large-scale, uncurated data, and those selected for refinement in next stages. While the former emphasizes diversity and scale, the latter focuses on specificity and task relevance, facilitating a more tailored approach to model optimization.

2.2 Evaluation Benchmarks for V&L Models

2.2.1 Image Captioning Benchmarks

A variety of datasets have been introduced over the years to evaluate image captioning models rigorously and support the development of general vision–language models. These benchmarks differ in scale, annotation quality, and domain coverage. Below, we summarize the most widely adopted resources in the field.

COCO [158]. This benchmark can be considered as the reference evaluation benchmark for image captioning models [251] as well as the largest dataset with human-collected annotations for the task. In particular, COCO contains more than 120,000 images, each of them manually annotated with five textual captions collected by using Amazon Mechanical Turk. Usu-

ally, the dataset is employed by following the splits defined by Karpathy *et al.* [117], where the training set is composed of 82,783 images while both the validation and test set contain 5,000 images.

CC3M [241]. Conceptual Captions 3M is a large-scale dataset designed to support vision–language pretraining through noisy but diverse web-scraped image–text pairs. Unlike COCO, which provides human-written captions, CC3M captions are automatically extracted from the alt-text metadata of images crawled from the web, followed by a series of filtering and cleaning steps aimed at removing personal information and overly descriptive or low-quality text. The dataset contains approximately 3 million images with a single associated caption each, offering broad visual and linguistic coverage at the cost of increased noise.

nocaps [3]. This dataset has been introduced for the novel object captioning task where the goal is to effectively describe objects not present in the image-caption pairs used to train the captioning model. The dataset contains 4,500 validation and 10,600 test images from the Open Images V4 dataset [129], where each image has been annotated with 10 human-written captions. Images can be further grouped into three subsets depending on their nearness to COCO (*i.e.* in-domain, near-domain, and out-of-domain images). Specifically, in-domain images only contain objects that are also present in the original COCO dataset, out-of-domain images exclusively contain object classes that are not present in COCO and thus represent the most challenging evaluation set, while near-domain images

2.2.2 Multimodal LLMs Benchmarks

MLLMs are evaluated across different benchmarks, taking into account both more classic visual comprehension and recognition skills and advanced multimodal conversation capabilities. One of the most important skills of MLLMs is their ability to effectively answer questions based on the

given input image. This ability is quantitatively evaluated across several visual question-answering datasets, measuring the accuracy [9] of the answers provided by the MLLM. In the following, we will present just some of the most important benchmarks.

VQAv2 [87]. is an extended and balanced version of VQA [9] built by collecting similar images for the same question, but whose answer is different compared to the original one. This makes it difficult to perform favorably for those models that ignore visual information and only rely on language priors while answering questions.

GQA [111]. is based on Visual Genome scene graph annotations [128] and comprises 113k images and 22M questions focusing on scene understanding and compositionality.

OKVQA [187]. is a benchmark to study how vision-and-language models can address visual questions whose answers cannot be completely found in the image, encouraging systems that also rely on external knowledge. The test set has 14,055 open-ended questions.

VizWiz [91]. originates from authentic situations involving individuals with visual impairments who have taken images and articulated accompanying inquiries about them, together with 10 responses. The validation split consists of 4,319 images paired with their corresponding questions, while the test split encompasses roughly 8,000 instances.

ScienceQA (SQA) [182]. evaluates models over challenging multimodal multiple-choice questions about 3 subjects (*i.e.*, natural science, language science, and social science), 26 topics, 127 categories, and 379 skills. Each question is annotated with explanations linked to relevant lectures. The test set includes 4,241 examples.

Visual Spatial Reasoning (VSR) [163]. contains images from COCO, each paired with a caption mentioning two concepts and the spatial relation between them. Models have to choose if a given caption is true or false

according to the picture. MLLMs are typically evaluated on the 616 samples from the zero-shot test split.

IconQA [183]. tests the visual reasoning abilities of vision-and-language models on three types of questions: multiple-image-choice, multiple-text-choice, and fill-in-the-blank. The dataset stems from real-world problems found in math textbooks and focuses on abstract images (*i.e.*, icons). There are 107,439 questions, 20% of which makes up for the test split.

TextVQA (VQA^T) [248]. is a dataset based on pictures from Open Images [130] and challenges OCR capabilities of vision-and-language models. The test set comprises 5,734 examples.

OCR-VQA [197]. presents a new task in visual question answering by interpreting text within images and involves a collection of 207,572 images of book covers, accompanied by more than 1M question-answer pairs.

POPE [151]. is a valuable benchmark for evaluating object hallucination challenges within MLLMs. It encompasses several distinct subsets, namely random, popular, and adversarial, which are generated utilizing a variety of sampling methodologies. Cumulatively, it is a binary classification query dataset that comprises 8,910 entries, facilitating comprehensive investigations into the phenomenon of object hallucination in MLLMs.

Comprehensively describing the visual input is another important skill desired in MLLMs. To evaluate this, various image captioning datasets are commonly employed. As regards the evaluation metric, the CIDEr score [271], which is the reference metric for the task, is used to compare generated image descriptions with ground-truth captions. In this case the majority of datasets employed are those presented in Section 2.2.1.

Thoroughly evaluating MLLMs is challenging and remains an open frontier. While evaluating on standard datasets represents a valid choice, many benchmarks designed for MLLMs have been recently proposed. They require very strong perception and cognitive skills to succeed, and often

they query for deep domain-specific knowledge. Among these efforts, the **Cambrian** evaluation suite [263] stands out as a unifying framework. It is a comprehensive benchmark comprising 16 tasks spanning four categories: General, Knowledge, OCR, and Vision-Centric.

Vision-Centric. In this category, Cambrian comprehends RealWorldQA [288], which evaluates commonsense reasoning based on visual inputs; MMVP [264], which probes the visual perception skills of a model across nine classes of questions; Blink [77] and CVBench2D [158, 324], which focus on questions concerning spatial relationships; and CVBench3D [21], which evaluates the ability of a model to assess the relative depth of objects from the camera, as well as the relative distance between objects.

General. The datasets in this category collectively evaluate perception and scene understanding through tasks such as quantification, color identification, and multi-domain visual comprehension. Specifically, it includes MME [75], GQA [111], MMBench (MMB) [173], and SEED-Bench (SEED) [143].

Knowledge. This group measures factual and discipline-specific reasoning, testing models on science, mathematics, and diagram understanding that require textual and visual knowledge. It comprises ScienceQA (SQA) [182], MMMU [311], MathVISTA [181], and AI2D [119].

OCR. It encompasses ChartQA [188], OCRBench [167], and TextVQA [248]. These benchmarks focus on recognizing and reasoning over embedded text and numerical information in images, assessing OCR accuracy and text-grounded visual reasoning.

3

Retrieval-Augmented V&L Models: Background

As large-scale language models become bigger, they gain the ability to retain more information from their training data which leads to improved performance on a variety of downstream tasks [216, 23]. This suggests that enhancing models with retrieval, thus fostering their memorization capabilities, may lead not only to further improvements but also to savings in model size.

3.1 Retrieval for Image Captioning

In the past, image [74, 65, 313, 244, 162] and video [243] tagging has been recognized as a successful practice to boost relevance matching for information retrieval. In fact, tagging is a mechanism for assigning a set of text labels (*e.g.* keywords or terms) to an image or a video, and can be treated as anchors to guide the visual-language alignment more explicitly. Some vision-and-language methods used tags as an additional input to boost the final performance [305, 84, 150, 63]. However, predicted tags may be incomplete, inconsistent, and sparse, especially when compared to sentences, longer paragraphs, or entire documents that usually contain more complete information.

Recently, the same idea has been applied to language models, gaining significant interest [194]. To integrate knowledge into a language model, this line of work retrieves, from an external memory, items that are related to the input, either from a single modality [20, 121, 94] or from multimodal documents [45, 44, 107], allowing the language model to exploit them and generate more accurate predictions. Approaches such as REALM [94], RAG [141], and RETRO [20] integrate Wikipedia passages and other web-scale sources as external memory to benefit downstream knowledge-intensive tasks as, for example, question answering. Some works train the retrieval model via contrastive learning [118], while others [93] train a single-document retriever by concatenating each retrieved result with the query, to compute the final loss independently. A recent work [107], instead, focuses on multimodal tasks and proposes to incorporate the retrieval scores directly into an attentive fusion module, allowing the gradients to be backpropagated through the retriever component.

External memories have been used in different ways in Transformer-based architectures, mainly in NLP. Khandelwal *et al.* [121] constructed a

memory for language generation as a large table of (key, token) pairs, while Sukhbaatar *et al.* [252] replace feed-forward layers with differentiable memory slots. Recently, Wu *et al.* propose a Memorizing Transformer [287] architecture in which they retrieve activations produced over long documents. In vision-and-language, learnable external memories have been successfully employed for image captioning [64], visual relationship recognition [39], and story generation [291].

3.2 Knowledge-Augmented Multimodal LLMs

Multimodal Large Language Models (MLLMs) [4, 168, 166, 13] unify tasks involving multiple modalities, such as text, images and videos [28, 233]. Many of these tasks can be framed as Visual Question Answering (VQA) [87, 9, 285], where a query may require understanding visual content, and the model must generate a faithful, correctly formatted response. Despite their broad pre-training corpora, even state-of-the-art MLLMs struggle with underrepresented, domain-specific queries [48, 191]. This problem, known as Knowledge-based Visual Question Answering (KB-VQA) [187], is commonly addressed by enriching MLLMs with domain-specific information from external sources via *i.e.*, Retrieval-Augmented Generation (RAG) [141].

3.2.1 Knowledge-Based Visual Question Answering

In the standard VQA task, a multimodal LLM, referred to as the generator model $\mathcal{G}$, must answer a question q about an image I_q. The task requires the model to understand the visual content and provide a correct answer. While MLLMs large-scale pretraining captures general knowledge, it may be insufficient for answering highly specific or domain-specific questions.

Knowledge-based VQA extends VQA by incorporating external knowl-

edge. In our setting, the external knowledge base $\mathcal{KB}$ is a collection of N multimodal documents (*e.g.*, Wikipedia pages) each containing textual passages and images. Formally, it can be represented as

$$\mathcal{KB} = \{d_1, \ldots, d_N\}, \quad d_i = (\mathcal{T}_i, I_i, \mathcal{P}_i), \tag{3.1}$$

where $\mathcal{T}_i$ is the metadata of the i-th document (*e.g.*, title and summary of a Wikipedia page), I_i is the associated image, if present, and $\mathcal{P}_i$ are the textual passages of the document.

A retrieval model $\mathcal{R}$ is employed to select the top-k relevant documents from $\mathcal{KB}$ and their associated passages $\tilde{\mathcal{P}} = \{p_0, \ldots, p_j\}$, which are then provided within the generator context window. Finally, the generator produces an answer A conditioned on both the image, the question, and the retrieved passages, as follows:

$$A \sim \mathcal{G}(A \mid q, I_q, \{p_0, \ldots, p_j\}). \tag{3.2}$$

During training, the generator $\mathcal{G}$ is optimized to maximize the likelihood of producing the correct answer given the visual and textual context. In particular, the retrieved passages $\tilde{\mathcal{P}}$ act as external conditioning signals that augment the understanding of the model of the visual scene and the question, enabling knowledge-grounded reasoning. This augmented context enables more accurate and informed reasoning, especially for queries that require factual or domain-specific information. The objective can thus be expressed as the negative log-likelihood of the ground-truth answer tokens, averaged over the training distribution, *i.e.*

$$\mathcal{L}(\theta) = \mathbb{E}_{(I_q, q, \tilde{\mathcal{P}}) \sim \mathcal{D}} \left[\frac{1}{|y|} \sum_{t=1}^{|y|} \log \mathcal{G}_\theta(y_t \mid q, I_q, \tilde{\mathcal{P}}, y_{<t}) \right], \tag{3.3}$$

where $\mathcal{D}$ denotes the training distribution.

3.2.2 Benchmarks for Knowledge-Intensive VQA

Early datasets [187, 237, 239] targeted specialized reasoning. However, with the advent of more powerful MLLMs, these datasets have become insufficient for evaluating performance in realistic and knowledge-intensive settings. To address this, benchmarks such as Encyclopedic-VQA [191] and InfoSeek [48] introduce more challenging, Wikipedia-scale scenarios requiring fine-grained and entity-specific reasoning over external knowledge, making retrieval essential.

Encyclopedic-VQA [191]. It contains 221k question-answer pairs, each linked to up to five images and covering 16.7k fine-grained entities. The questions are categorized into *single-hop* and *two-hop* types: single-hop questions can be answered using information from a single Wikipedia page, whereas two-hop questions require sequential retrieval across multiple pages. The dataset is divided into training, validation, and test splits comprising 1M, 13.6k, and 5.8k samples. E-VQA contains an external knowledge base derived from Wikipedia, consisting of approximately 2M pages. Each comprises the article title, its textual sections, and associated images.

InfoSeek [48]. It consists of approximately 1.3M image-question-answer triplets corresponding to around 11k distinct Wikipedia pages. It is partitioned into training, validation, and test splits, containing roughly 934k, 73k, and 348k samples. Both validation and test sets include questions about unseen entities. InfoSeek provides an external knowledge base of around 6M Wikipedia entities.

Evaluation Metrics. For E-VQA, generated answers are evaluated using the BERT-based matching score (BEM) [25], which measures semantic similarity of predicted and ground-truth answers. For InfoSeek, evaluation depends on the question type: we employ standard VQA accuracy [87] as well as relaxed accuracy [193].

3.3 Benefits and Limitations of Retrieval-Augmented Models

Retrieval-augmention extend purely parametric models by incorporating an external knowledge source that can be dynamically queried at inference time. This mechanism allows the model to access **up-to-date** and potentially **unbounded** information without requiring it to be stored in the model parameters. As a result, retrieval-augmented systems often outperform purely parametric models on knowledge-intensive and open-domain tasks, since they can ground their predictions in retrieved evidence rather than relying solely on memorized representations. From a scientific perspective, this improvement stems from the complementary roles of parametric and non-parametric memory: while the model encodes general capabilities and multimodal alignment, the retrieval component provides explicit access to factual or contextual information that may not be reliably encoded in model weights.

Despite these advantages, retrieval can also negatively affect performance in certain conditions. When the retrieved items are noisy, irrelevant, or semantically mismatched with the input query, they may introduce misleading context that degrades model predictions. Domain mismatch between the retrieval corpus and the target task can further reduce the usefulness of retrieved information. Additionally, retrieval introduces practical deployment challenges, including increased latency due to search operations and scalability constraints when operating over very large multimodal databases. Efficient indexing strategies, filtering mechanisms, and query-aware retrieval policies are therefore critical to ensure that the benefits of retrieval outweigh its potential drawbacks.

From a deployment perspective, the scalability of retrieval-augmented systems depends on both the efficiency of the retrieval backend and the

interaction mechanism between the retriever and the MLLM. While large-scale vector databases enable fast similarity search, maintaining low latency and high retrieval quality remains challenging in real-world applications. Designing lightweight retrieval pipelines and adaptive retrieval strategies – where retrieval is triggered only when necessary – can help mitigate these issues and make retrieval-augmented MLLMs more practical for large-scale deployment.

4

Learning to Retrieve: Models and Strategies for Effective Retrieval

Retrieving relevant information in response to an input query is a fundamental problem in Computer Vision and multimedia research, and it has been extensively studied for several decades. Early work in retrieval primarily focused on **unimodal** settings, where both the query and the searchable items belong to the same modality – either images

This chapter discusses topics from the following papers: D. Caffagni*, S. Sarto* *et al.*, "Recurrence-Enhanced Vision-and-Language Transformers for Robust Multimodal Document Retrieval", CVPR 2025 [30] and D. Caffagni, S. Sarto *et al.*, "Recurrence Meets Transformers for Universal Multimodal Retrieval", under submission [31].

or natural language – leading to appli cations such as image similarity search and text-based document retrieval [112, 323, 201, 214].

In the visual domain, image retrieval has been studied for decades, evolving from approaches based on hand-crafted local descriptors to deep convolutional representations, and more recently to Transformer-based architectures such as Vision Transformers (ViTs) [70]. In parallel, advances in large-scale language modeling have dramatically improved text-based retrieval systems: as language models grow in scale, they increasingly capture rich semantic information from vast training corpora, leading to substantial gains across a wide range of downstream tasks [216, 23]. This suggests that enhancing models with retrieval, thus fostering their memorization capabilities, may lead not only to further improvements but also to savings in model size.

As retrieval systems have matured, increasingly complex retrieval scenarios have emerged, including composed image retrieval [175], long-text-to-image retrieval [315], and multimodal query-to-multimodal document retrieval [48, 104]. These settings require models to jointly reason over multiple inputs, modalities, and contextual cues, going beyond simple similarity matching. Current approaches typically rely on task-specific architectures or specialized fine-tuning strategies [196, 22, 14]. However, designing a unified retrieval framework that can seamlessly accommodate heterogeneous queries and documents across modalities remains a central and largely open research challenge.

This challenge is further amplified by a broader shift toward **multimodal** data in real-world applications [158, 241, 235]. Modern retrieval systems are increasingly expected to interpret natural language queries to retrieve visual content—or conversely, to use visual inputs to access textual information [215]. Looking ahead, this trend naturally extends beyond image–text pairs, motivating retrieval engines capable of operating over richer modality sets, including video and audio [85].

4.1 Retrieval Foundations: Standard Settings and Baseline Models

In this context, the next chapter introduces CLIP as a unifying architecture that bridges visual and linguistic representations, enabling classical retrieval while also motivating the development of specialized architectures for increasingly complex multimodal scenarios.

4.1.1 CLIP as a Retrieval Backbone

Classical retrieval methods were largely unimodal, focusing on either text-based document search or content-based image retrieval [112, 323, 201]. While effective in their domains, they lacked the ability to bridge modalities. The advent of large-scale vision-language datasets [241, 235] and dual-encoder models such as CLIP [215] and its variants [314, 254, 268] marked a turning point, enabling contrastive learning to align images and text in a shared embedding space.

Revisiting CLIP Architecture. Contrastive Language-Image Pre-training (CLIP) focuses on learning rich visual and textual representations by understanding the relationships between images and their corresponding textual descriptions. CLIP employs an image encoder $E_v(\cdot)$ (*e.g.* a CNN [96] or a ViT [71]) along with a text encoder $E_t(\cdot)$ (*e.g.* a Transformer model [270]) to obtain visual and textual representations. The multimodal interaction is performed via late fusion by projecting the output of both encoders to the same dimension and then on the ℓ_2 hypersphere via normalization. The visual and the textual inputs can then be compared via cosine similarity.

During the training phase, CLIP utilizes a contrastive objective to encourage similar embeddings for matched image-text pairs and dissimilar embeddings for non-matched pairs. In a batch of N image-caption pairs

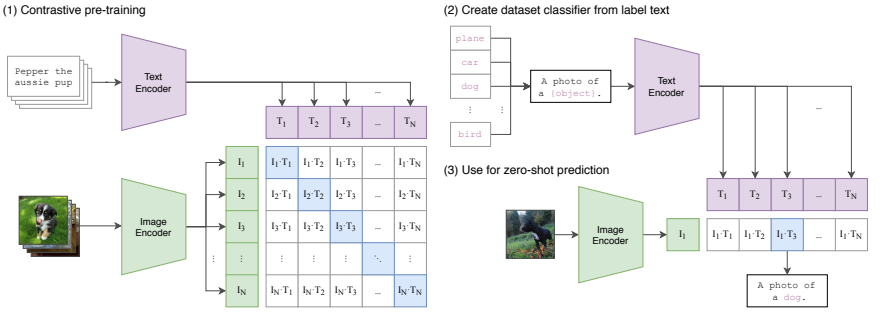


Figure 4.1: CLIP Architecture [215]. While standard image models jointly train an image feature extractor and a linear classifier to predict some label, CLIP jointly trains an image encoder and a text encoder to predict the correct pairings of a batch of (image, text) training examples. At test time the learned text encoder synthesizes a zero-shot linear classifier by embedding the names or descriptions of the target dataset's classes.*

$\{(v_i, t_i)\}_{i=1}^{N}$, CLIP employs the InfoNCE loss [202] that can be written as:

$$\mathcal{L}_{\mathcal{V},\mathcal{T}} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(\text{sim}(v_i,t_i)/\tau)}{\sum_{j=1}^{N}\exp(\text{sim}(v_i,t_j)/\tau)} + \\ -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(\text{sim}(v_i,t_i)/\tau)}{\sum_{j=1}^{N}\exp(\text{sim}(v_j,t_i)/\tau)}. \tag{4.1}$$

Here, the similarity function is defined as:

$$\text{sim}(v,t) = \cos(\text{Norm}(E_v(v)), \text{Norm}(E_t(t))),$$

where $\text{sim}(\cdot)$ is the CLIP-based cosine similarity between visual and textual inputs that are normalized via ℓ_2 normalization, and τ is a temperature parameter to scale the logits. With the symmetrical loss applied to both image and text encoders, the overall loss function $L_{\mathcal{V},\mathcal{T}}$ is computed as the average of the two. The overall architectures is shown in Figure 4.1.

Large-scale contrastive models like CLIP are trained using image-caption pairs collected from the web. These provide a large-scale source of supervision for learning scalable low-level and semantic visual and tex-

* Figure reproduced from the original CLIP paper [215].

tual features, as testified by their zero-shot classification performance and by their adaptability to different tasks [219, 189, 120].

CLIP for Cross-Modal Retrieval. Given CLIP's architecture and training strategy, which explicitly align visual and textual representations in a shared embedding space, it is a natural choice for cross-modal retrieval tasks. The large-scale contrastive pretraining and strong generalization capabilities of CLIP make it particularly well suited for retrieving semantically related content across modalities. In the standard cross-modal retrieval setting, inputs from different modalities—most commonly images and text—are independently encoded and projected into a common embedding space, where semantic similarity can be measured directly. Retrieval is formulated as a ranking problem: given a query in one modality, the system retrieves the most relevant items in the other modality by computing similarity scores, typically using cosine similarity between embeddings.

This formulation naturally supports both **text-to-image (T2I)** retrieval, where a textual query is used to retrieve relevant images, and **image-to-text (I2T)** retrieval, where an image serves as the query to retrieve corresponding textual descriptions. Thanks to its unified representation space, CLIP enables both retrieval directions within a single framework, without requiring task-specific architectures or additional supervision.

In Table 4.1, reproduced from the original CLIP paper [215], the authors evaluate zero-shot image–text retrieval performance on standard benchmarks such as Flickr30k and MS-COCO as a sanity check for the pretraining objective. The results show that, despite being trained on large-scale and noisy web data, CLIP achieves strong zero-shot transfer. In particular, zero-shot CLIP matches or outperforms previous zero-shot approaches on both text-to-image and image-to-text retrieval on Flickr30k, and remains competitive with state-of-the-art methods for text retrieval on this dataset. Although performance on image retrieval is comparatively weaker, zero-

Table 4.1: CLIP improves zero-shot retrieval and is competitive with the best fine-tuned result on Flickr30k text retrieval. Bold indicates best overall performance while an underline indicates best in category performance (zero-shot or fine-tuned). For all other models, best results from the paper are reported regardless of model size / variant. MSCOCO performance is reported on the 5k test set.

		Text Retrieval						Image Retrieval					
		Flickr30k			MSCOCO			Flickr30k			MSCOCO		
		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Finetune	Unicoder-VL [145]	86.2	96.3	99.0	62.3	87.1	92.8	71.5	90.9	94.9	46.7	76.0	85.3
	Uniter [50]	87.3	98.0	99.2	65.7	88.6	93.8	75.6	94.1	**96.8**	52.9	79.9	88.0
	VILLA [80]	87.9	97.5	98.8	-	-	-	76.3	**94.2**	**96.8**	-	-	-
	Oscar [150]	-	-	-	**73.5**	**92.2**	**96.0**	-	-	-	**57.5**	**82.8**	**89.8**
	ERNIE-ViL [308]	**88.7**	98.0	99.2	-	-	-	**76.7**	93.6	96.4	-	-	-
Zero-Shot	Visual N-Grams [142]	15.4	35.7	45.1	8.7	23.1	33.3	8.8	21.2	29.9	5.0	14.5	21.9
	ImageBERT [211]	-	-	-	44.0	71.2	80.4	-	-	-	32.3	59.0	70.2
	Unicoder-VL [145]	64.3	86.8	92.3	-	-	-	48.4	76.0	85.2	-	-	-
	Uniter [50]	83.6	95.7	97.7	-	-	-	68.7	89.2	93.9	-	-	-
	CLIP [215]	88.0	**98.7**	**99.4**	58.4	81.5	88.1	68.7	90.6	95.2	37.8	62.4	72.2

shot CLIP still performs on par with several fine-tuned models, highlighting the effectiveness of contrastive pretraining for cross-modal retrieval.

Overall Considerations. CLIP demonstrates remarkable efficiency and scalability by learning from large, diverse, and noisy image–text datasets using a contrastive learning objective. This design enables strong zero-shot performance across a wide range of tasks without task-specific fine-tuning, while the use of Vision Transformers further improves training efficiency, making large-scale models practical to train. Its ability to learn visual concepts directly from natural language grants CLIP exceptional flexibility and generalization: it can perform tasks ranging from object classification and action recognition to geo-localization and OCR, often outperforming traditional ImageNet-trained models in standard representation learning evaluations.

Despite these strengths, CLIP exhibits notable limitations. It struggles with tasks requiring precise reasoning or abstract understanding, such as counting objects, estimating spatial relationships, or performing fine-

grained classification of closely related categories like car models or aircraft variants. Its performance is also constrained by the quality and coverage of the pretraining data. Furthermore, CLIP's zero-shot predictions can be sensitive to the phrasing of textual inputs, often necessitating careful prompt engineering. Finally, the model's generalization to out-of-distribution images remains limited, highlighting ongoing challenges in robustness and adaptability.

Despite this considerations, CLIP model and variants are typically evaluated on relatively small benchmarks for retrieval like Flickr30k [210] and COCO [158], which emphasize simple queries and limit generalization.

4.2 Beyond Standard Retrieval: Complex Multimodal Scenarios

Beyond standard image-to-text and text-to-image retrieval, retrieval settings have grown increasingly complex. Emerging scenarios – such as composed image retrieval or multimodal queries over multimodal documents – require models to jointly reason over multiple modalities rather than relying on simple similarity matching. This capability becomes especially critical when retrieving information from very large and heterogeneous corpora, such as collections of Wikipedia pages, where relevant content may span text, images, and other modalities.

4.2.1 Multimodal Retrieval: Problem Settings

In multimodal retrieval, both queries and documents may consist of multiple modalities, typically text and images. A query q can be purely textual q^T, purely visual q^V, or a combination of both (q^T, q^V), while retrieval candidates d may similarly contain text d^T, images d^T, or paired image–text

content (d^T, d^V). This setting generalizes traditional cross-modal retrieval by allowing flexible interactions between heterogeneous query and document representations, enabling retrieval over rich multimodal corpora.

In a universal multimodal search engine, users initiate various search tasks based on their specific needs. Therefore, usually, the textual query is also associated or correspond to language instruction q_{inst} to represent the intention of the retrieval task. This instruction defines what the search aims to find, whether seeking images, text, or a mix of both, and specifies the relevant domain. Formally, the goal is to build a unified retriever model capable of taking any type of query q to retrieve any type of target d.

4.2.2 Multimodal Fusion Strategies

For multimodal scenarios, the UniIR paper [280] experimented with two multimodal fusion mechanisms, namely score-level fusion and feature-level fusion [104, 170]. To explore the effectiveness of these approaches, they adapted pre-trained models such as CLIP [215] and BLIP [148].

Score-level Fusion. As illustrated in Figure 4.2(a), the score-level fusion variants for CLIP and BLIP (denoted as CLIP_{SF} and BLIP_{SF}) employ distinct encoders for vision and text. Specifically, the vision encoder is marked as f_i and the uni-modal text encoder as f_t. In these methods, both image and text inputs (whether from a query or a target) are processed into two individual vectors. These vectors undergo a weighted sum to form a unified representation vector. This process is mathematically represented as $f(q_{\mathrm{i}}, q_{\mathrm{t}}, q_{\mathtt{inst}}) = w_1 f_I(q_{\mathrm{i}}) + w_2 f_T(q_{\mathrm{t}}, q_{\mathtt{inst}})$ for queries and $f(c_{\mathrm{i}}, c_{\mathrm{t}}) = w_3 f_I(c_{\mathrm{i}}) + w_4 f_T(c_{\mathrm{t}})$ for targets. Therefore, the similarity score between a query $\mathbf{q}$ and a target $\mathbf{c}$ is calculated as a weighted sum of the within-

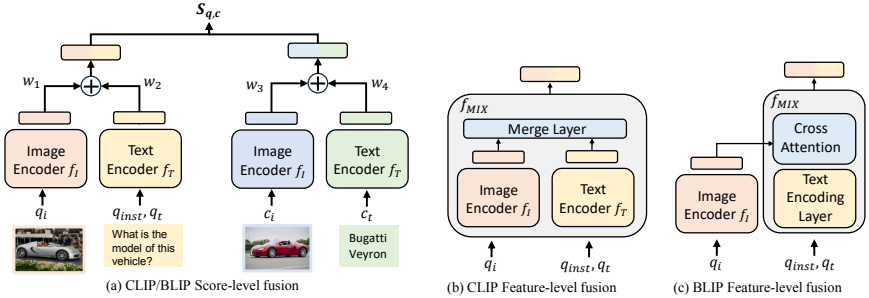


Figure 4.2: (a) Score-level fusion encodes each modality into a single feature; (b) CLIP feature-level fusion (CLIP_{FF}) fuses two modalities into a single feature with a mix-modality transformer layer; (c) BLIP feature-level fusion (BLIP_{FF}) adopts cross-attention to output a single feature vector.*

modality and cross-modality similarity scores:

$$\begin{aligned} s_{\mathbf{q},\mathbf{c}} &= f(q_{\text{i}}, q_{\text{t}}, q_{\texttt{inst}})^T \cdot f(c_{\text{i}}, c_{\text{t}}) \\ &= w_1 w_3 f_I(q_{\text{i}})^T f_I(c_{\text{i}}) + w_2 w_4 f_T(q_{\text{t}}, q_{\texttt{inst}})^T f_T(c_{\text{t}}) \\ &+ w_1 w_4 f_I(q_{\text{i}})^T f_T(c_{\text{t}}) + w_2 w_3 f_T(q_{\text{t}}, q_{\texttt{inst}})^T f_I(c_{\text{i}}). \end{aligned} \tag{4.2}$$

w_1, w_2, w_3, w_4 are learnable parameters that reflects importance weights.

Feature-level Fusion. Contrasting the approach of processing uni-modal data separately, feature-level fusion integrates features during the encoding phase. This fusion method computes a unified feature vector for multi-modal queries or candidates using mixed-modality attention layers. As illustrated in Figure 4.2 (b), for the CLIP feature-level fusion (CLIP_{FF}), we have enhanced the pre-trained vision encoder f_I and text encoder f_T with a 2-layer Multi-Modal Transformer, which follows the same architecture as T5 Transformer, forming a mixed-modality encoder f_{MIX}. In the case of BLIP feature-level fusion (BLIP_{FF}), the process begins with the extraction of image embeddings through the vision encoder f_I. These embeddings are then integrated with text embeddings through the cross-attention layers of BLIP's

* Figure reproduced from the original UniIR paper [280].

image-grounded text encoder, also labeled as f_{MIX}. In both CLIP_{FF} and BLIP_{FF}, the output from f_{MIX} is a comprehensive feature vector that combines information from both image and text modalities. The final representations for the query and target, denoted as $f_{\text{MIX}}(q_{\text{i}}, q_{\text{t}}, q_{\texttt{inst}})$ and $f_{\text{MIX}}(c_{\text{i}}, c_{\text{t}})$ respectively, are obtained separately but using the same f_{MIX}. The similarity score between the query and the target is then calculated by:

$$s_{\mathbf{q},\mathbf{c}} = f_{\text{MIX}}(q_{\text{i}}, q_{\text{t}}, q_{\texttt{inst}})^T \cdot f_{\text{MIX}}(c_{\text{i}}, c_{\text{t}}) \tag{4.3}$$

4.2.3 Recurrence-Augmented Transformers for Multimodal Retrieval

Recent advances in multimodal retrieval [280, 160] have largely focused on leveraging vision-and-language backbones by extracting features from their final layers and fusing them into a single representation for computing query–document similarity. While effective in many scenarios, this design choice implicitly assumes that high-level semantic representations are sufficient to address the full spectrum of retrieval queries.

Differently from existing approaches, we propose **ReT**, a method that explicitly incorporates multi-layer representations from both visual and textual backbones while retaining a late-interaction retrieval paradigm.

We posit that representations from shallower layers encode complementary information that is crucial for addressing diverse retrieval needs. For example, while a retrieval model may easily recognize an image of a well-known basketball player such as Kobe Bryant, low-level details – such as the number on the player's jersey – may be attenuated or lost as information flows through deeper layers that emphasize global semantics. In contrast, shallow layers naturally preserve fine-grained visual cues that can be decisive for certain queries (Figure 4.3).

In addition, unlike standard retrieval models that represent queries and

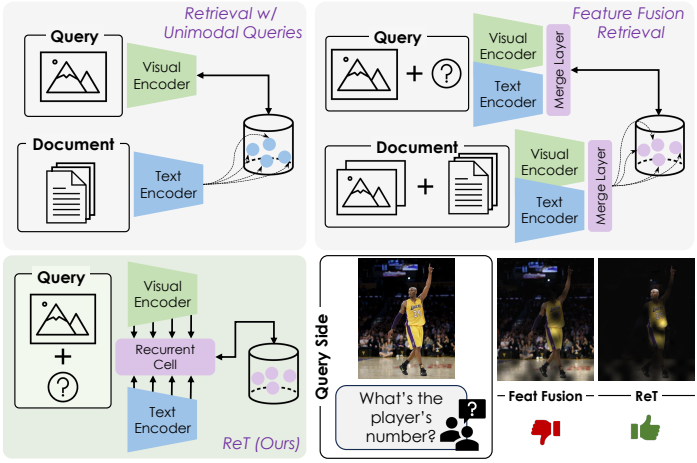


Figure 4.3: Comparison between cross-modal retrieval with unimodal queries (top left), retrieval with multimodal queries with feature fusion [280] (top right), and our ReT (bottom left). Our approach enables retrieval with multimodal queries employing a recurrent, multi-level feature extraction process.

documents as single global vectors and compute relevance via cosine similarity, ReT encodes both queries and documents as sets of tokens and performs late interaction between them. By comparing individual query tokens with document tokens, the model can capture fine-grained local correspondences that are aggregated into a more robust relevance score. This token-level interaction has been shown to be effective in retrieval settings where subtle cues and partial matches play a critical role.

ReT Model Explained. We introduce a novel Transformer-based recurrent cell that progressively integrates information across layers of the visual and textual backbones. Features from textual and visual backbones are considered across multiple layers, thus providing an effective multi-level encoding of each modality composing queries and documents. At each recurrent step, ReT dynamically fuses its internal state with the visual and textual representations obtained from the current layer of the two backbones. Differently from a standard recurrent network applied to a tempo-

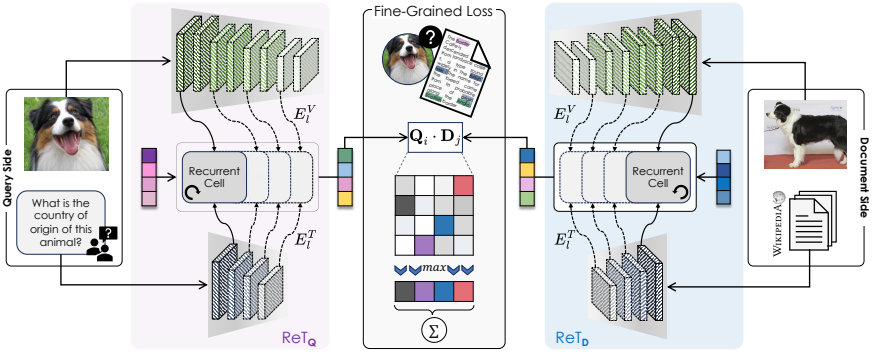


Figure 4.4: Overview of the proposed Recurrence-enhanced Transformer (ReT) for cross-modal retrieval with multimodal queries. Our model employs a Transformer-based recurrent cell to encode multiple vision-and-language layers into hidden vectors for similarity computation.

ral sequence, in ReT the concept of *past* does not pertain to previous time steps, but rather to lower-level representations of a given query or document, obtained from shallower layers. A graphical overview illustrating the architecture of our model is reported in Fig. 4.4.

Specifically, our model consists of dedicated encoders for queries and documents, ReT$_Q$ and ReT$_D$, with identical architecture but distinct learnable weights, optimized jointly. Given that ReT$_Q$ and ReT$_D$ share the same architecture, for simplicity we describe only ReT$_Q$ in the following paragraphs.

In particular, the ReT$_Q$ encoder consists of a recurrent cell and a pre-trained visual and textual backbone. In our notation, we define $E^m(q^m) = \{E_l^m(q^m)\}_{l=1}^L$ the set of activations gathered from each layer l of the unimodal backbone, where $E_l^m(q^m) \in \mathbb{R}^{N \times d}$. Here, $m \in \{T, V\}$ specifies the modality of the backbone, either textual T or visual V, and L represents the total number of layers. Also, we denote the self-attention operator [270] applied to a real-valued matrix $\mathbf{x}$ as $\mathrm{Attention}(\mathbf{x})$. For cross-attention between two matrices, $\mathbf{x}$ and $\mathbf{y}$, we use the notation $\mathrm{Attention}(\mathbf{x}, \mathbf{y})$.

Learnable Gating and Recurrent Cells. Although being a Transformer-based architecture, our model manages the sequence of input layers

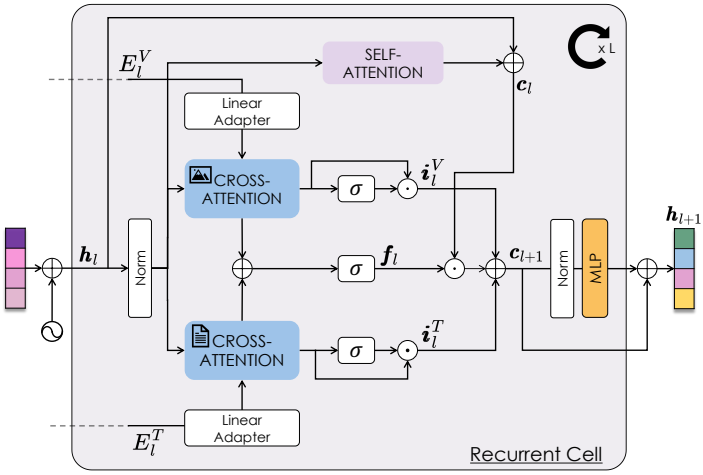


Figure 4.5: Graphical overview of the designed recurrent cell for the proposed retrieval model, which integrates layer-specific textual and visual features into a matricial hidden state.

through a learnable gating mechanism inspired by the LSTM architecture [99]. This design can selectively forget lower-level representations in favor of higher-level features. Similarly, an input gate within each layer modulates the contribution of each modality, balancing the visual and textual inputs and providing a more controllable encoding.

The architecture of the recurrent cell is illustrated in detail in Fig. 4.5. At each layer l, it performs *feature fusion* [280] across three different inputs: the output from the previous step $\mathbf{h}_l \in \mathbb{R}^{k \times d}$, the visual representation from the l-th layer of the visual backbone $\mathbf{E}_l^V$, and the textual representation from the l-th layer of the textual backbone $\mathbf{E}_l^T$. At the first iteration, the hidden state $\mathbf{h}_0$ is initialized to a set of k learnable vectors.

Within the recurrent cell, following a layer normalization [139], the input $\boldsymbol{h}_l$ is fed to three parallel branches. The first branch computes the candidate state $\mathbf{c}_l$ of the recurrent cell, employing a self-attention operation with a residual connection to enhance stability and retain contextual information.

Formally, the candidate state is computed as

$$\mathbf{c}_l = \text{Attention}(\hat{\boldsymbol{h}}_l) + \boldsymbol{h}_l, \tag{4.4}$$

where $\hat{\boldsymbol{h}}_l = \text{LayerNorm}(\boldsymbol{h}_l)$. Given the permutation invariance of the attention operator, a fixed positional encoding [270] is added to $\boldsymbol{h}_l$ before normalization. Notably, for $l \geq 1$, $\boldsymbol{h}_l$ will accumulate layer-specific encodings of the query image and of the text.

To incorporate contextual information from both modalities, two other parallel branches perform feature fusion with the unimodal representations from the l-th layers. Specifically, we employ two independent cross-attention modules to fuse the normalized input $\hat{\boldsymbol{h}}_l$ with the visual and textual representations respectively, as

$$\boldsymbol{z}_l^m = \text{Attention}(\hat{\boldsymbol{h}}_l, E_l^m). \tag{4.5}$$

The three branches are finally combined to compute the new internal state of the recurrent cell, which is obtained as a linear combination of the candidate state $\mathbf{c}_l$ and the outputs from the two feature fusion branches $\boldsymbol{z}_l^T$ and $\boldsymbol{z}_l^V$, modulated by forget and input gates. Conceptually, the forget gate $\boldsymbol{f}_l$ determines how much information to retain from previous applications of the recurrent cell across the shallower layers (*i.e.*, the "past"), given the current multimodal information flow $\boldsymbol{z}_l^m$. In contrast, the input gates $\boldsymbol{i}_l^m$ control the contribution from the l-th layer of each unimodal backbone. This potentially shuts down interferences from high-level representations whenever the user query requires focusing on low-level details (*e.g.*, colors and shapes). Formally, the next candidate state is obtained as

$$\boldsymbol{c}_{l+1} = \mathbf{c}_l \odot \boldsymbol{f}_l + \boldsymbol{z}_l^T \odot \boldsymbol{i}_l^T + \boldsymbol{z}_l^V \odot \boldsymbol{i}_l^V, \tag{4.6}$$

where $\boldsymbol{f}_l$, $\boldsymbol{i}_l^T$ and $\boldsymbol{i}_l^V$ indicate the learnable sigmoidal gates.

In particular, these are computed as follows:

$$\begin{aligned} \boldsymbol{f}_l &= \sigma(\boldsymbol{W}_f^T \cdot \boldsymbol{z}_l^T + \boldsymbol{W}_f^V \cdot \boldsymbol{z}_l^V + b_f), \\ \boldsymbol{i}_l^m &= \sigma(\boldsymbol{W}_i^m \cdot \boldsymbol{z}_l^m + b_i), \end{aligned} \tag{4.7}$$

where $\boldsymbol{W}_f^T, \boldsymbol{W}_f^V, \boldsymbol{W}_i^m$ are trainable weight matrices, and b_f, b_i are fixed scalar biases. The updated state $\boldsymbol{c}_{l+1}$ undergoes layer normalization and is passed through a residual two-layer feed-forward network to produce the output of the recurrent cell, as

$$\boldsymbol{h}_{l+1} = \boldsymbol{c}_{l+1} + \mathrm{MLP}(\mathrm{LayerNorm}(\boldsymbol{c}_{l+1})). \tag{4.8}$$

After going through different layers of the backbones, the output from the last iteration of the recurrent cell, $\boldsymbol{h}_L \in \mathbb{R}^{k \times d}$, consists of a set of latent tokens that serve to compute multiple query-document relevance scores. Specifically, the output $\boldsymbol{h}_L$ is transformed into a different vector space through a linear projection $\mathbf{W}_{final} \in \mathbb{R}^{d \times \overline{d}}$, *i.e.*

$$\overline{\boldsymbol{h}}_L = \boldsymbol{h}_L \cdot \mathbf{W}_{final}. \tag{4.9}$$

Training Procedure. Given a query-document pair (q, d), along with a set of distinct learnable tokens in input for both the query and document sides, we denote the corresponding final output $\overline{\boldsymbol{h}}_L$ of the query and the document encoders as

$$\mathbf{Q} = \mathrm{ReT}_{\mathbf{Q}}(\mathrm{q}) \in \mathbb{R}^{k \times \overline{d}} \tag{4.10}$$

$$\mathbf{D} = \mathrm{ReT}_{\mathbf{D}}(\mathrm{d})^{\intercal} \in \mathbb{R}^{\overline{d} \times k}. \tag{4.11}$$

At training time, these representations are used to compute a fine-grained late-interaction [228] score $s(\mathbf{Q}, \mathbf{D})$ between a query q and a doc-

ument d, which is computed as a sum of maximum similarity values, *i.e.*

$$s(\mathbf{Q}, \mathbf{D}) = \sum_{i=1}^{k} \max_{j=1\ldots k} \mathbf{Q}_i \cdot \mathbf{D}_j. \tag{4.12}$$

Notably, the similarity is implemented as the dot-product between the i-th query and the j-th document token. The use of the $\max$ operator ensures that only the most relevant document tokens are considered for each query token, effectively filtering out those that are locally irrelevant.

Training is performed by jointly optimizing both the query and the document encoder with a symmetric InfoNCE loss [215], where global query-document cosine similarities are replaced with the score defined in Eq. 4.12. A key difference from the common approach of using a symmetric representation for both queries and documents, our encoders optimize two independent sets of learnable parameters, thus allowing for an asymmetric representation.

Limitations of ReT. Our proposal demonstrated strong retrieval performance, validating the effectiveness of recurrent multimodal fusion. However, there remains meaningful room for improvement in both efficiency and efficacy. From a computational perspective, the recurrent nature of the architecture suggests that reducing the number of fusion layers could significantly speed up inference, improving scalability without necessarily sacrificing performance.

Additionally, ReT encodes queries and documents into 32×128 matrices (*i.e.*, $k \times \overline{d}$). We empirically observe that these matrices suffer from rank collapse [116], where their rows converge to a uniform representation, undermining the purpose of leveraging multiple embeddings to capture diverse nuances of the input. This raises the question of whether a single, larger embedding may be more effective than a small embedding matrix for multimodal retrieval.

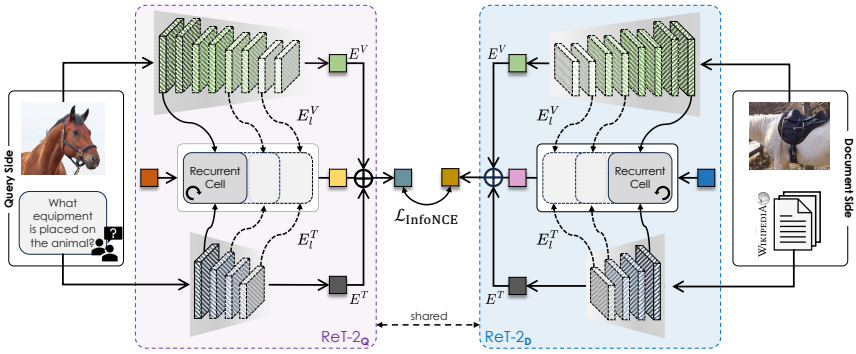


Figure 4.6: Overview of the proposed Recurrence-enhanced Transformer (**ReT-2**) for universal multimodal retrieval.

4.2.4 A Refined Recurrent Transformer for Multimodal Retrieval

In this section, we introduce an enhanced variant of ReT, referred to as **ReT-2**, which is specifically designed to address the limitations identified in the original model. ReT-2 aims to improve retrieval effectiveness and efficiency when dealing with heterogeneous data sources in large-scale, multimodal collections of entities. An architectural overview is shown in Fig. 4.6.

Overall Architecture. In our ReT-2 model, the architecture retains two dedicated encoders for queries and documents. However, in contrast to the previous version (which employed separate parameter sets optimized jointly), ReT-2 introduces a unified encoder architecture with shared weights for both modalities. Specifically, each encoder comprises a recurrent fusion cell coupled with pre-trained, learnable visual and textual backbones. This parameter sharing not only reduces model complexity and reduces overfitting, but also encourages consistent representation learning across queries and documents.

In the following, we retain the notation introduced previously and denote the cross-attention [270] between two matrices $\mathbf{x}$ and $\mathbf{y}$, as Attention$(\mathbf{x}, \mathbf{y})$.

Recurrent Cell. The architecture of the recurrent cell is illustrated in Fig. 4.5. Within the cell, the input hidden state $\boldsymbol{h}_l$ is processed through three parallel branches. The first branch retains the candidate hidden state $\mathbf{c}_l$ of the recurrent cell. Notably, for layers $l \geq 1$, $\boldsymbol{h}_l$ encodes accumulated, layer-specific representations of both the image and text. Rather than processing all layers of the visual and textual backbones, we consistently sample three representative layers: one from the lower (early), one from the middle, and one from the upper (final) sections of each backbone. This approach ensures a balanced capture of low-, mid-, and high-level features while maintaining computational efficiency and architectural compatibility across backbones of varying depth.

To effectively incorporate contextual information from both modalities, the remaining two branches perform feature fusion between $\boldsymbol{h}_l$ and the unimodal visual and textual representations extracted from the l-th layer of their respective backbones.

Specifically, we employ two independent cross-attention modules to fuse the normalized input $\hat{\boldsymbol{h}}_l$ with the visual and textual representations, respectively, as

$$\boldsymbol{z}_l^m = \text{Attention}\left(\hat{\boldsymbol{h}}_l,\, E_l^m\right), \tag{4.13}$$

where $m \in T, V$ and $\hat{\boldsymbol{h}}_l = \text{LayerNorm}(\boldsymbol{h}_l)$ [139].

The outputs of the three branches are combined to compute the updated internal state of the recurrent cell. This state is formed as a gated linear combination of the candidate state $\mathbf{c}_l$ and the outputs from the two feature fusion branches, denoted as $\boldsymbol{z}_l^T$ and $\boldsymbol{z}_l^V$. The combination is modulated by a set of learnable forget and input gates.

In detail, the forget gate $\boldsymbol{f}_l$ controls the extent to which information from earlier applications of the recurrent cell (corresponding to shallower layers, or the "past") is retained in the current step, based on the ongoing multimodal interaction $\boldsymbol{z}_l^m$. In parallel, the input gates $\boldsymbol{i}_l^m$ regulate the influence

of the unimodal features from the current (l-th) layer. This mechanism allows the model to attenuate noisy or less relevant high-level representations when fine-grained visual or textual details (*e.g.*, colors or shapes) are more pertinent to the query. Formally, the next candidate state is

$$\boldsymbol{c}_{l+1} = \mathbf{c}_l \odot \boldsymbol{f}_l + \boldsymbol{z}_l^T \odot \boldsymbol{i}_l^T + \boldsymbol{z}_l^V \odot \boldsymbol{i}_l^V, \tag{4.14}$$

where $\boldsymbol{f}_l$, $\boldsymbol{i}_l^T$ and $\boldsymbol{i}_l^V$ indicate the learnable sigmoidal gates. In particular, these are computed as follows:

$$\begin{aligned} \boldsymbol{f}_l &= \sigma\left(\boldsymbol{W}_f^T \cdot \boldsymbol{z}_l^T + \boldsymbol{W}_f^V \cdot \boldsymbol{z}_l^V + b_f\right), \\ \boldsymbol{i}_l^m &= \sigma\left(\boldsymbol{W}_i^m \cdot \boldsymbol{z}_l^m + b_i\right), \end{aligned} \tag{4.15}$$

where $\boldsymbol{W}_f^T, \boldsymbol{W}_f^V, \boldsymbol{W}_i^m$ are trainable weight matrices, and b_f, b_i are fixed scalar biases.

The updated state $\boldsymbol{c}_{l+1}$ undergoes layer normalization and is passed through a residual two-layer feed-forward network to produce the output of the recurrent cell, as

$$\boldsymbol{h}_{l+1} = \boldsymbol{c}_{l+1} + \mathrm{MLP}\left(\mathrm{LayerNorm}\left(\boldsymbol{c}_{l+1}\right)\right). \tag{4.16}$$

After going through different layers of the backbones, the output from the last iteration of the recurrent cell, $\boldsymbol{h}_L \in \mathbb{R}^{k \times d}$ (where $k = 1$ in our novel formulation), consists of a latent token that serves to compute query-document relevance scores. Specifically, the output $\boldsymbol{h}_L$ is transformed into a different vector space through a linear projection $\mathbf{W}_{final} \in \mathbb{R}^{d \times \overline{d}}$, *i.e.*

$$\overline{\boldsymbol{h}}_L = \boldsymbol{h}_L \cdot \mathbf{W}_{final}. \tag{4.17}$$

Global Feature Injection. At the output of the recurrent cell, $\overline{\boldsymbol{h}}_L$ encodes multimodal information that integrates details from multiple levels of ab-

straction. However, retaining access to the raw global features provides a broader contextual representation of the query or document. To leverage this complementary information, we augment the multimodal representation $\overline{\boldsymbol{h}}_L$ with the unimodal outputs of the visual and textual backbones, denoted as E^V and E^T, respectively. These typically correspond to the CLS visual pooler token and the EOS textual pooler token. The integration is performed by summing the global features with the output of the recurrent cell, obtaining the final representation of the query as

$$\overline{\boldsymbol{h}}_L = \overline{\boldsymbol{h}}_L \ + E^V(q^V) + E^T(q^T). \tag{4.18}$$

Training Procedure. Given a query-document pair (q, d), along with a learnable token in input for both the query and document sides, we denote the corresponding final output $\overline{\boldsymbol{h}}_L$ of the query and the document encoders as

$$\mathbf{Q} = \text{ReT-2}_{\mathsf{Q}}(\mathrm{q}) \in \mathbb{R}^{k \times \overline{d}} \tag{4.19}$$

$$\mathbf{D} = \text{ReT-2}_{\mathsf{D}}(\mathrm{d})^{\intercal} \in \mathbb{R}^{\overline{d} \times k}, \tag{4.20}$$

where $\text{ReT-2}_{\mathsf{Q}} = \text{ReT-2}_{\mathsf{D}}$ in our shared implementation.

Training is performed by optimizing both the query and document encoder with the InfoNCE loss [215], where query-document similarity is computed as the dot-product between the query and the document token.

From ReT to ReT-2. Overall, ReT-2 incorporates several targeted changes to enhance the efficiency, robustness, and simplicity of the original ReT architecture. A visual summary of the modifications and improvements implemented in ReT-2 is provided in Fig. 4.7. Specifically:

- **Shared Query-Document Encoder**: Unlike ReT, which used separate encoders with distinct learnable parameters for queries and documents, ReT-2 adopts a shared architecture with tied weights, promoting consistency and reducing model complexity.

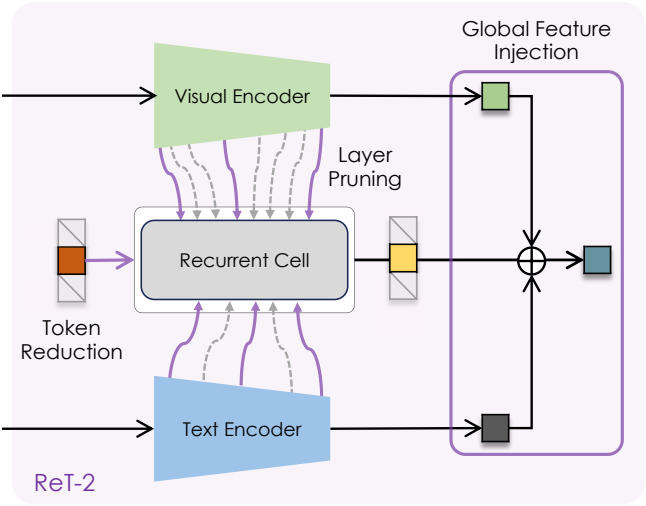


Figure 4.7: Visualization of the differences between the previous method (*i.e.*, ReT) and the newly proposed ReT-2. The new method introduces three key modifications: (i) **token reduction**, instead of multiple input tokens, only a single token is used; (ii) **layer pruning**, rather than using all textual and visual layers, we now select only three representative layers (early, middle, and late), independent of the architecture; and (iii) **global feature injection**, a newly added module that integrates global information to enhance the representation. Highlighted regions indicate the most significant differences.

- **Token Reduction**: The number of input tokens is reduced from 32 to a single token per modality. This choice addresses the issue of rank collapse observed in the output embeddings and encourages the model to produce more compact and meaningful representations.
- **Simplified Contrastive Objective**: The use of a single token per side eliminates the need for the fine-grained contrastive loss used in ReT. Instead, we apply a standard InfoNCE loss directly over the single fused token from both the query and document, significantly simplifying the retrieval pipeline and improving inference efficiency.
- **Layer Pruning**: Rather than relying on all layers of the textual and visual backbones or attempting to explicitly align architectures with different depths, we always sample three layers: one from the lower

(early), one from the middle, and one from the upper (final) part of each backbone. This strategy ensures compatibility and stability, especially when backbones differ in depth.

- **Global Feature Injection**: To enhance contextual understanding, ReT-2 integrates global feature representations alongside layer-specific features. This injection of global context helps the model capture general information, further helping retrieval accuracy and robustness.

4.3 Comparative Evaluation of Multimodal Retrieval Models

4.3.1 Evaluation Benchmarks and Metrics

We evaluate our models on the M2KR [160] and M-BEIR [280] benchmarks, which provide a diverse, large-scale collection of datasets for comprehensive assessment of multimodal retrieval performance across various domains and task configurations.

M2KR Benchmark. M2KR integrates heterogeneous sources, including WIT [250], IGLUE [24], KVQA [239], CC3M [241], MSMARCO [199], OVEN [104], LLaVA [169], InfoSeek [48], Encyclopedic-VQA [191], and OKVQA [187]. These datasets span a range of domains, enabling robust evaluation of retrieval models under varying degrees of complexity and multimodal reasoning. To better align with our setting, where both queries and documents are multimodal, we augment the M2KR splits of OVEN, InfoSeek, Encyclopedic-VQA, and OKVQA by enriching the reference documents with associated images [30], enabling an effectively evaluation of models that rely on both textual and visual signals during retrieval. In our experiments, we employ training, validation, and test splits used in previous works [160, 30].

M-BEIR Benchmark. M-BEIR comprises eight retrieval tasks and ten different datasets, with around 1.5M human-authored queries and a pool of 5.6M candidate documents. The benchmark spans diverse sources, including everyday images, fashion products, Wikipedia entries, and news articles. In addition to standard multimodal settings, it includes tasks with missing modalities on either the query or document side, enabling evaluation under incomplete conditions. To ensure consistency between training and testing, M-BEIR adapts datasets originally designed for different tasks, including OVEN [104], EDIS [170], CIRR [175], FashionIQ [283], COCO [158], Fashion200k [95], Visual News [165], and NIGHTS [76]. Moreover, M-BEIR defines a *global* retrieval scenario, where candidates are retrieved from the full 5.6M pool encompassing all tasks and datasets, and a *local* one, which restricts candidates to the task-specific pool provided by each dataset. We report results on the M-BEIR$_{\text{local}}$ setting, for fair comparison with existing state-of-the-art retrieval models.

Evaluation Metrics. Following the evaluation protocol of M2KR, we assess model performance using recall at K (*i.e.*, the percentage of queries for which the target document falls within the top-K most similar documents). The value of K is determined based on the experimental setup of each sub-dataset. For VQA splits, we also report the pseudo recall metric, as proposed in [160], which considers a retrieved document relevant whenever it contains the answer. For M-BEIR, we adhere to the original evaluation protocol and report standard recall at K values accordingly (using $K = 5$ for most datasets, and $K = 10$ for Fashion200k and FashionIQ).

4.3.2 Implementation Details

In our experiments, we evaluate multiple configurations of both visual and textual backbones. For the visual encoder, we consider CLIP ViT-B/32, CLIP ViT-L/14 [215], SigLIP2 ViT-L/14 [269], and OpenCLIP ViT-H/14 [53]. For the tex-

Table 4.2: Selected layers for each backbone in ReT-2. L denotes the depth of each backbone, measured in number of layers.

	Text Encoder		Visual Encoder	
Backbone	L	**Layer Indices**	L	**Layer Indices**
CLIP ViT-B	12	3, 7, 11	12	3, 7, 11
ColBERTv2	12	3, 7, 11	-	-
CLIP ViT-L	12	3, 7, 11	24	3, 18, 23
SigLIP2 ViT-L	24	3, 18, 23	24	3, 18, 23
OpenCLIP ViT-H	24	3, 18, 23	32	4, 25, 31

tual encoder, we use the corresponding CLIP/SigLIP variants as well as ColBERTv2 [228]. We retain only three representative layers from each backbone, corresponding to early, intermediate, and late stages. The specific layers selected for each configuration are detailed in Table 4.2.

We trained in mixed precision with the Adam optimizer [125] on 4 NVIDIA A100 64GB GPUs for up to 24 hours. When adding global features, we always unfreeze the pooling layer of the backbones, if present. This corresponds to the visual and textual linear projections for CLIP-based and ColBERTv2 models, and to the attention pooling layers for SigLIP2. Following ReT, the recurrent cell operates with a hidden size d equal to 1,024 and with the biases b_i and b_f equal to zero. The dimension of $\mathbf{W}_{final}$ (cf. Eq. 4.9) is set to match d with the dimension of the global features. When unfreezing the unimodal backbones, we activate gradient checkpointing, and we downscale their learning rate by 0.05 compared to the recurrent cell. At test time, we index passages using the Faiss library [115] for fast retrieval.

For M2KR, we use the same training recipe as ReT, setting the learning rate to 5×10^{-5} with a cosine scheduler and a batch size of 512, training for 75k steps. We observe that training further leads to overfitting on some benchmarks, particularly severe on InfoSeek. For M-BEIR, we train for 20 epochs with a batch size of 768, using the data sampling strategy proposed in [109]. The learning rate is linearly ramped up to 1×10^{-4} within the first 300 steps, and then decays accordingly to a cosine schedule.

4.3.3 Ablation Studies and Analyses

The original ReT model employs 32 input tokens and a dedicated recurrent cell on both the query and document sides. During training, the output tokens are used to compute a fine-grained late-interaction relevance score, following [122, 228]. Table 4.3 presents ablation studies supporting the architectural modifications introduced in ReT-2. Results are reported on the M2KR benchmark, using CLIP ViT-L as visual and textual backbones.

Score Fusion. We assess the impact of replacing the fine-grained late-interaction relevance score computation with a score fusion strategy. In practice, rather than computing 32×32 dot products for each query-document pair, we sum the rows of the output matrix of ReT before the late-interaction projection to dimension 128, obtaining a single embedding token, typically of a size varying from 768 to 1,024, to compute the query-document similarity via dot product. Note that this is equivalent to substituting the $\max$ operator in Eq. 4.12 with a new summation over j. However, thanks to the distributive property of the dot product, we do not need to compute the 32×32 similarity matrix explicitly. This shift enables faster and more memory-efficient training, as well as quicker inference retrieval, with minimal change in performance, as the average retrieval score moves from 61.5 to 61.4 – *i.e.*, *score fusion (32 tokens)*.

Sharing Weights. Building on the score fusion model, we experiment with sharing the weights between the query and document encoders, essentially setting $\text{ReT}_{\text{Q}} = \text{ReT}_{\text{D}}$. Apart from saving memory during training, switching to a shared architecture – *i.e.*, *shared architecture (32 tokens)* – raises the average score to 62.6, with an improvement of +1.2 points compared to having separate encoders. As most substantial gains come from InfoSeek and Encyclopedic-VQA, which present tens to hundreds of questions for the same Wikipedia entity, we credit the shared architecture approach for reducing overfitting on entities seen during training.

Table 4.3: Ablation study results on the M2KR benchmark. All experiments are with CLIP ViT-L for both visual and textual encoders.

	WIT	IGLUE	KVQA	OVEN	LLaVA	InfoSeek		E-VQA		OKVQA		
Model	R@10	R@1	R@5	R@5	R@1	R@5	PR@5	R@5	PR@5	R@5	PR@5	Avg
ReT [30]	73.4	81.8	63.5	82.0	79.9	47.0	60.5	44.5	57.9	20.2	66.2	61.5
+ global features	79.3	81.6	65.8	82.8	82.1	46.6	60.8	43.3	58.0	17.4	64.8	62.0
+ score fusion (32 tokens)	79.5	81.9	66.7	83.4	81.0	42.7	57.5	43.4	57.1	17.5	65.1	61.4
+ shared architecture (32 tokens)	78.0	83.4	66.7	83.5	84.2	48.0	59.9	48.0	61.1	13.3	62.8	62.6
+ shared architecture (16 tokens)	78.4	82.2	66.1	83.6	83.2	47.5	60.1	48.4	61.7	13.1	63.6	62.5
+ shared architecture (8 tokens)	78.1	82.6	62.4	83.5	82.7	47.2	60.9	48.4	61.1	14.2	65.6	62.4
+ shared architecture (4 tokens)	78.7	82.2	63.5	82.9	82.3	48.5	61.3	47.5	60.8	13.5	63.1	62.2
+ shared architecture (single token)	78.3	81.9	66.5	84.1	84.1	48.2	60.5	48.7	60.9	12.9	65.0	62.8
+ non-shared architecture (single token)	79.8	82.5	67.8	84.4	81.4	46.4	59.0	42.6	56.7	17.1	65.6	62.1
+ layer pruning	77.9	82.2	63.3	84.3	82.9	50.1	62.2	47.7	60.7	14.6	66.0	62.9
+ global features (**ReT-2**)	81.1	82.9	72.3	83.1	83.8	48.0	61.0	49.7	62.6	15.2	65.9	64.1

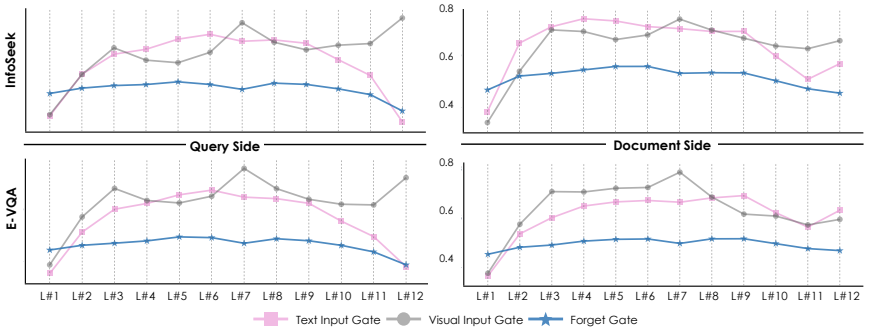


Figure 4.8: Analysis of average gate activation over 2k examples from the InfoSeek and Encyclopedic-VQA test split of the M2KR benchmark.

Token Reduction. The next change arises from an analysis of the output matrix of ReT, revealing that it suffers from rank collapse. Empirically, we register the rank collapse score [116] of the 32-row matrix generated by ReT when embedding samples from the InfoSeek test split of M2KR. The last recurrent step of ReT outputs $32 \times 1,024$ matrices. For them, we register an average rank collapse score of 0.18 when embedding queries and 0.22 when embedding documents. After applying the late-interaction linear projection to 32×128 dimensions, the average rank collapse scores further plummets to 0.09 and 0.11. Ideally, those scores would tend to 1.0, and our analysis indicates that the 32 token embeddings of the output matrix converge to a unified representation. Consequently, the purpose of using multiple token

embeddings to represent inputs is questionable. This motivates the exploration of a token reduction strategy, by applying score fusion to a number of tokens equal to 16, 8, 4, and 1 (*i.e.*, no score fusion at all). While reducing the number of tokens initially seems to degrade performance, we register an average improvement of +0.2 points when switching from 32 tokens to a single one – *i.e.*, *shared architecture (single token)*. Notably, this happens along with a reduction in trainable parameters and less computation, as with a single token, there is no need to apply self-attention in the recurrent cell of ReT. For completeness, we also include the single token version of ReT without sharing weights between the query and document encoders – *i.e.*, *non-shared architecture (single token)*.

Layer Pruning. Driven by the computational constraints of ReT, primarily due to the recurrent cell being applied to a predefined number of layers ranging from 12 to 16, we explore a layer pruning strategy to improve efficiency. In detail, we sample a total of three layers, corresponding to the early, middle, and late stages of both the visual and textual backbone*. This strategy preserve information from different abstraction levels, and it has been proven effective for the visual-language alignment of MLLMs [155]. Our choice is further supported by an empirical analysis of the average gate activations of ReT, conducted on the InfoSeek and Encyclopedic-VQA test splits of M2KR. As shown in Fig. 4.8, the visual input gate exhibits three prominent activation peaks, aligning with the selected layer groups. On the other hand, the textual input gate has a smoother behavior, peaking mainly across early-to-middle stages, thus highlighting the importance of including low-level textual features. Quantitative results validate the effectiveness of this pruning strategy: not only it preserves retrieval performance, but it also yields a +0.1 points improvement in accuracy.

*Because in Table 4.3 ReT-2 is paired with CLIP ViT-L/14, it follows that we employ the third, eighteenth, and second last layer from the visual backbone, and the third, seventh, and second last layer from the textual backbone. We refer to Table 4.2 for the layer selection in backbones with different depths.

Global Feature Injection. Finally, we incorporate global features, obtaining our final ReT-2 model. In detail, we apply score fusion by summing the multimodal, multilayer token coming from the recurrent cell with the pooler token of the visual backbone and the one from the textual backbone. This raises the average score to 64.1, with a $+2.6$ points improvement over ReT. For fairness of comparison, we apply global feature injection to ReT as well (*i.e.*, gray row). In this setting, the pooler tokens are first projected to dimension 128 and then concatenated to the 32 tokens of the recurrent cell. We highlight that, even in this scenario, global features raise the performance of ReT, while still falling behind ReT-2.

In summary, our final model, ReT-2, fuses multimodal and multi-layer features into a single learnable token, shares parameters between the query and document encoders, and incorporates global features. This design achieves superior performance without relying on the computationally expensive fine-grained contrastive loss, and is adopted as the final model for all subsequent experiments.

4.3.4 Comparison with Existing Approaches

Baselines and Competitors. To ensure a comprehensive comparison, we introduce three baselines within our experimental setup.

1. **CLIP (Feature Averaging).** This approach leverages the zero-shot capabilities of CLIP, which we adapt for the multimodal query/document setting by averaging the unimodal feature outputs from each encoder. On the query side, the query text and image are processed separately by two distinct, frozen CLIP encoders, and the resulting pooled feature vectors are then combined by their average. This same approach is applied on the document side.
2. **CLIP (Unimodal)** This is a straightforward extension of CLIP to the fine-

grained late-interaction paradigm. We use all the tokens from the last layer of the CLIP visual encoder to represent the query, whereas the document is encoded by taking the output of the last layer of the CLIP textual encoder. Because CLIP has not been trained using a fine-grained relevance score (cfr. Eq. 4.12), we apply LoRA [103] to the textual and visual encoders and train them along with the late-interaction linear projection $\mathbf{W}_{final}$.

3. **CLIP (Feature Fusion)** Inspired by the feature fusion approach proposed in [280], we build a multimodal retriever by training a cross-attention layer with residual connection on top of two frozen CLIP encoders. The attention queries in the cross-attention layers are the visual tokens on the query side and the textual tokens on the document side. For unimodal textual documents, we feed a black image to the CLIP visual encoder and mask out the cross-attention output before adding it to the residual connection.

We also introduce some competitors:

1. **FLMR [159] and PreFLMR [160]** These are multimodal retrievers pre-trained on a vision-language corpus of over ten million items where the query is multimodal, while the document side contains only textual passages. While FLMR only uses the CLS embedding from the visual backbone as the image representation, PreFLMR enhances this by also extracting embeddings of image patches from the penultimate layer to provide a more detailed visual representation. Notably, PreFLMR adopts a three-stage training strategy, with the second one dedicated exclusively to Encyclopedic-VQA to better handle its higher diversity, difficulty, and dataset size compared to other KB-VQA datasets.

2. **UniIR [280]** They propose strategies for encoding multimodal queries and documents, by leveraging pre-trained models like CLIP [215] and

BLIP [148] to integrate different modalities, with the goal of creating a unified retriever for diverse tasks. Here we consider the feature-level fusion version, that utilizes only the features from the last layer while fine-tuning both visual and textual backbones.

Results on the M2KR Benchmark. Table 4.4 presents a comparison of our proposed method, ReT-2, against a zero-shot CLIP baseline and other retrieval approaches. These include FLMR [159] and PreFLMR [160], two multimodal retrieval models trained on M2KR. Both models adopt a multimodal query and a text-only document setting. FLMR relies on the `CLS` token for image representation, whereas PreFLMR enriches visual information using patch embeddings from the penultimate layer, capturing more fine-grained features. For reference, we also report results from our earlier model, ReT [30]. We also include a variant of ReT-2 in which the visual and textual backbones are unfrozen during training (🔥).

Across all datasets and backbones, ReT-2 consistently outperforms the original ReT. For example, on WIT with a frozen CLIP ViT-L backbone, ReT-2 achieves a substantial gain of +7.7 points over ReT (81.1 vs. 73.4). When compared to other state-of-the-art methods, ReT-2 achieves the best average performance in most settings, with the only exceptions being Encyclopedic-VQA and OKVQA, where PreFLMR slightly outperforms it. In this regard, we notice that PreFLMR employs a three-stage training pipeline, with the second stage being dedicated to Encyclopedic-VQA and the third stage entailing a careful balancing and resampling of each sub-dataset. In contrast, our ReT-2 models are trained in a single-stage run on the entire M2KR dataset. The trainable variant of ReT-2 (🔥) further boosts performance – for instance, with the SigLIP2 backbone, the trainable version delivers an average improvement of +7.2 points. Similar trends are observed across all backbone architectures: CLIP ViT-B shows improvement from 55.2 to 59.1, CLIP ViT-L from 64.1 to 67.9, and OpenCLIP ViT-H from 63.4 to 69.2. Finally, scaling

Table 4.4: Experimental results on the M2KR benchmark [160], comparing ReT-2 to baselines and competitors when varying the visual backbone. Bold font denotes the best results under the same backbone. The † marker denotes our reproductions.

		WIT	IGLUE	KVQA	OVEN	LLaVA	InfoSeek		E-VQA		OKVQA		
Model	**Backbone**	R@10	R@1	R@5	R@5	R@1	R@5	PR@5	R@5	PR@5	R@5	PR@5	**Avg**
CLIP (ZS)	CLIP ViT-B	48.9	63.1	**57.8**	58.1	33.0	33.6	47.4	0.13	12.1	0.52	49.9	36.8
CLIP (Unimodal)	CLIP ViT-B	47.6	59.1	**33.7**	54.2	28.4	15.8	35.4	9.7	23.0	2.5	39.9	31.8
CLIP (Feature Fusion)	CLIP ViT-B	41.6	56.6	22.0	59.8	58.0	19.3	40.4	21.2	40.5	9.6	56.0	38.6
FLMR [159]	CLIP ViT-B	23.8	-	31.9	40.5	56.4	-	47.1	-	-	-	68.1	-
PreFLMR [160]	CLIP ViT-B	41.7	57.3	28.6	46.3	67.2	26.0	48.8	**55.0**	**67.9**	**27.2**	**66.1**	48.4
ReT [30]	CLIP ViT-B	60.1	73.9	26.9	72.9	76.6	30.2	48.1	33.0	48.9	13.9	58.3	49.3
ReT-2 (Ours)	CLIP ViT-B	**68.3**	**76.1**	56.6	**73.8**	**81.2**	**36.9**	**52.7**	36.1	52.9	12.0	60.7	**55.2**
ReT-2 (Ours)	CLIP ViT-B 🔥	73.7	77.7	66.6	77.3	86.0	38.3	53.8	42.0	57.6	14.9	62.6	59.1
CLIP (ZS)	CLIP ViT-L	65.9	74.9	73.3	68.5	36.6	48.0	58.4	0.17	12.0	0.59	49.2	45.0
PreFLMR [160]	CLIP ViT-L	60.5	69.2	43.6	59.8	71.8	37.4	57.9	**60.9**	**70.8**	**31.4**	**68.5**	57.4
ReT [30]	CLIP ViT-L	73.4	81.8	63.5	82.0	79.9	47.0	60.5	44.5	57.9	20.2	66.2	61.5
ReT-2 (Ours)	CLIP ViT-L	**81.1**	**82.9**	**72.3**	**83.1**	**83.8**	**48.0**	**61.0**	49.7	62.6	15.2	65.9	**64.1**
ReT-2 (Ours)	CLIP ViT-L 🔥	86.1	84.4	78.1	86.8	88.6	49.1	62.3	56.4	67.4	20.0	67.8	67.9
SigLIP2 (ZS)	SigLIP2 ViT-L	51.9	60.0	**48.4**	74.3	41.1	51.4	60.4	19.5	33.2	6.1	50.1	45.1
PreFLMR [160]†	SigLIP2 ViT-L	68.3	**76.1**	39.1	71.5	73.5	42.9	59.5	**51.6**	**64.1**	**17.8**	**70.6**	57.7
ReT [30]	SigLIP2 ViT-L	65.7	71.8	34.8	81.1	75.1	42.2	56.4	35.2	51.2	15.4	63.3	53.8
ReT-2 (Ours)	SigLIP2 ViT-L	**70.3**	71.2	48.2	**85.3**	**81.8**	**57.1**	**65.5**	44.5	58.1	10.8	61.5	**59.5**
ReT-2 (Ours)	SigLIP2 ViT-L 🔥	80.6	79.4	61.8	88.8	89.4	59.7	67.7	51.6	63.5	21.2	70.5	66.7
OpenCLIP (ZS)	OpenCLIP ViT-H	74.2	78.2	**68.0**	78.4	45.3	**53.2**	61.3	20.8	33.3	7.3	63.9	53.1
PreFLMR [160]	OpenCLIP ViT-H	60.5	71.2	39.4	61.5	72.3	39.2	59.5	**62.5**	**71.7**	**30.2**	**68.1**	57.8
ReT [30]	OpenCLIP ViT-H	71.4	80.0	59.3	83.0	79.8	47.3	60.7	44.8	57.8	18.2	63.4	60.5
ReT-2 (Ours)	OpenCLIP ViT-H	**80.2**	**82.3**	66.2	**83.3**	**86.1**	52.8	**63.1**	45.9	59.3	14.4	64.0	**63.4**
ReT-2 (Ours)	OpenCLIP ViT-H 🔥	85.5	84.2	75.8	88.4	91.1	58.0	66.7	58.9	69.3	18.3	65.1	69.2

the unfrozen visual backbone also correlates with stronger retrieval results: average performance increases from 59.1 with CLIP ViT-B, to 67.9 with CLIP ViT-L, and 69.2 with OpenCLIP ViT-H. In contrast, when the backbones are frozen, we observe a similar trend to that reported in both ReT and PreFLMR: the larger OpenCLIP ViT-H underperforms relative to the smaller CLIP ViT-L, suggesting that the benefits of scaling depend on the dataset and experimental setting. To provide a fairer comparison with the original PreFLMR model, which uses ColBERTv2 [228] as its textual backbone, in Table 4.5 we report the results obtained when replacing the textual backbone in both ReT and ReT-2 with ColBERTv2. This evaluation is conducted using both CLIP and SigLIP2 ViT-L visual backbones, ensuring consistency and comparability across architectures. The performance trends remain consistent: ReT-2 outperforms both the original ReT and PreFLMR, even when matched on backbone architecture. The largest improvements are again observed with the trainable variant of ReT-2, yielding average gains of +6.4 and +12.6 over ReT when using CLIP and SigLIP2 ViT-L, respectively.

Table 4.5: Experimental results on the M2KR benchmark [160], comparing ReT-2 to baselines and competitors when employing ColBERTv2 [228] as textual backbone. † indicates our reproductions.

		WIT	IGLUE	KVQA	OVEN	LLaVA	InfoSeek		E-VQA		OKVQA		
Model	Backbone	R@10	R@1	R@5	R@5	R@1	R@5	PR@5	R@5	PR@5	R@5	PR@5	Avg
PreFLMR [160]	CLIP ViT-L	60.5	69.2	43.6	59.8	71.8	37.4	57.9	**60.9**	**70.8**	**31.4**	**68.5**	57.4
ReT [30]	CLIP ViT-L	73.9	79.3	48.6	79.6	79.6	40.0	58.9	43.4	59.0	19.0	64.1	58.7
ReT-2 (Ours)	CLIP ViT-L	**78.6**	**80.3**	**48.8**	**81.2**	**80.3**	**50.9**	**64.9**	47.1	62.1	14.8	62.1	**61.0**
ReT-2 (Ours)	CLIP ViT-L 🔥	81.9	81.0	62.9	83.7	84.8	**52.1**	**66.2**	55.1	67.8	16.7	64.2	**65.1**
PreFLMR [160]†	SigLIP2 ViT-L	68.3	76.1	39.1	71.5	73.5	42.9	59.5	**51.6**	**64.1**	**17.8**	**70.6**	57.7
ReT [30]	SigLIP2 ViT-L	65.7	71.8	34.8	81.8	75.1	42.2	56.4	35.2	51.2	15.4	63.3	53.9
ReT-2 (Ours)	SigLIP2 ViT-L	**78.9**	**79.1**	**48.6**	**84.4**	**83.0**	**53.7**	**66.3**	49.1	63.2	15.2	61.9	**62.1**
ReT-2 (Ours)	SigLIP2 ViT-L 🔥	82.7	82.5	56.4	86.6	86.1	59.4	68.6	**56.5**	**68.6**	17.3	67.4	**66.5**

Results on M-BEIR Benchmark. In Table 4.6, we further evaluate the generalization capability of our proposed approach on M-BEIR$_{local}$. The benchmark comprises eight distinct tasks, each presenting different modality configurations and challenges. In this setting, we compare ReT-2 with zero-shot baselines and competitors like UniIR [280], GENIUS [124], and the previous version of our model (*i.e.*, ReT). Specifically, UniIR proposes strategies for encoding multimodal queries and documents, by leveraging pre-trained models like CLIP and BLIP [148] to integrate different modalities. In this table, we also include our reproduction of UniIR using the SigLIP2 backbone to ensure a fair and consistent comparison. GENIUS, on the other hand, is a versatile generative retrieval framework that discretizes multimodal inputs. As additional competitors, we include retrieval models based on MLLMs, such as MM-Embed [157], JFE [109], PUMA [186], and LamRA [174]. Due to their significantly larger model sizes and parameter counts, these methods are not directly comparable to ours.

The results show that ReT-2, using both the CLIP and SigLIP2 ViT-L backbones, significantly outperforms not only the original ReT version but also all other competitors. For instance, the SigLIP2 variant of ReT-2 achieves a notable improvement of +5.2 points over UniIR using the same backbone. Remarkably, despite being smaller in size and not relying on an LLM, the variant of ReT-2 based on SigLIP2 delivers the best overall performance

Table 4.6: Experimental results on the M-BEIR$_{local}$ benchmark [280]. † indicates our reproductions, and gray denotes MLLM-based methods.

		#1			#2	#3		#4			#5	#6		#7		#8		
	Backbone	VN	COCO	F200k	WQA	EDIS	WQA	VN	COCO	F200k	NIGHTS	OVEN	InfoSeek	FIQ	CIRR	OVEN	InfoSeek	Avg
CLIP (ZS)	CLIP ViT-L	43.4	61.1	6.6	36.2	43.3	45.1	41.3	79.0	7.7	26.1	24.2	20.5	7.0	13.2	38.8	26.4	32.5
SigLIP2 (ZS)	SigLIP2 ViT-L	40.0	77.5	34.8	33.7	27.3	42.5	40.4	88.1	35.3	28.4	30.0	30.2	20.4	29.3	41.9	34.3	39.6
PreFLMR [160]	CLIP ViT-L	–	–	–	68.1	21.8	37.6	0.1	8.1	0.0	–	19.9	21.7	–	–	27.4	23.5	–
ReT [30]	CLIP ViT-L	23.2	66.3	12.3	47.0	47.1	56.9	23.0	85.5	9.5	21.5	39.0	21.4	10.6	27.1	57.3	33.9	36.3
ReT [30]	CLIP ViT-L 🔥	24.2	72.8	14.5	54.3	48.5	65.6	24.1	87.6	15.7	25.6	37.5	20.2	13.0	37.2	56.3	35.2	39.5
GENIUS [124]	CLIP ViT-L 🔥	27.4	78.0	16.2	44.6	44.3	60.6	28.4	91.1	16.3	30.2	41.9	20.7	19.3	39.5	52.5	30.1	40.1
UniIR [280]	BLIP ViT-L 🔥	23.4	79.7	26.1	80.0	50.9	79.8	22.8	89.9	28.9	**33.0**	41.0	22.4	29.2	52.2	55.8	33.0	46.8
UniIR [280]	CLIP ViT-L 🔥	42.6	81.1	18.0	84.7	59.4	78.7	43.1	92.3	18.3	32.0	45.5	27.9	24.4	44.6	67.6	48.9	50.6
UniIR [280]†	SigLIP2 ViT-L 🔥	29.4	78.1	21.6	75.3	49.9	77.6	33.0	91.1	44.5	29.5	52.9	27.9	33.1	54.0	71.2	**50.7**	51.2
ReT-2 (Ours)	CLIP ViT-L 🔥	**47.3**	80.2	21.1	**86.0**	**56.7**	**80.2**	**46.8**	91.6	22.7	31.5	48.7	27.5	23.8	44.3	69.1	47.0	51.5
ReT-2 (Ours)	SigLIP2 ViT-L 🔥	38.9	**84.8**	**50.0**	76.3	53.7	78.4	42.0	**95.0**	**52.2**	31.5	**54.1**	**32.3**	**35.3**	**57.1**	**72.1**	48.3	**56.4**
MM-Embed [157]	LLaVA-NeXT-7B	41.0	71.3	17.1	95.9	68.8	85.0	41.3	90.1	18.4	32.4	42.1	42.3	25.7	50.0	64.1	57.7	52.7
JFE [109]	PaliGemma-3B	34.6	78.5	37.2	88.7	54.3	82.4	33.1	90.0	36.9	27.8	46.0	35.6	31.8	54.0	72.7	61.1	54.0
PUMA [186]	Qwen2-VL-7B	35.7	79.5	25.8	86.2	35.2	90.1	29.0	31.4	58.2	78.4	52.7	48.3	30.6	49.9	74.0	65.2	54.4
LamRA [174]	Qwen2-VL-7B	41.6	81.5	28.7	86.0	62.6	81.2	39.6	90.6	30.4	32.1	54.1	52.1	33.2	53.1	76.2	63.3	56.6

compared to nearly all MLLM-based competitors, falling just short of the LamRA model, which achieves only a +0.2-points average improvement.

Computational Analysis. In Table 4.7, we provide a computational analysis of ReT-2 and competitors in terms of resource demand for training and inference speed. The analysis employs a subset of the InfoSeek dataset comprising 100k image-text passages and 4.7k image-text queries. For CLIP ViT-L and SigLIP2, which we include as baselines for image-text retrieval, we mask out text on the query side and images on the document side. For ReT and PreFLMR, we follow the implementation in [122] to index passages, enabling efficient fine-grained late-interaction retrieval through GPU acceleration. This implementation runs the forward pass of the models in full precision, so we stick with full precision to measure the forward time of all the models. An exception is LamRA, which we run in half precision to account for the additional memory requirements due to its 7B MLLM backbone. For the other methods, we build a `GpuIndexFlat` using the Faiss library. All experiments are run on a single NVIDIA A100 GPU (64GB of VRAM).

Notably, ReT-2 benefits from the introduced layer pruning strategy and the use of a single input token to embed queries and documents, resulting in significantly faster forward and retrieval times compared to ReT and PreFLMR, which rely on the more computationally intensive fine-grained late-

Table 4.7: Comparison of training resources and inference times between ReT-2 and competing methods.

	Training Info			Inference Time (ms)			
Model	Backbones	#GPUs	Hrs	Forward	Retrieval	All ↓	#Tokens
CLIP (ZS)	T ❄ V ❄	-	-	18.6	0.7	19.3	1
SigLIP2 (ZS)	T ❄ V ❄	-	-	19.2	0.8	20.0	1
PreFLMR [160]	T 🔥 V ❄	4	864	32.7	406.1	438.8	320
UniIR [280]	T 🔥 V 🔥	8	72	23.8	0.8	33.2	1
LamRA [174]	MLLM 🔥	16	N/A	52.7	1.5	54.2	1
ReT [29]	T ❄ V ❄	4	80	31.4	3.5	34.9	32
ReT-2 (Ours)	T ❄ V ❄	4	80	26.8	0.8	27.6	1
ReT-2 (Ours)	T 🔥 V 🔥	4	160	26.8	0.8	27.6	1

interaction paradigm. Compared with UniIR, ReT-2 demonstrates competitive retrieval speed while generally requiring equal or lower training resources, depending on whether the unimodal backbones are trained together with the recurrent retrieval cell or kept frozen. It is worth noting that LamRA takes nearly twice the forward and retrieval time of ReT-2, not to mention the additional storage required for saving embeddings of size 3,584 rather than 768 as in our model. Ultimately, the decision to rely on MLLMs rather than smaller encoders based on vision-language foundation models is a trade-off between performance and efficiency.

5

Retrieval-Augmented Image Captioning Models

Retrieval-augmented image captioning leverages this idea by integrating retrieval components into the captioning pipeline, allowing the language model to condition its generation not only on the visual input but also on relevant information retrieved from a large multimodal corpus. By explicitly exploiting an external memory, such models can ground the generation process in additional visual and textual evi-

This chapter discusses topics from the following papers: S. Sarto *et al.*, "Retrieval-Augmented Transformer for Image Captioning", CBMI 2022 [230] and S. Sarto *et al.*, "Towards Retrieval-Augmented Architectures for Image Captioning", ACM TOMM [232] and M. Barraco, S. Sarto *et al.*, "With a little help from your own past: Prototypical memory networks for image captioning", ICCV 2023 [17].

dence, producing captions that are more detailed, accurate, and semantically rich. Moreover, retrieval augmentation improves generalization to novel concepts and long-tail distributions, alleviating the need to rely solely on increased model capacity or extensive retraining.

In the following sections, we first contextualize retrieval-augmented captioning by reviewing how image captioning architectures evolved over time. We then discuss how retrieval augmentation has been incorporated into these models to enhance caption generation. Finally, we broaden the perspective by surveying memory-augmented captioning models, with a focus on learnable memories that store prototypical representations.

5.1 Image Captioning: Models and Trends

State-of-the-art image captioning models have evolved through a sequence of architectural refinements aimed at improving visual grounding, linguistic fluency, and alignment between visual and textual representations. Most of the time, the performance on image captioning are evaluated on the COCO dataset and reporting the CIDEr metric.

Early competitive approaches are primarily based on LSTM language models coupled with visual feature extractors. Among these, the Up-Down model [7] introduced bottom-up and top-down attention mechanisms to dynamically select relevant image regions during generation. Subsequent works extended this paradigm by incorporating structured visual information, such as spatial relationships and scene graphs, as in GCN-LSTM [299] and SGAE [295], or by enhancing attention through self-attentive formulations, including AoANet [108], X-LAN [206], and DPA [164]. These models demonstrated that explicitly modeling relationships between visual elements and refining attention can substantially improve caption quality.

More recent methods have shifted toward fully Transformer-based ar-

Table 5.1: Performance comparison of representative image captioning approaches using standard evaluation metrics. For all metrics, higher values indicate better performance (↑). Table reproduced from [251]

Model	B-1	B-4	M	R	C	S
Show and Tell† [272]	72.4	31.4	25.0	53.1	97.2	18.1
SCST (FC)‡ [222]	74.7	31.7	25.2	54.0	104.5	18.4
Show, Attend and Tell† [289]	74.1	33.4	26.2	54.6	104.6	19.3
SCST (Att2in)‡ [222]	78.0	35.3	27.1	56.7	117.4	20.5
Up-Down‡ [7]	79.4	36.7	27.9	57.6	122.7	21.5
SGAE [295]	81.0	39.0	28.4	58.9	129.1	22.2
MT [247]	80.8	38.9	28.8	58.7	129.6	22.3
AoANet [108]	80.2	38.9	29.2	58.8	129.8	22.4
X-LAN [206]	80.8	39.5	29.5	59.2	132.0	23.4
DPA [164]	80.3	40.5	29.6	59.2	133.4	23.3
AutoCaption [327]	81.5	40.2	29.9	59.5	135.8	23.8
ORT [97]	80.5	38.6	28.7	58.4	128.3	22.6
CPTR [172]	81.7	40.0	29.1	59.4	129.4	–
$\mathcal{M}^2$ Transformer [64]	80.8	39.1	29.2	58.6	131.2	22.6
X-Transformer [206]	80.9	39.7	29.5	59.1	132.8	23.4
Unified VLP [325]	80.9	39.5	29.3	59.6	129.3	23.2
VinVL [316]	**82.0**	**41.0**	**31.1**	**60.9**	**140.9**	**25.2**

chitectures, motivated by their success in sequence modeling and their ability to capture long-range dependencies. Models such as ORT [97], the $\mathcal{M}^2$ Transformer [64], X-Transformer [206], and RSTNet [320] replace recurrent language models with self-attentive decoders, often employing cross-attention to integrate visual features more effectively. Building on this trend, some approaches further enhance representation capacity by combining visual features extracted from multiple backbones, as exemplified by DLCT [185] and DIFNet [284].

A comparative analysis of popular image captioning approaches, evaluated using standard captioning metrics, is presented in Table 5.1. This table has been taken from [251]. These models highlight the benefits of richer visual representations and fully attentive architectures for improving both descriptive accuracy and linguistic coherence.

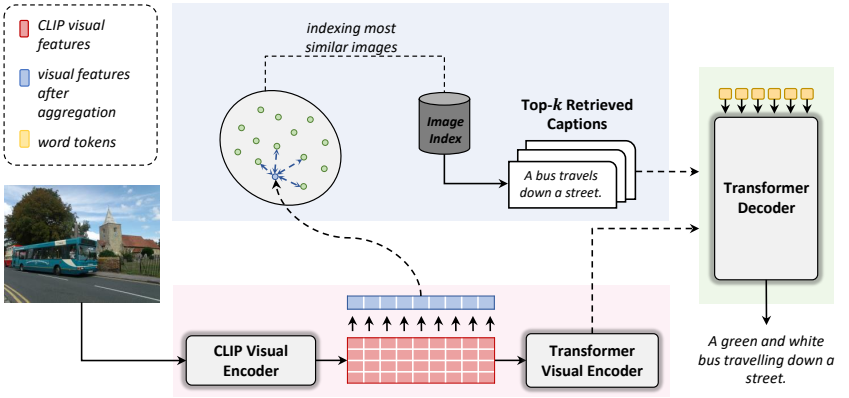


Figure 5.1: Schema of a general retrieval-augmented architecture for image captioning. Given an input image, visual features are extracted using a CLIP-based image encoder. These features are then used to retrieve a set of similar textual sentences, starting from the corresponding image representations, that are employed as additional knowledge during the generation of the caption.

5.2 A Retrieval-Augmented Transformer for Image Captioning

The goal of an image captioner is that of modeling a distribution $p(y|I)$ over possible captions y given an input image I. With the aim of separating the language modeling and memorization capabilities of the captioner, we augment the model with an external memory of textual descriptions (Fig. 5.1), which will serve as the memory of the model. Under this setting, we can decompose the probability distribution $p(y|I)$ into two steps: ① *retrieval* of relevant textual items from the external memory and ② *prediction* of the textual description (or language modeling), conditioned on retrieved items. Firstly, given an image I we retrieve a set of descriptions $\{z_i\}_i$ from the external memory, performing kNN searches in a visual similarity space. Then, we condition our language model on both the image I and the retrieved descriptions $\{z_i\}_i$. From a probabilistic point of view, this amounts to modeling $p(y|\{z_i\}_i, I)$ and marginalizing over the set of retrieved captions.

5.2.1 External Memory and Knowledge Retrieval

The retrieval of relevant textual items from the external memory aims at modeling $p(z|I)$ given a corpus of image-text pairs and an input query image I. The knowledge retrieval component performs an approximate k-nearest-neighbor search into the external memory, defined through an inner product similarity between image embeddings, *i.e.*:

$$f(I_1, I_2) = \text{Embed}(I_1)^\intercal \, \text{Embed}(I_2), \tag{5.1}$$

where $\text{Embed}(\cdot)$ is a function that maps an image to a vector. The relevance $f(\cdot, \cdot)$ between the query image and images in the corpus is employed to sort images by decreasing similarity. Then, the knowledge retriever returns all captions associated with the selected images, as a source of conditioning for the language model.

To model the visual embedding function, we employ the visual encoder of one of the CLIP models [215], which have been trained contrastively to match image-text pairs. Empirically, we found this relevance function to be more robust in our scenario when compared to vision-only descriptors, as also reported in recent literature [16, 245]. While the maximum inner product search is carried out employing visual queries and values in Eq. 5.1, the search is implicitly multimodal as it happens inside a visual-semantic space. In contrast to performing a pure multi-modal search with visual queries and textual keys, however, our strategy is computationally lighter as it does not require to forward through a textual encoder.

Specifically, we select a CLIP ResNet-based [96] visual encoder. In this kind of encoder, the image is processed through a sequence of residual layers, then the grid of activations from the last convolutional layer is fed to an attention pool layer. Here, a single query is built from the global average-pooled feature vector, and all elements of the grid act as keys and values.

To get a more fine-grained representation of the image and have a higher control on the pooling strategy, we directly take the grid of features from the last convolutional layer and define the Embed function as an aggregation (*e.g.* average, max) of the features contained in the grid (see Fig.5.1).

5.2.2 Designing of Retrieval-Augmented Language Models

Given an external memory from which a set of relevant captions $\{z_i\}_i$ can be extracted, we now discuss the design of a retrieval-augmented language model $p(y|\{z_i\}_i, I)$, which is in charge of **predicting** the output caption while being conditioned on both the input image and items retrieved from the external memory. Compared to a traditional image captioning model, which only models $p(y|I)$, a retrieval-augmented model must implement a connection between its inherent language modeling capabilities, which in a Transformer-based model take place in self-attention layers, and the sequences of tokens that form the retrieved captions. The framework we employ for caption generation is an encoder-decoder Transformer [270], where the encoder is in charge of processing the input image, while the decoder acts as language model.

Visual Encoder. The input of the encoder consists of a flattened sequence of grid feature vectors (as described in Sec. 5.2.1) which are linearly projected into a vector space. Each encoder layer is then composed of a self-attention layer and a feed-forward layer, as in the standard Transformer [270].

In particular, all intra-modality interactions between image-level features are modeled via scaled dot-product attention, without using recurrence. Attention operates on three sets of vectors, namely a set of queries $\boldsymbol{Q}$, keys $\boldsymbol{K}$ and values $\boldsymbol{V}$, and takes a weighted sum of value vectors ac-

cording to a similarity distribution between query and key vectors. In the case of scaled dot-product attention, the operator is defined as

$$\text{Attention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d}}\right)\boldsymbol{V}, \tag{5.2}$$

where $\boldsymbol{Q}$ is a matrix of n_q query vectors, $\boldsymbol{K}$ and $\boldsymbol{V}$ both contain n_k keys and values, all with the same dimensionality, and d is a scaling factor.

Given a set of grid image features $\boldsymbol{X}$ extracted from an input image, attention is used to obtain a permutation invariant encoding of $\boldsymbol{X}$ through the self-attention operations used in the Transformer [270]. In this case, queries, keys, and values are obtained by linearly projecting the input features, and the operator can be defined as

$$\mathcal{S}(\boldsymbol{X}) = \text{Attention}(W_q\boldsymbol{X}, W_k\boldsymbol{X}, W_v\boldsymbol{X}), \tag{5.3}$$

where W_q, W_k, W_v are matrices of learnable weights. The output of the self-attention operator is a new set of elements $\mathcal{S}(\boldsymbol{X})$, with the same cardinality as $\boldsymbol{X}$, in which each element of $\boldsymbol{X}$ is replaced with a weighted sum of the values, *i.e.* of linear projections of the input (Eq. 5.2). The output of the self-attention attention is then applied to a position-wise feed-forward layer composed of two affine transformations with a single non-linearity, which are independently applied to each element of the set. Each of these sub-components (self-attention and position-wise feed-forward) is then encapsulated within a residual connection and a layer norm operation. The complete definition of an encoding layer can be finally written as:

$$\begin{aligned} \boldsymbol{J} &= \text{AddNorm}(\mathcal{S}(\boldsymbol{X})) \\ \tilde{\boldsymbol{X}} &= \text{AddNorm}(\mathcal{F}(\boldsymbol{J})), \end{aligned} \tag{5.4}$$

where $\mathcal{F}$ indicates a feed-forward layer and AddNorm indicates the composition of a residual connection and of a layer normalization.

Given the aforementioned structure, multiple encoding layers are stacked in sequence, so that the i-th layer consumes the output set computed by layer $i-1$. This amounts to creating multi-level encodings of the relationships between image features, in which higher encoding layers can exploit and refine relationships already identified by previous layers. A stack of N encoding layers will therefore produce an output $\tilde{\mathcal{X}} = \tilde{\boldsymbol{X}}^N$, obtained from the output of the last encoding layer.

Textual Decoder. The decoder, instead, takes as input the sequence of tokens comprising the ground-truth caption and is asked to predict a left-shifted version of it. The self-attention here is masked so that each token can attend only elements to its left, and the decoder then effectively models an autoregressive generation process. Each layer of the decoder comprises at least one self-attention layer, a cross-attention layer with the encoder output, and one feed-forward layer.

Given a sequence of vectors $\boldsymbol{Y}$, and outputs from the last encoder layer $\tilde{\mathcal{X}}$, the cross-attention connects $\boldsymbol{Y}$ to all elements in $\tilde{\mathcal{X}}$. Formally, the operator uses queries from the decoder and keys and values from the encoder:

$$\mathcal{C}(\tilde{\boldsymbol{X}}^i, \boldsymbol{Y}) = \mathrm{Attention}(W_q \boldsymbol{Y}, W_k \tilde{\boldsymbol{X}}^i, W_v \tilde{\boldsymbol{X}}^i). \tag{5.5}$$

As the prediction of a word should only depend on previously predicted words, the decoder layer comprises a masked self-attention operation that connects queries derived from the t-th element of its input sequence $\boldsymbol{Y}$ with keys and values obtained from the left-hand subsequence, *i.e.* $\boldsymbol{Y}_{\leq t}$. Also, the decoder layer contains a position-wise feed-forward layer, and all components are encapsulated within $\mathrm{AddNorm}$ operations. The final structure of the decoder layer can be written as:

$$\begin{aligned} \boldsymbol{J} &= \mathrm{AddNorm}(\mathcal{C}(\tilde{\mathcal{X}}, \mathrm{AddNorm}(\mathcal{S}_{\mathsf{mask}}(\boldsymbol{Y})))) \\ \tilde{\boldsymbol{Y}} &= \mathrm{AddNorm}(\mathcal{F}(\boldsymbol{J})), \end{aligned} \tag{5.6}$$

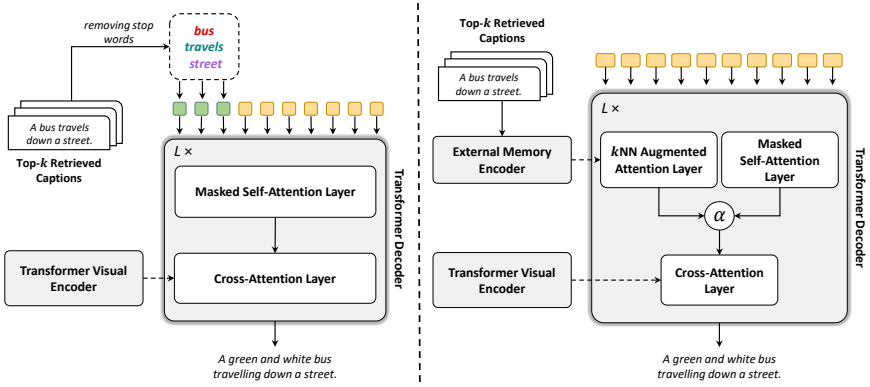


Figure 5.2: Architectural schema of the $RA{-}T^S$ (self-attention-based) and $RA{-}T^X$ (cross-attention-based) language models.In $RA-T^S$, retrieved captions are employed as prefix of the decoder textual sequence, after removing stop words and duplicate words. In $RA-T^X$, instead, retrieved captions are first passed through a Transformer encoder and then used in a kNN cross-attention layer inside the captioner decoder. The contribution of retrieved captions is regulated by a learnable gating mechanism that combines the output of the kNN cross-attention layer with those of the standard self-attention over the input sequence.

where $\boldsymbol{Y}$ is the input sequence of vectors and $\mathcal{S}_{\text{mask}}$ indicates a masked self-attention over time. Finally, our decoder stacks together multiple decoder layers, helping to refine both the understanding of the textual input and the generation of the next tokens. Overall, the decoder takes as input word vectors, and the t-th element of its output sequence encodes the prediction of a word at time $t+1$, conditioned on $\boldsymbol{Y}_{\leq t}$. After a linear projection and a softmax, this encodes a probability over tokens in the dictionary.

Generation Stage. We devise two retrieval-augmented transformer (**RA-T**) variants for realizing the connection between the decoder self-attention and items retrieved from the external memory, one based on self-attentive connections, termed $RA-T^S$, and one based on cross-attention connections, $RA-T^X$. The two different architectures are shown in Fig. 5.2.

$RA-T^S$**.** Under this configuration, retrieved captions are employed as a prefix of the decoder sequence, so that the self-attention operator can naturally retrieve relevant suggestions coming from the external memory

while generating a caption. This might also be seen as a variant of the prompting technique [10]. A naïve concatenation of the retrieved captions would be computationally intractable with the growth of k; furthermore, the self-attention layer would not have a principled way of distinguishing retrieved and generated tokens.

Therefore, we adopt two strategies: we clean the retrieved captions by removing stop words and eliminating duplicate words that appear in more than one caption, so to obtain a set of unique words. Formally, the input of the decoder can be defined as $\boldsymbol{Y}_{\text{RA-T}} = [\text{unique}(\{z_i\}_i), \boldsymbol{Y}]$, where $[\cdot, \cdot]$ indicates concatenation, and unique$(\cdot)$ indicates a function that removes stop words and eliminates duplicates.

To increase the effectiveness of this strategy, we also employ two different learnable segment embeddings [68] to distinguish between retrieved words and generated ones (*i.e.*, $\boldsymbol{Y}$). Also, as cleaned words represent an unordered set, we do not apply position embeddings to this segment, so to keep the permutation invariance of the self-attention operator.

$\boldsymbol{RA-T^X}$**.** In this case, an additional cross-attention layer is placed in parallel to the masked self-attention layer of the decoder. Retrieved captions are firstly encoded independently through a bidirectional Transformer encoder to get a refined representation of the tokens, then the aforementioned cross-attention layer performs a cross-attention over the resulting outputs. As the cross-attention layer is placed in parallel to the masked self-attention layer, the same queries are employed for both layers. Formally, given the input sequence of tokens $\boldsymbol{Y}$ and the set of retrieved captions $\boldsymbol{Z} = \{z_i\}_i$, this configuration can be written as follows:

$$\tilde{\boldsymbol{Z}} = \text{Encoder}(\boldsymbol{Z}) \tag{5.7}$$

$$\tilde{\boldsymbol{L}} = \text{AddNorm}(\mathcal{S}_{\text{mask}}(\boldsymbol{Y})) \tag{5.8}$$

$$\tilde{\boldsymbol{M}} = \text{AddNorm}(\mathcal{C}(\tilde{\boldsymbol{Z}}, \boldsymbol{Y})), \tag{5.9}$$

where $\mathrm{Encoder}$ indicates a Transformer encoder, such as the one employed for visual features encoding. The second equation refers to the self-attention between tokens of the caption. The last equation, instead, refers to the additional cross-attention operation with retrieved captions. Noticeably, in this layer, all tokens from all retrieved captions are attended.

Finally, the outputs coming from the two parallel layers need to be combined. To this aim, we devise a learnable gate, with which the model can regulate the importance of the output coming from the self-attention layer and that coming from the cross-attention layer. Conceptually, this amounts to choosing between the local context encoding and retrieved captions. Formally,

$$\tilde{\boldsymbol{J}} = \alpha \cdot \tilde{\boldsymbol{L}} + (1 - \alpha)\tilde{\boldsymbol{M}}, \tag{5.10}$$

where α represents the learnable gate. In practice, this is learned as the sigmoid of a single scalar network parameter. The output of this linear combination is then passed to the usual cross-attention with visual features and the feed-forward network to obtain the output sequence.

At training time the input of the decoder is the ground-truth sentence $\{\text{BOS}, \hat{y}_1, \hat{y}_2, ..., \hat{y}_n\}$, and the model is trained with a cross-entropy loss to predict the shifted ground-truth sequence, *i.e.* $\{\hat{y}_1, \hat{y}_2, ..., \hat{y}_n, \text{EOS}\}$. Here, BOS and EOS denotes special tokens that explicitly mark the beginning and the end of the caption, respectively. This training setup, commonly referred to as teacher forcing, allows the model to condition each prediction on the correct preceding tokens, facilitating stable and efficient learning.

While during training, the model predicts all output tokens simultaneously, the prediction process at inference happens sequentially. In each step, the model takes the partially decoded sequence as input, then selects the next token by sampling from its output probability distribution, continuing this process until an EOS marker is generated.

5.2.3 The Effectiveness of Retrieval-augmentation

Datasets and Evaluation Protocol. We first analyze the effectiveness of our retrieval-augmented architecture on the COCO dataset [158]. To train and test our solution, we follow the splits defined by Karpathy *et al.* [117]. However, there are also 30,504 images that were originally in the validation set of the original COCO dataset but were left out in this split. As done by previous image captioning [64, 108, 206, 7] and image-text matching [137, 52, 298, 149, 192] literature, we add these images in the training set thus obtaining a total of 113,287 images to train the model. Additionally, we perform experiments on the nocaps dataset [3]. contain both in-domain and out-of-domain object classes. Under this setting, we train our model on COCO and evaluate on nocaps validation set, submitting generated captions to the nocaps evaluation server*.

Following captioning literature, we measure the performance of our approach using the standard captioning evaluating metrics. During evaluation, we compare each generated caption with all ground-truth sentences associated with the corresponding image, using the COCO caption evaluation library† to obtain the final scores.

Retrieval Index. We build our retrieval index on the COCO training set. During training, to reduce overfitting risks, we avoid retrieving captions that belong to the current training image. We employ approximate *k*NN search rather than exact *k*NN search because it significantly improves the computational speed of our model. To this aim, we employ the Faiss library [115] and a graph-based HNSW index with 32 links per vertex, which has a size of 6.7 GB. For simplicity, we do not employ any vector transform (*e.g.* PCA) or vector quantization, although they might be employed to reduce the index size and scale to larger datasets.

*https://eval.ai/web/challenges/challenge-page/355/overview
†https://github.com/tylin/coco-caption

Implementation Details. To represent images, we employ CLIP-RN50×16 [215] intermediate features. To represent words, of both the input subsequence and retrieved sentences, we use Byte Pair Encoding (BPE) [238] with a vocabulary size of $49,408$. We use standard sinusoidal positional encodings [270] to represent word positions. For efficiency, the length of the output token sequence is limited to 40 tokens. Visual features and word tokens are projected into d-dimensional vectors with $d = 384$ and fed to our Transformer-based model, which has $L = 3$ layers in both encoder and decoder with six attention heads. The external memory encoder has the same number of heads and dimensionality as the rest of the model. The gate α is initialized to zero at the beginning of the training.

Pre-training with cross-entropy loss is performed using the LAMB optimizer [306] and following the learning rate scheduling strategy of [270] with a warmup equal to 6,000 iterations and a batch size of 1,080. For the CIDEr-based fine-tuning, we adopt the SCST strategy [222] sampling over the $k = 5$ best sequences from a beam-search scheme, using Adam [125] as optimizer, a batch size equal to 80, and a fixed learning rate of 5×10^{-6}.

Experiments are performed training on two Quadro RTX-5000 GPUs, using five gradient accumulation steps during both cross-entropy pre-training and CIDEr optimization. ZeRo memory offloading [218] and mixed-precision [195] are used to accelerate training and save memory.

Quality of Nearest Neighbor Captions. To confirm that nearest neighbor captions are a suitable source of additional knowledge and that can be thus employed to improve the final performance, we first need to evaluate their relevance with respect to the ground-truth captions. To do this, given an image from the test set, we retrieve the k nearest captions from one of the created nearest neighbor indexes using our relevance function to compare visual elements (*i.e.* in this experiment we use a standard average pooling to aggregate image features). Then, we measure the similar-

Table 5.2: Performance of the k nearest-neighbor captions using distinct indexes.

	$k=5$						$k=10$					
	B-1	B-4	M	R	C	S	B-1	B-4	M	R	C	S
index: *COCO*												
mean score	49.4	10.6	1.70	36.1	44.1	12.0	49.2	10.4	16.8	35.9	43.1	11.8
max score (Oracle)	65.5	14.4	24.8	49.4	77.8	19.1	72.3	22.1	28.5	55.1	96.5	22.6
index: *CC3M (original)*												
mean score	25.5	0.3	9.3	20.4	13.5	5.8	25.3	0.3	9.2	20.2	13.0	5.6
max score (Oracle)	40.9	1.4	15.2	31.9	32.1	11.6	46.5	2.6	17.4	36.0	40.4	13.8
index: *CC3M*												
mean score	59.2	10.7	20.9	44.2	61.4	13.8	58.7	10.4	20.6	43.8	59.8	13.6
max score (Oracle)	74.1	24.9	28.6	56.4	101.8	21.2	78.5	31.4	31.3	60.3	116.6	23.8
	$k=20$						$k=40$					
	B-1	B-4	M	R	C	S	B-1	B-4	M	R	C	S
index: *COCO*												
mean score	48.9	10.2	16.7	35.7	42.1	11.6	48.6	10.0	16.5	35.4	41.1	11.4
max score (Oracle)	77.7	30.8	31.9	60.2	114.2	25.4	82.0	39.4	34.9	64.5	130.4	28.0
index: *CC3M (original)*												
mean score	25.2	0.3	9.1	20.1	12.6	5.5	25.0	0.3	9.0	20.0	12.2	5.3
max score (Oracle)	51.8	4.2	19.3	39.9	49.3	16.1	56.6	6.9	21.3	43.7	58.6	18.1
index: *CC3M*												
mean score	58.2	9.9	20.3	43.4	58.1	13.3	57.6	9.5	20.0	43.0	56.3	13.0
max score (Oracle)	81.9	37.7	33.7	63.6	129.4	25.9	85.1	43.6	36.0	66.7	141.8	27.8

ity between retrieved and ground-truth captions by calculating the mean captioning scores and the score of the retrieved caption with the highest similarity to the ground-truth. The latter can be considered as an upper-bound score, where an oracle evaluator is used to select the best caption among the k nearest ones. We perform this analysis using three different retrieval indexes: one containing image-text pairs from the COCO dataset and the others containing elements from the CC3M dataset, either using the original CC3M textual descriptions (*i.e. CC3M (original)*) or the textual sentences predicted by the BLIP model [148].

The results are presented in Table 5.2 as the number k of retrieved sentences varies. As it can be noticed, retrieving a limited number of captions (*e.g.* $k = 5$) leads, for all retrieval indexes, to a set of captions that only partially correlates with the ground-truth. On the other hand, increasing the number of retrieved captions slightly degrades the performance, with a decrease in the mean CIDEr score from 44.1 to 41.1 when using the index

containing COCO elements. The maximum (oracle) score, instead, shows considerably higher results. Specifically, it reaches up to 130.4 and 141.8 CIDEr points, respectively using the COCO and CC3M indexes, when retrieving a large number of captions (*i.e.* $k = 40$). The worst results, in terms of both mean and maximum scores, are obtained with the retrieval index with the original CC3M corpus which leads to 58.6 CIDEr points in terms of maximum score using $k = 40$. These results can be explained by the quality of CC3M textual sentences which are crawled from the web and, although semantically richer, have a substantially different style from the human-annotated captions contained in the COCO dataset.

Although our embedding space is built on top of state-of-the-art descriptors, the high quality achieved by the oracle captions for higher values of k indicates that there is still significant room for improvement in the quality of the embedding space. It is worth noting, also, that the quality of the captions plays a crucial role in determining the quality of the embedding space, and it is not only dependent on the size of the retrieval index, as demonstrated by the results using original CC3M captions. Therefore, even a smaller index with high-quality captions can potentially result in a better embedding space than a larger one with lower-quality descriptions.

Role of Different Aggregation Functions. We then move to our full model, and first analyze the results of different aggregation functions to embed visual features and retrieve the most similar images. Specifically, we consider a standard average pooling over grid features, a max pooling, and a sum of ℓ_2-normalized features followed by an ℓ_2-norm of the result, which has demonstrated to be effective in previous image and video retrieval works [262]. Results are reported in Table 5.3, after cross-entropy pre-training, in comparison with a standard Transformer-based encoder-decoder model without retrieval. We can first notice that *all configurations with the external memory encoder achieve better performance than the*

Table 5.3: Performance of the two versions of our retrieval-augmented Transformer, by varying the number of retrieved sentences k and the aggregation function used. Results are reported after cross-entropy pre-training using the COCO index.

		$RA-T^S$						$RA-T^X$					
Aggregation Function	k	B-1	B-4	M	R	C	S	B-1	B-4	M	R	C	S
ℓ_2-norm sum	5	78.2	38.0	28.5	58.0	122.2	21.6	78.3	38.6	28.9	58.3	123.1	21.8
ℓ_2-norm sum	10	77.9	37.5	28.4	57.6	121.5	21.7	78.5	38.6	28.8	58.3	122.7	22.1
ℓ_2-norm sum	20	77.9	38.0	28.7	57.9	123.0	21.7	78.3	38.6	28.9	58.3	123.8	21.9
ℓ_2-norm sum	40	78.5	38.1	28.6	58.0	122.9	21.8	78.2	39.1	28.7	57.9	122.8	22.0
max	5	78.5	38.2	28.8	58.3	122.7	21.8	78.6	38.6	28.9	58.3	123.6	22.0
max	10	78.2	38.1	28.4	58.0	123.1	21.6	78.3	38.5	28.9	58.2	123.8	22.2
max	20	78.4	38.3	28.6	58.4	123.1	21.7	78.3	38.6	29.0	58.3	124.0	22.1
max	40	79.0	38.3	28.6	58.4	122.9	21.9	78.3	38.3	28.9	58.3	123.6	22.0
mean	5	78.7	38.3	28.7	58.1	123.3	22.0	78.6	38.7	29.1	58.5	124.0	22.0
mean	10	78.5	38.4	28.9	58.1	**123.7**	21.9	78.9	38.9	28.9	58.5	**124.5**	22.1
mean	20	78.6	38.3	28.6	58.2	123.4	21.8	78.5	38.6	28.9	58.3	124.2	22.0
mean	40	78.4	37.8	28.7	58.3	122.9	21.0	78.4	38.4	28.9	58.3	123.1	22.0

baseline which obtains 121.6 CIDEr points, thus demonstrating the effectiveness of our retrieval-augmented architecture. When comparing the different aggregation functions, the results show that a standard mean of grid features performs generally better than the other considered aggregation functions, also according to a different number k of retrieved captions.

Results on COCO. In Table 5.4 we report the results on the standard Karpathy test split after CIDEr-based finetuning, comparing our model performance with that of different state-of-the-art captioning models. Although several architectures pre-trained on large-scale datasets and then fine-tuned on COCO have recently been proposed [150, 316, 106], in this analysis we only consider captioning models trained exclusively on the COCO dataset. Specifically, we compare against methods with language models based on LSTMs such as Up-Down [7], eventually enhanced with spatial and scene graphs like GCN-LSTM [299] and SGAE [295] or self-attentive mechanisms such as AoANet [108], X-LAN [206], and DPA [164]. Moreover, we consider captioning architectures entirely based on the standard Transformer model such as ORT [97], $\mathcal{M}^2$ Transformer [64], X-Transformer [206], and RSTNet [320], even combining visual features from multiple backbones as in the case of DLCT [185] and DIFNet [284].

Table 5.4: Comparison with state-of-the-art models on the Karpathy-test split. Overall best results are underlined.

	B-1	B-4	M	R	C	S
Up-Down [7]	79.8	36.3	27.7	56.9	120.1	21.4
ORT [97]	80.5	38.6	28.7	58.4	128.3	22.6
GCN-LSTM [299]	80.9	38.3	28.6	58.5	128.7	22.1
SGAE [295]	81.0	39.0	28.4	58.9	129.1	22.2
AoANet [108]	80.2	38.9	29.2	58.8	129.8	22.4
$\mathcal{M}^2$ Transformer [64]	80.8	39.1	29.2	58.6	131.2	22.6
X-LAN [206]	80.8	39.5	29.5	59.2	132.0	23.4
X-Transformer [206]	80.9	39.7	29.5	59.1	132.8	23.4
DPA [164]	80.3	40.5	29.6	59.2	133.4	23.3
DLCT [185]	81.4	39.8	29.5	59.1	133.8	23.0
RSTNet [320]	81.8	40.1	29.8	59.5	135.6	23.3
DIFNet [284]	81.7	40.0	29.7	59.4	136.2	23.2
Transformer (w/o external memory)	81.9	39.7	29.6	59.4	135.3	**23.6**
$RA-T^S$ (index: *COCO*)	82.0	40.1	29.6	59.4	136.4	23.2
$RA-T^S$ (index: *CC3M*)	**82.5**	**40.8**	**29.7**	**59.8**	**136.7**	**23.6**
Transformer (w/o external memory)	81.9	39.7	29.6	59.4	135.3	23.6
$RA-T^X$ (index: *COCO*)	**82.4**	40.5	29.8	**59.8**	136.5	23.8
$RA-T^X$ (index: *CC3M*)	82.2	**41.0**	**30.0**	**59.8**	**136.7**	**23.9**

Results of both versions of our complete retrieval-augmented architecture are reported using both COCO and CC3M retrieval indexes and compared with those of a standard Transformer-based model without the retrieval component. As it can be seen, the efficacy of the kNN-augmented language model is confirmed even after finetuning with CIDEr-based optimization, with an increase of 1.1 and 1.2 CIDEr points respectively comparing $RA-T^S$ and $RA-T^X$ with COCO retrieval index to the standard Transformer-based architecture. The use of a larger index such as the one containing a cleaned version of CC3M captions can further boost the performance, leading to an overall CIDEr score of 136.7 for both model variations. It can also be noticed that, while after cross-entropy pre-training the $RA-T^X$ version slightly outperforms the $RA-T^S$ model, after reinforcement learning finetuning the two model variants perform comparably, thus demonstrating that both architectures can be a valid solution for incorporating external knowledge. Furthermore, we observe that the proposed

Table 5.5: Performances on nocaps validation set. We report the overall best results underlined.

	Near		Out		Overall	
	C	S	C	S	C	S
NBT [3]	61.2	9.9	62.4	8.9	60.2	9.5
Up-Down [3]	73.6	11.3	66.4	9.7	73.1	11.1
$\mathcal{M}^2$ Transformer [64]	75.4	11.7	69.4	10.0	75.0	11.4
Transformer (w/o external memory)	87.4	12.7	66.7	10.8	85.3	12.5
$RA-T^S$ (index: *COCO*)	88.2	12.5	68.6	10.6	86.3	12.3
$RA-T^S$ (index: *CC3M*)	**89.3**	**13.0**	**69.5**	**11.0**	**86.8**	**12.7**
Transformer (w/o external memory)	87.4	12.7	66.7	10.8	85.3	12.5
$RA-T^X$ (index: *COCO*)	88.5	12.8	**68.6**	11.0	86.3	12.6
$RA-T^X$ (index: *CC3M*)	**89.4**	**13.1**	**68.6**	**11.1**	**87.0**	**12.8**

retrieved-augmented model achieves promising and competitive performance compared to other state-of-the-art methods, and surpasses them in terms of all evaluation metrics.

Results on nocaps. In Table 5.5, we extend our analysis on the nocaps dataset, using $RA-T^S$ and $RA-T^X$ after finetuning with CIDEr optimization on the COCO dataset. Also in this case, we compare our results against a standard Transformer model and employ both versions of our retrieval index (*i.e.* the one containing COCO captions and the other composed of CC3M sentences predicted by the BLIP model). The effectiveness of the proposed retrieval-augmented strategy is confirmed also in this setting, with an improvement of 1.5 and 1.7 on the entire validation set, respectively for the self- and cross-attention model variants with CC3M index. These results highlight the robustness of the approach when applied to different configurations. The contribution of a larger retrieval index becomes more evident, especially on near-domain and out-of-domain image-text pairs, which contain visual concepts outside of the COCO dataset and thus can benefit from a larger and semantically richer set of retrievable items. In fact, the CIDEr score on out-of-domain images is equal to 69.5 for the $RA-T^S$ model with CC3M index compared to 68.6 achieved by the same version of the model augmented with COCO retrieval index.

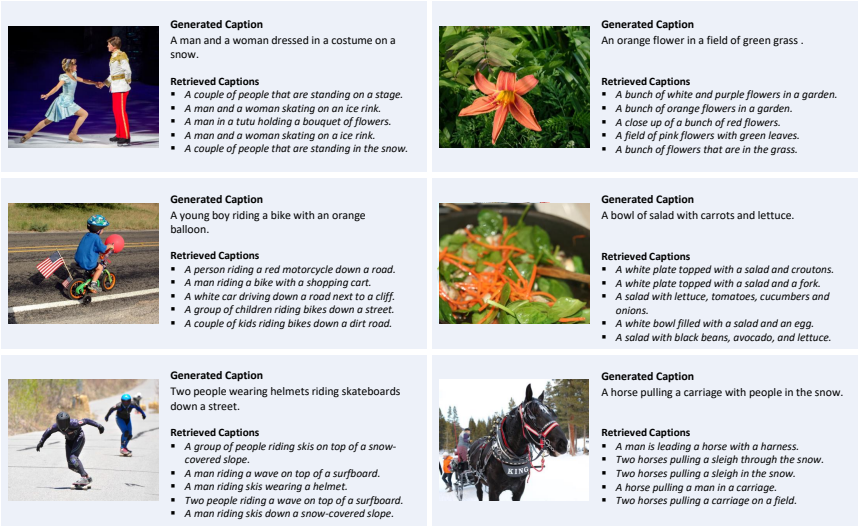


Figure 5.3: Generated captions on sample images from the COCO dataset, along with five retrieved captions.

Qualitative Results. Finally, in Fig. 5.3 and 5.4 we report sample captions generated by our model on images respectively from COCO and nocaps. For COCO (Fig. 5.3), we additionally display a subset of the captions retrieved from the external memory, so as to explicitly illustrate the type of textual evidence that is made available to the generator at inference time. For nocaps (Fig. 5.4), instead, we compare captions produced by our model using either the COCO or the CC3M index against those generated by a Transformer-based captioner without retrieval. Retrieving captions related to the input image helps the language model generate more relevant and accurate captions exploiting additional contextual information.

For example, in the top-right example of Fig. 5.3, we can observe that the caption generated by our model has highly comparable content to the retrieved sentences (*i.e.* "orange flower" and "green grass"), providing evidence of the efficacy of our retrieval-based approach.

When instead comparing our generated captions with those of a stan-

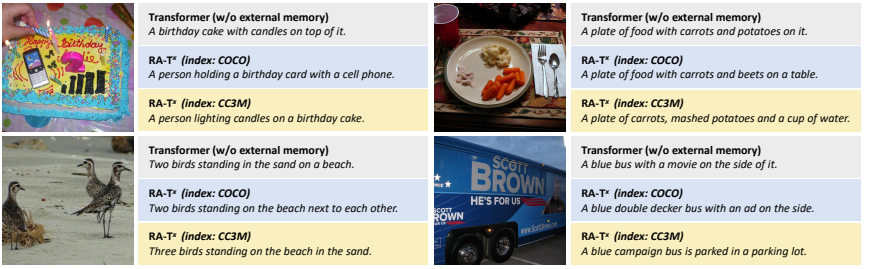


Figure 5.4: Results on sample images from nocaps, comparing captions generated by RA-T with those generated by a standard Transformer without retrieval.

dard Transformer model without retrieval (Fig. 5.4), we can observe that our results are generally more coherent with the visual content of input images and semantically richer, especially when using the retrieval index containing CC3M elements. For example, in the top-right image of Fig. 5.4, the baseline model without external memory correctly recognizes the "carrots" and "potataoes" on a "plate". However, the caption predicted by our $RA-T^X$ model with CC3M index is more detailed and complete, also describing the presence of a "cup of water" and recognizing that the potatoes are "mashed". Similar observations can be made for the other examples, further confirming from a qualitative point of view the effectiveness of our retrieval-augmented solution for image captioning.

5.3 Memory-Augmented Captioning Models

In both variants of RA-T, we observe the central role of attention operators. One of the key properties of attention layers is that their output is computed as a weighted combination of linear projections of the inputs. While this mechanism provides an effective framework for visual understanding and sequence generation, it also naturally allows the injection of additional information that cannot be directly inferred from the current input, for instance by incorporating relevant cues obtained through retrieval.

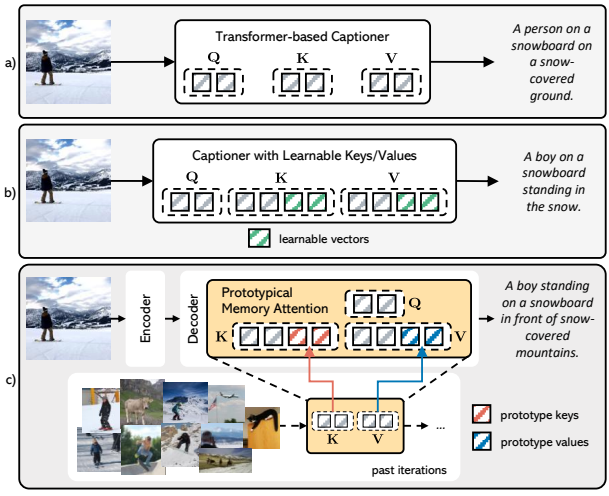


Figure 5.5: Comparison between (a) a standard Transformer-based captioner; (b) one with learnable memory vectors [64] and (c) our prototypical memory network.

Closely related, yet conceptually distinct from retrieval-based approaches, the Meshed-Memory architecture proposes to augment the visual encoder with additional learnable key–value vectors in order to inject a priori knowledge into the model. While effective, such learnable vectors are global parameters and therefore capture information that is shared across the entire training set, rather than acting as a true memory of individual past samples. In contrast, having access to specific training examples at generation time can provide a richer and more adaptive source of information, ultimately improving caption quality. For example, given an image depicting a boy snowboarding in a mountain landscape, a model that can access similar training images—containing boys, snowboards, and mountainous scenes, even in different contexts—may exploit this information compositionally to generate a more accurate and fluent description (Fig. 5.5).

We devise a *prototypical memory* network, which can recall and exploit past activations generated during training. Our memory is built upon network activations obtained during recent training iterations so that the net-

work has access to a vast set of activations produced while processing other samples. The memory represents *past knowledge processed by the network itself* and is fully integrated into attention layers through the addition of keys and values which represent activations from the memory.

Memory Augmented Attention. Attention layers operate on triplets of queries, keys and values $(\mathbf{Q}, \mathbf{K}, \mathbf{V})$, which are obtained by linearly projecting items from the same input sequence (self-attention) or from a pair of different input sequences (cross-attention). We are interested in breaking the constraint of operating exclusively on input-dependent data and letting the attention operator consider quantities which are not derived from the current input [64, 287]. In memory-augmented attention [64], this is achieved by extending the set of keys and values to include additional memory vectors. As a result, the attention operation can employ both input-dependent and memory-specific keys and values, as follows:

$$\begin{aligned} \tilde{\mathbf{K}} &= [\mathcal{M}_{\mathbf{K}}; \mathbf{K}(x)], \quad \tilde{\mathbf{V}} = [\mathcal{M}_{\mathbf{V}}; \mathbf{V}(x)] \\ \text{Attention}(\mathbf{Q}, \tilde{\mathbf{K}}, \tilde{\mathbf{V}}) &= \frac{\mathbf{Q}\tilde{\mathbf{K}}^{\intercal}}{\sqrt{d}}\tilde{\mathbf{V}} \end{aligned} \tag{5.11}$$

where the exclusive dependency between regular keys and values and the current input sample x has been made explicit, $[\cdot;\cdot]$ indicates concatenation, $\mathcal{M}_{\mathbf{K}}$ the set of memory keys and $\mathcal{M}_{\mathbf{V}}$ the set of memory values.

Previous memory-augmented works [252, 64, 213, 39, 291] have treated $\mathcal{M}_{\mathbf{K}}$ and $\mathcal{M}_{\mathbf{V}}$ as learnable parameters and, thus, optimized them directly through SGD during the learning process. This imposes a constraint on what can be stored in memory, as memories will be the result of accumulating gradient averaged over sequential mini-batches. This encourages the storage of information which is averagely beneficial to the entire training set and prevents focusing on the peculiarities of the single training items. As a consequence, learning a proper set of disentangled memory vectors turns out to be non-trivial and initialization-dependent [252].

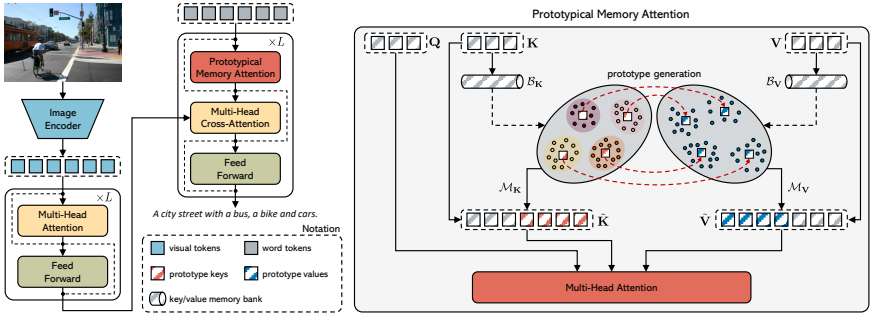


Figure 5.6: Overview of our approach with Prototypical Memory Attention.

5.3.1 Prototypical Memory Networks

We redefine memory keys and values as a means to let the network attend previous activations produced while processing other training samples. The network, at test time, will be able to attend to its own (past) activations produced while processing similar samples, thus aiding the generation process. Conceptually, we might see this as a more principled design of a *memory*, as in our case memory vectors will be actually storing past experiences of the network instead of being plain learnable parameters.

In our architecture, which we name **PMA-Net**, we apply memories in a core position of the encoder-decoder structure, *i.e.* inside each self-attention layer of the captioner. This is different from what has been done in previous works (*e.g.* [64] considered only the Transformer encoder), and also represents a privileged placement for vision-and-language architectures, as the self-attention layer is in charge of modeling the temporal consistency of the generation and of integrating it with the result of the previous cross-attention layer, which connects with the visual modality. Figure 5.6 (left) presents an overview of this design. Considering a stream of mini-batches $[\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_t, ...]$ containing randomly sampled training items, for each layer we define two memory banks $\mathcal{B}_\mathbf{K}$, $\mathcal{B}_\mathbf{V}$ which store all keys

and values produced from past training samples, up to a maximum temporal distance of T iterations. Intuitively, the two memory banks model the manifold of keys and values seen over past training iterations. We then define the set of memory keys and values to be employed at the t-th training iteration as a function of the vectors contained in the respective memory banks:

$$\begin{aligned} \mathcal{B}_{\mathbf{K}} &= [\mathbf{K}(\mathbf{x}_{t-1}), \mathbf{K}(\mathbf{x}_{t-2}), ..., \mathbf{K}(\mathbf{x}_{t-\mathrm{T}})] \,, \\ \mathcal{B}_{\mathbf{V}} &= [\mathbf{V}(\mathbf{x}_{t-1}), \mathbf{V}(\mathbf{x}_{t-2}), ..., \mathbf{V}(\mathbf{x}_{t-\mathrm{T}})] \,, \\ \mathcal{M}_{\mathbf{K}} &= f\left(\mathcal{B}_{\mathbf{K}}\right), \quad \mathcal{M}_{\mathbf{V}} = f\left(\mathcal{B}_{\mathbf{V}}\right). \end{aligned} \tag{5.12}$$

In the equations above, for ease of notation, we denote with $\mathbf{K}(\mathbf{x})$ the set of keys produced by a layer while processing all items contained in a mini-batch $\mathbf{x}$. In practice, the temporal window T should be chosen to be sufficiently large to reasonably model the training set distribution (as shown in Sec. 5.3.2). Also, memory banks need to be updated frequently and in a sufficiently smooth manner, so to follow the evaluation of the keys and values manifold and not to alter the training process, as will be discussed in the following.

Building memory prototypes. Naively placing all keys and values produced during a given time window in the memory (*i.e.* setting $f(\cdot)$ to the identity in Eq. 5.12) would require, approximately, $\mathrm{T} \cdot \mathrm{B} \cdot \mathrm{h} \cdot \tau$ memory slots per layer, where T represents the number of iterations executed inside the time window, B the mini-batch size, h the number of heads, and τ the average ground-truth sequence length. Under this setting, storing an entire COCO epoch would require storing around 96M memory vectors to both key and value sequences[‡], which would make the problem intractable in terms of memory occupation and computational complexity, because of the additional memory required to store vectors and because of the resulting growth of the attention matrix size. Further, as keys and values have been

[‡]Considering a network with 8 heads, and BPE tokenization.

trained to summarize the information contained in a token with respect to other tokens of the same sequence, multiple elements in the memory bank could produce similar attention scores, thus increasing the entropy of the resulting attention distributions.

For this reason, we instead build synthetic key/value pairs as *prototypical memory vectors* which are representative of the distribution of the entire memory bank. In doing this, we satisfy two design requirements: (1) building prototypes should be fast, as we will be performing this on every layer of the architecture and several times during training; (2) memory prototypes should evolve during training to adapt to the changing distribution of keys and values.

In our method, prototype key memory vectors are obtained by clustering the manifold identified by the memory bank of keys and taking the resulting centroids. Value memory vectors are, instead, computed by interpolating between the values corresponding to the keys that lie in each cluster. Formally, being m the target size of the memory, key memory vectors are:

$$\mathcal{M}_{\mathbf{K}} = \left[\mathcal{M}^1_{\mathbf{K}}, \mathcal{M}^2_{\mathbf{K}}, ..., \mathcal{M}^m_{\mathbf{K}}\right] = \text{K-Means}_m(\mathcal{B}_{\mathbf{K}}), \tag{5.13}$$

where function $\text{K-Means}_m(\cdot)$ returns the m centroids obtained by performing a K-Means clustering over the key memory bank. Value memory are computed by taking a linear combination of vectors in the value manifold that correspond to keys that lie close to key prototypes $\mathcal{M}^i_{\mathbf{K}}$, according to a distance function $d(\cdot)$ which compares items in the key manifold:

$$\begin{aligned} \mathcal{M}_{\mathbf{V}} &= \left[\mathcal{M}^1_{\mathbf{V}}, \mathcal{M}^2_{\mathbf{V}}, ..., \mathcal{M}^m_{\mathbf{V}}\right], \\ \mathcal{M}^i_{\mathbf{V}} &= \sum_{(\mathbf{K}^j, \mathbf{V}^j) \in \texttt{top-k}(\mathcal{M}^i_{\mathbf{K}})} e^{-d(\mathcal{M}^i_{\mathbf{K}}, \mathbf{K}^j)} \mathbf{V}^j, \end{aligned} \tag{5.14}$$

where, here, function $\texttt{top-k}(\mathcal{M}^i_{\mathbf{K}})$ returns the closest (key, value) pairs in the memory bank with respect to $\mathcal{M}^i_{\mathbf{K}}$.

It shall be noted that the `top-k` operation can be implemented by fitting a k-NN index on the keys memory bank ($\mathcal{B}_{\mathbf{K}}$) and using the centroid $\mathcal{M}^i_{\mathbf{K}}$ as query. The resulting list of keys close to $\mathcal{M}^i_{\mathbf{K}}$ can then be paired with their corresponding values to compute Eq. 5.14. We use the L_2 distance as distance function inside both the key and value manifolds, as we found it to perform favorably compared to the inner product.

Discussion. With the strategy defined above, we obtain a set of m memory keys and values, where m can be controlled a-priori. Taking prototypes as centroids ensures, when m is sufficiently high, that memory keys model the memory bank distribution properly. Further, the distance between centroids and cluster members in the key manifold is small, which has a positive effect on the resulting attention distribution, compared to the one obtainable by setting $f(\cdot)$ to the identity – *i.e.*, when storing all keys and values from the memory bank in the self-attention layer. Similar keys in an L_2 space, indeed, result in similar attention distributions.

Proposition – Given a query q and a set of keys $\mathbf{K}$, if a key $k \in \mathbf{K}$ is replaced with $\tilde{k}$ such that $\|k - \tilde{k}\|_2 \leq \varepsilon$ to form $\tilde{\mathbf{K}}$, then $\|\text{softmax}(q\mathbf{K}^\intercal) - \text{softmax}(q\tilde{\mathbf{K}}^\intercal)\|_2 \leq \varepsilon\|q\|_2$.

Proof. As the softmax operator has Lipschitz constant less than 1 [273, 81] and because the L_2 matrix norm is subordinate, $\|\text{softmax}(q\mathbf{K}^\intercal) - \text{softmax}(q\tilde{\mathbf{K}}^\intercal)\|_2 \leq \|q\mathbf{K}^\intercal - q\tilde{\mathbf{K}}^\intercal\|_2 \leq \|q\|_2\|(\mathbf{K} - \tilde{\mathbf{K}})^\intercal\|_2$. Recalling that, for any matrix A, $\|A\|_2 \leq \|A\|_F$, $\|(\mathbf{K} - \tilde{\mathbf{K}})^\intercal\|_2 \leq \|k - \tilde{k}\|_2$, from which the thesis follows.

Memory bank update. To ensure that memories are refreshed during training while lowering the total cost of prototypes generation, we adopt a strided sliding window approach to update the memory banks, which is visually depicted in Fig. 5.7. Chosen a maximum length T for the banks (Eq. 5.12), at regular intervals we take the last T batches from the key/value stream produced by a layer, create a memory bank with those and generate prototype vectors to be placed in $\mathcal{M}_{\mathbf{K}}$ and $\mathcal{M}_{\mathbf{V}}$ (Eq. 5.13, 5.14).

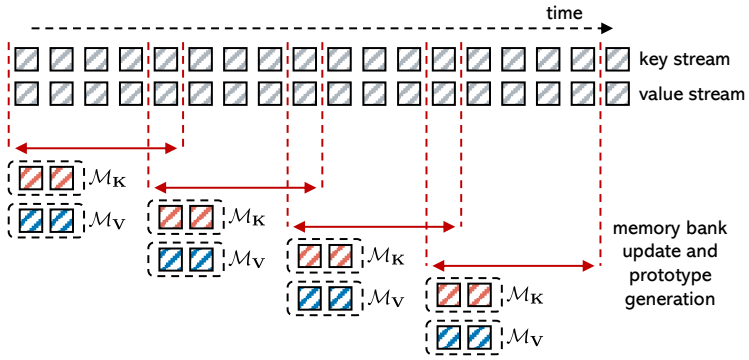


Figure 5.7: Memory bank update approach.

In practice, the process is repeated twice per epoch and memory banks store around two epochs of samples, so to have a significant overlap between the memory banks obtained at two consecutive update steps, which helps to stabilize the training. This process is also illustrated in pseudo-code in Algorithm 1 for clarity.

Segment embeddings. As the final set of keys of the layer is a concatenation of memory-specific and input-specific keys (Eq. 5.11), we add two different, learnable, segment embeddings to $\mathbf{K}$ and $\mathcal{M}_\mathbf{K}$, to help the network distinguish between the two key types.

Computational complexity. Computing memory prototypes requires executing a K-Means clustering over the key memory bank ($\mathrm{T}\cdot\mathrm{B}\cdot\mathrm{h}\cdot\tau$ datapoints, m clusters) and a kNN search over the key memory bank (which contains the same number of items) and is executed every s training steps, being s the stride employed over the key/value stream to update the memory banks (Fig. 5.7). Further, the addition of the m memory vectors to a mini-batch having a sequence length of T implies growing the attention matrix from $T \times T$ to $(T + m) \times T$. As prototype generation is only required at training time, the latter is the only cost that is added at test time with respect to a standard attention layer.

Algorithm 1 PMA-Net pseudocode

```
# m: number of prototypes
# T: maximum length of the memory bank
# stride: memory bank update stride
# bank_k, bank_v: key/value memory banks
bank_k = [], bank_v = []
for img, caption in dataloader:
  output, act_k, act_v = net(img, caption)
  bank_k.append(act_k)
  bank_v.append(act_v)
  if len(bank_k) == T:
    compute_prototypes(m, bank_k, bank_v) # Eq. 3, 4
  bank_k = bank_k[stride:]
  bank_v = bank_v[stride:]
  loss = loss_fn(output, caption)
  loss.backward()
```


In practice, adding prototypical memories does not increase inference times significantly with respect to a naive Transformer as the increase of the attention matrix is well amortized by the GPU parallelism. During training, computing the K-Means clustering and the k-NN index for solving Eq. 5.14 requires around 10s with a V100 GPU every time the memory needs to be refreshed. As the memory occupied for this can be de-allocated after prototypes computation, we did not need to decrease the batch size with respect to a baseline with learnable memory vectors.

5.3.2 Evaluation of Memory-Augmented Captioning Models

Dataset. We analyze the effectiveness of our PMA-Net on the widely used COCO benchmark [158] employing the splits defined in [117]. We also evaluate on the COCO online test server composed of more than 40k images for which ground-truth captions are not publicly available.

Additionally, we perform experiments on robust COCO, a different split of the COCO dataset introduced in [180] to verify sensitivity to object hal-

lucination and nocaps [3] for novel object captioning. The former dataset guarantees that object pairs mentioned in captions of different sets do not overlap (with 110,234, 3,915, and 9,138 images for training, validation, and test), while the latter contains images annotated with 10 human-written captions, that can be further divided in in-domain, near-domain and out-of-domain pairs depending on their nearness to COCO.

To evaluate our results, we employ all standard captioning metrics, namely BLEU [207], METEOR [15], ROUGE [154], CIDEr [271], and SPICE [5], and some more recent evaluation scores like BERT-S [318], CLIP-S [98], and PAC-S [229] in both their reference-free and reference-based versions. When evaluating our results on robust COCO, we also employ the CHAIR metric [224] that measures which fraction of objects mentioned in the generated sentences is hallucinated (CHi) and the portion of sentences that includes a hallucinated object (CHs).

Implementation details. Both the encoder and decoder are constructed with $L = 6$ Transformer layers, with a hidden size of 512 and 8 attention heads. Unless otherwise specified, we employ 1,024 memory vectors and a size of the memory banks T equal to 1,500. We use a CLIP [215] ViT-L/14 image encoder. The rationale behind this choice is that they have higher quality, adaptability to different tasks, and lower computational load compared to detection-based ones. To ensure fair comparison, we re-train recent and publicly-available models using the same features.

Our code is based on Huggingface [281], using the GPU-based implementations of K-Means and k-NN search from FAISS [115]. At training stage, the overall objective of PMA-Net is the typical cross-entropy loss for generation. Next, following [222], PMA-Net can be further optimized with sentence-level reward, using the CIDEr score. Specifically, we first pre-train with the LAMB optimizer [306], a batch size of 1,024 and for 20,000 steps. We use the following learning rate schedule: we linearly warmup for 1,000 steps, then

Table 5.6: Ablation study (m is the number of memory vectors and T is the size of the memory banks).

	m	T	B-4	M	R	C	S
Transformer [270]	-	-	37.4	30.3	58.9	127.8	23.3
Transformer (w/ learnable mem.)	64	-	37.7	30.2	58.1	127.9	23.4
Transformer (w/ learnable mem.)	1024	-	37.2	30.1	58.3	127.6	23.3
PMA-Net	256	1500	38.8	30.1	59.4	129.4	23.5
PMA-Net	512	1500	39.0	30.1	59.5	130.0	23.5
PMA-Net	1024	500	37.8	30.3	59.0	128.6	23.5
PMA-Net	1024	1000	38.2	**30.5**	59.5	129.4	23.5
PMA-Net (w/o mem. in 1st layer)	1024	1500	38.3	30.2	59.0	129.2	23.3
PMA-Net (w/o segment emb.)	1024	1500	38.6	30.4	59.4	130.1	23.4
PMA-Net	1024	1500	**39.5**	30.4	**59.6**	**131.5**	**23.6**

keep a constant learning rate of $2.5 \cdot 10^{-4}$ until 10,000 steps, then sub-linearly decrease until 15,000 steps to 10^{-5} and keep the value constant for the rest of the training. For the second stage, we further optimize PMA-Net using the Adam optimizer [125] and with $1 \cdot 10^{-6}$ as learning rate, for 50,000 steps using a batch size of 64. We employ a beam size equal to 5.

Ablation studies. We conduct an ablation study to investigate how each design choice in our PMA-Net influences the overall performance on COCO dataset. Table 5.6 details the performance comparisons among different ablated runs. Note that all the results reported here are without self-critical training strategy. We start from a base Transformer encoder-decoder architecture, which is also a degraded version of PMA-Net without memory banks and prototype vectors. Subsequently, we compare by adding learnable memory vectors as defined in [252, 64] but in the same position of PMA-Net, *i.e.* in place of the self-attention layer in the sentence decoder instead of the visual encoder. Then, we add memory banks and prototype vectors and vary the number of clusters and the size of the memory banks.

Firstly, we notice that the basic learnable memory vectors do not give a significant contribution when placed in the sentence decoder, outlining that in this core position of the captioner, in which activations coming

from both modalities are merged, learning appropriate memory vectors becomes more complex. Instead, the proposed prototype vectors increase caption quality significantly, up to 3.7 CIDEr points, highlighting the appropriateness of the proposed strategy. We notice that increasing the number of clusters and the size of the memory banks exhibits better performances, as we hypothesize that this provides a better estimation of the key and value manifold and more fine-grained prototypes.

The sliding window contains the last $\mathrm{T} \cdot \mathrm{B}$ captions seen, which in our best configuration amounts to 1.5M samples. Being COCO 0.6M image-text pairs, this models the training set distribution and its evolution across more than two epochs. Increasing T further does not enhance performance; reducing it, especially to less than one epoch, is instead detrimental.

In the lower part of Table 5.6, we run additional ablations on two design choices: the use of segment embeddings to distinguish prototypes from input-dependent keys, and the incorporation of prototypical vectors in the first layer of the captioner. The second experiment arises from the fact that the first layer is not influenced by cross-attention results and, therefore, by multimodal connections with the input image. It can be observed that the segment embeddings provide a relevant contribution, and that memory prototypes have an impact both on the first layer and on the other layers, outlining that its advantage is inherently multimodal.

Results on COCO. We then compare PMA-Net with different state-of-the-art approaches. In Table 5.7 we report results on the standard Karpathy test split, in a single-model setting. The upper part of the table shows the results reported by the compared approaches, using their original features. In the lower part, instead, we re-train different approaches on the same CLIP grid features we employ for training PMA-Net. Specifically, in addition to a Transformer, we re-train the $\mathcal{M}^2$ Transformer [64] and CaMEL [18], which both represent recent and complementary approaches which could also be inte-

Table 5.7: Comparison with the state of the art on the Karpathy test. The † marker indicates models re-trained with the same visual features used by our approach, while ∗ indicates finetuning of the visual backbone.

	Cross-Entropy Loss								CIDEr Optimization							
	B-1	B-2	B-3	B-4	M	R	C	S	B-1	B-2	B-3	B-4	M	R	C	S
Up-Down [7]	77.2	-	-	36.2	27.0	56.4	113.5	20.3	79.8	-	-	36.3	27.7	56.9	120.1	21.4
GCN-LSTM [299]	77.3	-	-	36.8	27.9	57.0	116.3	20.9	80.9	-	-	38.3	28.6	58.5	128.7	22.1
SGAE [295]	77.6	-	-	36.9	27.7	57.2	116.7	20.9	81.0	-	-	39.0	28.4	58.9	129.1	22.2
AoANet [108]	77.4	-	-	37.2	28.4	57.5	119.8	21.3	80.2	-	-	38.9	29.2	58.8	129.8	22.4
$\mathcal{M}^2$ Transformer [64]	-	-	-	-	-	-	-	-	80.8	-	-	39.1	29.2	58.6	131.2	22.6
X-Transformer [206]	77.3	61.5	47.8	37.0	28.7	57.5	120.0	21.8	80.9	65.8	51.5	39.7	29.5	59.1	132.8	23.4
DLCT [185]	-	-	-	-	-	-	-	-	81.4	-	-	39.8	29.5	59.1	133.8	23.0
RSTNet [320]	-	-	-	-	-	-	-	-	81.8	-	-	40.1	29.8	59.5	135.6	23.3
DIFNet [284]	-	-	-	-	-	-	-	-	81.7	-	-	40.0	29.7	59.4	136.2	23.2
CaMEL [18]	78.3	-	-	39.1	29.4	58.5	125.7	22.2	82.8	-	-	41.3	30.2	60.1	140.6	23.9
COS-Net [152]	**79.2**	63.8	50.2	39.2	29.7	58.9	127.4	22.7	82.7	68.2	54.0	42.0	**30.6**	60.6	141.1	**24.6**
GRIT* [200]	-	-	-	-	-	-	-	-	84.2	-	-	42.4	30.6	60.7	144.2	24.3
Transformer†	76.4	61.0	47.9	37.4	30.3	58.9	127.8	23.3	83.4	68.6	54.2	42.0	30.0	60.6	140.3	23.5
$\mathcal{M}^2$ Transformer† [64]	78.8	63.3	49.5	38.7	29.6	58.9	127.8	23.3	83.7	69.2	54.8	42.3	30.5	61.0	141.2	23.6
CaMEL† [18]	78.8	63.5	50.3	39.2	30.0	59.3	129.9	23.4	83.6	69.0	54.7	42.4	**30.6**	60.9	142.4	23.6
PMA-Net	79.0	**64.2**	**50.7**	**39.5**	**30.4**	**59.6**	**131.5**	**23.6**	**83.8**	**69.3**	**55.0**	**43.0**	**30.6**	**61.1**	**144.1**	24.0

Table 5.8: Comparison with additional metrics. † indicates models re-trained with the same visual features used by our approach.

	Training	BERT-S	CLIP-S	RefCLIP-S	PAC-S	RefPAC-S
Transformer†	XE	0.947	0.741	0.804	0.819	0.865
$\mathcal{M}^2$ Transformer† [64]	XE	0.946	0.744	0.806	0.815	0.864
CaMEL† [18]	XE	0.947	0.745	0.807	0.818	0.865
PMA-Net	XE	**0.948**	**0.754**	**0.812**	**0.821**	**0.868**
Transformer†	SCST	**0.947**	0.749	0.807	0.818	0.864
$\mathcal{M}^2$ Transformer† [64]	SCST	0.946	0.749	0.809	0.817	0.865
CaMEL† [18]	SCST	0.945	0.751	0.810	0.818	0.865
PMA-Net	SCST	0.946	**0.755**	**0.814**	**0.821**	**0.869**

grated with our proposal. With respect to a standard Transformer, PMA-Net exhibits a margin of 3.8 CIDEr points also under CIDEr optimization, similarly to the XE training setting. Further, when compared with recent approaches using the same features, PMA-Net also provides better performance. As shown in the table, with the exception of GRIT [200] that differently from us finetunes the visual backbone, PMA-Net consistently outperforms all the state-of-the-art approaches according to all metrics. Notably, our model is still competitive even compared to GRIT, despite the latter using more powerful visual features. In Table 5.8, we also compare against baselines trained on the same features using more recent learnable metrics, *i.e.* BERT-

GT: A group of horse mounted police standing in front of a crowd.
Transformer: A group of police officers standing in a street.
PMA-Net: A group of police officers on horses in a street.

GT: A man takes a picture of snowy mountains with his cell phone.
Transformer: A man taking a picture of the mountains.
PMA-Net: A man taking a picture of mountains with a cell phone.

GT: A Subway sandwich with chips raisins and a coffee cup.
Transformer: A sandwich and a bag of chips on a table.
PMA-Net: A table with a sandwich and chips and a cup of coffee.

Figure 5.8: Qualitative results on COCO sample images.

S, CLIP-S and, PAC-S. To qualitatively validate the effectiveness of our solution, we report images and corresponding predicted captions in Fig. 5.8.

COCO Test Server. We also report the performances of our approach obtained on the official COCO test split, through the online test server[§]. Table 5.9 reports the performance with respect to 5 reference captions (c5) and 40 reference captions (c40). Following previous literature [64, 152], we report the results using an ensemble of four models. As it can be seen, also in this setting PMA-Net surpasses the compared approaches by a large margin, further demonstrating its effectiveness on the COCO dataset.

Robust COCO split and sensitivity to hallucination. As our approach relies on the memorization of other training samples, we verify whether the proposed strategy has an impact in terms of object hallucination. We perform this analysis by employing the robust COCO splits defined in [180] and re-

[§]https://codalab.lisn.upsaclay.fr/competitions/7404

Table 5.9: Leaderboard of various methods on the online COCO test server.

	BLEU-1		BLEU-2		BLEU-3		BLEU-4		METEOR		ROUGE		CIDEr	
	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40
Up-Down [7]	80.2	95.2	64.1	88.8	49.1	79.4	36.9	68.5	27.6	36.7	57.1	72.4	117.9	120.5
SGAE [295]	81.0	95.3	65.6	89.5	50.7	80.4	38.5	69.7	28.2	37.2	58.6	73.6	123.8	126.5
AoANet [108]	81.0	95.0	65.8	89.6	51.4	81.3	39.4	71.2	29.1	38.5	58.9	74.5	126.9	129.6
$\mathcal{M}^2$ Transformer [64]	81.6	96.0	66.4	90.8	51.8	82.7	39.7	72.8	29.4	39.0	59.2	74.8	129.3	132.1
X-Transformer [206]	81.9	95.7	66.9	90.5	52.4	82.5	40.3	72.4	29.6	39.2	59.5	75.0	131.1	133.5
RSTNet [320]	82.1	96.4	67.0	91.3	52.2	83.0	40.0	73.1	29.6	39.1	59.5	74.6	131.9	134.0
DLCT [185]	82.4	96.6	67.4	91.7	52.8	83.8	40.6	74.0	29.8	39.6	59.8	75.3	133.3	135.4
COS-Net [152]	83.3	96.8	68.6	92.3	54.2	84.5	42.0	74.7	30.4	40.1	60.6	76.4	136.7	138.3
CaMEL [18]	83.2	97.3	68.3	92.7	53.6	84.8	41.2	74.9	30.2	39.7	60.2	75.6	137.5	140.0
PMA-Net	**84.7**	**97.9**	**70.2**	**93.8**	**55.7**	**86.5**	**43.4**	**77.1**	**30.5**	**40.3**	**61.3**	**76.8**	**141.5**	**143.4**

Table 5.10: Results on robust COCO test set. The † marker indicates a model re-trained with the same visual features of our approach.

	B-1	B-4	M	R	C	S	CHs	CHi
Att2In [222]	-	-	24.0	-	85.8	16.9	14.1	10.1
Up-Down [7]	-	-	24.7	-	89.8	17.7	11.3	7.9
Transformer[152]	76.9	36.3	27.4	56.1	109.3	20.5	7.9	5.1
COS-Net [152]	78.0	37.3	27.9	56.8	112.1	21.2	6.2	3.9
Transformer†	77.4	37.8	**29.4**	58.1	119.6	22.3	4.6	2.8
PMA-Net	**79.5**	**39.3**	**29.4**	**58.7**	**122.0**	**22.5**	**4.3**	**2.6**

port the results in Table 5.10, comparing with state-of-the-art approaches and with a Transformer trained with the same visual features. Both PMA-Net and the Transformer baseline are re-trained from scratch on this dataset using cross-entropy loss only. In addition to standard evaluation metrics, we employ the CHAIR score, in its variants CHi and CHs, to measure object hallucination. From this analysis, it can be seen that the addition of prototypes memory vectors reduces the hallucination rate with respect to a Transformer, and that PMA-Net performs favorably with respect to previous approaches also in this case.

Novel object captioning. We also evaluate PMA-Net on the nocaps dataset [3] for novel object captioning. It shall be noted that our approach does not leverage components which are explicitly designed to deal with

Table 5.11: Results on nocaps validation set. The † marker indicates a model retrained with the same visual features of our approach, while ∗ indicates finetuning of the visual backbone.

	In		Out		Overall	
	C	S	C	S	C	S
NBT [3]	62.1	10.1	62.4	8.9	60.2	9.5
Up-Down [3]	80.0	12.0	66.4	9.7	73.1	11.1
Transformer [64]	78.0	11.0	29.7	7.8	54.7	9.8
$\mathcal{M}^2$ Transformer [64]	85.7	12.1	38.9	8.9	64.5	11.1
GRIT* [200]	105.9	13.6	72.6	11.1	90.2	12.8
Transformer†	105.9	13.3	73.9	11.3	90.9	12.6
PMA-Net	**107.5**	**13.7**	**75.9**	**11.4**	**92.6**	**12.8**

the naming of novel objects, still the nocaps dataset provides a relevant test bed to compare PMA-Net with other approaches from the literature. To conduct this analysis, we employ our model and the Transformer-based baseline trained on the standard COCO dataset. Results are reported in Table 5.11, both in the in-domain and out-of-domain splits of the datasets and without employing constrained beam search [6]. We observe that PMA-Net achieves the best performance among all the compared approaches and with respect to the base Transformer which does not employ memory prototypes. In this setting, PMA-Net also outperforms the results of GRIT, which employs a finetuned visual backbone. This highlights that the addition of prototypes memory vectors improves the description of novel objects.

6

Retrieval-Augmented Multimodal LLMs

While image captioning focuses on generating descriptive language grounded in visual content, recent advances have led to the emergence of Multimodal Large Language Models (MLLMs), which extend this paradigm to a much broader set of vision-and-language tasks. By jointly processing multiple modalities—such

This Chapter is related to the publication F. Cocchi *et al.*, "LLaVA-MORE : A Comparative Study of LLMs and Visual Backbones for Enhanced Visual Instruction Tuning", ICCVW 2025 [57] and D. Caffagni *et al.*, "Wiki-LLaVA: Hierarchical Retrieval-Augmented Generation for Multimodal LLMs", CVPRW 2024 [29] and A. Compagnoni *et al.*, "ReAG: Reasoning-Augmented Generation for Knowledge-based Visual Question Answering", under submission 2025 [59]

as text, images, and videos – MLLMs exhibit strong reasoning capabilities and can tackle complex multimodal problems that go beyond simple description. Owing to this flexibility, many multimodal tasks can be naturally framed as instances of Visual Question Answering (VQA), where the model must interpret visual inputs and produce faithful, well-structured responses. However, despite their large-scale pretraining, MLLMs often struggle with underrepresented or domain-specific knowledge. To address these limitations, retrieval augmentation has become a key component in modern MLLMs, enabling accessing external information sources and enhance their reasoning and generalization capabilities in knowledge-intensive settings.

In the following sections, we consider a setting in which the model has access to a large external multimodal corpus, such as collections of Wikipedia pages. The datasets used in our experiments are specifically designed for Visual Question Answering tasks (see Section 2.2.2).

6.1 Architectures and Design Trends

Table 6.1 provides an overview of representative Multimodal Large Language Models, summarizing the architectural design choices adopted to connect visual encoders with large language models. Earlier approaches such as VisualGPT [40] explored one of the first mechanisms for integrating pretrained GPT-style language models with visual backbones and in the table we highlight the wide range of strategies explored next. For instance, Flamingo [4] and PaLI [47] rely on frozen or lightly adapted LLMs combined with cross-attention-based vision-to-language adapters, while more recent models introduce structured intermediary modules to mediate vision–language interaction [40]. Notably, methods such as BLIP-2 [146] and InstructBLIP [66] employ the Q-Former to efficiently extract and align visual representations before passing them to the language model, strik-

Table 6.1: Summary of generalist MLLMs for vision-to-language tasks. For each model, we indicate the LLM used in its best configuration as shown in the original paper (♢: LLM training from scratch; ♦: LLM fine-tuning; ▲: LLM fine-tuning with PEFT techniques; ★: frozen LLM). The ✻ marker indicates variants to the reported vision-to-language adapter, while gray color indicates models not publicly available.

Model	LLM	Visual Encoder	V2L Adapter	VInstr. Tuning	Main Tasks & Capabilities
BLIP-2 [148]	FlanT5-XXL-11B★	EVA ViT-g	Q-Former	✗	Visual Dialogue, VQA, Captioning, Retrieval
FROMAGe [127]	OPT-6.7B★	CLIP ViT-L	Linear	✗	Visual Dialogue, Captioning, Retrieval
Kosmos-1 [110]	Magneto-1.3B♢	CLIP ViT-L	Q-Former✻	✗	Visual Dialogue, VQA, Captioning
LLaMA-Adapter V2 [82]	LLaMA-7B▲	CLIP ViT-L	Linear	✗	VQA, Captioning
OpenFlamingo [11]	MPT-7B★	CLIP ViT-L	XAttn LLM	✗	VQA, Captioning
Flamingo [4]	Chinchilla-70B★	NFNet-F6	XAttn LLM	✗	Visual Dialogue, VQA, Captioning
PaLI [47]	mT5-XXL-13B♦	ViT-e	XAttn LLM	✗	Multilingual, VQA, Captioning, Retrieval
PaLI-X [46]	UL2-32B♦	ViT-22B	XAttn LLM	✗	Multilingual, VQA, Captioning
LLaVA [168]	Vicuna-13B♦	CLIP ViT-L	Linear	✓	Visual Dialogue, VQA, Captioning
MiniGPT-4 [326]	Vicuna-13B★	EVA ViT-g	Linear	✓	VQA, Captioning
mPLUG-Owl [301]	LLaMA-7B▲	CLIP ViT-L	Q-Former✻	✓	Visual Dialogue, VQA
InstructBLIP [66]	Vicuna-13B★	EVA ViT-g	Q-Former	✓	Visual Dialogue, VQA, Captioning
MultiModal-GPT [86]	LLaMA-7B▲	CLIP ViT-L	XAttn LLM	✓	Visual Dialogue, VQA, Captioning
LaVIN [184]	LLaMA-13B▲	CLIP ViT-L	MLP	✓	Visual Dialogue, VQA, Captioning
Otter [144]	LLaMA-7B★	CLIP ViT-L	XAttn LLM	✓	VQA, Captioning
Kosmos-2 [208]	Magneto-1.3B♢	CLIP ViT-L	Q-Former✻	✓	Visual Dialogue, VQA, Captioning, Referring, REC
Shikra [42]	Vicuna-13B♦	CLIP ViT-L	Linear	✓	Visual Dialogue, VQA, Captioning, Referring, REC, GroundCap
Clever Flamingo [37]	LLaMA-7B▲	CLIP ViT-L	XAttn LLM	✓	Visual Dialogue, VQA, Captioning
SVIT [321]	Vicuna-13B♦	CLIP ViT-L	MLP	✓	Visual Dialogue, VQA, Captioning
BLIVA [105]	Vicuna-7B★	EVA ViT-g	Q-Former+Linear	✓	Visual Dialogue, VQA, Captioning
IDEFICS [133]	LLaMA-65B★	OpenCLIP ViT-H	XAttn LLM	✓	Visual Dialogue, VQA, Captioning
Qwen-VL [12]	Qwen-7B♦	OpenCLIP ViT-bigG	Q-Former✻	✓	Visual Dialogue, Multilingual, VQA, Captioning, REC
StableLLaVA [153]	Vicuna-13B♦	CLIP ViT-L	Linear	✓	Visual Dialogue, VQA, Captioning
Ferret [304]	Vicuna-13B♦	CLIP ViT-L	Linear	✓	Visual Dialogue, Captioning, Referring, REC, GroundCap
LLaVA-1.5 [166]	Vicuna-13B♦	CLIP ViT-L	MLP	✓	Visual Dialogue, VQA, Captioning
MiniGPT-v2 [41]	LLaMA-2-7B▲	EVA ViT-g	Linear	✓	Visual Dialogue, VQA, Captioning, Referring, REC, GroundCap
Pink [290]	Vicuna-7B▲	CLIP ViT-L	Linear	✓	Visual Dialogue, VQA, Captioning, Referring, REC, GroundCap
CogVLM [277]	Vicuna-7B♦	EVA ViT-E	MLP	✓	Visual Dialogue, VQA, Captioning, REC
DRESS [49]	Vicuna-13B▲	EVA ViT-g	Linear	✓	Visual Dialogue, VQA, Captioning
LION [38]	FlanT5-XXL-11B★	EVA ViT-g	Q-Former+MLP	✓	Visual Dialogue, VQA, Captioning, REC
mPLUG-Owl2 [302]	LLaMA-2-7B♦	CLIP ViT-L	Q-Former✻	✓	Visual Dialogue, VQA, Captioning
SPHINX [161]	LLaMA-2-13B♦	Mixture	Linear	✓	Visual Dialogue, VQA, Captioning, Referring, REC, GroundCap
Honeybee [34]	Vicuna-13B♦	CLIP ViT-L	ResNet blocks	✓	Visual Dialogue, VQA, Captioning
VILA [156]	LLaMA-2-13B♦	CLIP ViT-L	Linear	✓	Visual Dialogue, VQA, Captioning
SPHINX-X [83]	Mixtral-8×7B♦	Mixture	Linear	✓	Visual Dialogue, Multilingual, VQA, Captioning, Referring, REC

ing a balance between expressiveness and computational efficiency. This architectural diversity reflects different trade-offs between model capacity, training cost, and flexibility. While these models differ substantially in their architectural choices, they share the common goal of enabling flexible multimodal interaction within a unified framework. The architectural diversity illustrated in Table 6.1 provides important context for understanding the performance trends observed across benchmarks.

Table 6.2 reports quantitative results for a subset of models across benchmarks for Visual Question Answering, image captioning, and broader MLLM evaluation protocols. The results show a steady improvement in multimodal reasoning over time, with newer models achieving stronger and more balanced performance. Importantly, the comparison highlights how

Table 6.2: Performance analysis on 14 evaluation benchmarks for VQA, image captioning, and MLLM evaluation. Best scores are in bold, second best are underlined.

	VQA					Captioning		MLLM Evaluation						
Model	VQA^{v2}	GQA	VizWiz	SQA	VQA^{T}	COCO	Flickr	POPE	MME	MMB	SEED	$LLaVA^{W}$	MM-Vet	$Math^{V}$
Flamingo [4]	82.0	-	**65.7**	-	57.1	138.1	75.4	-	-	-	-	-	-	-
BLIP-2 [146]	65.0	41.0	19.6	61.0	42.5	144.5	-	85.3	1293.8	-	46.4	38.1	22.4	-
OpenFlamingo [11]	52.7	-	27.5	-	24.2	75.9	59.5	-	-	-	-	-	-	-
MiniGPT-4 [326]	53.7	32.2	-	-	-	-	-	-	581.7	23.0	42.8	45.1	22.1	23.1
mPLUG-Owl [301]	59.5	40.9	-	-	-	-	-	-	967.3	46.6	34.0	-	-	-
ChatBridge [322]	-	41.8	-	-	-	-	82.5	-	-	-	-	-	-	-
InstructBLIP [66]	69.4	49.5	33.4	63.1	50.7	102.2	82.8	78.9	1212.8	36.0	53.4	58.2	25.6	25.3
Shikra [42]	77.4	-	-	-	-	117.5	-	-	-	58.8	-	**79.9**	-	-
Emu [256]	62.0	46.0	38.3	-	-	117.7	-	-	-	-	-	-	36.3	-
SVIT [321]	80.3	64.1	56.4	70.0	60.8	-	-	-	1565.8	69.1	61.9	-	-	-
BLIVA [105]	-	-	42.9	-	58.0	-	87.1	-	**1669.2**	-	-	-	-	-
IDEFICS [133]	60.0	45.2	36.0	-	30.9	-	-	-	-	54.5	-	-	-	
Qwen-VL [12]	78.2	57.5	38.9	68.2	61.5	120.2	81.0	-	1487.6	60.6	58.2	56.7	-	-
DreamLLM [69]	56.6	-	38.1	-	34.9	115.4	-	-	-	49.9	-	-	35.9	-
LLaVA-1.5 [166]	80.0	63.3	53.6	71.6	61.3	-	-	85.9	1531.3	67.7	61.6	70.7	35.4	23.6
CogVLM [277]	82.3	-	-	-	-	**148.7**	**94.9**	87.9	-	**77.6**	72.5	77.8	**51.1**	34.5
LION [38]	-	51.6	-	-	-	139.3	87.1	88.9	-	-	-	-	-	-
mPLUG-Owl2 [302]	79.4	56.1	54.5	-	-	137.3	-	86.2	1450.2	64.5	57.8	25.0	36.2	25.3
SPHINX [161]	80.2	62.9	46.8	69.1	-	-	-	**90.8**	1560.2	67.1	71.6	74.3	36.6	27.5
Emu2 [253]	**84.9**	**65.1**	54.9	-	**66.6**	-	-	-	-	-	62.8	-	48.5	-
Honeybee [34]	-	-	-	-	-	-	-	-	1632.0	73.6	68.6	77.5	-	-
Unified-IO 2 [179]	79.4	-	-	**88.7**	-	125.4	-	87.7	-	71.5	61.8	-	-	-
VILA [156]	80.8	63.3	60.6	73.7	**66.6**	115.7	74.2	84.2	1570.1	70.3	62.8	73.0	38.8	
SPHINX-X [83]	81.1	63.8	61.9	74.5	-	-	-	89.6	1485.3	71.3	**73.0**	70.2	40.9	**42.7**

architectural decisions—such as the choice of vision–language adapter, the extent of language-model fine-tuning, and model scale—directly impact performance across different evaluation settings.

6.1.1 A Comparative Study of LLMs and Visual Backbones

Among multimodal large language models, while current systems demonstrate impressive performance, the field has largely converged on a relatively narrow set of technical design choices. Most approaches rely on LLaMA-derived language models and LLaVA-style training protocols, together with visual encoders trained via contrastive learning objectives. In particular, CLIP [215] and its derivatives [269, 73, 314] have become the default choice for visual feature extraction, as they are specifically designed to produce embeddings that integrate seamlessly with language models.

Although contrastive learning has proven highly effective for aligning

images and text within a shared representation space, it is not the only paradigm capable of learning strong visual features. Recent vision models trained in a purely self-supervised manner, without relying on textual supervision, have demonstrated robust and transferable representations along with intriguing emergent properties [33, 204]. Despite these advances, the application of such visual encoders to multimodal large language models remains relatively understudied.

To address this gap, we conduct a comprehensive empirical study that systematically pairs diverse language models – ranging from efficient architectures [1] to significantly larger models [266, 260] – with a variety of visual backbones [215, 314, 204, 269]. By exploring different architectural combinations, we aim to uncover the strengths and limitations of alternative vision–language integration strategies, shedding light on overlooked design choices and their impact on multimodal learning.

6.1.2 LLaVA-MORE: A New Family of Multimodal LLM

We propose **LLaVA-MORE**, a new family of models that extend the standard LLaVA architecture by combining the visual encoder with various LLMs, ranging from small- to medium-scale models. As small-scale models, we utilize Gemma-2 2B [260] and Phi-4-Mini [1] (with 3.8B parameters), both designed for strong reasoning scalability, effectively challenging larger models. For medium-scale models, we select the variant of Gemma-2 [260] with 9B parameters alongside two recent architectures: the original LLaMA-3.1 [88] LLM with 8B parameters and its distilled DeepSeek-R1 version [89] (*i.e.*, DeepSeek-R1-Distill-LLaMA-8B). For each LLM category, we assess the impact of varying the visual backbone, with the aim to identify the optimal configuration. Specifically, our study compares the standard CLIP ViT-L/14 encoder employed in the LLaVA-1.5 model [167] with two variants of DINOv2 differentiated by the presence or the absence of visual register to-

Table 6.3: Performance analysis when changing the underlying LLMs. Results are reported considering both small- and medium-scale LLMs, comparing LLaVA-MORE with existing LLaVA-based variants. All models employ the CLIP ViT-L/14@336 as the visual backbone.

	VQA Benchmarks				MLLM Benchmarks								
	GQA	Science-QA	TextVQA	AI2D	POPE	MME-P	MME-C	MMB-Cn	MMB-En	SEED-All	SEED-V	SEED-I	MMMU
Small-Scale MLLMs													
LLaVA-Phi-2.7B	-	68.4	-	-	85.0	1335.1	-	-	59.8	-	-	-	-
LLaVA-MORE (Ours)													
Gemma-2-2B	**62.4**	71.1	**54.4**	57.1	**86.0**	**1401.1**	**337.8**	**65.8**	53.3	62.2	41.9	67.6	33.4
Phi-4-3.8B	62.1	**71.3**	54.0	**61.1**	85.9	1372.2	281.1	64.2	**69.2**	**63.5**	**42.3**	**69.1**	**38.8**
Medium-Scale MLLMs													
LLaVA-1.5-7B	62.4	69.0	58.2	56.4	85.6	1474.3	314.6	56.5	65.3	61.6	42.0	66.8	34.2
LLaVA-1.5-LLaMA3-8B	63.5	74.2	57.6	60.7	85.4	**1544.4**	330.3	65.4	70.3	64.3	42.0	**70.1**	37.3
LLaVA-MORE (Ours)													
LLaMA-3.1-8B	63.6	**76.3**	58.4	61.8	85.1	1531.5	**353.3**	**68.2**	**72.4**	64.1	42.4	69.8	**39.4**
DeepSeek-R1-Distill-LLaMA-8B	63.0	74.5	56.3	58.8	85.1	1495.1	295.0	66.8	61.3	63.5	43.5	68.6	38.1
Gemma-2-9B	**64.2**	75.4	**60.7**	**64.8**	**86.8**	1522.5	307.5	65.9	71.9	**64.5**	**44.1**	**69.9**	37.9

Table 6.4: Performance analysis with varying visual backbones. Results are reported for the best small- and medium-scale LLaVA-MORE configurations using Phi-4-3.8B and Gemma-2-9B, respectively. Input resolution and the number of visual tokens are also included.

			VQA Benchmarks				MLLM Benchmarks								
	Resolution	# Tokens	GQA	Science-QA	TextVQA	AI2D	POPE	MME-P	MME-C	MMB-Cn	MMB-En	SEED-All	SEED-V	SEED-I	MMMU
LLaVA-MORE-3.8B (Ours)															
CLIP ViT-L/14 [215]	336^2	576	62.1	71.3	54.0	61.1	85.9	1372.2	281.1	64.2	69.2	63.5	42.3	69.1	38.8
DINOv2 ViT-L/14 [204]	224^2	256	60.9	66.6	41.4	58.2	85.5	1236.6	281.1	53.8	58.9	59.8	40.6	64.8	37.9
$DINOv2_{reg}$ ViT-L/14 [67]	224^2	256	60.4	69.0	41.3	56.4	85.2	1263.2	**288.2**	57.4	51.4	58.7	41.4	63.2	38.6
SigLIP ViT-L/14 [314]	384^2	729	**63.6**	**73.8**	57.6	**62.9**	86.4	1379.0	282.9	66.5	**71.4**	65.7	46.4	70.8	**40.0**
SigLIP2 ViT-L/14 [269]	384^2	729	63.4	71.8	**59.7**	**62.9**	**86.5**	**1406.7**	282.5	**66.8**	69.8	**66.4**	**47.4**	**71.4**	38.8
LLaVA-MORE-9B (Ours)															
CLIP ViT-L/14 [215]	336^2	576	64.2	75.4	60.7	64.8	**86.8**	**1522.5**	307.5	65.9	71.9	64.5	44.1	69.9	37.9
DINOv2 ViT-L/14 [204]	224^2	256	63.1	71.5	48.1	61.3	85.3	1394.4	**334.3**	56.4	63.8	61.0	40.6	66.4	38.7
$DINOv2_{reg}$ ViT-L/14 [67]	224^2	256	62.8	69.1	47.9	59.1	84.0	1413.9	295.4	60.1	53.8	60.1	42.3	64.7	38.3
SigLIP ViT-L/14 [314]	384^2	729	64.8	**76.3**	63.9	64.7	86.1	1487.9	299.3	**69.1**	**74.4**	66.6	**46.6**	71.9	**39.7**
SigLIP2 ViT-L/14 [269]	384^2	729	**65.6**	76.2	**66.7**	**65.3**	86.1	1510.9	308.2	68.0	72.0	**67.5**	46.0	**73.1**	38.7

kens [204, 67], known for their strong, semantically rich visual features, as well as SigLIP [314] and its more advanced successor, SigLIP2 [269].

To train our models, we follow the two-stage training paradigm commonly used in the literature. However, our approach stands out by applying a consistent training and evaluation strategy across all models, ensuring fairness in comparisons. In the first stage, only the vision-to-language adapter is optimized to align the image features with the text embedding space. In the second stage, visual instruction-following training is conducted to enhance multimodal conversational capabilities. During this phase, the parameters of both the multimodal adapter and the LLM are updated.

Table 6.5: Performance analysis when changing the training dataset during the first pre-training stage. Results are reported for the best small- and medium-scale LLaVA-MORE configurations using Phi-4-3.8B and Gemma-2-9B.

	VQA Benchmarks				MLLM Benchmarks								
	GQA	Science-QA	TextVQA	AI2D	POPE	MME-P	MME-C	MMB-Cn	MMB-En	SEED-All	SEED-V	SEED-I	MMMU
LLaVA-MORE-3.8B (Ours)													
LLaVA Pre-Train LCS (558k)	63.4	71.8	59.7	62.9	86.5	1406.7	282.5	66.8	69.8	66.4	47.4	71.4	38.8
LAION (558k)	64.3	**72.5**	**62.3**	**65.2**	**86.8**	**1453.2**	287.1	**67.2**	**72.3**	66.6	46.4	71.9	**39.7**
Recap (558k)	**64.6**	71.7	61.4	64.5	86.5	1428.7	**297.9**	67.1	71.6	67.3	**47.7**	72.5	39.0
LAION+Recap (558k)	**64.6**	71.8	61.3	63.9	86.6	1425.8	297.5	65.8	71.7	**67.6**	47.5	**72.9**	39.2
LLaVA-MORE-9B (Ours)													
LLaVA Pre-Train LCS (558k)	65.6	76.2	66.7	**65.3**	86.1	1510.9	308.2	68.0	72.0	67.5	46.0	73.1	38.7
LAION (558k)	65.6	76.0	67.0	65.1	**86.9**	**1579.8**	**350.7**	68.5	73.9	67.4	45.4	**73.2**	41.1
Recap (558k)	**65.9**	76.2	67.2	64.3	**86.9**	1540.8	318.2	**69.4**	**74.7**	**67.6**	**47.2**	73.0	40.2
LAION+Recap (558k)	65.8	**77.1**	**67.8**	65.2	86.7	1537.8	335.4	65.9	73.5	67.4	45.2	73.1	**41.2**

6.1.3 Design Principles for Multimodal LLMs

In our analysis, we systematically vary both the underlying language models (Table 6.3), the visual backbones (Table 6.4), and the pre-training datasets (Table 6.5). Beyond architectural choices, we also study how training data and input resolution affect performance. Taken together, these experiments provide several key principles to consider when designing effective multimodal large language models.

Model Scaling. Carefully designed small language models (e.g., Phi-4-3.8B) can match or even outperform medium-scale MLLMs from previous generations, demonstrating that efficient architectures and high-quality training strategies can rival brute-force scaling.

Visual Backbone Choice. Visual encoders trained with image–text contrastive learning, such as CLIP and SigLIP, outperform purely self-supervised models like DINOv2, due to their stronger alignment with language representations. Among contrastive approaches, SigLIP and SigLIP2 emerge as the most effective visual backbones across benchmarks, benefiting from large-scale image–text pre-training and improved training objectives.

Visual Resolution. Increasing image resolution and the number of visual tokens significantly benefits small-scale MLLMs, enabling finer-grained visual understanding. However, these gains become task-dependent and tend

to diminish as model size increases, suggesting different optimal regimes for small and medium-scale architectures.

Training Data. At smaller scales, model performance is highly sensitive to the source and composition of pre-training data, whereas medium-scale models exhibit greater robustness to dataset variations. These observations underscore the central role of data selection when operating under constrained model capacity, while also suggesting that scaling the model can partially mitigate sensitivity to dataset-specific biases.

Overall, this analysis shows that modern MLLMs have evolved into versatile multimodal systems capable of addressing a wide range of vision-and-language tasks within a unified architecture. Importantly, no single combination of language model, visual backbone, resolution, and data consistently dominates across all benchmarks, underscoring the need for task-aware and application-specific design choices.

These insights naturally motivate the integration of retrieval augmentation, which further enhances performance and robustness in knowledge-intensive and domain-specific scenarios.

6.2 Hierarchical Retrieval-Augmented Generation for MLLMs

Our goal is to equip Multimodal LLMs (MLLMs) with the ability to answer complex and specific questions that cannot be addressed solely through the image content and pre-trained knowledge. To achieve this, we propose **Wiki-LLaVA**, which integrates external knowledge derived from an external memory into the LLaVA model, without significantly altering its design. Instead, we augment the capabilities of the model by incorporating retrieval information as additional input context.

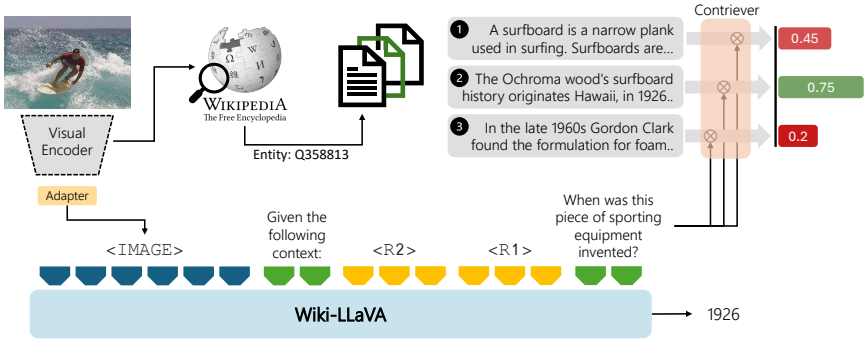


Figure 6.1: Overview of the architecture of WikiLLaVA, which augments a multimodal LLM with external knowledge through a hierarchical retrieval pipeline.

Overview. Overall, WikiLLaVA comprises three components, as shown in Fig. 6.1: a visual encoder, which is employed to provide the MLLM with visual context and as a query to retrieve from an external knowledge base, the knowledge base itself (*e.g.*, Wikipedia), and a hierarchical retrieval module which retrieves relevant documents and passages from the external knowledge base, to be employed as additional context for the MLLM.

An MLLM usually takes as input a multimodal input query, comprising both image and text, and generates a textual output in an autoregressive manner. Formally, the architecture is trained to model a probability distribution $p(w_t|I, w_0, w_1, ..., w_{t-1}, \theta)$, where θ denotes the parameters of the model, I represents an input image, and $w_0, .., w_{t-1}$ denotes the textual prompt. The textual prompt usually includes a pre-defined system-level prompt and a question related to the input image, given by the user. Clearly, a standard MLLM can only rely on the user prompt, the input image, and the knowledge stored in its internal parameters (*i.e.*, θ) to accommodate requests, thus limiting its ability to answer questions that rely on external knowledge.

In the following, we employ LLaVA [168] as our reference MLLM. LLaVA exploits the capabilities of a pre-trained LLM (*i.e.*, Vicuna [54]) and a pre-

trained visual model (*i.e.*, a CLIP-based visual encoder [215]), which are interconnected through an MLP adapter, in charge of converting CLIP features to dense input tokens. For an input image I, therefore, LLaVA utilizes a pre-trained CLIP visual encoder E_v, extracts a dense grid of visual features $Z_v = E_v(I)$, which is then projected via a learnable MLP to produce a sequence of dense embedding tokens $v_o, v_1, ..., v_N$. Finally, these are prepended to the system prompt, and the full sequence of visual and textual tokens is then given as input to the LLM component of the model.

6.2.1 Hierarchical Knowledge Retrieval

Augmentation with external knowledge. To augment the MLLM with external knowledge, we enrich the input context by injecting relevant textual data from an external memory composed of documents. Formally, the distribution of the MLLM is conditioned on additional textual retrieval-knowledge tokens, leading to

$$p(w_t | \overbrace{v_o, v_1, ..., v_N}^{\text{Visual tokens}}, \underbrace{w_0, w_1, ..., w_{t-1}}_{\text{System + user prompt}}, \overbrace{e_0, e_1, ..., e_\tau}^{\text{External memory tokens}}), \tag{6.1}$$

where $e_0, ..., e_\tau$ represents the added tokens retrieved from the external memory. Differently from the standard formulation of MLLMs, by enriching the input context we allow the model to generate more specific answers by exploiting tokens retrieved from the memory.

Hierarchical retrieval from an external memory. The external memory comprises a collection of (document, image, text-title) triplets taken from documents, denoted as $\mathcal{D} = \{(d_i, t_i)_i\}$. Within this memory, we conduct a hierarchical two-step search to retrieve appropriate information. Initially, we locate the most pertinent document, followed by identifying the relevant passage inside a particular document, which is subsequently exploited as additional input context in the MLLM.

In the first stage, given an input query image I we perform an approximate k-nearest neighbor search into the external memory, using document titles as retrievable keys. The similarity between the query image and the text titles is modeled as the inner product between their respective embeddings, which are computed through the visual and textual CLIP encoders (*i.e.*, E_v and E_t), as follows:

$$\text{sim}(I_i, t_i) = E_v(I) \cdot E_t(t_i)^T. \tag{6.2}$$

Then, the knowledge retriever returns the top-k documents associated with the most relevant items retrieved using the aforementioned procedure.

Retrieving document passages. Then we analyze the retrieved documents to identify the most relevant passages corresponding to the user's question. Each document is defined as a sequence of chunks, denoted as $d_i = [c_{i_0}, .., c_{i_T}]$, and, given the input question, we retrieve the chunks with the highest similarity to the question. We employ the Contriever [112] to embed each chunk of the selected document, along with the query (*i.e.*, the question provided by the user), and compute the similarity as an inner product between embeddings. By retrieving the n most appropriate passages inside each of the retrieved documents, we obtain $k \cdot n$ passages.

Context enrichment. Once we find the most relevant chunks, we employ their raw contents as an additional input to the MLLM. Specifically, the final prompt that we employ includes the image tokens, the retrieved raw chunks, the system-level prompt, and the user question. Formally, considering three retrieved passages, the final prompt is defined as follows:

```
<IMAGE>\nGiven the following context:\n
<R1>\n<R2>\n<R3>\n <QUESTION>
Give a short answer. ASSISTANT:
```
(6.3)

Training Protocol. While the aforementioned approach could work in a zero-shot fashion, using the original weights θ of the pre-trained MLLM, we also investigate the case of fine-tuning the model to augment its capabilities of exploiting retrieved passages. In particular, the model is trained on pairs of questions and ground-truth answers requiring external knowledge. As this would potentially reduce the capabilities of the MLLM on tasks not requiring external knowledge (*i.e.*, all the other tasks on which the model has been originally trained), we apply a data mixing approach in which ground-truth pairs requiring external knowledge are mixed with ground-truth pairs not requiring external knowledge in the same mini-batch.

6.2.2 Showing the Effectiveness of Retrieval-augmentation

LLaVA fine-tuning and Retrieval. We employ two distinct fine-tuning approaches,each tailored and exclusively applied to one of the considered datasets, while carefully preserving the general-purpose capabilities of the underlying LLaVA model.

Specifically, we supplement fine-tuning data with samples from the LLaVA-Instruct dataset [168]. Given its size of 158k, we double the probability of having examples from this dataset in each mini-batch. To reduce the number of trainable parameters, we train using low-rank adapters [103] with a total batch size of 512 samples.

Textual documents sourced from Wikipedia content are embedded using the Contriever architecture [112], segmenting the text into chunks of 600 characters each. Furthermore, for streamlined efficiency, the process involves utilizing a single visual encoder. Specifically, following the LLaVA architecture [168], we employ the CLIP ViT-L/14@336 backbone to embed images to give as input to the MLLM, while simultaneously leveraging it to

Table 6.6: Entity retrieval results on the Encyclopedic-VQA test set and InfoSeek validation set. To comply with the visual encoder employed in LLaVA, all results are obtained using CLIP ViT-L/14@336.

Dataset	KB	R@1	R@10	R@20	R@50
Encyclopedic-VQA	2M	3.3	9.9	13.2	17.5
InfoSeek	100k	36.9	66.1	71.9	78.4

extract query visual features in the initial hierarchical retrieval step, facilitating the integration of an external memory component.

To perform entity retrieval, we employ approximate *k*NN search rather than exact *k*NN search because it significantly improves the computational speed of the entire pipeline. To this aim, we employ the Faiss library [115] and a graph-based HNSW index with 32 links per vertex.

Analyzing CLIP performance. We start by evaluating entity retrieval results using CLIP. In this setting, we consider images from the Encyclopedic-VQA test set and InfoSeek validation set and measure the CLIP ability to find the correct entity within the knowledge base of each respective dataset (*i.e.*, composed of 2M entries for Encyclopedic-VQA and 100k entries for InfoSeek). As previously mentioned, we perform retrieval using images as queries and Wikipedia titles as retrievable items.

Results are reported in Table 6.6 in terms of recall@k (R@k) with $k = 1, 10, 20, 50$ which measures the percentage of times the correct entity is found in the top-k retrieved elements. Notably, correctly retrieving the Wikipedia entity associated with the input image strongly depends on the size of the employed knowledge base. In fact, when using 100k items, as in the case of InfoSeek, the correct entity is retrieved as the first item 36.9% of the time and among the top-10 66.1% of the time. Instead, when using a significantly larger knowledge base as in the case of Encyclopedic-VQA, which contains 2M items, retrieval results are significantly lower with 3.3% and 9.9% respectively in terms of R@1 and R@10.

Table 6.7: Accuracy results on the Encyclopedic-VQA test set and InfoSeek validation set. **Yellow color** indicates models employing the CLIP model to perform entity retrieval, while **gray color** indicates the use of ground-truth entities (*i.e.*, oracle). k denotes the number of retrieved entities, and n represents the number of textual chunks retrieved for each entity that are given to the MLLM as additional context.

					Enc-VQA		InfoSeek		
Model	**LLM**	**KB**	k	n	Single-Hop	All	Unseen-Q	Unseen-E	All
Zero-shot Models									
BLIP-2 [146]	Flan-T5$_{XL}$	✗	-	-	12.6	12.4	12.7	12.3	12.5
InstructBLIP [66]	Flan-T5$_{XL}$	✗	-	-	11.9	12.0	8.9	7.4	8.1
LLaVA-1.5 [166]	Vicuna-7B	✗	-	-	16.3	16.9	9.6	9.4	9.5
Fine-tuned Models									
LLaVA-1.5 [166]	Vicuna-7B	✗	-	-	23.3	28.5	19.4	16.7	17.9
WikiLLaVA	Vicuna-7B	✓	1	1	21.8	26.4	26.6	24.6	25.5
WikiLLaVA	Vicuna-7B	✓	1	2	19.9	23.2	29.1	26.3	27.6
WikiLLaVA	Vicuna-7B	✓	1	3	17.7	20.3	30.1	27.8	28.9
WikiLLaVA	Vicuna-7B	✓	2	1	21.3	25.4	27.8	24.6	26.1
WikiLLaVA	Vicuna-7B	✓	3	1	20.5	24.3	27.4	24.5	25.3
WikiLLaVA	Vicuna-7B	✓	1	1	34.7	37.2	41.1	41.1	41.1
WikiLLaVA	Vicuna-7B	✓	1	2	39.2	40.2	49.1	46.5	47.8
WikiLLaVA	Vicuna-7B	✓	1	3	38.5	38.6	52.7	50.3	51.5

Results on Encyclopedic-VQA and InfoSeek. We then report visual question-answering results in Table 6.7. We include the performance of zero-shot models like BLIP-2 [146], InstructBLIP [66], and the LLaVA-1.5 baseline model [168], which are not fine-tuned on the considered datasets and that do not leverage the external knowledge base. Moreover, we consider the accuracy results of LLaVA-1.5 when fine-tuned on the training set of Encyclopedic-VQA and InfoSeek, but not augmented with retrieved context. The results of our approach (*i.e.*, WikiLLaVA) are reported both in the standard setting in which CLIP is used to retrieve the most representative entity from the knowledge base and in its *oracle* version, which employs the entity corresponding to the input image-question pair. For both cases, we consider a different number n of retrieved textual chunks, all corresponding to the top-1 (or ground-truth) entity. When employing CLIP, we also vary the number k of retrieved entities (*i.e.*, $k = 1, 2, 3$) using $n = 1$ when k is greater than 1. This choice is given by the maximum context length that

Vicuna takes as input, which is set to 2,048 tokens.

As it can be seen, zero-shot MLLMs face difficulties in correctly answering the given questions as these models can only rely on the knowledge embedded inside the LLM. When instead using an external knowledge base, the accuracy results significantly increase especially on the InfoSeek dataset with 100k retrievable items. The limited performance of the CLIP model in retrieving the correct entity on larger knowledge bases, instead, leads to a slight degradation of accuracy scores. This is due to the noisy textual passages that are provided to the MLLM as additional external context which, being related to a different entity, often do not contain informative content.

Overall, retrieving passages from different entities does not always help increase the results. Instead, using more than one textual chunk as additional context for the MLLM generally improves the final accuracy on the InfoSeek validation set with an overall improvement of 2.1 and 3.4 accuracy points with $n = 2$ and $n = 3$ respectively. Furthermore, it is worth noting that employing oracle entities significantly boosts the final accuracy. In particular, oracle entities lead to an improvement of 13.8% on Encyclopedic-VQA and 22.6% on InfoSeek, comparing the best-performing configuration with CLIP-based entity retrieval (*i.e.*, $k = 1$ and $n = 1$ for Encyclopedic-VQA and $k = 1$ and $n = 3$ for InfoSeek) with the best performing oracle-based version (*i.e.*, $k = 1$ and $n = 2$ for Encyclopedic-VQA and $k = 1$ and $n = 3$ for InfoSeek). These results confirm the effectiveness of directly employing retrieved passages to augment a pre-trained MLLM and further highlight the importance of having a good entity retrieval model to limit the possibility of feeding the MLLM with irrelevant content.

Qualitative Results. Some qualitative results on sample image-question pairs from Encyclopedic-VQA (first row) and InfoSeek (second row) are reported in Fig. 6.2, comparing the answers given by Wiki-LLaVA with those coming from the original LLaVA-1.5 model. For completeness, we also re-

In what state is this building located?

LLaVA-1.5:
California ✗
Wiki-LLaVA:
Washington ✓

When was this building constructed?

LLaVA-1.5:
1970 ✗
Wiki-LLaVA:
1927 ✓

Which geographic area is this fish found?

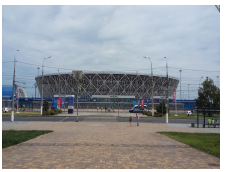
LLaVA-1.5:
Gulf of Mexico ✗
Wiki-LLaVA:
Brazil ✓

What is the oldest age of this animal?

LLaVA-1.5:
10 years ✗
Wiki-LLaVA:
24.9 ✓

Which culture is associated with this place?
`Ancient Greek`

LLaVA-1.5:
Roman ✗
Wiki-LLaVA:
Nuragic Civilization ✗



What is the name of the main club of this stadium?
`FC Rotor`

LLaVA-1.5:
Real Madrid ✗
Wiki-LLaVA:
FC Dynamo Kyiv ✗

Figure 6.2: Qualitative results on sample image-question pairs from Encyclopedic-VQA (first row) and InfoSeek (second row) comparing the proposed approach with the original LLaVA-1.5 model. Some failure cases are shown in the third row with the corresponding ground-truth.

port some failure cases (third row) in which both models are not able to correctly answer the given question.

Preservation of LLaVA performance. Finally, we analyze the impact of LLaVA fine-tuning on knowledge-based VQA datasets when evaluating the model on common MLLM evaluation benchmarks [28]. In particular, we include results on MME [75] which contains image-question pairs covering 14 different tasks grouped in two macro-categories (*i.e.*, cognition and perception), MMMU [311] that is composed of multiple-choice and open-ended questions possibly interleaved with one or more images and extracted from diverse university textbooks and online courses, MMBench (MMB) [173] that includes multiple-choice questions across 20 domains, and POPE [151] that is focused on evaluating object hallucinations and comprises binary classification entries, each related to an image. Details about the evaluation and number of samples are in the original paper of each dataset.

Table 6.8: Performance preservation analysis with respect to the original LLaVA-1.5 model (first row) on diverse benchmarks for MLLM evaluation.

	MME		MMMU	MMB	POPE	
Fine-tuning	Cogn	Perc	Acc	Acc	Acc	F1
-	355.7	1513.3	35.1	71.6	86.9	85.8
Enc-VQA	200.7	802.8	36.6	67.7	72.9	63.4
Enc-VQA + LLaVA-Instruct	290.0	1170.1	36.6	70.4	87.2	86.6
InfoSeek	296.8	1377.2	35.2	71.7	82.0	79.6
InfoSeek + LLaVA-Instruct	341.3	1438.9	35.6	71.1	85.8	84.2

Results are shown in Table 6.8 comparing the original LLaVA model with the two fine-tuned versions on Encyclopedic-VQA and InfoSeek, with and without the use of visual instruction tuning data. Overall, employing samples from the LLaVA-Instruct dataset can better preserve the results of the original model, only partially degrading the performance on the considered benchmarks compared to the original model. While the most significant deterioration is achieved on the MME dataset, in the other settings the original performances are better preserved, also leading to an improvement on MMMU and POPE benchmarks compared to the LLaVA-1.5 results.

Limitations and Future Works. While WikiLLaVA represents an important step toward MLLMs that can effectively exploit external multimodal knowledge, several substantial research challenges remain in two key directions. The first challenge concerns the definition of robust embedding spaces that enable accurate retrieval of relevant documents based on both questions and input images—an essential component for improving the higher level of our hierarchical retrieval pipeline. The second challenge involves developing an efficient and scalable paradigm for selecting and aggregating the most useful information from one or more retrieved documents, especially in large knowledge bases where retrieved content is often noisy, redundant, or partially relevant, and where naive context concatenation can hinder rather than help the reasoning capabilities of the model.

6.3 The Impact of Retrieval Quality on Performance

As shown in Table 6.6, the CLIP embedding space exhibits several limitations, particularly in handling long textual inputs, which can be attributed to CLIP's architectural design. Motivated by this observation, we investigate whether adopting stronger retrieval techniques can mitigate these limitations and lead to improved performance in knowledge-intensive VQA.

In Table 6.9, we compare our proposal different multimodal retrievers, such as our ReT and ReT-2 and UniIR, and PreFLMR, and also include the results of the original CLIP and SigLIP2 models as baselines. We also include state-of-the-art methods as reference, namely RORA-VLM [212], EchoSight [293], CoMEM [286], mR^{2}AG-7B [319], ReflectiVA [58], and our Wiki-LLaVA [29]. These methods are specifically designed for knowledge-intensive VQA, involving fine-tuning of the MLLM and, in several cases, a two-stage retrieval process, where the first stage identifies multimodal candidate documents, while the second refines the selection by extracting the most relevant textual passages.

In the bottom part of the table, we rely on off-the-shelf MLLMs, thus isolating the role of retrieval in the downstream performance, and apply text–image-to-text–image retrieval directly, allowing us to also manage documents where the visual component is missing.

First of all, we observe that in general, both LLaVA-MORE and Qwen2.5-VL benefit from retrieval-augmented generation, demonstrating the challenge posed by Encyclopedic-VQA and InfoSeek. When LLaVA-MORE is used as the generator, ReT-2 stands out as the best multimodal retriever across both benchmarks, even outscoring PreFLMR on Encyclopedic-VQA, despite PreFLMR having undergone a dedicated training stage on that dataset. This suggests that at a large scale, the fine-grained late-interaction mecha-

Table 6.9: VQA accuracy scores on the Encyclopedic-VQA test set and the InfoSeek validation set.

		E-VQA		InfoSeek		
Model	**Retrieval Model**	Single-Hop	All	Un-Q	Un-E	All
Task-Specific Architectures						
RORA-VLM-7B [212]	CLIP ViT-L+GSearch	-	20.3	25.1	27.3	-
Wiki-LLaVA-7B [29]	CLIP ViT-L+Contr.	17.7	20.3	30.1	27.8	28.9
EchoSight-8B [293]	EVA-CLIP-8B	26.4	24.9	30.0	30.7	30.4
CoMEM-7B [286]	Custom VLM	-	-	32.8	28.5	-
mR^{2}AG-7B [319]	CLIP ViT-L	-	-	40.6	39.8	40.2
ReflectiVA-8B [58]	EVA-CLIP-8B	35.5	35.5	40.4	39.8	40.1
General-Purpose MLLMs						
BLIP-2 [146]	-	12.6	12.4	12.7	12.3	12.5
InstructBLIP [66]	-	11.9	12.0	8.9	7.4	8.1
LLaVA-1.5-7B [167]	-	16.3	16.9	9.6	9.4	9.5
LLaVA-MORE-8B [57]	-	13.8	14.9	8.9	8.0	8.4
LLaVA-MORE-8B [57]	CLIP ViT-L	17.9	19.0	14.5	13.6	14.1
LLaVA-MORE-8B [57]	SigLIP2 ViT-L	17.5	18.6	16.0	15.1	15.5
LLaVA-MORE-8B [57]	PreFLMR [30]	27.8	26.9	13.0	11.7	12.3
LLaVA-MORE-8B [57]	ReT [30]	21.9	21.8	21.1	15.0	17.5
LLaVA-MORE-8B [57]	UniIR [280]	16.9	18.2	**25.1**	18.8	21.5
LLaVA-MORE-8B [57]	ReT-2	**28.5**	**27.1**	24.3	**21.5**	**22.8**
Qwen2.5-VL-7B [13]	-	19.8	19.7	18.6	18.1	18.3
Qwen2.5-VL-7B [13]	CLIP ViT-L	19.5	20.4	18.7	17.9	18.3
Qwen2.5-VL-7B [13]	SigLIP2 ViT-L	20.1	20.9	19.8	19.5	19.7
Qwen2.5-VL-7B [13]	PreFLMR [30]	**34.4**	**33.0**	18.0	15.8	16.8
Qwen2.5-VL-7B [13]	ReT [30]	26.6	26.2	24.5	17.9	20.7
Qwen2.5-VL-7B [13]	UniIR [280]	18.6	19.2	**29.0**	22.4	25.3
Qwen2.5-VL-7B [13]	ReT-2	33.5	31.6	27.9	**25.1**	**26.4**

nism may be exposed to the size of the knowledge base more severely than single-token retrieval. Switching to Qwen2.5-VL, the results are better than LLaVA-MORE, testifying the superior capabilities of this more recent MLLM. In this context, ReT falls slightly behind PreFLMR on Encyclopedic-VQA, but compensates for that by confirming itself as the best retriever on InfoSeek, scoring 9.6 points higher than PreFLMR, which even underperforms compared to Qwen2.5-VL without retrieval. Overall, these results confirm the effectiveness of our approach, showing that off-the-shelf MLLMs can achieve competitive performance in knowledge-intensive VQA without task-specific fine-tuning.

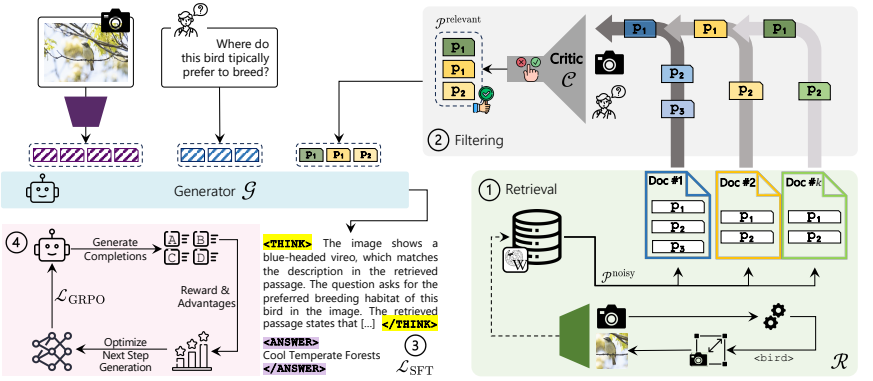


Figure 6.3: Overview of the proposed ReAG model. A multi-level retriever module extracts noisy passages, which are refined by a critic model. The resulting relevant passages are fed to a generator trained via SFT and a reinforcement learning stage designed for the KB-VQA task.

6.4 Reasoning-Augmented Generation for KB-VQA

As discussed earlier, despite the limitations of CLIP as a retrieval model, our analysis of WikiLLaVA highlights an additional and equally important challenge: developing an efficient and scalable paradigm for selecting the most useful information from one or more retrieved documents. Simply concatenating retrieved content into the model's context is often insufficient and can even introduce noise. For this reason, we propose enhancing the model's ability to assess the appropriateness and reliability of retrieval.

Overview. To address the existing challenges of retrieval-augmented models, **ReAG** enhances KB-VQA performance by employing retrieval, filtering, and reasoning over the retrieved passages. The approach consists of two key components: a **critic model** that filters retrieval results, and a **generator** trained to reason over the filtered documents before producing the final answer. The overall pipeline is organized into four main stages: (1) a multi-

level retrieval stage to gather candidate passages, (2) a filtering stage where the critic selects relevant content, (3) a cold-start supervised fine-tuning (SFT) stage to instill initial reasoning capabilities in the generator, and (4) a reinforcement learning stage to further refine reasoning and answer generation. Together, these components reduce noise and improve the ability of the generator to produce accurate, knowledge-intensive answers. An overview of our methodology is shown in Fig. 6.3.

6.4.1 Retrieving Evidence at Multiple Granularities

The retrieval stage identifies potentially informative passages related to the query image, which are subsequently filtered to provide the generator with relevant external knowledge for reasoning and answer generation. This process comprises two complementary steps: a *coarse-grained* retrieval, which retrieves candidate documents based on the entire query image, and a *fine-grained* retrieval, which performs retrieval using localized cues. Notably, ReAG is agnostic to the choice of retriever, so $\mathcal{R}$ can be any cross-modal encoder that maps the query image and either the metadata $\mathcal{T}_i$ or the image I_i associated with each document d_i into a shared embedding space. Relevance between queries and documents is then computed via cosine similarity.

Coarse-Grained Retrieval. An initial set of relevant textual passages, denoted as $\mathcal{P}^{\text{cg}}$, is constructed by aggregating all the passages contained in the top-k retrieved documents when using the original image I_q as query to the retriever $\mathcal{R}$. Since each document contains a variable number of passages, the resulting collection is represented as $\mathcal{P}^{\text{cg}} = \{p_1^{\text{cg}}, \ldots, p_m^{\text{cg}}\}$, where m denotes the total number of passages from the top-k documents.

Fine-Grained Retrieval. To improve retrieval recall, we introduce a fine-grained retrieval stage that focuses on the specific visual region relevant to the question. Given the input image I_q and the question q, we identify a

bounding box corresponding to the subject of the question, employing an off-the-shelf detection model. If such a region is detected, we crop the image accordingly, obtaining a focused image patch I_q^{crop}. This cropped image is then used as input to the retriever model $\mathcal{R}$, which computes scores with respect to each document d_i, as in the coarse-grained stage.

The top-k documents retrieved in this stage form the fine-grained candidate passages, denoted as $\mathcal{P}^{\text{fg}} = \{p_1^{\text{fg}}, \ldots, p_l^{\text{fg}}\}$. By restricting the visual input to the region of interest, this stage allows the retriever to focus on more fine-grained visual details, yielding passages that are more likely to be relevant to the specific question.

Final Set of Retrieved Passages. The documents comprising $\mathcal{P}^{\text{cg}}$ and $\mathcal{P}^{\text{fg}}$ are merged and ranked by their relevance scores, and all passages contained in the top-k ranked documents are retained to form the final set $\mathcal{P}^{\text{noisy}}$ from $\mathcal{KB}$.

6.4.2 Selecting Reliable Passages

After the retrieval steps, we obtain a set $\mathcal{P}^{\text{noisy}}$ of passages from the k retrieved documents. While increasing k generally improves recall by including more potentially relevant passages, this typically comes at the cost of a lower precision, as the probability of introducing noisy information rises as well. To mitigate this, we design a critic model $\mathcal{C}$ to filter out irrelevant passages, resulting in a refined set of relevant passages $\mathcal{P}^{\text{relevant}}$.

Critic Model. Given a question q and its corresponding image I_q, the critic model $\mathcal{C}$ predicts if each retrieved textual passage in $\mathcal{P}^{\text{noisy}}$ is useful for answering the question. In ReAG, the critic model is implemented as an autoregressive MLLM fine-tuned with a next-token prediction objective optimized on an annotated dataset. Specifically, starting from a subset of samples drawn from the dataset employed in [58], we extract tuples (I_q, q, p, y), where p is a textual passage to be evaluated and $y \in \{\textit{Yes}, \textit{No}\}$ indicates

whether the passage is relevant. The critic model is trained to predict y conditioned on (I_q, q, p), enabling it to robustly discriminate between relevant and irrelevant passages.

At inference time, only passages yielding a positive prediction with probability above a threshold are kept, yielding the final subset of relevant passages $\mathcal{P}^{\text{relevant}}$, defined as:

$$\mathcal{P}^{\text{relevant}} = \{\, p \in \mathcal{P}^{\text{noisy}} \mid \Pr(\text{Yes} \,|\mathcal{C}, q, I_q, p) > \text{thresh} \,\}. \tag{6.4}$$

The resulting set $\mathcal{P}^{\text{relevant}}$ is fed to the generator $\mathcal{G}$, which leverages these passages to produce the final answer.

6.4.3 Training with Reasoning Supervision

Following the approach popularized by DeepSeek-R1 [89], we train our generator $\mathcal{G}$ using a multi-stage strategy. The initial stage is designed to enhance the reasoning and zero-shot capabilities of the model, while mitigating potential instabilities during the subsequent reinforcement learning stage. Unlike standard SFT, which focuses solely on answer prediction, our cold-start phase exposes $\mathcal{G}$ to explicit reasoning trajectories that link the visual content, retrieved passages, and the question.

Collecting Reasoning Traces. To achieve this, we fine-tune $\mathcal{G}$ using high-quality reasoning data. Starting from the same subset used for training the critic model, we extend each tuple (I_q, q, p, y) with a reasoning trace tr. Specifically, each tuple is provided as input to an MLLM, which is prompted to generate an explicit reasoning trace that logically explains how the passage p contributes to answering the question q given the image I_q. To guide this reasoning, the prompt includes both the final answer and the relevance label y, indicating whether the reasoning should be grounded in the passage or not. By explicitly conditioning on these signals, the MLLM

produces structured reasoning traces that reflect a coherent inference process from the evidence to the answer. These traces are used as supervision for the cold-start fine-tuning of the generator $\mathcal{G}$.

Training Protocol. Having collected the reasoning-augmented dataset, the generator $\mathcal{G}$ is trained to optimize both its reasoning ability and answer accuracy. To guide the model towards structured reasoning behavior, we encourage a templated output format, where the reasoning trace and the final answer are delimited by special tokens which are explicitly added to the vocabulary, *i.e.*

`<think>` reasoning trace `</think>`
`<answer>` answer `</answer>`.

This structure encourages the model to separate intermediate reasoning from the final prediction, improving interpretability and stability during generation. Training is performed using a next-token prediction objective over both the reasoning trace and the final answer. The overall SFT loss balances the two components as follows:

$$\mathcal{L}_{\mathrm{SFT}} = \alpha \mathcal{L}_{\mathrm{A}} + (1-\alpha)\mathcal{L}_{\mathrm{T}}, \tag{6.5}$$

where $\mathcal{L}_{\mathrm{A}}$ and $\mathcal{L}_{\mathrm{T}}$ denote the negative log-likelihood losses computed over the answer and reasoning trace, respectively.

6.4.4 Training Optimization via Reinforcement Learning

While supervised fine-tuning equips the generator with basic reasoning skills and coherent chain-of-thought generation, we enhance its quality and robustness through a subsequent reinforcement learning stage.

Task-specific RL with Retrieved Passages. Our generator model is optimized with a custom objective inspired by GRPO [240], incorporating modifications from DAPO [309]. Formally, the objective is defined as follows:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{(I_q,q,p)\sim\mathcal{D},\{o_i\}_{i=1}^{N}\sim\mathcal{G}_{\theta_{\text{old}}}(\cdot|I_q,q,p)}\Bigg[\frac{1}{\sum_{i=1}^{N}|o_i|}\sum_{i=1}^{N}\sum_{t=1}^{|o_i|}\min\Big(r_{i,t}(\theta)\hat{A}_{i,t}, \text{clip}\left(r_{i,t}(\theta), 1-\varepsilon, 1+\varepsilon\right)\hat{A}_{i,t}\Big)\Bigg], \tag{6.6}$$

where $\mathcal{G}_{\theta_{\text{old}}}$ is the generator initialized from the SFT cold-start phase and $\mathcal{G}_\theta$ the generator after optional off-policy updates. Moreover, $\{o_i\}_{i=1}^{N}$ are the generated completions with associated rewards R_i, and $\hat{A}_i$ the corresponding GRPO advantages [240].

In the formula, $r_{i,t}$ is computed as:

$$r_{i,t}(\theta) = \frac{\mathcal{G}_\theta(o_{i,t} \mid I_q, q, p, o_{i,<t})}{\mathcal{G}_{\theta_{\text{old}}}(o_{i,t} \mid I_q, q, p, o_{i,<t})}. \tag{6.7}$$

As in our setting the updates are never off-policy, $\mathcal{G}_\theta$ coincides with $\mathcal{G}_{\theta_{\text{old}}}$, thus the ratio $r_{i,t}(\theta)$ is always 1. Unlike the GRPO formulation and following DAPO, we omit the KL divergence penalty, which overly constrains exploration of alternative reasoning trajectories. This also improves memory efficiency and training speed by removing the need for a reference model and an extra forward pass. Furthermore, the loss is computed at the token level, as averaging over variable-length sequences would reduce the contribution of tokens in longer sequences and weaken their updates.

At each training iteration, the generator $\mathcal{G}$ is prompted with (I_q, q, p) to autoregressively generate a group of N completions $\{o_i\}_{i=1}^{N}$. Each generated completion is then evaluated by one or more rule-based reward functions, producing a reward R_i to compute the advantage as:

$$\hat{A}_{i,t} = \frac{R_i - \text{mean}(\{R_i\}_{i=1}^{N})}{\text{std}(\{R_i\}_{i=1}^{N})}. \tag{6.8}$$

Advantages guide the policy updates: completions with above-average rewards have their likelihood increased, while those below the mean are

down-weighted. This exposure allows the generator to explore diverse strategies for interacting with passages, gradually steering it toward producing reasoning trajectories that yield correct answers.

Rule-Based Reward Design. In our setting, we employ two complementary rule-based binary reward functions: a task-specific *accuracy reward* and a *format reward*. The task-specific accuracy reward $R_{\text{task}}(o_i)$ verifies whether a generated completion is correct by parsing the prediction according to the question type (*i.e.*, numerical or textual, single- or multi-answer). The format reward $R_{\text{fmt}}(o_i)$ enforces adherence to the expected output template. Both functions return 1 in case of success and 0 otherwise. The final reward associated to a completion o_i is defined as a weighted sum of the two. Formally,

$$R_i = \gamma R_{\text{task}}(o_i) + \delta R_{\text{fmt}}(o_i), \tag{6.9}$$

where γ and δ are two hyperparameters.

Reward Design Details. As discussed in the previous paragraph, we employ a task-specific accuracy reward. The reward function evaluates only the final answer rather than the intermediate reasoning. To extract the predicted answer, we first search for content enclosed within the `<answer></answer>` tags. If no such content is found, we extract all text following the first `<answer>` tag. If this is unsuccessful, we instead use the text following the `</think>` tag. When none of these patterns appear, the entire model output is used.

In every case, format-specific special tokens are removed. We then apply the same normalization procedure used in the InfoSeek and Encyclopedic-VQA evaluations, including the removal of articles, punctuation, extra whitespace, and capitalization, along with standardization of digits and contractions. The final task-specific reward depends on the source dataset and task type, and is computed as follows:

$$R_{\text{task}}(\tilde{o}_i, o_i^*, \tau_i) = \begin{cases} \mathbb{1}\big[\Psi_{\text{num}}(\tilde{o}_i, o_i^*)\big], & \text{if } \tau_i = \text{numerical}, \\ \mathbb{1}\big[\,\text{IoU}(\tilde{o}_i, o_i^*) \geq 0.5\big], & \text{if } \tau_i = \text{scalar}, \\ \mathbb{1}\big[\tilde{o}_i = o_i^*\big], & \text{otherwise}\,. \end{cases} \tag{6.10}$$

where $\tilde{o}_i$, o_i^* and τ_i denote respectively the extracted prediction, the ground-truth answer and the task type of the i-th sample, and Ψ_{num} evaluates success or failure in numerical match. When multiple alternative ground-truths are provided for a sample, we compute the reward with respect to each and take the maximum. For samples from InfoSeek, we use exact string matching for *entity* and *time* questions, while *numerical* questions are evaluated with Ψ_{num}:

$$\Psi_{\text{num}}(\tilde{o}, o^*) = \begin{cases} |\tilde{o} - o^*| \leq 0.1, & \text{if } \text{is\_scalar}(\tilde{o}) \\ & \quad \wedge\ \text{is\_scalar}(o^*), \\ \tilde{o} \in o^*, & \text{if } \text{is\_scalar}(\tilde{o}) \\ & \quad \wedge\ \text{is\_interval}(o^*), \\ \text{IoU}(\tilde{o}, o^*) \geq 0.5, & \text{if } \text{is\_interval}(\tilde{o}) \\ & \quad \wedge\ \text{is\_interval}(o^*). \end{cases} \tag{6.11}$$

For samples from Encyclopedic-VQA dataset, we adopt exact match scoring for *single-answer* questions, while for *multi-answer* questions the prediction is rewarded as correct only if intersection-over-union between predicted and ground-truth items reaches or surpasses 0.5.

6.5 Comparative Analysis of Retrieval-augmented Architectures

6.5.1 Implementation Details

Retrieval Details. To retrieve potentially informative documents for a query image, we employ EVA-CLIP-8B [255]. In the coarse-grained stage, the entire query image is encoded through EVA-CLIP to perform large-scale retrieval over the knowledge base. For InfoSeek, we use an image-to-text retrieval setup that computes similarity between the query image and document metadata (*i.e.*, the title of the page and the summary). For Encyclopedic-VQA, we adopt image-to-image retrieval, comparing the query image with the images inside Wikipedia pages. In the fine-grained stage, we extract the visual subject mentioned in the question using the spaCy library* and localize it in the image via GroundingDINO [171], whose bounding box is re-encoded through EVA-CLIP. Retrieval is done using the FAISS library [115], with the top-k results with $k = 20$ retained at each stage.

Critic Model and Dataset. Independently from the generator scale, our critic model builds upon Qwen2.5-VL-3B, fine-tuned on a curated subset of the ReflectiVA dataset [58] The model is trained for 1 epoch with a learning rate of 2×10^{-6} and a global batch size of 32.

Generator Training. We build two versions of our generator, both based on Qwen2.5-VL [13], using the 3B and 7B model variants, and optimize them using the SFT plus RL training scheme. In the SFT phase, we use the same ReflectiVA [58] subset employed to train the critic model, collecting reasoning traces from Qwen2.5-VL-7B. We set $\alpha = 0.8$ to give more importance to final-answer tokens. The generator is trained for one epoch with SFT

*https://github.com/explosion/spaCy

loss using AdamW [176], a learning rate of 2×10^{-6} and an effective batch size of 128. For RL post-training, we use the full Encyclopedic-VQA and InfoSeek sets from ReflectiVA, excluding samples from LLaVA-Instruct [166]. Each batch includes 128 prompts with 8 completions per prompt. Training is conducted with Adam [125], a learning rate 1×10^{-6}. Rewards weigh accuracy $\gamma = 1.0$ over format $\delta = 0.2$. In all our experiments, we update the MLP adapter and LLM weights while keeping the vision encoder frozen.

6.5.2 Comparison with the State of the Art

Main Results. We present a comprehensive comparison of ReAG on the E-VQA test set and the InfoSeek validation set against both zero-shot MLLMs and retrieval-augmented baselines. Specifically, we evaluate BLIP-2 [146], LLaVA-v1.5 [166], LLaVA-MORE [57], and Qwen2.5-VL [13] in a zero-shot setting, where the models receive only the query image and question as input. We further include retrieval-augmented approaches such as DPR [140], RORA-VLM [212], EchoSight [293], COMEM [286], WikiLLaVA [29], mR^{2}AG [319], mKG-RAG [310], ReflectiVA [58], and VLM-PRF [101]. For fairness, we reproduce ReflectiVA using Qwen2.5-VL backbones at both 3B and 7B scales.

As shown in Table 6.10, zero-shot MLLMs, which rely solely on internal knowledge, are unable to accurately answer questions in knowledge-intensive benchmarks, underscoring the need for external retrieval. With the introduction of a retrieval pipeline, performance improves substantially. For example, on InfoSeek, the overall accuracy rises from around 20% for the zero-shot Qwen2.5-VL-7B model to roughly 40% with retrieval-augmented methods such as mKG-RAG. ReAG further advances these results, achieving state-of-the-art performance on both E-VQA and InfoSeek across model scales. Specifically, on E-VQA ReAG yields a $+7.7$ point gain over ReflectiVA when using Qwen2.5-VL-3B and a $+7.8$ point improvement over VLM-PRF when leveraging the stronger InternVL3-8B backbone.

Table 6.10: VQA accuracy scores on the Encyclopedic-VQA test set and the InfoSeek validation set. The marker † represents our reproductions, while gray color indicates models tested with non-comparable knowledge bases.

Model	Generator	Retriever	E-VQA		InfoSeek		
			Single-Hop	All	Unseen-Q	Unseen-E	All
BLIP-2 [146]	-	-	12.6	12.4	12.7	12.3	12.5
LLaVA-v1.5-7B [166]	-	-	16.3	16.9	9.6	9.4	9.5
LLaVA-MORE-8B [57]	-	-	16.0	16.9	8.3	8.9	7.8
Qwen2.5-VL-3B [13]	-	-	21.9	21.9	18.9	17.7	18.3
Qwen2.5-VL-7B [13]	-	-	23.6	23.2	22.8	24.1	23.7
DPR_{V+T} [140]	Multi-passage BERT	CLIP ViT-B/32	29.1	-	-	-	12.4
RORA-VLM [212]	LLaVA-v1.5-7B	CLIP ViT-L/14	-	20.3	25.1	27.3	-
EchoSight [293]	Mistral-7B/LLaMA-3-8B	EVA-CLIP-8B	41.8	-	-	-	31.3
CoMEM [286]	Qwen2.5-VL-7B	Custom VLM	-	-	32.8	28.5	-
Wiki-LLaVA [29]	LLaVA-v1.5-7B	CLIP ViT-L/14+Contriever	17.7	20.3	30.1	27.8	28.9
EchoSight [293]	LLaMA-3.1-8B	EVA-CLIP-8B	36.3	34.2	30.0	30.7	30.4
ReflectiVA [58]	LLaVA-MORE-8B	EVA-CLIP-8B	35.5	35.5	40.4	39.8	40.1
mR^2AG [319]	LLaVA-v1.5-7B	CLIP ViT-L/14	-	-	40.6	39.8	40.2
VLM-PRF [101]	LLaVA-MORE-8B	EVA-CLIP-8B	36.3	35.5	41.3	40.6	40.8
mKG-RAG [310]	LLaVA-MORE-8B	Custom VLM	38.4	36.3	41.4	39.6	40.5
ReflectiVA [58]†	Qwen2.5-VL-3B	EVA-CLIP-8B	33.7	35.2	39.6	38.1	38.9
VLM-PRF [101]	Qwen2.5-VL-3B	EVA-CLIP-8B	31.1	32.4	39.7	38.8	39.0
ReAG (Ours)	Qwen2.5-VL-3B	EVA-CLIP-8B	**41.3** Δ+7.6	**42.9** Δ+7.7	**43.7** Δ+4.0	**42.9** Δ+4.1	**43.3** Δ+4.3
ReflectiVA [58]†	Qwen2.5-VL-7B	EVA-CLIP-8B	36.8	36.8	43.5	44.3	43.9
VLM-PRF [101]	Qwen2.5-VL-7B	EVA-CLIP-8B	37.1	36.0	43.3	42.7	42.8
VLM-PRF [101]	InternVL3-8B	EVA-CLIP-8B	40.1	39.2	43.5	42.1	42.5
ReAG (Ours)	Qwen2.5-VL-7B	EVA-CLIP-8B	**44.9** Δ+4.8	**47.0** Δ+7.8	**48.3** Δ+4.8	**46.2** Δ+1.9	**47.2** Δ+3.3

Similar gains are observed on InfoSeek, with overall improvements of $+4.3$ and $+3.3$ points for Qwen2.5-VL-3B and 7B, respectively. These consistent improvements across both datasets demonstrate the effectiveness and robustness of our approach.

Results with OMGM Retrieval Mode. We also compare against the OMGM framework [294], which adopts a coarse-to-fine, multi-stage retrieval strategy, leveraging an image-to-text summary retriever in the first step. To ensure a fair comparison, in Table 6.11 we evaluate our method using the same retrieval modality. Notably, our method consistently outperforms OMGM [294] across both E-VQA and InfoSeek benchmarks. With the Qwen2.5-VL-3B generator, our approach achieves substantial improvements over ReflectiVA [58], yielding gains of $+4.5$ and $+5.1$ points on E-VQA and InfoSeek. When scaling to Qwen2.5-VL-7B, performance further increases, reaching 52.5 on E-VQA and 49.2 on InfoSeek, surpassing OMGM

Table 6.11: VQA accuracy scores on Encyclopedic-VQA and InfoSeek with OMGM as retrieval modality.

Model	Generator	E-VQA	InfoSeek		
		Single-Hop	Un-Q	Un-E	All
ReflectiVA [58]	LLaVA-MORE-8B	41.8	33.8	34.5	34.1
OMGM [294]	LLaVA-v1.5-7B	50.2	43.5	43.5	43.5
ReflectiVA [58]	Qwen2.5-VL-3B	44.6	40.5	41.6	41.1
ReAG (Ours)	Qwen2.5-VL-3B	**49.1**	**47.2**	**45.3**	**46.2**
ReflectiVA [58]	Qwen2.5-VL-7B	44.0	43.3	44.0	43.6
ReAG (Ours)	Qwen2.5-VL-7B	**52.5**	**50.3**	**48.2**	**49.2**

Table 6.12: VQA accuracy scores on Encyclopedic-VQA and InfoSeek with oracle Wikipedia pages.

Model	Generator	E-VQA	InfoSeek		
		Single-Hop	Un-Q	Un-E	All
Qwen2.5-VL-3B [13]	Qwen2.5-3B	72.1	47.0	43.0	44.9
Qwen2.5-VL-7B [13]	Qwen2.5-7B	78.3	41.6	41.3	41.4
ReflectiVA [58]	Qwen2.5-VL-3B	72.9	53.4	**53.9**	53.7
ReAG (Ours)	Qwen2.5-VL-3B	**79.0**	**55.1**	53.3	**54.2**
Wiki-LLaVA [29]	LLaVA-v1.5-7B	38.5	52.7	50.3	51.5
ReflectiVA [58]	LLaVA-MORE-8B	75.2	57.8	57.4	57.6
ReflectiVA [58]	Qwen2.5-VL-7B	71.3	56.0	56.0	56.0
ReAG (Ours)	Qwen2.5-VL-7B	**81.5**	**60.7**	**58.9**	**59.7**

by 2.3 and 5.7 points, respectively. These results indicate that, even when using only the initial retrieval step of OMGM, our critic-based filtering, together with ReAG reasoning capabilities, leads to higher performance compared to both prior methods and the full multi-stage retrieval of OMGM.

Results with Oracle Documents. We also experiment under an oracle setting (Table 6.12), where the ground-truth document (*i.e.*, the Wikipedia page corresponding to the query) is provided. We compare results from zero-shot models (Qwen2.5-VL [13] in both 3B and 7B variants), which take the entire Wikipedia pages as input, and retrieval-based methods (WikiLLaVA [29], ReflectiVA [58], and ReAG), which process retrieved passages through an additional model-specific filtering stage. In this configuration, ReAG re-

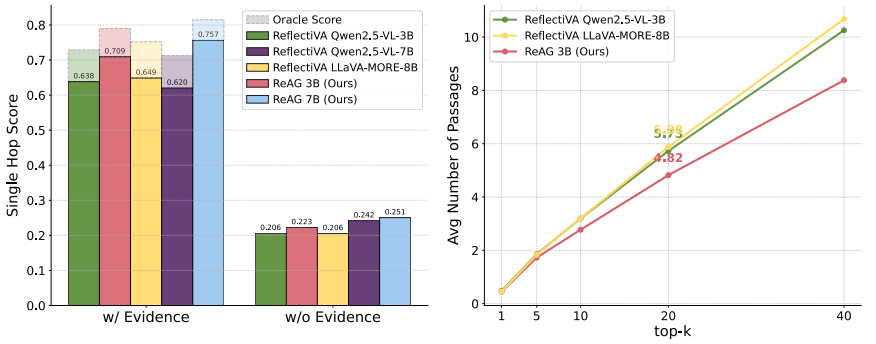


Figure 6.4: Comparison of performance on E-VQA with and without evidence, including oracle upper bounds (left). Analysis on average number of passages retained at different top-k values (right).

ceives all passages from the oracle document, which are then passed to the critic model for filtering before being fed to the generator.

Notably, ReAG achieves the best performance across both E-VQA and InfoSeek at all model scales. On the 3B variant, ReAG outperforms ReflectiVA by $+6.1$ points on E-VQA, while on Infoseek the 7B version consistently improves over ReflectiVA by $+3.7$ and still surpasses it by $+2.1$ even when ReflectiVA employs a larger generator (LLaVA-MORE-8B).

Retrieval and Generation Pipeline Analysis. The performance of RAG-based approaches strongly depends on the presence of the evidence passage in the retrieved set, and on the passages provided to the generator.

In Fig. 6.4 (left), we evaluate the ability to produce the correct answer when the evidence passage is either present or absent in the context. ReAG consistently outperforms all competitors of comparable scale, demonstrating that our reasoning-enhanced approach is robust even in the absence of direct evidence. Each model is also accompanied by its oracle performance, clearly showing that ReAG consistently gets closer to the oracle upper bound than other approaches.

In Fig. 6.4 (right), we instead report the average number of passages

Q: What is the closest upper taxonomy of this bird?

Qwen2.5-VL-7B (ZS) [13]: The closest taxonomy of this bird is the family Laridae. ✗
ReflectiVA [58]: Sterna ✗
ReAG (Ours): Thalasseus ✓

Q:Which road, railway or canal does this bridge carry?

Qwen2.5-VL-7B (ZS) [13]: The bridge in the picture is Blackfriars Bridge. ✗
ReflectiVA [58]: Road ✗
ReAG (Ours): A201 ✓

Figure 6.5: Qualitative results on InfoSeek image-question pairs comparing ReAG, ReflectiVA [58], and the corresponding zero-shot model.

passed to the generation at different k values, comparing the filtering behavior of ReflectiVA [58] with that of our critic model. ReAG reduces noise introduced in the generator context by achieving a reduction of 18.0% and 15.9% in the number of passages compared to ReflectiVA based on LLaVA-MORE-8B and Qwen2.5VL-3B respectively, further emphasizing the advantage of our filtering pipeline.

KB-VQA Qualitative Results. In Fig. 6.5, we present a qualitative comparison on image-question pairs from InfoSeek. Notably, the zero-shot model tends to produce longer and detailed answers, whereas ReflectiVA and ReAG generate responses that follow the dataset-specific format. Overall, the results consistently demonstrate that ReAG answers accurate and outperforms competing approaches.

Reasoning Traces Qualitatives. To further interpret the behavior of our model, we visualize qualitative examples of the reasoning traces generated by ReAG in Fig. 6.6. The zero-shot baseline produces partial reasoning but lacks a consistent structure and does not adhere to the output format defined by the evaluation datasets. In contrast, ReflectiVA follows the correct answer format but fails to generate explicit reasoning traces, limiting interpretability. In contrast, ReAG generates coherent, well-structured traces that reveal the step-by-step logic behind its predictions. These examples highlight the ability of the proposed solution to integrate visual and textual cues, assess the relevance of retrieved passages, and main-

Q: What genus are this plant and *Schefflera actinophylla* in?

Qwen2.5-VL-7B (ZS) [13]:
Not enough information is provided to determine the genus in question ✗

ReflectiVA [58]:
Schefflera ✗

ReAG (Ours):
<think> The image shows a plant that matches the description of *Schefflera actinophylla* in the retrieved passage. The question asks for the genus shared by the plant in the image and *Schefflera actinophylla*. Both belong to the same genus according to the retrieved evidence. </think> <answer> Heptapleurum ✓ </answer>

Figure 6.6: Qualitative example from Encyclopedic-VQA illustrating ReAG's ability to reason over the question–image pair and retrieved passages. For comparison, we report generations from ReflectiVA [58] and the corresponding zero-shot model.

tain consistent reasoning even under noisy or irrelevant evidence, where baselines often over-rely on passages or hallucinate unsupported details.

6.5.3 Ablation Studies

Architectural Components. We finally perform an ablation study by progressively enabling key components of the final architecture. Table 6.13 reports results at each stage with k equal to 20, evaluating the effect of the critic and fine-grained retrieval modules, as well as design choices in the generation pipeline.

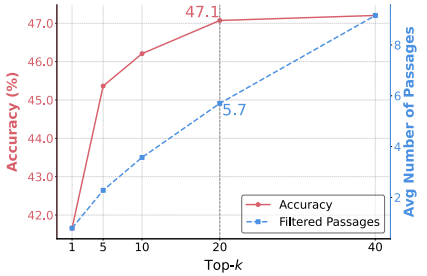
We first examine the zero-shot setup under different retrieval configurations, using the 3B-scale model. As shown in the first two rows of Table 6.13, directly passing all passages from the top-20 documents into the generator severely degrades performance due to excessive noise. Introducing the critic model (third row) effectively filters irrelevant passages, yielding more than a twofold gain. Adding the fine-grained retriever (fourth row) further improves retrieval recall and yields additional gains on both datasets.

Table 6.13: Ablation results on E-VQA and InfoSeek to validate the effectiveness of our model component. CG and FG stands for coarse- and fine-grained retrieval.

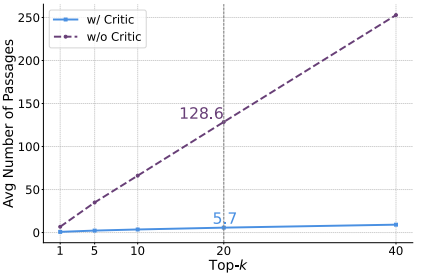
		Retrieval Pipeline			Generation Pipeline			E-VQA	InfoSeek		
	Generator	CG	FG	Critic	SFT	Reasoning Traces	RL	Single-Hop	Unseen-Q	Unseen-E	All
		-	-	-	-	-	-	21.9	18.9	17.7	18.3
		✓	-	-	-	-	-	19.2	10.2	10.0	10.1
		✓	-	✓	-	-	-	38.0	27.9	26.1	27.0
		✓	✓	✓	-	-	-	40.2	28.1	26.1	27.1
		✓	✓	✓	✓	-	-	39.3	37.9	37.1	37.5
		✓	✓	✓	✓	✓	-	38.1	41.9	40.6	41.3
		✓	✓	✓	✓	-	✓	39.5	39.8	39.4	39.6
ReAG (Ours)	Qwen2.5-VL-3B	✓	✓	✓	✓	✓	✓	**41.3**	**43.7**	**42.9**	**43.3**
		✓	✓	✓	-	-	-	41.7	29.3	27.8	28.5
		✓	✓	✓	✓	✓	-	42.0	42.0	40.7	41.4
ReAG (Ours)	Qwen2.5-VL-7B	✓	✓	✓	✓	✓	✓	**44.9**	**48.3**	**46.2**	**47.2**

Fixing the retrieval pipeline to its configuration, we then assess the impact of different generation strategies. As shown in [89], introducing a cold-start phase can help prepare the model for reasoning by exposing it to intermediate traces before full supervision. Results show that applying reinforcement learning after this cold-start phase outperforms standard SFT, indicating that the cold-start phase effectively prepares the model for multi-step reasoning, allowing the RL algorithm to operate on a model already prepared for structured thinking. A similar trend is observed with the 7B variant, where both training phases contribute significantly to the final performance, validating the robustness of the proposed ReAG pipeline.

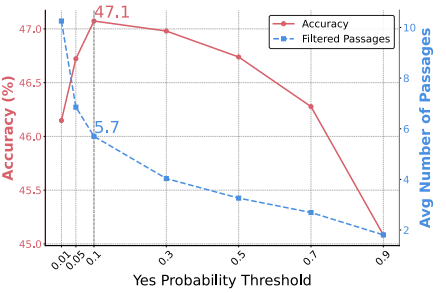
Varying the Number of Retrieved Documents. In Fig. 6.7a, we analyze the effect of varying the number of retrieved documents k on the overall performance and on the average number of filtered passages fed to the generator. As shown, the model achieves the best results around $k = 20$, which represents the optimal trade-off between coverage and noise, and is therefore adopted as the default retrieval depth in our pipeline. Retrieving too few documents results in insufficient contextual evidence, causing a drop in recall and limiting the ability of the model to access the necessary information. Conversely, increasing k beyond this point does not yield meaningful performance gains while substantially inflating the computational cost of the filtering stage.



(a) Performance as a function of the number of retrieved documents k.



(b) Effect of critic filtering on the number of passages.



(c) Performance as a function of the critic yes-probability threshold.

Figure 6.7: Analysis of the ReAG retrieval and filtering pipeline. Accuracy and number of filtered passages are averaged over E-VQA and InfoSeek.

Effectiveness of the Critic Model. In Fig. 6.7b, we provide a detailed analysis on the effectiveness of the proposed critic model, employed in ReAG to filter relevant passages. Specifically, the plot reports the average number of passages retained after the filtering performed by the critic model when varying the number k of retrieved documents. Across all retrieval sizes, the critic model substantially reduces the number of retained passages (*e.g.*, from an average of 128.6 to 5.7 at $k = 20$), while preserving answer-relevant information. This highlights the strong ability of the critic model to discard noisy or off-topic passages, leading to a more compact and semantically aligned evidence set for multimodal reasoning.

Critic Threshold. In Fig. 6.7c, we report how the performance varies as a function of the *yes*-probability threshold used in our critic model (cf. Eq. 6.4). The results show that instead of simply letting the fine-tuned MLLM decide if the passage is relevant or not (*i.e.*, thresh $= 0.5$), leveraging the confidence of the model in predicting the "Yes" token allows us to gain more control over the filtering phase. A threshold thresh $= 0.1$ yields the best trade-off between precision and recall in retrieving relevant passages. This setting enables the critic model to reliably filter out passages for which it is most confident of their non-relevance to the query.

7

Image Captioning Evaluation Metrics: Background

As captioning models improve, first through retrieval augmentation and more recently through their integration within multimodal large language models, the role of evaluation metrics becomes increasingly critical. A good metric must assess captions across multiple dimensions to ensure alignment with human expectations both linguistically and semantically. A strong captioning metric should reward captions that are **fluent**, **accurate**, and **faithful** to the image content, while also ensuring they focus on the **most salient aspects** and strike a balance be-

This chapter discusses topics from the following papers: S. Sarto *et al.*, "Image Captioning Evaluation in the Age of Multimodal LLMs: Challenges and Future Perspectives", IJCAI 2025 [233].

tween **detail** and **conciseness**. Furthermore, it should penalize hallucinated or misleading information. With the rise of MLLMs as image description generators, the nature of captions has changed, as these models generate captions that differ in length and specificity. Conventional evaluation metrics may no longer reliably assess caption quality in this new paradigm, posing new challenges for the task.

7.1 Taxonomy

In this section, we present the evolution of captioning metrics, outlining their development from traditional non-learning-based evaluation strategies to recent advancements incorporating LLMs. Fig. 7.1 presents a taxonomy that highlights the key characteristics of the most commonly used metrics, including their reliance on reference captions or image input.

In captioning literature, the performance of vision-and-language tasks are mainly measured using standard captioning evaluating metrics, namely BLEU [207], METEOR [15], ROUGE [154], CIDEr [271], and SPICE [5].

BLEU. It gauges the precision of word n-grams by comparing predicted and ground-truth. As in previous works, we mostly report performance with BLEU using n-grams of length 1 and 4 (referred to as B-1 and B-4, respectively).

ROUGE. It calculates an F-measure with a recall bias, using a technique based on identifying the longest common subsequence.

METEOR. It instead, assesses captions by aligning them to one or more ground-truth sentences, utilizing alignments based on various matches like exact, stem, synonym, and paraphrase between words and phrases.

CIDEr. It computes the average cosine similarity between n-grams present in the generated caption and reference sentences, employing TF-IDF weighting.

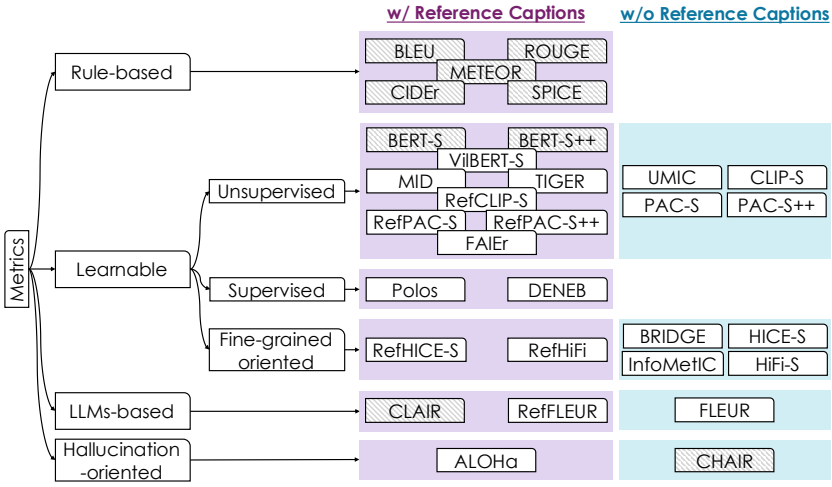


Figure 7.1: Taxonomy of image captioning metrics, distinguishing between reference-based and reference-free approaches. Grey dashed lines indicate the absence of the input image during evaluation.

SPICE. It prioritizes semantic content over fluency in generated captions by matching tuples from the candidate and reference scene graphs.

Empirical evidence suggests that BLEU and ROUGE exhibit weaker correlations with human judgment compared to other metrics [271, 229], yet the standard practice in image captioning literature involves reporting all mentioned metrics.

In our taxonomy, we group the aforementioned metrics under the category of **rule-based metrics**, as they rely on predefined similarity functions rather than learned evaluators. As discussed above, these metrics predominantly operate through n-gram matching or structured comparisons against reference captions and do not take the input image into account during evaluation.

We then organize the remaining metrics into three broad categories.

Learnable Metrics. All the aforementioned metrics primarily rely on textual-level comparisons and struggle to adequately account for synonym matches from a linguistic perspective. Moreover, they assume that human-

written reference captions perfectly reflect the content of the image, which may not always be the case. To overcome these limitations, recent metrics leverage pre-trained models to compare textual-only and visual-textual content, using both supervised and unsupervised approaches. BERT-S [318], for instance, utilizes learned embeddings from the pre-trained language model BERT [68] to more effectively measure semantic similarities between candidate and reference captions. Building on this, BERT-S++ [303] enhances the approach by incorporating the variance across multiple reference captions, improving robustness. However, these metrics still do not integrate the visual modality, which can limit their ability to fully evaluate the performance of captioning models. To overcome this, metrics that explicitly include the image as input have been introduced. These methods leverage pre-trained vision-and-language models, such as UNITER [51], ViLBERT [178], and CLIP [215], to evaluate captions in a way that aligns more closely with both textual and visual content.

Leveraging the success of employing pre-trained vision-and-language models, numerous metrics started exploiting the CLIP extensive pre-training to compute image captioning evaluation scores. Among these, CLIP-S [98] was the first metric to utilize a modified cosine similarity between image and candidate caption representations derived from the visual and textual encoders of CLIP. In a similar way, MID [123] leverages CLIP-based visual-textual features to compute negative Gaussian cross-mutual information, yielding a more effective evaluation metric.

However, CLIP was not originally designed specifically for computing evaluation scores, which can limit its effectiveness. To address this challenge, a less explored research direction focuses on refining the CLIP embedding space through supervised training approaches. For instance, the Polos metric [274] is based on Polaris, a dataset specifically designed for predicting evaluation scores under direct supervision from human-annotated judgments. As part of this work, the authors propose a multi-

modal metric learning framework based on human feedback, which handles both image and text inputs and learns directly from human evaluations based on multimodal inputs.

Building on this foundation, the DENEB metric [190] introduces a supervised evaluation approach specifically designed to be robust against hallucinations. DENEB processes multiple reference captions simultaneously to effectively capture similarities between an image, a candidate caption, and reference captions. Furthermore, to improve the visual diversity of the Polaris dataset, DENEB introduces Nebula, an extended dataset containing approximately three times the number of images, further enhancing its robustness and scalability for evaluation tasks.

More recent metrics still utilize multimodal models (*e.g.* CLIP) while integrating additional components to enhance performance, particularly by improving their ability to reward fine-grained details. InfoMetIC [102] is designed to provide fine-grained feedback on caption quality. Beyond assessing precision and recall, it identifies specific errors, such as incorrect words and unmentioned image regions. To achieve this, InfoMetIC incorporates a fusion module to model intra- and inter-modality relationships, along with a fine-grained module that enhances the accuracy of error localization in both text and image content. Available in both reference-free and reference-based versions, HICE-S [312] highlights the limitations of CLIP-based metrics, which primarily assess global image-text compatibility but often struggle with detecting local textual hallucinations and maintaining sensitivity to small visual objects. To overcome these challenges, HICE-S introduces a hierarchical scoring mechanism that leverages the SAM model [126] to generate masks for localized visual regions and corresponding textual phrases. These are then processed through a modified version of the CLIP architecture [257], enhancing the model's ability to capture fine-grained visual-semantic relationships. Following a similar hierarchical approach, HiFi-S [300] is a fine-grained image description evaluation metric

that represents both text and images as parsing graphs. These graphs organize multi-granular instances into a hierarchical structure based on their inclusion relationships, enabling a comprehensive scene analysis across modalities from global to local levels. Additionally, HiFi-S incorporates an LLM to evaluate the fluency of candidate descriptions, further enhancing the evaluation process.

LLMs-based Metrics. In recent years, the integration of LLMs into the captioning evaluation pipeline has gained popularity, as demonstrated by HiFi-S. While first attempts compared generated sentences with reference captions using BERT-based embeddings, more recent approaches leverage the advanced reasoning and extensive pre-training capabilities of LLMs to produce more robust evaluation scores. For example, CLAIR [35] utilizes an LLM to rate the alignment of a candidate caption with a set of reference captions. Notably, CLAIR evaluates captions without considering image content. In response, FLEUR [138] is a reference-free, explainable evaluation metric for image captioning, which directly leverages a score generated by an MLLM and adjusted with a smoothing function to better align with human judgments.

Hallucination-oriented Metrics. In image captioning, hallucination refers to the inclusion of information, objects, or details in a generated caption that are not present in the corresponding image, compromising the reliability and quality of the description. Accurately identifying captions with potential object hallucinations is therefore essential. Metrics such as CHAIR [224] and ALOHa [209] are specifically designed to address this challenge. Specifically, the CHAIR metric calculates what proportion of words generated is actually in the image according to the ground-truth sentences and detected object. However, this metric is restricted to a fixed set of COCO objects and their synonyms. To overcome this limitation, ALOHa introduces an open-vocabulary approach that utilizes LLMs to detect ob-

ject hallucinations. Specifically, ALOHa extracts groundable objects from the candidate caption using an LLM, measures their semantic similarity to reference objects in captions or detections, and applies Hungarian matching to compute the final hallucination score.

7.2 Benchmarks

The evaluation spans multiple dimensions, including correlation with human judgments, ranking accuracy, and sensitivity to object hallucinations. In this section, we present the main setting in which captions are evaluated and the dataset employed.

7.2.1 Correlation with Human Judgment

We analyze the correlation of metrics with human judgment, highlighting their varying behaviours. Experimental results are reported on standard captioning evaluation datasets, including Flickr8k-Expert, Flickr8k-CF, and Composite, along with the more recent Polaris and Nebula datasets.

Datasets. Flickr8k-Expert [100] contains 17k annotations for 5,664 images, with each pair scored from 1 (no correlation) to 4 (accurate depiction). Flickr8k-CF [100] provides 145k binary quality judgments for 48k pairs across 1,000 unique images, using the mean proportion of "yes" votes as the alignment score. The Composite dataset [2] includes 12k human ratings for image-caption pairs (with around 4k unique images), assessed on a 1–5 scale. However, these datasets lack model-generated captions, leading to a domain gap when applying them to train evaluation metrics. To address this limitation, the Polaris dataset [274] introduces 131k human judgments from 550 evaluators (*i.e.*, around ten times larger than previous datasets) covering both human-written and machine-generated captions from ten

image captioning models. Expanding on Polaris, the Nebula dataset [190] includes 32,978 images with human judgments from 805 annotators. In line with prior studies [98, 274], we use Kendall's correlation coefficient τ_b for Flickr8k-CF and Kendall's τ_c for the other datasets.

7.2.2 Pairwise Ranking

We focus on the pairwise ranking ability of current captioning metrics, measuring performance on the Pascal-50S dataset.

Dataset. Pascal-50S [271] evaluates captioning metrics using pairwise preference judgments. It consists of 4,000 sentence pairs linked to 1,000 images, each with 50 reference captions. Human judges determine which caption better describes the image, categorizing pairs into human-correct, human with one incorrect caption, human vs. machine-generated, and machine-generated. In this setting, we compute accuracy instead of correlation scores, reporting the averaged results across the four categories.

7.2.3 Sensitivity to Object Hallucinations

We evaluate robustness to hallucinations on the FOIL dataset.

Dataset. FOIL [242] consists of image-caption pairs derived from COCO [158], where captions are intentionally altered by introducing a single erroneous word, which makes the modified caption closely resembles the original but includes a specific mistake. In our evaluation, we compute the percentage of times the original caption gets the highest score, according to each metric.

8

Evolution of Image Captioning Evaluation Metrics

As mentioned in the previous section, leveraging the CLIP architecture and its extensive pre-training to compute captioning scores has become a common and effective strategy, as evidenced by the widespread use of CLIP-S [98]. However, despite the appropriateness of using contrastive-based embedding spaces for evaluation, large-scale models pre-trained on web-collected data also have limitations, due to

This chapter discusses topics from the following papers: S. Sarto *et al.*, "Positive-Augmented Contrastive Learning for Image and Video Captioning Evaluation", CVPR 2023 [229] and S. Sarto *et al.*, "Positive-Augmented Contrastive Learning for Vision-and-Language Evaluation and Training", IJCV 2025 [234] and S. Sarto *et al.*, "BRIDGE: Bridging Gaps in Image Captioning Evaluation with Stronger Visual Cues", ECCV 2024 [231].

the lack in style of captions collected from alt-tags and of the distribution of web-scale images which is not aligned with those on which captioning systems are evaluated. In contrast, while curated datasets are smaller, recent advances in image [219, 226, 225, 79] and text generation [316, 279, 148] enable the creation of high-quality synthetic data with controlled style.

8.1 Advanced Metrics for Image Captioning Evaluation

Following this insight, we propose different metrics that fuses together the advantages of both these scenarios, by leveraging the quality of the pre-training on web-collected data and that of cleaned data, and also regularizing the training by considering additional positive samples hailing from visual and textual generators. Specifically, our proposed metrics, **PAC-S** and **PAC-S++**, are trained via a newly conceived positive-augmented contrastive learning approach, in which pairs of generated images and texts act as additional positives in addition to real images and human-annotated captions taken from a cleaned data source. We demonstrate that the combination of these factors, *i.e.* the usage of a cleaned data source and the pairing with multimodal generated data, when used to fine-tune a large-scale contrastive model, results in an embedding space with significantly higher alignment with the human judgment (Fig. 8.1).

8.1.1 Positive-Augmented Contrastive Learning

We aim to develop an image and video captioning metric based on a shared embedding space where visual data and text can be represented and evaluated. To achieve this, we adopt the dual-encoder architecture introduced by CLIP [215].

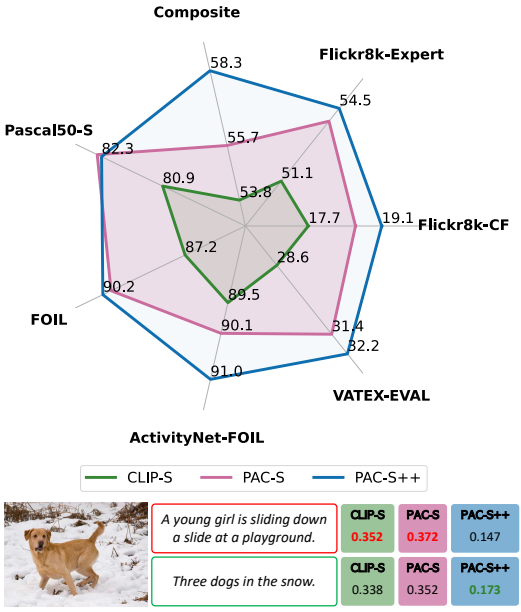


Figure 8.1: Comparison between evaluation scores predicted by our evaluation metric, PAC-S++, in comparison with its original version, PAC-S [229], and CLIP-S [98]. The plot shows the results across different benchmarks, demonstrating the superior performance of PAC-S++ in terms of correlation with human judgment. In the bottom example, the caption highlighted in green is the one preferred by humans.

Contrastive Language-Image Pre-training (CLIP) focuses on learning rich visual and textual representations by understanding the relationships between images and their corresponding textual descriptions. CLIP employs an image encoder $E_v(\cdot)$ (*e.g.* a CNN [96] or a ViT [71]) along with a text encoder $E_t(\cdot)$ (*e.g.* a Transformer model [270]) to obtain visual and textual representations. The multimodal interaction is performed via late fusion by projecting the output of both encoders to the same dimension and then on the ℓ_2 hypersphere via normalization. The visual and the textual inputs can then be compared via cosine similarity.

During the training phase, CLIP utilizes a contrastive objective to encourage similar embeddings for matched image-text pairs and dissimilar em-

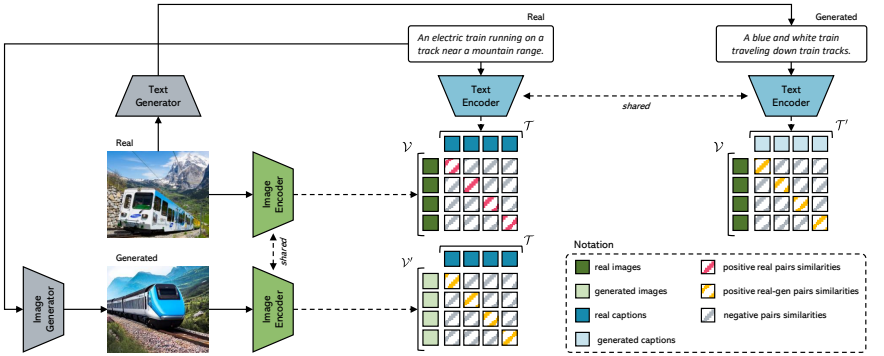


Figure 8.2: Overview of our positive-augmented contrastive learning approach.

beddings for non-matched pairs. In a batch of N image-caption pairs $\{(v_i, t_i)\}_{i=1}^{N}$, CLIP employs the InfoNCE loss [202] that can be written as:

$$\mathcal{L}_{\mathcal{V},\mathcal{T}} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(\text{sim}(v_i,t_i)/\tau)}{\sum_{j=1}^{N}\exp(\text{sim}(v_i,t_j)/\tau)} + \\ -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(\text{sim}(v_i,t_i)/\tau)}{\sum_{j=1}^{N}\exp(\text{sim}(v_j,t_i)/\tau)}. \tag{8.1}$$

Here, the similarity function is defined as:

$$\text{sim}(v,t) = \cos(\text{Norm}(E_v(v)), \text{Norm}(E_t(t))),$$

where $\text{sim}(\cdot)$ is the CLIP-based cosine similarity between visual and textual inputs that are normalized via ℓ_2 normalization, and τ is a temperature parameter to scale the logits. With the symmetrical loss applied to both image and text encoders, the overall loss function $L_{\mathcal{V},\mathcal{T}}$ is computed as the average of the two.

Large-scale contrastive models like CLIP [215] are trained on web-collected image-caption pairs. These provide a large-scale source of supervision for learning scalable low-level and semantic visual and textual features, as testified by their zero-shot classification performance and by their adaptability to different tasks [219, 16, 189, 120]. Nevertheless, it shall

be noted that the textual annotations contained in alt-tags are far from the quality level that a captioning evaluator should look for, and that the distribution of web-scale images might not be properly aligned with those on which image captioning systems are evaluated.

To solve this issue, one might think of learning the metric directly on cleaned data sources. However, recent attempts of learning contrastive-based evaluation metrics on cleaned datasets like COCO [158] perform poorly when compared to traditional metrics, potentially because of the lack of training data [114]. We, therefore, advocate the usage of synthetic generators of both visual and textual data, which showcase sufficiently high quality levels when generating both images and texts, do lack in terms of style, and are controllable in terms of visual distribution.

Positives Generation. In light of these problems, we propose utilizing synthetic generators for both visual and textual data, which showcase sufficiently high-quality levels of generation. Additionally, they are controllable in terms of visual distribution.

Specifically, given a positive image-text pair (v, t), we augment it by generating a synthetic caption t' from v using an image captioning model [148]. Similarly, we generate a synthetic image v' from t via a diffusion-based text-to-image architecture [225], thus building a dataset consisting of tuples of four elements (v, t, v', t'). Next, we train our evaluation model by considering the contrastive relationships between real and generated matching image-caption pairs, as shown in Fig. 8.2.

Formally, given a batch of N real images and their captions, these are processed through the corresponding encoders to obtain the visual $\mathcal{V} = \{E_v(v_i)\}_{i=1}^N$ and textual features $\mathcal{T} = \{E_t(t_i)\}_{i=1}^N$. For generated images and texts, we define $\mathcal{V}' = \{E_v(v_i')\}_{i=1}^N$ and $\mathcal{T}' = \{E_v(t_i')\}_{i=1}^N$. We then define multiple $N \times N$ matrices containing pairwise cosine similarities between the different inputs. We then adopt a symmetric InfoNCE loss, which aims

at maximizing the cosine similarity between the N matching pairs and minimizing those of the $N^2 - N$ non-matching pairs.

In addition to the loss term between real images and real texts $\mathcal{L}_{\mathcal{V},\mathcal{T}}$, defined in Eq. 8.1, we also add symmetrical loss terms between cross-modal generated and real pairs, *i.e.* between generated images and human-annotated texts, and between original images and generated texts. The loss which compares real images $\mathcal{V}$ with respect to generated texts $\mathcal{T}'$ can be defined as:

$$\mathcal{L}_{\mathcal{V},\mathcal{T}'} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(\text{sim}(v_i, t'_i)/\tau)}{\sum_{j=1}^{N}\exp(\text{sim}(v_i, t'_j))/\tau)} + \\ -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(\text{sim}(v_i, t'_i)/\tau)}{\sum_{j=1}^{N}\exp(\text{sim}(v_j, t'_i)/\tau)}. \tag{8.2}$$

In this way, generated items act as additional positive samples for the real matching pairs, thus adding a supervisory signal without being affected by the potential noise present in the data used to train contrastive-based feature extractors like CLIP. In summary, the final loss is a weighted combination of the three loss terms, *i.e.*

$$\mathcal{L} = \mathcal{L}_{\mathcal{V},\mathcal{T}} + \lambda_v\mathcal{L}_{\mathcal{V}',\mathcal{T}} + \lambda_t\mathcal{L}_{\mathcal{V},\mathcal{T}'}, \tag{8.3}$$

where $\mathcal{L}_{\mathcal{V}',\mathcal{T}}$ is the counterpart of Eq. 8.1.1 using generated image and real textual sentences, and the λ values are hyperparameters used to weight the contribution of each loss function.

Captioning evaluation score for images. After training with positive-augmented contrastive learning, we employ two evaluation scores for evaluating images in both a reference-free and a reference-based setting. For evaluating images, we adopt the equation proposed by [98] as our reference-free score:

$$\text{Score}(v, t) = w \cdot \max(\text{sim}(v, t), 0), \tag{8.4}$$

that given an image-text pair (v, t) defines the evaluation score as a linear projection of thresholded cosine similarities. To incorporate reference ground-truth captions into the evaluation process, following [98], we first calculate the representation of each reference caption using our positive-augmented trained textual encoder. Then, we compute the harmonic mean between the reference-free score, defined in Eq. 8.4, and the maximum cosine similarity between the candidate caption and all reference captions. Formally, given a set of M reference captions $R = \{r^j\}_{j=1}^{M}$, the score is computed as:

$$\begin{aligned} &\text{Ref-Score}(v, t, R) = \text{H-Mean}\left(\text{Score}(v, t), \text{top-r}(t)\right) \\ &\text{where} \quad \text{top-r}(t) = \max\left(0, \max_{r \in R}(\text{sim}(t, r))\right). \end{aligned} \tag{8.5}$$

Here, $\text{Score}(\cdot)$ represents the reference-free score defined in Eq. 8.4, and $\text{H-Mean}(\cdot)$ indicates the harmonic mean.

Captioning evaluation score for videos. To test the proposed positive-augmented strategy for evaluating video captions, we extend the above defined metric following the approach of [246]. In this case, matching scores are computed at two granularity levels, *i.e.* a coarse-grained level in which the global representation of the candidate caption is compared with the global representation of the video, and a fine-grained level in which the embeddings of single words are compared to those of single frames.

Specifically, we use the positive-augmented CLIP visual encoder to extract the embeddings of single frames and average-pool them to get the representation of the entire video. Similarly, we employ the corresponding textual encoder to get single tokens and whole caption embeddings. The fine-grained score is then computed by taking the F1-score of pairwise word-frame similarities and TF-IDF [223] weighting, and the coarse-grained score is computed as the similarity between the global video and caption representations. Given a source video V and a candidate caption c, the

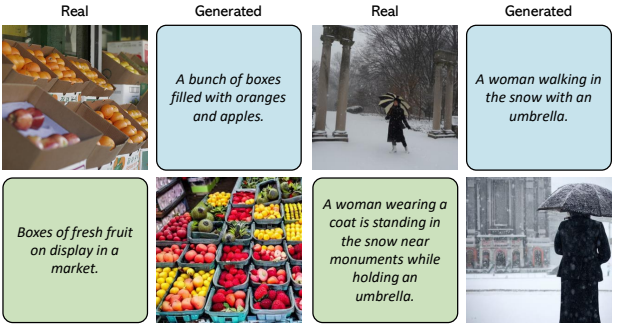


Figure 8.3: Sample real and generated image-text data used for positive-augmented contrastive learning.

overall score is defined as

$$\text{Score}(c, V) = \frac{\text{Score}(c, V)_c + \text{Score}(c, V)_f}{2}, \tag{8.6}$$

where Score_c represents the coarse-grained embedding matching and Score_f stands for the fine-grained similarity. Finally, to include a set of reference captions R, we follow the reference version of the aforementioned approach:

$$\text{Ref-Score}(c, V, r) = \frac{\text{Score}(c, V) + \max_{r \in R} \text{Score}(c, r)}{2}, \tag{8.7}$$

where $\text{Score}(c, r)$ is computed as defined in Eq. 8.6 by using the word-level embeddings of the reference caption.

8.1.2 From PAC-S to PAC-S++: LoRA-Enhanced Contrastive Learning

Following the PAC-S, keeping the same architecture, we enhance it through fine-tuning with low-rank adaptation (LoRA) techniques [103]. By introducing low-rank decompositions into the network parameters, we obtain a fine-tuned visual encoder $E_v(\cdot)$ and text encoder $E_t(\cdot)$. Specifically, we em-

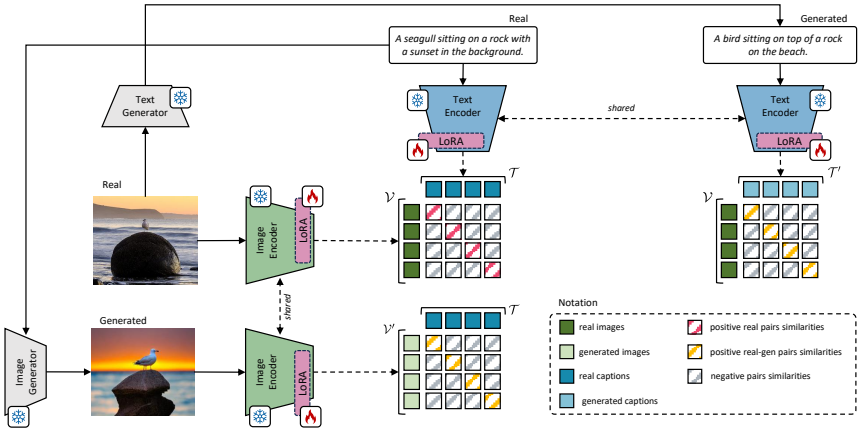


Figure 8.4: Overview of our positive-augmented contrastive learning approach in which both encoders are fine-tuned with low-rank adaptation (LoRA) using additional positive samples generated by text-to-image and image-to-text generative models.

ploy LoRA [103] which preserves the pre-trained model weights while injecting trainable rank decomposition matrices into each layer of the architecture. This approach significantly reduces the overall number of trainable parameters, mitigates the risk of overfitting, and regularizes the training procedure, thus making it a suitable option for the fine-tuning phase.

8.1.3 Implementation Details

Architecture and training details. In continuity with existing literature [98, 123, 246], we use CLIP ViT-B/32 [215] as backbone to encode images (or video frames) and textual sentences. Regarding the PAC-S, we finetune the visual and textual final projections on the COCO dataset [158], which contains more than 120k images annotated with five captions. In particular, we employ the splits introduced by Karpathy *et al.* [117], where 5,000 images are used for validation, 5,000 images are used for test and the rest for training. During finetuning, we use AdamW [177] as optimizer with a learning

rate equal to 0.0001 and a batch size of 256. The λ_v and λ_t values are selected with a grid search, choosing the combination that provides the best average across datasets. Specifically, we set λ_v to 0.05 and λ_t to 0.1, and stop the training stage when the validation loss stops decreasing for 1,500 iterations.

Instead, in PAC-S++, during fine-tuning, we freeze the pre-trained model weights and exploit LoRA [103]. The rank of the decomposition r is set to 4, as it performed favorably in our initial experiments. We use AdamW [177] as optimizer with a learning rate equal to $1 \cdot 10^{-4}$ and a batch size of 256. The λ_v and λ_t values are selected with a grid search, choosing the combination that provides the best average across datasets. Specifically, we set λ_v to 0.1 and λ_t to 0.001, and stop the training stage when the validation loss stops decreasing for 1,500 iterations.

Positive image-text generation. To augment the training set with new positive examples, we use Stable Diffusion* [225] for generating new visual data and the BLIP model [148] for generating new textual descriptions. Specifically, to generate images, we employ the model pre-trained on the English image-text pairs of the LAION-5B dataset [235] and finetuned at a resolution equal to 512×512 on the LAION-Aesthetics subset†, which has been filtered with aesthetic requirements. During generation, we employ the safety checker module to reduce the probability of explicit images and disable the invisible watermarking of the outputs to avoid easy identification of the images as machine-generated. To generate text, instead, we use the ViT-L/14 version‡ of the BLIP model pre-trained on 129M image-text pairs and finetuned on the COCO dataset. After this generation phase, we get a new version of the COCO dataset in which each image is additionally associated with a machine-generated caption and each human-

*https://github.com/CompVis/stable-diffusion
†https://laion.ai/blog/laion-aesthetics/
‡https://github.com/salesforce/BLIP

Table 8.1: Ablation study results of PAC-S++ / RefPAC-S++, using different hyperparameters and synthetic data generators. For each experiment, we report in the last column the averaged improvement compared to the previous version of our metric (*i.e.* PAC-S [229]), reported in the first row.

			Synthetic Data		Flickr8k-Exp	Flickr8k-CF	Composite	VATEX-EVAL	Pascal-50S	FOIL	ActivityNet-FOIL	
LoRA r	λ_v	λ_t	Visual	Textual	Kendall τ_c	Kendall τ_b	Kendall τ_c	Kendall τ_b	Accuracy	Accuracy	Accuracy	$\bar{\Delta}$
-	0.05	0.1	SDv1.5	BLIP	53.9 / 55.5	36.0 / 37.6	55.7 / 57.3	25.1 / 31.4	82.4 / 84.7	89.9 / 93.7	90.1 / 93.5	
Effect of Varying LoRA r												
2	0.1	0.001	SDv1.5	BLIP	54.3 / 55.6	36.9 / 38.0	58.2 / 59.1	27.4 / 32.3	82.2 / 84.4	90.1 / 93.5	91.1 / 93.6	+0.71
4	0.1	0.001	SDv1.5	BLIP	54.5 / 55.7	37.0 / 37.9	58.3 / 59.1	28.1 / 32.2	82.3 / 84.5	90.2 / 93.5	91.0 / 93.4	**+0.78**
8	0.1	0.001	SDv1.5	BLIP	54.5 / 55.6	36.9 / 38.0	58.3 / 59.2	27.2 / 32.1	82.0 / 84.5	90.0 / 93.6	90.7 / 93.5	+0.66
16	0.1	0.001	SDv1.5	BLIP	54.3 / 55.5	37.0 / 37.9	58.5 / 59.3	27.8 / 32.3	81.8 / 84.5	90.1 / 93.6	90.9 / 93.7	+0.74
Effect of Varying λ_v and λ_t												
4	0.001	0.001	SDv1.5	BLIP	54.6 / 55.8	37.1 / 38.0	57.4 / 58.2	27.9 / 32.4	81.9 / 84.5	90.1 / 93.6	90.4 / 93.3	+0.60
4	0.05	0.001	SDv1.5	BLIP	54.7 / 55.8	36.9 / 37.9	58.3 / 59.1	27.5 / 32.0	82.1 / 84.5	90.3 / 93.7	91.0 / 93.7	+0.76
4	0.1	0.001	SDv1.5	BLIP	54.5 / 55.7	37.0 / 37.9	58.3 / 59.1	28.1 / 32.2	82.3 / 84.5	90.2 / 93.5	91.0 / 93.4	**+0.78**
4	0.5	0.001	SDv1.5	BLIP	54.4 / 55.8	36.8 / 37.8	57.9 / 58.9	27.6 / 31.9	81.8 / 84.1	89.8 / 93.6	90.4 / 93.4	+0.53
4	0.1	0.05	SDv1.5	BLIP	54.7 / 55.8	37.0 / 38.0	58.0 / 58.9	27.7 / 32.2	81.9 / 84.4	90.4 / 93.6	90.9 / 93.5	+0.73
4	0.1	0.1	SDv1.5	BLIP	54.6 / 55.8	37.0 / 38.0	57.5 / 58.5	27.7 / 32.0	82.3 / 84.3	89.9 / 93.4	91.3 / 93.3	+0.63
Synthetic Data Contribution												
4	0.1	0.001	-	-	53.9 / 54.9	36.7 / 37.7	57.5 / 58.4	26.7 / 31.9	81.7 / 83.9	89.8 / 93.2	90.1 / 93.2	+0.20
4	0.1	0.001	-	BLIP	54.4 / 55.4	36.8 / 37.8	57.3 / 58.3	26.7 / 32.0	82.1 / 84.4	89.8 / 93.3	90.9 / 93.6	+0.43
4	0.1	0.001	SDv1.5	-	54.2 / 55.2	37.0 / 37.9	57.9 / 58.7	27.0 / 32.1	82.0 / 84.4	89.9 / 93.3	90.4 / 93.7	+0.49
4	0.1	0.001	SDv1.5	BLIP	54.5 / 55.7	37.0 / 37.9	58.3 / 59.1	28.1 / 32.2	82.3 / 84.5	90.2 / 93.5	91.0 / 93.4	**+0.78**
Effect of Varying Synthetic Data Sources												
4	0.1	0.001	SDv1.5	BLIP	54.5 / 55.7	37.0 / 37.9	58.3 / 59.1	28.1 / 32.2	82.3 / 84.5	90.2 / 93.5	91.0 / 93.4	**+0.78**
4	0.1	0.001	SDv1.5	IDEFICS-3	54.2 / 55.4	36.6 / 37.8	58.0 / 58.9	26.9 / 31.9	81.9 / 84.2	89.6 / 93.3	90.9 / 93.8	+0.47
4	0.1	0.001	SDv1.5	LLaMA-3.2	54.6 / 55.7	36.9 / 38.0	57.9 / 58.7	27.1 / 31.9	82.1 / 84.3	90.2 / 93.7	90.9 / 93.8	+0.64
4	0.1	0.001	SDv3.5	BLIP	54.7 / 56.0	36.8 / 37.9	58.1 / 58.9	27.9 / 31.9	82.0 / 84.4	90.2 / 93.4	90.9 / 93.6	+0.71
4	0.1	0.001	FLUX.1	BLIP	54.8 / 56.0	37.0 / 38.0	58.2 / 59.0	28.0 / 32.0	81.8 / 84.4	90.2 / 93.4	91.1 / 93.6	+0.76

annotated caption is instead associated with a newly generated image. Sample image-text data employed for finetuning are shown in Fig. 8.3.

8.1.4 Ablation Studies

Low-Rank Analysis. All the analyses conducted so far employ the PAC-S++ version with low-rank adaptation. In Table 8.1, we investigate the effect of different ranks (*i.e.* 2, 4, 8, 16) across the selected datasets. Overall, the best performance are achieved with a rank of 4, in both reference-free and reference-based settings.

Notably, comparing the results of the previous version of our metric (*i.e.* PAC-S [229], first row), in which only the last visual and textual projections of the model are fine-tuned, the version of our metric with LoRA consistently outperforms the original version regardless of the rank, with the only exception of Pascal-50S and FOIL datasets. For example, employing PAC-S++ with the CLIP ViT-B/32 backbone fine-tuned with LoRA yields

superior results compared to its counterpart without LoRA, achieving 58.3 on the Composite dataset (+2.6) and 28.1 on the VATEX-EVAL dataset (+3.0). This demonstrates the effectiveness of this strategy, even in the context of video settings.

Choice of Hyperparameters. Subsequent to determining the optimal rank dimension as $r = 4$, we also conduct a comprehensive grid search to determine the optimal values of the λ_v and λ_t hyperparameters for our model across multiple datasets. The results are summarized in Table 8.1. While the results show notable variation in performance depending on the dataset, we observe that the configuration corresponding $\lambda_v = 0.1$ and $\lambda_t = 0.001$ consistently yields strong improvements across most datasets. Accordingly, we adopt this setting as our final configuration, which is used in the loss function defined in Eq. 8.3.

Synthetic Data Generator Analysis. To assess the overall contribution of synthetic data and the impact of different text and image generators, we conduct an ablation study, always reported in Table 8.1. We start by analyzing the contribution of each generator. Specifically, we conduct experiments without synthetic data (*i.e.* only using pairs from COCO during fine-tuning), as well as analyze the effects of removing only the textual or the visual positive augmentations. The averaged improvements compared to the previous version of the metric highlight that both visual and textual synthetic data contribute to enhanced performance compared to the version without any positive augmentation. Furthermore, when both augmentations are applied simultaneously, further performance improvement is observed.

As additional analysis, we replace synthetic data generators with more recent models to assess their impact on performance. For text generation, we compare our default setup, which employs BLIP [148], with LLaMA 3.2 [72] and IDEFICS-3 [132] as caption generators. For image generation,

instead, we assess Stable Diffusion v3.5 (SDv3.5)[§] and FLUX [¶] as alternatives to our baseline, Stable Diffusion V1.5 (SDv1.5). Our analysis indicates that the original configuration remains the most effective overall. In particular, on the text side, substituting newer models does not yield noticeable performance improvements. This can be due to the behavior of modern text generators, which tend to produce longer and more verbose captions. CLIP textual encoder often truncates these, thereby limiting their effective contribution. On the image side, although newer generators may offer improved visual quality, results are comparable with the initial version using SDv1.5, confirming the effectiveness of our positive-augmented fine-tuning strategy.

Effect of Changing Backbone. We investigate the impact of changing the backbone architecture by replacing the standard CLIP visual and textual encoders with those from SigLIP [314] and SigLIP2 [268], which differ from CLIP in both training data and training strategies. To ensure a fair comparison, we use the same ViT-L/14 backbone for all three models and we report results in Table 8.2. For each, we evaluate both the reference-free and reference-based variants, applying the standard CLIP-S setup (analogous to a zero-shot setting), as well as PAC-S and our PAC-S++ training strategies. Our results show that CLIP-S with the CLIP ViT-L/14 backbone, while underperforming compared to PAC-S and PAC-S++, still achieves competitive results. However, when using SigLIP or SigLIP2, CLIP-S performance significantly degrades, particularly in terms of human correlation, pairwise ranking, and hallucination detection. This suggests that the zero-shot capabilities of SigLIP-like models are less aligned with the image captioning evaluation tasks. By applying the PAC-S training strategy to SigLIP, we observe substantial improvements, often surpassing even the performance of PAC-S++ with the CLIP ViT-L/14 backbone across several data-

[§] https://huggingface.co/stabilityai/stable-diffusion-3.5-large
[¶] https://huggingface.co/black-forest-labs/FLUX.1-dev

Table 8.2: Human correlation and accuracy results when varying visual and textual backbones. The best results for each backbone are highlighted in bold.

		Flickr8k-Exp		Flickr8k-CF		Composite		Pascal-50S	FOIL
Backbone		Kendall τ_b	Kendall τ_c	Kendall τ_b	Kendall τ_c	Kendall τ_b	Kendall τ_c	Accuracy	Accuracy
CLIP ViT-L/14	CLIP-S	52.6	53.0	35.2	18.2	51.3	55.4	81.7	90.9
	PAC-S	55.1	55.5	36.8	19.0	52.3	56.5	82.2	91.9
	PAC-S++	57.0	57.4	38.5	19.9.	57.3	**62.0**	82.4	93.6
	RefCLIP-S	54.1	54.5	36.5	18.9	51.9	56.1	84.3	94.0
	RefPAC-S	56.7	57.1	37.7	19.5	53.1	57.2	85.0	**95.3**
	RefPAC-S++	57.5	**57.9**	**38.8**	20.1	60.0	61.6	**85.4**	95.1
SigLIP ViT-L/14	CLIP-S	0.49	0.49	-1.1	-0.6	26.1	28.2	62.2	52.5
	PAC-S	58.8	54.0	39.7	**19.9**	54.5	58.9	81.7	92.3
	PAC-S++	61.3	42.7	**48.5**	19.7	**58.8**	61.5	81.8	94.1
	RefCLIP-S	33.3	33.6	18.8	9.7	30.9	33.4	73.0	69.8
	RefPAC-S	59.0	**54.2**	40.0	**20.0**	55.1	59.5	83.1	93.6
	RefPAC-S++	**61.4**	42.8	**48.5**	19.7	**58.8**	**61.6**	**82.7**	**94.4**
SigLIP2 ViT-L/14	CLIP-S	4.6	4.5	1.2	0.6	19.6	21.2	59.3	61.0
	PAC-S	4.5	4.1	1.1	0.5	8.6	8.6	54.9	41.2
	PAC-S++	60.1	47.4	**42.6**	**19.3**	59.1	62.6	81.0	93.4
	RefCLIP-S	4.7	4.6	1.2	0.4	29.9	32.4	72.7	77.3
	RefPAC-S	4.5	4.2	1.0	0.5	9.7	9.7	55.3	43.4
	RefPAC-S++	**60.2**	**47.5**	**42.6**	**19.3**	**59.2**	**62.7**	**81.6**	**93.9**

sets. Further gains are achieved using the PAC-S++ strategy. For instance, with the SigLIP backbone, RefPAC-S++ achieves τ_c scores of 61.4 and 58.8 on the Flickr8k-Expert and Composite datasets, respectively, which are significantly higher than the CLIP-based version. These results highlight the importance of both the model architecture and the fine-tuning strategy in achieving robust performance across captioning evaluation benchmarks.

8.2 Metric with Stronger Visual Cues

As already explained in previous sections, the objective of image captioning is to produce natural language descriptions conditioned on input images, that closely resemble human language and align to human intentions. As such, the captioning task involves the recognition and understanding of the visual content of the image, including fine-grained elements such as objects, attributes, and their relationships. To evaluate the quality of a generated caption, various metrics have been proposed that take an image—and, optionally, human-written reference captions—into account.

However, it is important to note that obtaining these reference captions can often be challenging and expensive, adding complexity to the evaluation process. Despite the recent advancements in captioning capabilities, standard automatic evaluation metrics have mainly relied on translation metrics [207, 154, 15] or text-only ones [271, 5, 318] which often fall short in capturing aspects such as grammatical correctness, semantic relevance, and specificity. These limitations are worsened by the limited coverage of image content in available references, resulting in inaccurate penalties when generated captions accurately describe novel elements not mentioned in the references.

In response to these limitations, advanced metrics aligning visual and textual data have emerged [136, 135, 98, 123]. Notably, recent metrics leverage the CLIP embedding space [215], which shows a strong correlation with human judgment. Despite the effectiveness of contrastive-based embedding spaces, metrics based on dual-encoder architectures tend to focus on global alignment between an image and its caption, often ignoring fine-grained details or penalizing hallucinations.

8.2.1 A Learnable Reference-free Metric

Following this insight, we introduce **BRIDGE**, a novel learnable and *reference-free* image captioning metric that enhances the alignment of more fine-grained visual features. Specifically, our model provides a pre-trained dual-encoder architecture with a mapping module designed to effectively exploit visual cues. This is done by internally creating multimodal pseudo-captions, containing both textual and dense visual features. The process for building these pseudo-captions involves the creation of a template caption, which focuses on the syntactical structure of the scene, and a mapping module. The latter refines the template caption by enriching it with more fine-grained visual features about the subjects depicted in the

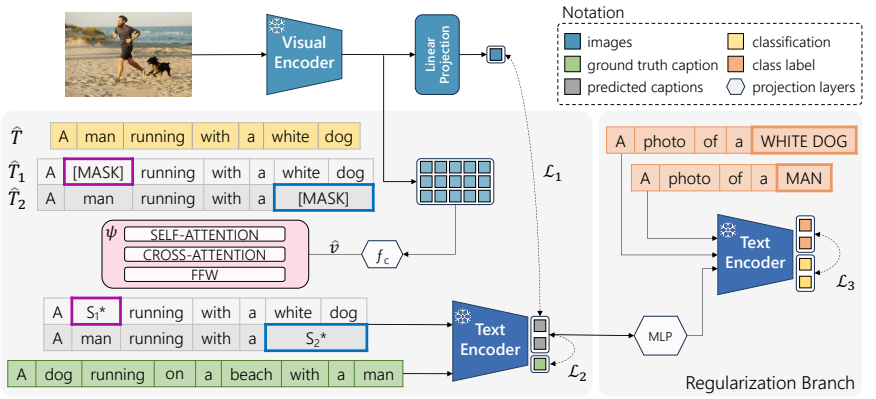


Figure 8.5: Overview of the BRIDGE evaluation approach. Starting from a template caption, a mapping network augments it with dense visual features, obtaining a pseudo-caption which is then used for computing image-text similarities.

8

image. Subsequently, the overall model is trained with a combination of contrastive losses which promote multimodal alignment. An overview of our model is depicted in Fig. 8.5.

Unlike CLIP-Score, our approach does not rely exclusively on global image descriptors for evaluating image-text alignments. Instead, we focus on employing stronger visual information. To do so, we draw inspiration from the Pic2Word approach [227] and represent the input image through a multimodal pseudo-caption, an embedding representation that contains stronger visual elements.

Preliminaries. Our approach relies on CLIP (Contrastive Language–Image Pre-training) [215], a powerful vision and language model designed to align images and corresponding text captions within a shared embedding space. For a given input image I, the image encoder E_V extracts the visual information $v = E_V(I) \in \mathbb{R}^d$. On the textual side, an input caption T is tokenized and a textual representation is obtained by passing it through the textual encoder E_T, obtaining $t = E_T(T) \in \mathbb{R}^d$. Once the textual and visual features, t and v respectively, are projected in a common space, visual and

textual inputs can be compared via cosine similarity.

The relationships learned by CLIP can be exploited to build an image captioning evaluator. In CLIP-Score [98] the authors directly compare candidate captions and images in the embedding space and show that this achieves a good correlation with human judgments. In detail, to assess the quality of a candidate generation, they feed both the image and the candidate caption through their respective feature extractors, and compute the cosine similarity of the resultant embeddings:

$$\text{CLIP-Score}(I, T) = w \cdot \max(\cos(v, t), 0), \tag{8.8}$$

where w is a rescaling factor employed to stretch the score distribution while ensuring the ranking results remain unchanged.

Building Template Captions. In order to create multimodal pseudo-captions, we first build *template captions* for a given input image. These are skeletal textual representations of the image, obtained by masking out all the relevant textual concepts from the descriptions generated by a captioner. Through these template captions, we aim to provide the model only with a templated textual structure which can then be filled with more fine-grained visual features. In particular, given an automatically generated caption describing the input image, such as `A man running with a white dog', we remove the main subjects within the caption (*e.g.* `man' and `white dog') and mask them with a [MASK] token. This will allow the model to fill in these gaps by incorporating more fine-grained features from the image encoder. Since a primary subject might be described by words other than just its corresponding nouns (*e.g.* adjectives), we utilize noun chunks. Fig. 8.6 reports template captions and corresponding noun chunks.

Given a sentence containing N noun chunks, we independently encode them through the mapping network. To this aim, we replicate the template

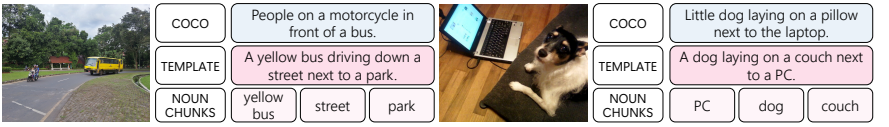


Figure 8.6: COCO captions with template captions and associated noun chunks.

caption as many times as the number of noun chunks and mask a different noun chunk in each of the replicas. We thus obtain N different versions of the template caption, each one masking a single noun chunk, for instance

```
[`A [MASK] running with a white dog',
 `A man running with a [MASK]'].
```

8

Mapping with Fine-Grained Visual Features. The above-described masked replicas are then fed to the mapping module ψ. Specifically, our approach exploits the visual information extracted from the visual encoder E_V to enrich the replicas with the informative content of the image I. To get a more fine-grained representation of the image, we directly take the grid of features from the last layer, $\widehat{v}$. For instance, in the case of a ViT-B/32 backbone, this will have a shape of $50 \times d$, where d is the dimensionality of the last embedding of the network.

The mapping network is implemented as a stack of Transformer [270] encoder layers interleaved with cross-attention layers. Its role is to refine each template captions with visual information. Since each template caption is processed independently, the mapping module returns a set of sequences, each with the same length as the corresponding input template caption. From the output of the mapping module, we keep only the predictions for the masked tokens in each template caption and copy them back into the original templates.

Therefore, by providing a masked input template in the form $\hat{T}_i = [w_1, ..., w_{j-1}, \mathsf{MASK}, w_{j+1}, ..., w_T]$, where $\{w_j\}_j$ represent original tokens from

the input caption, we obtain $\hat{T_i}^* = \left[w_1, ..., w_{j-1}, \psi(\hat{T_i})_j, w_{j+1}, ..., w_T\right]$, where $\psi(\hat{T_i}, \widehat{v})_j$ represents the output of the mapping network at the position corresponding to the masked input token position. In the case of noun chunks consisting of more than one token, multiple consecutive tokens are replaced with the corresponding outputs from the mapping network. By injecting the outputs of the mapping token into the initial template caption, we effectively complete the original templates with visually enriched vectors. Notably, these newly generated pseudo-captions combine word sequences from the template captions with *dense vectors* obtained by the mapping module. Consequently, they cannot be decoded as standard captions. As a last step, the obtained pseudo-captions are fed into the pre-trained CLIP language encoder.

Training Protocol. To train our mapping network, the loss is defined as a weighted version of the symmetric InfoNCE loss [202], where positive and negative items are weighted according to the number of noun chunks in each caption.

Specifically, given a batch in the form $\mathcal{B} = \{(I_i, T_i)\}_{i=1}^{N}$, where I_i and T_i represent image-caption pairs, each image I_i is expanded in N_i multi-modal pseudo-captions as outlined above, where N_i is the number of noun chunks in caption T_i. Further, let $\hat{t_{ij}}^*$ represent the embedding vector of the j-th pseudo-caption derived from the i-th image, v_i the embedding vector of the i-th image and t_i the embedding vector of the i-th ground-truth caption. Finally, let M be the total number of noun chunks in the mini-batch, *i.e.* $M = \sum_{i=1}^{N} N_i$.

The first loss, denoted as $\mathcal{L}_1$, tries to align the pseudo-captions $\hat{t_{ij}}^*$ with the global visual features of the corresponding images v_i. In addition to this loss term, we define a second loss component $\mathcal{L}_2$ that promotes the alignment between pseudo-captions $\hat{t_{ij}}^*$ and the textual feature vector of the ground-truth caption t_i corresponding to the input image. This ensure

that pseduo-captions are aligned also on a textual space, in addition to being aligned in the image space.

Regularization Branch. With the aforementioned loss terms, our objective is encouraging an association between each pseudo-caption and its corresponding image and caption. However, it is also important to differentiate each pseudo-caption from the others of the same image. To achieve this, we define a regularization loss which promotes a precise alignment between each pseudo-caption and the corresponding noun chunk.

First, we create prompts like "`a photo of a` <`NOUNCHUNK`>" and encode them with the text encoder E_T. In parallel, each pseudo-caption is fed into a dedicated multilayer perceptron (MLP) projection, which consist of two linear layers with a ReLU activation in between. Formally, the branch is defined as

$$\mathcal{C}(x) = \text{Linear}(\text{ReLU}(\text{Linear}(x))). \tag{8.9}$$

Since our goal is to emphasize each pseudo-caption's alignment with its corresponding noun chunk, we employ a regular contrastive loss $\mathcal{L}_3$ between the prompts mentioned earlier and the outputs of the projection branch.

Finally, the overall loss function we train BRIDGE is defined as a weighted summation of two aforementioned losses, plus the regularization loss, as

$$\mathcal{L} = \lambda_1 \mathcal{L}_1 + \lambda_2 \mathcal{L}_2 + \lambda_3 \mathcal{L}_r. \tag{8.10}$$

Inference and Score Computation. At inference time, given an image-candidate caption pair (I, T), we extract all pseudo-captions from I using our mapping network. Subsequently, we compute the mean pseudo-caption embedding as $t^* = \frac{1}{N}\hat{t_i}^*$, where $\hat{t_i}^*$ indicates the i-th pseudo-caption extracted from I and N here indicates the overall number of pseudo-captions associated with I.

At that point, given the visual embedding v of the image and the embedding of the candidate caption t, the matching score between I and T is defined as

$$\text{BRIDGE}(I,T) = 0.5 \cdot [\text{CLIP-Score}(I,T) + w \cdot \max(\cos(t^*,t),0)], \tag{8.11}$$

where $\cos$ indicates the cosine similarity and w is a constant scaling factor.

8.2.2 Implementation Details

Architecture and Training Details. Building upon prior research [98, 123, 246], we use either CLIP [215] ViT-B/32 or ViT-L/14 as backbone for the visual and textual encoder. The mapping module is composed of two Transformer layers and is trained on the COCO dataset [158], which contains more than 120k images annotated with five captions. In particular, we employ the splits introduced by Karpathy *et al.* [117], where 5,000 images are used for both validation and testing and the rest for training. To map the grid visual features to an embedding space of dimension 512, we employ a simple linear projection. For the regularization branch, we utilize a two-layer multi-layer perceptron.

During training, we use AdamW [177] as optimizer with a learning rate equal to 0.0001 and a batch size of 256. The λ_1, λ_2, and λ_3 values are selected with a grid search, choosing the combination that provides the best validation loss. Specifically, we set both λ_1 and λ_3 to 0.01, while λ_2 is set to 1.0. The training stage lasts around one day on a single A100 GPU.

Template Caption Generation. The template captions used as input for the mapping module are generated using the BLIP model [148]. In particular, we use the ViT-L/14 version pre-trained on 129M image-text pairs and finetuned on the COCO dataset. After this generation phase, the primary subjects of the template sentences are extracted by using the NLTK

Table 8.3: Ablation study results. US indicates the number of pseudo tokens.

	Flickr8k-Expert			Flickr8k-CF			Pascal-50S
	Kend. τ_b	Kend. τ_c	Spear. ρ	Kend. τ_b	Kend. τ_c	Spear. ρ	Accuracy
Architectural Components							
w/o mapping module (*i.e.* MLP)	53.1	53.5	65.3	35.5	18.3	43.5	81.9
w/o template captions	53.7	54.1	66.0	35.5	18.4	43.6	82.5
w/o regularization branch	54.1	54.5	66.5	35.7	18.5	43.6	82.7
Score Formulation							
w/o textual similarity	51.1	51.2	63.0	34.4	17.7	30.5	80.9
w/o visual similarity	53.8	54.2	66.0	35.4	18.3	43.7	81.9
Pseudo-token Size							
w/ US $= 1$	54.1	54.5	66.4	35.1	17.1	30.3	82.6
w/ US $= 2$	54.1	54.5	66.4	**36.1**	**18.7**	44.4	**82.8**
w/ US $= 4$	54.3	54.7	66.6	35.4	18.3	43.8	82.4
w/ US $= 8$	54.0	54.4	66.3	35.9	18.6	**44.5**	81.7
BRIDGE (US $= 3$)	**54.4**	**54.8**	**67.7**	**36.1**	**18.7**	**44.5**	82.6

library [19]. During training, two noun chunks are randomly chosen from the set identified during the extraction step. In the evaluation phase, otherwise, all identified noun chunks are included.

8.2.3 Ablation Studies

To evaluate the effectiveness of our metric, we start by analyzing variations of our main architectural components. Then, we assess the impact of caption templates in our score formulation. All these experiments are performed using CLIP ViT-B/32 as backbone and reported in Table 8.3.

Contribution of Architectural Components. We first investigate the performance of the most straightforward implementation of a mapping module, structured as a two-layer MLP following [227]. We also validate the importance of the template captions through a model variant in which a set of learnable tokens S^* serves as input for the mapping module, without relying on template captions. In both variants, given the absence of template captions, we construct a template such as `` `a photo of S*' `` and extract its features using the CLIP text encoder. In the Table, it can be seen that, regardless of any architectural changes, it is important to provide a

simple sentence structure to the mapping module to achieve competitive performance.

In addition to these baselines, we devise a variant to analyze the contribution of the regularization branch. In this setting, we employ template captions as input for the mapping module, resulting in a substantial improvement of +1.0 Kendall τ_b and +0.8 accuracy points compared to the MLP variant, respectively on the Flickr8k-Expert and on the Pascal-50S dataset. When introducing the regularization branch (*i.e.* the complete BRIDGE architecture), further enhancements can be observed especially on the Flickr8k-CF with an improvement of +0.4 points in terms of the Kendall τ_b score.

We also emphasize the importance of each component in our score formulation. Specifically, we present correlation results when employing only the visual similarity within our architecture, which is the original CLIP-Score formulation. As observed, performance drops drastically when relying only on visual information. A less significant drop is observed when employing only textual similarity.

As an additional analysis, we report the effect of changing the number of the pseudo tokens for each noun chunk, denoting it as unit size (US). Specifically, we compute the scores employing US $= 1, 2, 3, 4, 8$. From the results, it can be seen that US $= 3$ generally leads to the best performance across nearly all evaluation metrics. This configuration is used in all reported experiments.

Analysis on Caption Templates. We analyze the effect of changing the initial template captions for our model. Specifically, we employ template captions generated by a conventional Transformer-based captioner that uses CLIP features as input and is trained only on the COCO dataset, as well as those generated by BLIP [148], and BLIP-2 [147]. Notably, we select template captions of different quality based on both standard metric evaluations and correlations with human judgment. To assess the quality of the

Table 8.4: Impact of different template captions.

	Expert	CF	Pascal-50S
	Kend. τ_b	Kend. τ_b	Acc.
Transformer Templates			
w/o mapping module	46.1	31.6	80.4
BRIDGE	**54.0** (+7.9)	**35.9** (+4.3)	**82.7** (+2.3)
BLIP Templates			
w/o mapping module	48.6	33.8	81.0
BRIDGE	**54.4** (+5.8)	**36.1** (+2.3)	**82.6** (+1.6)
BLIP-2 Templates			
w/o mapping module	49.4	34.2	82.4
BRIDGE	**54.4** (+5.0)	**36.2** (+2.0)	**82.9** (+0.5)

raw generated templates, we observe that the CIDEr score of these models on the COCO test set is equal to 114.2, 131.4, and 145.8 respectively for the standard Transformer model, BLIP, and BLIP-2. Note that all models were trained with cross-entropy loss only. Results on the Flickr8k-Expert, Flickr8k-CF, and Pascal-50S datasets are reported in Table 8.4. To qualitatively validate the generated templates, we include sample captions generated by the three models compared to a ground-truth caption from the COCO test set. Specifically, captions generated by BLIP-2 are generally more detailed and effectively describe the visual content of the input image compared to those generated by BLIP and, notably, the standard Transformer model.

For each caption template source, we compute the correlation scores when the mapping module is disabled and the captions are directly fed to the text encoder. We compare it with our standard BRIDGE score, considering the different caption templates. Across all datasets, it is evident that directly using the caption templates as input to the text encoder leads to poor performance. This highlights the intended flexibility of template captions as skeletal representations, allowing the model to enhance them with fine-grained visual features.

In fact, starting from these simple template captions and following our approach, we achieve improvements of +5.8 and +2.3 Kendall τ_b points

and +1.6 accuracy points, respectively, on the Flickr8k-Expert, Flickr8k-CF, and Pascal50-S datasets when using BLIP caption templates. The overall best results are with captions from the BLIP-2 model, confirming that better templates can indeed lead to improved results. However, even when using lower-quality captions, the final correlation results are very close to those obtained with higher-quality captions. This highlights the robustness of our metric to caption templates of varying quality and that it is not necessary to rely on captions generated by large-scale captioners to achieve strong correlation scores.

8.3 Comparison of Metrics

Having introduced the main families of captioning evaluation metrics – ranging from traditional reference-based measures, to CLIP-based and learnable metrics such as PAC-S and PAC-S++, and finally to reference-free approaches like BRIDGE -– this section presents a comprehensive analysis of captioning evaluation metrics through quantitative experiments. The evaluation spans multiple dimensions, including correlation with human judgments, ranking accuracy, and sensitivity to object hallucinations. Experiments are conducted on diverse datasets, with results summarized in Table 8.5.

Correlation with Human Judgment. We analyze the correlation of metrics with human judgment, highlighting their varying behaviours. Experimental results are reported on standard captioning evaluation datasets, including Flickr8k-Expert, Flickr8k-CF, and Composite, along with the more recent Polaris and Nebula datasets.

Among rule-based methods, CIDEr demonstrates the highest performance across most datasets, with the exception of Flickr8k-Expert and Composite, where SPICE outperforms it by +1 and +2.6 points, respectively.

Table 8.5: A quantitative comparison between metrics, reporting correlation scores on Flickr8k-Expert, Flickr8k-CF, Polaris, Nebula, and Composite, along with accuracy on Pascal-50S and FOIL. The best scores within each category are in bold, overall best scores are underlined.

		Inputs		Flickr8k-Expert	Flickr8k-CF	Polaris	Nebula	Composite	Pascal-50S	FOIL
	External Models	Image	Refs	Kendall τ_c	Kendall τ_b	Kendall τ_c	Kendall τ_c	Kendall τ_c	Accuracy	Accuracy
Rule-based										
BLEU-4 [207]	-		✓	30.8	16.9	46.3	40.4	30.6	74.0	66.2
ROUGE [154]	-		✓	32.3	19.9	46.3	42.6	32.4	78.0	54.6
METEOR [15]	-		✓	41.8	22.2	51.2	46.8	38.9	**81.1**	70.1
CIDEr [271]	-		✓	43.9	**24.6**	**52.1**	**48.1**	37.7	80.1	**85.7**
SPICE [5]	-		✓	**44.9**	24.4	51.0	44.0	**40.3**	76.7	75.5
Learnable Unsupervised										
FAIEr [276]	-	✓	✓	-	-	-	-	-	81.4	-
TIGEr [114]	SCAN	✓	✓	49.3	-	-	-	45.4	80.7	-
UMIC [135]	$\text{UNITER}_{\text{BASE}}$	✓		46.8	-	49.8	-	56.1	85.1	-
BERT-S [318]	$\text{BERT}_{\text{BASE}}$		✓	39.2	22.8	51.6	47.0	30.1	80.1	88.6
BERT-S++ [303]	$\text{BERT}_{\text{BASE}}$		✓	46.7	-	-	-	44.9	80.1	-
ViLBERT-S [136]	$\text{ViLBERT}_{\text{BASE}}$	✓	✓	50.1	-	-	-	52.4	79.6	-
MID [123]	CLIP ViT-B	✓	✓	54.9	37.3	51.3	**51.3**	-	85.2	90.5
CLIP-S [98]	CLIP ViT-B	✓		51.2	34.4	52.3	46.9	53.8	80.9	87.2
RefCLIP-S [98]	CLIP ViT-B	✓	✓	53.0	36.4	52.3	46.9	55.4	83.3	91.0
PAC-S [229]	CLIP ViT-B	✓		54.3	36.0	52.3	47.2	55.7	82.4	89.9
RefPAC-S [229]	CLIP ViT-B	✓	✓	55.9	37.6	55.2	50.6	57.3	84.7	93.7
PAC-S++ [234]	CLIP ViT-B	✓		54.5	37.0	52.4	-	58.3	82.3	90.2
RefPAC-S++ [234]	CLIP ViT-B	✓	✓	55.7	37.9	54.8	-	59.1	84.5	93.5
InfoMetIC [102]	CLIP ViT-B	✓		55.5	36.6	-	-	59.3	**86.5**	-
CLIP-S [98]	CLIP ViT-L	✓		52.6	35.2	53.2	-	55.4	81.7	90.9
RefCLIP-S [98]	CLIP ViT-L	✓	✓	54.4	36.5	55.5	-	-	85.0	94.9
PAC-S [229]	CLIP ViT-L	✓		55.5	36.8	52.4	47.9	56.5	82.2	91.9
RefPAC-S [229]	CLIP ViT-L	✓	✓	57.1	37.7	55.5	50.4	-	85.0	**95.3**
PAC-S++ [234]	CLIP ViT-L	✓		57.4	38.5	53.6	-	**62.0**	82.4	-
RefPAC-S++ [234]	CLIP ViT-L	✓	✓	**57.9**	**38.8**	**55.6**	-	61.6	84.7	-
Learnable Supervised										
Polos [274]	CLIP ViT-B	✓	✓	56.4	37.8	**57.8**	53.9	57.6	86.5	93.3
DENEB [190]	CLIP ViT-B	✓	✓	56.5	38.0	-	54.1	57.9	87.0	95.1
DENEB [190]	CLIP ViT-L	✓	✓	**56.8**	**38.3**	-	**54.3**	**58.2**	**87.8**	**95.4**
Learnable Fine-grained Oriented										
BRIDGE [231]	CLIP ViT-B	✓		54.8	36.1	-	-	55.0	82.6	91.5
BRIDGE [231]	CLIP ViT-L	✓		55.8	36.3	-	-	57.2	82.9	93.0
HICE-S [312]	SAM+Alpha-CLIP ViT-L	✓		56.4	37.2	-	-	57.9	86.1	93.1
RefHICE-S [312]	SAM+Alpha-CLIP ViT-L	✓	✓	57.7	**38.2**	-	-	58.7	**87.3**	**96.4**
HiFi-S [300]	SAM+BLIP-2	✓		**58.4**	-	-	-	**65.8**	83.0	-
LLMs-based										
CLAIR [35]	GPT-3.5		✓	48.3	-	-	**52.7**	61.0	78.7	81.4
FLEUR [138]	LLaVA v1.5-13B	✓		53.0	38.6	-	-	63.5	83.2	96.8
RefFLEUR [138]	LLaVA v1.5-13B	✓	✓	**51.9**	**38.8**	-	-	**64.2**	**85.5**	**97.3**

Within the learnable metrics, RefPAC-S++ (ViT-L) achieves the best results on Flickr8k-Expert, Flickr8k-CF, and Polaris, while ViLBERT-S yields the highest scores among metrics evaluated on the Nebula dataset. Notably, reference-based metrics generally outperform their reference-free counterparts. However, among the reference-free methods, InfoMetIC and PAC-S variants demonstrate superior performance, emphasizing the importance of refining large pre-trained backbones in the absence of reference captions. When scaling the backbone size, PAC-S variants maintain superior performance, demonstrating their robustness and scalability. In a

supervised setting, comparisons are more complex due to differences in backbone sizes. Overall, DENEB (ViT-L) achieves the highest results across multiple datasets. Interestingly, despite leveraging a supervised approach and incorporating an additional learnable module, DENEB is outperformed by RefPAC-S++ (ViT-L), which is trained in an unsupervised manner. This highlights the effectiveness of leveraging CLIP pre-training and enhancing it with regularization through additional generated visual-textual pairs. For fine-grained evaluation, the HiFi metric, employing a hierarchical parsing graph, achieves the best results in a reference-free setting, surpassing HICE-S by +7.9 and BRIDGE (ViT-L) by +8.6 on the Composite dataset. On the Flickr8k-CF dataset, the best performance is instead achieved by RefHICE-S. Among LLM-based metrics, FLEUR performs best, highlighting the importance of leveraging a multimodal approach that incorporates input images, unlike the CLAIR metric which relies solely on reference captions. This underscores the critical role of visual information in the captioning task.

Overall, no single metric consistently outperforms all others across datasets. Interestingly, on the Flickr8k-CF dataset, comparable results are achieved using both RefPAC-S++ and the MLLM-based metric FLEUR. This comparison is particularly noteworthy, as the FLEUR metric, based on a LLaVA model with 13B parameters, achieves similar performance to metrics that utilize a smaller CLIP model with around 400M parameters. This highlights that for specific tasks like captioning evaluation, refining a pre-trained embedding space may be more effective than relying on a larger, multi-task embedding. Moreover, there is a significant performance gap between rule-based metrics and newer approaches, highlighting the importance of leveraging input images and the advantages of large-scale pre-training in achieving superior performance.

Pairwise Ranking. We focus on the pairwise ranking ability of current captioning metrics, measuring performance on the Pascal-50S dataset.

Among rule-based metrics, METEOR achieves the highest accuracy, outperforming CIDEr by +1 point. For learnable unsupervised metrics, the results deviate from trends seen in correlation-based evaluations, where larger backbones and reference-based metrics typically excel. In this case, InfoMetIC, a reference-free metric using the ViT-B backbone, emerges as the best among all metrics. Its ability to identify incorrect semantic words and unmentioned visual regions proves advantageous for this task. Conversely, for other learnable and LLM-based metrics, the results closely align with those observed in human correlation evaluations, with the highest overall accuracy achieved by DENEB (ViT-L).

Sensitivity to Object Hallucinations. We evaluate robustness to hallucinations on the FOIL dataset. In this task, CIDEr outperforms all other rule-based metrics by a substantial margin, with a gain of around +10 points. Among learnable metrics, RefHICE achieves the highest accuracy, followed by DENEB and RefPAC-S, both using the ViT-L backbone. Reference-free versions of HICE and PAC-S experience a significant drop, indicating the need for reference captions to detect hallucinated objects. The results also highlight the advantages of stronger backbones. For instance, PAC-S (ViT-L) outperforms PAC-S (ViT-B) by +2 points, demonstrating the impact of model architecture. Notably, rule-based metrics like CIDEr are significantly surpassed by modern multimodal approaches, with a performance gap of approximately 10 points compared to RefPAC-S (ViT-L) and nearly 12 points against the LLM-based RefFLEUR. This disparity highlights the limitations of rule-based approaches in effectively capturing the nuanced semantics of image captions, particularly in detecting subtle errors such as hallucinations. Metrics like RefPAC-S and RefFLEUR benefit not only from powerful backbones but also from their ability to align text with visual content, allowing them to detect discrepancies much more accurately than traditional metrics. These results empathize the growing need for evaluation metrics

that assess not just linguistic fluency but also semantic accuracy and visual relevance through advanced multimodal understanding.

8.4 Metric-Guided Fine-Tuning for Image Captioning

Beyond evaluation, captioning metrics can also be leveraged as training signals to improve the quality of generated descriptions. In this section, we show that our metric can be effectively used during the fine-tuning stage of image captioning models, leading to semantically richer descriptions without compromising grammatical correctness. Reinforcement learning (RL) has become a common strategy for fine-tuning captioning systems, where models are treated as agents and optimized to maximize the expected value of a reward function. Building on prior work that employs metrics such as CIDEr as optimization objectives, we investigate the use of PAC-S++ as a reward signal for fine-tuning captioning models, demonstrating its effectiveness in guiding generation toward more informative and semantically accurate captions.

Revisiting Standard Self-Critical Sequence Training. Self-Critical Sequence Training (SCST) [222] for image captioning is a two-step training methodology which (i) pre-trains a captioning network f_θ using a time-wise cross-entropy loss, and (ii) fine-tunes the same network by maximizing the CIDEr score [271] on the training set using reinforcement learning.

While SCST effectively improves the quality of generated captions over single-stage cross-entropy training, it has been shown to introduce a bias towards generating captions that conform to the "average" description of the training set [43]. This results in less descriptive, semantically rich, and discriminative captions. Moreover, these problems are amplified by unin-

formative image-caption pairs in captioning datasets, and by the reliance on the CIDEr metric as a reward signal, which has been questioned due to its relatively low correlation with human judgments and dependence on reference captions.

Recent attempts to replace CIDEr with semantic embedding-based metrics, like CLIP-S [55], have led to excessively long captions that, while detailed, may contain errors, *e.g.* repetitions, due to the noisy nature of the large-scale data used for CLIP pre-training.

8.4.1 Learnable Metric for RL-based Captioning Fine-tuning

By combining pre-training on both web-collected and cleaned data, our metric, PAC-S++, addresses many of the issues associated with CIDEr and CLIP-S. As demonstrated in our previous work [229], this approach results in a more refined embedding space and stronger correlations with human judgments. Consequently, we propose using PAC-S++ to improve the training of image captioning models.

First Training Stage (Cross-Entropy Loss). Formally, we can assume that f_θ is an autoregressive Transformer-based captioning network [270], where θ represents the trainable parameters, which takes as input an image v, described with a sequence of R visual features $\{e^i\}_{i=1}^R$, and a ground-truth sequence t of words within the vocabulary. Notably, $\{e^i\}_{i=1}^R$ represents the grid of features before the last layer normalization and linear projection W in the CLIP architecture:

$$E_v(v) = W \cdot \text{LayerNorm}(e^1, \ldots, e^R). \tag{8.12}$$

During the first training stage, the network is conditioned on all visual features and ground-truth tokens of length T up to the current prediction step

k. The model f_θ is optimized with cross-entropy loss (*i.e. teacher forcing*):

$$\mathcal{L}_{\text{XE}}(v, t; \theta) = -\sum_{k=1}^{T} \log f_\theta(t^k | t^1, \ldots, t^{k-1}, e^1, \ldots, e^R), \tag{8.13}$$

where f_θ outputs a categorical probability distribution over the vocabulary.

Second Training Stage (SCST). In the second training stage, designed to enhance the capabilities of the model, the network is conditioned on the input image and previously generated words. The output of the captioning model f_θ is a generated caption $\hat{t} = \{\hat{t}^i\}_{i=1}^{S}$ of length S, where each word is sampled from the output probability distribution generated at the prior time step k. For instance, the k-th token $\hat{t}^k$ is chosen as the one that maximizes the model probability distribution over possible tokens:

$$\hat{t}^k = \text{argmax}\, f_\theta(\hat{t^k} | \hat{t}^{k-1}, ..., \hat{t}^1, e^1, ..., e^R). \tag{8.14}$$

Given the caption $\hat{t}$ and the image v, PAC-S++ score is computed and used as the reward $r(\cdot)$ for guiding a policy-gradient RL update step:

$$r(v, \hat{t}) = \text{Score}(v, \hat{t}), \tag{8.15}$$

where $\text{Score}(\cdot)$ is computed as in Eq. 8.4. Additionally, we also employ Eq. 8.5 to compute the reward. To mitigate the variance in the reward signal, a baseline value b, computed as the average of the reward of all descriptions generated for v, is subtracted from the reward.

The parameters are optimized using gradient-based methods with the SCST loss function [222]. Beam search is employed to explore multiple possible sequences. Formally,

$$\nabla_\theta \mathcal{L}_{\text{SCST}}(v, \hat{t}; \theta) = -\frac{1}{l} \sum_{i=1}^{l} (r(v, \hat{t}^i) - b) \nabla_\theta \log f_\theta(\hat{t}^i)), \tag{8.16}$$

where l is the beam size and t_i the i-th sentence in the beam.

8.4.2 Effectiveness of metrics as reward

We evaluate the effectiveness of the proposed PAC-S++ metric when employed as reward for fine-tuning a captioning model, using the fine-tuning strategy described in Sec. 8.4. In this setting, we compare our metric in both its reference-free and reference-based versions respectively against CLIP-S and RefCLIP-S. For completeness, we also report the results of the model trained with cross-entropy loss only (*i.e.* without reinforcement learning) and using the standard CIDEr score as reward. To evaluate generated captions, we employ a combination of traditional metrics, like BLEU, METEOR, CIDEr, and SPICE, and more recent ones such as CLIP-S, Polos and the proposed PAC-S++ metric, considering in both cases reference-based and reference-free settings. Additionally, we introduce novel metrics to assess the grammatical correctness of the generated captions, which is crucial especially when directly optimizing CLIP-based scores. Specifically, we measure the average number of repeated n-grams (Rep-n) and the percentage of captions ending with undesirable words like prepositions, conjunctions, or determiners (%Incorrect).

In-domain Evaluation. Captioning results on the COCO test set are reported in Table 8.6. Notably, although CLIP remains an excellent model for aligning bag-of-words with visual input, it disregards syntax and logical connections among words within captions. On the contrary, despite sharing the same architecture, our proposal mitigates this issue, favouring the use of PAC-S++ as a reward metric in a captioning model. In particular, directly optimizing CLIP-S leads to protracted and repetitive captions, as demonstrated by the lower scores in terms of standard reference-based metrics and grammar measures. In contrast, PAC-S++ significantly stabilizes the fine-tuning process, yielding significant enhancements in reference-based metrics (*e.g.* 36.3 and 51.8 CIDEr points using PAC-S++ with ViT-B/32 and ViT-L/14 features vs. 1.1 and 1.4 obtained by CLIP-S). Con-

Table 8.6: Captioning results in terms of reference-based, reference-free, and grammar evaluation metrics on COCO test set, using visual features extracted from different CLIP-based backbones as input to the captioning model.

		Reference-based ↑							Reference-free ↑		Grammar ↓				
Backbone	**Reward**	B-4	M	C	S	RefCLIP-S	Polos	RefPAC-S++	CLIP-S	PAC-S++	Rep-1	Rep-2	Rep-3	Rep-4	%Incorrect
ViT-B/32	-	33.1	28.2	112.4	20.5	0.804	0.651	0.794	0.755	0.712	1.468	0.091	0.017	0.005	0.3
	CIDEr	40.4	29.4	129.6	21.6	0.806	0.651	0.799	0.751	0.714	1.318	0.038	0.006	0.004	24.7
	CLIP-S	12.1	23.5	1.1	20.0	0.767	0.635	0.776	**0.844**	0.744	12.226	4.736	1.884	0.795	99.2
	PAC-S++	**19.4**	**27.1**	**36.3**	**22.4**	**0.801**	**0.658**	**0.795**	0.813	**0.755**	**5.129**	**1.431**	**0.544**	**0.229**	**0.7**
	RefCLIP-S	26.3	27.6	92.5	21.4	**0.829**	**0.679**	0.807	**0.799**	0.735	2.571	0.626	0.236	0.103	**0.3**
	RefPAC-S++	**30.5**	**28.5**	**109.1**	**22.2**	0.822	0.677	**0.811**	0.784	**0.740**	**1.791**	**0.247**	**0.069**	**0.026**	**0.3**
ViT-L/14	-	34.8	29.9	119.4	22.5	0.802	0.078	0.708	0.749	0.708	1.469	0.064	0.008	0.002	0.3
	CIDEr	43.6	30.8	143.3	23.2	0.809	0.668	0.804	0.750	0.713	1.432	0.047	0.005	0.002	32.3
	CLIP-S	13.1	24.6	1.4	20.0	0.782	0.656	0.780	**0.840**	0.736	11.225	4.447	2.08	1.012	34.8
	PAC-S++	**20.9**	**28.0**	**51.8**	**23.9**	**0.806**	**0.675**	**0.797**	0.812	**0.751**	**4.157**	**0.974**	**0.33**	**0.129**	**1.3**
	RefCLIP-S	27.8	28.8	101.9	23.3	**0.833**	0.700	0.811	**0.800**	0.734	2.161	0.386	0.13	0.046	0.7
	RefPAC-S++	**32.5**	**29.6**	**118.9**	**23.5**	0.826	**0.702**	**0.814**	0.782	**0.736**	**1.468**	**0.145**	**0.037**	**0.011**	**0.9**

currently, it enables the generation of semantically rich and grammatically correct captions that better correlate with human-generated content. This phenomenon is particularly notable in repetitiveness metrics, where the average number of repeated 1-grams in the generated captions decreases from 11.225 to 4.157, when using VIT-L/14 as visual backbone.

Similar considerations apply to the reference-based version, where a reduction in caption generation creativity is observed to align more closely with ground-truth sentences. This approach results in a softer degradation of reference-based metrics, producing values nearly identical to those obtained by the baseline model trained with cross-entropy loss, but achieving higher scores in learnable metrics (*e.g.* 0.708 and 0.713 in terms of PAC-S++ respectively with cross-entropy loss only and CIDEr as reward vs. 0.736 achieved when employing RefPAC-S++ as a reward).

To validate the quality of generated captions, qualitative results on sample images from the COCO dataset are reported in Fig. 8.7, where we compare captions generated by the model fine-tuned using PAC-S++ as reward with those generated using CIDEr or CLIP-S. As it can be seen, our proposal can generate more descriptive and detailed captions, while reducing repetitions and grammatical errors. Specifically, while CIDEr generally leads to shorter captions, both CLIP-S and PAC-S++ can comprehensively

Figure 8.7: Qualitative image captioning results with different metrics as reward.

Table 8.7: Captioning results in terms of reference-based and reference-free metrics on nocaps and VizWiz validation sets.

		nocaps						VizWiz					
Backbone	**Reward**	C	CLIP-S	PAC-S++	RefCLIP-S	Polos	RefPAC-S++	C	CLIP-S	PAC-S++	RefCLIPS	Polos	RefPAC-S++
ViT-B/32	-	67.6	0.686	0.694	0.699	0.432	0.733	27.8	0.655	0.675	0.691	0.360	0.729
	CIDEr	76.2	0.695	0.703	0.709	0.418	0.741	28.9	0.663	0.686	0.704	0.345	0.739
	CLIP-S	1.6	0.780	0.726	0.675	0.467	0.724	1.1	**0.735**	0.703	0.686	0.403	0.722
	PAC-S++	**34.6**	**0.751**	**0.743**	**0.713**	**0.485**	**0.748**	**17.5**	0.721	**0.729**	**0.717**	**0.434**	**0.751**
	Ref-CLIP-S	64.0	**0.736**	0.724	**0.734**	**0.475**	0.753	25.0	**0.703**	0.708	**0.723**	0.405	0.747
	RefPAC-S++	**73.1**	0.724	**0.729**	0.728	0.474	**0.758**	**29.4**	0.694	**0.715**	**0.723**	**0.412**	**0.758**
ViT-L/14	-	75.2	0.698	0.704	0.710	0.473	0.743	35.0	0.655	0.679	0.701	**0.407**	0.740
	CIDEr	91.3	0.698	0.711	0.718	0.438	0.755	39.6	0.667	0.683	0.722	0.365	0.751
	CLIP-S	2.1	**0.791**	0.741	0.705	0.508	0.746	1.6	**0.727**	0.703	0.711	0.427	0.741
	PAC-S++	**49.1**	0.769	**0.754**	**0.735**	**0.536**	**0.764**	**26.1**	0.713	**0.723**	**0.726**	**0.474**	**0.759**
	Ref-CLIP-S	79.0	**0.756**	0.742	**0.756**	0.528	0.774	35.0	**0.705**	0.708	**0.738**	0.456	0.761
	RefPAC-S++	**89.8**	0.741	**0.744**	0.750	**0.530**	**0.776**	**41.3**	0.695	**0.715**	0.737	**0.468**	**0.770**

describe the visual content of the images. At the same time, however, using CLIP-S as reward significantly reduces the grammatical correctness of generated captions. This drawback is mitigated when employing PAC-S++ as reward, further demonstrating the effectiveness of our solution.

Out-of-domain Evaluation. Finally, we evaluate the out-of-domain performance of our model on the nocaps [3] and VizWiz [92] datasets, both of which present distinct image descriptions compared to the COCO dataset used for training. Specifically, the nocaps dataset, which is designed for the novel object captioning task, includes image-caption pairs featuring objects not present in the COCO training set. In contrast, VizWiz consists of images taken by visually impaired individuals, often showcasing challeng-

ing perspectives, such as close-up shots or unconventional viewpoints. The results, summarized in Table 8.7, are evaluated using both reference-free and reference-based metrics. Also in these challenging settings, our approach demonstrates greater semantic richness while preserving fluidity and grammatical correctness in text generation. This behavior is not observed when CLIP-S is used as reward. Specifically, although the use of CLIP-S results in high scores on learnable metrics, the values of traditional metrics remain notably low. For instance, on the nocaps dataset and using VIT-L/14 as visual backbone, the CIDEr score drops from 49.1 points when using PAC-S++ as reward to just 2.1 points with CLIP-S as reward, further highlighting the advantages of our metric for training captioning models.

8.5 From Captioning to Multimodal LLMs

With the rise of MLLMs as image description generators, the nature of image captionshas undergone a substantial transformation. Unlike traditional captioning models, which typically produce short, concise, and largely descriptive sentences, MLLMs tend to generate captions that are significantly longer, more detailed, and often enriched with implicit reasoning, background knowledge, or explanatory content.

This shift poses new challenges for caption evaluation. Conventional metrics, which were primarily designed to measure surface-level similarity to a limited set of reference captions, may fail to reliably assess the quality, factual correctness, and relevance of these more complex descriptions.

Consequently, the transition from classical captioning models to MLLMs calls for a fundamental rethinking of caption evaluation protocols, with metrics that can better account for semantic consistency, visual grounding, and reasoning-based content.

Table 8.8: Evaluation scores on the COCO test set comparing traditional captioning models with general-purpose MLLMs. For MLLMs, we assess both short captions and longer, unconstrained descriptions generated using the default prompt of the model.

		LLM	Length	BLEU-4	METEOR	CIDEr	CLIP-S	PAC-S++	RefCLIP-S	RefPAC-S++	Polos	BRIDGE
Captioning Models	Show and Tell [272]	-	9.1	31.4	25.0	97.2	0.715	0.654	0.779	0.752	0.585	0.788
	Show, Attend and Tell [289]	-	9.1	33.4	26.2	104.6	0.727	0.670	0.790	0.766	0.609	0.804
	Up-Down [7]	-	9.5	36.7	27.9	122.7	0.740	0.680	0.804	0.778	0.640	0.821
	SGAE [295]	-	9.4	38.6	28.8	129.8	0.750	0.691	0.812	0.786	0.655	0.833
	AoANet [108]	-	9.5	39.1	29.0	128.9	0.753	0.693	0.813	0.787	0.660	0.836
	$\mathcal{M}^2$ Transformer [64]	-	9.7	39.1	29.2	131.2	0.757	0.699	0.813	0.791	0.629	0.841
	X-Transformer [206]	-	9.6	39.7	29.5	132.8	0.762	0.701	0.819	0.792	0.668	0.845
	VinVL [316]	-	10.0	41.0	31.1	140.9	0.784	0.715	0.836	0.805	0.708	0.865
	COS-Net [152]	-	9.6	42.0	30.6	141.1	0.773	0.712	0.829	0.803	0.692	0.859
	BLIP-2 [147]	Flan T5-XL	9.6	**43.8**	**31.7**	**146.0**	**0.782**	**0.719**	**0.838**	**0.810**	**0.716**	**0.868**
General Purpose MLLMs	InstructBLIP [66]	Vicuna-7B	10.2	**41.2**	**31.8**	**142.2**	0.786	0.721	0.838	**0.810**	**0.714**	0.871
	LLaVA-1.5 [167]	Vicuna-7B	14.5	8.1	28.0	69.6	0.785	0.707	0.828	0.794	0.666	0.867
	IDEFICS [133]	Llama-7B	8.8	36.8	28.3	125.1	0.788	0.711	**0.840**	0.803	0.699	0.863
	IDEFICS-2 [134]	Mistral-7B	12.2	19.1	24.3	74.1	0.800	0.711	0.819	0.780	0.626	0.865
	IDEFICS-3 [132]	Llama-3-8B	14.2	17.4	24.5	62.2	0.801	0.710	0.811	0.771	0.615	0.865
	Llama-3.2 [72]	Llama-3.2-11B	22.2	14.3	25.8	46.0	**0.827**	**0.734**	0.828	0.796	0.692	**0.900**
	InstructBLIP [66]	Vicuna-7B	43.3	9.8	**25.0**	2.8	0.828	**0.738**	0.817	**0.792**	**0.709**	**0.912**
	LLaVA-1.5 [167]	Vicuna-7B	49.5	8.3	23.0	0.3	0.813	0.722	0.808	0.783	0.689	0.898
	IDEFICS [133]	Llama-7B	29.0	**11.4**	24.8	**43.9**	0.792	0.719	0.815	0.788	0.679	0.867
	IDEFICS-2 [134]	Mistral-7B	22.1	7.3	20.0	31.7	0.728	0.683	0.773	0.760	0.575	0.794
	IDEFICS-3 [132]	Llama-3-8B	123.6	2.0	12.2	8.3	0.777	0.697	0.770	0.748	0.605	0.856
	Llama-3.2 [72]	Llama-3.2-11B	31.4	10.2	23.9	30.5	**0.832**	0.736	**0.818**	0.790	0.689	0.906
	Humans		*10.4*	-	*24.1*	*87.6*	*0.782*	*0.710*	*0.822*	*0.792*	*0.654*	*0.856*

8.5.1 Evaluating Captioning Metrics in the Era of Multimodal LLMs

We assess whether well-established captioning metrics effectively evaluate captions generated by modern MLLMs, especially when their style deviates from traditional COCO captions. The potential limitations of these metrics originate from their design: rule-based metrics rely on reference captions that adhere to COCO-style annotations, while learnable metrics are primarily fine-tuned on COCO or similar datasets. As a result, their ability to evaluate captions with diverse linguistic structures remains uncertain.

To investigate this, we assess MLLM-generated captions in two formats: short captions, comparable in length to COCO ones, and longer, unconstrained descriptions[8]. This enables us to analyze metric performance across varying caption styles and lengths.

[8]For short captions, we use the prompt "`Briefly describe the image`". For longer ones, we rely on the MLLM default prompt (*e.g.* "`What is the content of the image?`").

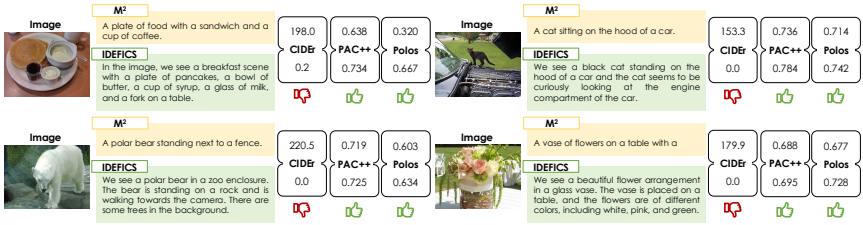


Figure 8.8: Qualitative examples showing the differences between captions generated by a traditional captioning model (*i.e.*, $\mathcal{M}^2$ Transformer) and MLLM-style captions (*i.e.*, IDEFICS). Length and style variations affect rule-based metrics, while learnable metrics remain robust.

In Table 8.8, we evaluate popular captioning models on the COCO test set using a range of metrics to assess their effectiveness and highlight differences in their behavior**. Our analysis includes both traditional captioning models and modern MLLMs designed for broader tasks. Additionally, we establish a human-based baseline by randomly selecting one human-annotated caption from the five provided in COCO and comparing it against the remaining references††.

For evaluation, we employ standard scores such as BLEU, METEOR and CIDEr, as well as two widely used learnable metrics, CLIP-S and PAC-S++, along with their reference-based counterparts. Additionally, we include two recent evaluation methods: Polos, a supervised metric, and BRIDGE, which focuses on fine-grained visual details. This selection allows for assessing how different methods capture caption quality across models.

As it can be seen, for models explicitly trained for the image captioning task, such as $\mathcal{M}^2$ Transformer, the generated captions tend to align with COCO in both length and style. Under these conditions, traditional evaluation metrics like CIDEr, effectively recognize the superior quality of BLIP-2 captions. Similarly, recently developed metrics leveraging large-scale pre-

**Note that for BLIP-2 we employ the captioning-specific model fine-tuned on COCO image-caption pairs.

††BLEU is not reported for the human-based baseline as its value is sensitive to the number of reference captions.

trained models maintain strong performance.

When evaluating captions generated by MLLMs interesting patterns emerge. If the captions remain similar in length to the ones contained in COCO, rule-based metrics still perform effectively: CIDEr, for instance, assigns a score of 142.2 to captions generated by InstructBLIP. However, as caption length increases, these metrics exhibit a sharp decline in scores. Notably, when evaluating LLaVA-1.5, which produces longer captions than COCO ones, CIDEr score drops dramatically to just 0.3. Qualitative results showing this trend are reported in Fig. 8.8. This decline stems from rule-based metrics relying on COCO-style captions, which are shorter and simpler than detailed MLLM outputs, resulting in misleading evaluations.

In contrast, more recent metrics that incorporate large-scale components exhibit greater robustness to variations in caption length. A slight decline in performance is still observed: for example, PAC-S++ decreases from 0.710 to 0.697 when caption length increases nearly ninefold (*i.e.*, for IDEFICS-3). This primarily reflects a lower confidence level rather than a failure of the metric. In fact, although these metrics were not explicitly designed for evaluating long, highly detailed captions, they still maintain a high degree of reliability. For learnable reference-based metrics such as RefCLIP-S, RefPAC-S++, and Polos, performance degrades more significantly compared to their reference-free counterparts, reflecting a trend similar to that observed in rule-based metrics. However, unlike rule-based solutions, these metrics retain their ability to distinguish high-quality captions, correctly rewarding models such as Llama-3.2 and InstructBLIP.

Overall, these results indicate that recent learnable metrics designed for image captioning remain valuable for evaluating longer and more detailed captions generated by MLLMs, demonstrating reliability despite variations in length and style. However, reference-free metrics are generally preferable, as they more effectively distinguish high-quality captions and exhibit greater robustness to variations in length.

9

Conclusions and Future Works

This thesis investigated the evolving landscape of vision-and-language models through the lens of retrieval augmentation and evaluation, focusing on how external knowledge, memory, and improved evaluation strategies can enhance multimodal reasoning. Starting from classical image captioning models and progressing toward modern multimodal large language models, the work examined how retrieval mechanisms fundamentally change the way visual and linguistic information are processed, generated, and evaluated.

9.1 Future Directions and Open Problems

Despite progress in vision-and-language tasks, several challenges remain, offering opportunities for future research.

Benchmark Evolution. Creating new benchmarks plays a fundamental role in advancing vision-and-language research, both for the development of Multimodal LLMs and for the design of reliable evaluation methodologies. Traditional vision datasets mainly feature short captions similar to those in COCO, creating a gap with the longer, more detailed outputs of modern MLLMs. While recent efforts, like Polaris and Nebula, incorporate captions from standard models, they often maintain the concise style of COCO descriptions. A valuable direction involve creating new benchmarks specifically designed to assess the quality of metrics on longer, richer captions that better reflect the MLLM outputs, addressing challenges like synonyms, paraphrases, and domain-specific terms. This need extends beyond captioning evaluation to retrieval-augmented and knowledge-intensive tasks: benchmarks such as Encyclopedic-VQA and InfoSeek could be further enhanced to more thoroughly stress-test retrieval quality, evidence selection, and the integration of retrieved information into generation.

Hallucinations. Hallucination represents a critical challenge not only in image captioning but also in multimodal large language models more broadly. As MLLMs gain the ability to produce longer, more detailed, and more expressive descriptions, the risk of hallucinating unsupported or incorrect content increases substantially. These hallucinations often go beyond simple object-level errors and instead manifest as distorted relationships, incorrect attributes, or entirely fabricated entities and actions. Addressing this issue requires progress along two complementary directions. On the one hand, future research should focus on mitigating hallucinations at the model level, for instance through improved grounding, retrieval-aware

generation, or stronger reasoning constraints. On the other hand, there is a pressing need for evaluation metrics capable of reliably detecting hallucinations in complex multimodal outputs. While existing benchmarks such as FOIL target relatively simple hallucination patterns, more diverse datasets and evaluation protocols are needed to assess higher-order semantic errors and to challenge metrics on subtle, knowledge-based inconsistencies characteristic of MLLM-generated captions.

Personalization. Personalization remains an important and largely underexplored challenge for both multimodal large language models and their evaluation. Modern Multimodal LLMs have demonstrated impressive performance in generic visual understanding and question answering tasks. However, they remain fundamentally limited when queries depend on user-specific concepts and external or private knowledge. Addressing personalization represents an open challenge for multimodal systems. Moreover, current evaluation metrics rely on reference captions or image-caption alignment, but they fail to accommodate the diversity of user preferences or task-specific requirements. Future research should focus on the development of personalized evaluation metrics, that can let users prioritize factors such as detail, brevity, stylistic preferences, or domain relevance. By incorporating these custom priorities, evaluation can become more adaptable and meaningful across a variety of applications and user needs.

Explainability in Metrics. Most evaluation metrics generate scores without offering insights into the rationale behind their assessments, limiting their usefulness for improving captioning models. A few metrics, such as CLAIR, have advanced in this area by leveraging the interpretative capabilities of LLMs to offer explanations alongside scores. Building upon these advancements, future research should focus on further enhancing explainability, facilitating a deeper understanding of the strengths and weaknesses of evaluation models.

9.1.1 Summary of Contributions

A first major contribution of this thesis is a systematic analysis of retrieval-augmented architectures for image captioning. By exploring both external retrieval and internal memory mechanisms, the proposed models demonstrate how access to past examples and retrieved knowledge can significantly improve caption quality, semantic richness, and generalization to novel concepts. The introduction of retrieval-augmented and memory-augmented Transformers highlights how external information can be effectively integrated into attention-based architectures, yielding consistent gains over strong baselines.

Building on these foundations, the thesis extended retrieval augmentation to multimodal large language models, addressing knowledge-intensive Visual Question Answering scenarios. Through the design and evaluation of hierarchical and reasoning-augmented retrieval pipelines, this work showed that retrieval quality plays a decisive role in downstream performance. In particular, the proposed approaches demonstrate that off-the-shelf MLLMs, when paired with carefully designed retrieval and filtering strategies, can achieve competitive or state-of-the-art results without task-specific fine-tuning, even at large scale and under challenging benchmark conditions.

A further central theme of this thesis concerns evaluation. As captioning models and MLLMs generate increasingly complex and diverse outputs, traditional evaluation metrics become insufficient to fully capture improvements in grounding, factual correctness, and semantic coherence. To address this, the thesis presented a comprehensive taxonomy of captioning evaluation metrics and introduced learnable, visually grounded metrics based on positive-augmented contrastive learning. The proposed PAC-S and PAC-S++ metrics demonstrate stronger correlation with human judgments, improved robustness to stylistic variation, and sensitivity to object

hallucinations. Moreover, the use of learned metrics as training objectives was shown to effectively guide reinforcement learning–based fine-tuning, further closing the gap between evaluation and generation.

Overall, this work underscores the importance of treating retrieval and evaluation as first-class components in modern vision-and-language systems. Retrieval augmentation emerges not merely as an auxiliary mechanism, but as a key enabler for scalable, knowledge-aware, and reasoning-capable multimodal models. At the same time, the evolution of generation capabilities necessitates a corresponding evolution in evaluation methodologies, particularly in the era of multimodal large language models.

Closing Remarks. By jointly advancing retrieval strategies, model architectures, and evaluation methodologies, this work contributes toward a more holistic view of vision-and-language systems, where generation, knowledge access, and evaluation are tightly interconnected. Looking forward, continued progress in this area will require integrated solutions that combine effective retrieval, reliable reasoning, and adaptive evaluation, ultimately enabling multimodal models that are not only more powerful, but also more trustworthy and better suited to real-world applications.

List of Publications

1. **Sarto Sara**, Cornia Marcella, Baraldi Lorenzo, and Cucchiara Rita. "Retrieval-augmented transformer for image captioning." In *Proceedings of the International Conference on Content-based Multimedia Indexing (CBMI)*. 2022. Best Student Paper.
2. Barraco Manuele, **Sarto Sara**, Cornia Marcella, Baraldi Lorenzo, and Cucchiara Rita. "With a little help from your own past: Prototypical memory networks for image captioning." In *Proceedings of the IEEE/CVF International Conference on Computer Vision. (ICCV)*. 2023.
3. **Sarto Sara**, Barraco Manuele, Cornia Marcella, Baraldi Lorenzo, and Cucchiara Rita. "Positive-Augmented Contrastive Learning for Image and Video Captioning Evaluation." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023. Highlight Paper.
4. Caffagni Davide*, Cocchi Federico*, Barsellotti Luca*, Moratelli Nicholas*, **Sarto Sara***, Baraldi Lorenzo, Cornia Marcella, and Cucchiara Rita. "The Revolution of Multimodal Large Language Models: A Survey." In *Findings of the Association for Computational Linguistics (ACL)*. 2024.
5. Cucchiara Rita, Baraldi Lorenzo, Cornia Marcella, and **Sarto Sara**. "Video Surveillance and Privacy: A Solvable Paradox?" *Computer*. 2024.
6. Caffagni Davide*, Cocchi Federico*, Moratelli Nicholas*, **Sarto Sara***, Cornia Marcella, Baraldi Lorenzo, and Cucchiara Rita. "Wiki-LLaVA: Hi-

erarchical Retrieval-Augmented Generation for Multimodal LLMs." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2024.

7. **Sarto Sara**, Cornia Marcella, Baraldi Lorenzo, Nicolosi Alessandro, and Cucchiara Rita. "Towards Retrieval-Augmented Architectures for Image Captioning." *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*. 2024.

8. Poppi Samuele, **Sarto Sara**, Cornia Marcella, Baraldi Lorenzo, and Cucchiara Rita. "Multiclass Unlearning for Image Classification via Weight Filtering." *IEEE Intelligent Systems*. 2024.

9. **Sarto Sara**, Cornia Marcella, Baraldi Lorenzo, and Cucchiara Rita. "BRIDGE: Bridging Gaps in Image Captioning Evaluation with Stronger Visual Cues." In *Proceedings of the European Conference on Computer Vision (ECCV)*. 2024.

10. Poppi Samuele, **Sarto Sara**, Cornia Marcella, Baraldi Lorenzo, and Cucchiara Rita. "Unlearning Vision Transformers without Retaining Data via Low-Rank Decompositions." In *Proceedings of the International Conference on Pattern Recognition (ICPR)*. 2024.

11. Pipoli Vittorio, Bolelli Federico, **Sarto Sara**, Cornia Marcella, Baraldi Lorenzo, Grana Costantino, Cucchiara Rita, and Ficarra Elisa. "Semantically Conditioned Prompts for Visual Recognition under Missing Modality Scenarios." In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2025.

12. Caffagni Davide*, **Sarto Sara***, Cornia Marcella, Baraldi Lorenzo, and Cucchiara Rita. "Recurrence-Enhanced Vision-and-Language Transformers for Robust Multimodal Document Retrieval." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2025.

13. **Sarto Sara**, Cornia Marcella, and Cucchiara Rita. "Image Captioning Evaluation in the Age of Multimodal LLMs: Challenges and Future Perspectives." In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. 2025.

14. Cocchi Federico, Moratelli Nicholas, Caffagni Davide, **Sarto Sara**, Baraldi Lorenzo, Cornia Marcella, and Cucchiara Rita. "LLaVA-MORE: A Comparative Study of LLMs and Visual Backbones for Enhanced Visual Instruction Tuning." In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*. 2025.

15. **Sarto Sara**, Moratelli Nicola, Cornia Marcella, Baraldi Lorenzo, and Cucchiara Rita. "Positive-Augmented Contrastive Learning for Vision-and-Language Evaluation and Training." *International Journal of Computer Vision (IJCV)*, 2025.

16. Compagnoni Alberto, Morini Marco, **Sarto Sara**, Cocchi Federico, Caffagni Davide, Cornia Marcella, Baraldi Lorenzo, and Cucchiara Rita. "ReAG: Reasoning-Augmented Generation for Knowledge-Based Visual Question Answering." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2026.

List of Publications Under Submissions

1. Mattioli Gabriele, Turri Evelyn, **Sarto Sara**, Baraldi Lorenzo, Cornia Marcella, Baraldi Lorenzo, and Cucchiara Rita. "RaTA-Tool: Retrieval-based Tool Selection with Multimodal Large Language Models." *under submission*. 2026.
2. Caffagni Davide, **Sarto Sara**, Cornia Marcella, Baraldi Lorenzo, and Cucchiara Rita. "Recurrence Meets Transformers for Universal Multimodal Retrieval." *under submission*. 2025.
3. Caffagni Davide, **Sarto Sara**, Cornia Marcella, Baraldi Lorenzo, Dovesi Pier Luigi, Roohi Shaghayegh, Granroth-Wilding Mark, and Cucchiara Rita. "Seeing Beyond Words: Self-Supervised Visual Learning for Multimodal Large Language Models." *under submission*. 2025.

Ph.D. Activities

This section presents a list of the activities carried out by the candidate during the Ph.D. program in Information and Communication Technologies (ICT).

Internship

November 2024 - May 2025: Internship at Amazon London – Prime Video Team.

Workshops Organizer

June 2024: Trust What You learN (TWYN) Workshop, ECCV.

Honors and Awards

2025: Doctoral Consortium (CVPR);

2024: Travel Award for Workshop "Women in Computer Vision", received for the paper: "BRIDGE: Bridging Gaps in Image Captioning Evaluation with Stronger Visual Cues";

2023: Best Poster Award, received for the poster: "Augmented Architectures for Vision and Language";

2023: Travel Award for Workshop "Women in Computer Vision", received for the paper: "With a Little Help from your own Past: Prototypical Memory Networks for Image Captioning";

2023: Master Thesis Award, Premio alla Memoria Davide Rabotti

2023: Travel Award for Workshop "Women in Computer Vision", received for the paper: "Positive-Augmented Contrastive Learning for Image and Video Captioning Evaluation";

2022: Best Student Paper Award, received for the paper: "Retrieval-augmented Transformer for Image Captioning".

Conference Attendances

September 2025: International Conference on Image Analysis and Processing (ICIAP), Invited Speaker, Rome, Italy;

August 2025: International Joint Conferences on Artificial Intelligence (IJCAI), Paper presentation, Montreal, Canada;

June 2025: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Paper presentation, Nashville, Tennessee;

June 2024: European Conference on Computer Vision (ECCV), Paper presentation, Milan, Italy;

October 2023: IEEE/CVF International Conference on Computer Vision (ICCV), Paper presentation, Paris, France;

June 2023: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Paper presentation, Vancouver, Canada;

November 2022: International Conference on Content-based Multimedia Indexing (CBMI), Paper presentation, Graz, Austria.

Seminars and Workshops Attendances

May 2023: Academic English Workshop II;

May 2023: Workshop AI per l'Industria;

May 2023: Workshop AI per la Finanza ed il Commercio;

December 2022: 3D Computer Vision for animals (GENDER UNBALANCED AI), Speaker: Silvia Zuffi;

December 2022: High Performance Computing and Cloud Computing, key enablers for digital transformation, Speaker: Carlo Cavazzoni;

December 2022: From Handcrafted to End-to-End Learning, and Back: a Journey far Multi-Object Tracking (GENDER UNBALANCED AI), Speaker: Laura Leal-Taixé;

November 2022: Graph Signal Processing for Machine Learning: Challenges and Use-cases (GENDER UNBALANCED AI), Speaker: Laura Toni;

———————————————————————————————

June 2024: Women in Computer Vision, European Conference on Computer Vision (ECCV), Paper presentation, Milan, Italy;

October 2023: Women in Computer Vision, IEEE/CVF International Conference on Computer Vision (ICCV), Paper presentation, Paris, France;

June 2023: Women in Computer Vision, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Paper presentation, Vancouver, Canada.

Summer or Winter Schools

March 2024: ELLIS Winter School on Foundation Models, Amsterdam, The Netherlands;

September 2024: ELLIS Summer School on Large-Scale AI for Research and Industry, Modena, Italy;

September 2023: VISMAC International Summer School on Machine Vision, Padova, Italy.

Participation to Research Projects

2025: IT4LIA – Italy for Artificial Intelligence, EuroHPC JU;

2025: ELLIOT – European Large Open Multi-Modal Foundation Models For Robust Generalization On Arbitrary Data Streams, Horizon Europe RIA. GA No. 101214398;

2024: MINERVA – EuroHPC JU. GA No. 101182737;

2024: ELIAS – European Lighthouse of AI for Sustainability, Horizon Europe RIA. GA No. 101120237;

2024: ELSA – European Lighthouse on Secure and Safe AI, Horizon Europe RIA. GA No. 101070617.

Teaching Activities

- Lecturer for the "Scuola Python, Deep Learning and Computer Vision";
- Lecturer on Transfomer and Attention for the Course "Computer Vision and Cognitive System" in the Master Course of Artificial intelligence engineering;
- Lecture for the Project "Intensive Master AI and ML for Smart Factory", organized by Experis s.r.l..

Reviewing Service

- IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR);
- EEE/CVF International Conference on Computer Vision (ICCV);
- European Conference on Computer Vision (ECCV);
- ACM Multimedia;

- International Journal of Computer Vision (IJCV);
- International Conference on Geometric Modeling and Processing (GMP);
- Elsevier Pattern Recognition Letters.

Bibliography

[1] Abdin, M., Aneja, J., Behl, H., Bubeck, S., Eldan, R., Gunasekar, S., Harrison, M., Hewett, R. J., Javaheripi, M., Kauffmann, P., et al. (2024). Phi-4 Technical Report. *arXiv preprint arXiv:2412.08905*.

[2] Aditya, S., Yang, Y., Baral, C., Fermuller, C., et al. (2015). From Images to Sentences through Scene Description Graphs using Commonsense Reasoning and Knowledge. *arXiv:1511.03292*.

[3] Agrawal, H., Desai, K., Wang, Y., Chen, X., Jain, R., Johnson, M., Batra, D., Parikh, D., Lee, S., and Anderson, P. (2019). nocaps: novel object captioning at scale. In *IEEE International Conference on Computer Vision*.

[4] Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., et al. (2022). Flamingo: a Visual Language Model for Few-Shot Learning. In *NeurIPS*.

[5] Anderson, P., Fernando, B., Johnson, M., and Gould, S. (2016). SPICE: Semantic Propositional Image Caption Evaluation. In *Proceedings of the European Conference on Computer Vision*.

[6] Anderson, P., Fernando, B., Johnson, M., and Gould, S. (2017). Guided open vocabulary image captioning with constrained beam search. In *Conference on Empirical Methods in Natural Language Processing*.

[7] Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., and Zhang, L. (2018). Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[8] Aneja, J., Deshpande, A., and Schwing, A. G. (2018). Convolutional image captioning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[9] Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C. L., and Parikh, D. (2015). Vqa: Visual Question Answering. In *IEEE International Conference on Computer Vision*.

[10] Arora, S., Narayan, A., Chen, M. F., Orr, L. J., Guha, N., Bhatia, K., Chami, I., Sala, F., and Ré, C. (2022). Ask me anything: A simple strategy for prompting language models. *arXiv preprint arXiv:2210.02441*.

[11] Awadalla, A., Gao, I., Gardner, J., Hessel, J., Hanafy, Y., Zhu, W., Marathe, K., Bitton, Y., Gadre, S., Sagawa, S., et al. (2023). OpenFlamingo: An Open-Source Framework for Training Large Autoregressive Vision-Language Models. *arXiv preprint arXiv:2308.01390*.

[12] Bai, J., Bai, S., Yang, S., Wang, S., Tan, S., Wang, P., Lin, J., Zhou, C., and Zhou, J. (2023). Qwen-VL: A Versatile Vision-Language Model for Understanding, Localization, Text Reading, and Beyond. *arXiv preprint arXiv:2308.12966*.

[13] Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., et al. (2025). Qwen2.5-VL Technical Report. *arXiv preprint arXiv:2502.13923*.

[14] Baldrati, A., Bertini, M., Uricchio, T., and Del Bimbo, A. (2022). Conditioned and Composed Image Retrieval Combining and Partially Fine-Tuning CLIP-Based Features. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[15] Banerjee, S. and Lavie, A. (2005). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *ACL Workshops*.

[16] Barraco, M., Cornia, M., Cascianelli, S., Baraldi, L., and Cucchiara, R. (2022a). The Unreasonable Effectiveness of CLIP Features for Image Captioning: An Experimental Analysis. In *IEEE International Conference on Computer Vision and Pattern Recognition Workshops*.

[17] Barraco, M., Sarto, S., Cornia, M., Baraldi, L., and Cucchiara, R. (2023). With a Little Help from your own Past: Prototypical Memory Networks for Image Captioning. In *IEEE International Conference on Computer Vision*.

[18] Barraco, M., Stefanini, M., Cornia, M., Cascianelli, S., Baraldi, L., and Cucchiara, R. (2022b). CaMEL: Mean Teacher Learning for Image Captioning. In *International Conference on Pattern Recognition*.

[19] Bird, S., Klein, E., and Loper, E. (2009). *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media, Inc.

[20] Borgeaud, S., Mensch, A., Hoffmann, J., Cai, T., Rutherford, E., Millican, K., Driessche, G. v. d., Lespiau, J.-B., Damoc, B., Clark, A., et al. (2022). Improving Language Models by Retrieving from Trillions of Tokens. In *International Conference on Machine Learning*.

[21] Brazil, G., Kumar, A., Straub, J., Ravi, N., Johnson, J., and Gkioxari, G. (2023). Omni3D: A Large Benchmark and Model for 3D Object Detection in the Wild. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[22] Brown, A., Xie, W., Kalogeiton, V., and Zisserman, A. (2020a). Smooth-AP: Smoothing the Path Towards Large-Scale Image Retrieval. In *Proceedings of the European Conference on Computer Vision*.

[23] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020b). Language models are few-shot learners. In *NeurIPS*.

[24] Bugliarello, E., Liu, F., Pfeiffer, J., Reddy, S., Elliott, D., Ponti, E. M., and Vulić, I. (2022). IGLUE: A Benchmark for Transfer Learning Across Modalities, Tasks, and Languages. In *International Conference on Machine Learning*.

[25] Bulian, J., Buck, C., Gajewski, W., Boerschinger, B., and Schuster, T. (2022). Tomayto, Tomahto. Beyond Token-level Answer Equivalence for Question Answering Evaluation. *arXiv preprint arXiv:2202.07654*.

[26] Byeon, M., Park, B., Kim, H., Lee, S., Baek, W., and Kim, S. (2022). COYO-700M: Image-Text Pair Dataset.

[27] Caffagni, D., Cocchi, F., Barsellotti, L., Moratelli, N., Sarto, S., Baraldi, L., Cornia, M., and Cucchiara, R. (2024a). The Revolution of Multimodal Large Language Models: A Survey. In *ACL Findings*.

[28] Caffagni, D., Cocchi, F., Barsellotti, L., Moratelli, N., Sarto, S., Baraldi, L., Cornia, M., and Cucchiara, R. (2024b). The (R)Evolution of Multimodal Large Language Models: A Survey. *arXiv preprint arXiv.2402.12451*.

[29] Caffagni, D., Cocchi, F., Moratelli, N., Sarto, S., Cornia, M., Baraldi, L., and Cucchiara, R. (2024c). Wiki-llava: Hierarchical retrieval-augmented generation for multimodal llms. In *IEEE International Conference on Computer Vision and Pattern Recognition Workshops*.

[30] Caffagni, D., Sarto, S., Cornia, M., Baraldi, L., and Cucchiara, R. (2025a). Recurrence-Enhanced Vision-and-Language Transformers for Robust Multimodal Document Retrieval. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[31] Caffagni, D., Sarto, S., Cornia, M., Baraldi, L., and Cucchiara, R. (2025b). Recurrence Meets Transformers for Universal Multimodal Retrieval. *arXiv preprint arXiv:2509.08897*.

[32] Cagrandi, M., Cornia, M., Stefanini, M., Baraldi, L., and Cucchiara, R. (2021). Learning to Select: A Fully Attentive Approach for Novel Object Captioning. In *ICMR*.

[33] Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. (2021). Emerging Properties in Self-Supervised Vision Transformers. In *IEEE International Conference on Computer Vision*.

[34] Cha, J., Kang, W., Mun, J., and Roh, B. (2023). Honeybee: Locality-enhanced Projector for Multimodal LLM. *arXiv preprint arXiv:2312.06742*.

[35] Chan, D., Petryk, S., Gonzalez, J. E., Darrell, T., et al. (2023). CLAIR: Evaluating Image Captions with Large Language Models. In *Conference on Empirical Methods in Natural Language Processing*.

[36] Changpinyo, S., Sharma, P., Ding, N., and Soricut, R. (2021). Conceptual 12M: Pushing Web-Scale Image-Text Pre-Training To Recognize Long-Tail Visual Concepts. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[37] Chen, D., Liu, J., Dai, W., and Wang, B. (2023a). Visual Instruction Tuning with Polite Flamingo. *arXiv preprint arXiv:2307.01003*.

[38] Chen, G., Shen, L., Shao, R., Deng, X., and Nie, L. (2023b). LION: Empowering Multimodal Large Language Model with Dual-Level Visual Knowledge. *arXiv preprint arXiv:2311.11860*.

[39] Chen, J., Agarwal, A., Abdelkarim, S., Zhu, D., and Elhoseiny, M. (2022a). Reltransformer: A transformer-based long-tail visual relationship recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[40] Chen, J., Guo, H., Yi, K., Li, B., and Elhoseiny, M. (2022b). VisualGPT: Data-efficient Adaptation of Pretrained Language Models for Image Captioning . In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[41] Chen, J., Zhu, D., Shen, X., Li, X., Liu, Z., Zhang, P., Krishnamoorthi, R., Chandra, V., Xiong, Y., and Elhoseiny, M. (2023c). MiniGPT-v2: Large Language Model As a Unified Interface for Vision-Language Multi-task Learning. *arXiv preprint arXiv:2310.09478*.

[42] Chen, K., Zhang, Z., Zeng, W., Zhang, R., Zhu, F., and Zhao, R. (2023d). Shikra: Unleashing Multimodal LLM's Referential Dialogue Magic. *arXiv preprint arXiv:2306.15195*.

[43] Chen, Q., Deng, C., and Wu, Q. (2022c). Learning Distinct and Representative Modes for Image Captioning. In *NeurIPS*.

[44] Chen, W., Hu, H., Chen, X., Verga, P., and Cohen, W. W. (2022d). MuRAG: Multimodal Retrieval-Augmented Generator for Open Question Answering over Images and Text. In *Conference on Empirical Methods in Natural Language Processing*.

[45] Chen, W., Hu, H., Saharia, C., and Cohen, W. W. (2022e). Re-Imagen: Retrieval-Augmented Text-to-Image Generator. *arXiv preprint arXiv:2209.14491*.

[46] Chen, X., Djolonga, J., Padlewski, P., Mustafa, B., Changpinyo, S., Wu, J., Ruiz, C. R., Goodman, S., Wang, X., Tay, Y., et al. (2023e). PaLI-X: On Scaling up a Multilingual Vision and Language Model. *arXiv preprint arXiv:2305.18565*.

[47] Chen, X., Wang, X., Changpinyo, S., Piergiovanni, A., Padlewski, P., Salz, D., Goodman, S., Grycner, A., Mustafa, B., Beyer, L., et al. (2023f). PaLI: A Jointly-Scaled Multilingual Language-Image Model. In *International Conference on Learning Representations Workshop*.

[48] Chen, Y., Hu, H., Luan, Y., Sun, H., Changpinyo, S., Ritter, A., and Chang, M.-W. (2023g). Can Pre-trained Vision and Language Models Answer Visual Information-Seeking Questions? In *Conference on Empirical Methods in Natural Language Processing*.

[49] Chen, Y., Sikka, K., Cogswell, M., Ji, H., and Divakaran, A. (2023h). DRESS: Instructing Large Vision-Language Models to Align and Interact with Humans via Natural Language Feedback. *arXiv preprint arXiv:2311.10081*.

[50] Chen, Y.-C., Li, L., Yu, L., El Kholy, A., Ahmed, F., Gan, Z., Cheng, Y., and Liu, J. (2020a). Uniter: Universal image-text representation learning. In *Proceedings of the European Conference on Computer Vision*.

[51] Chen, Y.-C., Li, L., Yu, L., El Kholy, A., Ahmed, F., Gan, Z., Cheng, Y., and Liu, J. (2020b). UNITER: UNiversal Image-TExt Representation Learning. In *Proceedings of the European Conference on Computer Vision*.

[52] Cheng, Y., Zhu, X., Qian, J., Wen, F., and Liu, P. (2022). Cross-modal Graph Matching Network for Image-text Retrieval. *ACM TOMM*, 18(4):1–23.

[53] Cherti, M., Beaumont, R., Wightman, R., Wortsman, M., Ilharco, G., Gordon, C., Schuhmann, C., Schmidt, L., and Jitsev, J. (2023). Reproducible scaling laws for contrastive language-image Learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[54] Chiang, W.-L., Li, Z., Lin, Z., Sheng, Y., Wu, Z., Zhang, H., Zheng, L., Zhuang, S., Zhuang, Y., Gonzalez, J. E., Stoica, I., and Xing, E. P. (2023). Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality.

[55] Cho, J., Yoon, S., Kale, A., Dernoncourt, F., Bui, T., and Bansal, M. (2022). Fine-grained Image Captioning with CLIP Reward. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*.

[56] Chung, H. W., Hou, L., Longpre, S., Zoph, B., Tay, Y., Fedus, W., Li, Y., Wang, X., Dehghani, M., Brahma, S., et al. (2022). Scaling Instruction-Finetuned Language Models. *arXiv preprint arXiv:2210.11416*.

[57] Cocchi, F., Moratelli, N., Caffagni, D., Sarto, S., Baraldi, L., Cornia, M., and Cucchiara, R. (2025a). LLaVA-MORE: A Comparative Study of LLMs and Visual Backbones for Enhanced Visual Instruction Tuning. In *ICCV Workshops*.

[58] Cocchi, F., Moratelli, N., Cornia, M., Baraldi, L., and Cucchiara, R. (2025b). Augmenting Multimodal LLMs with Self-Reflective Tokens for Knowledge-based Visual Question Answering. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[59] Compagnoni, A., Morini, M., Sarto, S., Cocchi, F., Caffagni, D., Cornia, M., Baraldi, L., and Cucchiara, R. (2025). ReAG: Reasoning-Augmented Generation for Knowledge-based Visual Question Answering. *arXiv preprint arXiv:2511.22715*.

[60] Cornia, M., Baraldi, L., and Cucchiara, R. (2020a). SMArT: Training Shallow Memory-aware Transformers for Robotic Explainability. In *International Conference on Robotics and Automation*.

[61] Cornia, M., Baraldi, L., and Cucchiara, R. (2021). Explaining transformer-based image captioning models: An empirical analysis. *AI Communications*, pages 1–19.

[62] Cornia, M., Baraldi, L., Fiameni, G., and Cucchiara, R. (2022). Universal Captioner: Inducing Content-Style Separation in Vision-and-Language Model Training. *arXiv preprint arXiv:2111.12727*.

[63] Cornia, M., Baraldi, L., Fiameni, G., and Cucchiara, R. (2023). Generating More Pertinent Captions by Leveraging Semantics and Style on Multi-Source Datasets. *International Journal of Computer Vision*, pages 1–20.

[64] Cornia, M., Stefanini, M., Baraldi, L., and Cucchiara, R. (2020b). Meshed-Memory Transformer for Image Captioning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[65] Cui, C., Shen, J., Ma, J., and Lian, T. (2015). Social Tag Relevance Estimation via Ranking-Oriented Neighbour Voting. In *ACM International Conference on Multimedia*.

[66] Dai, W., Li, J., Li, D., Tiong, A. M. H., Zhao, J., et al. (2023). InstructBLIP: Towards General-purpose Vision-Language Models with Instruction Tuning. *arXiv:2305.06500*.

[67] Darcet, T., Oquab, M., Mairal, J., and Bojanowski, P. (2024). Vision Transformers Need Registers. In *International Conference on Learning Representations Workshop*.

[68] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*.

[69] Dong, R., Han, C., Peng, Y., Qi, Z., Ge, Z., Yang, J., Zhao, L., Sun, J., Zhou, H., Wei, H., et al. (2023). DreamLLM: Synergistic Multimodal Comprehension and Creation. *arXiv preprint arXiv:2309.11499*.

[70] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021a). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations Workshop*.

[71] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021b). An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations Workshop*.

[72] Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., et al. (2024). The Llama 3 Herd of Models. *arXiv:2407.21783*.

[73] Fang, Y., Wang, W., Xie, B., Sun, Q., Wu, L., Wang, X., Huang, T., Wang, X., and Cao, Y. (2023). EVA: Exploring the Limits of Masked Visual Representation Learning at Scale. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[74] Fruchard, B., Malacria, S., Casiez, G., and Huot, S. (2023). User Preference and Performance using Tagging and Browsing for Image Labeling. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*.

[75] Fu, C., Chen, P., Shen, Y., Qin, Y., Zhang, M., Lin, X., Yang, J., Zheng, X., Li, K., Sun, X., et al. (2023a). MME: A Comprehensive Evaluation Benchmark for Multimodal Large Language Models. *arXiv preprint arXiv:2306.13394*.

[76] Fu, S., Tamir, N., Sundaram, S., Chai, L., Zhang, R., Dekel, T., and Isola, P. (2023b). DreamSim: Learning New Dimensions of Human Visual Similarity using Synthetic Data. In *NeurIPS*.

[77] Fu, X., Hu, Y., Li, B., Feng, Y., Wang, H., Lin, X., Roth, D., Smith, N. A., Ma, W.-C., and Krishna, R. (2024). BLINK: Multimodal Large Language Models Can See But Not Perceive. In *Proceedings of the European Conference on Computer Vision*.

[78] Gadre, S. Y., Ilharco, G., Fang, A., Hayase, J., Smyrnis, G., Nguyen, T., Marten, R., Wortsman, M., Ghosh, D., Zhang, J., et al. (2023). Datacomp: In search of the next generation of multimodal datasets. In *NeurIPS*.

[79] Gafni, O., Polyak, A., Ashual, O., Sheynin, S., Parikh, D., and Taigman, Y. (2022). Make-A-Scene: Scene-Based Text-to-Image Generation with Human Priors. In *Proceedings of the European Conference on Computer Vision*.

[80] Gan, Z., Chen, Y.-C., Li, L., Zhu, C., Cheng, Y., and Liu, J. (2020). Large-scale adversarial training for vision-and-language representation learning. *NeurIPS*.

[81] Gao, B. and Pavel, L. (2017). On the Properties of the Softmax Function with Application in Game Theory and Reinforcement Learning. *arXiv preprint arXiv:1704.00805*.

[82] Gao, P., Han, J., Zhang, R., Lin, Z., Geng, S., Zhou, A., Zhang, W., Lu, P., He, C., Yue, X., et al. (2023). LLaMA-Adapter V2: Parameter-Efficient Visual Instruction Model. *arXiv preprint arXiv:2304.15010*.

[83] Gao, P., Zhang, R., Liu, C., Qiu, L., Huang, S., Lin, W., Zhao, S., Geng, S., Lin, Z., Jin, P., Zhang, K., Shao, W., Xu, C., He, C., He, J., Shao, H., Lu, P., Li, H., and Qiao, Y. (2024). SPHINX-X: Scaling Data and Parameters for a Family of Multi-modal Large Language Models. *arXiv preprint arXiv:2402.05935*.

[84] Gao, Y., Wang, M., Zha, Z.-J., Shen, J., Li, X., and Wu, X. (2012). Visual-Textual Joint Relevance Learning for Tag-Based Social Image Search. *IEEE Trans. Image Processing*, 22(1):363–376.

[85] Girdhar, R., El-Nouby, A., Liu, Z., Singh, M., Alwala, K. V., Joulin, A., and Misra, I. (2023). ImageBind: One Embedding Space To Bind Them All. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[86] Gong, T., Lyu, C., Zhang, S., Wang, Y., Zheng, M., Zhao, Q., Liu, K., Zhang, W., Luo, P., and Chen, K. (2023). MultiModal-GPT: A Vision and Language Model for Dialogue with Humans. *arXiv preprint arXiv:2305.04790*.

[87] Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., and Parikh, D. (2017). Making the v in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[88] Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., et al. (2024). The Llama 3 Herd of Models. *arXiv preprint arXiv:2407.21783*.

[89] Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., et al. (2025). DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. *arXiv preprint arXiv:2501.12948*.

[90] Guo, L., Liu, J., Zhu, X., Yao, P., Lu, S., and Lu, H. (2020). Normalized and Geometry-Aware Self-Attention Network for Image Captioning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[91] Gurari, D., Li, Q., Stangl, A. J., Guo, A., Lin, C., Grauman, K., Luo, J., and Bigham, J. P. (2018). VizWiz Grand Challenge: Answering Visual Questions From Blind People. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[92] Gurari, D., Zhao, Y., Zhang, M., and Bhattacharya, N. (2020). Captioning Images Taken by People Who Are Blind. In *Proceedings of the European Conference on Computer Vision*.

[93] Guu, K., Lee, K., Tung, Z., Pasupat, P., and Chang, M. (2020a). Retrieval augmented language model pre-training. In *International Conference on Machine Learning*.

[94] Guu, K., Lee, K., Tung, Z., Pasupat, P., and Chang, M.-W. (2020b). REALM: Retrieval-Augmented Language Model Pre-Training. In *International Conference on Machine Learning*.

[95] Han, X., Wu, Z., Huang, P. X., Zhang, X., Zhu, M., Li, Y., Zhao, Y., and Davis, L. S. (2017). Automatic Spatially-Aware Fashion Concept Discovery. In *IEEE International Conference on Computer Vision*.

[96] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

[97] Herdade, S., Kappeler, A., Boakye, K., and Soares, J. (2019). Image Captioning: Transforming Objects into Words. In *NeurIPS*.

[98] Hessel, J., Holtzman, A., Forbes, M., Bras, R. L., and Choi, Y. (2021). CLIPScore: A Reference-free Evaluation Metric for Image Captioning. *arXiv preprint arXiv:2104.08718*.

[99] Hochreiter, S. and Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9.

[100] Hodosh, M., Young, P., and Hockenmaier, J. (2013). Framing image description as a ranking task: Data, models and evaluation metrics. *JAIR*.

[101] Hong, Y., Gu, J., Yang, Q., Fan, L., Wu, Y., Wang, Y., Ding, K., Xiang, S., and Ye, J. (2025). Knowledge-based Visual Question Answer with Multimodal Processing, Retrieval and Filtering. In *NeurIPS*.

[102] Hu, A., Chen, S., Zhang, L., and Jin, Q. (2023a). InfoMetIC: An Informative Metric for Reference-free Image Caption Evaluation. In *ACL*.

[103] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. (2021). Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

[104] Hu, H., Luan, Y., Chen, Y., Khandelwal, U., Joshi, M., Lee, K., Toutanova, K., and Chang, M.-W. (2023b). Open-domain Visual Entity Recognition: Towards Recognizing Millions of Wikipedia Entities. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[105] Hu, W., Xu, Y., Li, Y., Li, W., Chen, Z., and Tu, Z. (2024). BLIVA: A Simple Multimodal LLM for Better Handling of Text-Rich Visual Questions. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[106] Hu, X., Gan, Z., Wang, J., Yang, Z., Liu, Z., Lu, Y., and Wang, L. (2022). Scaling Up Vision-Language Pre-Training for Image Captioning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[107] Hu, Z., Iscen, A., Sun, C., Wang, Z., Chang, K.-W., Sun, Y., Schmid, C., Ross, D. A., and Fathi, A. (2023c). REVEAL: Retrieval-Augmented Visual-Language Pre-Training with Multi-Source Multimodal Knowledge Memory. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[108] Huang, L., Wang, W., Chen, J., and Wei, X.-Y. (2019). Attention on Attention for Image Captioning. In *IEEE International Conference on Computer Vision*.

[109] Huang, L., Wu, Q., Miao, Z., and Yamasaki, T. (2025). Joint Fusion and Encoding: Advancing Multimodal Retrieval from the Ground Up. *arXiv preprint arXiv:2502.20008*.

[110] Huang, S., Dong, L., Wang, W., Hao, Y., Singhal, S., Ma, S., Lv, T., Cui, L., Mohammed, O. K., Liu, Q., et al. (2023). Language Is Not All You Need: Aligning Perception with Language Models. *arXiv preprint arXiv:2302.14045*.

[111] Hudson, D. A. and Manning, C. D. (2019). GQA: A New Dataset for Real-World Visual Reasoning and Compositional Question Answering. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[112] Izacard, G., Caron, M., Hosseini, L., Riedel, S., Bojanowski, P., Joulin, A., and Grave, E. (2021). Unsupervised Dense Information Retrieval with Contrastive Learning. *arXiv preprint arXiv:2112.09118*.

[113] Jia, C., Yang, Y., Xia, Y., Chen, Y.-T., Parekh, Z., Pham, H., Le, Q. V., Sung, Y., Li, Z., and Duerig, T. (2021). Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*.

[114] Jiang, M., Huang, Q., Zhang, L., Wang, X., et al. (2019). TIGEr: Text-to-Image Grounding for Image Caption Evaluation. In *Conference on Empirical Methods in Natural Language Processing*.

[115] Johnson, J., Douze, M., and Jégou, H. (2019). Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3):535–547.

[116] Joseph, F. A., Sieber, J., Zeilinger, M., and Alonso, C. A. (2025). Lambda-Skip Connections: the Architectural Component that Prevents Rank Collapse. In *International Conference on Learning Representations Workshop*.

[117] Karpathy, A. and Fei-Fei, L. (2015). Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[118] Karpukhin, V., Oğuz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., and Yih, W.-t. (2020). Dense Passage Retrieval for Open-Domain Question Answering. In *Conference on Empirical Methods in Natural Language Processing*.

[119] Kembhavi, A., Salvato, M., Kolve, E., Seo, M., Hajishirzi, H., and Farhadi, A. (2016). A diagram is worth a dozen images. In *Proceedings of the European Conference on Computer Vision*.

[120] Khandelwal, A., Weihs, L., Mottaghi, R., and Kembhavi, A. (2022). Simple but Effective: CLIP Embeddings for Embodied AI. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[121] Khandelwal, U., Levy, O., Jurafsky, D., Zettlemoyer, L., and Lewis, M. (2020). Generalization through Memorization: Nearest Neighbor Language Models. In *International Conference on Learning Representations Workshop*.

[122] Khattab, O. and Zaharia, M. (2020). ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT. In *ACM SIGIR*.

[123] Kim, J.-H., Kim, Y., Lee, J., Yoo, K. M., and Lee, S.-W. (2022). Mutual Information Divergence: A Unified Metric for Multimodal Generative Models. In *NeurIPS*.

[124] Kim, S., Zhu, X., Lin, X., Bastan, M., Gray, D., and Kwak, S. (2025). GENIUS: A Generative Framework for Universal Multimodal Search. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[125] Kingma, D. P. and Ba, J. (2015). Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations Workshop*.

[126] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., et al. (2023). Segment Anything. In *IEEE International Conference on Computer Vision*.

[127] Koh, J. Y., Salakhutdinov, R., and Fried, D. (2023). Grounding Language Models to Images for Multimodal Inputs and Outputs. In *International Conference on Machine Learning*.

[128] Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.-J., Shamma, D. A., et al. (2017). Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *International Journal of Computer Vision*, 123:32–73.

[129] Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., Kamali, S., Popov, S., Malloci, M., Duerig, T., et al. (2018). The Open Images Dataset V4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision*.

[130] Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., Kamali, S., Popov, S., Malloci, M., Kolesnikov, A., et al. (2020). The Open Images Dataset V4: Unified Image Classification, Object Detection, and Visual Relationship Detection at Scale. *International Journal of Computer Vision*, 128:1956–1981.

[131] Landi, F., Baraldi, L., Cornia, M., and Cucchiara, R. (2021). Working Memory Connections for LSTM. *Neural Networks*, 144:334–341.

[132] Laurençon, H., Marafioti, A., Sanh, V., and Tronchon, L. (2024a). Building and better understanding vision-language models: insights and future directions. In *NeurIPS Workshops*.

[133] Laurençon, H., Saulnier, L., Tronchon, L., Bekman, S., Singh, A., Lozhkov, A., Wang, T., Karamcheti, S., Rush, A., Kiela, D., et al. (2024b). Obelics: An open web-scale filtered dataset of interleaved image-text documents. In *NeurIPS*.

[134] Laurençon, H., Tronchon, L., Cord, M., and Sanh, V. (2024c). What matters when building vision-language models? In *NeurIPS*.

[135] Lee, H., Yoon, S., Dernoncourt, F., Bui, T., and Jung, K. (2021). UMIC: An Unreferenced Metric for Image Captioning via Contrastive Learning. In *ACL*.

[136] Lee, H., Yoon, S., Dernoncourt, F., Kim, D. S., Bui, T., et al. (2020). ViLBERTScore: Evaluating Image Caption Using Vision-and-Language BERT. In *EMNLP Workshops*.

[137] Lee, K.-H., Chen, X., Hua, G., Hu, H., and He, X. (2018). Stacked Cross Attention for Image-Text Matching. In *Proceedings of the European Conference on Computer Vision*.

[138] Lee, Y., Park, I., and Kang, M. (2024). FLEUR: An Explainable Reference-Free Evaluation Metric for Image Captioning Using a Large Multimodal Model. In *ACL*.

[139] Lei Ba, J., Kiros, J. R., and Hinton, G. E. (2016). Layer Normalization. *arXiv preprint arXiv:1607.06450*.

[140] Lerner, P., Ferret, O., and Guinaudeau, C. (2024). Cross-modal Retrieval for Knowledge-based Visual Question Answering. In *ECIR*.

[141] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., and Kiela, D. (2020). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *NeurIPS*.

[142] Li, A., Jabri, A., Joulin, A., and Van Der Maaten, L. (2017). Learning visual n-grams from web data. In *IEEE International Conference on Computer Vision*.

[143] Li, B., Wang, R., Wang, G., Ge, Y., Ge, Y., and Shan, Y. (2023a). SEED-Bench: Benchmarking Multimodal LLMs with Generative Comprehension. *arXiv preprint arXiv:2307.16125*.

[144] Li, B., Zhang, Y., Chen, L., Wang, J., Yang, J., and Liu, Z. (2023b). Otter: A Multi-Modal Model with In-Context Instruction Tuning. *arXiv preprint arXiv:2305.03726*.

[145] Li, G., Duan, N., Fang, Y., Gong, M., and Jiang, D. (2020a). Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. In *Proceedings of the AAAI conference on artificial intelligence*.

[146] Li, J., Li, D., Savarese, S., and Hoi, S. (2023c). BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models. In *International Conference on Machine Learning*.

[147] Li, J., Li, D., Savarese, S., and Hoi, S. (2023d). BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models. In *International Conference on Machine Learning*.

[148] Li, J., Li, D., Xiong, C., and Hoi, S. (2022a). BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation. In *International Conference on Machine Learning*.

[149] Li, W., Su, X., Song, D., Wang, L., Zhang, K., and Liu, A.-A. (2023e). Towards Deconfounded Image-Text Matching with Causal Inference. In *ACM International Conference on Multimedia*.

[150] Li, X., Yin, X., Li, C., Zhang, P., Hu, X., Zhang, L., Wang, L., Hu, H., Dong, L., Wei, F., et al. (2020b). Oscar: Object-semantics aligned pre-training for vision-language tasks. In *Proceedings of the European Conference on Computer Vision*.

[151] Li, Y., Du, Y., Zhou, K., Wang, J., Zhao, W. X., and Wen, J.-R. (2023f). Evaluating Object Hallucination in Large Vision-Language Models. In *Conference on Empirical Methods in Natural Language Processing*.

[152] Li, Y., Pan, Y., Yao, T., and Mei, T. (2022b). Comprehending and ordering semantics for image captioning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[153] Li, Y., Zhang, C., Yu, G., Wang, Z., Fu, B., Lin, G., Shen, C., Chen, L., and Wei, Y. (2023g). StableLLaVA: Enhanced Visual Instruction Tuning with Synthesized Image-Dialogue Data. *arXiv preprint arXiv:2308.10253*.

[154] Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. In *ACL Workshops*.

[155] Lin, J., Chen, H., Fan, Y., Fan, Y., Jin, X., Su, H., Fu, J., and Shen, X. (2025a). Multi-Layer Visual Feature Fusion in Multimodal LLMs: Methods, Analysis, and Best Practices. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[156] Lin, J., Yin, H., Ping, W., Lu, Y., Molchanov, P., Tao, A., Mao, H., Kautz, J., Shoeybi, M., and Han, S. (2023a). VILA: On Pre-training for Visual Language Models. *arXiv preprint arXiv:2312.07533*.

[157] Lin, S.-C., Lee, C., Shoeybi, M., Lin, J., Catanzaro, B., and Ping, W. (2025b). MM-Embed: Universal Multimodal Retrieval with Multimodal LLMs. In *International Conference on Learning Representations Workshop*.

[158] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft COCO: Common Objects in Context. In *Proceedings of the European Conference on Computer Vision*.

[159] Lin, W., Chen, J., Mei, J., Coca, A., and Byrne, B. (2023b). Fine-grained Late-interaction Multi-modal Retrieval for Retrieval Augmented Visual Question Answering. In *NeurIPS*.

[160] Lin, W., Mei, J., Chen, J., and Byrne, B. (2024). PreFLMR: Scaling Up Fine-Grained Late-Interaction Multi-modal Retrievers. In *ACL*.

[161] Lin, Z., Liu, C., Zhang, R., Gao, P., Qiu, L., Xiao, H., Qiu, H., Lin, C., Shao, W., Chen, K., et al. (2023c). SPHINX: The Joint Mixing of Weights, Tasks, and Visual Embeddings for Multi-modal Large Language Models. *arXiv preprint arXiv:2311.07575*.

[162] Liu, D., Hua, X.-S., Yang, L., Wang, M., and Zhang, H.-J. (2009). Tag Ranking. In *Proceedings of the International Conference on World Wide Web*.

[163] Liu, F., Emerson, G., and Collier, N. (2023a). Visual Spatial Reasoning. *Transactions of the Association for Computational Linguistics*, 11:635–651.

[164] Liu, F., Ren, X., Wu, X., Ge, S., Fan, W., Zou, Y., and Sun, X. (2020). Prophet Attention: Predicting Attention with Future Attention. In *NeurIPS*.

[165] Liu, F., Wang, Y., Wang, T., and Ordonez, V. (2021a). Visual News: Benchmark and Challenges in News Image Captioning. In *Conference on Empirical Methods in Natural Language Processing*.

[166] Liu, H., Li, C., Li, Y., and Lee, Y. J. (2024a). Improved Baselines with Visual Instruction Tuning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[167] Liu, H., Li, C., Li, Y., and Lee, Y. J. (2024b). Improved Baselines with Visual Instruction Tuning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[168] Liu, H., Li, C., Wu, Q., and Lee, Y. J. (2023b). Visual Instruction Tuning. In *NeurIPS*.

[169] Liu, H., Li, C., Wu, Q., and Lee, Y. J. (2024c). Visual Instruction Tuning. *NeurIPS*.

[170] Liu, S., Feng, W., Fu, T.-j., Chen, W., and Wang, W. Y. (2023c). EDIS: Entity-Driven Image Search over Multimodal Web Content. In *Conference on Empirical Methods in Natural Language Processing*.

[171] Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J., et al. (2023d). Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection. *arXiv preprint arXiv:2303.05499*.

[172] Liu, W., Chen, S., Guo, L., Zhu, X., and Liu, J. (2021b). CPTR: Full Transformer Network for Image Captioning. *arXiv preprint arXiv:2101.10804*.

[173] Liu, Y., Duan, H., Zhang, Y., Li, B., Zhang, S., Zhao, W., Yuan, Y., Wang, J., He, C., Liu, Z., et al. (2024d). MMBench: Is Your Multi-modal Model an All-around Player? In *Proceedings of the European Conference on Computer Vision*.

[174] Liu, Y., Zhang, Y., Cai, J., Jiang, X., Hu, Y., Yao, J., Wang, Y., and Xie, W. (2025). LamRA: Large Multimodal Model as Your Advanced Retrieval Assistant. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[175] Liu, Z., Rodriguez-Opazo, C., Teney, D., and Gould, S. (2021c). Image Retrieval on Real-Life Images With Pre-Trained Vision-and-Language Models. In *IEEE International Conference on Computer Vision*.

[176] Loshchilov, I. and Hutter, F. (2019a). Decoupled Weight Decay Regularization. In *International Conference on Learning Representations Workshop*.

[177] Loshchilov, I. and Hutter, F. (2019b). Decoupled Weight Decay Regularization. In *International Conference on Learning Representations Workshop*.

[178] Lu, J., Batra, D., Parikh, D., and Lee, S. (2019). ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. In *NeurIPS*.

[179] Lu, J., Clark, C., Lee, S., Zhang, Z., Khosla, S., Marten, R., Hoiem, D., and Kembhavi, A. (2023). Unified-IO 2: Scaling Autoregressive Multimodal Models with Vision, Language, Audio, and Action. *arXiv preprint arXiv:2312.17172*.

[180] Lu, J., Yang, J., Batra, D., and Parikh, D. (2018). Neural Baby Talk. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[181] Lu, P., Bansal, H., Xia, T., Liu, J., Li, C., Hajishirzi, H., Cheng, H., Chang, K.-W., Galley, M., and Gao, J. (2024). Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *International Conference on Learning Representations Workshop*.

[182] Lu, P., Mishra, S., Xia, T., Qiu, L., Chang, K.-W., Zhu, S.-C., Tafjord, O., Clark, P., and Kalyan, A. (2022). Learn to Explain: Multimodal Reasoning via Thought Chains for Science Question Answering. In *NeurIPS*.

[183] Lu, P., Qiu, L., Chen, J., Xia, T., Zhao, Y., Zhang, W., Yu, Z., Liang, X., and Zhu, S.-C. (2021). IconQA: A New Benchmark for Abstract Diagram Understanding and Visual Language Reasoning. In *NeurIPS*.

[184] Luo, G., Zhou, Y., Ren, T., Chen, S., Sun, X., and Ji, R. (2023). Cheap and Quick: Efficient Vision-Language Instruction Tuning for Large Language Models. *arXiv preprint arXiv:2305.15023*.

[185] Luo, Y., Ji, J., Sun, X., Cao, L., Wu, Y., Huang, F., Lin, C.-W., and Ji, R. (2021). Dual-Level Collaborative Transformer for Image Captioning. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[186] Lyu, Y., Shao, R., Chen, G., Zhu, Y., Guan, W., and Nie, L. (2025). PUMA: Layer-Pruned Language Model for Efficient Unified Multimodal Retrieval with Modality-Adaptive Learning. *ACM International Conference on Multimedia*.

[187] Marino, K., Rastegari, M., Farhadi, A., and Mottaghi, R. (2019). Ok-VQA: A Visual Question Answering Benchmark Requiring External Knowledge. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[188] Masry, A., Do, X. L., Tan, J. Q., Joty, S., and Hoque, E. (2022). Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In *ACL*.

[189] Materzyńska, J., Torralba, A., and Bau, D. (2022). Disentangling Visual and Written Concepts in CLIP. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[190] Matsuda, K., Wada, Y., and Sugiura, K. (2024). DENEB: A Hallucination-Robust Automatic Evaluation Metric for Image Captioning. In *ACCV*.

[191] Mensink, T., Uijlings, J., Castrejon, L., Goel, A., Cadar, F., Zhou, H., Sha, F., Araujo, A., and Ferrari, V. (2023). Encyclopedic VQA: Visual Questions About Detailed Properties of Fine-Grained Categories. In *IEEE International Conference on Computer Vision*.

[192] Messina, N., Stefanini, M., Cornia, M., Baraldi, L., Falchi, F., Amato, G., and Cucchiara, R. (2022). ALADIN: Distilling Fine-grained Alignment Scores for Efficient Image-Text Matching and Retrieval. In *International Workshop on Content-Based Multimedia Indexing*.

[193] Methani, N., Ganguly, P., Khapra, M. M., and Kumar, P. (2020). PlotQA: Reasoning over Scientific Plots. In *IEEE/CVF Winter Conference on Applications of Computer Vision*.

[194] Mialon, G., Dessì, R., Lomeli, M., Nalmpantis, C., Pasunuru, R., Raileanu, R., Rozière, B., Schick, T., Dwivedi-Yu, J., Celikyilmaz, A., Grave, E., LeCun, Y. L., and Scialom, T. (2023). Augmented Language Models: a Survey. *arXiv preprint arXiv:2302.07842*.

[195] Micikevicius, P., Narang, S., Alben, J., Diamos, G., Elsen, E., Garcia, D., Ginsburg, B., Houston, M., Kuchaiev, O., Venkatesh, G., and Wu, H. (2018). Mixed Precision Training. In *International Conference on Learning Representations Workshop*.

[196] Miech, A., Alayrac, J.-B., Laptev, I., Sivic, J., and Zisserman, A. (2021). Thinking Fast and Slow: Efficient Text-to-Visual Retrieval With Transformers. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[197] Mishra, A., Shekhar, S., Singh, A. K., and Chakraborty, A. (2019). Ocr-vqa: Visual question answering by reading text in images. In *International Conference on Document Analysis and Recognition*.

[198] MosaicML (2023). Introducing MPT-7B: A New Standard for Open-Source, Commercially Usable LLMs.

[199] Nguyen, T., Rosenberg, M., Song, X., Gao, J., Tiwary, S., Majumder, R., and Deng, L. (2016). MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. In *NeurIPS*.

[200] Nguyen, V.-Q., Suganuma, M., and Okatani, T. (2022). GRIT: Faster and Better Image captioning Transformer Using Dual Visual Features. In *Proceedings of the European Conference on Computer Vision*.

[201] Noh, H., Araujo, A., Sim, J., Weyand, T., and Han, B. (2017). Large-scale image retrieval with attentive deep local features. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[202] Oord, A. v. d., Li, Y., and Vinyals, O. (2018). Representation Learning with Contrastive Predictive Coding. In *arXiv preprint arXiv:1807.03748*.

[203] OpenAI (2022). Introducing ChatGPT.

[204] Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al. (2024). DINOv2: Learning Robust Visual Features without Supervision. *Transactions on Machine Learning Research*, pages 1–31.

[205] Ordonez, V., Kulkarni, G., and Berg, T. (2011). Im2text: Describing images using 1 million captioned photographs. In *NeurIPS*.

[206] Pan, Y., Yao, T., Li, Y., and Mei, T. (2020). X-Linear Attention Networks for Image Captioning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[207] Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: a method for automatic evaluation of machine translation. In *ACL*.

[208] Peng, Z., Wang, W., Dong, L., Hao, Y., Huang, S., Ma, S., and Wei, F. (2023). Kosmos-2: Grounding Multimodal Large Language Models to the World. *arXiv preprint arXiv:2306.14824*.

[209] Petryk, S., Chan, D. M., Kachinthaya, A., Zou, H., et al. (2024). ALOHa: A New Measure for Hallucination in Captioning Models. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*.

[210] Plummer, B. A., Wang, L., Cervantes, C. M., Caicedo, J. C., Hockenmaier, J., and Lazebnik, S. (2015). Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *IEEE International Conference on Computer Vision*.

[211] Qi, D., Su, L., Song, J., Cui, E., Bharti, T., and Sacheti, A. (2020). Imagebert: Cross-modal pre-training with large-scale weak-supervised image-text data. *arXiv preprint arXiv:2001.07966*.

[212] Qi, J., Xu, Z., Shao, R., Chen, Y., Di, J., Cheng, Y., Wang, Q., and Huang, L. (2024). RoRA-VLM: Robust Retrieval-Augmented Vision Language Models. *arXiv preprint arXiv:2410.08876*.

[213] Qi, M., Qin, J., Huang, D., Shen, Z., Yang, Y., and Luo, J. (2021). Latent memory-augmented graph transformer for visual storytelling. In *ACM International Conference on Multimedia*.

[214] Radenović, F., Tolias, G., and Chum, O. (2018). Fine-tuning CNN image retrieval with no human annotation. *IEEE Trans. PAMI*, 41(7):1655–1668.

[215] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. (2021). Learning Transferable Visual Models From Natural Language Supervision. In *International Conference on Machine Learning*.

[216] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2019). Language Models are Unsupervised Multitask Learners. *OpenAI Blog*, 1(8):9.

[217] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(1):5485–5551.

[218] Rajbhandari, S., Rasley, J., Ruwase, O., and He, Y. (2020). ZeRO: Memory optimizations Toward Training Trillion Parameter Models. In *SC*.

[219] Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. (2022). Hierarchical Text-Conditional Image Generation with CLIP Latents. In *arXiv preprint arXiv:2204.06125*.

[220] Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. (2021). Zero-Shot Text-to-Image Generation. *arXiv preprint arXiv:2102.12092*.

[221] Reed, S., Akata, Z., Lee, H., and Schiele, B. (2016). Learning deep representations of fine-grained visual descriptions. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[222] Rennie, S. J., Marcheret, E., Mroueh, Y., Ross, J., and Goel, V. (2017). Self-critical sequence training for image captioning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[223] Robertson, S. (2004). Understanding inverse document frequency: on theoretical arguments for IDF. *Journal of Documentation*, 60(5):503–520.

[224] Rohrbach, A., Hendricks, L. A., Burns, K., Darrell, T., and Saenko, K. (2018). Object Hallucination in Image Captioning. In *Conference on Empirical Methods in Natural Language Processing*.

[225] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022). High-Resolution Image Synthesis With Latent Diffusion Models. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[226] Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E., Ghasemipour, S. K. S., Ayan, B. K., Mahdavi, S. S., Lopes, R. G., Salimans, T., Ho, J., Fleet, D. J., and Norouzi, M. (2022). Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding. In *NeurIPS*.

[227] Saito, K., Sohn, K., Zhang, X., Li, C.-L., Lee, C.-Y., Saenko, K., and Pfister, T. (2023). Pic2Word: Mapping Pictures to Words for Zero-Shot Composed Image Retrieval. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[228] Santhanam, K., Khattab, O., Saad-Falcon, J., Potts, C., and Zaharia, M. (2022). ColBERTv2: Effective and Efficient Retrieval via Lightweight Late Interaction. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*.

[229] Sarto, S., Barraco, M., Cornia, M., Baraldi, L., and Cucchiara, R. (2023). Positive-Augmented Contrastive Learning for Image and Video Captioning Evaluation. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[230] Sarto, S., Cornia, M., Baraldi, L., and Cucchiara, R. (2022). Retrieval-Augmented Transformer for Image Captioning. In *International Workshop on Content-Based Multimedia Indexing*.

[231] Sarto, S., Cornia, M., Baraldi, L., and Cucchiara, R. (2024a). BRIDGE: Bridging Gaps in Image Captioning Evaluation with Stronger Visual Cues. In *Proceedings of the European Conference on Computer Vision*.

[232] Sarto, S., Cornia, M., Baraldi, L., Nicolosi, A., and Cucchiara, R. (2024b). Towards Retrieval-Augmented Architectures for Image Captioning. *ACM TOMM*, 20(8):1–22.

[233] Sarto, S., Cornia, M., and Cucchiara, R. (2025). Image Captioning Evaluation in the Age of Multimodal LLMs: Challenges and Future Perspectives. *International Joint Conference on Artificial Intelligence*.

[234] Sarto, S., Moratelli, N., Cornia, M., Baraldi, L., et al. (2024c). Positive-Augmented Contrastive Learning for Vision-and-Language Evaluation and Training. *International Journal of Computer Vision*.

[235] Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., Schramowski, P., Kundurthy, S., Crowson, K., Schmidt, L., Kaczmarczyk, R., and Jitsev, J. (2022). LAION-5B: An open large-scale dataset for training next generation image-text models. In *NeurIPS*.

[236] Schuhmann, C., Vencu, R., Beaumont, R., Kaczmarczyk, R., Mullis, C., Katta, A., Coombes, T., Jitsev, J., and Komatsuzaki, A. (2021). LAION-400M: Open Dataset of CLIP-Filtered 400 Million Image-Text Pairs. In *NeurIPS Workshops*.

[237] Schwenk, D., Khandelwal, A., Clark, C., Marino, K., and Mottaghi, R. (2022). A-OKVQA: A Benchmark for Visual Question Answering Using World Knowledge. In *Proceedings of the European Conference on Computer Vision*.

[238] Sennrich, R., Haddow, B., and Birch, A. (2016). Neural Machine Translation of Rare Words with Subword Units. In *ACL*.

[239] Shah, S., Mishra, A., Yadati, N., and Talukdar, P. P. (2019). KVQA: Knowledge-aware Visual Question Answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[240] Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y., Wu, Y., et al. (2024). Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

[241] Sharma, P., Ding, N., Goodman, S., and Soricut, R. (2018). Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*.

[242] Shekhar, R., Pezzelle, S., Klimovich, Y., Herbelot, A., Nabi, M., Sangineto, E., et al. (2017). FOIL it! Find One mismatch between Image and Language caption. In *ACL*.

[243] Shen, J., Wang, M., and Chua, T. (2016). Accurate online video tagging via probabilistic hybrid modeling. *Multimedia Systems*, 22:99–113.

[244] Shen, J., Wang, M., Yan, S., and Hua, X.-S. (2011). Multimedia Tagging: Past, Present and Future. In *ACM International Conference on Multimedia*.

[245] Shen, S., Li, L. H., Tan, H., Bansal, M., Rohrbach, A., Chang, K.-W., Yao, Z., and Keutzer, K. (2021). How Much Can CLIP Benefit Vision-and-Language Tasks? *arXiv preprint arXiv:2107.06383*.

[246] Shi, Y., Yang, X., Xu, H., Yuan, C., Li, B., Hu, W., and Zha, Z.-J. (2022). EMScore: Evaluating Video Captioning via Coarse-Grained and Fine-Grained Embedding Matching. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[247] Shi, Z., Zhou, X., Qiu, X., and Zhu, X. (2020). Improving Image Captioning with Better Use of Captions. In *ACL*.

[248] Singh, A., Natarajan, V., Shah, M., Jiang, Y., Chen, X., Batra, D., Parikh, D., and Rohrbach, M. (2019). Towards VQA Models That Can Read. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[249] Socher, R. and Fei-Fei, L. (2010). Connecting modalities: Semi-supervised segmentation and annotation of images using unaligned text corpora. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[250] Srinivasan, K., Raman, K., Chen, J., Bendersky, M., and Najork, M. (2021). WIT: Wikipedia-based Image Text Dataset for Multimodal Multilingual Machine Learning. *arXiv preprint arXiv:2103.01913*.

[251] Stefanini, M., Cornia, M., Baraldi, L., Cascianelli, S., Fiameni, G., and Cucchiara, R. (2022). From Show to Tell: A Survey on Deep Learning-based Image Captioning. *IEEE Trans. PAMI*, 45(1):539–559.

[252] Sukhbaatar, S., Grave, E., Lample, G., Jegou, H., and Joulin, A. (2019). Augmenting self-attention with persistent memory. *arXiv preprint arXiv:1907.01470*.

[253] Sun, Q., Cui, Y., Zhang, X., Zhang, F., Yu, Q., Luo, Z., Wang, Y., Rao, Y., Liu, J., Huang, T., et al. (2023a). Generative Multimodal Models are In-Context Learners. *arXiv preprint arXiv:2312.13286*.

[254] Sun, Q., Fang, Y., Wu, L., Wang, X., and Cao, Y. (2023b). EVA-CLIP: Improved Training Techniques for CLIP at Scale. *arXiv preprint arXiv:2303.15389*.

[255] Sun, Q., Wang, J., Yu, Q., Cui, Y., Zhang, F., Zhang, X., and Wang, X. (2024a). EVA-CLIP-18B: Scaling CLIP to 18 Billion Parameters. *arXiv preprint arXiv:2402.04252*.

[256] Sun, Q., Yu, Q., Cui, Y., Zhang, F., Zhang, X., Wang, Y., Gao, H., Liu, J., Huang, T., and Wang, X. (2023c). Generative Pretraining in Multimodality. *arXiv preprint arXiv:2307.05222*.

[257] Sun, Z., Fang, Y., Wu, T., Zhang, P., Zang, Y., Kong, S., et al. (2024b). Alpha-CLIP: A CLIP Model Focusing on Wherever You Want. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[258] Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *NeurIPS*.

[259] Taori, R., Gulrajani, I., Zhang, T., Dubois, Y., Li, X., Guestrin, C., Liang, P., and Hashimoto, T. B. (2023). Stanford Alpaca: An Instruction-Following LLaMA Model.

[260] Team, G., Riviere, M., Pathak, S., Sessa, P. G., Hardin, C., Bhupatiraju, S., Hussenot, L., Mesnard, T., Shahriari, B., Ramé, A., et al. (2024). Gemma 2: Improving Open Language Models at a Practical Size. *arXiv preprint arXiv:2408.00118*.

[261] Thomee, B., Shamma, D. A., Friedland, G., Elizalde, B., Ni, K., Poland, D., Borth, D., and Li, L.-J. (2016). YFCC100M: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73.

[262] Tolias, G., Sicre, R., and Jégou, H. (2016). Particular object retrieval with integral max-pooling of cnn activations. In *International Conference on Learning Representations Workshop*.

[263] Tong, P., Brown, E., Wu, P., Woo, S., IYER, A. J. V., Akula, S. C., Yang, S., Yang, J., Middepogu, M., Wang, Z., et al. (2024a). Cambrian-1: A Fully Open, Vision-Centric Exploration of Multimodal LLMs. In *NeurIPS*.

[264] Tong, S., Liu, Z., Zhai, Y., Ma, Y., LeCun, Y., and Xie, S. (2024b). Eyes Wide Shut? Exploring the Visual Shortcomings of Multimodal LLMs. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[265] Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., and Jégou, H. (2021). Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*.

[266] Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. (2023a). LLaMA: Open and Efficient Foundation Language Models. *arXiv preprint arXiv:2302.13971*.

[267] Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. (2023b). Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

[268] Tschannen, M., Gritsenko, A., Wang, X., Naeem, M. F., Alabdulmohsin, I., Parthasarathy, N., Evans, T., Beyer, L., Xia, Y., Mustafa, B., et al. (2025a). SigLIP 2: Multilingual Vision-Language Encoders with Improved Semantic Understanding, Localization, and Dense Features. *arXiv preprint arXiv:2502.14786*.

[269] Tschannen, M., Gritsenko, A., Wang, X., Naeem, M. F., Alabdulmohsin, I., Parthasarathy, N., Evans, T., Beyer, L., Xia, Y., Mustafa, B., et al. (2025b). SigLIP 2: Multilingual Vision-Language Encoders with Improved Semantic Understanding, Localization, and Dense Features. *arXiv preprint arXiv:2502.14786*.

[270] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *NeurIPS*.

[271] Vedantam, R., Lawrence Zitnick, C., and Parikh, D. (2015). CIDEr: Consensus-based Image Description Evaluation. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[272] Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2015). Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[273] Vyas, A., Katharopoulos, A., and Fleuret, F. (2020). Fast transformers with clustered attention. In *NeurIPS*.

[274] Wada, Y., Kaneda, K., Saito, D., and Sugiura, K. (2024). Polos: Multimodal Metric Learning from Human Feedback for Image Captioning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[275] Wang, H., Ma, S., Huang, S., Dong, L., Wang, W., Peng, Z., Wu, Y., Bajaj, P., Singhal, S., Benhaim, A., et al. (2023a). Magneto: A Foundation Transformer. In *International Conference on Machine Learning*.

[276] Wang, S., Yao, Z., Wang, R., Wu, Z., and Chen, X. (2021a). FAIEr: Fidelity and Adequacy Ensured Image Caption Evaluation. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[277] Wang, W., Lv, Q., Yu, W., Hong, W., Qi, J., Wang, Y., Ji, J., Yang, Z., Zhao, L., Song, X., et al. (2023b). CogVLM: Visual Expert for Pretrained Language Models. *arXiv preprint arXiv:2311.03079*.

[278] Wang, Z., Yu, J., Yu, A. W., Dai, Z., Tsvetkov, Y., and Cao, Y. (2021b). SimVLM: Simple visual language model pretraining with weak supervision. *arXiv preprint arXiv:2108.10904*.

[279] Wang, Z., Yu, J., Yu, A. W., Dai, Z., Tsvetkov, Y., and Cao, Y. (2022). SimVLM: Simple Visual Language Model Pretraining with Weak Supervision. In *International Conference on Learning Representations Workshop*.

[280] Wei, C., Chen, Y., Chen, H., Hu, H., Zhang, G., Fu, J., Ritter, A., and Chen, W. (2024). UniIR: Training and Benchmarking Universal Multimodal Information Retrievers. In *Proceedings of the European Conference on Computer Vision*.

[281] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. M. (2020). Transformers: State-of-the-Art Natural Language Processing. In *Conference on Empirical Methods in Natural Language Processing*.

[282] Wortsman, M., Ilharco, G., Kim, J. W., Li, M., Kornblith, S., Roelofs, R., Lopes, R. G., Hajishirzi, H., Farhadi, A., Namkoong, H., and Schmidt, L. (2022). Robust Fine-Tuning of Zero-Shot Models. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[283] Wu, H., Gao, Y., Guo, X., Al-Halah, Z., Rennie, S., Grauman, K., and Feris, R. (2021). Fashion IQ: A New Dataset Towards Retrieving Images by Natural Language Feedback. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[284] Wu, M., Zhang, X., Sun, X., Zhou, Y., Chen, C., Gu, J., Sun, X., and Ji, R. (2022a). DIFNet: Boosting Visual Information Flow for Image Captioning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[285] Wu, Q., Teney, D., Wang, P., Shen, C., Dick, A., and Van Den Hengel, A. (2017). Visual question answering: A survey of methods and datasets. *Computer Vision and Image Understanding*, 163:21–40.

[286] Wu, W., Song, Z., Zhou, K., Shao, Y., Hu, Z., and Huang, B. (2025). Towards General Continuous Memory for Vision-Language Models. *arXiv preprint arXiv:2505.17670*.

[287] Wu, Y., Rabe, M. N., Hutchins, D., and Szegedy, C. (2022b). Memorizing Transformers. In *International Conference on Learning Representations Workshop*.

[288] xAI (2024). Grok.

[289] Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., Zemel, R. S., and Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning*.

[290] Xuan, S., Guo, Q., Yang, M., and Zhang, S. (2023). Pink: Unveiling the Power of Referential Comprehension for Multi-modal LLMs. *arXiv preprint arXiv:2310.00582*.

[291] Xue, D., Qian, S., Fang, Q., and Xu, C. (2022). Mmt: Image-guided story ending generation with multimodal memory transformer. In *ACM International Conference on Multimedia*.

[292] Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A., and Raffel, C. (2020). mT5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2010.11934*.

[293] Yan, Y. and Xie, W. (2024). EchoSight: Advancing Visual-Language Models with Wiki Knowledge. In *EMNLP Findings*.

[294] Yang, W., Fu, J., Wang, R., Wang, J., Song, L., and Bian, J. (2025). OMGM: Orchestrate Multiple Granularities and Modalities for Efficient Multimodal Retrieval. In *ACL*.

[295] Yang, X., Tang, K., Zhang, H., and Cai, J. (2019a). Auto-Encoding Scene Graphs for Image Captioning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[296] Yang, X., Zhang, H., and Cai, J. (2019b). Learning to Collocate Neural Modules for Image Captioning. In *IEEE International Conference on Computer Vision*.

[297] Yao, B. Z., Yang, X., Lin, L., Lee, M. W., and Zhu, S.-C. (2010). I2T: Image parsing to text description. *Proceedings of the IEEE*.

[298] Yao, T., Li, Y., Li, Y., Zhu, Y., Wang, G., and Yue, J. (2023). Cross-Modal Semantically Augmented Network for Image-Text Matching. *ACM TOMM*.

[299] Yao, T., Pan, Y., Li, Y., and Mei, T. (2018). Exploring Visual Relationship for Image Captioning. In *Proceedings of the European Conference on Computer Vision*.

[300] Yao, Z., Wang, R., and Chen, X. (2024). HiFi-Score: Fine-Grained Image Description Evaluation with Hierarchical Parsing Graphs. In *Proceedings of the European Conference on Computer Vision*.

[301] Ye, Q., Xu, H., Xu, G., Ye, J., Yan, M., Zhou, Y., Wang, J., Hu, A., Shi, P., Shi, Y., et al. (2023a). mPLUG-Owl: Modularization Empowers Large Language Models with Multimodality. *arXiv preprint arXiv:2304.14178*.

[302] Ye, Q., Xu, H., Ye, J., Yan, M., Liu, H., Qian, Q., Zhang, J., Huang, F., and Zhou, J. (2023b). mPLUG-Owl2: Revolutionizing Multi-modal Large Language Model with Modality Collaboration. *arXiv preprint arXiv:2311.04257*.

[303] Yi, Y., Deng, H., and Hu, J. (2020). Improving Image Captioning Evaluation by Considering Inter References Variance. In *ACL*.

[304] You, H., Zhang, H., Gan, Z., Du, X., Zhang, B., Wang, Z., Cao, L., Chang, S.-F., and Yang, Y. (2023). Ferret: Refer and Ground Anything Anywhere at Any Granularity. *arXiv preprint arXiv:2310.07704*.

[305] You, Q., Jin, H., Wang, Z., Fang, C., and Luo, J. (2016). Image captioning with semantic attention. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[306] You, Y., Li, J., Reddi, S., Hseu, J., Kumar, S., Bhojanapalli, S., Song, X., Demmel, J., Keutzer, K., and Hsieh, C.-J. (2020). Large Batch Optimization for Deep Learning: Training BERT in 76 minutes. In *International Conference on Learning Representations Workshop*.

[307] Young, P., Lai, A., Hodosh, M., and Hockenmaier, J. (2014). From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*.

[308] Yu, F., Tang, J., Yin, W., Sun, Y., Tian, H., Wu, H., and Wang, H. (2021). Ernie-vil: Knowledge enhanced vision-language representations through scene graphs. In *Proceedings of the AAAI conference on artificial intelligence*.

[309] Yu, Q., Zhang, Z., Zhu, R., Yuan, Y., Zuo, X., Yue, Y., Dai, W., Fan, T., Liu, G., Liu, L., et al. (2025). DAPO: An Open-Source LLM Reinforcement Learning System at Scale. *arXiv preprint arXiv:2503.14476*.

[310] Yuan, X., Ning, L., Fan, W., and Li, Q. (2025). mKG-RAG: Multimodal Knowledge Graph-Enhanced RAG for Visual Question Answering. *arXiv preprint arXiv:2508.05318*.

[311] Yue, X., Ni, Y., Zhang, K., Zheng, T., Liu, R., Zhang, G., Stevens, S., Jiang, D., Ren, W., Sun, Y., et al. (2024). MMMU: A Massive Multi-discipline Multimodal Understanding and Reasoning Benchmark for Expert AGI. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[312] Zeng, Z., Sun, J., Zhang, H., Wen, T., Su, Y., Xie, Y., Wang, Z., and Chen, B. (2024). HICEScore: A Hierarchical Metric for Image Captioning Evaluation. In *ACM International Conference on Multimedia*.

[313] Zha, Z.-J., Wang, M., Shen, J., and Chua, T.-S. (2012). Text Mining in Multimedia. *Mining Text Data*, pages 361–384.

[314] Zhai, X., Mustafa, B., Kolesnikov, A., and Beyer, L. (2023). Sigmoid Loss for Language Image Pre-Training. In *IEEE International Conference on Computer Vision*.

[315] Zhang, B., Zhang, P., Dong, X., Zang, Y., and Wang, J. (2024a). Long-CLIP: Unlocking the Long-Text Capability of CLIP. In *Proceedings of the European Conference on Computer Vision*.

[316] Zhang, P., Li, X., Hu, X., Yang, J., Zhang, L., Wang, L., Choi, Y., and Gao, J. (2021a). VinVL: Revisiting visual representations in vision-language models. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[317] Zhang, S., Roller, S., Goyal, N., Artetxe, M., Chen, M., Chen, S., Dewan, C., Diab, M., Li, X., Lin, X. V., et al. (2022). Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

[318] Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., and Artzi, Y. (2020). BERTScore: Evaluating Text Generation with BERT. In *International Conference on Learning Representations Workshop*.

[319] Zhang, T., Zhang, Z., Ma, Z., Chen, Y., Qi, Z., Yuan, C., Li, B., Pu, J., Zhao, Y., Xie, Z., et al. (2024b). mR^{2}AG: Multimodal Retrieval-Reflection-Augmented Generation for Knowledge-Based VQA. *arXiv preprint arXiv:2411.15041*.

[320] Zhang, X., Sun, X., Luo, Y., Ji, J., Zhou, Y., Wu, Y., Huang, F., and Ji, R. (2021b). RSTNet: Captioning with Adaptive Attention on Visual and Non-Visual Words. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[321] Zhao, B., Wu, B., and Huang, T. (2023a). SVIT: Scaling up Visual Instruction Tuning. *arXiv preprint arXiv:2307.04087*.

[322] Zhao, Z., Guo, L., Yue, T., Chen, S., Shao, S., Zhu, X., Yuan, Z., and Liu, J. (2023b). ChatBridge: Bridging Modalities with Large Language Model as a Language Catalyst. *arXiv preprint arXiv:2305.16103*.

[323] Zheng, L., Yang, Y., and Tian, Q. (2017). SIFT meets CNN: A decade survey of instance retrieval. *IEEE Trans. PAMI*, 40(5):1224–1244.

[324] Zhou, B., Zhao, H., Puig, X., Xiao, T., Fidler, S., Barriuso, A., and Torralba, A. (2019). Semantic Understanding of Scenes through the ADE20K Dataset. *International Journal of Computer Vision*, 127(3):302–321.

[325] Zhou, L., Palangi, H., Zhang, L., Hu, H., Corso, J. J., and Gao, J. (2020). Unified Vision-Language Pre-Training for Image Captioning and VQA. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[326] Zhu, D., Chen, J., Shen, X., Li, X., and Elhoseiny, M. (2023). MiniGPT-4: Enhancing Vision-Language Understanding with Advanced Large Language Models. *arXiv preprint arXiv:2304.10592*.

[327] Zhu, X., Wang, W., Guo, L., and Liu, J. (2020). AutoCaption: Image Captioning with Neural Architecture Search. *arXiv preprint arXiv:2012.09742*.