

# DECAP: DECODING CLIP LATENTS FOR ZERO-SHOT CAPTIONING VIA TEXT-ONLY TRAINING

Wei Li<sup>1</sup> Linchao Zhu<sup>1</sup> Longyin Wen<sup>2</sup> Yi Yang<sup>1\*</sup>

<sup>1</sup>CCAI, Zhejiang University <sup>2</sup>ByteDance Inc., San Jose, USA

{weili6,zhulinchao,yangyics}@zju.edu.cn longyin.wen@bytedance.com

## ABSTRACT

Large-scale pre-trained multi-modal models (e.g., CLIP) demonstrate strong zero-shot transfer capability in many discriminative tasks, e.g., image classification. Their adaptation to zero-shot image-conditioned text generation tasks has drawn increasing interest. Prior arts approach to zero-shot captioning by either utilizing the existing large language models (e.g., GPT-2) or pre-training the encoder-decoder network in an end-to-end manner. However, the large language models may not generate sensible descriptions due to the task discrepancy between captioning and language modeling, while the end-to-end pre-training requires paired data and extensive computational resources. In this work, we propose a simple framework, named DeCap, for zero-shot captioning. We introduce a lightweight visual-aware language decoder. This decoder is both data-efficient and computation-efficient: 1) it only requires the *text* data for training, easing the burden on the collection of paired data. 2) it does not require end-to-end training. When trained with text-only data, the decoder takes the text embedding extracted from the off-the-shelf CLIP encoder as a prefix embedding. The challenge is that the decoder is trained on the text corpus but at the inference stage, it needs to generate captions based on visual inputs. Though the CLIP text embedding and the visual embedding are correlated, the *modality gap* issue is widely observed in multi-modal contrastive models that prevents us from directly taking the visual embedding as the prefix embedding. We propose a training-free mechanism to reduce the modality gap. We project the visual embedding into the CLIP text embedding space, while the projected embedding retains the information of the visual input. Taking the projected embedding as the prefix embedding, the decoder generates high-quality descriptions that match the visual input. The experiments show that DeCap outperforms other zero-shot captioning methods and unpaired captioning methods by a large margin on the typical image captioning benchmarks, i.e., MSCOCO and NoCaps. We apply DeCap to video captioning and achieve state-of-the-art zero-shot performance on MSR-VTT and ActivityNet-Captions. The code is available at <https://github.com/dhg-wei/DeCap>.

## 1 INTRODUCTION

The goal of image captioning is to automatically generate descriptions for given images. Models (Anderson et al., 2018; Lu et al., 2017; Rennie et al., 2017; Zhang et al., 2021; Huang et al., 2021) trained on human-annotated image-text pairs have achieved impressive results on typical image captioning benchmarks. However, due to the small size and limited visual concepts of human-annotated datasets, these models generalize poorly to images in the wild (Agrawal et al., 2019; Tran et al., 2016; Wu et al., 2018). In this paper, to reduce the reliance on human-annotated paired data and improve the generalization in real-world captioning scenarios, we propose a new zero-shot captioning framework that requires text-only data for training.

Pre-training on web-scale noisy paired data has been demonstrated to be effective in learning robust multi-modal representations (Radford et al., 2021; Jia et al., 2021; Li et al., 2021; Alayrac et al., 2022; Yu et al., 2022a; Wang et al., 2022; Zhu & Yang, 2020). Changpinyo et al. (2021) and

## DECAP: GIẢI MÃ CLIP ẨN CHO ZERO-SHOT

## CHỈ DẪN QUA ĐÀO TẠO CHỈ VĂN BẢN

Wei Li<sup>1</sup> Linchao Zhu<sup>1</sup> Longyin Wen<sup>2</sup> Yi Yang<sup>1</sup> CCAI, Đại học Chiết

Giang 2ByteDance Inc., San Jose, Hoa Kỳ {weili6,zhulinchao,yangyics}

@zju.edu.cn longyin.wen@bytedance.com

## TÓM TẮT

Các mô hình đa phương thức được đào tạo trước quy mô lớn (ví dụ: CLIP) chứng minh khả năng truyền dữ liệu không cần xử lý mạnh mẽ trong nhiều tác vụ phân biệt, ví dụ: phân loại hình ảnh. Việc thích ứng của họ với các tác vụ tạo văn bản có điều kiện hình ảnh zero-shot đã thu hút sự quan tâm ngày càng tăng. Phương pháp tiếp cận nghệ thuật trước đây đối với chủ thích zero-shot bằng cách sử dụng các mô hình ngôn ngữ lớn hiện có (ví dụ: GPT-2) hoặc đào tạo trước mạng mã hóa-giải mã theo cách đầu cuối. Tuy nhiên, các mô hình ngôn ngữ lớn có thể không tạo ra các mô tả hợp lý do sự khác biệt về tác vụ giữa chủ thích và mô hình hóa ngôn ngữ, trong khi đào tạo trước đầu cuối yêu cầu dữ liệu được ghép nối và các tài nguyên tính toán mở rộng. Trong tác phẩm này, chúng tôi đề xuất một khuôn khổ đơn giản, có tên là DeCap, để tạo chủ thích zero-shot. Chúng tôi giới thiệu một bộ giải mã ngôn ngữ nhận biết hình ảnh nhẹ. Bộ giải mã này vừa hiệu quả về dữ liệu vừa hiệu quả về tính toán: 1) nó chỉ yêu cầu dữ liệu văn bản để đào tạo, giúp giảm bớt gánh nặng cho việc thu thập dữ liệu được ghép nối. 2) nó không yêu cầu đào tạo đầu cuối.

Khi được đào tạo với dữ liệu chỉ có văn bản, bộ giải mã sẽ lấy những văn bản được trích xuất từ bộ mã hóa CLIP có sẵn làm những tiền tố. Thách thức là bộ giải mã được đào tạo trên ngữ liệu văn bản nhưng ở giai đoạn suy luận, nó cần tạo chủ thích dựa trên đầu vào trực quan. Mặc dù những văn bản CLIP và những trực quan có tương quan, vẫn đề khoáng cách phương thức được quan sát thấy rộng rãi trong các mô hình tương phản đa phương thức ngắn chung ta trực tiếp lấy những trực quan làm những tiền tố. Chúng tôi đề xuất một cơ chế không cần đào tạo để giảm khoảng cách phương thức. Chúng tôi chiếu những trực quan vào không gian những văn bản CLIP, trong khi những được chiếu giữ lại thông tin của đầu vào trực quan. Lấy những được chiếu làm những tiền tố, bộ giải mã tạo ra các mô tả chất lượng cao khớp với đầu vào trực quan. Các thí nghiệm cho thấy DeCap vượt trội hơn các phương pháp chủ thích không có cảnh quay khác và các phương pháp chủ thích không ghép nối với biên độ lớn trên các chuẩn mực chủ thích hình ảnh thông thường, tức là MSCOCO và NoCaps. Chúng tôi áp dụng DeCap vào phụ đề video và đạt được hiệu suất zero-shot tiên tiến nhất trên MSR-VTT và ActivityNet-Captions. Mã có sẵn tại <https://github.com/dhg-wei/DeCap>.

## 1 GIỚI THIỆU

Mục tiêu của chủ thích hình ảnh là tự động tạo mô tả cho các hình ảnh nhất định. Các mô hình (Anderson và cộng sự, 2018; Lu và cộng sự, 2017; Rennie và cộng sự, 2017; Zhang và cộng sự, 2021; Huang và cộng sự, 2021) được đào tạo trên các cặp hình ảnh-văn bản có chủ thích của con người đã đạt được kết quả ấn tượng trên các chuẩn mực chủ thích hình ảnh thông thường. Tuy nhiên, do kích thước nhỏ và các khái niệm trực quan hạn chế của các tập dữ liệu có chủ thích của con người, các mô hình này tổng quát hóa kém đối với các hình ảnh trong tự nhiên (Agrawal và cộng sự, 2019; Tran và cộng sự, 2016; Wu và cộng sự, 2018). Trong bài báo này, để giảm sự phụ thuộc vào dữ liệu ghép nối có chủ thích của con người và cải thiện khả năng tổng quát hóa trong các tình huống chủ thích trong thế giới thực, chúng tôi đề xuất một khuôn khổ chủ thích zero-shot mới yêu cầu dữ liệu chỉ có văn bản để đào tạo.

Việc đào tạo trước trên dữ liệu ghép nối nhiều quy mô web đã được chứng minh là có hiệu quả trong việc học các biểu diễn đa phương thức mạnh mẽ (Radford và cộng sự, 2021; Jia và cộng sự, 2021; Li và cộng sự, 2021; Alayrac và cộng sự, 2022; Yu và cộng sự, 2022a; Wang và cộng sự, 2022; Zhu & Yang, 2020). Changpinyo và cộng sự (2021) và

Yi Yang là tác giả liên hệ.

Wang et al. (2021b) use web-scale image-text pairs to train a captioning model and achieve great improvements on MSCOCO (Chen et al., 2015) and NoCaps (Agrawal et al., 2019) through the pretraining-finetuning paradigm. However, these models show inferior zero-shot captioning performance on MSCOCO, indicating that these methods still rely on human-annotated paired data for fine-tuning. Besides, training with the captioning objective on web-scale data is not efficient, e.g., Wang et al. (2021b) train their model on ALIGN (Jia et al., 2021) and C4 (Raffel et al., 2020) about 1M steps using 512 TPU v3 chips (Jouppi et al., 2017).

Instead of directly training a captioning model in an end-to-end manner on web-scale image-text pairs, another line of work (Tewel et al., 2022b; Su et al., 2022) achieves zero-shot captioning by combining existing pre-trained models. Specifically, they use a pre-trained multi-modal model CLIP (Radford et al., 2021) to guide a pre-trained language model (PLM), i.e., GPT-2 (Radford et al., 2019), to generate sentences that match the given image. However, the inference speed of these methods is slow because each word generation involves a CLIP text encoder forward. Besides, language models pre-trained on various documents from webpages do not match well with captioning tasks that aim to describe visual concepts and their relationships in a given image, resulting in inferior performance on image captioning benchmarks.

In this paper, we propose a new framework, named DeCap, for zero-shot captioning. We aim to decode sensible visual descriptions from the CLIP multi-modal embedding space. We do not use paired image-text data during the decoder pre-training but only leverage the text data. This is more flexible and efficient when the alignment between images and texts became noisier. Our DeCap framework is described below: During **pre-training**, the text decoder is trained from scratch. The goal is to invert the CLIP text encoder, i.e., a sentence is first encoded into an embedding by the CLIP text encoder and later reconstructed by our text decoder. The decoder takes the text embedding obtained from the CLIP text encoder as the prefix embedding. During **zero-shot inference**, the difficulty lies in how to obtain a prefix embedding that can match the input image and be well decoded by the decoder. The modality gap phenomenon (Liang et al., 2022b) is observed in multi-modal contrastive models which prevents us from directly taking the visual embedding as the prefix embedding. Ramesh et al. (2022) use paired data to learn a model to map the text embedding to a corresponding image embedding. Instead of learning a model, we propose a training-free mechanism to project the image embedding into the CLIP text embedding space. Combining the text decoder with the projection mechanism, we generate high-quality descriptions for given images.

Our main contributions are summarized as follows:

(1) We propose a new framework for zero-shot captioning. Our DeCap framework contains a pre-trained contrastive model (i.e., CLIP) and a lightweight visual-aware language decoder taking the CLIP embedding as input. Though our decoder is trained only on the text corpus, it can associate both the visual embedding and the text embedding, thanks to the encoded multi-modal correlation in the CLIP embedding space.

(2) We propose a training-free projection mechanism to reduce the *modality gap* in CLIP multi-modal embedding space. We incorporate a simple support memory containing embeddings of the text corpus in the pre-training stage. We project a visual embedding into the CLIP text embedding space via the support memory. Experiments show that our proposed mechanism effectively reduces the modality gap and significantly improves performance.

(3) Extensive experiments demonstrate DeCap can flexibly apply to various captioning scenarios. DeCap outperforms other zero-shot captioning methods by a large margin on image captioning benchmarks MSCOCO and NoCaps. DeCap trained on text-only data outperforms other unpaired captioning methods on MSCOCO and Flickr30k. We apply DeCap to video captioning and achieve state-of-the-art zero-shot results on MSR-VTT and ActivityNet-Captions.

## 2 RELATED WORK

**CLIP in Captioning.** Vision-language models (Radford et al., 2021; Jia et al., 2021; Yang et al., 2022) trained with a contrastive loss show impressive ability in many discriminative tasks. However, due to the absence of a text decoder during pre-training, these models can not be directly applied to generative tasks, e.g., captioning. Prior work (Mokady et al., 2021; Barraco et al., 2022; Shen et al., 2022) has applied CLIP to the image captioning task as a visual encoder. However, they ignore

Wang et al. (2021b) sử dụng các cặp hình ảnh-văn bản quy mô web để huấn luyện mô hình chủ thích và đạt được những cải tiến lớn trên MSCOCO (Chen et al., 2015) và NoCaps (Agrawal et al., 2019) thông qua mô hình tiền huấn luyện-tinh chỉnh. Tuy nhiên, các mô hình này cho thấy hiệu suất chủ thích zero-shot kém hơn trên MSCOCO, cho thấy các phương pháp này vẫn dựa vào dữ liệu ghép nối có chủ thích của con người để tinh chỉnh. Bên cạnh đó, việc huấn luyện với mục tiêu chủ thích trên dữ liệu quy mô web không hiệu quả, ví dụ, Wang et al. (2021b) huấn luyện mô hình của họ trên ALIGN (Jia et al., 2021) và C4 (Raffel et al., 2020) khoảng 1 triệu bước bằng cách sử dụng 512 chip TPU v3 (Jouppi et al., 2017).

Thay vì trực tiếp đào tạo một mô hình chủ thích theo cách đầu cuối trên các cặp hình ảnh-văn bản quy mô web, một hướng nghiên cứu khác (Tewel và cộng sự, 2022b; Su và cộng sự, 2022) đạt được chủ thích zero-shot bằng cách kết hợp các mô hình được đào tạo trước hiện có. Cụ thể, họ sử dụng một mô hình đa phương thức được đào tạo trước CLIP (Radford và cộng sự, 2021) để hướng dẫn một mô hình ngôn ngữ được đào tạo trước (PLM), tức là GPT-2 (Radford và cộng sự, 2019), để tạo ra các câu khớp với hình ảnh đã cho. Tuy nhiên, tốc độ suy luận của các phương pháp này chậm vì mỗi lần tạo từ đều liên quan đến bộ mã hóa văn bản CLIP chuyển tiếp. Bên cạnh đó, các mô hình ngôn ngữ được đào tạo trước trên nhiều tài liệu khác nhau từ các trang web không phù hợp với các tác vụ chủ thích nhằm mục đích mô tả các khái niệm trực quan và mối quan hệ của chúng trong một hình ảnh nhất định, dẫn đến hiệu suất kém hơn trên các điểm chuẩn chủ thích hình ảnh.

Trong bài báo này, chúng tôi đề xuất một khuôn khổ mới, có tên là DeCap, cho chủ thích zero-shot. Chúng tôi muốn giải mã các mô tả trực quan hợp lý từ không gian nhúng đa phương thức CLIP. Chúng tôi không sử dụng dữ liệu hình ảnh-văn bản được ghép nối trong quá trình đào tạo trước của bộ giải mã mà chỉ tận dụng dữ liệu văn bản. Điều này linh hoạt và hiệu quả hơn khi sự cân chỉnh giữa hình ảnh và văn bản trở nên nhiều hơn. Khuôn khổ DeCap của chúng tôi được mô tả dưới đây: Trong quá trình đào tạo trước, bộ giải mã văn bản được đào tạo từ đầu. Mục tiêu là đảo ngược bộ mã hóa văn bản CLIP, tức là, một câu đầu tiên được mã hóa thành một nhúng bởi bộ mã hóa văn bản CLIP và sau đó được bộ giải mã văn bản của chúng tôi tái tạo. Bộ giải mã lấy nhúng văn bản thu được từ bộ mã hóa văn bản CLIP làm nhúng tiền tố. Trong quá trình suy luận zero-shot, khó khăn nằm ở cách lấy nhúng tiền tố có thể khớp với hình ảnh đầu vào và được bộ giải mã giải mã tốt. Hiện tượng khoảng cách phương thức (Liang và cộng sự, 2022b) được quan sát thấy trong các mô hình tương phản đa phương thức, ngăn cản chúng ta trực tiếp lấy nhúng hình ảnh làm nhúng tiền tố. Ramesh và cộng sự (2022) sử dụng dữ liệu ghép nối để học một mô hình nhằm ánh xạ nhúng văn bản thành nhúng hình ảnh tương ứng. Thay vì học một mô hình, chúng tôi đề xuất một cơ chế không cần đào tạo để chiết nhúng hình ảnh vào không gian nhúng văn bản CLIP. Kết hợp bộ giải mã văn bản với cơ chế chiết, chúng tôi tạo ra các mô tả chất lượng cao cho các hình ảnh nhất định.

Những đóng góp chính của chúng tôi được tóm tắt như sau:

(1) Chúng tôi đề xuất một khuôn khổ mới cho chủ thích zero-shot. Khuôn khổ DeCap của chúng tôi chứa một mô hình tương phản được đào tạo trước (tức là CLIP) và một bộ giải mã ngôn ngữ nhận biết hình ảnh nhẹ nhàng CLIP làm đầu vào. Mặc dù bộ giải mã của chúng tôi chỉ được đào tạo trên ngữ liệu văn bản, nhưng nó có thể liên kết cả nhúng hình ảnh và nhúng văn bản, nhờ vào mối tương quan đa phương thức được mã hóa trong không gian nhúng CLIP.

(2) Chúng tôi đề xuất một cơ chế chiết không cần đào tạo để giảm khoảng cách phương thức trong không gian nhúng đa phương thức CLIP. Chúng tôi kết hợp một bộ nhớ hỗ trợ đơn giản chứa các nhúng của ngữ liệu văn bản trong giai đoạn tiền đào tạo. Chúng tôi chiết một nhúng trực quan vào không gian nhúng văn bản CLIP thông qua bộ nhớ hỗ trợ. Các thí nghiệm cho thấy cơ chế chúng tôi đề xuất có hiệu quả làm giảm khoảng cách phương thức và cải thiện đáng kể hiệu suất.

(3) Các thí nghiệm mở rộng chứng minh DeCap có thể áp dụng linh hoạt vào nhiều tình huống chủ thích khác nhau. DeCap vượt trội hơn các phương pháp chủ thích zero-shot khác với biên độ lớn trên các chuẩn chủ thích hình ảnh MSCOCO và NoCaps. DeCap được đào tạo trên dữ liệu chỉ có văn bản vượt trội hơn các phương pháp chủ thích không ghép nối khác trên MSCOCO và Flickr30k. Chúng tôi áp dụng DeCap vào chủ thích video và đạt được kết quả zero-shot tiên tiến trên MSR-VTT và ActivityNet-Captions.

## 2 CÔNG TRÌNH LIÊN QUAN

CLIP trong chủ thích. Các mô hình ngôn ngữ thị giác (Radford và cộng sự, 2021; Jia và cộng sự, 2021; Yang và cộng sự, 2022) được đào tạo với mắt mèo tương phản cho thấy khả năng ẩn tượng trong nhiều nhiệm vụ phân biệt. Tuy nhiên, do không có bộ giải mã văn bản trong quá trình đào tạo trước, các mô hình này không thể được áp dụng trực tiếp vào các nhiệm vụ tạo ra, ví dụ như chủ thích. Các công trình trước đây (Mokady và cộng sự, 2021; Barraco và cộng sự, 2022; Shen và cộng sự, 2022) đã áp dụng CLIP vào nhiệm vụ chủ thích hình ảnh như một bộ mã hóa trực quan. Tuy nhiên, chúng bỏ qua

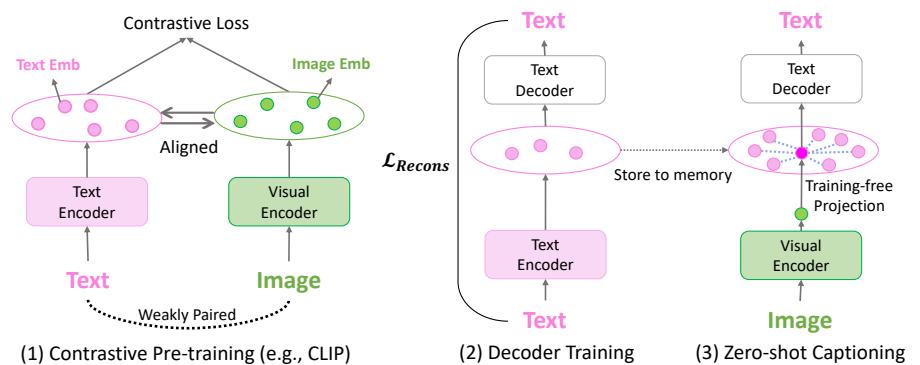


Figure 1: An overview of our framework. Our method is based on a pre-trained contrastive model CLIP containing a text encoder and a visual encoder. We first learn a text decoder to generate sentences conditioned on the CLIP text embedding. At inference, a training-free mechanism is used to project the image embedding into the text embedding space with the help of a support memory. The projected embedding is further decoded by the text decoder.

the CLIP text encoder and overlook the aligned multi-modal latent space provided by CLIP. In this work, we train a text decoder with text-only data to invert the CLIP text encoder. By leveraging CLIP multi-modal latent space, we apply CLIP to captioning tasks without additional pairwise training.

**Zero-shot Captioning.** Zero-shot captioning aims to generate image/video captions without human-annotated data. Changpinyo et al. (2021); Wang et al. (2021b); Alayrac et al. (2022) train vision-language models on noisy paired image-text data collected from the Web and evaluate on downstream benchmarks without fine-tuning. Another line of work achieves zero-shot captioning by combining existing web-scale pre-trained models. ZeroCap (Tewel et al., 2022b) combines a multi-modal model (e.g., CLIP) with a PLM (e.g., GPT-2). In each generation step, they use CLIP to guide GPT-2 toward a desired visual direction via the proposed CLIP loss. Socratic Models (Zeng et al., 2022) use a pre-trained VLM (Gu et al., 2021) to generate prompt templates for GPT-3 (Brown et al., 2020) and then use CLIP to retrieve the closest description to the image from the generated candidates. In this work, we employ CLIP for zero-shot captioning. Different from the above work using PLMs, we use text-only data to train a decoder from scratch.

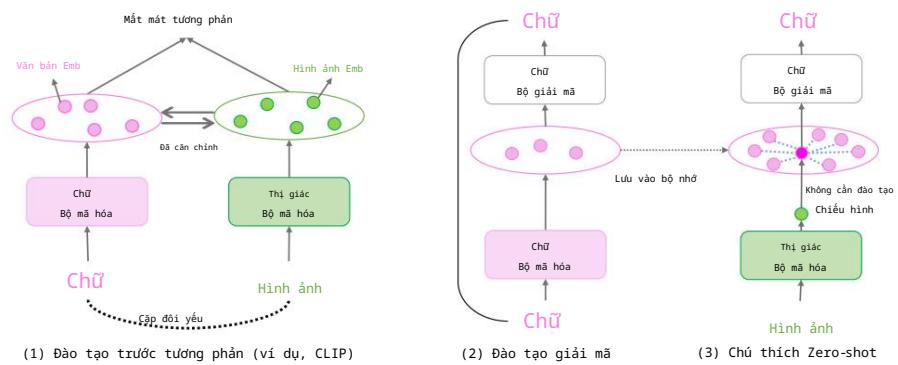
**Text Reconstruction.** Prior work (Feng et al., 2019; Laina et al., 2019; Liu et al., 2021a;b) employ a text reconstruction task to train a decoder for unpaired/unsupervised captioning tasks. Lacking a well-aligned multi-modal latent space, most of these methods require complex pseudo-training or adversarial training to align the decoder and visual input. Liu et al. (2021b) construct a knowledge graph to correlate the representations of the visual and textual domains. However, this method needs a well-defined knowledge graph and a multi-label classification task to train the knowledge graph, which is difficult to apply to captioning tasks other than medical report generation. Benefiting from CLIP, on the one hand, our decoder can be directly associated with visual input by utilizing the aligned cross-modal embedding space of CLIP. On the other hand, our decoder can be trained on various text data and applied to various captioning tasks.

### 3 METHOD

Our framework is shown in Figure 1. We learn a text decoder to convert the CLIP text encoder (Sec. 3.1). This text decoder allows us to generate sentences based on the CLIP text embedding. At inference, we propose a training-free mechanism to project the image embedding into the text embedding space to reduce the modality gap between the text embedding space and image embedding space (Sec. 3.2.1). We introduce more inference strategies for comparison (Sec. 3.2.2).

#### 3.1 TEXT-ONLY DECODER PRE-TRAINING

Previous approaches (Tewel et al., 2022b; Su et al., 2022; Zeng et al., 2022) employ PLMs to generate diverse sentences for zero-shot captioning. However, PLMs trained on various documents



Hình 1: Tổng quan về khung khổ của chúng tôi. Phương pháp của chúng tôi dựa trên mô hình tương phản được đào tạo trước CLIP chứa bộ mã hóa văn bản và bộ mã hóa hình ảnh. Đầu tiên, chúng tôi học bộ giải mã văn bản để tạo ra các câu có điều kiện trên nhung văn bản CLIP. Khi suy luận, một cơ chế không cần đào tạo được sử dụng để chiếu hình ảnh nhung vào không gian nhung văn bản với sự trợ giúp của bộ nhớ hỗ trợ.

Nội dung nhung được chiếu sẽ được giải mã thêm bằng bộ giải mã văn bản.

bộ mã hóa văn bản CLIP và bỏ qua không gian tiềm ẩn đa phương thức được cẩn chỉnh do CLIP cung cấp. Trong công việc này, chúng tôi đào tạo một bộ giải mã văn bản với dữ liệu chỉ có văn bản để đảo ngược bộ mã hóa văn bản CLIP. Bằng cách tận dụng không gian tiềm ẩn đa phương thức CLIP, chúng tôi áp dụng CLIP vào các tác vụ chủ thích mà không cần đào tạo từng cặp bổ sung.

Chú thích Zero-shot. Chú thích Zero-shot nhằm mục đích tạo chú thích hình ảnh/video mà không cần dữ liệu chủ thích của con người. Changpinyo và cộng sự (2021); Wang và cộng sự (2021b); Alayrac và cộng sự (2022) đào tạo các mô hình ngôn ngữ thị giác trên dữ liệu hình ảnh-văn bản ghép đôi nhiều được thu thập từ Web và đánh giá trên các điểm chuẩn hạ lưu mà không cần tinh chỉnh. Một hướng nghiên cứu khác đạt được chú thích Zero-shot bằng cách kết hợp các mô hình được đào tạo trước ở quy mô web hiện có. ZeroCap (Tewel và cộng sự, 2022b) kết hợp một mô hình đa phương thức (ví dụ: CLIP) với PLM (ví dụ: GPT-2). Trong mỗi bước tạo, họ sử dụng CLIP để hướng dẫn GPT-2 theo hướng quan trọng thông qua mắt mát CLIP được đề xuất. Socratic Models (Zeng và cộng sự, 2022) sử dụng VLM được đào tạo trước (Gu và cộng sự, 2021) để tạo mẫu nhắc nhở cho GPT-3 (Brown và cộng sự, 2020) và sau đó sử dụng CLIP để lấy mô tả gần nhất với hình ảnh từ các ứng viên được tạo. Trong công trình này, chúng tôi sử dụng CLIP để chú thích zero-shot. Khác với công trình trên sử dụng PLM, chúng tôi sử dụng dữ liệu chỉ có văn bản để đào tạo bộ giải mã từ đầu.

Tái tạo văn bản. Các công trình trước đây (Feng và cộng sự, 2019; Laina và cộng sự, 2019; Liu và cộng sự, 2021a;b) sử dụng tác vụ tái tạo văn bản để đào tạo bộ giải mã cho các tác vụ chủ thích không ghép nối/không giám sát. Do thiếu không gian tiềm ẩn đa phương thức được cẩn chỉnh tốt, hầu hết các phương pháp này đều yêu cầu đào tạo giả phức tạp hoặc đào tạo đối nghịch để cẩn chỉnh bộ giải mã và đầu vào trực quan. Liu và cộng sự (2021b) xây dựng biểu đồ kiến thức để tương quan các biểu diễn của miền trực quan và văn bản. Tuy nhiên, phương pháp này cần một biểu đồ kiến thức được xác định rõ ràng và tác vụ phân loại đa nhãn để đào tạo biểu đồ kiến thức, điều này khó áp dụng cho các tác vụ chủ thích khác ngoài việc tạo báo cáo y khoa. Một mặt, nhờ được hưởng lợi từ CLIP, bộ giải mã của chúng tôi có thể được liên kết trực tiếp với đầu vào trực quan bằng cách sử dụng không gian nhung đa phương thức được cẩn chỉnh của CLIP. Mặt khác, bộ giải mã của chúng tôi có thể được đào tạo trên nhiều dữ liệu văn bản khác nhau và được áp dụng cho nhiều tác vụ chủ thích khác nhau.

### 3 PHƯƠNG PHÁP

Khung của chúng tôi được thể hiện trong Hình 1. Chúng tôi học một bộ giải mã văn bản để chuyển đổi bộ mã hóa văn bản CLIP (Phần 3.1). Bộ giải mã văn bản này cho phép chúng tôi tạo ra các câu dựa trên nhung văn bản CLIP. Khi suy luận, chúng tôi đề xuất một cơ chế không cần đào tạo để chiếu hình ảnh nhung vào không gian nhung văn bản để giảm khoảng cách phương thức giữa không gian nhung văn bản và không gian nhung hình ảnh (Phần 3.2.1). Chúng tôi giới thiệu thêm các chiến lược suy luận để so sánh (Phần 3.2.2).

#### 3.1 ĐÀO TẠO TRƯỚC BỘ GIẢI MÃ CHỈ VĂN BẢN

Các phương pháp tiếp cận trước đây (Tewel et al., 2022b; Su et al., 2022; Zeng et al., 2022) sử dụng PLM để tạo ra các câu đa dạng cho chủ thích zero-shot. Tuy nhiên, PLM được đào tạo trên nhiều tài liệu khác nhau

from the webpages do not match well with captioning tasks that aim to describe visual concepts and relationships in the given image.

Instead of employing a PLM, we train a text decoder from scratch to invert the CLIP text encoder. Following recent work (Mokady et al., 2021; Wang et al., 2021b), we train our decoder using the prefix language modeling. Specifically, given a sentence  $t = \{word_1, word_2, \dots, word_{|t|}\}$ , the prefix language model  $P_\theta$  learns to reconstruct  $t$  conditioned on the text embedding extracted by a fixed CLIP text encoder. We regard the text embedding as a prefix to the caption. Our objective can be described as:

$$\mathcal{L}_{Recons}(\theta) = -\frac{1}{|t|} \sum_{i=1}^{|t|} \log P_\theta(word_i | word_{<i}, E_{text}(t)), \quad (1)$$

where  $E_{text}(\cdot)$  means mapping a sentence to a  $\ell_2$ -normalized embedding space via the CLIP text encoder. This decoder trained with text-only data in a self-supervised manner brings two benefits. On the one hand, we can control the style of the generated sentences by adjusting the source of text-only data. To generate task-specific descriptive captions, we train our decoder on text data from human-annotated image descriptions and web-collected image captions. On the other hand, this text decoder takes CLIP text embedding as the prefix embedding. The CLIP text embedding is optimized to be correlated with the CLIP image embedding, making it possible to associate the text decoder with visual input without any pairwise training.

### 3.2 INFERENCE STRATEGIES

In Sec. 3.1, we obtain a decoder that can generate descriptions conditioned on the CLIP text embedding. At inference, the question is how to use the decoder to generate descriptions given the CLIP image embedding. Due to the modality gap between CLIP image embedding space and text embedding space, it is impractical to directly take the CLIP image embedding as the prefix embedding. Ramesh et al. (2022) learn a prior model to map the text embedding to a corresponding image embedding. However, this process requires paired data for training. We propose a training-free mechanism to project the image embedding into text embedding space.

#### 3.2.1 PROJECTION-BASED DECODING (PD)

Assuming that the language model  $P_\theta$  is trained on a given text set  $T = \{t_1, t_2, \dots, t_N\}$ , where  $N$  denotes the size of  $T$ . To represent the CLIP text embedding space, we maintain a support memory  $M = \{\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_N\}$ , where  $\mathbf{m}_i = E_{text}(t_i)$ . At inference, we aim to generate a caption for a given image  $I$ . With the help of the support memory  $M$ , we can project the image embedding into the text embedding space. Specifically, given the image embedding  $v = E_{image}(I)$ , we obtain its representation in text embedding space by performing a weighted combination of all the embeddings in support memory. To obtain the weights of these text embeddings, the cosine similarity between  $v$  and  $m$  is calculated, scaled by a temperature parameter  $\tau$ , and normalized by a softmax function. The combined project vector  $v_{proj}$  is calculated as:

$$v_{proj} = \sum_{i=1}^N w_i * \mathbf{m}_i = \sum_{i=1}^N \frac{\exp((\mathbf{m}_i^\top v) / \tau)}{\sum_{k=1}^N \exp((\mathbf{m}_k^\top v) / \tau)} * \mathbf{m}_i, \quad (2)$$

where  $w_i$  is the weight of  $i$ -th text embedding in support memory.  $v_{proj}$  is a combination of CLIP text embeddings that can be used as the prefix embedding. We denote  $P_\theta(x)$  as the auto-regressive process of generating a sentence conditioned on the prefix embedding  $x$ . The final output can be generated by  $P_\theta(\frac{v_{proj}}{\|v_{proj}\|_2})$ .

This projection-based method does not require additional training. It performs well across many datasets and is flexible. The projected vector  $v_{proj}$  can absorb the information from text embeddings in the support memory, thereby generating diverse and accurate descriptions. On the other hand, the text data used for training and stored in support memory can be different. We can select appropriate text data to construct a new support memory according to the target domain. The image embedding will then be projected into the new text embedding space, enabling DeCap to generalize quickly to new domains without retraining.

từ các trang web không phù hợp với nhiệm vụ chú thích nhằm mục đích mô tả các khái niệm trực quan và mối quan hệ trong hình ảnh cho sẵn.

Thay vì sử dụng PLM, chúng tôi đào tạo bộ giải mã văn bản từ đầu để đảo ngược bộ mã hóa văn bản CLIP. Tiếp theo công trình gần đây (Mokady và cộng sự, 2021; Wang và cộng sự, 2021b), chúng tôi đào tạo bộ giải mã của mình bằng cách sử dụng mô hình ngôn ngữ tiền tố. Cụ thể, với một câu  $t = \{word_1, word_2, \dots, word_{|t|}\}$ , mô hình ngôn ngữ tiền tố  $P_\theta$  học cách tái tạo  $t$  có điều kiện dựa trên nhúng văn bản được trích xuất bởi bộ mã hóa văn bản CLIP cố định. Chúng tôi coi nhúng văn bản là tiền tố của chú thích. Mục tiêu của chúng tôi có thể được mô tả như sau:

$$L_{Recons}(\theta) = \frac{1}{|t|} \sum_{i=1}^{|t|} \log P_\theta(word_i | word_{<i}, E_{text}(t)), \quad (1)$$

trong đó  $E_{text}(\cdot)$  có nghĩa là ánh xạ một câu vào không gian nhúng chuẩn hóa 2 thông qua bộ mã hóa văn bản CLIP. Bộ giải mã này được đào tạo bằng dữ liệu chỉ có văn bản theo cách tự giám sát mang lại hai lợi ích. Một mặt, chúng ta có thể kiểm soát phong cách của các câu được tạo ra bằng cách điều chỉnh nguồn dữ liệu chỉ có văn bản. Để tạo ra các chủ thích mô tả cụ thể cho từng tác vụ, chúng tôi đào tạo bộ giải mã của mình trên dữ liệu văn bản từ các mô tả hình ảnh do con người chú thích và các chủ thích hình ảnh được thu thập trên web. Mặt khác, bộ giải mã văn bản này lấy nhúng văn bản CLIP làm nhúng tiền tố. Nhúng văn bản CLIP được tối ưu hóa để tương quan với nhúng hình ảnh CLIP, giúp có thể liên kết bộ giải mã văn bản với đầu vào trực quan mà không cần bất kỳ quá trình đào tạo từng cặp nào.

#### 3.2 CHIẾN LƯỢC SUY LUẬN

Trong Phần 3.1, chúng ta có được một bộ giải mã có thể tạo ra các mô tả có điều kiện trên nhúng văn bản CLIP. Khi suy ra, câu hỏi là làm thế nào để sử dụng bộ giải mã để tạo ra các mô tả cho nhúng hình ảnh CLIP. Do khoảng cách phương thức giữa không gian nhúng hình ảnh CLIP và không gian nhúng văn bản, nên không thực tế khi lấy trực tiếp nhúng hình ảnh CLIP làm tiền tố nhúng-ding. Ramesh và cộng sự (2022) tìm hiểu một mô hình trước đó để ánh xạ nhúng văn bản thành nhúng hình ảnh tương ứng. Tuy nhiên, quá trình này yêu cầu dữ liệu được ghép nối để đào tạo. Chúng tôi đề xuất một cơ chế không cần đào tạo để chiết nhúng hình ảnh vào không gian nhúng văn bản.

#### 3.2.1 GIẢI MÃ DỰA TRÊN PHÉP CHIỀU (PD)

Giả sử rằng mô hình ngôn ngữ  $P_\theta$  được đào tạo trên một tập văn bản cho trước  $T = \{t_1, t_2, \dots, t_N\}$ , trong đó  $N$  biểu thị kích thước của  $T$ . Để biểu diễn không gian nhúng văn bản CLIP, chúng ta duy trì bộ nhớ hỗ trợ  $M = \{\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_N\}$ , trong đó  $\mathbf{m}_i = E_{text}(t_i)$ . Khi suy luận, chúng ta đặt mục tiêu tạo chủ thích cho một hình ảnh cho trước  $I$ . Với sự trợ giúp của bộ nhớ hỗ trợ  $M$ , chúng ta có thể chiết hình ảnh nhúng vào không gian nhúng văn bản. Cụ thể, với hình ảnh nhúng  $v = E_{image}(I)$ , chúng ta thu được biểu diễn của nó trong không gian nhúng văn bản bằng cách thực hiện kết hợp có trọng số của tất cả các nhúng trong bộ nhớ hỗ trợ. Để thu được trọng số của các nhúng văn bản này, độ tương đồng cosin giữa  $v$  và  $m$  được tính toán, chia tỷ lệ theo tham số nhiệt độ  $\tau$  và chuẩn hóa theo hàm softmax.

Vectơ dự án kết hợp  $v_{proj}$  được tính như sau:

$$v_{proj} = \sum_{i=1}^N w_i * \mathbf{m}_i = \frac{\sum_{i=1}^N \frac{\exp((\mathbf{m}_i^\top v) / \tau)}{\sum_{k=1}^N \exp((\mathbf{m}_k^\top v) / \tau)} * \mathbf{m}_i}{\sum_{i=1}^N \frac{\exp((\mathbf{m}_i^\top v) / \tau)}{\sum_{k=1}^N \exp((\mathbf{m}_k^\top v) / \tau)}}, \quad (2)$$

trong đó  $w_i$  là trọng số của nhúng văn bản thứ  $i$  trong bộ nhớ hỗ trợ.  $v_{proj}$  là sự kết hợp của nhúng văn bản CLIP có thể được sử dụng làm nhúng tiền tố. Chúng tôi biểu thị  $P_\theta(x)$  là quá trình tự hồi quy để tạo ra một câu có điều kiện trên nhúng tiền tố  $x$ . Đầu ra cuối cùng có thể là  $v_{proj}$  được tạo bởi  $P_\theta(\cdot)$ .  $\|v_{proj}\|_2$

Phương pháp dựa trên phép chiết này không yêu cầu đào tạo bổ sung. Nó hoạt động tốt trên nhiều tập dữ liệu và linh hoạt. Vectơ chiết  $v_{proj}$  có thể hấp thụ thông tin từ nhúng văn bản trong bộ nhớ hỗ trợ, do đó tạo ra các mô tả đa dạng và chính xác. Mặt khác, dữ liệu văn bản được sử dụng để đào tạo và lưu trữ trong bộ nhớ hỗ trợ có thể khác nhau. Chúng ta có thể chọn dữ liệu văn bản phù hợp để xây dựng bộ nhớ hỗ trợ mới theo miền mục tiêu. Sau đó, nhúng hình ảnh sẽ được chiết vào không gian nhúng văn bản mới, cho phép DeCap tổng quát hóa nhanh chóng sang các miền mới mà không cần đào tạo lại.

### 3.2.2 DISCUSSION

In order to investigate the impact of our decoder and projection-based mechanism, we have included the following inference strategies for comparative analysis.

**1) CLIPRe.** We first consider a simple retrieval-based approach that does not require a decoder. This approach is mentioned in Su et al. (2022). Given the image  $I$  and text set  $T = \{t_1, t_2, \dots, t_n\}$ , CLIPRe retrieves the most relevant texts from  $T$  based on the image-text similarity measured by CLIP. This process can be formulated as:  $\arg \max_{t \in T} sim(E_{image}(I), E_{text}(t))$ , where  $sim$  denotes the cosine similarity. In all experiments, we use CLIPRe as our baseline, since it can well reflect the zero-shot performance of the original CLIP without the decoder.

**2) Visual Decoding (VD).** Considering that text embeddings and image embeddings are correlated, a simple approach is to directly use image embedding as the prefix embedding. We refer to this method as Visual Decoding. This process can be formulated as  $P_\theta(E_{image}(I))$ . However, across the experiments, this method does not achieve satisfying results in most scenarios, indicating that there is a modality gap between CLIP image embeddings and text embeddings.

**3) Nearest-neighbor Decoding (NND).** Another simple method is to use the nearest text embedding as the prefix embedding. Specifically, we first calculate the similarity between the image embedding  $E_{image}(I)$  and the text embeddings in  $M$ . Then, the nearest text embedding is directly used as the prefix embedding. We refer to this method as Nearest-neighbor Decoding. This process can be formulated as  $P_\theta(\arg \max_{\mathbf{m} \in M} sim(E_{image}(I), \mathbf{m}))$ . Ideally, NND and CLIPRe should attain similar performance since the decoder learns to recover the origin text conditioned on the text embedding. Interestingly, across our experiments, NND achieves better performance than CLIPRe in most scenarios, suggesting that our decode may generate more descriptive sentences. Moreover, we find that the performance could be further improved by reconstructing a new text corpus using the decoder. More results and discussions can be found in Appendix B.

## 4 EXPERIMENTS

We conduct extensive experiments on captioning tasks including zero-shot image captioning, unpaired image captioning, and video captioning. We demonstrate that DeCap can efficiently achieve impressive results in diverse settings. In Sec. 4.1, we focus on zero-shot image captioning without any human annotation. In Sec. 4.2, we focus on unpaired image captioning where the images and the sentences are treated independently. In Sec. 4.3, we further apply DeCap to video captioning tasks. In Sec. 4.4, we conduct detailed ablation studies for DeCap.

**Implementation Details.** We employ a frozen pre-trained Vit-B/32 CLIP model. We adopt a 4-layer Transformer (Subramanian et al., 2018) with 4 attention heads as our language model. The size of the hidden state is 768. By default, we use all the text data in the training set to train the language model from scratch with a naive cross-entropy loss. All the text embeddings from the training corpus are stored in the support memory unless specified otherwise. At inference, the temperature  $\tau$  in Eq. 2 is set to 1/150 in video captioning experiments, and 1/100 in image captioning experiments. We report the results over four standard captioning evaluation metrics: BLEU@N (Papineni et al., 2002), METEOR (Banerjee & Lavie, 2005), CIDEr (Vedantam et al., 2015), and SPICE (Anderson et al., 2016). Additionally, we use CLIP-S<sup>Ref</sup> (Hessel et al., 2021) and CLIP-S to measure the text-text similarity and text-image similarity, respectively. The beam search or constrained beam search (Anderson et al., 2017) is **not** used in all our results.

#### 4.1 ZERO-SHOT IMAGE CAPTIONING

In this section, we conduct zero-shot image captioning using webly-collected corpora. Traditional image captioning methods rely on paired human-annotated data for training, which is difficult to obtain and limited in scale and diversity. We consider three webly-collected corpora for DeCap training: (1) **CC3M** (Sharma et al., 2018) contains three million image-description pairs collected from the web. We only use the text descriptions (CC3M-text) for training. We use one million descriptions randomly sampled from the 3M descriptions to construct the support memory. (2) **SSIM** is a webly-collected corpus specifically designed for MSCOCO caption. Feng et al. (2019) use the name of the eighty object classes in MSCOCO as keywords to crawl the descriptions from

### 2.2 THẢO LUÂN

Đến nghiên cứu tác động của bộ giải mã và cơ chế dựa trên phép chiếu, chúng tôi đã đưa vào các chiến lược suy luận để phân tích so sánh.

CLIPRe. Đầu tiên chúng ta xem xét một phương pháp tiếp cận dựa trên truy xuất đơn giản không yêu cầu bộ giải mã. Phương pháp này được đề cập trong Su et al. (2022). Với hình ảnh  $I$  và tập văn bản  $T = \{t_1, t_2, \dots, t_n\}$ , CLIPRe sẽ truy xuất các văn bản có liên quan nhất từ  $T$  dựa trên độ tương đồng giữa hình ảnh và văn bản được do bằng  $\text{sim}(\text{Eimage}(I), \text{Etext}(t))$ , trong đó  $\text{sim}$  de-CLIP. Quá trình này có thể được xây dựng như sau:  $\arg \max_t \text{sim}(I, t)$ . Khi CLIPRe làm đường cơ sở của mình, vì nó có thể phản ánh tốt hiệu suất zero-shot của CLIP, chủ độ tương đồng cosin. Trong tất cả các thí nghiệm, chúng tôi sử dụng mã không cần bộ giải mã.

Giải mã trực quan (VD). Xem xét rằng nhúng văn bản và nhúng hình ảnh có tương quan với nhau, một cách tiếp cận giản là sử dụng trực tiếp nhúng hình ảnh làm nhúng tiền tố. Chúng tôi gọi phương pháp này là Giải mã trực quan. Á trình này có thể được xây dựng thành P0 (Eimage(I)). Tuy nhiên, trong các thí nghiệm, phương pháp này không đạt được kết quả thỏa đáng trong hầu hết các tình huống, cho thấy có một khoảng cách phương thức giữa nhúng hình ảnh IP và nhúng văn bản.

Giải mã lân cận gần nhất (NNM). Một phương pháp đơn giản khác là sử dụng nhúng văn bản gần nhất làm nhúng tiền tố. Cụ thể, trước tiên chúng ta tính toán độ tương tự giữa nhúng hình ảnh Eimage(I) và nhúng văn bản trong M. Sau đó, nhúng văn bản gần nhất được sử dụng trực tiếp làm nhúng tiền tố. Chúng tôi gọi phương pháp này là Giải mã lân cận gần nhất. Quá trình này có thể được xây dựng thành P0 ( $\arg \max_m M \sim(Eimage(I), m)$ ). Về mặt lý tưởng, NNM và CLIPRe sẽ đạt được hiệu suất tương tự vì bộ giải mã học cách khôi phục văn bản gốc có điều kiện trên nhúng văn bản. Điều thú vị là trong suốt các thử nghiệm của chúng tôi, NNM đạt được hiệu suất tốt hơn CLIPRe trong hầu hết các trường hợp, cho thấy rằng giải mã của chúng tôi có thể tạo ra nhiều câu mô tả hơn. Hơn nữa, chúng tôi thấy rằng hiệu suất có thể được cải thiện hơn nữa bằng cách tái tạo một ngữ liệu văn bản mới bằng bộ giải mã. Có thể tìm thấy thêm kết quả và thảo luận trong Phụ lục B.

THÍ NGHIỆM

úng tôi tiến hành các thử nghiệm mở rộng về các tác vụ chú thích bao gồm chú thích hình ảnh zero-shot, chú thích ảnh không ghép nối và chú thích video. Chúng tôi chứng minh rằng DeCap có thể đạt được hiệu quả ấn tượng trong nhiều bối cảnh khác nhau. Trong Phần 4.1, chúng tôi tập trung vào chú thích hình ảnh zero-shot mà không có bất kỳ chú thích nào của con người. Trong Phần 4.2, chúng tôi tập trung vào chú thích hình ảnh không ghép nối, trong đó ảnh và câu được xử lý độc lập. Trong Phần 4.3, chúng tôi tiếp tục áp dụng DeCap vào các tác vụ chú thích video. Cuối Phần 4.4, chúng tôi tiến hành các nghiên cứu cốt bô chi tiết cho DeCap.

hi tiết triển khai. Chúng tôi sử dụng mô hình Vit-B/32 CLIP được đào tạo trước đông lạnh. Chúng tôi áp dụng transformer 4 lớp (Subramanian và cộng sự, 2018) với 4 đầu chú ý làm mô hình ngôn ngữ của mình. Kích thước của trạng thái ẩn là 768. Theo mặc định, chúng tôi sử dụng tất cả dữ liệu văn bản trong tập huấn luyện để huấn luyện mô hình ngôn ngữ từ đầu với mất mát entropy chéo ngay thơ. Tất cả các nhúng văn bản từ ngữ liệu huấn luyện được lưu trữ trong bộ nhớ hỗ trợ trữ khi được chỉ định khác. Khi suy luận, nhiệt độ  $\tau$  trong Công thức 2 được đặt thành 1/150 trong các thử nghiệm chú thích video và 1/100 trong các thử nghiệm chú thích hình ảnh.

Chúng tôi báo cáo kết quả qua bốn số liệu đánh giá phụ đề chuẩn: BLEU@N (Papineni và cộng sự, 2002), METEOR (Banerjee & Lavie, 2005), CIDEr (Vedantam và cộng sự, 2015) và SPICE (Anderson và cộng sự, 2016). Ngoài ra, chúng tôi sử dụng CLIP-SRef (Hessel và cộng sự, 2021) và CLIP-S để đo độ tương đồng giữa văn bản-văn bản và độ tương đồng giữa văn bản-hình ảnh. Tìm kiếm chùm tia hoặc tìm kiếm chùm tia bị ràng buộc (Anderson và cộng sự, 2017) không được áp dụng trong tất cả các kết quả của chúng tôi.

#### 1.1 CHÚ THÍCH HÌNH ẢNH KHÔNG CHUP

Trong phần này, chúng tôi tiến hành chú thích hình ảnh zero-shot bằng cách sử dụng các tập hợp dữ liệu được thu thập trên webly. Các phương pháp chú thích hình ảnh truyền thống dựa vào dữ liệu được chú thích của con người theo cách để đào tạo, rất khó để có được và hạn chế về quy mô và tính đa dạng. Chúng tôi xem xét ba tập hợp dữ liệu được thu thập trên webly để đào tạo DeCap: (1) CC3M (Sharma và cộng sự, 2018) chứa ba triệu cặp hình ảnh-mô tả được thu thập từ web. Chúng tôi chỉ sử dụng các mô tả văn bản (CC3M-text) để đào tạo. Chúng tôi sử dụng một triệu mô tả được lấy mẫu ngẫu nhiên từ các mô tả 3M để xây dựng bộ nhớ hỗ trợ. (2)

CC3M là một tập hợp dữ liệu được thu thập trên web được thiết kế riêng cho chú thích MSCOCO. Feng et al. (2019) sử dụng tên của tám mươi lăm lớp đối tượng trong MSCOCO làm từ khóa để thu thập các mô tả từ

Methods	Pre-training stage	MSCOCO			NoCaps val (CIDEr)				
		B@4	M	C	S	In	Near	Out	Overall
Changpinyo et al. (2021)	CC3M	-	-	-	-	29.2	27.5	37.3	29.7
Changpinyo et al. (2021)	CC12M	-	-	-	-	20.7	24.1	41.6	27.1
ZeroCap	CLIP+GPT-2	2.6	11.5	14.6	5.5	-	-	-	-
CLIPRe	CLIP+CC3M-text	4.6	13.3	25.6	9.2	23.3	26.8	36.5	28.2
DeCap-VD	CLIP+CC3M-text	1.2	10.4	8.1	5.8	8.4	8.0	10.2	8.5
DeCap-NND	CLIP+CC3M-text	5.3	13.7	27.1	9.1	24.2	27.1	37.6	28.8
DeCap	CLIP+CC3M-text	8.8	16.0	42.1	10.9	34.8	37.7	49.9	39.7
DeCap	CLIP+SS1M	<b>8.9</b>	<b>17.5</b>	<b>50.6</b>	<b>13.1</b>	<b>41.9</b>	<b>41.7</b>	<b>46.2</b>	<b>42.7</b>
DeCap	CLIP+Book Corpus	6.6	12.9	31.9	8.7	26.8	31.8	44.3	33.6

Table 1: Zero-shot captioning results on MSCOCO Karpathy-test split and NoCaps validation set. (In: in-domain; Near: near-domain; Out: out-of-domain; B@4: BLEU@4; M: METEOR; C: CIDEr; S: SPICE).

Shutterstock<sup>1</sup>, resulting in 2,322,628 distinct image descriptions in total. We reuse this corpus and further remove sentences with more than fifteen words, obtaining 978,662 sentences. (3) **Book Corpus** (Zhu et al., 2015) is a large collection of free novel books. Book Corpus is often used for unsupervised pre-training of language models (Devlin et al., 2018) and we also use it to train our language decoder, but for captioning tasks. The original Book Corpus data is large and many sentences are not visual-related, which makes our decoder training inefficient. In practice, we find that the norm of CLIP text embedding can coarsely filter out some sentences that are not related to visual concepts. A sentence with a large norm is usually not visual-related. To improve training efficiency, we only keep sentences with lengths less than 15 and norms less than 10 and obtain 6,217,799 sentences for training. We use one million sentences randomly sampled from the training data to construct the support memory. In addition, we use “Attention! There is/are” as a prompt for the model trained on Book Corpus. We find that DeCap trained on Book Corpus benefits from prompt engineering, whereas DeCap trained on CC3M does not. More other prompts, results, and analyses are in Appendix F.

The following zero-shot captioning methods are compared in this study. Changpinyo et al. (2021) train a captioning model on webly-collected paired data and directly transfer it to downstream datasets without fine-tuning. **ZeroCap** (Tewel et al., 2022b) is a training-free zero-shot captioning method leveraging CLIP and GPT-2. DeCap also utilizes CLIP but trains a decoder from scratch on a webly-collected corpus. Our **DeCap** uses projection-based decoding (PD) by default. We compare it with another two inference strategies introduced in Sec. 3.2. We denote visual decoding as **DeCap-VD** and nearest-neighbor decoding as **DeCap-NND**. All these methods target zero-shot image captioning and do not use human-annotated data.

**Results.** Table 1 shows the zero-shot results on MSCOCO and NoCaps. DeCap attains a new state-of-the-art on all metrics. On NoCaps, models pre-trained on webly-collected data achieve better out-of-domain results. This is because the webly-collected data contain diverse visual concepts. On MSCOCO, DeCap pre-trained on CC3M-text outperforms ZeroCap by 27.5% in CIDEr. DeCap pre-trained on SS1M outperforms ZeroCap by 36% in CIDEr. DeCap trained on SS1M achieves better performance than trained on CC3M (CIDEr: 50.6% vs. 42.1%), indicating that the task-specific webly-collected corpus can further improve the performance of downstream datasets. Besides, DeCap trained on Book Corpus still achieves better performance than ZeroCap. Notably, both DeCap-BookCorpus and ZeroCap have not seen caption-related data.

#### 4.2 UNPAIRED IMAGE CAPTIONING

To explore the potential of DeCap in more captioning scenarios, we consider the unpaired image captioning setting, where the human-annotated image-sentence pairs are treated as unpaired images and sentences. In Sec. 4.2.1, we investigate in-domain captioning where training data and test data come from the same dataset, but the training data are unpaired. In Sec. 4.2.2, we consider the cross-domain situation where training data and test data come from different distributions.

Phương pháp	Giai đoạn tiền đào tạo	MSCOCO				NoCaps val (CIDEr)	
		B@4	MCS	Trong Miền	Gần Miền	Tổng Thể	
Changpinyo và cộng sự (2021)	CC3M	-	-	-	-	29,2	27,5
Changpinyo và cộng sự (2021)	CC12M	-	-	-	-	24,1	41,6
Clip	KEP+GPT-2	2,6	11,5	14,6	5,5	13,3	-
ZeroCap	CLIP+CC3M-văn bản	4,6	25,6	9,2	-	-	-
DeCap-VD	CLIP+CC3M-văn bản	1,2	10,4	8,1	5,8	10,2	5,3
DeCap-NND	CLIP+CC3M-văn bản	37,6	8,8	16,0	42,1	10,9	34,8
Bỏ mũ	CLIP+CC3M-văn bản	50,6	13,1	41,9	41,7	46,2	12,9
Bỏ mũ	CLIP+SS1M	44,3	-	-	-	-	-
Bỏ mũ	CLIP+Sách Corpus	6,6	-	-	-	-	-

Bảng 1: Kết quả chú thích Zero-shot trên bộ phân tách kiểm định Karpathy của MSCOCO và bộ xác thực NoCaps.

(Trong: trong miền; Gần: gần miền; Ngoài: ngoài miền; B@4: BLEU@4; M: METEOR; C: Rượu táo; S: GIA VI).

Ảnh: Shutterstock<sup>1</sup>, dẫn đến tổng cộng 2.322.628 mô tả hình ảnh riêng biệt. Chúng tôi sử dụng lại tập hợp này và tiếp tục loại bỏ các câu có hơn mười lăm từ, thu được 978.662 câu. (3) Sách Corpus (Zhu et al., 2015) là một bộ sưu tập lớn các cuốn tiểu thuyết miễn phí. Book Corpus thường được sử dụng để đào tạo trước không giám sát các mô hình ngôn ngữ (Devlin và cộng sự, 2018) và chúng tôi cũng sử dụng nó để đào tạo bộ giải mã ngôn ngữ của chúng tôi, nhưng dành cho các tác vụ chú thích. Dữ liệu Book Corpus gốc lớn và nhiều câu không liên quan đến hình ảnh, điều này làm cho quá trình đào tạo giải mã của chúng tôi kém hiệu quả. Trong thực tế, chúng tôi thấy rằng chuẩn mực nhúng văn bản CLIP có thể lọc bỏ một số câu không liên quan đến khái niệm trực quan. Một câu có chuẩn mực lớn thường không liên quan đến trực quan. Để cải thiện đào tạo hiệu quả, chúng tôi chỉ giữ lại các câu có độ dài nhỏ hơn 15 và chuẩn mực nhỏ hơn 10 và thu được 6.217.799 câu để đào tạo. Chúng tôi sử dụng một triệu câu được lấy mẫu ngẫu nhiên từ quá trình đào tạo dữ liệu để xây dựng bộ nhớ hỗ trợ. Ngoài ra, chúng tôi sử dụng “Chú ý! Có/có” như một lời nhắc đối với mô hình được đào tạo trên Book Corpus. Chúng tôi thấy rằng DeCap được đào tạo trên Book Corpus được hưởng lợi từ kỹ thuật nhanh chóng, trong khi DeCap được đào tạo về CC3M thì không. Nhiều lời nhắc, kết quả và phân tích được nêu trong Phụ lục F.

Các phương pháp chú thích zero-shot sau đây được so sánh trong nghiên cứu này. Changpinyo và cộng sự (2021) đào tạo mô hình chú thích trên dữ liệu ghép được thu thập trên web và chuyển trực tiếp đến hổ lưu bộ dữ liệu không cần tinh chỉnh. ZeroCap (Tewel và cộng sự, 2022b) là phương pháp chú thích zero-shot không cần đào tạo tận dụng CLIP và GPT-2. DeCap cũng sử dụng CLIP nhưng đào tạo bộ giải mã từ đầu trên một tập hợp webly. DeCap của chúng tôi sử dụng giải mã dựa trên phép chiếu (PD) theo mặc định. Chúng tôi so sánh nó với hai chiến lược suy luận khác được giới thiệu trong Phần 3.2. Chúng tôi biểu thị giải mã trực quan như DeCap-VD và giải mã lân cận gần nhất như DeCap-NND. Tất cả các phương pháp này đều nhằm mục tiêu zero-shot chú thích hình ảnh và không sử dụng dữ liệu có chủ đích của con người.

Kết quả. Bảng 1 cho thấy kết quả zero-shot trên MSCOCO và NoCaps. DeCap đạt được một tiên tiến nhất về mọi số liệu. Trên NoCaps, các mô hình được đào tạo trước trên dữ liệu thu thập được trên webby đạt được kết quả ngoài miền tốt hơn. Điều này là do dữ liệu thu thập trên web chứa nhiều khái niệm trực quan khác nhau. Trên MSCOCO, DeCap được đào tạo trước trên CC3M-text vượt trội hơn ZeroCap 27,5% trong CIDEr. DeCap được đào tạo trước trên SS1M vượt trội hơn ZeroCap 36% trong CIDEr. DeCap được đào tạo trên SS1M đạt được hiệu suất tốt hơn so với đào tạo trên CC3M (CIDEr: 50,6% so với 42,1%), cho thấy rằng ngũ liệu thu thập trên web theo từng tác vụ cụ thể có thể cải thiện hiệu suất của các tập dữ liệu hổ lưu. Bên cạnh đó, DeCap được đào tạo trên Book Corpus vẫn đạt được hiệu suất tốt hơn ZeroCap. Đáng chú ý, cả DeCap-BookCorpus và ZeroCap đều chưa thấy dữ liệu liên quan đến phụ đề.

#### 4.2 CHÚ THÍCH HÌNH ẢNH KHÔNG GHÉP ĐỒ

Để khám phá tiềm năng của DeCap trong nhiều tình huống chú thích hơn, chúng tôi xem xét hình ảnh không ghép nối cài đặt chú thích, trong đó các cặp hình ảnh-câu được chú thích của con người được coi là hình ảnh không ghép nối và câu. Trong Mục 4.2.1, chúng tôi điều tra chú thích trong miền nơi dữ liệu đào tạo và dữ liệu thử nghiệm đến từ cùng một tập dữ liệu, nhưng dữ liệu đào tạo không được ghép nối. Trong Phần 4.2.2, chúng tôi xem xét tình huống liên miên trong đó dữ liệu đào tạo và dữ liệu thử nghiệm đến từ các phân phối khác nhau.

<sup>1</sup><https://www.shutterstock.com>

Method	Data			MSCOCO			Flickr30K				
	P.	I.	T.	B@4	M	C	S	B@4	M	C	S
<i>Supervised Methods</i>											
BUTD	✓			36.2	27.0	113.5	20.3	27.3	21.7	56.6	16.0
CLIPCap	✓			33.5	27.5	113.1	21.1	-	-	-	-
Barraco et al. (2022)	✓			36.0	27.8	114.9	20.8	-	-	-	-
CLIP-VL	✓			37.5	28.1	123.1	21.9	-	-	-	-
<i>Train on unpaired data. Zero-shot inference on image-text pairs</i>											
UVC-VI	†			22.0	21.4	72.3	-	-	-	-	-
Feng et al. (2019)	✓	✓		18.6	17.9	54.9	11.1	-	-	-	-
Laina et al. (2019)	✓	✓		19.3	20.2	61.8	12.9	-	-	-	-
ESPER-Style	✓	✓		21.9	21.9	78.2	-	-	-	-	-
ESPER-Free	✓			6.3	13.3	29.1	-	-	-	-	-
ZeroCap*	✓			7.0	15.4	34.5	9.2	5.4	11.8	16.8	6.2
Magic	✓			12.9	17.4	49.3	11.3	6.4	13.1	20.4	7.1
CLIPRe	✓			12.4	20.4	53.4	14.8	9.8	18.2	31.7	12.0
DeCap-VD	✓			5.0	15.5	25.7	9.8	5.8	15.0	13.0	8.2
DeCap-NND	✓			15.3	21.2	62.9	15.8	12.9	17.2	35.2	10.9
DeCap	✓			<b>24.7</b>	<b>25.0</b>	<b>91.2</b>	<b>18.7</b>	<b>21.2</b>	<b>21.8</b>	<b>56.7</b>	<b>15.2</b>

Table 2: In-domain captioning results on MSCOCO and Flickr30K. “\*\*” denotes results from Su et al. (2022). “P.”, “I.” and “T.” denote paired data, unpaired image data and unpaired text data, respectively. †: UVC-VI is a special approach that requires image-Chinese paired data for training, and we regard it as an unpaired method here because it does not use image-English pairs.

#### 4.2.1 IN-DOMAIN CAPTIONING

We compare DeCap with supervised methods and other unpaired image captioning methods. (1) Supervised methods: **BUTD** (Anderson et al., 2018) is a classic method that uses Faster R-CNN (Ren et al., 2015) to extract visual features. **CLIPCap** (Mokady et al., 2021), **CLIP-VL** (Shen et al., 2021) and Barraco et al. (2022) are recent approaches employing CLIP as the visual encoder. (2) Unpaired methods: Laina et al. (2019) and Feng et al. (2019) treat the images and captions from the MSCOCO training set as unpaired data. UVC-VI (Liu et al., 2021a) uses image-Chinese pairs (Wu et al., 2019) for training. (3) (CLIP+GPT2)-based methods: **ZeroCap** (Tewel et al., 2022b), **Magic** (Su et al., 2022) and **ESPER-Style** (Yu et al., 2022b) finetune the GPT-2 on captions from the training set. (4) **ESPER-Free** (Yu et al., 2022b) uses reinforcement learning to align multimodal inputs to language model generations. (5) **CLIPRe** is a retrieval-based baseline. (6) Our **DeCap**, **DeCap-VD** and **DeCap-NND**. Our decoder is trained on captions from the training set, and text embeddings of all the training captions are maintained in the support memory.

**Results.** Table 2 shows the results on MSCOCO and Flickr30K. Overall, DeCap outperforms recent unpaired approaches by a large margin. Especially on Flickr30K, DeCap is competitive with the supervised learning method BUTD. Two conclusions can be drawn: (1) **CLIP provides aligned multi-modal representations for captioning tasks.** Compared to the unpaired methods that use a visual concept detector to construct a multi-modal embedding space, the CLIP-based methods could achieve competitive results using only text data. (2) **Our decoder and the projection mechanism are crucial for high performance.** Compared to CLIPRe, DeCap-NND further decodes the nearest-neighbor text embeddings resulting in higher performance, indicating that our decoder can generate more descriptive sentences. DeCap-VD achieves inferior performance, demonstrating that there is a large modality gap between CLIP image embedding and text embedding, demonstrating the necessity of our projection mechanism.

#### 4.2.2 CROSS-DOMAIN CAPTIONING

We evaluate the following methods on MSCOCO and Flickr30K in the cross-domain setting where the training data and testing data are from different datasets. (1) Zhao et al. (2020) generate pseudo image-text pairs for the target domain using a retrieval model trained on the source domain. (2) **Magic** (Su et al., 2022) finetunes GPT-2 on text data from the source domain. (3) **CLIPRe-S** uses text data from the source domain as galleries. (4) **DeCap** trains the decoder on text data from the

Phương pháp	MSCOCO			Flickr30K		
	P.	I.	T. B@4 MC	SB@4 MC	S	
<i>Phương pháp giám sát</i>						
NHƯNG			36,2 27,0 113,5 20,3 27,3 21,7 56,6 16,0			
CLIPCap			33,5 27,5 113,1 21,1 36,0			
Barraco và cộng sự. (2022)			27,8 114,9 20,8 37,5 28,1			
CLIP-VL			123,1 21,9			
<i>Đào tạo trên dữ liệu không ghép đôi. Suy luận Zero-shot trên các cặp hình ảnh-văn bản</i>						
UVC-VI	†		22,0 21,4 72,3	-	-	-
Feng và cộng sự (2019)			18,6 17,9 54,9 11,1 19,3	-	-	-
Laina và cộng sự (2019)			20,2 61,8 12,9	-	-	-
ESPER-Style			21,9 21,9 78,2	-	-	-
ESPER-Free			6,3 13,3 29,1 7,0	-	-	-
ZeroCap			15,4 34,5 9,2 12,9 17,4	5,4	11,8 16,8 6,2	
Magic			49,3 11,3 12,4 20,4 53,4	14,8	6,4	13,1 20,4 7,1
CLIPRe					9,8	18,2 31,7 12,0
DeCap-VD			5,0 15,5 25,7 15,0 13,0 8,8	5,8		
DeCap-NND			15,3 21,2 62,9 15,8 12,9 17,2	35,2 10,9		
Bộ mū			24,7 25,0 91,2 18,7 21,2 21,8	56,7 15,2		

Bảng 2: Kết quả chú thích trong miền trên MSCOCO và Flickr30K. “” biều thị kết quả từ Su et al. (2022). “P.”, “I.” và “T.” biều thị dữ liệu ghép nối, dữ liệu hình ảnh không ghép nối và dữ liệu văn bản không ghép nối, tương ứng. †: UVC-VI là một phương pháp tiếp cận đặc biệt đòi hỏi dữ liệu ghép nối hình ảnh-trung để đào tạo, và chúng tôi coi đây là phương pháp không ghép đôi vì nó không sử dụng cặp hình ảnh-Anh.

#### 4.2.1 CHÚ THÍCH TRONG MIỀN

Chúng tôi so sánh DeCap với các phương pháp có giám sát và các phương pháp chú thích hình ảnh không ghép nối khác. (1) Phương pháp có giám sát: BUTD (Anderson và cộng sự, 2018) là phương pháp có diễn sử dụng Faster R-CNN (Ren et al., 2015) để trích xuất các đặc điểm trực quan. CLIPCap (Mokady et al., 2021), CLIP-VL (Shen et al., 2021) và Barraco et al. (2022) là những phương pháp tiếp cận gần đây sử dụng CLIP làm bộ mã hóa hình ảnh. (2) Phương pháp không ghép đôi: Laina et al. (2019) và Feng et al. (2019) xử lý hình ảnh và chú thích từ bộ đào tạo MSCOCO là dữ liệu không ghép nối. UVC-VI (Liu và cộng sự, 2021a) sử dụng các cặp hình ảnh-Trung Quốc (Wu et al., 2019) để đào tạo. (3) Phương pháp dựa trên (CLIP+GPT2): ZeroCap (Tewel et al., 2022b), Magic (Su và cộng sự, 2022) và ESPER-Style (Yu và cộng sự, 2022b) tinh chỉnh GPT-2 trên phụ đề từ bộ đào tạo. (4) ESPER-Free (Yu et al., 2022b) sử dụng học tăng cường để căn chỉnh đa phương thức đầu vào cho các thể hệ mô hình ngôn ngữ. (5) CLIPRe là một đường cơ sở dựa trên truy xuất. (6) DeCap của chúng tôi, DeCap-VD và DeCap-NND. Bộ giải mã của chúng tôi được đào tạo trên các chú thích từ bộ đào tạo và văn bản nhúng của tất cả các chú thích đào tạo được duy trì trong bộ nhớ hỗ trợ.

Kết quả. Bảng 2 cho thấy kết quả trên MSCOCO và Flickr30K. Nhìn chung, DeCap vượt trội hơn so với cách tiếp cận không ghép đôi với biên độ lớn. Đặc biệt trên Flickr30K, DeCap có tính cạnh tranh với phương pháp học có giám sát BUTD. Có thể rút ra hai kết luận: (1) CLIP cung cấp biểu diễn đa phương thức cho các nhiệm vụ chú thích. So với các phương pháp không ghép nối sử dụng máy dò khái niệm trực quan để xây dựng không gian đa phương thức, các phương pháp dựa trên CLIP có thể đạt được kết quả cạnh tranh chỉ bằng cách sử dụng dữ liệu văn bản. (2) Bộ giải mã và cơ chế chiếu của chúng tôi rất quan trọng đối với hiệu suất cao. So với CLIPRe, DeCap-NND giải mã thêm các nhúng văn bản lần lượt gần nhất dẫn đến hiệu suất cao hơn, cho thấy bộ giải mã của chúng tôi có thể tạo ra câu mô tả nhiều hơn. DeCap-VD đạt được hiệu suất kém hơn, chứng minh rằng có là một khoảng cách lớn về phương thức giữa nhúng hình ảnh CLIP và nhúng văn bản, chứng minh sự cần thiết của cơ chế chiếu của chúng ta.

#### 4.2.2 CHỈ DẪN LIÊN MIỀN

Chúng tôi đánh giá các phương pháp sau đây trên MSCOCO và Flickr30K trong bối cảnh liên miền, trong đó dữ liệu đào tạo và dữ liệu thử nghiệm đến từ các tập dữ liệu khác nhau. (1) Zhao et al. (2020) tạo ra cặp hình ảnh-văn bản cho miền đích bằng cách sử dụng mô hình truy xuất được đào tạo trên miền nguồn. (2) Magic (Su et al., 2022) tinh chỉnh GPT-2 trên dữ liệu văn bản từ miền nguồn. (3) CLIPRe-S sử dụng dữ liệu văn bản từ miền nguồn dưới dạng thư viện ảnh. (4) DeCap đào tạo bộ giải mã trên dữ liệu văn bản từ

source domain. (5) **DeCap-TT** trains the decoder on text data from the source domain and uses captions from the target domain to construct the support memory.

**Results.** Table 3 shows the results. Unlike the traditional cross-domain method (Zhao et al., 2020) which relies on paired source domain data and requires training on the target domain, recent CLIP-based text-only methods require text-only data from the source domain for training. DeCap significantly outperforms other text-only methods, e.g., Magic (Su et al., 2022) and CLIPRe-S, on the cross-domain evaluation. Moreover, if the text data from the target domain is accessible, DeCap-TT significantly improves the captioning performance (e.g., CIDEr is improved from 44.4% to 63.1%) without any additional training. It simply employs text embedding from the target domain as the support memory. These results demonstrate the strong capabilities of DeCap in cross-domain generalization and the effectiveness of our projection-based decoding mechanism.

Methods	Data				MSCOCO to Flickr30K				Flickr30K to MSCOCO			
	S.P.	S.T.	T.I.	T.T.	B@4	M	C	S	B@4	M	C	S
Zhao et al. (2020)*	✓	✓	✓	✓	24.1	19.5	52.8	-	-	-	-	-
Magic		✓			6.2	12.2	17.5	5.9	5.2	12.5	18.3	5.7
CLIPRe-S		✓			9.8	16.7	30.1	10.3	6.0	16.0	26.5	10.2
DeCap-VD		✓			6.5	13.8	19.1	7.0	3.6	13.7	9.4	6.7
DeCap-NND		✓			12.0	15.5	28.6	10.1	7.5	15.9	28.0	9.6
DeCap		✓			16.3	17.9	35.7	11.1	12.1	18.0	44.4	10.9
DeCap-TT		✓			17.7	20.2	42.0	13.8	19.7	20.9	63.1	13.9

Table 3: Cross-domain image captioning evaluation. “\*” means using CIDEr optimization (Rennie et al., 2017). (“S.P.”: Source paired data; “S.T.”: Source text data; “T.I.”: Target image data; “T.T.”: Target text data).

#### 4.3 VIDEO CAPTIONING

In this section, we apply DeCap to the video captioning task. We conduct the experiments on MSR-VTT (Xu et al., 2016), Activity-Captions (Caba Heilbron et al., 2015), and VATEX (Wang et al., 2019). Notably, we only download 5182 raw test videos out of 6000 VATEX public test videos because some videos are unavailable. In Activity-Captions, we use ground-truth proposals following Krishna et al. (2017). We apply the same DeCap for video captioning. We consider three different data sources for decoder training: (1) **Generic corpus**. We train our decoder on Book Corpus which is a generic corpus used for unsupervised learning of language models. (2) **Image captions**. We train our decoder on captions from MSCOCO and CC3M, which are collected or annotated for image captioning tasks. (3) **Video captions**. We extract the text annotations in the training set of video captioning datasets. The former two can be viewed as the zero-shot video captioning setting without any video-related data for training.

At inference, we use a pooling mechanism on frame-level features to obtain a video-level feature. Specifically, for each proposal, we directly randomly sample  $k$  frames  $f_1, f_2, \dots, f_k$  from the clip. We use a mean pooling mechanism on the frame-level features extracted by the CLIP image encoder to obtain a video-level feature. In all experiments,  $k$  is set to 10.

**Results.** Table 4 shows the results. DeCap trained on image captions outperforms the recent zero-shot captioning approaches on standard captioning metrics and achieves competitive results on CLIP-S and CLIP-S<sup>Ref</sup> metrics. Notably, unlike other methods, DeCap does not directly take CLIP visual-text similarity as the optimization objective. Moreover, DeCap trained on video captions can further improve performance. These results demonstrate that DeCap can easily apply to video captioning with a simple random sampling strategy and temporal mean pooling mechanism.

#### 4.4 ABLATION STUDY

**The size of training data.** A key question is how much text data we need to train the decoder from scratch. To investigate the effect of training data size, we sample different scale data from MSCOCO. At inference, we use the same support memory (full training set, 560K captions) for all experiments. The results are in Figure 2 (**left**). Overall, DeCap benefits from a large data size. Compared with training on the full set, the CIDEr score drops from 91.2% to 81.5% when using only 1% of data (5.6K captions). The result indicates that DeCap is data-efficient. It shows a promising direction in its application in data-limited scenarios.

miền nguồn. (5) DeCap-TT đào tạo bộ giải mã trên dữ liệu văn bản từ miền nguồn và sử dụng chủ thích từ miền mục tiêu để xây dựng bộ nhớ hỗ trợ.

Kết quả. Bảng 3 cho thấy kết quả. Không giống như phương pháp miền chéo truyền thống (Zhao et al., 2020) dựa trên dữ liệu miền nguồn được ghép nối và yêu cầu đào tạo trên miền đích, các phương pháp chỉ văn bản dựa trên CLIP gần đây yêu cầu dữ liệu chỉ văn bản từ miền nguồn để đào tạo. DeCap vượt trội đáng kể so với các phương pháp chỉ văn bản khác, ví dụ, Magic (Su et al., 2022) và CLIPRe-S, trên đánh giá liên miền. Hơn nữa, nếu dữ liệu văn bản từ miền đích có thể truy cập được, DeCap-TT cải thiện đáng kể hiệu suất chủ thích (ví dụ, CIDEr được cải thiện từ 44,4% lên 63,1%) không cần bất kỳ đào tạo bổ sung nào. Nó chỉ đơn giản sử dụng những văn bản từ miền mục tiêu làm hỗ trợ bộ nhớ. Những kết quả này chứng minh khả năng mạnh mẽ của DeCap trong việc khai hóa xuyên miền và tính hiệu quả của cơ chế giải mã dựa trên phép chiếu của chúng tôi.

Phương pháp	Dữ liệu SPSTTTT B@4 MC				MSCOCO đến Flickr30K				Flickr30K đến MSCOCO								
	SP	ST	TT	B@4	MC	SB@4	MC	S	SP	ST	TT	S					
Zhao và cộng sự (2020)						24.1	19.5	52.8	-	-	-	-					
Magic CLIPRe-S						6.2	12.2	17.5	5.9	16.7	5.2	12.5	18.3	5.7			
DeCap-VD						9.8	30.1	10.3			6.0	16.0	26.5	10.2			
DeCap-NND									6,5	13,8	19,1	7,0	12,0	3,6	13,7	9,4	6,7
Bộ nhớ									15,5	28,6	10,1	16,3	17,9	7,5	15,9	28,0	9,6
DeCap-TT									35,7	11,1	17,7	20,2	42,0	12,1	18,0	44,4	10,9
									13,8					19,7	20,9	63,1	13,9

Bảng 3: Đánh giá chủ thích hình ảnh đa miền. “\*” có nghĩa là sử dụng tối ưu hóa CIDEr (Rennie et al., 2017). (“SP”: Dữ liệu ghép nối nguồn; “ST”: Dữ liệu văn bản nguồn; “TI”: Dữ liệu hình ảnh mục tiêu; “TT”: Dữ liệu văn bản mục tiêu).

#### 4.3 PHỤ ĐỀ CHO VIDEO

Trong phần này, chúng tôi áp dụng DeCap vào nhiệm vụ chủ thích video. Chúng tôi tiến hành các thí nghiệm trên MSR-VTT (Xu và cộng sự, 2016), Activity-Captions (Caba Heilbron và cộng sự, 2015) và VATEX (Wang và cộng sự, 2019). Đáng chú ý, chúng tôi chỉ tải xuống 5182 video thử nghiệm thô trong số 6000 video thử nghiệm công khai VATEX vì một số video không khả dụng. Trong Activity-Captions, chúng tôi sử dụng các đề xuất thực tế sau Krishna et al. (2017). Chúng tôi áp dụng cùng một DeCap cho phụ đề video. Chúng tôi xem xét ba nguồn dữ liệu để đào tạo bộ giải mã: (1) Ngữ liệu chung. Chúng tôi đào tạo bộ giải mã của mình trên Ngữ liệu Sách là một ngữ liệu chung được sử dụng cho việc học không giám sát các mô hình ngôn ngữ. (2) Chủ thích hình ảnh. Chúng tôi đào tạo bộ giải mã của chúng tôi về các chủ thích từ MSCOCO và CC3M, được thu thập hoặc chủ thích cho nhiệm vụ chủ thích hình ảnh. (3) Chủ thích video. Chúng tôi trích xuất các chủ thích văn bản trong tập huấn luyện của bộ dữ liệu chủ thích video. Hai cái trước có thể được xem như là cái đặt chủ thích video không có cảnh quay không có bất kỳ dữ liệu liên quan đến video nào để đào tạo.

Khi suy ra, chúng tôi sử dụng cơ chế gộp các tính năng ở cấp độ khung hình để có được tính năng ở cấp độ video. Cụ thể, đối với mỗi đề xuất, chúng tôi lấy mẫu ngẫu nhiên trực tiếp k khung hình  $f_1, f_2, \dots, f_k$  từ clip. Chúng tôi sử dụng cơ chế gộp trung bình trên các tính năng cấp khung được trích xuất bởi bộ mã hóa hình ảnh CLIP để có được tính năng cấp độ video. Trong tất cả các thí nghiệm, k được đặt thành 10.

Kết quả. Bảng 4 cho thấy kết quả. DeCap được đào tạo trên chủ thích hình ảnh vượt trội hơn so với gần đây phương pháp chủ thích không cần quay phim dựa trên số liệu chủ thích chuẩn và đạt được kết quả cạnh tranh trên số liệu CLIP-S và CLIP-SRef. Đáng chú ý là không giống như các phương pháp khác, DeCap không trực tiếp lấy CLIP tương tự văn bản trực quan là mục tiêu tối ưu hóa. Hơn nữa, DeCap được đào tạo về phụ đề video có thể cải thiện hiệu suất hơn nữa. Những kết quả này chứng minh rằng DeCap có thể dễ dàng áp dụng cho việc phụ đề video với chiến lược lấy mẫu ngẫu nhiên đơn giản và cơ chế gộp trung bình theo thời gian.

#### 4.4 NGHIÊN CỨU PHÁ VẾT TẮC

Kích thước của dữ liệu đào tạo. Một câu hỏi quan trọng là chúng ta cần bao nhiêu dữ liệu văn bản để đào tạo bộ giải mã từ đầu. Để nghiên cứu tác động của kích thước dữ liệu đào tạo, chúng tôi lấy mẫu dữ liệu tý lệ khác nhau từ MSCOCO. Khi suy luận, chúng tôi sử dụng cùng một bộ nhớ hỗ trợ (bộ đào tạo đầy đủ, 560K chủ thích) cho tất cả các thí nghiệm. Kết quả được thể hiện trong Hình 2 (bên trái). Nhìn chung, DeCap được hưởng lợi từ kích thước dữ liệu lớn. So với việc đào tạo trên toàn bộ bộ, điểm CIDEr giảm từ 91,2% xuống 81,5% khi chỉ sử dụng 1% dữ liệu (5,6 nghìn chủ thích). Kết quả cho thấy DeCap có hiệu quả về dữ liệu. Nó cho thấy một triển vọng ứng dụng của nó trong các tình huống dữ liệu hạn chế.

Methods	Setting	Metrics				
		B@4	M	C	CLIP-S <sup>Ref</sup>	CLIP-S
Results on MSR-VTT test set						
VNS-GRU <sup>†</sup> (Chen et al., 2020)	Supervised	45.3	29.9	53.0	0.739	0.626
SemSynAN <sup>†</sup> (Perez-Martin et al., 2021)		46.4	30.4	51.9	0.733	0.619
UVC-VI	Trained on VATEX-Chinese (Wang et al., 2019)	38.9	27.8	44.5	-	-
ZeroCap <sup>†</sup>	Zero-shot	2.3	12.9	5.8	0.739	0.710
MAGIC <sup>†</sup>		5.5	13.3	7.4	0.628	0.566
Tewel et al. (2022a) <sup>†</sup>		3.0	14.6	11.3	0.785	<b>0.775</b>
DeCap-BookCorpus		6.0	12.7	12.3	0.772	0.719
DeCap-CC3M		6.2	14.9	15.0	<b>0.792</b>	0.736
DeCap-COCO		<b>14.7</b>	<b>20.4</b>	<b>18.6</b>	0.761	0.697
CLIPRe-MSR	MSR-VTT text only	10.2	18.8	19.9	<b>0.835</b>	<b>0.852</b>
DeCap-VD-MSR		5.9	16.3	10.2	0.765	0.722
DeCap-NND-MSR		13.1	20.2	24.4	0.805	0.771
DeCap-MSR		<b>23.1</b>	<b>23.6</b>	<b>34.8</b>	0.823	0.770
Results on ActivityNet- Caption ae-test (Lei et al., 2020)						
Reasoner (Liang et al., 2022a)	Supervised	12.5	16.4	30.0	-	-
PDVC (Wang et al., 2021a)		11.8	15.9	27.3	-	-
DeCap-BookCorpus	Zero-shot	0.4	4.4	10.0	0.734	0.750
DeCap-CC3M		0.7	5.3	12.4	<b>0.761</b>	<b>0.814</b>
DeCap-COCO		<b>1.1</b>	<b>6.6</b>	<b>15.0</b>	0.727	0.753
CLIPRe-ACT	ActivityNet-Captions text only	1.4	8.2	15.1	<b>0.830</b>	<b>0.871</b>
DeCap-VD-ACT		1.1	6.6	10.2	0.682	0.712
DeCap-NND-ACT		1.9	8.3	15.5	0.745	0.775
DeCap-ACT		<b>2.3</b>	<b>9.4</b>	<b>20.6</b>	0.767	0.797
Results on VATEX public test set						
VaTeX (Wang et al., 2019)	Supervised	28.4	21.7	45.1	-	-
DeCap-BookCorpus	Zero-shot	4.1	9.9	11.8	0.761	0.731
DeCap-CC3M		7.3	12.6	18.4	<b>0.804</b>	<b>0.802</b>
DeCap-COCO		<b>13.1</b>	<b>15.3</b>	<b>18.7</b>	0.769	0.755
CLIPRe-VATEX	VATEX-Captions text only	11.1	17.0	27.1	<b>0.835</b>	<b>0.877</b>
DeCap-VD-VATEX		7.4	12.9	13.8	0.732	0.733
DeCap-NND-VATEX		14.8	18.1	32.4	0.809	0.811
DeCap-VATEX		<b>21.3</b>	<b>20.7</b>	<b>43.1</b>	0.834	0.824

Table 4: Video captioning evaluation results. “†” denotes the result from Tewel et al. (2022a). DeCap-BookCorpus, DeCap-CC3M, DeCap-COCO, DeCap-MSR, DeCap-ACT and DeCap-VATEX denote the model is trained on text data from Book Corpus, CC3M, MSCOCO, MSR-VTT, Activity-Captions, and DeCap-VATEX, respectively.

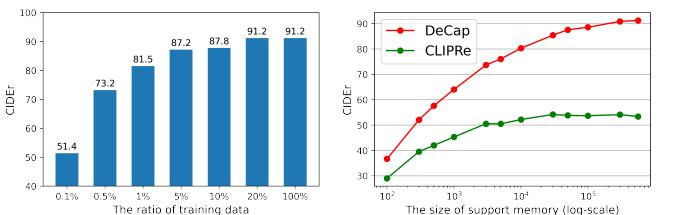


Figure 2: Ablation study on the training data size (left) and the support memory size (right).

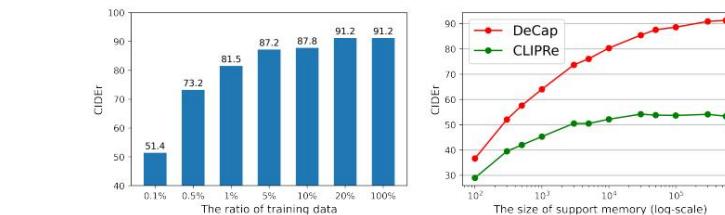
**The size of the support memory.** To investigate the effect of support memory size, we first train the language model on the full training set (560K captions). At inference, we randomly sample different ratio text embeddings as the support memory. The results are in Figure 2 (right). Overall, DeCap and CLIPRe both benefit from a large support memory. Moreover, when using only 1% data as the support memory, the performance drops slightly (3.8% performance drop in CIDEr). It indicates that we can maintain a relatively small support memory to achieve competitive results with acceptable storage and computation costs. Additionally, we provide a filtering strategy to reduce the number of support embeddings in Appendix E. We visualize the support memory and the projection embedding in Appendix G. We add an inference speed analysis to Appendix D.

## 5 CONCLUSION

We propose a simple framework for zero-shot captioning and introduce a lightweight visual-aware language decoder that is both data-efficient and computation-efficient. We propose a training-free mechanism to project the visual embedding to text embedding space, significantly reducing the modality gap issue. By combining the decoder with the projection mechanism, we significantly outperform existing zero-shot methods, establishing a new state-of-the-art in MSCOCO, MSR-VTT, and ActivityNet-Captions. In the future, our DeCap framework may be adapted to other zero-shot text generation problems, e.g., visual dialog.

Phương pháp	Cài đặt	Số liệu			
		B@4	MC	CLIP-SRef	CLIP-S
Kết quả trên bộ thử nghiệm MSR-VTT					
VNS-GRU <sup>†</sup> (Chen và cộng sự, 2020)	Được giám sát	45,3	29,9	53,0	0,739
SemSynAN <sup>†</sup> (Perez-Martin và cộng sự, 2021)	Được đào tạo về VATEX-Chinese (Wang et al., 2019)	46,4	30,4	51,9	0,733
UVC-VI		38,9	27,8	44,5	-
ZeroCap <sup>†</sup>	Zero-shot	2,3	12,9	5,8	0,739
MAGIC <sup>†</sup>		5,5	13,3	7,4	0,628
Tewel et al. (2022a) <sup>†</sup>		3,0	14,6	11,3	0,775
DeCap-BookCorpus		6,0	12,7	12,3	0,772
DeCap-CC3M		6,2	14,9	15,0	0,792
DeCap-COCO		<b>14,7</b>	<b>20,4</b>	<b>18,6</b>	0,697
CLIPRe-MSR	MSR-VTT text only	10,2	18,8	19,9	<b>0,835</b>
DeCap-VD-MSR		5,9	16,3	10,2	0,765
DeCap-NND-MSR		13,1	20,2	24,4	0,805
DeCap-MSR		<b>23,1</b>	<b>23,6</b>	<b>34,8</b>	0,823
Kết quả trên ActivityNet- Caption ae-test (Lei et al., 2020)					
Người lý luận (Liang và cộng sự, 2022a)	Được giám sát	12,5	16,4	30,0	11,8
PDVC (Wang và cộng sự, 2021a)		15,9	27,3	-	-
DeCap-BookCorpus	Không bắn	0,4	4,4	10,0	0,734
DeCap-CC3M DeCap-COCO CLIPRe-ACT		0,7	5,3	12,4	0,814
DeCap-VD-ACT DeCap-NND-ACT DeCap-ACT		1,1	6,6	15,0	0,727
Kết quả trên bộ thử nghiệm công khai VATEX					
VaTeX (Wang và cộng sự, 2019)	Có giám sát	4,1	9,9	11,8	7,3
DeCap-BookCorpus	Không bắn	18,4	13,1	15,3	18,7
DeCap-CC3M DeCap-COCO CLIPRe-VATEX		-	-	-	-
DeCap-VD-VATEX DeCap-NND-VATEX DeCap-VATEX		11,1	17,0	27,1	7,4
VATEX-Chi văn bản chủ thích		12,9	13,8	14,8	18,1

Bảng 4: Kết quả đánh giá phụ đề video. “†” biểu thị kết quả từ Tewel et al. (2022a). DeCap-BookCorpus, DeCap-CC3M, DeCap-COCO, DeCap-MSR, DeCap-ACT và DeCap-VATEX biểu thị mô hình được đào tạo trên dữ liệu văn bản từ Book Corpus, CC3M, MSCOCO, MSR-VTT, Activity-Captions, và DeCap-VATEX.



Hình 2: Nghiên cứu cắt bỏ trên kích thước dữ liệu đào tạo (trái) và kích thước bộ nhớ hỗ trợ (phải).

Kích thước của bộ nhớ hỗ trợ. Đề nghiên cứu tác động của kích thước bộ nhớ hỗ trợ, trước tiên chúng tôi đào tạo mô hình ngôn ngữ trên tập huấn luyện đầy đủ (560K chủ thích). Khi suy luận, chúng tôi lấy mẫu ngẫu nhiên những văn bản tỷ lệ khác nhau làm bộ nhớ hỗ trợ. Kết quả nằm trong Hình 2 (bên phải). Nhìn chung, DeCap và CLIPRe đều được hưởng lợi từ bộ nhớ hỗ trợ lớn. Hơn nữa, khi chỉ sử dụng 1% dữ liệu như bộ nhớ hỗ trợ, hiệu suất giảm nhẹ (hiệu suất giảm 3.8% trong CIDEr). Nó chỉ ra rằng chúng ta có thể duy trì một bộ nhớ hỗ trợ tương đối nhỏ để đạt được kết quả cạnh tranh với chi phí lưu trữ và tính toán chấp nhận được. Ngoài ra, chúng tôi cung cấp một chiến lược lọc để giảm số lượng những hỗ trợ trong Phụ lục E. Chúng tôi hình dung bộ nhớ hỗ trợ và phép chiếu những vào Phụ lục G. Chúng tôi thêm phân tích tốc độ suy luận vào Phụ lục D.

## 5 KẾT LUẬN

Chúng tôi đã xuất một khuôn khổ đơn giản cho chú thích không có cảnh quay và giới thiệu một công cụ nhận biết hình ảnh nhẹ bộ giải mã ngôn ngữ vừa hiệu quả và dữ liệu vừa hiệu quả về tính toán. Chúng tôi đã xuất một chương trình đào tạo miễn phí cơ chế để chiếu những hình ảnh vào không gian những văn bản, giảm đáng kể vấn

## ACKNOWLEDGMENTS

This work is supported by National Key R&D Program of China under Grant No. 2020AAA0108800. This work is partially supported by the Fundamental Research Funds for the Central Universities (No. 226-2022-00051).

## REFERENCES

- Harsh Agrawal, Karan Desai, Yufei Wang, Xinlei Chen, Rishabh Jain, Mark Johnson, Dhruv Batra, Devi Parikh, Stefan Lee, and Peter Anderson. Nocaps: Novel object captioning at scale. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8948–8957, 2019.
- Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*, 2022.
- Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Spice: Semantic propositional image caption evaluation. In *European conference on computer vision*, pp. 382–398. Springer, 2016.
- Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Guided open vocabulary image captioning with constrained beam search. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 936–945, 2017.
- Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 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*, pp. 6077–6086, 2018.
- Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pp. 65–72, 2005.
- Manuele Barraco, Marcella Cornia, Silvia Cascianelli, Lorenzo Baraldi, and Rita Cucchiara. The unreasonable effectiveness of clip features for image captioning: An experimental analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4662–4670, 2022.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.
- Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pp. 961–970, 2015.
- Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3558–3568, 2021.
- Haoran Chen, Jianmin Li, and Xiaolin Hu. Delving deeper into the decoder for video captioning. In *ECAI 2020*, pp. 1079–1086. IOS Press, 2020.
- Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
- Yang Feng, Lin Ma, Wei Liu, and Jiebo Luo. Unsupervised image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4125–4134, 2019.

## LỜI CẢM ƠN

Công trình này được hỗ trợ bởi Chương trình R&D trọng điểm quốc gia của Trung Quốc theo Khoản tài trợ số 2020AAA0108800. Công trình này được hỗ trợ một phần bởi Quỹ nghiên cứu cơ bản cho các trường đại học trung ương (Số 226-2022-00051).

## TÀI LIỆU THAM KHẢO

Harsh Agrawal, Karan Desai, Yufei Wang, Xinlei Chen, Rishabh Jain, Mark Johnson, Dhruv Batra, Devi Parikh, Stefan Lee và Peter Anderson. Nocaps: Chú thích đối tượng mới ở quy mô lớn. Trong *Biên bản báo cáo Hội nghị quốc tế IEEE/CVF về thị giác máy tính*, trang 8948-8957, 2019.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, v.v. Flamingo: mô hình ngôn ngữ trực quan cho việc học ít lần. Bản in trước arXiv arXiv:2204.14198, 2022.

Peter Anderson, Basura Fernando, Mark Johnson và Stephen Gould. Spice: Đánh giá chú thích hình ảnh để xuất ngữ nghĩa. Trong *hội nghị châu Âu về thị giác máy tính*, trang 382-398. Nhà xuất bản Springer, 2016.

Peter Anderson, Basura Fernando, Mark Johnson và Stephen Gould. Chú thích hình ảnh từ vựng mở có hướng dẫn với tìm kiếm chùm bị hạn chế. Trong *Biên bản báo cáo Hội nghị năm 2017 về Phương pháp thực nghiệm trong Xử lý ngôn ngữ tự nhiên*, trang 936-945, 2017.

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould và Lei Zhang. Sự chú ý từ dưới lên và từ trên xuống để chú thích hình ảnh và trả lời câu hỏi trực quan. Trong *Biên bản báo cáo của hội nghị IEEE về thị giác máy tính và nhận dạng mẫu*, trang 6077-6086, 2018.

Satanjeev Banerjee và Alon Lavie. Meteor: Một thước đo tự động để đánh giá mt với mối tương quan được cải thiện với các phán đoán của con người. Trong *Biên bản báo cáo của hội thảo acl về các biện pháp đánh giá nội tại và bên ngoài cho dịch máy và/hoặc tóm tắt*, trang 65-72, 2005.

Manuele Barraco, Marcella Cornia, Silvia Cascianelli, Lorenzo Baraldi và Rita Cucchiara. Hiệu quả không hợp lý của các tính năng clip cho chú thích hình ảnh: Một phân tích thử nghiệm. Trong *Biên bản báo cáo của Hội nghị IEEE/CVF về Thị giác máy tính và Nhận dạng mẫu*, trang 4662-4670, 2022.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Các mô hình ngôn ngữ là những người học ít lần. Những tiến bộ trong hệ thống xử lý thông tin thần kinh, 33:1877-1901, 2020.

Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem và Juan Carlos Niebles. Activitynet: Một chuẩn mực video quy mô lớn để hiểu hoạt động của con người. Trong *Biên bản báo cáo của hội nghị IEEE về thị giác máy tính và nhận dạng mẫu*, trang 961-970, 2015.

Soravit Changpinyo, Piyush Sharma, Nan Ding và Radu Soricut. Conceptual 12m: Đầy mạnh quá trình đào tạo trước hình ảnh-văn bản quy mô web để nhận dạng các khái niệm trực quan dài. Trong *Biên bản báo cáo của Hội nghị IEEE/CVF về Thị giác máy tính và Nhận dạng mẫu*, trang 3558-3568, 2021.

Haoran Chen, Jianmin Li và Xiaolin Hu. Đò sâu hơn vào bộ giải mã để thêm phụ đề cho video. Trong *ECAI 2020*, trang 1079-1086. IOS Press, 2020.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollar và C Lawrence Zitnick. Chú thích coco của Microsoft: Máy chủ thu thập và đánh giá dữ liệu. Bản in trước arXiv arXiv:1504.00325, 2015.

Jacob Devlin, Ming-Wei Chang, Kenton Lee và Kristina Toutanova. Bert: Đào tạo trước các bộ biến đổi song hướng sâu để hiểu ngôn ngữ. Bản in trước arXiv arXiv:1810.04805, 2018.

Yang Feng, Lin Ma, Wei Liu và Jiebo Luo. Chú thích hình ảnh không giám sát. Trong *Biên bản báo cáo Hội nghị IEEE/CVF về Thị giác máy tính và Nhận dạng mẫu*, trang 4125-4134, 2019.

Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. In *International Conference on Learning Representations*, 2021.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. CLIPScore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 7514–7528, November 2021.

Yupan Huang, Hongwei Xue, Bei Liu, and Yutong Lu. Unifying multimodal transformer for bi-directional image and text generation. In *Proceedings of the 29th ACM International Conference on Multimedia*, pp. 1138–1147, 2021.

Chao Jia, Yinfai Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pp. 4904–4916. PMLR, 2021.

Norman P Jouppi, Cliff Young, Nishant Patil, David Patterson, Gaurav Agrawal, Raminder Bajwa, Sarah Bates, Suresh Bhatia, Nan Boden, Al Borchers, et al. In-datacenter performance analysis of a tensor processing unit. In *Proceedings of the 44th annual international symposium on computer architecture*, pp. 1–12, 2017.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *Proceedings of the IEEE international conference on computer vision*, pp. 706–715, 2017.

Iro Laina, Christian Rupprecht, and Nassir Navab. Towards unsupervised image captioning with shared multimodal embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7414–7424, 2019.

Jie Lei, Liwei Wang, Yelong Shen, Dong Yu, Tamara Berg, and Mohit Bansal. Mart: Memory-augmented recurrent transformer for coherent video paragraph captioning. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 2603–2614, 2020.

Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705, 2021.

Chen Liang, Wenguan Wang, Tianfei Zhou, and Yi Yang. Visual abductive reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15565–15575, 2022a.

Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. *arXiv preprint arXiv:2203.02053*, 2022b.

Fenglin Liu, Xian Wu, Chenyu You, Shen Ge, Yuexian Zou, and Xu Sun. Aligning source visual and target language domains for unpaired video captioning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021a.

Fenglin Liu, Chenyu You, Xian Wu, Shen Ge, Xu Sun, et al. Auto-encoding knowledge graph for unsupervised medical report generation. *Advances in Neural Information Processing Systems*, 34: 16266–16279, 2021b.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2018.

Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 375–383, 2017.

Ron Mokady, Amir Hertz, and Amit H Bermano. Clipcap: Clip prefix for image captioning. *arXiv preprint arXiv:2111.09734*, 2021.

Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo và Yin Cui. Phát hiện đối tượng từ vựng mở thông qua việc chung kết kiến thức ngôn ngữ và thị giác. Trong Hội nghị quốc tế về biểu diễn học tập, 2021.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras và Yejin Choi. CLIPScore: Một thước đo đánh giá không tham chiếu cho chú thích hình ảnh. Trong Biên bản báo cáo Hội nghị năm 2021 về Phương pháp thực nghiệm trong Xử lý ngôn ngữ tự nhiên, trang 7514–7528, tháng 11 năm 2021.

Yupan Huang, Hongwei Xue, Bei Liu và Yutong Lu. Thông nhất bộ chuyển đổi đa phương thức để tạo hình ảnh và văn bản hai chiều. Trong Biên bản báo cáo Hội nghị quốc tế lần thứ 29 của ACM về đa phương tiện, trang 1138–1147, 2021.

Chao Jia, Yinfai Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li và Tom Duerig. Mở rộng quy mô học tập biểu diễn ngôn ngữ thị giác và thị giác với giám sát văn bản nhiều. Trong Hội nghị quốc tế về học máy, trang 4904–4916. PMLR, 2021.

Norman P Jouppi, Cliff Young, Nishant Patil, David Patterson, Gaurav Agrawal, Raminder Bajwa, Sarah Bates, Suresh Bhatia, Nan Boden, Al Borchers, et al. Phân tích hiệu suất trong trung tâm dữ liệu của một đơn vị xử lý tensor. Trong Biên bản báo cáo của hội thảo quốc tế thường niên lần thứ 44 về kiến trúc máy tính, trang 1–12, 2017.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei và Juan Carlos Niebles. Sự kiện chú thích dày đặc trong video. Trong Biên bản báo cáo hội nghị quốc tế IEEE về thị giác máy tính, trang 706–715, 2017.

Iro Laina, Christian Rupprecht và Nassir Navab. Hướng tới chú thích hình ảnh không giám sát với nhung đa phương thức được chia sẻ. Trong Biên bản báo cáo Hội nghị quốc tế về thị giác máy tính của IEEE/CVF, trang 7414–7424, 2019.

Jie Lei, Liwei Wang, Yelong Shen, Dong Yu, Tamara Berg và Mohit Bansal. Mart: Bộ biến đổi tuần hoàn tăng cường bộ nhớ để chú thích đoạn video mạch lạc. Trong Biên bản báo cáo của Hội nghị thường niên lần thứ 58 của Hiệp hội Ngôn ngữ học tính toán, trang 2603–2614, 2020.

Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong và Steven Chu Hong Hoi. Căn chỉnh trước khi kết hợp: Học biểu diễn ngôn ngữ và thị giác với chứng cất động lượng. Những tiến bộ trong hệ thống xử lý thông tin thần kinh, 34:9694–9705, 2021.

Chen Liang, Wenguan Wang, Tianfei Zhou và Yi Yang. Suy luận suy diễn trực quan. Trong Biên bản báo cáo của Hội nghị IEEE/CVF về Thị giác máy tính và Nhận dạng mẫu, trang 15565–15575, 2022a.

Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung và James Zou. Mind the gap: Hiểu về khoảng cách phương thức trong học tập biểu diễn tương phản đa phương thức. Bản in trước arXiv arXiv:2203.02053, 2022b.

Fenglin Liu, Xian Wu, Chenyu You, Shen Ge, Yuexian Zou và Xu Sun. Căn chỉnh miền ngôn ngữ nguồn trực quan và ngôn ngữ đích cho phụ đề video không ghép đôi. Giao dịch IEEE về Phân tích mẫu và Trí tuệ máy móc, 2021a.

Fenglin Liu, Chenyu You, Xian Wu, Shen Ge, Xu Sun, et al. Đò thị kiến thức mã hóa tự động để tạo báo cáo y tế không giám sát. Những tiến bộ trong Hệ thống xử lý thông tin thần kinh, 34: 16266–16279, 2021b.

Ilya Loshchilov và Frank Hutter. Chính quy hóa phân rã trọng lượng tách rời. Trong Hội nghị quốc tế về Biểu diễn học tập, 2018.

Jiasen Lu, Caiming Xiong, Devi Parikh và Richard Socher. Biết khi nào cần nhìn: Sự chú ý thích ứng thông qua một linh canh trực quan để chú thích hình ảnh. Trong Biên bản báo cáo của hội nghị IEEE về thị giác máy tính và nhận dạng mẫu, trang 375–383, 2017.

Ron Mokady, Amir Hertz và Amit H Bermano. Clipcap: Tiền tố clip để chú thích hình ảnh. arXiv bản in trước arXiv:2111.09734, 2021.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311–318, 2002.

Jesus Perez-Martin, Benjamin Bustos, and Jorge Pérez. Improving video captioning with temporal composition of a visual-syntactic embedding. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3039–3049, 2021.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67, 2020.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7008–7024, 2017.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2556–2565, 2018.

Sheng Shen, Liunian Harold Li, Hao Tan, Mohit Bansal, Anna Rohrbach, Kai-Wei Chang, Zhewei Yao, and Kurt Keutzer. How much can clip benefit vision-and-language tasks? In *International Conference on Learning Representations*, 2021.

Sheng Shen, Liunian Harold Li, Hao Tan, Mohit Bansal, Anna Rohrbach, Kai-Wei Chang, Zhewei Yao, and Kurt Keutzer. How much can CLIP benefit vision-and-language tasks? In *International Conference on Learning Representations*, 2022. URL [https://openreview.net/forum?id=zf\\_Ll3HZWgy](https://openreview.net/forum?id=zf_Ll3HZWgy).

Yixuan Su, Tian Lan, Yahui Liu, Fangyu Liu, Dani Yogatama, Yan Wang, Lingpeng Kong, and Nigel Collier. Language models can see: Plugging visual controls in text generation. *arXiv preprint arXiv:2205.02655*, 2022.

Sandeep Subramanian, Adam Trischler, Yoshua Bengio, and Christopher J Pal. Learning general purpose distributed sentence representations via large scale multi-task learning. In *International Conference on Learning Representations*, 2018.

Yoad Tewel, Yoav Shalev, Roy Nadler, Idan Schwartz, and Lior Wolf. Zero-shot video captioning with evolving pseudo-tokens. *arXiv preprint arXiv:2207.11100*, 2022a.

Yoad Tewel, Yoav Shalev, Idan Schwartz, and Lior Wolf. Zerocap: Zero-shot image-to-text generation for visual-semantic arithmetic. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17918–17928, 2022b.

Kenneth Tran, Xiaodong He, Lei Zhang, Jian Sun, Cornelia Carapcea, Chris Thrasher, Chris Buehler, and Chris Sienkiewicz. Rich image captioning in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 49–56, 2016.

Kishore Papineni, Salim Roukos, Todd Ward và Wei-Jing Zhu. Bleu: phương pháp đánh giá tự động bản dịch máy. Trong Biên bản báo cáo của cuộc họp thường niên lần thứ 40 của Hiệp hội Ngôn ngữ học tính toán, trang 311–318, 2002.

Jesus Perez-Martin, Benjamin Bustos và Jorge Perez. Cải thiện phụ đề video bằng cách kết hợp thời gian nhúng cú pháp trực quan. Trong Biên bản báo cáo Hội nghị mùa đông IEEE/CVF về Ứng dụng của Thị giác máy tính, trang 3039–3049, 2021.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Các mô hình ngôn ngữ là những người học đa nhiệm không có giám sát. Blog OpenAI, 1(8):9, 2019.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Học các mô hình trực quan có thể chuyển giao từ giám sát ngôn ngữ tự nhiên. Trong Hội nghị quốc tế về học máy, trang 8748–8763. PMLR, 2021.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. Khám phá giới hạn của việc học chuyển giao với bộ chuyển đổi văn bản sang văn bản thông nhất. *J. Mach. Learn. Res.*, 21(140):1–67, 2020.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu và Mark Chen. Tạo hình ảnh có điều kiện văn bản phân cấp với clip tiềm ẩn. Bản in trước arXiv arXiv:2204.06125, 2022.

Shaoqing Ren, Kaiming He, Ross Girshick và Jian Sun. Faster r-cnn: Hướng tới phát hiện đối tượng theo thời gian thực với mạng để xuất vùng. Những tiến bộ trong hệ thống xử lý thông tin thần kinh, 28, 2015.

Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross và Vaibhava Goel. Đào tạo trình tự tự phê bình cho chủ thích hình ảnh. Trong Biên bản báo cáo của hội nghị IEEE về thị giác máy tính và nhận dạng mẫu, trang 7008–7024, 2017.

Piyush Sharma, Nan Ding, Sebastian Goodman và Radu Soricut. Chủ thích khái niệm: Một tập dữ liệu văn bản thay thế hình ảnh đã được làm sạch, có siêu ẩn danh để tạo chủ thích hình ảnh tự động. Trong Biên bản báo cáo của Hội nghị thường niên lần thứ 56 của Hiệp hội Ngôn ngữ học tính toán (Tập 1: Bài báo dài), trang 2556–2565, 2018.

Sheng Shen, Liunian Harold Li, Hao Tan, Mohit Bansal, Anna Rohrbach, Kai-Wei Chang, Zhewei Yao và Kurt Keutzer. Clip có thể mang lại lợi ích gì cho các nhiệm vụ về thị giác và ngôn ngữ? Trong Hội nghị quốc tế về Biểu diễn học tập, 2021.

Sheng Shen, Liunian Harold Li, Hao Tan, Mohit Bansal, Anna Rohrbach, Kai-Wei Chang, Zhewei Yao và Kurt Keutzer. CLIP có thể mang lại lợi ích gì cho các nhiệm vụ về thị giác và ngôn ngữ? Trong Hội nghị quốc tế về Biểu diễn học tập, 2022. URL [https://openreview.net/forum?id=zf\\_Ll3HZWgy](https://openreview.net/forum?id=zf_Ll3HZWgy).

Yixuan Su, Tian Lan, Yahui Liu, Fangyu Liu, Dani Yogatama, Yan Wang, Lingpeng Kong và Nigel Collier. Mô hình ngôn ngữ có thể thấy: Cảm các điều khiển trực quan vào việc tạo văn bản. Bản in trước arXiv arXiv:2205.02655, 2022.

Sandeep Subramanian, Adam Trischler, Yoshua Bengio và Christopher J Pal. Học các biểu diễn câu phân tán mục đích chung thông qua học tập đa nhiệm vụ quy mô lớn. Trong Hội nghị quốc tế về biểu diễn học tập, 2018.

Yoad Tewel, Yoav Shalev, Roy Nadler, Idan Schwartz và Lior Wolf. Phụ đề video zero-shot với các mã thông báo giả đang phát triển. Bản in trước arXiv arXiv:2207.11100, 2022a.

Yoad Tewel, Yoav Shalev, Idan Schwartz và Lior Wolf. Zerocap: Tạo ảnh thành văn bản Zero-shot cho số học ngữ nghĩa thị giác. Trong Biên bản báo cáo của Hội nghị IEEE/CVF về Thị giác máy tính và Nhận dạng mẫu, trang 17918–17928, 2022b.

Kenneth Tran, Xiaodong He, Lei Zhang, Jian Sun, Cornelia Carapcea, Chris Thrasher, Chris Buehler và Chris Sienkiewicz. Chủ thích hình ảnh phong phú trong tự nhiên. Trong Biên bản báo cáo hội nghị IEEE về thị giác máy tính và hội thảo nhận dạng mẫu, trang 49–56, 2016.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4566–4575, 2015.

Teng Wang, Ruimao Zhang, Zhichao Lu, Feng Zheng, Ran Cheng, and Ping Luo. End-to-end dense video captioning with parallel decoding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6847–6857, 2021a.

Wenhai Wang, Yifan Sun, Zongxin Yang, and Yi Yang. V<sup>2</sup>l: Leveraging vision and vision-language models into large-scale product retrieval. *arXiv preprint arXiv:2207.12994*, 2022.

Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. Vatex: A large-scale, high-quality multilingual dataset for video-and-language research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4581–4591, 2019.

Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. Simvilm: Simple visual language model pretraining with weak supervision. In *International Conference on Learning Representations*, 2021b.

Jiahong Wu, He Zheng, Bo Zhao, Yixin Li, Baoming Yan, Rui Liang, Wenjia Wang, Shipei Zhou, Guosen Lin, Yanwei Fu, et al. Large-scale datasets for going deeper in image understanding. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1480–1485. IEEE, 2019.

Yu Wu, Linchao Zhu, Lu Jiang, and Yi Yang. Decoupled novel object captioner. In *Proceedings of the 26th ACM international conference on Multimedia*, pp. 1029–1037, 2018.

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5288–5296, 2016.

Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Bin Xiao, Ce Liu, Lu Yuan, and Jianfeng Gao. Unified contrastive learning in image-text-label space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19163–19173, 2022.

Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022a.

Youngjae Yu, Jiwan Chung, Heeseung Yun, Jack Hessel, JaeSung Park, Ximing Lu, Prithviraj Ammanabrolu, Rowan Zellers, Ronan Le Bras, Gunhee Kim, et al. Multimodal knowledge alignment with reinforcement learning. *arXiv preprint arXiv:2205.12630*, 2022b.

Andy Zeng, Adrian Wong, Stefan Welker, Krzysztof Choromanski, Federico Tombari, Aveek Purohit, Michael Ryoo, Vikas Sindhwani, Johnny Lee, Vincent Vanhoucke, et al. Socratic models: Composing zero-shot multimodal reasoning with language. *arXiv preprint arXiv:2204.00598*, 2022.

Xuying Zhang, Xiaoshuai Sun, Yunpeng Luo, Jiayi Ji, Yiyi Zhou, Yongjian Wu, Feiyue Huang, and Rongrong Ji. Rstnet: Captioning with adaptive attention on visual and non-visual words. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15465–15474, 2021.

Wentian Zhao, Xinxiao Wu, and Jiebo Luo. Cross-domain image captioning via cross-modal retrieval and model adaptation. *IEEE Transactions on Image Processing*, 30:1180–1192, 2020.

Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8746–8755, 2020.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pp. 19–27, 2015.

Laurens Van der Maaten và Geoffrey Hinton. Hình dung dữ liệu bằng t-sne. *Tạp chí nghiên cứu học máy*, 9(11), 2008.

Ramakrishna Vedantam, C Lawrence Zitnick và Devi Parikh. Cider: Đánh giá mô tả hình ảnh dựa trên sự đồng thuận. Trong *Biên bản báo cáo của hội nghị IEEE về thị giác máy tính và nhận dạng mẫu*, trang 4566–4575, 2015.

Teng Wang, Ruimao Zhang, Zhichao Lu, Feng Zheng, Ran Cheng và Ping Luo. Phụ đề video dài đặc từ đầu đến cuối với giải mã song song. Trong *Biên bản báo cáo Hội nghị quốc tế về thị giác máy tính của IEEE/CVF*, trang 6847–6857, 2021a.

Wenhai Wang, Yifan Sun, Zongxin Yang và Yi Yang. V2 l: Tận dụng tầm nhìn và ngôn ngữ tầm nhìn mô hình vào quá trình thu thập sản phẩm quy mô lớn. *Bản in trước của arXiv arXiv:2207.12994*, 2022.

Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang và William Yang Wang. Vatex: Một bộ dữ liệu đa ngôn ngữ chất lượng cao, quy mô lớn dành cho nghiên cứu ngôn ngữ và video. Trong *Kỷ yếu Hội nghị quốc tế về thị giác máy tính của IEEE/CVF*, trang 4581–4591, 2019.

Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov và Yuan Cao. Simvilm: Huấn luyện trước mô hình ngôn ngữ trực quan đơn giản với sự giám sát yếu. Trong *Hội nghị quốc tế về biểu diễn học tập*, 2021b.

Jiahong Wu, He Zheng, Bo Zhao, Yixin Li, Baoming Yan, Rui Liang, Wenjia Wang, Shipei Zhou, Guosen Lin, Yanwei Fu, et al. Bộ dữ liệu quy mô lớn để hiểu sâu hơn về hình ảnh. Năm 2019 Hội nghị quốc tế về đa phương tiện và hội chợ triển lãm IEEE (ICME), trang 1480–1485. IEEE, 2019.

Yu Wu, Linchao Zhu, Lu Jiang và Yi Yang. Chú thích đối tượng tiêu thuyết tách rời. Trong *Kỷ yếu của Hội nghị quốc tế ACM lần thứ 26 về Đa phương tiện*, trang 1029–1037, 2018.

Jun Xu, Tao Mei, Ting Yao và Yong Rui. Msr-vtt: Một tập dữ liệu mô tả video lớn để kết nối video và ngôn ngữ. Trong *Biên bản báo cáo của hội nghị IEEE về thị giác máy tính và nhận dạng mẫu*, trang 5288–5296, 2016.

Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Bin Xiao, Ce Liu, Lu Yuan và Jianfeng Gao. Học tập tương phản thống nhất trong không gian hình ảnh-văn bản-nhân. Trong *Biên bản báo cáo của Hội nghị IEEE/CVF về Thị giác máy tính và Nhận dạng mẫu*, trang 19163–19173, 2022.

Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini và Yonghui Wu. Coca: Các chú thích tương phản là mô hình nền tảng hình ảnh-văn bản. *Bản in trước arXiv arXiv:2205.01917*, 2022a.

Youngjae Yu, Jiwan Chung, Heeseung Yun, Jack Hessel, JaeSung Park, Ximing Lu, Prithviraj Ammanabrolu, Rowan Zellers, Ronan Le Bras, Gunhee Kim, et al. Cân chỉnh kiến thức đa phương thức với học tăng cường. *Bản in trước arXiv arXiv:2205.12630*, 2022b.

Andy Zeng, Adrian Wong, Stefan Welker, Krzysztof Choromanski, Federico Tombari, Aveek Purohit, Michael Ryoo, Vikas Sindhwani, Johnny Lee, Vincent Vanhoucke, et al. Các mô hình Socratic: Soạn thảo lý luận đa phương thức zero-shot với ngôn ngữ. *Bản in trước arXiv arXiv:2204.00598*, 2022.

Xuying Zhang, Xiaoshuai Sun, Yunpeng Luo, Jiayi Ji, Yiyi Zhou, Yongjian Wu, Feiyue Huang và Rongrong Ji. Rstnet: Chú thích với sự chú ý thích ứng trên các từ trực quan và không trực quan. Trong *Biên bản hội nghị IEEE/CVF về thị giác máy tính và nhận dạng mẫu*, trang 15465–15474, 2021.

Wentian Zhao, Xinxiao Wu và Jiebo Luo. Chú thích hình ảnh liên thông qua truy xuất liên phương thức và điều chỉnh mô hình. *IEEE Transactions on Image Processing*, 30:1180–1192, 2020.

Linchao Zhu và Yi Yang. Actbert: Học các biểu diễn video-văn bản toàn cục-cục bộ. Trong *Biên bản báo cáo của hội nghị IEEE/CVF về thị giác máy tính và nhận dạng mẫu*, trang 8746–8755, 2020.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba và Sanja Fidler. Sắp xếp sách và phim: Hướng tới các giải thích trực quan giống như câu chuyện bằng cách xem phim và đọc sách. Trong *Biên bản báo cáo của hội nghị quốc tế IEEE về thị giác máy tính*, trang 19–27, 2015.

## A MORE IMPLEMENTATION DETAILS

We employ a frozen pre-trained Vit-B/32 CLIP model as our cross-modal feature extractor. We adopt a lighting 4-layer Transformer (Subramanian et al., 2018) with 4 attention heads as our decoder (hidden state size 768) following the details (Radford et al., 2019). A linear layer trained with the decoder is used to project the CLIP embedding from 512 to 768 dimensions. The training data size and hyper-parameter for different datasets are summarized in Table 5.

	MSCOCO	Flickr30K	CC3M	SS1M	MSR-VTT	Activity-Captions	Book Corpus
Training size	560K	30K	3M	978K	140K	37K	6M
Training steps	40K	20K	200K	150K	20K	8K	400K
Warmup steps	2K	2K	2K	2K	2K	1K	2K
Batch size				128			
Learning rate				$1e^{-5}$			
Label smoothing				0.1			
Optimizer	AdamW (Loshchilov & Hutter, 2018) with default hyperparameters						

Table 5: Training data size and hyper-parameter

## B DISCUSSION ABOUT THE RECONSTRUCTION

In Sec. 3.2.2, we introduce the CLIPRe and Nearest-neighbor Decoding (NND) method. Given an image and its CLIP image embedding, both CLIPRe and NND first retrieve the most relative text embedding  $\mathbf{m}_t$  in the support memory according to the image-text cosine similarity. CLIPRe then adopts the original sentence  $t$  of the  $\mathbf{m}_t$  as the caption. NND uses the decoder to generate a sentence  $t^*$  conditioned on  $\mathbf{m}_t$ . Ideally, the generated sentence  $t^*$  should be the same as the original sentence  $t$ , because the decoder learns to reconstruct  $t$  conditioned on the  $\mathbf{m}_t$ . However, according to the experiments in Sec. 4.2.1, we find NND outperforms CLIPRe in most metrics. To figure out the reason, we conduct the following experiments.

We first train our decoder on the MSCOCO training set with Eq. 1. To investigate the effect of the text decoder, we construct a new corpus  $T^* = \{t_1^*, t_2^*, \dots, t_N^*\}$ , where  $t^* = P_\theta(E_{text}(t))$ ,  $t$  is the original sentence in MSCOCO training set and  $t^*$  is the reconstructed sentence. We adopt this new corpus as the support memory.

Table 6 shows the results. The reconstructed dataset improves the performance of CLIPRe on all metrics, especially on CIDEr, from 53.4% to 63.6% (+10.2%). DeCap adopting the new corpus as the support memory could further improve the CIDEr score to 95.1% (+3.9%). The result demonstrates that the sentences generated by our method can better describe the images in MSCOCO. We think the reason is that our decoding process has a denoising effect, which can remove some outliers captions in the training set. Another open question here is whether such a denoised dataset can improve the performance of other fully supervised methods. We leave this as our future work.

	B@4	M	C	S
CLIPRe	12.4	20.4	53.4	14.8
CLIPRe-recons	14.9 (+2.5)	21.5 (+1.1)	63.6 (+10.2)	16.2 (+1.4)
DeCap	24.7	25.0	91.2	18.7
DeCap-recons	26.5 (+1.8)	24.9 (-0.1)	95.1 (+3.9)	18.6 (-0.1)

Table 6: Results on MSCOCO Karpathy-test split. CLIPRe-recons and DeCap-recons denote using the reconstructed corpus as the support memory.

## C PRETRAINING-FINETUNING.

An interesting question is whether DeCap can benefit from the pretraining-finetuning paradigm. Table 7 shows the results. Notably, we only use text data for training in both pre-training and fine-tuning. Compared to training on MSCOCO, the model trained on CC3M achieves better performance in the out-of-domain case, improving the CIDEr metric from 25.8% to 48.7%. This is

## MỘT CHI TIẾT THÊM VỀ VIỆC THỰC HIỆN

Chúng tôi sử dụng mô hình Vit-B/32 CLIP được đào tạo trước đóng lạnh làm trình trích xuất tính năng đa phương thức của chúng tôi. Chúng tôi áp dụng một mảng biến áp chiều sáng 4 lớp (Subramanian và cộng sự, 2018) với 4 đầu chú ý làm bộ giải mã của chúng tôi (kích thước trạng thái là 768) theo các chi tiết (Radford và cộng sự, 2019). Một lớp tuyến tính được đào tạo với bộ giải mã được sử dụng để chiều nhúng CLIP từ 512 đến 768 chiều. Kích thước dữ liệu đào tạo và siêu tham số cho các tập dữ liệu khác nhau được tóm tắt trong Bảng 5.

	MSCOCO	Flickr30K	CC3M	SS1M	MSR-VTT	Hoạt động-Chú thích	Sách	Corpus
Kích thước đào tạo	560K	30K	3M	978K	200K	140K	37K	6 phút
Các bước đào tạo	40K	20K	150K	2K	128	20K	8K	400K
Các bước khởi động	2K	2K	2K	2K	2K	2K	1K	2K
Kích thước lô								
Tỷ lệ học tập						1e-5		
Làm mịn nhãn						0.1		
Trình tối ưu hóa	AdamW (Loshchilov & Hutter, 2018) với các siêu tham số mặc định							

Bảng 5: Kích thước dữ liệu đào tạo và siêu tham số

## B THẢO LUẬN VỀ CÔNG TRÌNH TÁI TẠO

Trong Phần 3.2.2, chúng tôi giới thiệu phương pháp CLIPRe và Giải mã lân cận gần nhất (NND). Cho một hình ảnh và những hình ảnh CLIP của nó, cả CLIPRe và NND đều lấy văn bản có liên quan nhất trước tiên nhúng mt vào bộ nhớ hỗ trợ theo độ tương đồng cosin hình ảnh-văn bản. CLIPRe sau đó sử dụng câu gốc t của mt làm chủ thích. NND sử dụng bộ giải mã để tạo ra một câu  $t^*$  có điều kiện trên mt. Lý tưởng nhất là câu được tạo ra  $t^*$  phải giống như câu gốc  $t$ , vì bộ giải mã học cách tái tạo  $t$  có điều kiện trên mt. Tuy nhiên, theo các thí nghiệm trong Mục 4.2.1, chúng tôi thấy NND vượt trội hơn CLIPRe trong hầu hết các số liệu. Để tìm ra lý do đó, chúng tôi tiến hành các thí nghiệm sau.

Đầu tiên chúng tôi đào tạo bộ giải mã của mình trên bộ đào tạo MSCOCO với Công thức 1. Để nghiên cứu hiệu ứng của bộ giải mã văn bản, chúng tôi xây dựng một ngữ liệu  $T$  mới  $= \{t_1, t_2, \dots, t_N\}$ , trong đó  $t_i = P_\theta(E_{text}(t))$ ,  $t$  là câu gốc trong tập huấn luyện MSCOCO và  $t_i$  là câu được xây dựng lại. Chúng tôi áp dụng câu mới này ngữ liệu là bộ nhớ hỗ trợ.

Bảng 6 cho thấy kết quả. Bộ dữ liệu được xây dựng lại cải thiện hiệu suất của CLIPRe trên tất cả số liệu, đặc biệt là trên CIDEr, từ 53,4% lên 63,6% (+10,2%). DeCap áp dụng ngữ liệu mới làm bộ nhớ hỗ trợ có thể cải thiện thêm điểm CIDEr lên 95,1% (+3,9%). Kết quả chứng minh rằng các câu được tạo ra bởi phương pháp của chúng tôi có thể mô tả tốt hơn các hình ảnh trong MSCOCO. Chúng tôi nghĩ lý do là quá trình giải mã của chúng tôi có hiệu ứng khử nhiễu, có thể loại bỏ một số chú thích ngoại lệ trong tập huấn luyện. Một câu hỏi mở khác ở đây là liệu một tập dữ liệu khử nhiễu như vậy có thể cải thiện hiệu suất của các phương pháp được giám sát đầy đủ khác. Chúng tôi để lại đây như công việc tương lai của chúng tôi.

	B@4	Tổi	C	S
CLIPRe	12.4	20.4	53.4	14.8
Recons	14.9 (+2.5)	21.5 (+1.1)	63.6 (+10.2)	16.2 (+1.4)
DeCap	24.7	25.0	91.2	18.7
recons	26.5 (+1.8)	24.9 (-0.1)	95.1 (+3.9)	18.6 (-0.1)

Bảng 6: Kết quả phân tách thử nghiệm Karpathy của MSCOCO. CLIPRe-recons và DeCap-recons biểu thị bằng cách sử dụng ngữ liệu được tái tạo như là bộ nhớ hỗ trợ.

## C ĐÀO TẠO TRƯỚC-CHỈNH SỬA.

Một câu hỏi thú vị là liệu DeCap có thể hưởng lợi từ mô hình điều chỉnh trước khi đào tạo hay không.

Bảng 7 cho thấy kết quả. Đáng chú ý là chúng tôi chỉ sử dụng dữ liệu văn bản để đào tạo trong cả đào tạo trước và tinh chỉnh. So với đào tạo trên MSCOCO, mô hình được đào tạo trên CC3M đạt được hiệu suất tốt hơn trong trường hợp ngoài miền, cải thiện số liệu CIDEr từ 25,8% lên 48,7%. Đây là

Pre-training data	Fine-tuning data	Memory data	CIDEr			
			in	near	out	overall
MSCOCO	-	MSCOCO	65.2	47.8	25.8	45.9
CC3M	-	CC3M	34.7	35.9	<b>48.7</b>	38.3
CC3M	MSCOCO	MSCOCO	<b>72.7</b>	<b>61.9</b>	43.9	58.2
CC3M	-	MSCOCO	70.1	60.4	44.5	<b>58.6</b>

Table 7: Results of DeCap on NoCaps validation split. We only use the text data for both pre-training and fine-tuning.

because CC3M covers more diverse classes than MSCOCO. By fine-tuning the pre-trained model on MSCOCO, we find that the overall performance is greatly improved, obtaining an overall CIDEr of 58.2%. It indicates that our method benefits from the pretraining-finetuning paradigm. By directly changing the support memory without fine-tuning, DeCap achieves comparable performance as fine-tuning. It suggests that our method can be easily adapted to new domains without training, requiring only some text data from new domains.

## D THE INFERENCE SPEED

Table 8 shows the inference speed of DeCap. Decap is 113x faster than ZeroCap. Because DeCap does not involve gradient updates and multiple text encoder forwards during the inference. Besides, the decoder used in DeCap is more lightweight compared to the GPT-2 employed in ZeroCap. It is worth mentioning that the time cost of embedding projection is negligible compared to image encoding and text decoding.

	Image encoding (CLIP image encoder)	Embedding projection (1M support memory)	Language decoding	All	FPS
ZeroCap	32.68 ms	-	11285.36 ms	11318.04 ms	0.088
DeCap	31.75 ms	0.38 ms	68.54 ms	100.67 ms	9.933

Table 8: The inference speed of ZeroCap and DeCap. The experiment is conducted on a single Nvidia RTX2080Ti GPU. Both DeCap and ZeroCap do not use the beam search. We report the average time cost of captioning 100 images with batch size 1.

## E AN EFFICIENT STRATEGY TO REDUCE THE NUMBER OF SUPPORT EMBEDDINGS

To make DeCap more practical, we provide a method that does not degrade DeCap performance but can significantly reduce the number of support embeddings. In the original DeCap, we randomly sample sentences from the training set to construct the support memory. However, the semantics between sentences is highly repetitive. A simple but effective method is to filter the features in the support memory according to the cosine similarity. Specifically, given a text feature and the existing support memory, if the maximum cosine similarity between the feature and the support memory is greater than a threshold, the feature will not be stored in the support memory. We set the threshold to 0.8 and construct a new support memory with the filtering strategy. Table 9 shows that this strategy can significantly reduce the number of support embeddings from 1M to 0.14M and thus reduce the additional memory cost from 1.02GB to 0.14GB without performance degradation.

Similarity filter	The number of support embeddings	Additional memory cost	CIDEr
False	1M	1.02GB	42.2
False	0.14M (randomly sampled from 1M)	0.14GB	38.2 (-4.0)
True	0.14M (Filtering from 1M)	0.14GB	42.3 (+0.1)

Table 9: The result of filtering strategy. We use the same 1M sentences in this experiment.

Dữ liệu tiền đào tạo	Dữ liệu tinh chỉnh	Dữ liệu bộ nhớ	Rút ngắn	
			TỔNG	gắn ra
MSCOCO	-	MSCOCO	65,2	45,9
CC3M	-	CC3M	35,9	38,3
CC3M	MSCOCO	CC3M	43,9	58,2
CC3M	-	CC3M	-	58,6

Bảng 7: Kết quả của DeCap trên phân tách xác thực NoCaps. Chúng tôi chỉ sử dụng dữ liệu văn bản cho cả hai giai đoạn đào tạo trước và tinh chỉnh.

vì CC3M bao gồm nhiều lớp đa dạng hơn MSCOCO. Bằng cách tinh chỉnh mô hình được đào tạo trước trên MSCOCO, chúng tôi thấy rằng hiệu suất tổng thể được cải thiện đáng kể, đạt được CIDEr tổng thể của 58,2%. Điều này cho thấy phương pháp của chúng tôi được hưởng lợi từ mô hình tinh chỉnh trước khi đào tạo. Bằng cách thay đổi trực tiếp bộ nhớ hỗ trợ mà không cần tinh chỉnh, DeCap đạt được hiệu suất tương đương như tinh chỉnh. Nó cho thấy phương pháp của chúng tôi có thể dễ dàng thích ứng với các lĩnh vực mới mà không cần đào tạo, chỉ yêu cầu một số dữ liệu văn bản từ miền mới.

## D TỐC ĐỘ SUY LUẬN

Bảng 8 cho thấy tốc độ suy luận của DeCap. Decap nhanh hơn ZeroCap 113 lần. Bởi vì DeCap không liên quan đến việc cập nhật gradient và nhiều bộ mã hóa văn bản chuyển tiếp trong quá trình suy luận. Bên cạnh đó, bộ giải mã được sử dụng trong DeCap nhẹ hơn so với GPT-2 được sử dụng trong ZeroCap. Nó đáng nói đến là chi phí thời gian nhúng phép chiếu không đáng kể so với hình ảnh mã hóa và giải mã văn bản.

	Mã hóa hình ảnh (Bộ mã hóa hình ảnh CLIP)	Nhúng phép chiếu (Bộ nhớ hỗ trợ 1M)	Giải mã ngôn ngữ	Tổng	FPS
Không có nắp	32,68 giây	-	11285,36 ms	11318,04 ms	0,088
Bộ mõm	31,75 giây	0,38 giây	68,54 ms	100,67 ms	9,933

Bảng 8: Tốc độ suy luận của ZeroCap và DeCap. Thí nghiệm được tiến hành trên một GPU Nvidia RTX2080Ti. Cả DeCap và ZeroCap đều không sử dụng tìm kiếm chùm tia. Chúng tôi báo cáo chi phí thời gian trung bình để thêm chú thích cho 100 hình ảnh với kích thước lô là 1.

## E MỘT CHIẾN LƯỢC HIỆU QUẢ ĐỂ GIẢM SỐ LƯỢNG HỖ TRỢ NHUNG

Để làm cho DeCap thực tế hơn, chúng tôi cung cấp một phương pháp không làm giảm hiệu suất của DeCap nhưng có thể giảm đáng kể số lượng nhung hỗ trợ. Trong DeCap ban đầu, chúng tôi ngẫu nhiên các câu mẫu từ tập huấn luyện để xây dựng bộ nhớ hỗ trợ. Tuy nhiên, ngữ nghĩa giữa các câu có tính lặp lại cao. Một phương pháp đơn giản nhưng hiệu quả là lọc các tính năng trong hỗ trợ bộ nhớ theo độ tương đồng cosin. Cụ thể, cho một tính năng văn bản và hiện có bộ nhớ hỗ trợ, nếu độ tương đồng cosin tối đa giữa tính năng và bộ nhớ hỗ trợ là lớn hơn ngưỡng, tính năng sẽ không được lưu trữ trong bộ nhớ hỗ trợ. Chúng tôi đặt ngưỡng thành 0,8 và xây dựng bộ nhớ hỗ trợ mới với chiến lược lọc. Bảng 9 cho thấy chiến lược này có thể giảm đáng kể số lượng nhung hỗ trợ từ 1M xuống 0,14M và do đó giảm chi phí bộ nhớ bổ sung từ 1,02GB đến 0,14GB mà không làm giảm hiệu suất.

Bộ lọc tương tự	Số lượng nhung hỗ trợ	Chi phí bộ nhớ bổ sung	Rút ngắn
SAI	1M	1,02GB	42,2
SAI	0,14M (lấy mẫu ngẫu nhiên từ 1M)	0,14GB	38,2 (-4,0)
BỘNG VẤY	0,14M (Lọc từ 1M)	0,14GB	42,3 (+0,1)

Bảng 9: Kết quả của chiến lược lọc. Chúng tôi sử dụng cùng 1M câu trong thí nghiệm này.

## F PROMPT ENGINEERING

Prior work (Tewel et al., 2022b; Wang et al., 2021b) found that a prefix prompt “a picture of” improves the quality of decoded captions. We study the effect of the prompt on our special decoder trained with a text reconstruction task. We consider two decoders trained on CC3M-text and Book Corpus, respectively. At inference, we take the “prefix embedding + prompts” as the input of the decoder. We test a set of prompts as shown in Table 10. The results show that the decoder trained on Book Corpus benefits from the prompt engineering, while the decoder trained on CC3M hurts from the prompt engineering in most cases. Although CC3M is a dataset collected automatically from the Web, it is well-filtered by some human-designed strategies. Therefore, most of the text data in CC3M are caption-related, and the redundant prompt design will destroy its original text structure, resulting in performance degradation. BookCorpus is a popular large-scale text corpus, especially for unsupervised learning of language models. Most of the sentences in Bookcorpus are not originally intended to describe pictures. A well-designed prompt can allow the decoder to generate sentences that match the captioning task.

Prompt	DeCap-BookCorpus	DeCap-CC3M
None	21.8	42.1 (+0.0)
“A photo of”	20.4 (-1.4)	38.3 (-3.8)
“A picture of”	20.8 (-1.0)	36.8 (-5.3)
“There is/are”	27.1 (+5.3)	40.7 (-1.4)
“See! There is/are”	30.4 (+8.6)	40.9 (-1.2)
“Attention! There is/are”	31.9 (+10.1)	42.2 (+0.1)

Table 10: Zero-shot captioning results on MSCOCO Karpathy split with different prompts.

## G VISUALIZATION OF THE EMBEDDINGS

Figure 3 shows the support embeddings and category embeddings from MSCOCO. The support embeddings from the clip text encoder are divided into different clusters according to the semantics. In Figure 4(a), we sample 500 image-text pairs from the MSCOCO training set and visualize their embeddings extracted by the CLIP encoder. As we can see, there is a clear modality gap between CLIP text embeddings and image embeddings. Figure 4(b) and Figure 4(c) show that the projection method can effectively reduce this modality gap. Figure 4(d) shows that the projected embedding is close to the embeddings of human-labeled captions in the latent space. Besides, compared to CLIPRe embedding, the projected embedding is more central, indicating that the projected embedding absorbs the information of the support embeddings and nicely preserves the visual information.

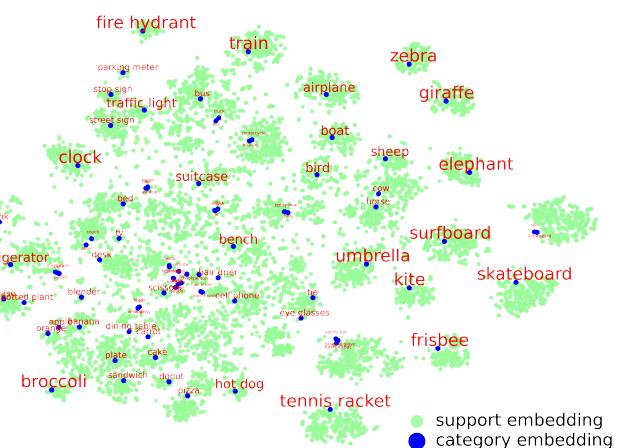


Figure 3: Visualization of support embeddings and category embeddings from MSCOCO in 2D space by t-SNE (Van der Maaten & Hinton, 2008). We randomly sample 10,000 embeddings from the support memory for visualization.

## F KỸ THUẬT NHANH CHÓNG

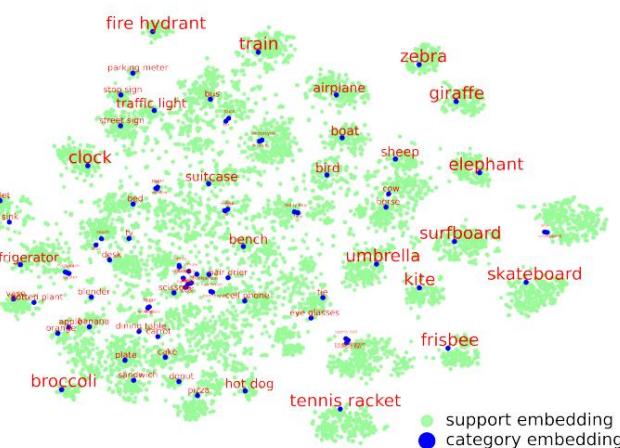
Công trình trước đây (Tewel et al., 2022b; Wang et al., 2021b) phát hiện ra rằng tiền tố nhắc nhở “một bức ảnh của” cải thiện chất lượng của phụ đề được giải mã. Chúng tôi nghiên cứu tác động của lời nhắc trên bộ giải mã đặc biệt của chúng tôi được đào tạo với một nhiệm vụ tái tạo văn bản. Chúng tôi xem xét hai bộ giải mã được đào tạo trên CC3M-text và Book Corpus, tương ứng. Khi suy luận, chúng tôi lấy “những tiền tố + lời nhắc” làm đầu vào của bộ giải mã. Chúng tôi kiểm tra một tập hợp các lời nhắc như được hiển thị trong Bảng 10. Kết quả cho thấy bộ giải mã đã được đào tạo trên Book Corpus được hưởng lợi từ kỹ thuật nhanh chóng, trong khi bộ giải mã được đào tạo trên CC3M gây tổn hại từ kỹ thuật nhanh chóng trong hầu hết các trường hợp. Mặc dù CC3M là một tập dữ liệu được thu thập tự động từ Web, nó được lọc tốt bởi một số chiến lược do con người thiết kế. Do đó, hầu hết các văn bản dữ liệu trong CC3M liên quan đến chủ thích và thiết kế lời nhắc dựa trên các đặc điểm gốc của nó, dẫn đến suy giảm hiệu suất. BookCorpus là một kho văn bản quy mô lớn phổ biến, đặc biệt là đối với việc học không giám sát các mô hình ngôn ngữ. Hầu hết các câu trong Bookcorpus ban đầu không có ý định mô tả hình ảnh. Một lời nhắc được thiết kế tốt có thể cho phép bộ giải mã tạo ra các câu phù hợp với nhiệm vụ chủ thích.

Nhắc nhở	DeCap-BookCorpus	DeCap-CC3M
Không có	21,8	42,1 (+0,0)
“Một bức ảnh của”	20,4 (-1,4)	38,3 (-3,8)
“Một bức ảnh của”	20,8 (-1,0)	36,8 (-5,3)
“Có/có”	27,1 (+5,3)	40,7 (-1,4)
“Nhìn kia! Có/có”	30,4 (+8,6)	40,9 (-1,2)
“Chú ý! Có/có”	31,9 (+10,1)	42,2 (+0,1)

Bảng 10: Kết quả chú thích Zero-shot trên MSCOCO Karpathy được chia thành các lời nhắc khác nhau.

## G HÌNH DUNG CÁC PHẦN NHÚNG

Hình 3 cho thấy các nhúng hỗ trợ và nhúng danh mục từ MSCOCO. Các nhúng hỗ trợ từ bộ mã hóa văn bản clip được chia thành các cụm khác nhau theo ngữ nghĩa. Trong Hình 4(a), chúng tôi lấy mẫu 500 cặp hình ảnh-văn bản từ bộ đào tạo MSCOCO và trực quan hóa chúng nhúng được trích xuất bởi bộ mã hóa CLIP. Như chúng ta có thể thấy, có một khoảng cách phương thức rõ ràng giữa nhúng văn bản CLIP và nhúng hình ảnh. Hình 4(b) và Hình 4(c) cho thấy phép chiếu phương pháp có thể làm giảm hiệu quả khoảng cách phương thức này. Hình 4(d) cho thấy rằng nhúng dự kiến gần với các nhúng của chủ thích được gắn nhãn của con người trong không gian tiềm ẩn. Bên cạnh đó, so với nhúng CLIPRe, nhúng được chiếu ở trung tâm hơn, cho thấy rằng nhúng được chiếu hấp thụ thông tin của nhúng hỗ trợ và bảo toàn thông tin trực quan một cách tốt đẹp.



Hình 3: Hình ảnh hóa các nhúng hỗ trợ và nhúng danh mục từ MSCOCO ở dạng 2D không gian của t-SNE (Van der Maaten & Hinton, 2008). Chúng tôi lấy mẫu ngẫu nhiên 10.000 nhúng từ bộ nhớ hỗ trợ cho việc hình dung.

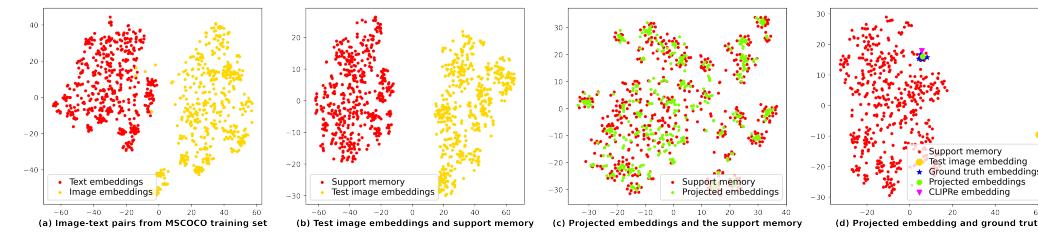
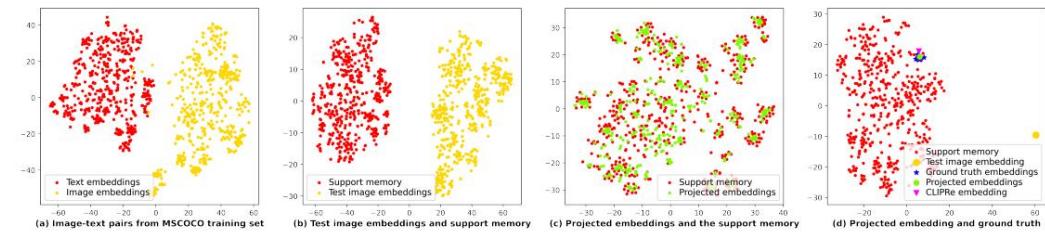


Figure 4: Visualization of embeddings in 2D space by t-SNE. We construct the support memory using text embeddings from MSCOCO training set and randomly sample 500 embeddings from the support memory for visualization.

## H EXAMPLES OF GENERATED CAPTIONS

We visualize the generated captions of some images from the MSCOCO Karpathy-test split in Figure 5. We show the captions generated by the DeCap model trained on MSCOCO and CC3M. The captions from DeCap-MSCOCO and DeCap-CC3M have visible style differences.



Hình 4: Hình ảnh hóa các nhúng trong không gian 2D bằng t-SNE. Chúng tôi xây dựng bộ nhớ hỗ trợ bằng cách sử dụng nhúng văn bản từ bộ đào tạo MSCOCO và lấy mẫu ngẫu nhiên 500 nhúng từ bộ nhớ hỗ trợ để hình ảnh hóa.

## H VÍ DỤ VỀ CHÚ THÍCH ĐƯỢC TẠO

Chúng tôi hình dung các chú thích được tạo ra của một số hình ảnh từ phép chia tách kiểm tra Karpathy của MSCOCO trong Hình 5. Chúng tôi hiển thị các chú thích được tạo ra bởi mô hình DeCap được đào tạo trên MSCOCO và CC3M. Các chú thích từ DeCap-MSCOCO và DeCap-CC3M có sự khác biệt rõ ràng về phong cách.

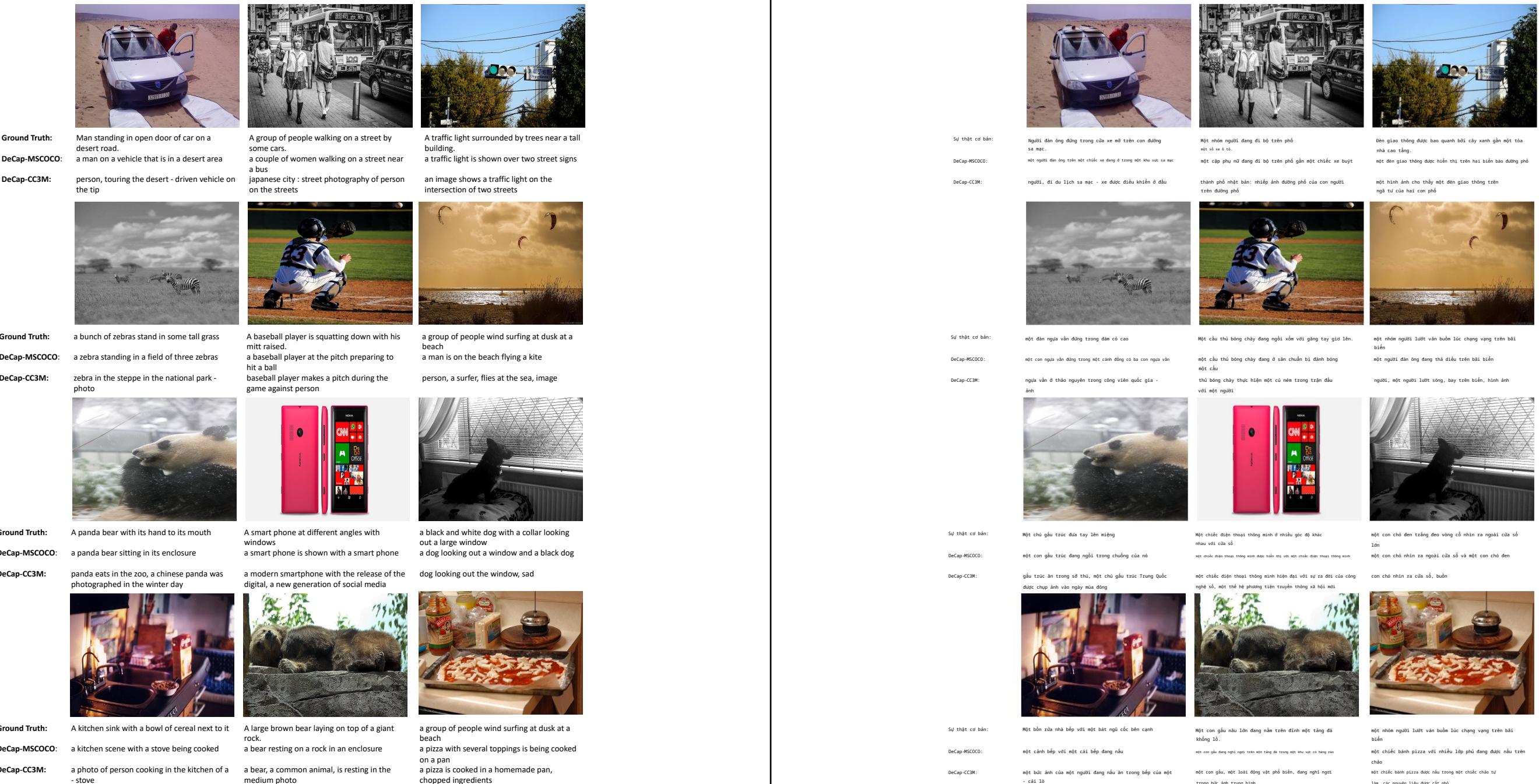


Figure 5: Generated captions for images from the MSCOCO Karpathy-test split. DeCap-MSCOCO and DeCap-CC3M denote DeCap trained on the MSCOCO training set and CC3M training set, respectively.

Hình 5: Tạo chú thích cho hình ảnh từ phép chia tách thử nghiệm Karpathy của MSCOCO. DeCap-MSCOCO và DeCap-CC3M biểu thị DeCap được đào tạo trên bộ đào tạo MSCOCO và bộ đào tạo CC3M, tương ứng.