UNIVERSITÀ
DI TRENTO

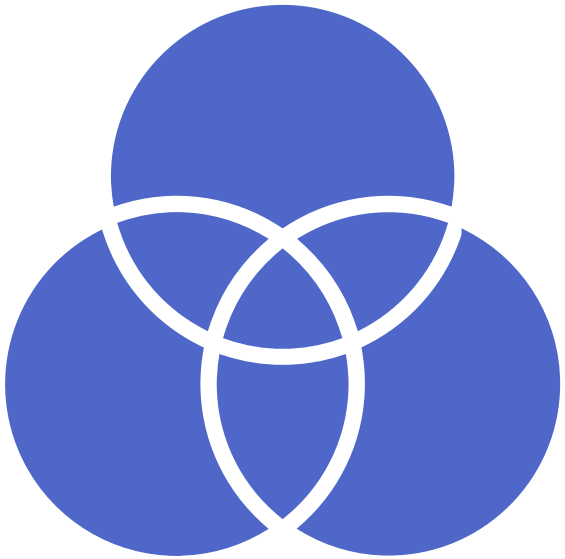
Zero-Shot Composed Image Retrieval with Textual Inversion

Advanced Computer Vision

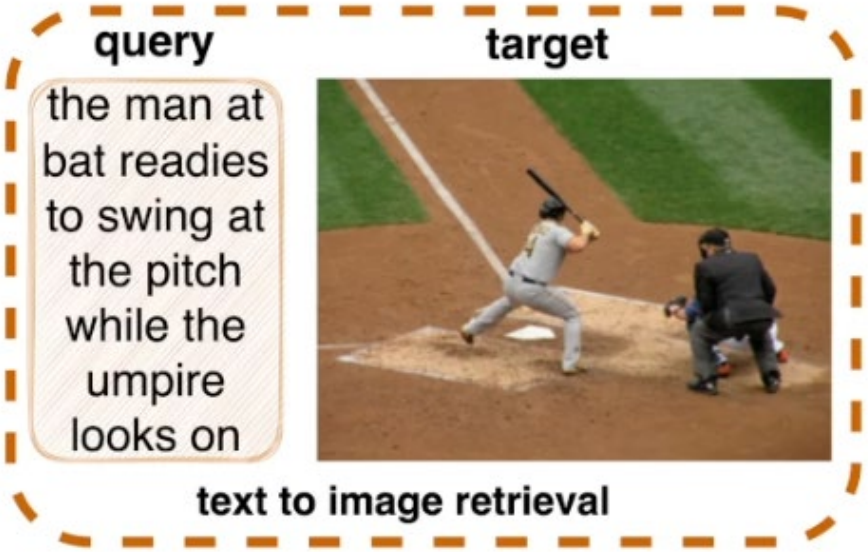
Adnan Irshad, Hira Afzal

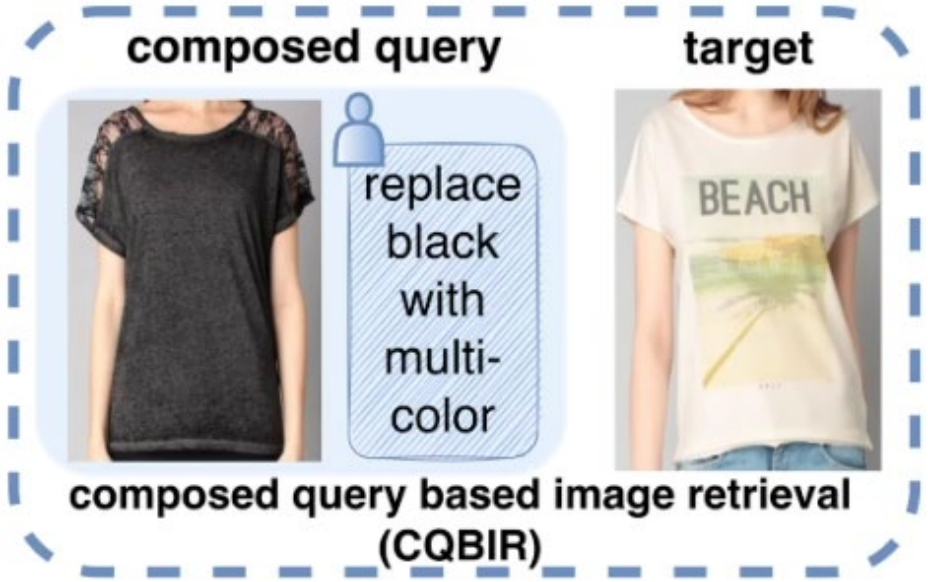
Outline

- Introduction
- Related Works
- ZS-CIR with Textual Inversion
- Results & Experiments
- Limitations with ZS-CIR
- Improved Works



What do we expect to do?

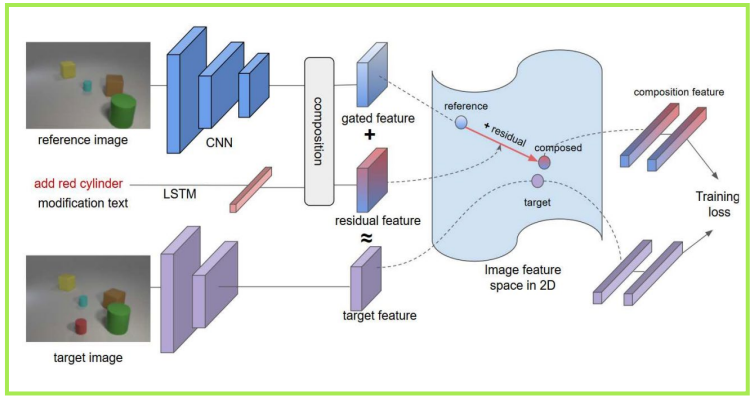




Related Works-1

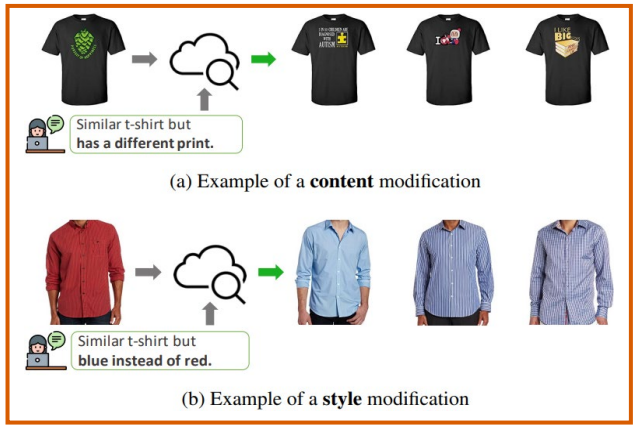
Composing Text and Image for Image Retrieval - An Empirical Odyssey

(CVPR 2019)

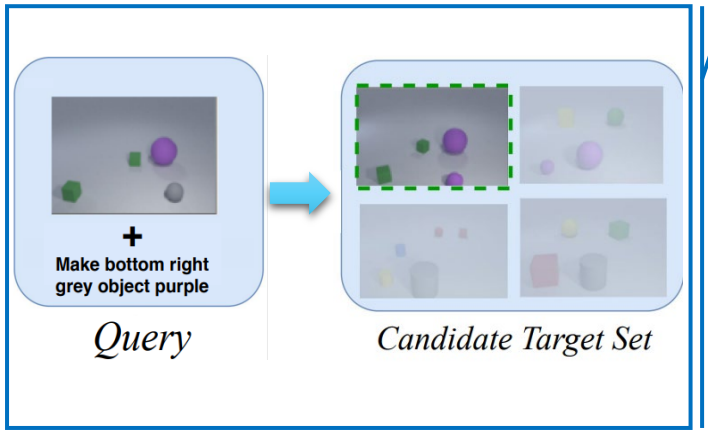


CoSMo: Content-style modulation for image retrieval with text feedback

(CVPR 2021)

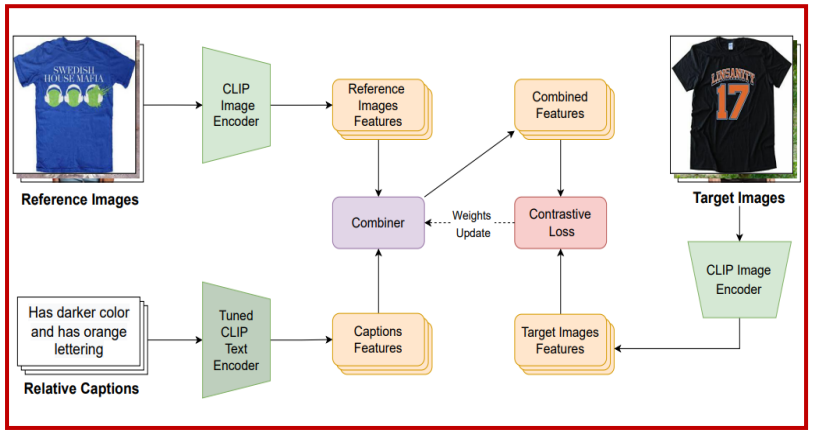


2019 | 2020 | 2021 | 2022



Composed Query Image Retrieval using Locally Bounded Features

(CVPR 2020)

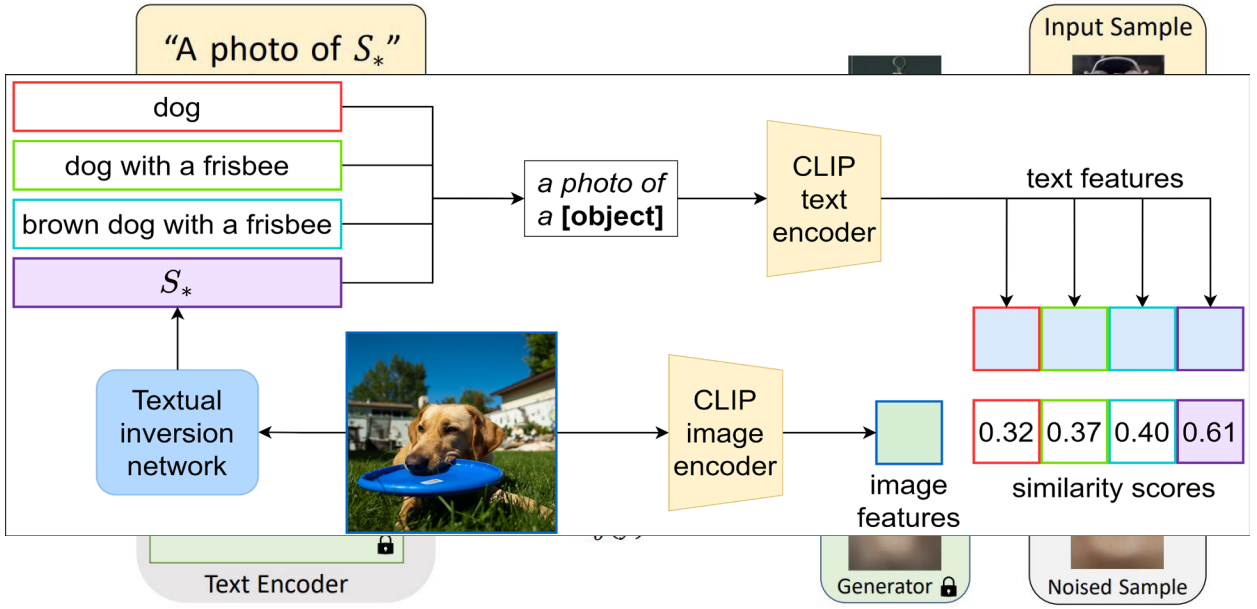


Conditioned and composed image retrieval combining and partially fine-tuning clip-based features

(CVPR 2022)

Literature Review-2: Textual Inversion

- This technique maps visual information (from images) in a textual format (pseudo-words).
- Personalized text-to-image synthesis
- Expands CLIP vocabulary by defining a new pseudo-word **S*** which encapsulates the visual information of the image
- In CoIR task, following methods used textual inversion:
  - **PALAVRA (**PersonAlizing LAnguage Vision RepresentAtions**)**
  - **Pic2Word**
  - **SEARLE**



An Image is Worth One Word: Personalizing Text-to-Image Generation using Textual Inversion (CVPR 2022)

PALAVRA (PersonAlizing LAnguage Vision RepresentAtions)

- **Two Stages:**
  - **Pre-training:**
  - **Optimization:**
- PALAVRA needs a labeled image-caption dataset for pre-training and a specific input word concept for each retrieval query during optimization.

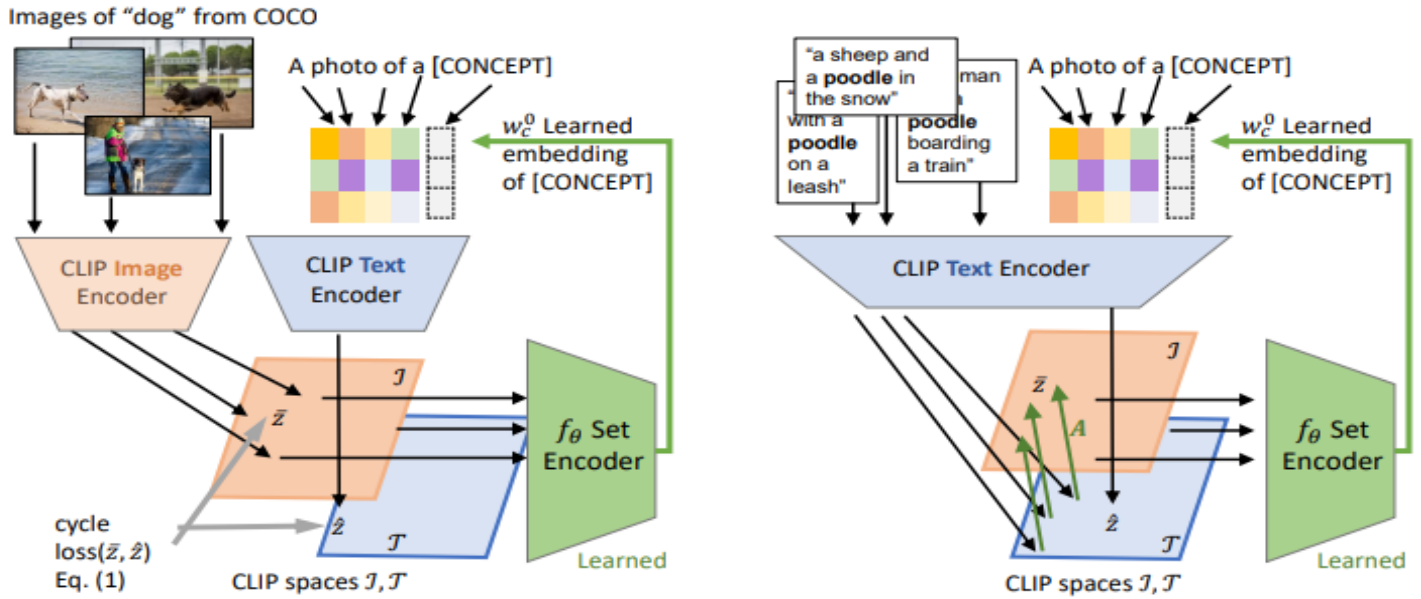


Fig. 3. Architecture outline: Learning f_θ. We start with a *large-scale-data* training step. A set encoder f_θ is trained to map CLIP-space output embeddings to a code in CLIP's input space. It is alternatingly trained with a batch of either image examples (left), or sentence examples (right) with augmented concept types. We use a cycle loss by mapping the code back to CLIP's output embedding, using a template sentence.

Pic2Word

The Pic2Word, a method that deals with ZS-CIR.

Pic2Word uses a network trained on a large dataset (CC3M with 3 million images) using a specific type of loss (cycle contrastive loss).

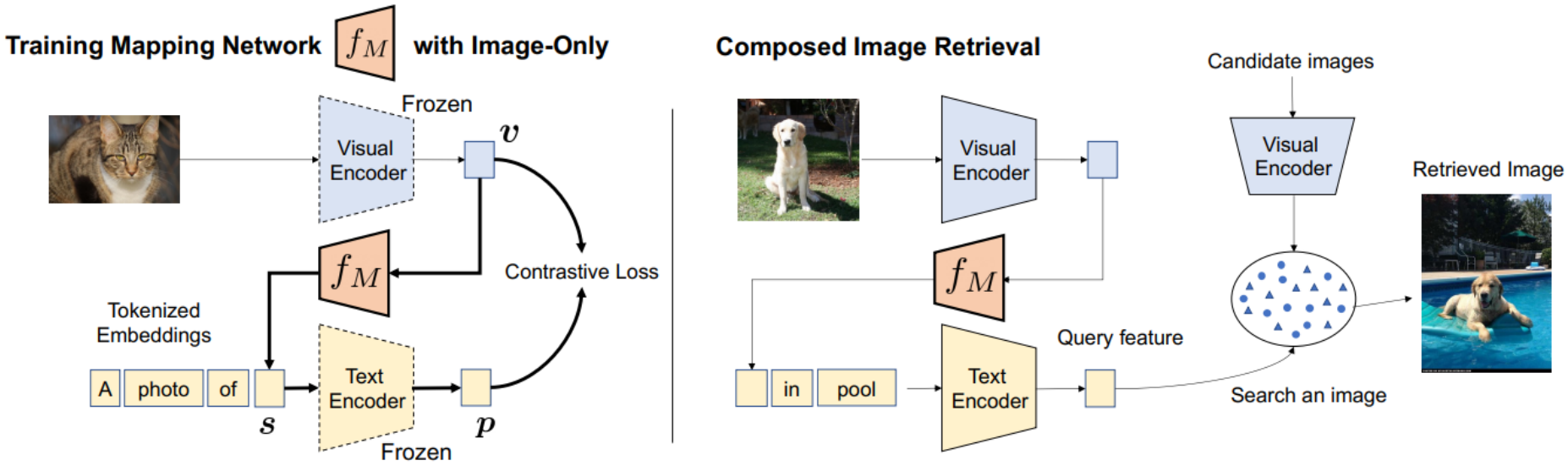
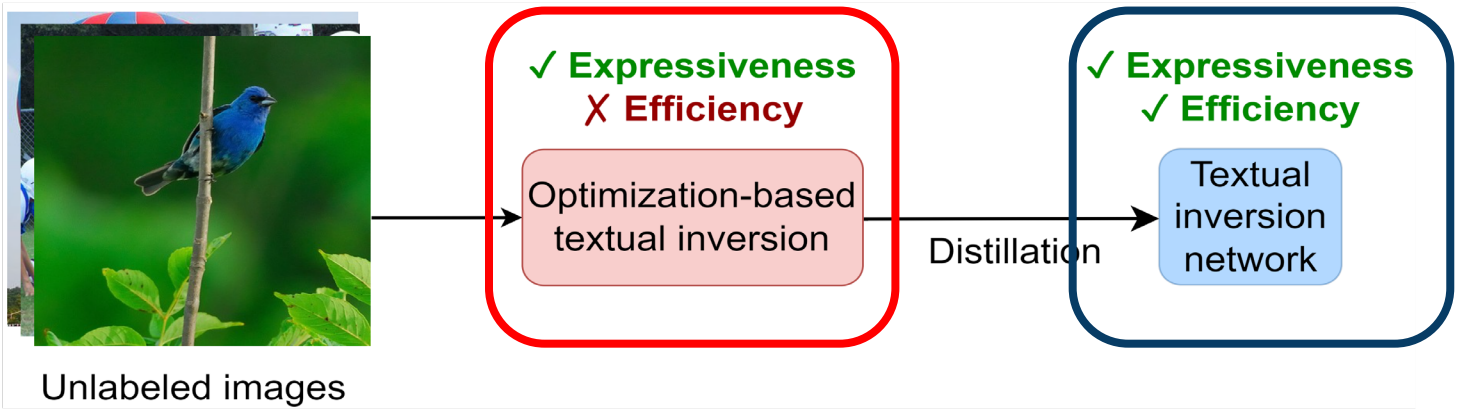


Figure 2. **Left**: Overview of training our Pic2Word mapping network. Given a frozen visual and text encoder, the mapping network, f_M, is optimized to minimize the contrastive loss between the image embedding and the language embedding generated by the pseudo word token s. **Right**: Overview of inference. The estimated token is used to fill in the given prompt sentence.

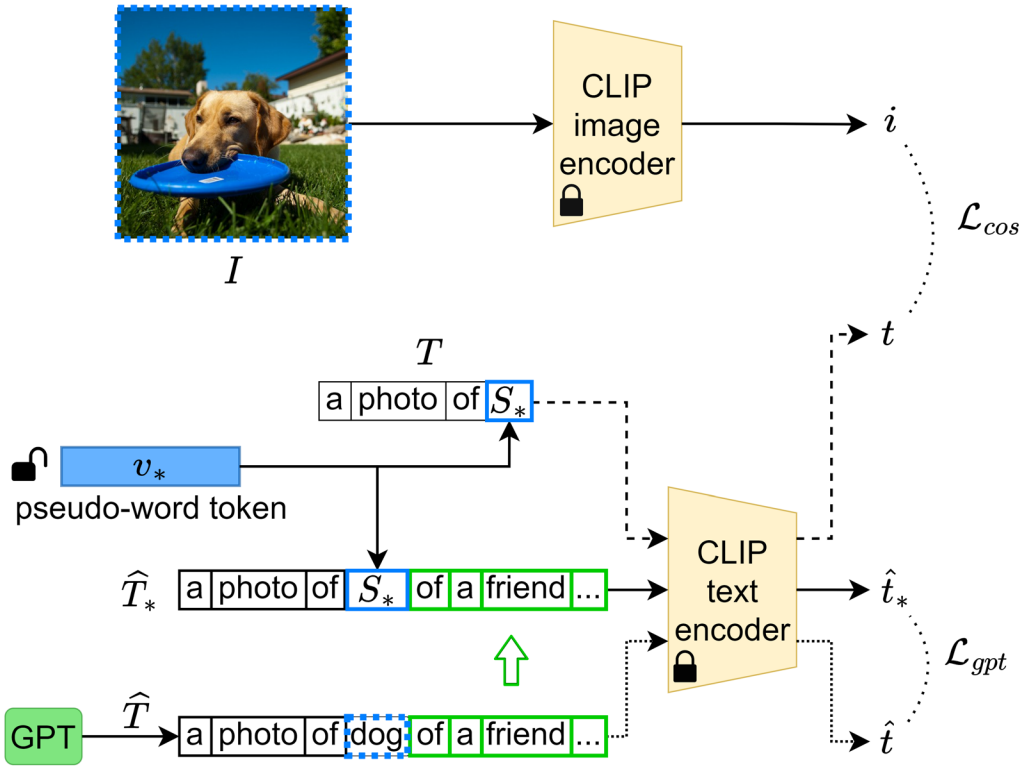
Zero-Shot Composed Image Retrieval with Textual Inversion **(SEARLE)**

- CLIP-based method that addresses CIR in a zero-shot manner, thus without requiring a labeled training dataset.
- Given an **unlabeled** dataset, SEARLE training involves two stages:
  1. **Optimization-based Textual Inversion (OTI)**
  2. **Textual inversion network**



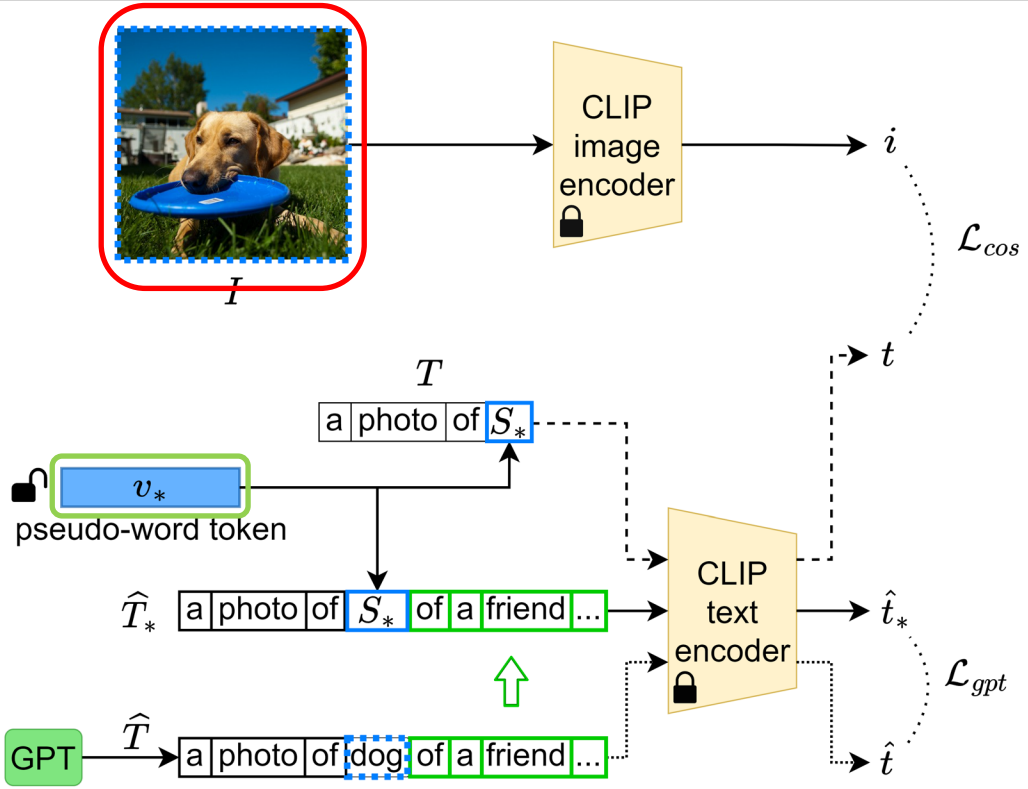
SEARLE Training: OTI

- Step 1: OTI approach iteratively generate a pseudo-word token per image from an unlabeled dataset, enabling image retrieval using text-based queries.



SEARLE Training: OTI

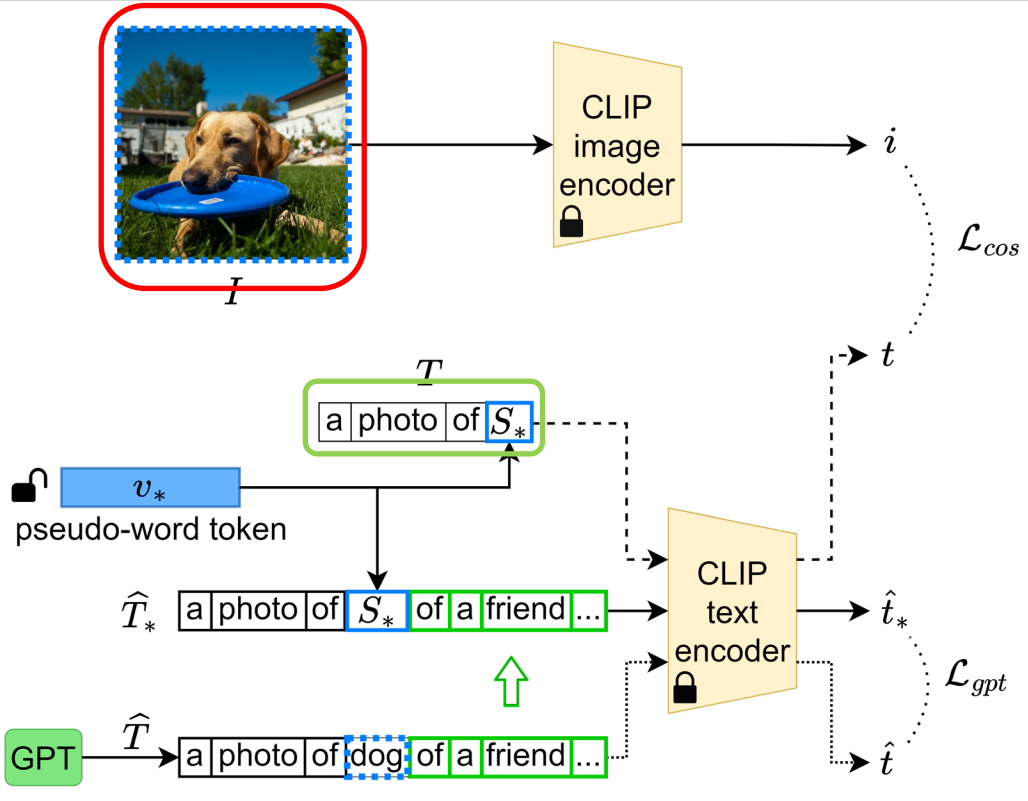
- Step 1: OTI approach to iteratively generate a pseudo-word token per image from an unlabeled dataset, enabling image retrieval using text-based queries.



- **Initialization**: A pseudo-word token **v∗** is randomly initialized. And associating the pseudo-word **S*** to it.

SEARLE Training: OTI

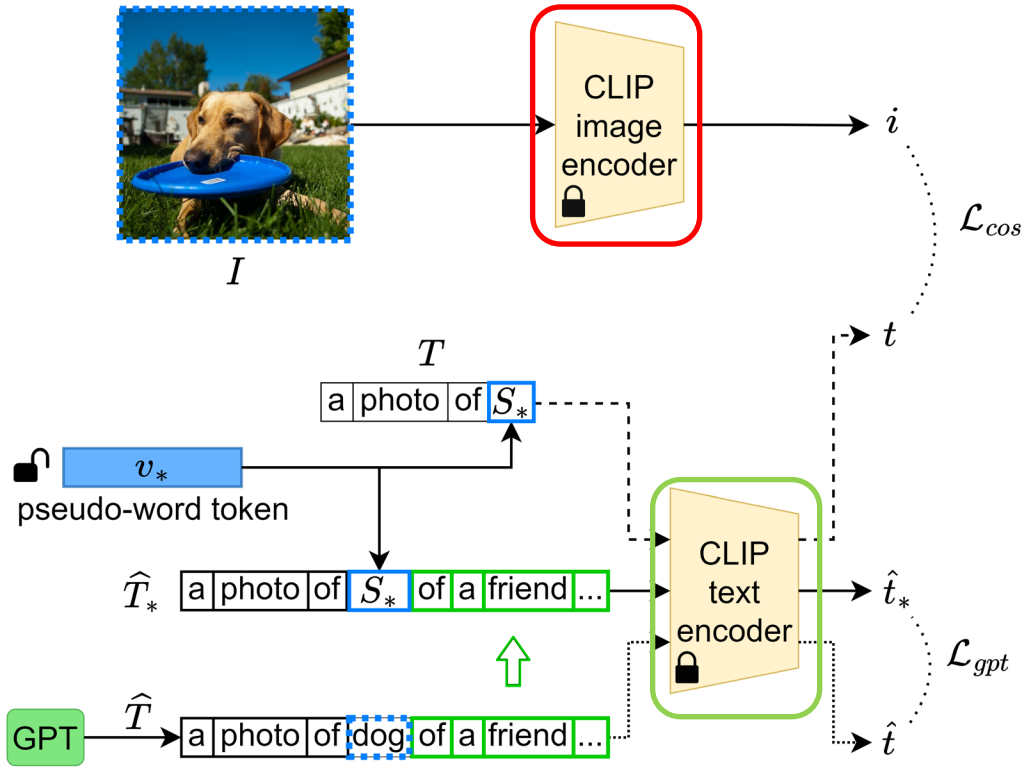
- Step 1: OTI approach to iteratively generate a pseudo-word token per image from an unlabeled dataset, enabling image retrieval using text-based queries.



- **Initialization**: A pseudo-word token **v∗** is randomly initialized. And associating the pseudo-word **S*** to it.
- **Template Creation**: A template sentence (*T*), like "**a photo of *S∗***", is created and fed to the CLIP text encoder to get text features.

SEARLE Training: OTI

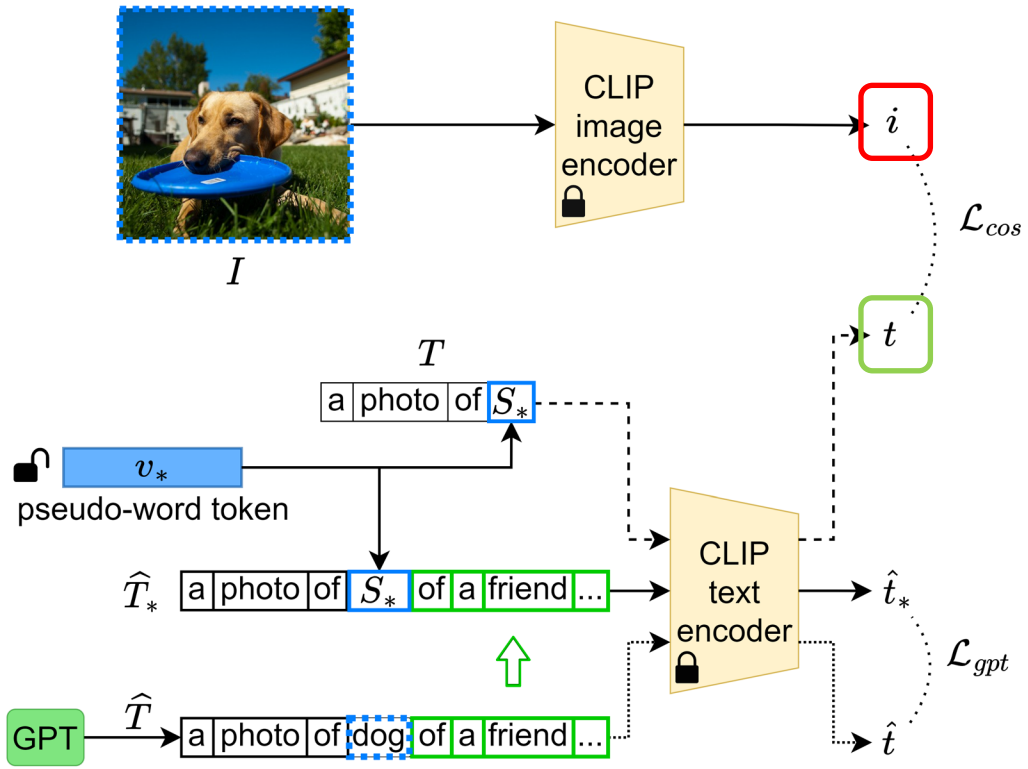
- Step 1: OTI approach to iteratively generate a pseudo-word token per image from an unlabeled dataset, enabling image retrieval using text-based queries.



- **Initialization**: A pseudo-word token **v∗** is randomly initialized. And associating the pseudo-word **S*** to it.
- **Template Creation**: A template sentence (*T*), like "**a photo of *S∗***", is created and fed to the CLIP text encoder to get text features.
- **CLIP Encoders**: The CLIP text encoder $\psi_T(T)$ and image encoder $\psi_I(I)$ are used to extract the features of template text **i.e.,** $t = \psi_T(T)$ and Image **i.e.,** $i = \psi_I(I)$.

SEARLE Training: OTI

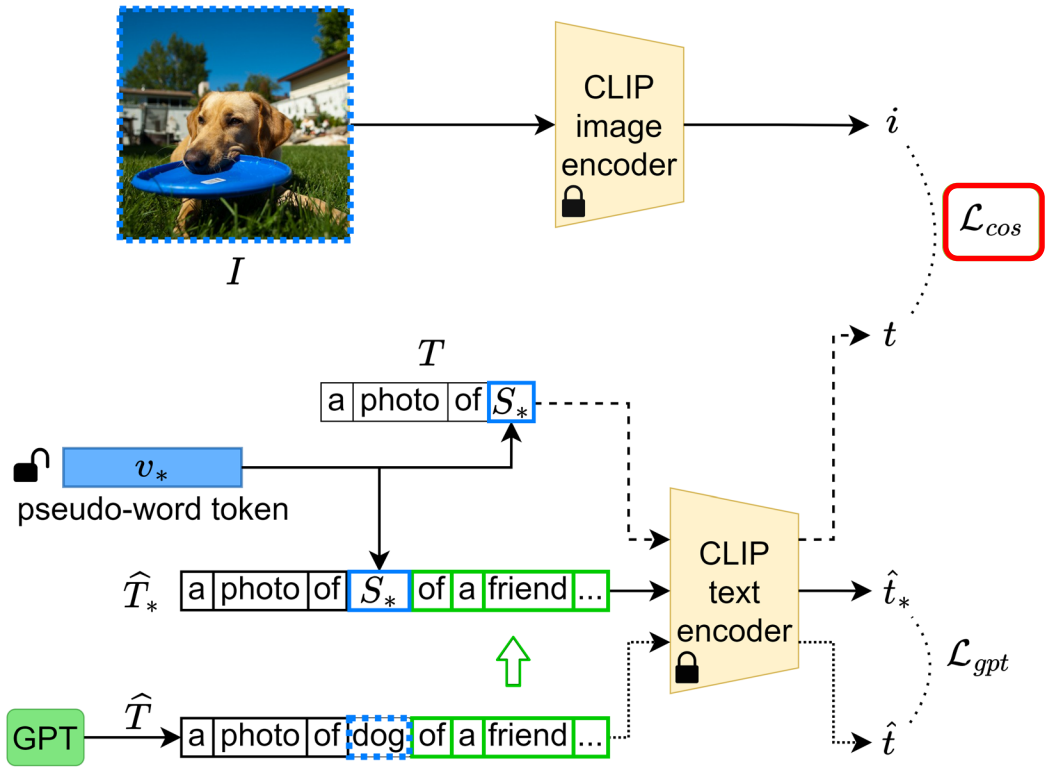
- Step 1: OTI approach to iteratively generate a pseudo-word token per image from an unlabeled dataset, enabling image retrieval using text-based queries.



- **Initialization**: A pseudo-word token **v∗** is randomly initialized. And associating the pseudo-word **S*** to it.
- **Template Creation**: A template sentence (*T*), like "**a photo of *S∗***", is created and fed to the CLIP text encoder to get text features.
- **CLIP Encoders**: The CLIP text encoder $\psi_T(T)$ and image encoder $\psi_I(I)$ are used to extract the features of template text **i.e.,** $t = \psi_T(T)$ and Image **i.e.,** $i = \psi_I(I)$.

SEARLE Training: OTI

- Step 1: OTI approach to iteratively generate a pseudo-word token per image from an unlabeled dataset, enabling image retrieval using text-based queries.

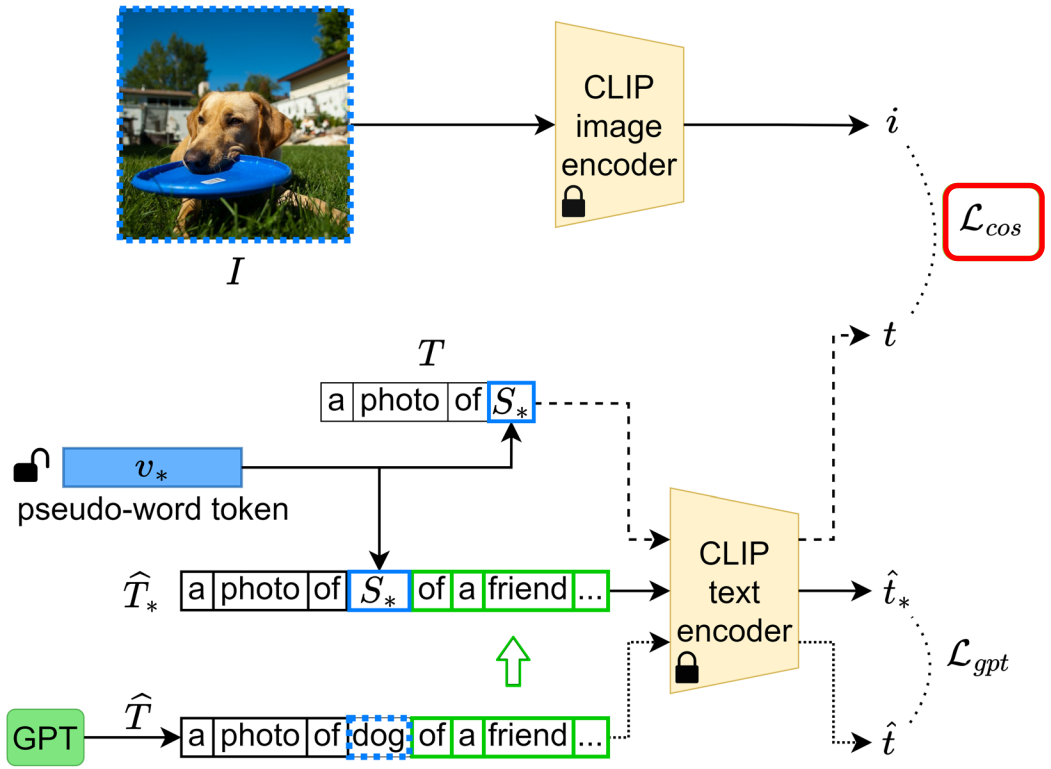


- **Initialization**: A pseudo-word token **v∗** is randomly initialized. And associating the pseudo-word **S*** to it.
- **Template Creation**: A template sentence (*T*), like "**a photo of *S∗***", is created and fed to the CLIP text encoder to get text features.
- **CLIP Encoders**: The CLIP text encoder $\psi_T(T)$ and image encoder $\psi_I(I)$ are used to extract the features of template text **i.e.,** $t = \psi_T(T)$ and Image **i.e.,** $i = \psi_I(I)$.
- **Cosine CLIP-based Loss (Lcos)**: Encapsulates the informative content of ***I*** into **v*** by closing the gap between the image and text features

$$\mathcal{L}_{cos} = 1 - \cos(i, t)$$

SEARLE Training: OTI

- Step 1: OTI approach to iteratively generate a pseudo-word token per image from an unlabeled dataset, enabling image retrieval using text-based queries.



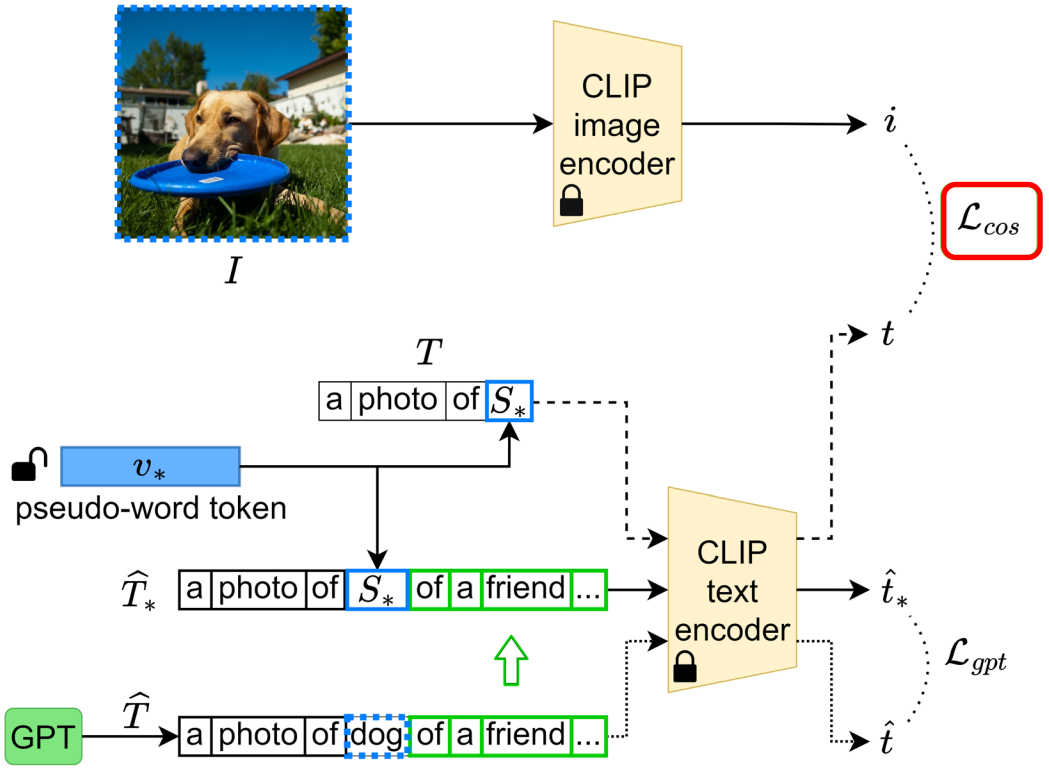
- **Initialization**: A pseudo-word token **v∗** is randomly initialized. And associating the pseudo-word **S*** to it.
- **Template Creation**: A template sentence (*T*), like "**a photo of *S∗***", is created and fed to the CLIP text encoder to get text features.
- **CLIP Encoders**: The CLIP text encoder $\psi_T(T)$ and image encoder $\psi_I(I)$ are used to extract the features of template text **i.e.,** $t = \psi_T(T)$ and Image **i.e.,** $i = \psi_I(I)$.
- **Cosine CLIP-based Loss (Lcos)**: Encapsulates the informative content of ***I*** into **v*** by closing the gap between the image and text features

$$\mathcal{L}_{cos} = 1 - \cos(i, t)$$

- **However, Lcos alone isn't enough**

SEARLE Training: OTI

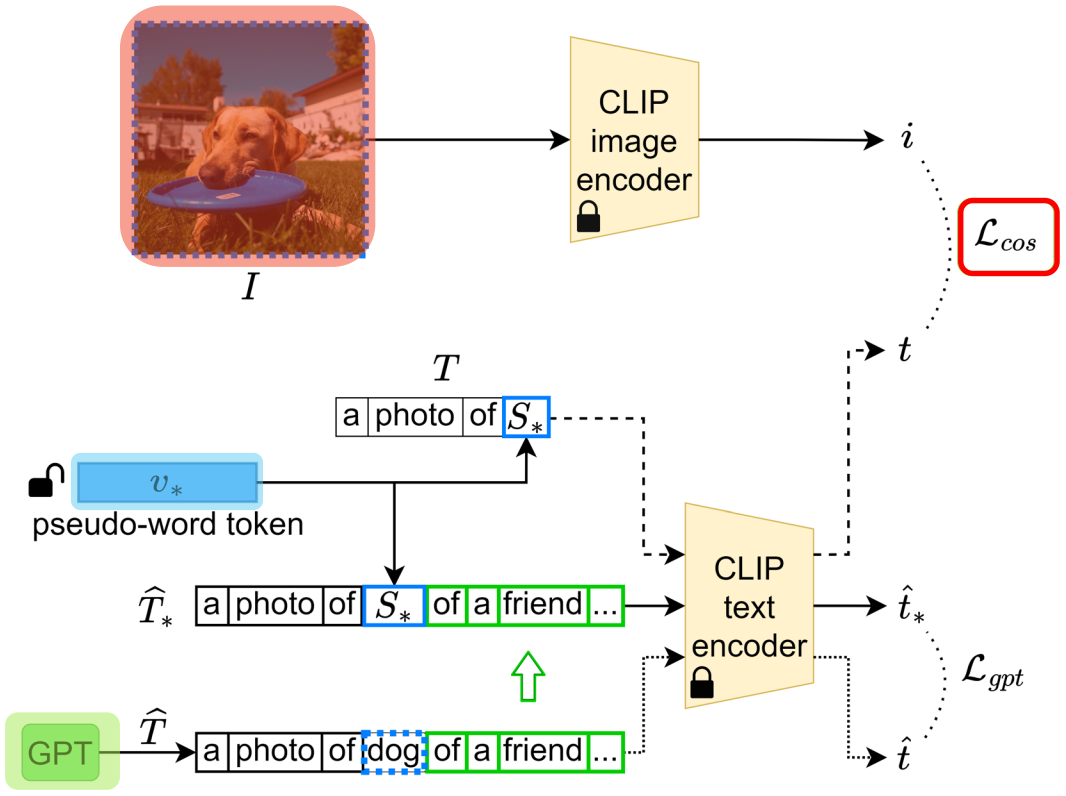
- Step 1: OTI approach to iteratively generate a pseudo-word token per image from an unlabeled dataset, enabling image retrieval using text-based queries.



- To handle L_{cos} only limitation, L_{gpt} regularization loss was introduced.

SEARLE Training: OTI

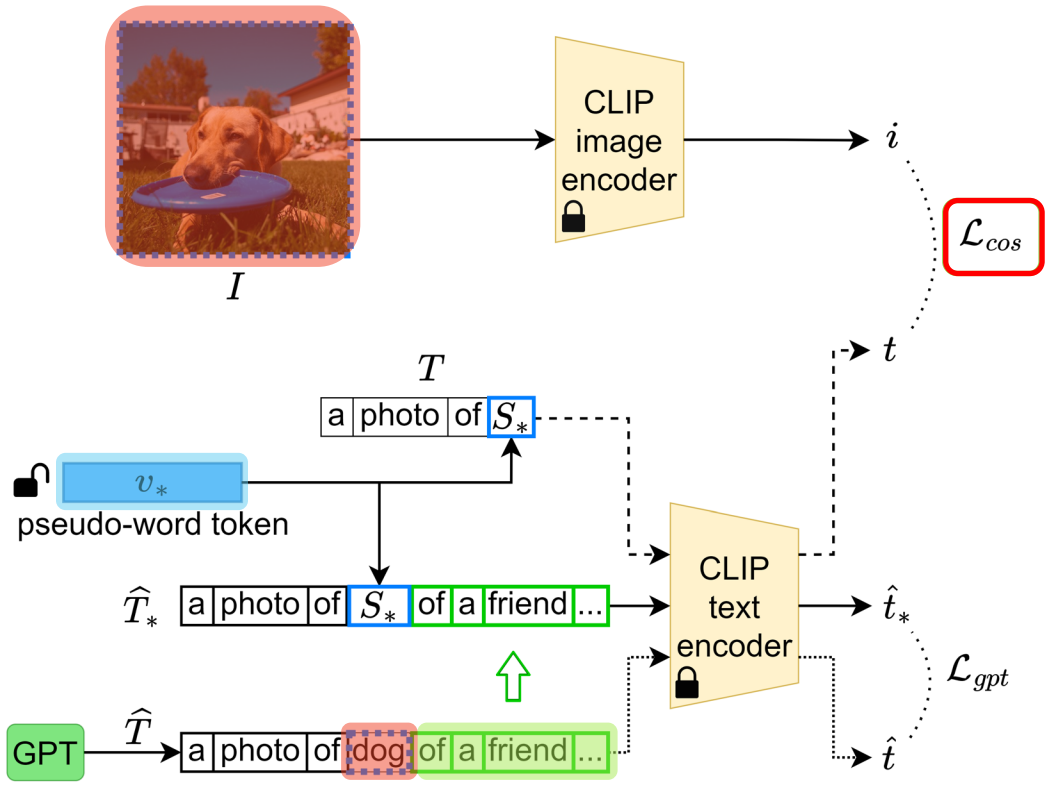
- Step 1: OTI approach to iteratively generate a pseudo-word token per image from an unlabeled dataset, enabling image retrieval using text-based queries.



- To handle L_{cos} only limitation, L_{gpt} regularization loss was introduced.
- **zero-shot classification:** Associate each image with a set of categories(concepts). These concepts serve as a starting point for generating textual representations that are contextually related to the image.

SEARLE Training: OTI

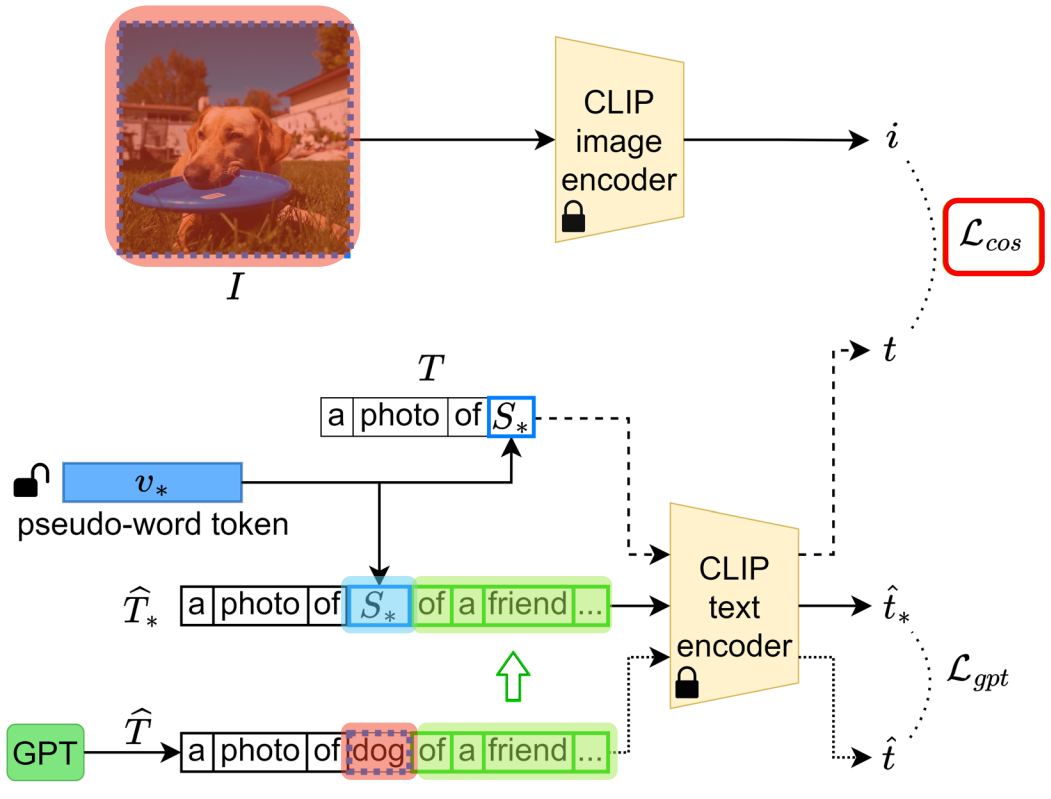
- Step 1: OTI approach to iteratively generate a pseudo-word token per image from an unlabeled dataset, enabling image retrieval using text-based queries.



- To handle L_{cos} only limitation, L_{gpt} regularization loss was introduced.
- **zero-shot classification:** Associate each image with a set of categories(concepts). These concepts serve as a starting point for generating textual representations that are contextually related to the image.
- **GPT-generated Phrases:** For each concept, a phrase **($\hat{T}$)** is generated using a GPT model.

SEARLE Training: OTI

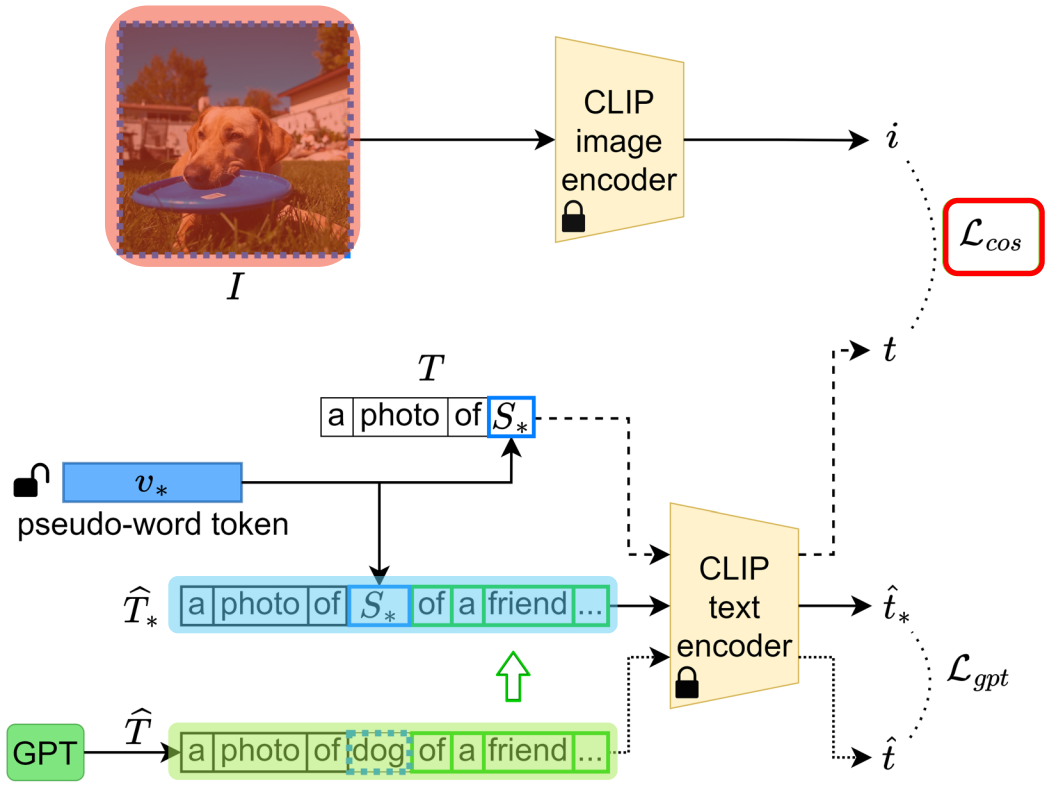
- Step 1: OTI approach to iteratively generate a pseudo-word token per image from an unlabeled dataset, enabling image retrieval using text-based queries.



- To handle L_{cos} only limitation, L_{gpt} regularization loss was introduced.
- **zero-shot classification:** Associate each image with a set of categories(concepts). These concepts serve as a starting point for generating textual representations that are contextually related to the image.
- **GPT-generated Phrases ($\hat{T}$):** For each concept, a phrase is generated using a GPT model.
- **Replacing Concept with Pseudo-word ($\hat{T}_*$):** In these phrases, the concept is replaced with the pseudo-word **(S_*)**

SEARLE Training: OTI

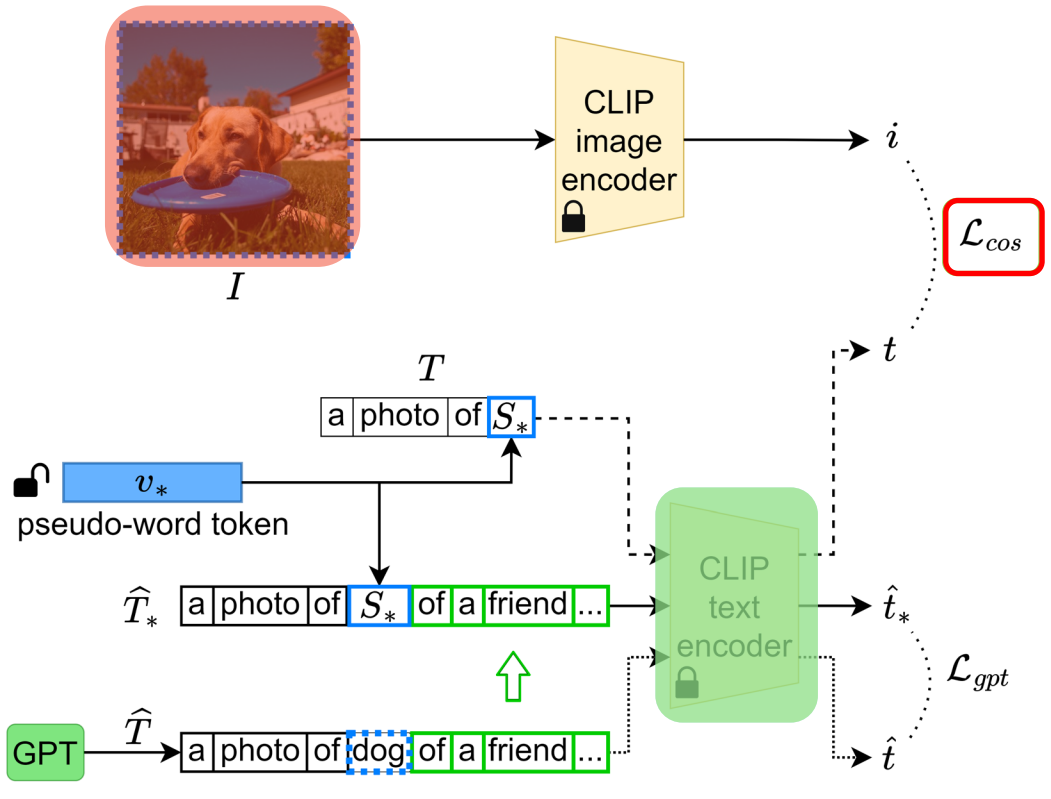
- Step 1: OTI approach to iteratively generate a pseudo-word token per image from an unlabeled dataset, enabling image retrieval using text-based queries.



- To handle L_{cos} only limitation, L_{gpt} regularization loss was introduced.
- **zero-shot classification:** Associate each image with a set of categories(concepts). These concepts serve as a starting point for generating textual representations that are contextually related to the image.
- **GPT-generated Phrases ($\hat{T}$):** For each concept, a phrase is generated using a GPT model.
- **Replacing Concept with Pseudo-word ($\hat{T}_*$):** In these phrases, the concept is replaced with the pseudo-word **(S_*)**

SEARLE Training: OTI

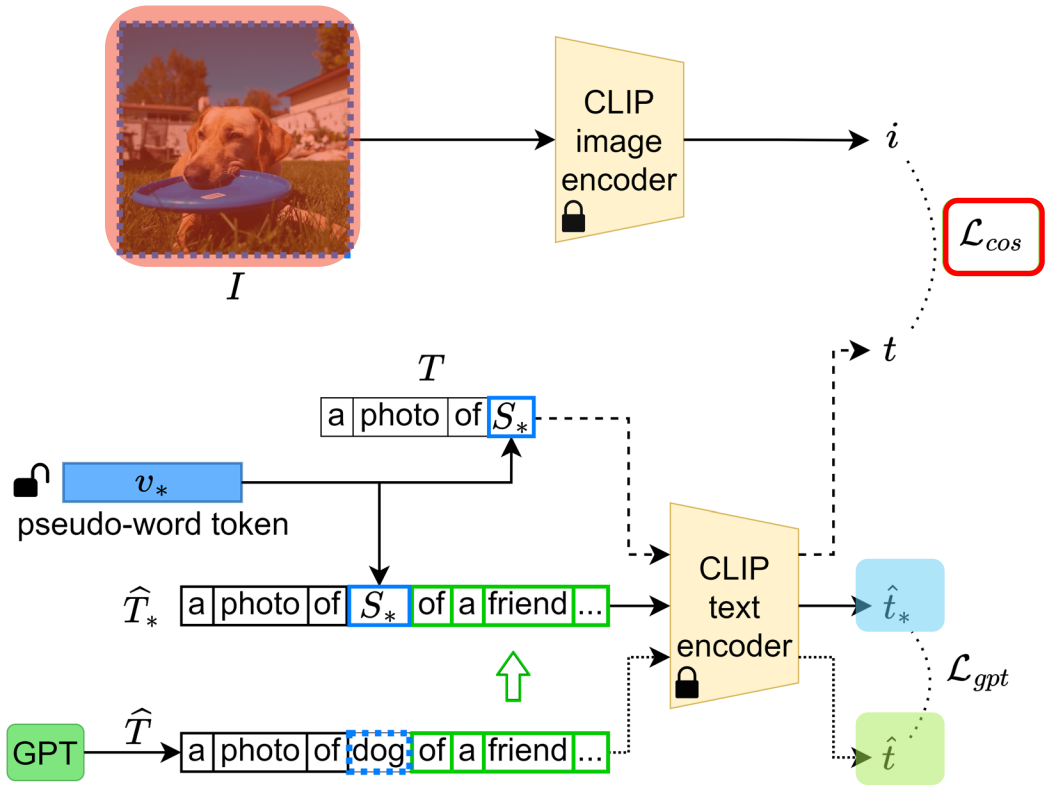
- Step 1: OTI approach to iteratively generate a pseudo-word token per image from an unlabeled dataset, enabling image retrieval using text-based queries.



- To handle L_{cos} only limitation, L_{gpt} regularization loss was introduced.
- **zero-shot classification:** Associate each image with a set of categories(concepts). These concepts serve as a starting point for generating textual representations that are contextually related to the image.
- **GPT-generated Phrases ($\hat{T}$):** For each concept, a phrase is generated using a GPT model.
- **Replacing Concept with Pseudo-word ($\hat{T}_*$):** In these phrases, the concept is replaced with the pseudo-word **(S_*)**
- The features of both $\hat{T}$ and $\hat{T}_*$ are then extracted using the CLIP text encoder.

SEARLE Training: OTI

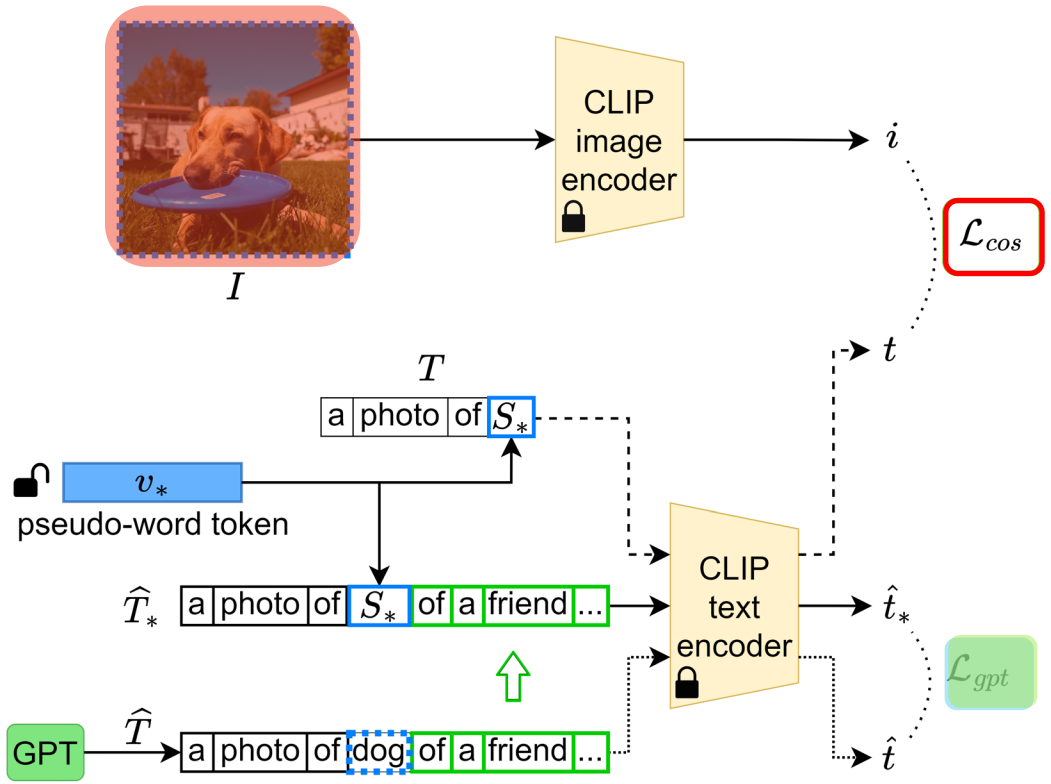
- Step 1: OTI approach to iteratively generate a pseudo-word token per image from an unlabeled dataset, enabling image retrieval using text-based queries.



- To handle L_{cos} only limitation, L_{gpt} regularization loss was introduced.
- **zero-shot classification:** Associate each image with a set of categories(concepts). These concepts serve as a starting point for generating textual representations that are contextually related to the image.
- **GPT-generated Phrases ($\hat{T}$):** For each concept, a phrase is generated using a GPT model.
- **Replacing Concept with Pseudo-word ($\hat{T}_*$):** In these phrases, the concept is replaced with the pseudo-word **(S_*)**
- The features of both $\hat{T}$ and $\hat{T}_*$ are then extracted using the CLIP text encoder.

SEARLE Training: OTI

- Step 1: OTI approach to iteratively generate a pseudo-word token per image from an unlabeled dataset, enabling image retrieval using text-based queries.



- To handle L_{cos} only limitation, L_{gpt} regularization loss was introduced.
- **zero-shot classification:** Associate each image with a set of categories(concepts). These concepts serve as a starting point for generating textual representations that are contextually related to the image.
- **GPT-generated Phrases ($\hat{T}$):** For each concept, a phrase is generated using a GPT model.
- **Replacing Concept with Pseudo-word ($\hat{T}_*$):** In these phrases, the concept is replaced with the pseudo-word **(S_*)**
- The features of both $\hat{T}$ and $\hat{T}_*$ are then extracted using the CLIP text encoder.
- **Contextualized Regularization (L_{gpt})**:

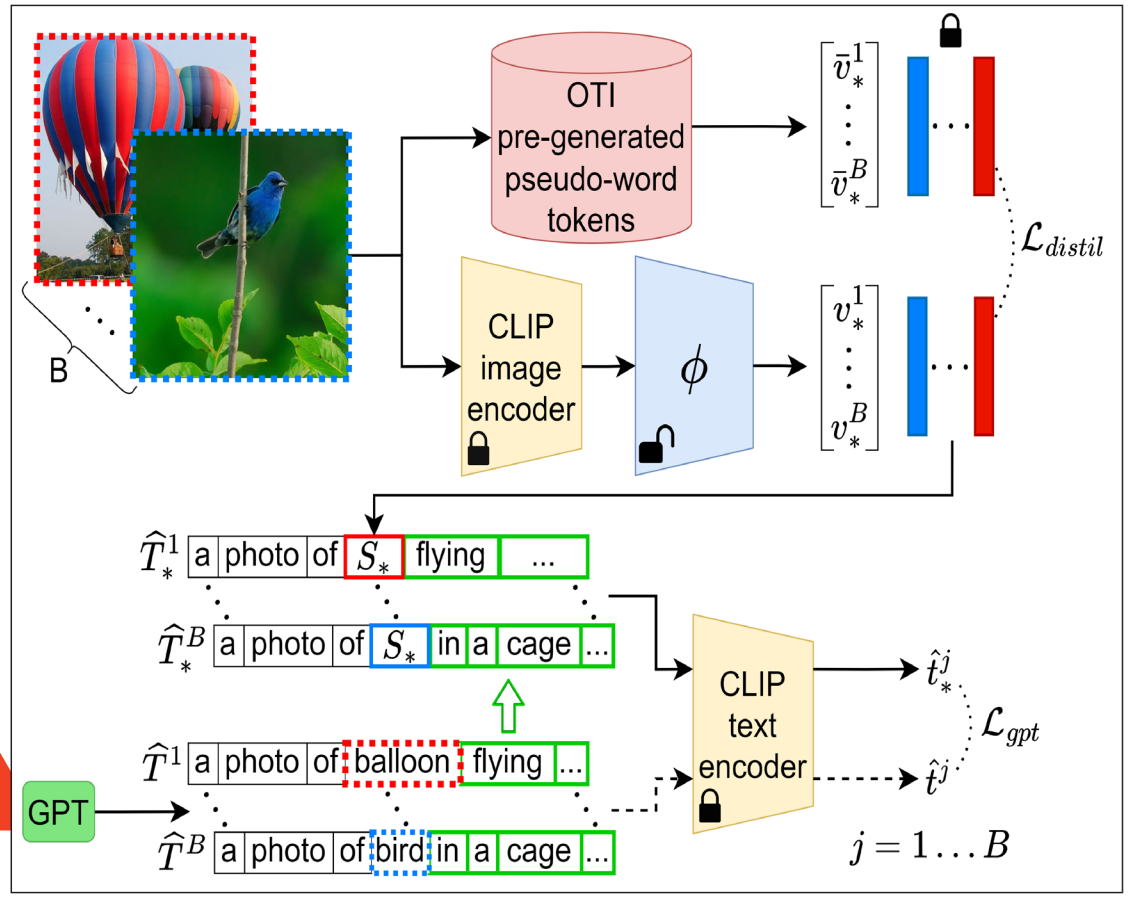
$$\mathcal{L}_{gpt} = 1 - \cos(\hat{t}, \hat{t}_*)$$

- The final loss function for OTI:

$$\mathcal{L}_{OTI} = \lambda_{cos}\mathcal{L}_{cos} + \lambda_{OTIgpt}\mathcal{L}_{gpt}$$

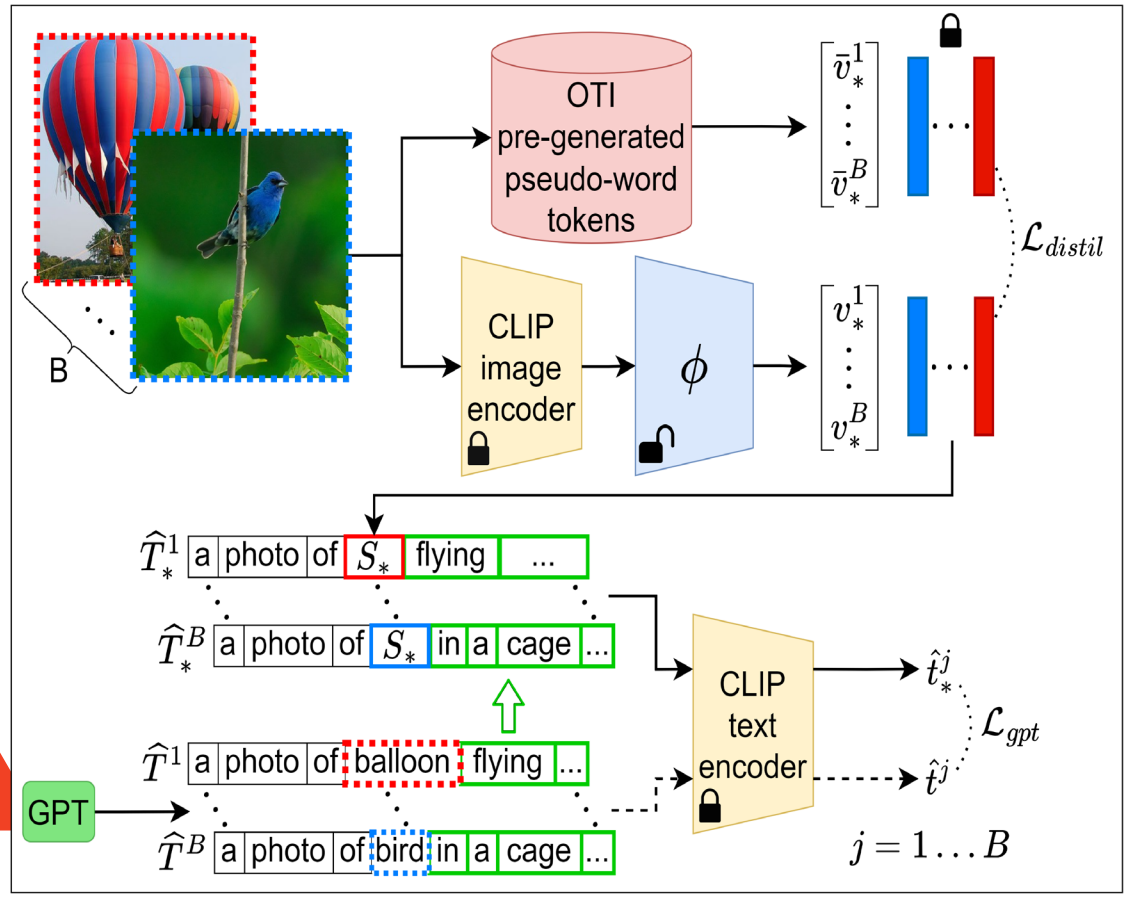
SEARLE: Textual Inversion Network (φ) Pre-training

- Step 2: OTI is effective in generating pseudo-words but requires a significant amount of time. φ address the time-consuming nature of the OTI method while retraining its power.



SEARLE: Textual Inversion Network (φ) Pre-training

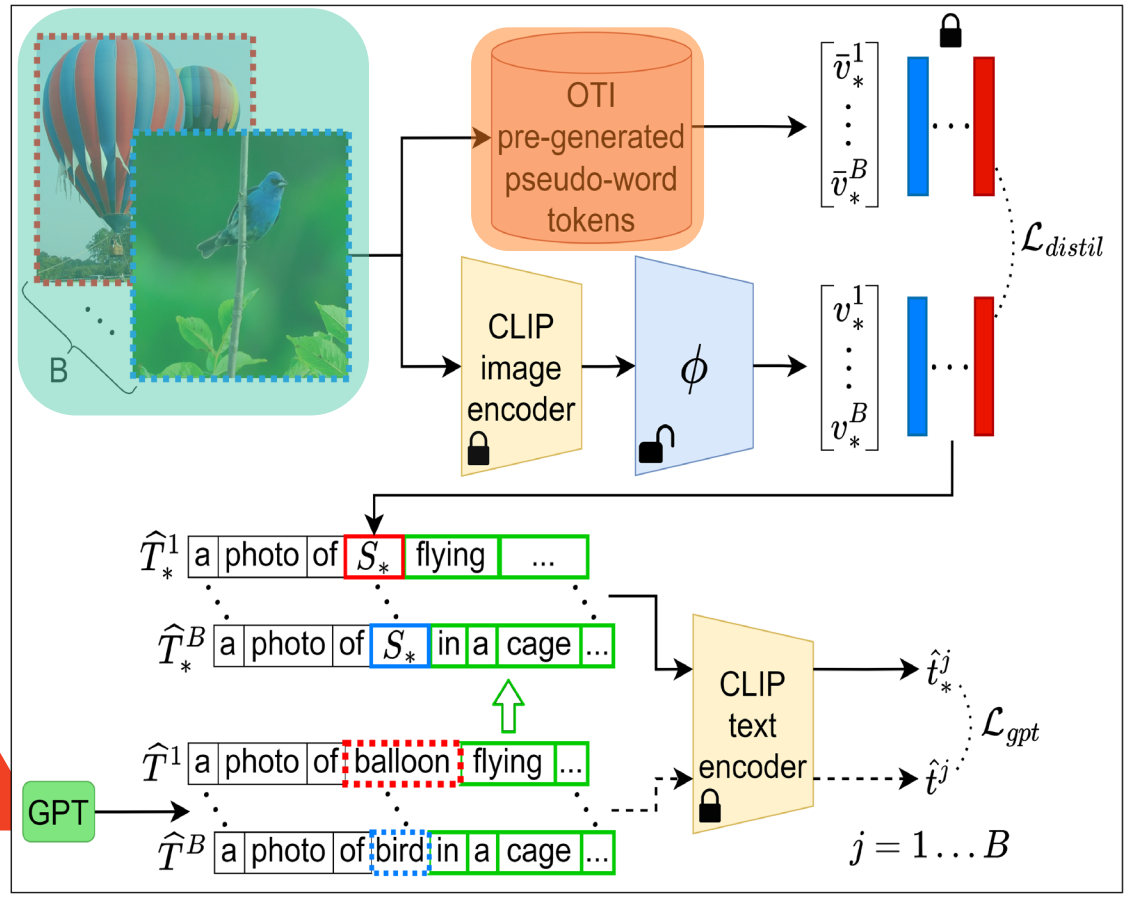
- Step 2: OTI is effective in generating pseudo-words but requires a significant amount of time. φ address the time-consuming nature of the OTI method while retraining its power.



- Φ learns to replicate the outcome of the OTI
- Φ is composed of three layers each followed by GELU and dropout layer.

SEARLE: Textual Inversion Network (φ) Pre-training

- Step 2: OTI is effective in generating pseudo-words but requires a significant amount of time. φ address the time-consuming nature of the OTI method while retraining its power.

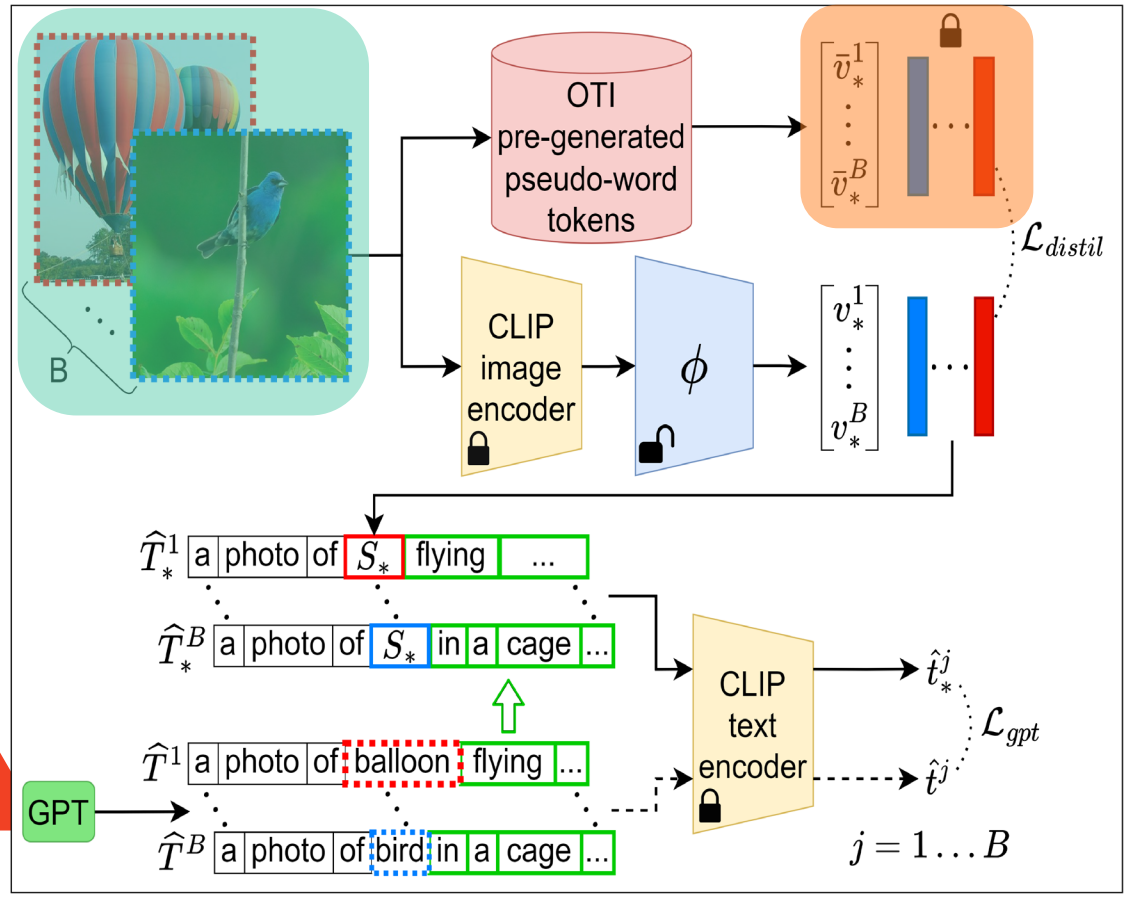


- Φ learns to replicate the outcome of the OTI
- Φ is composed of three layers each followed by GELU and dropout layer.
- Initially, OTI is applied to each image in an unlabeled dataset **D** and compute set of pseudo-word tokens $\overline{\boldsymbol{V}}_*$

$$\overline{\mathcal{V}}_* = \{\bar{v}_*^j\}_{j=1}^N$$

SEARLE: Textual Inversion Network (φ) Pre-training

- Step 2: OTI is effective in generating pseudo-words but requires a significant amount of time. φ address the time-consuming nature of the OTI method while retraining its power.

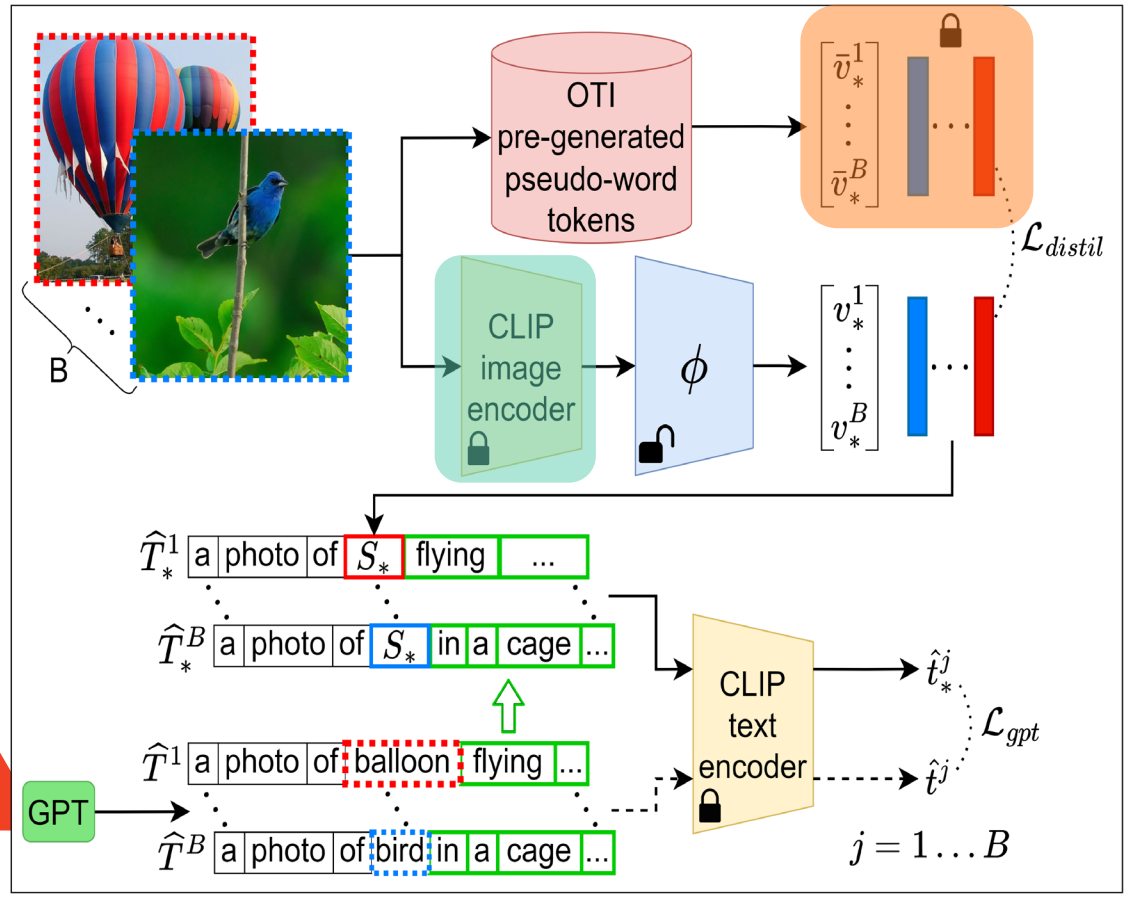


- Φ learns to replicate the outcome of the OTI
- Φ is composed of three layers each followed by GELU and dropout layer.
- Initially, OTI is applied to each image in an unlabeled dataset **D** and compute set of pseudo-word tokens $\overline{\boldsymbol{V}}_*$

$$\overline{\mathcal{V}}_* = \{\bar{v}_*^j\}_{j=1}^N$$

SEARLE: Textual Inversion Network (φ) Pre-training

- Step 2: OTI is effective in generating pseudo-words but requires a significant amount of time. φ address the time-consuming nature of the OTI method while retraining its power.



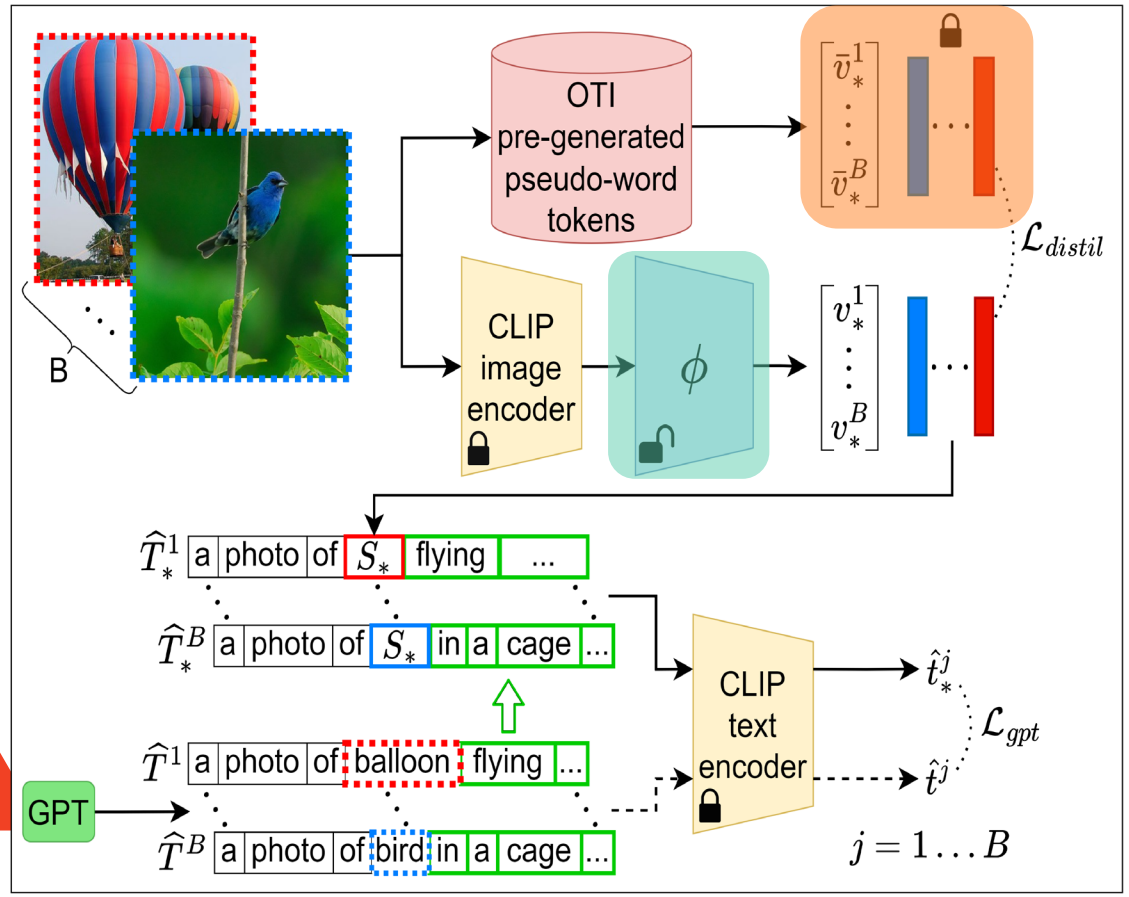
- Φ learns to replicate the outcome of the OTI
- Φ is composed of three layers each followed by GELU and dropout layer.
- Initially, OTI is applied to each image in an unlabeled dataset **D** and compute set of pseudo-word tokens $\overline{\boldsymbol{V}}_*$

$$\overline{\mathcal{V}}_* = \{\bar{v}_*^j\}_{j=1}^N$$

- Starting from an image $\boldsymbol{I} \in \mathbf{D}$, extract its features using the visual encoder obtaining $\boldsymbol{i} = \boldsymbol{\psi}_{\mathbf{I}}(\boldsymbol{I})$

SEARLE: Textual Inversion Network (φ) Pre-training

- Step 2: OTI is effective in generating pseudo-words but requires a significant amount of time. φ address the time-consuming nature of the OTI method while retraining its power.



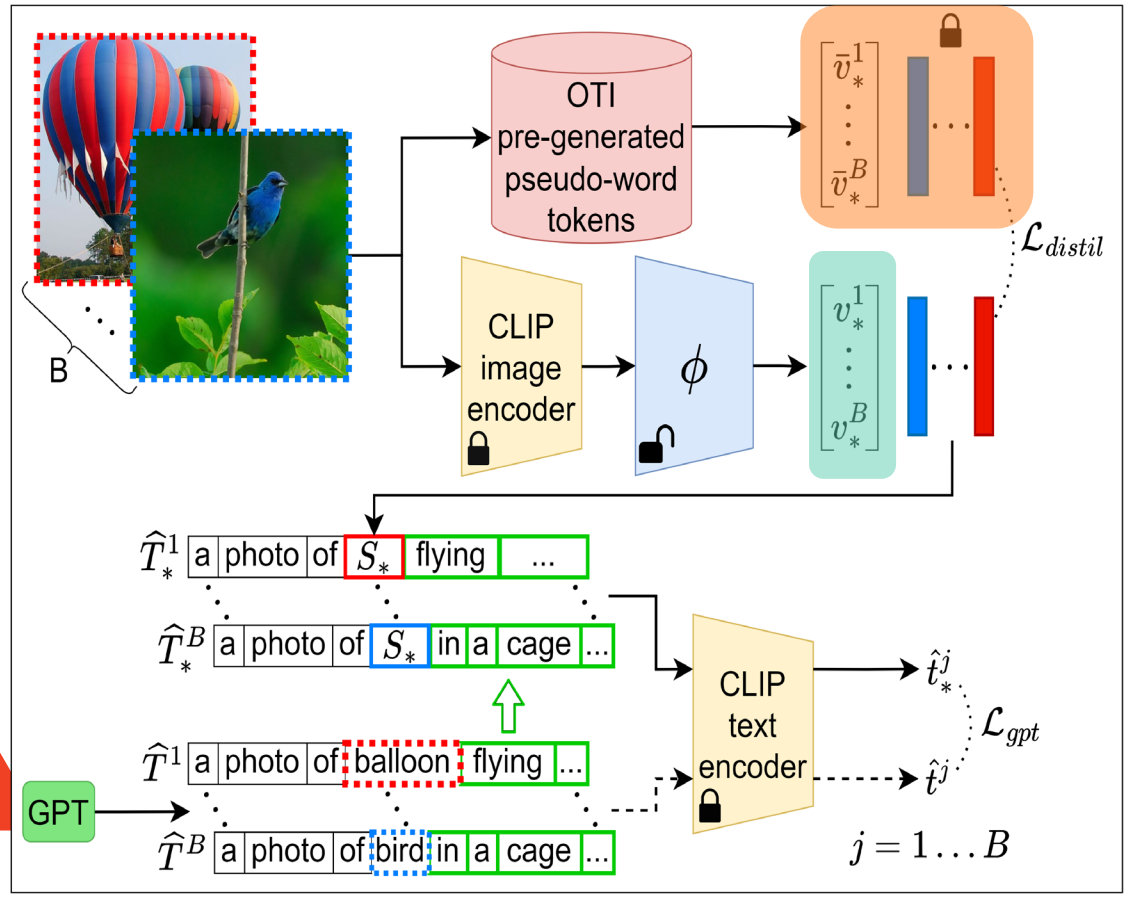
- Φ learns to replicate the outcome of the OTI
- Φ is composed of three layers each followed by GELU and dropout layer.
- Initially, OTI is applied to each image in an unlabeled dataset **D** and compute set of pseudo-word tokens $\overline{V}_*$

$$\overline{\mathcal{V}}_* = \{\bar{v}_*^j\}_{j=1}^N$$

- Starting from an image $I \in \mathbf{D}$, extract its features using the visual encoder obtaining $i = \Psi_I(I)$
- Φ to predict the pseudo-word token $V_* = \varphi(i)$

SEARLE: Textual Inversion Network (φ) Pre-training

- Step 2: OTI is effective in generating pseudo-words but requires a significant amount of time. φ address the time-consuming nature of the OTI method while retraining its power.



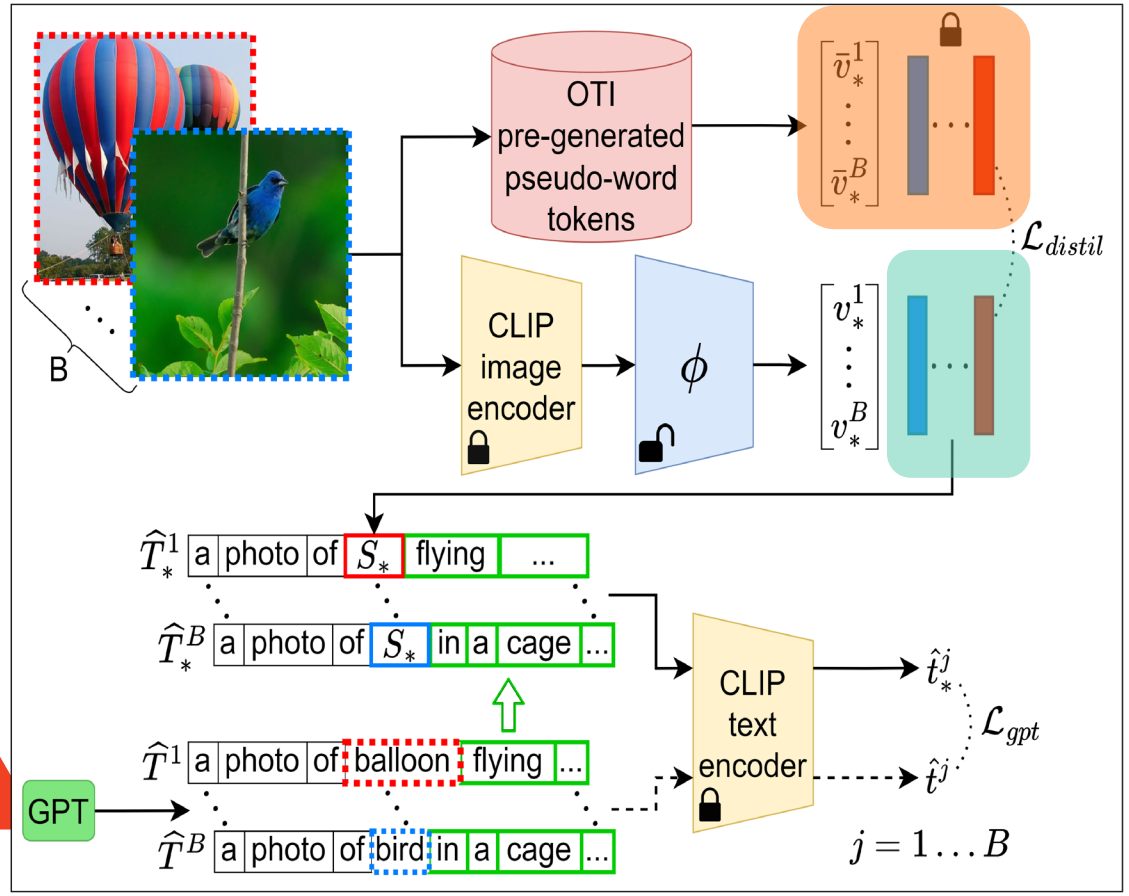
- Φ learns to replicate the outcome of the OTI
- Φ is composed of three layers each followed by GELU and dropout layer.
- Initially, OTI is applied to each image in an unlabeled dataset **D** and compute set of pseudo-word tokens $\overline{V}_*$

$$\overline{\mathcal{V}}_* = \{\bar{v}_*^j\}_{j=1}^N$$

- Starting from an image $I \in \mathbf{D}$, extract its features using the visual encoder obtaining $i = \Psi_I(I)$
- Φ to predict the pseudo-word token $V_* = \varphi(i)$

SEARLE: Textual Inversion Network (φ) Pre-training

- Step 2: OTI is effective in generating pseudo-words but requires a significant amount of time. φ address the time-consuming nature of the OTI method while retraining its power.



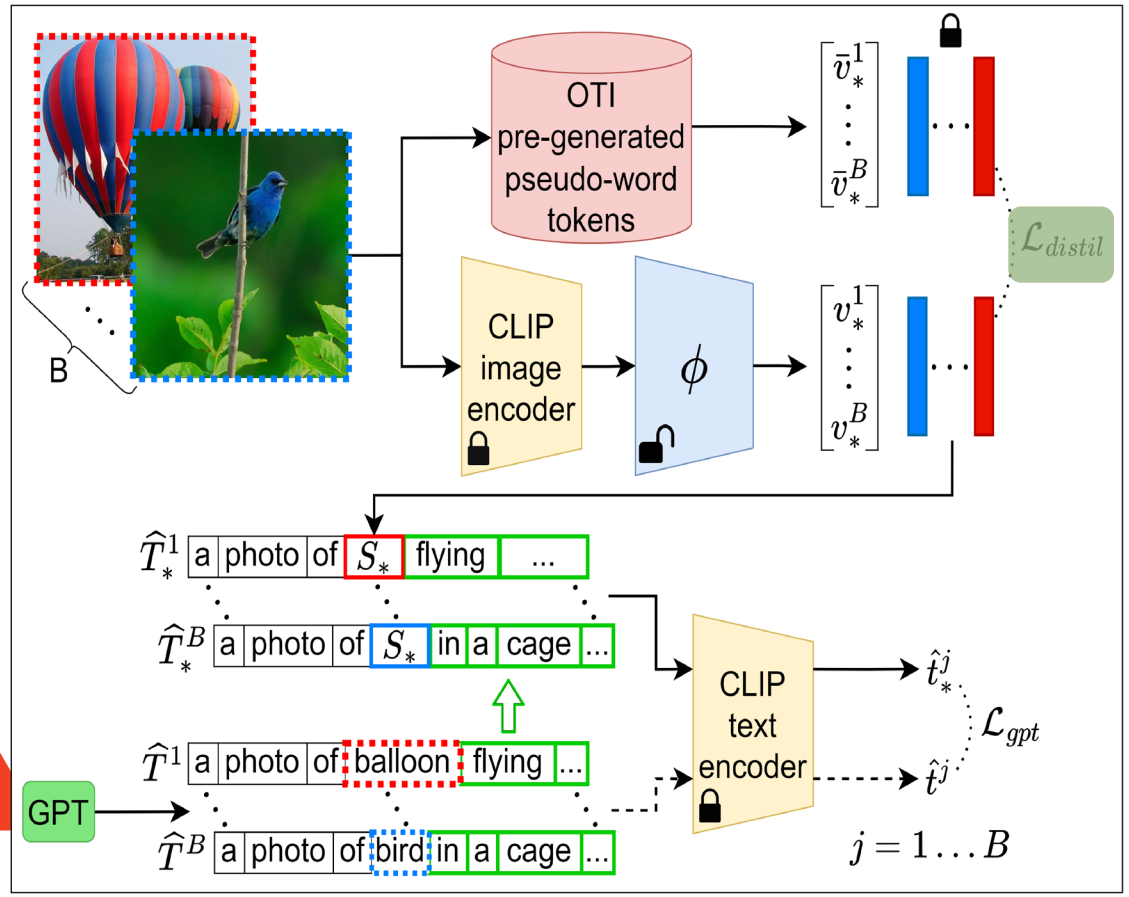
- Φ learns to replicate the outcome of the OTI
- Φ is composed of three layers each followed by GELU and dropout layer.
- Initially, OTI is applied to each image in an unlabeled dataset **D** and compute set of pseudo-word tokens $\overline{V}_*$

$$\overline{\mathcal{V}}_* = \{\bar{v}_*^j\}_{j=1}^N$$

- Starting from an image $I \in \mathbf{D}$, extract its features using the visual encoder obtaining $i = \Psi_I(I)$
- Φ to predict the pseudo-word token $V_* = \varphi(i)$
- Like OTI, same (L_{gpt}) to regularize Φ training.

SEARLE: Textual Inversion Network (φ) Pre-training

- Step 2: OTI is effective in generating pseudo-words but requires a significant amount of time. φ address the time-consuming nature of the OTI method while retraining its power.



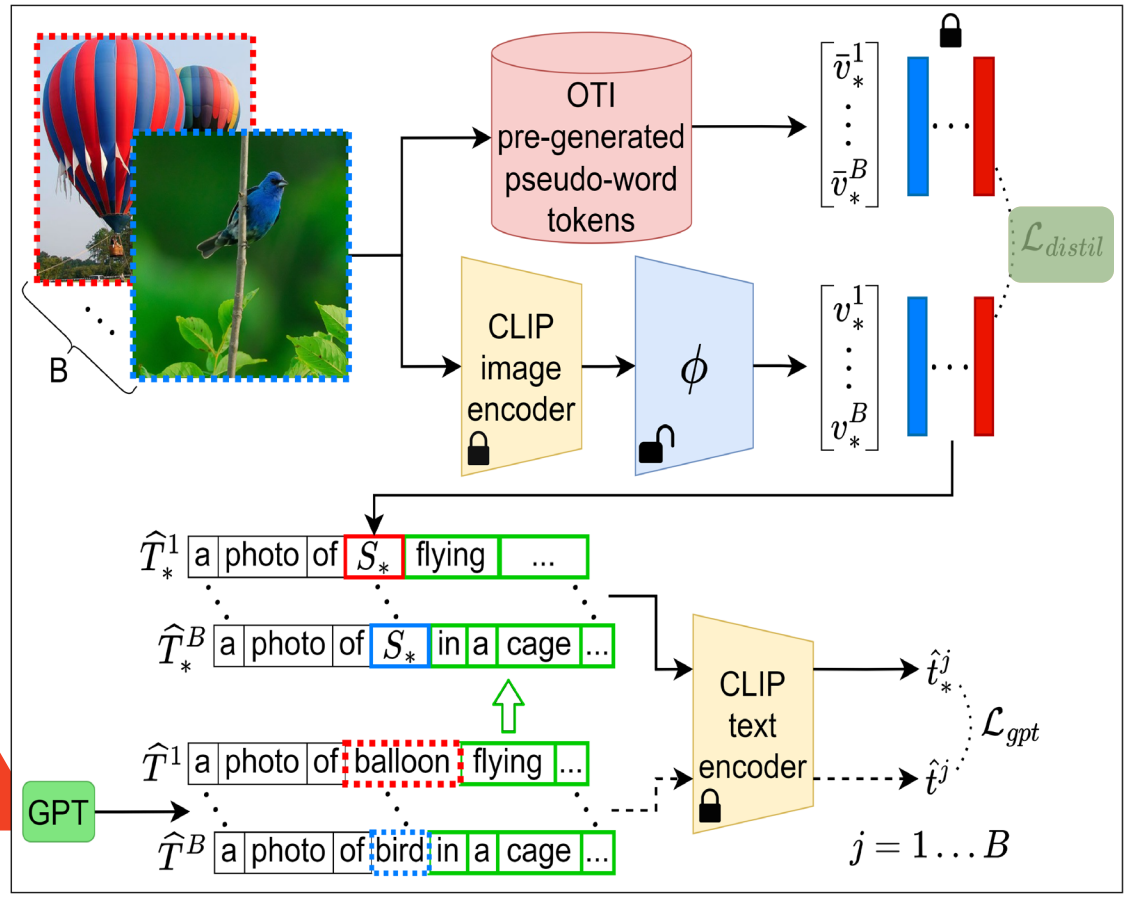
- Φ learns to replicate the outcome of the OTI
- Φ is composed of three layers each followed by GELU and dropout layer.
- Initially, OTI is applied to each image in an unlabeled dataset **D** and compute set of pseudo-word tokens $\overline{V}_*$

$$\overline{\mathcal{V}}_* = \{\bar{v}_*^j\}_{j=1}^N$$

- Starting from an image $I \in \mathbf{D}$, extract its features using the visual encoder obtaining $i = \psi_I(I)$
- Φ to predict the pseudo-word token $V_* = \varphi(i)$
- Like OTI, same GPT-powered (L_{gpt}) to regularize Φ training.
- Minimize the distance between the predicted pseudo-word token V_* and its corresponding pre-generated token $\overline{V}_* \in V_*$ using (L_{distil})

SEARLE: Textual Inversion Network (φ) Pre-training

- Step 2: OTI is effective in generating pseudo-words but requires a significant amount of time. φ address the time-consuming nature of the OTI method while retraining its power.



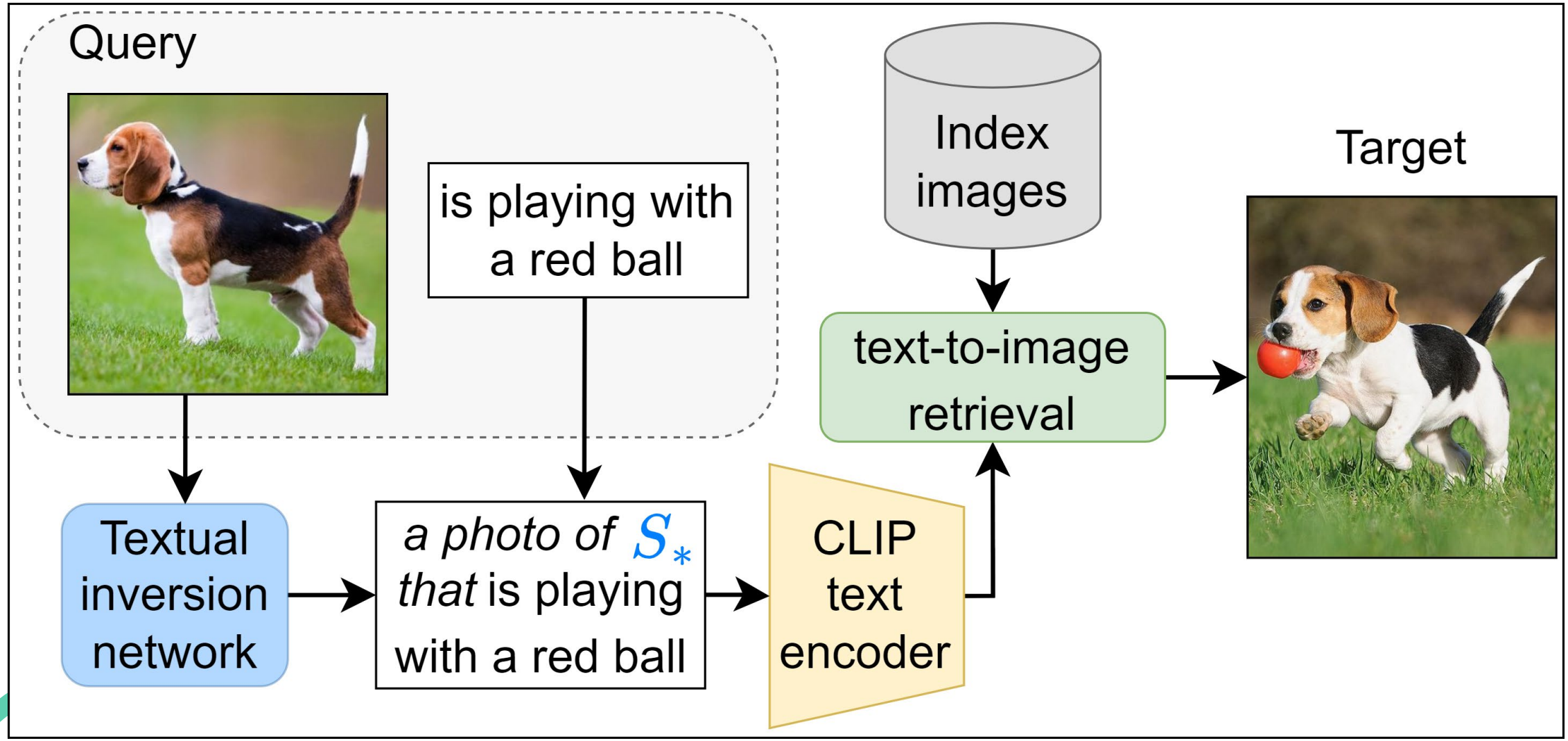
- Φ learns to replicate the outcome of the OTI
- Φ is composed of three layers each followed by GELU and dropout layer.
- Initially, OTI is applied to each image in an unlabeled dataset **D** and compute set of pseudo-word tokens $\overline{V}_*$

$$\overline{\mathcal{V}}_* = \{\bar{v}_*^j\}_{j=1}^N$$

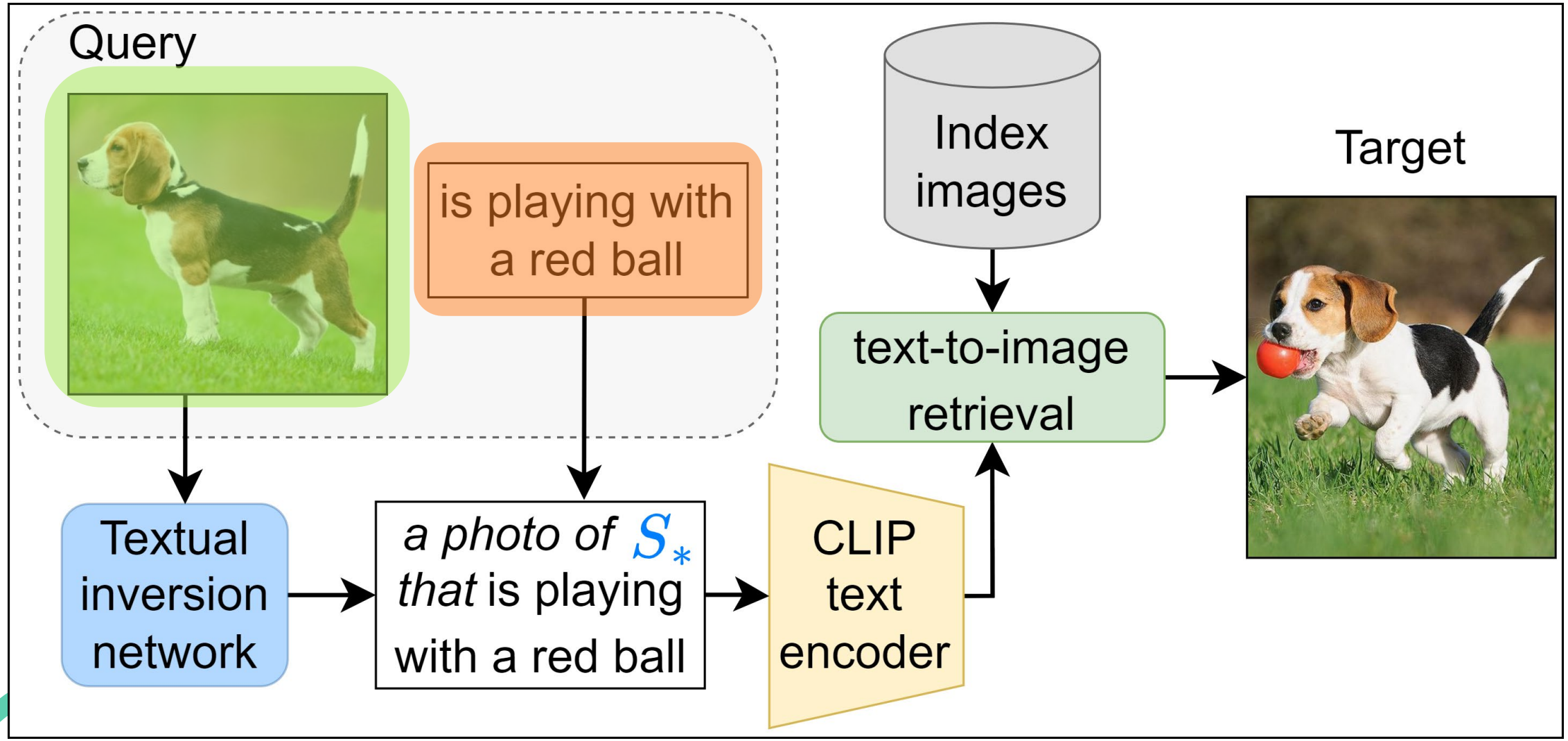
- Starting from an image $I \in \mathbf{D}$, extract its features using the visual encoder obtaining $i = \Psi_I(I)$
- Φ to predict the pseudo-word token $V_* = \varphi(i)$
- Like OTI, same GPT-powered (L_{gpt}) to regularize Φ training.
- Minimize the distance between the predicted pseudo-word token V_* and its corresponding pre-generated token $\overline{V}_* \in V_*$ using (L_{distil})
- Final Loss in Φ:

$$\mathcal{L}_\phi = \lambda_{distil}\mathcal{L}_{distil} + \lambda_{\phi gpt}\mathcal{L}_{gpt}$$

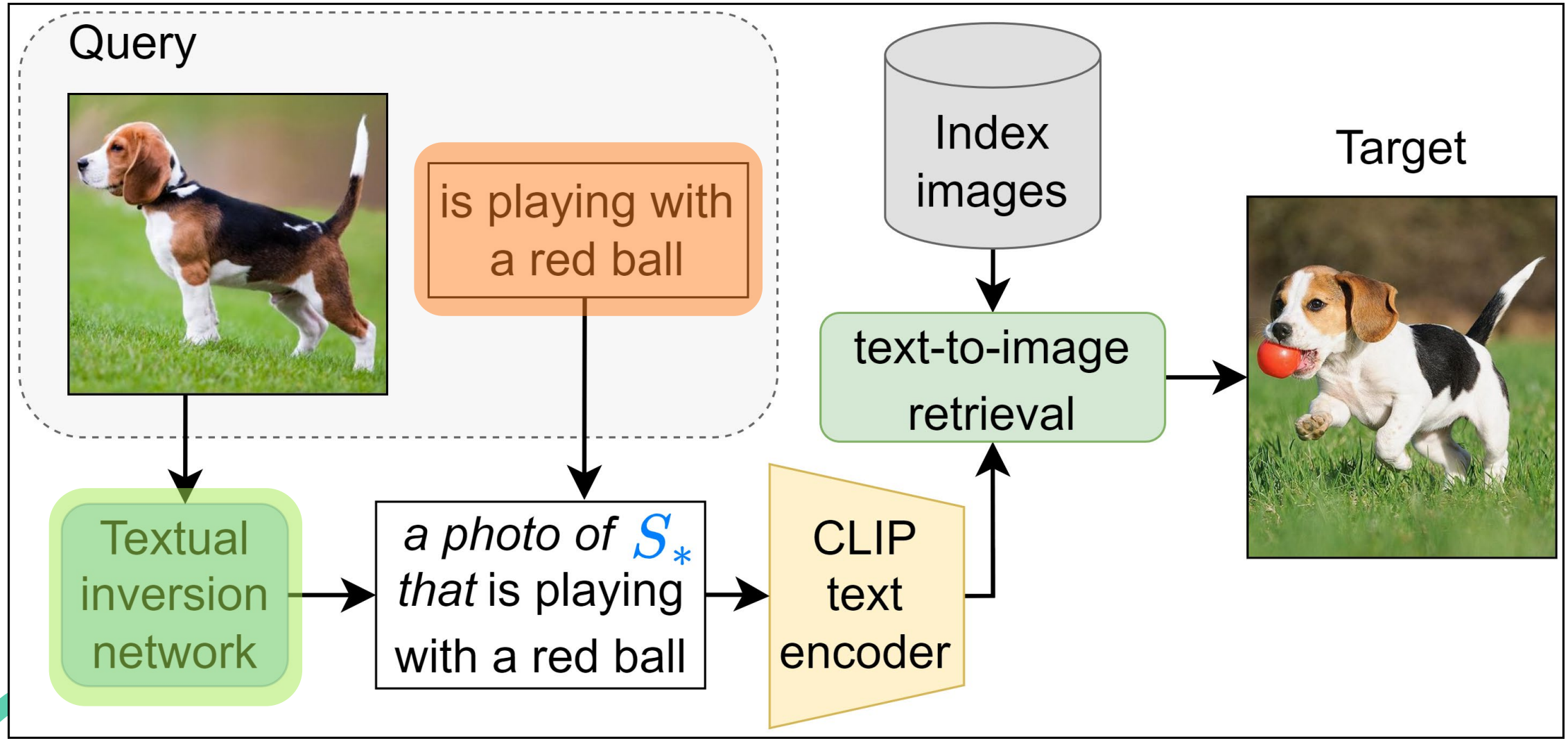
SEARLE: Inference Time



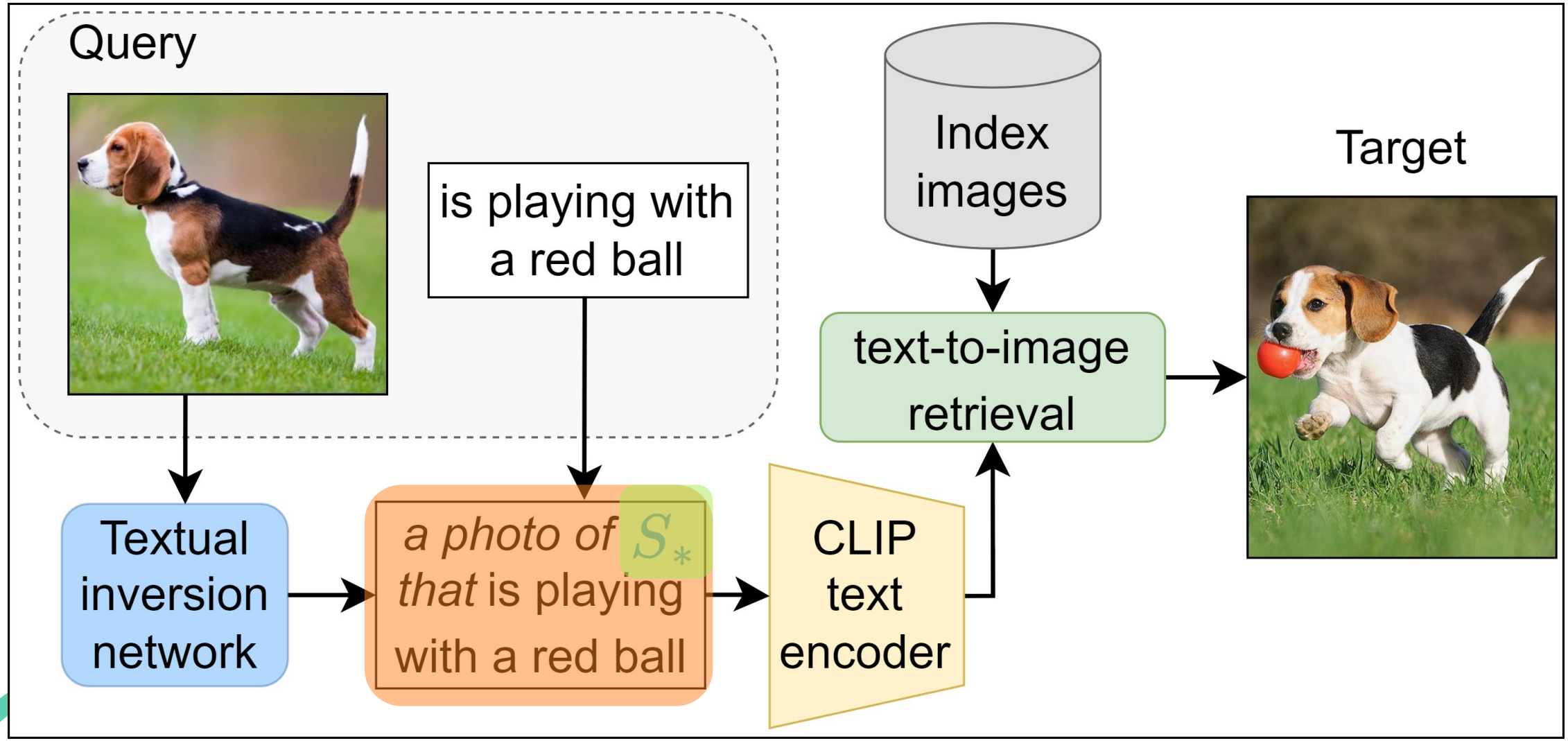
SEARLE: Inference Time



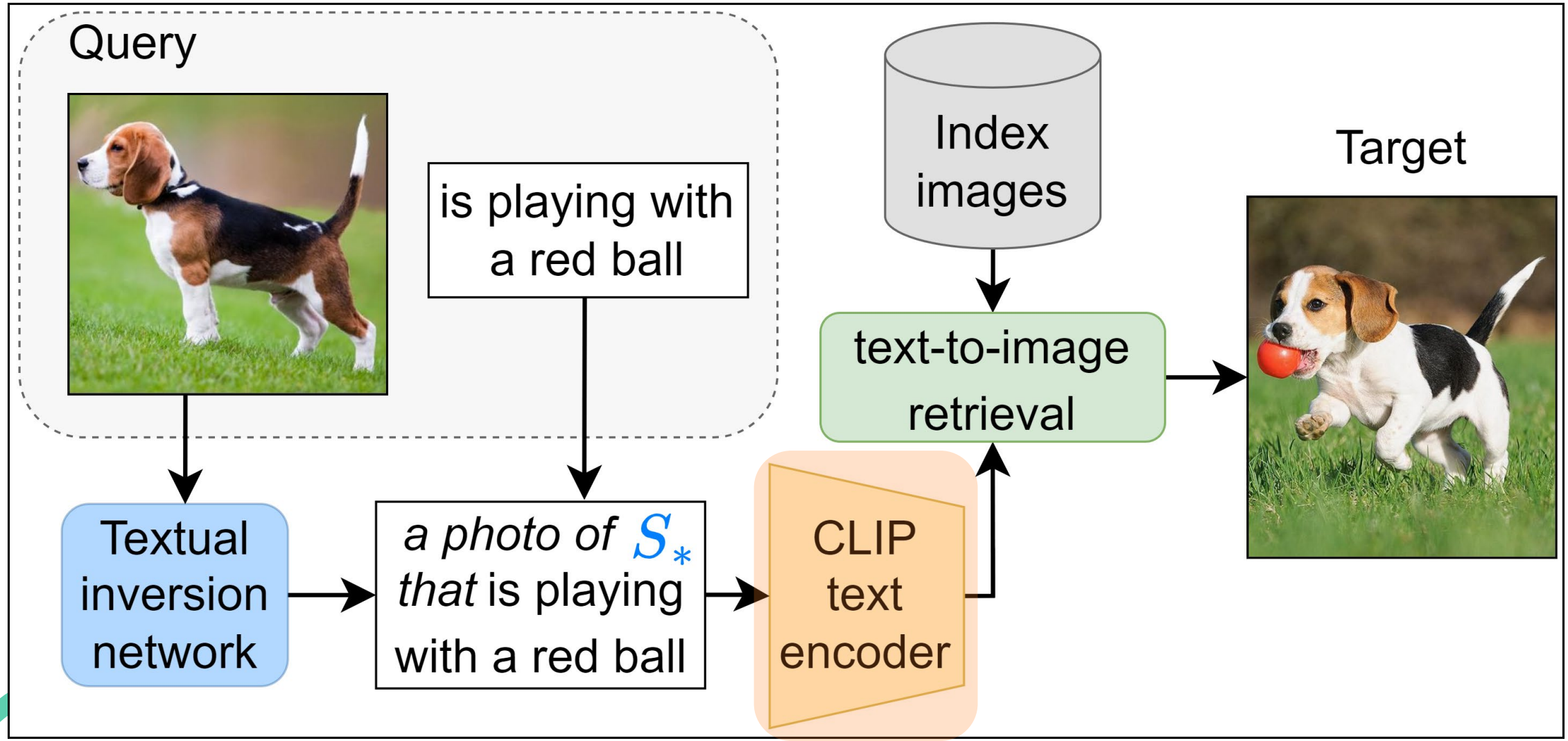
SEARLE: Inference Time



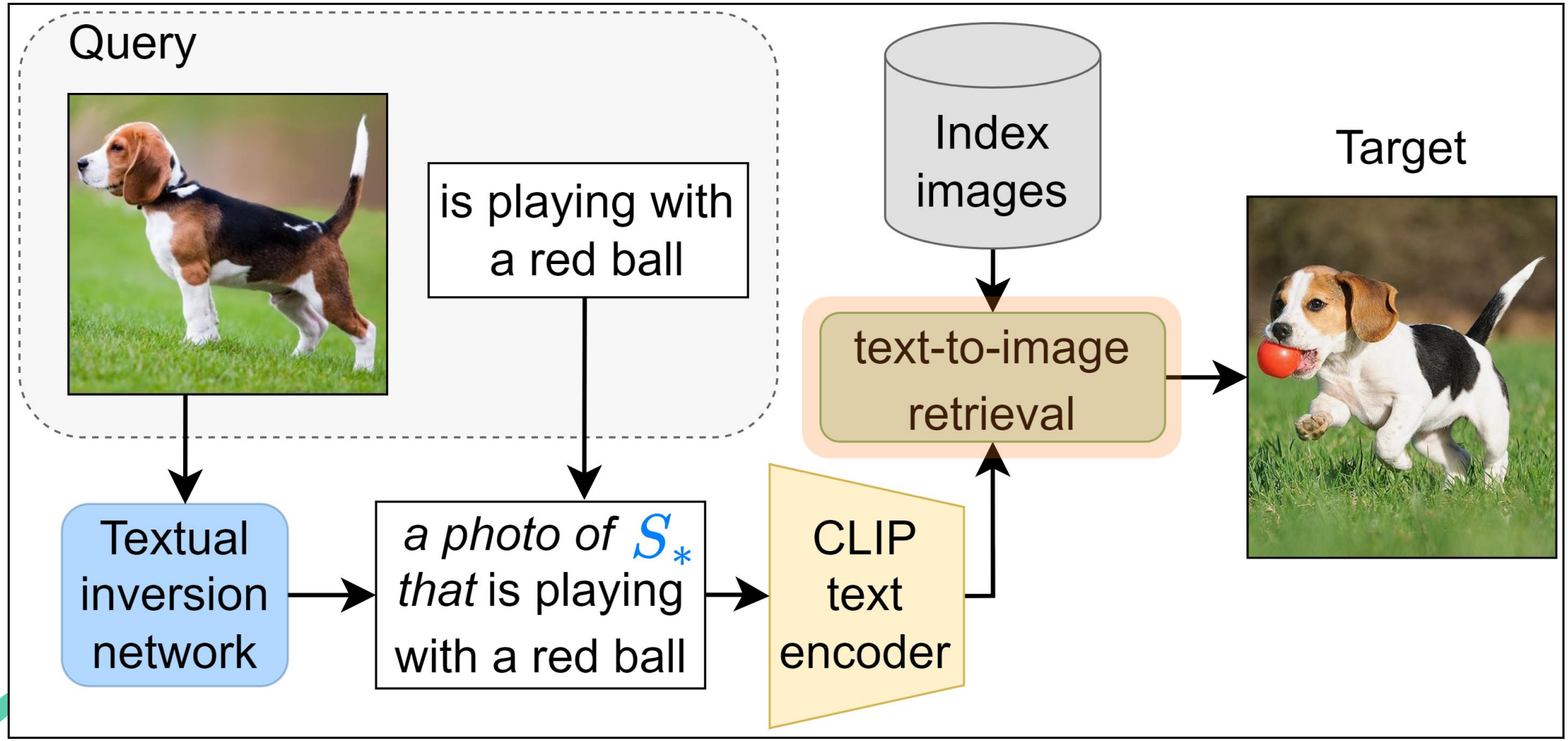
SEARLE: Inference Time



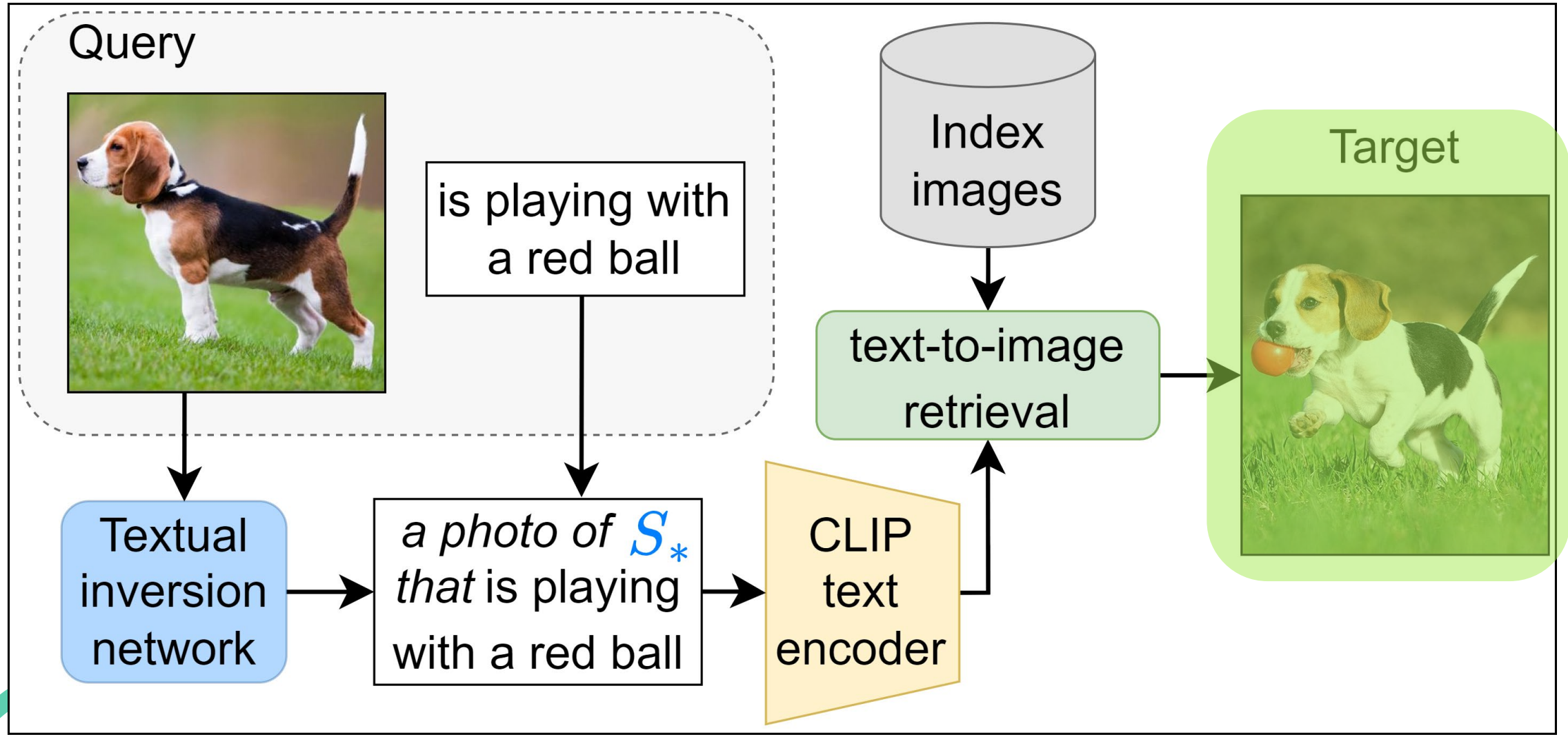
SEARLE: Inference Time



SEARLE: Inference Time

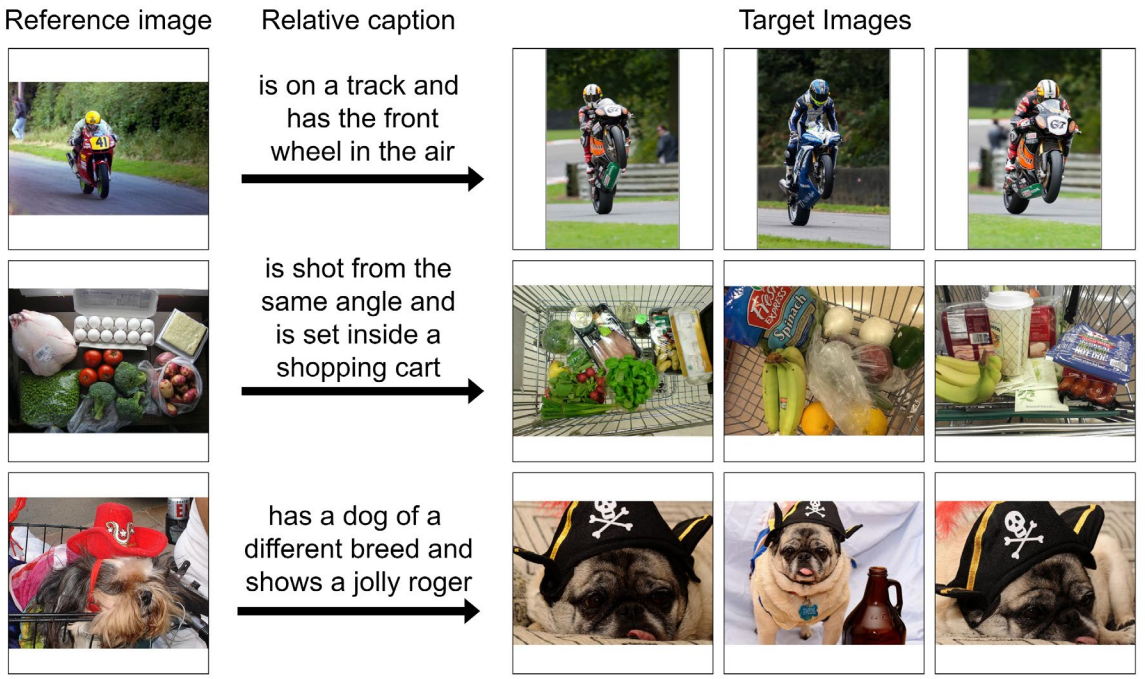


SEARLE: Inference Time



CIRCO Dataset

- Domain Specific CIR datasets: **FashionIQ**, **birds**, or **synthetic objects**
- Open domain Dataset: CIRR with natural images
  - Have false negatives, lead inaccurate evaluation
  - Do not consider visual content of reference image
  - Have only one annotated ground truth
- To support CIR research they introduce an open-domain dataset named **CIRCO**
  - It is based on real world images from COCO 2017 unlabeled set.
- First CIR dataset with multiple annotated ground truths



Experiments & Results

- **Backbones**
  - **SEARLE**: based on CLIP ViT-B/32
  - **SEARLE-XL**: using CLIP ViT-L/14

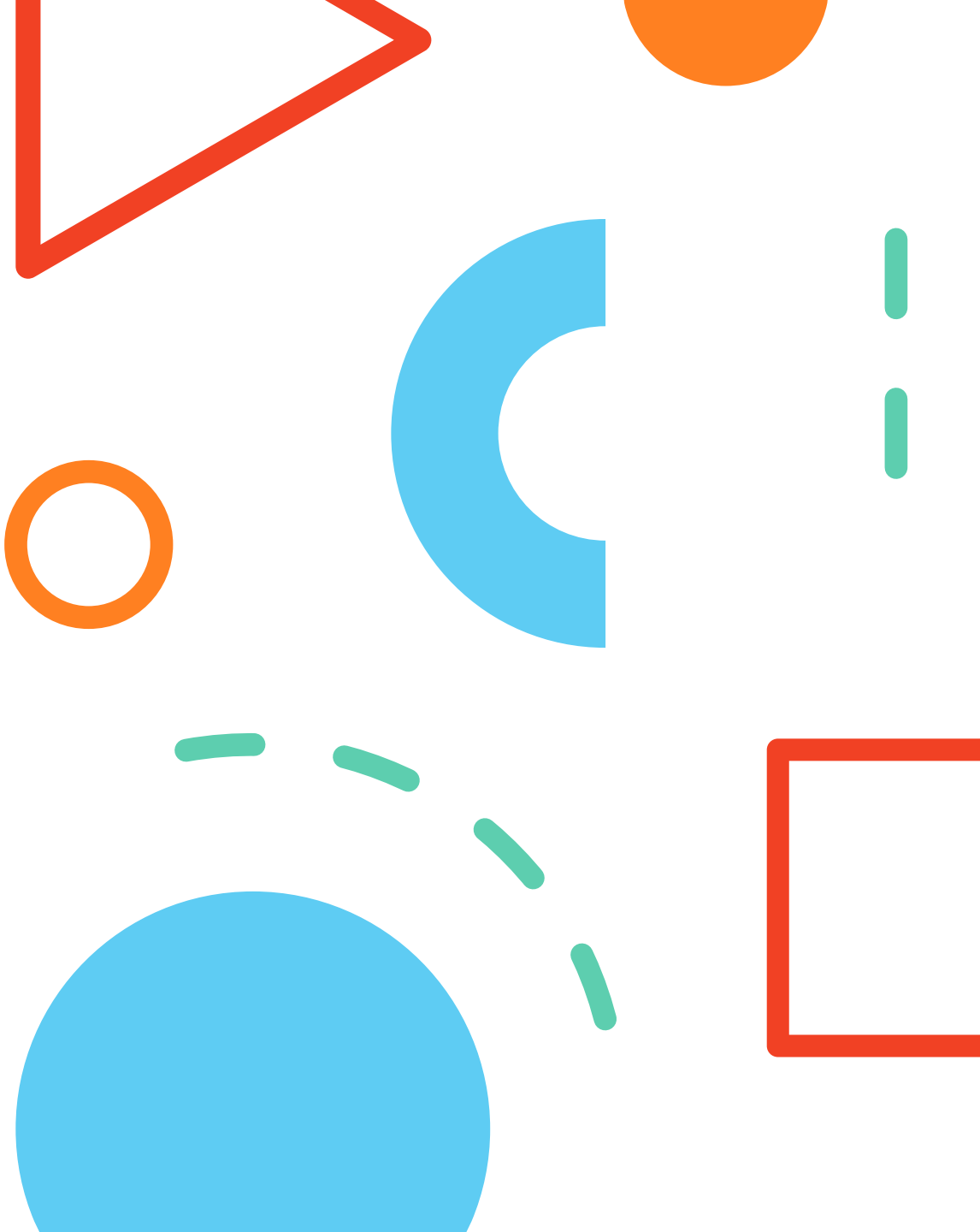
Backbone	Method	Shirt		Dress		Toptee		Average	
		R@10	R@50	R@10	R@50	R@10	R@50	R@10	R@50
B/32	Image-only	6.92	14.23	4.46	12.19	6.32	13.77	5.90	13.37
	Text-only	19.87	34.99	15.42	35.05	20.81	40.49	18.70	36.84
	Image + Text	13.44	26.25	13.83	30.88	17.08	31.67	14.78	29.60
	Captioning	17.47	30.96	9.02	23.65	15.45	31.26	13.98	28.62
	PALAVRA [8]	21.49	37.05	17.25	35.94	20.55	38.76	19.76	37.25
	SEARLE-OTI	**25.37**	41.32	17.85	**39.91**	24.12	45.79	22.44	42.34
	SEARLE	24.44	**41.61**	**18.54**	39.51	**25.70**	**46.46**	**22.89**	**42.53**
L/14	Pic2Word† [33]	26.20	43.60	20.00	40.20	27.90	47.40	24.70	43.70
	SEARLE-XL-OTI	**30.37**	**47.49**	**21.57**	**44.47**	**30.90**	**51.76**	**27.61**	**47.90**
	SEARLE-XL	26.89	45.58	20.48	43.13	29.32	49.97	25.56	46.23

SEARLE - Limitations & Future Work

- **Dependence of φ's Performance on OTI**
- **Non-Human-Understandable Text Embeddings:**
  - Pic2Word, in a zero-shot setting, reducing dependence on templates.
- **Rigid Template Filling & Template Dependence**
- **Limitation in Disentangling Multiple Objects and Attributes:** tokens are limited & small length, limits them in handling cases where multiple objects are involved and required complex changes
- **Generalizability:** While the methodology shows promising results, but there were performance gap due to Domain Specificity of FashionIQ.

Improvements

1. Reduce the dependency of φ's performance on OTI?
2. Interpretable and human-understandable text embeddings?
3. Handling multiple objects or remove irrelevant objects in CIR?
4. Overcome the reliance on rigid templates?



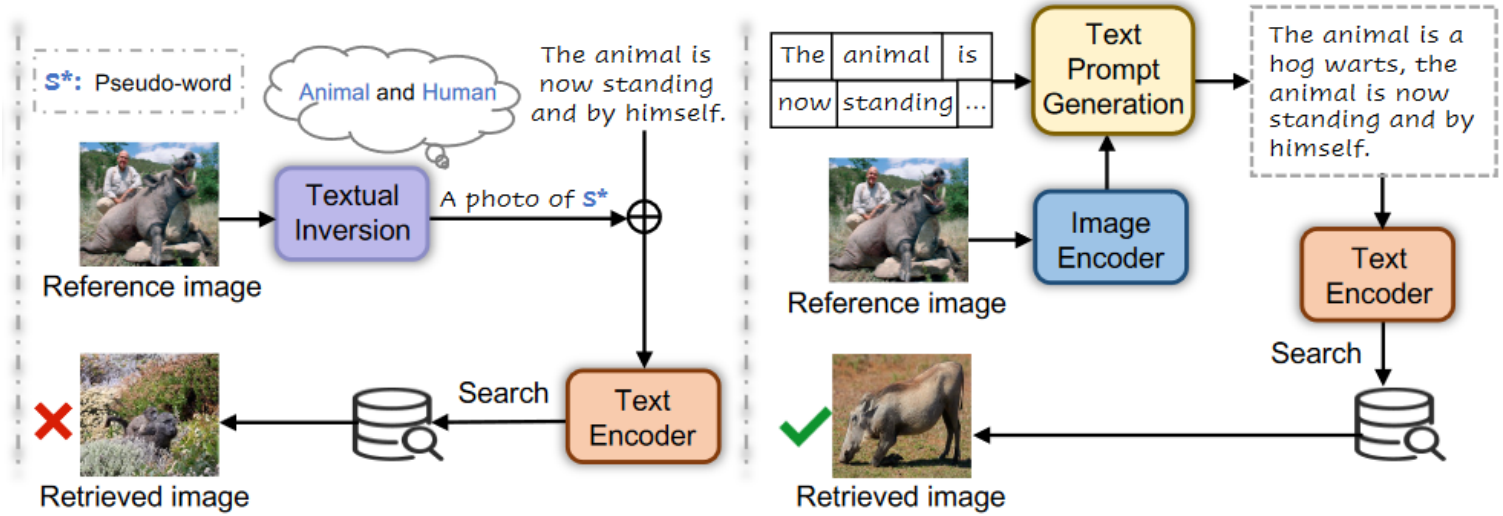
Improvements

Ideas:

- Prompt with pseudo-word ("**a photo of S***") are short which limits the ability to handle multiple objects or complex user queries.
- Incorporating LLMs for flexible reasoning, ensuring interpretability in CIR.
- Use LLMs or text prompt generation models to extend phrases.
- Encapsulate pseudo-word-tokens with both reference image and relative caption.
- Use of captioning or image-text models can avoid the use of Textual Inversion.
- Human-understandable text embeddings by replacing traditional token inversion.

Proposed Work-1: CoIR with SENTENCE-LEVEL PROMPTS

- Sentence-level prompts aim to provide more precise descriptions of specific elements in the reference image that are pertinent to the relative caption.
- Sentence-level prompts enable the model to avoid interference from extraneous elements, such as irrelevant background and other objects in the image.



Bai, Y. (2023, October 9). Sentence-level prompts benefit composed image retrieval. arXiv.org.

Proposed Work-1: Architecture

Learning Objective

Image-text contrastive loss

$$\mathcal{L}_c = -\frac{1}{\mathcal{B}} \sum_{i \in \mathcal{B}} \log \frac{\exp\left(\tau \boldsymbol{u}_i^T \boldsymbol{v}_i\right)}{\sum_{j \in \mathcal{B}} \exp\left(\tau \boldsymbol{u}_i^T \boldsymbol{v}_j\right)},$$

Text prompt alignment loss

$$\mathcal{L}_a = \|\boldsymbol{p}_i - \boldsymbol{p}_i'\|_2,$$

Where, **p′i** denotes the auxiliary text prompt generated using an optimization-based process

Final Loss

L = Lc + γLa

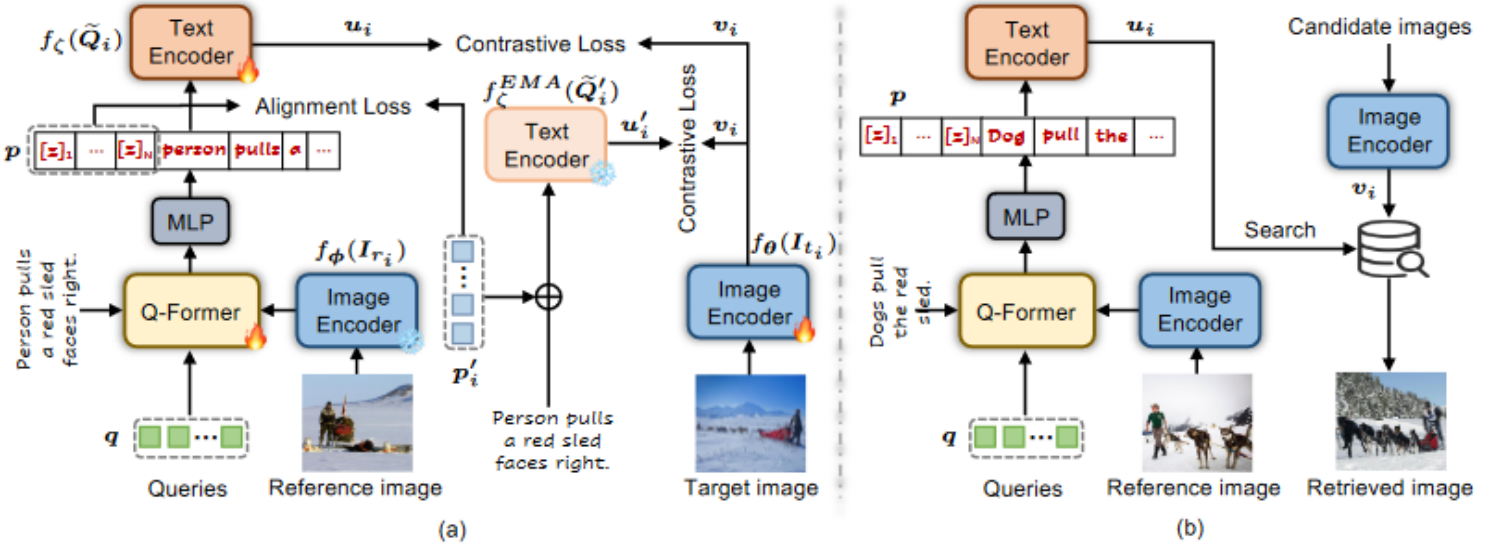
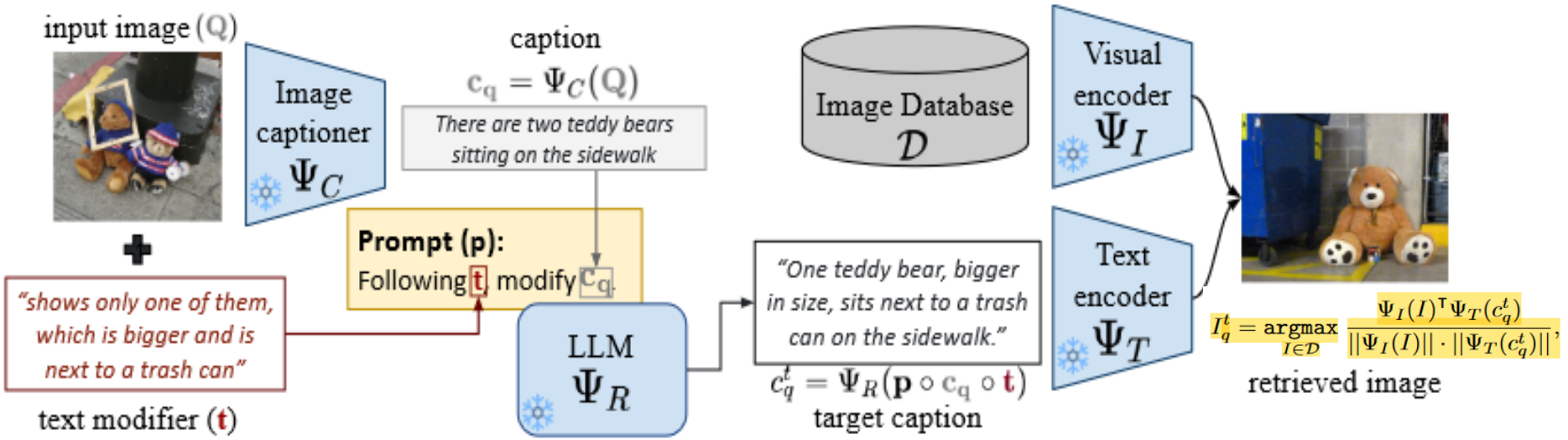


Figure 2: **Overall pipeline** of our proposed SPRC. **(a)** Illustration of the training phase, where proper sentence-level prompts are integrated into the relative captions thereby providing essential textual prompts containing information about the objects depicted in the reference images and their relative captions. Even in cases where the reference images involve multiple objects and the relative captions encompass complex modifications, these sentence-level prompts guarantee that the retrieval target image can be correctly reflected. **(b)** Illustration of the test phase, the learned sentence-level prompts are used to enhance the relative caption for text-image retrieval.

Bai, Y. (2023, October 9). Sentence-level prompts benefit composed image retrieval. arXiv.org.

Proposed Work-2 CIR with VISION-BY-LANGUAGE

- A model which leverages large-scale vision-language models (VLMs) and large language models (LLMs) to achieve CIR in a training-free manner.
- It uses a pre-trained generative VLM to caption the reference image and then asks an LLM to recompose the caption based on the target modification text. The resulting composition is used for retrieval through models like CLIP.
- It eliminates the need for task-specific model training, relying on pre-trained models for efficient and scalable performance.



Karthik, S., Roth, K., Mancini, M., & Akata, Z. (2023). Vision-by-Language for Training-Free Compositional Image Retrieval. arXiv preprint arXiv:2310.09291.

Thank you for your Attention!

References

- Vo, Nam, et al. Composing Text and Image for Image Retrieval -an Empirical Odyssey. CVPR 2019
- Hosseinzadeh, M., & Wang, Y. (2020). Composed Query Image Retrieval Using Locally Bounded Features. CVPR 2020.
- Lee, S., Kim, D., & Han, B. (2021). CoSMo: Content-Style Modulation for Image Retrieval with Text Feedback. CVPR 2021
- Baldrati, A., Bertini, M., Uricchio, T., & Del Bimbo, A. (2022). Conditioned and composed image retrieval combining and partially fine-tuning CLIP-based features. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPR 2022).
- Gal, R., Alaluf, Y., Atzmon, Y., Patashnik, O., Bermano, A. H., Chechik, G., & Cohen–Or, D. (2022). An Image is Worth One Word: Personalizing Text-to-Image Generation using Textual Inversion. arXiv (Cornell University).
- Cohen, N., Gal, R., Meirom, E. A., Chechik, G., & Atzmon, Y. (2022). “This is my unicorn, Fluffy”: Personalizing frozen Vision-Language representations. In Lecture Notes in Computer Science (pp. 558–577).
- Saito, K. (2023, February 6). Pic2Word: Mapping pictures to words for zero-shot composed image retrieval. arXiv.org. https://arxiv.org/abs/2302.03084
- Baldrati, A. (2023, March 27). Zero-Shot Composed Image Retrieval with Textual Inversion. arXiv.org. https://arxiv.org/abs/2303.15247
- Bai, Y. (2023, October 9). Sentence-level prompts benefit composed image retrieval. arXiv.org. https://arxiv.org/abs/2310.05473
- Karthik, S., Roth, K., Mancini, M., & Akata, Z. (2023). Vision-by-Language for Training-Free Compositional Image Retrieval. arXiv preprint arXiv:2310.09291.