Class Concept Representation from Contextual Texts for Training-Free Multi-Label Recognition

Anonymous Author(s)
Affiliation
Address
email

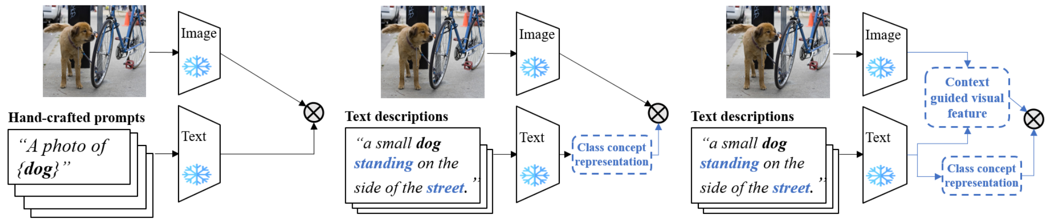
Abstract

The power of large vision-language models (VLMs) has been demonstrated for diverse vision tasks including multi-label recognition with training-free approach or prompt tuning by measuring the cosine similarity between the text features related to class names and the visual features of images. Prior works usually formed the class-related text features by averaging simple hand-crafted text prompts with *class names* (e.g., *"a photo of {class name}"*). However, they may not fully exploit the capability of VLMs considering how humans form the concepts on words using rich contexts with the patterns of co-occurrence with other words. Inspired by that, we propose *class concept* representation for zero-shot multi-label recognition to better exploit rich contexts in the massive descriptions on images (e.g., captions from MS-COCO) using large VLMs. Then, for better aligning visual features of VLMs to our class concept representation, we propose context-guided visual representation that is in the same linear space as class concept representation. Experimental results on diverse benchmarks show that our proposed methods substantially improved the performance of zero-shot methods like Zero-Shot CLIP and yielded better performance than zero-shot prompt tunings that require additional training like TaI-DPT. In addition, our proposed methods can *synergetically* work with existing prompt tuning methods, consistently improving the performance of DualCoOp and TaI-DPT in a training-free manner with negligible increase in inference time.

1 Introduction

The goal of multi-label image recognition is to assign all semantic labels (or class names) within an image [10, 44, 48, 11, 27, 33, 31]. Differing from single-label recognition, multi-label recognition addresses a broader range of practical applications such as image retrieval [36, 39], recommendation systems [52, 8], medical diagnosis recognition [43] and retail checkout recognition [17, 45]. However, one of the challenges in multi-label recognition is the difficulty of collecting full label annotations, which is laborious and prone to missing. To alleviate it, recent works have investigated training with incomplete labels such as partial labels [37, 6, 31, 15, 9] or a single positive label [13, 46].

Recent advances of large vision-language models (VLMs) [32, 2, 22, 25, 47, 49] has demonstrated their strong transferability on various downstream tasks with great performance. Contrastive Language-Image Pretraining (CLIP) achieved impressive performance in zero-shot classification by measuring the cosine similarity between images and class-related hand-crafted text prompts [32]. Fine-tuning VLMs for adapting desired downstream datasets [32] can further improve performance for targeted tasks, but tuning millions of parameters is usually undesirable due to computation burden and possible forgetting. Prompt tuning has been investigated as an efficient and low-cost training paradigm [54, 53], learning only a few context tokens of VLMs for a given task. In multi-label recognition, prompt tuning with CLIP has been investigated for distinguishing multiple objects in an



(a) ZSCLIP (mAP 57.4) w/ Class name

(b) ZSCLIP (mAP 61.5) + Class concept representation

(c) ZSCLIP (mAP 70.0) + Class concept representation + Context-guided visual feature

Figure 1: Illustration of our methods applied to zero-shot CLIP (ZSCLIP) [32]. (a→b) Class concept is formed from the text descriptions that contain rich contextual information with relevant class names and other related words, yielding substantially improved performance without aligning with visual features yet. (b→c) Context-guided visual feature is transformed from visual feature so that it is in the same linear space as class concept representation, yielding significantly improved performance.

image [37, 18, 41], mitigating the difficulty of acquiring annotated samples. However, prompt tuning inherently requires labeled data with additional training and may be susceptible to overfitting for context tokens, hindering generalization. The capability of VLMs for label-free and/or training-free classification has been exploited using prompt engineering [32, 34, 50, 4]. However, prompt ensembles by averaging text features from simple hand-crafted prompts (*e.g.*, *"a sketch of {class name}"*) yielded marginal improvements and struggled with multi-label recognition. Thus, the approach of prior works on zero-shot or prompt-tuning based multi-label recognition using *class names* to obtain class-related text features from VLMs may not use the full capacity of VLMs properly.

Humans form concepts on words from past experience, especially using their patterns of co-occurrence with *other words* [5, 29, 20]. Inspired by this perspective in cognitive neuroscience, we propose a novel approach of exploiting VLMs for multi-label recognition by replacing single *class name*-related hand-crafted prompts with our proposed *class concept* representation using text descriptions such as "A **person** holding a large pair of scissors," capturing rich contextual information with target class names (e.g., person) as well as related words (e.g., holding, scissors). Our class concept will be constructed from rich contextual descriptions on classes that may contain diverse and realistic patterns of co-occurrence with target class name and other related class names. Then, this novel text features with class concept representation requires aligned visual features with them for multi-label recognition to properly match them with our class concepts. Thus, we propose context-guided visual features to bring VLM's visual features to the same representation domain as our class concept representation by using our sequential attention. See Fig. 1 for the differences of performing multi-label recognition using (a) prior zero-shot approach (ZS-CLIP), (b) our proposed class concepts from text descriptions and (c) our proposed context-guided visual features on the same space as the class concepts. We demonstrated that our proposed methods achieved improved performance on multiple benchmark datasets without additional training (tuning), without additional labels (text-image pairs) and with negligible increase in inference time. Here is the summary of the contributions:

- Proposing a novel class concept representation for training-free multi-label recognition tasks using VLMs from massive text descriptions inspired by how human forms concept on words.
- Proposing a context-guided visual feature, transformed onto the same text feature space as class concepts using sequential attention for better aligning multi-modal features.
- Demonstrating that our methods synergetically improve the performance of ZSCLIP and other state-of-the-art prompt tuning methods with a negligible increase in inference time.

2 Related Works

Multi-label image recognition with CLIP. Multi-Label Recognition (MLR) aims to identify all semantic labels within an image. However, it is difficult to collect the annotation of multi-label images which involve complex scenes and diverse objects. Recently, prompt tuning with the pre-trained vision-language model CLIP has been developed to address the high labeling costs of multi-label images in incomplete label setting. Among them, DualCoOp [37] proposed a novel prompt tuning approach that trains positive and negative learnable contexts with class names in the partially labeled setting. For mitigating data-limited or label-limited issues, TaI-DPT [18] proposed effective dual-grained prompt tuning method using easily accessible text descriptions. It is worth noting that TaI-DPT used the same text descriptions as ours not for performing training-free multi-label recognition itself, but for label-free prompt tuning by replacing the image features with the contextual text features (text as image) under the conventional framework of multi-label recognition with class name. SCPNet [14] is designed to leverage the structured semantic prior from CLIP to complement deficiency of label supervision for MLR with incomplete labels. CDUL [1] proposed unsupervised multi-label recognition through pseudo-labeling using CLIP, alleviating the annotation burden. Even though recent works has demonstrated outstanding performance of multi-label recognition task, they still require tuning costs or labeled dataset to adapt pre-trained CLIP to various downstream tasks. In this work, our method enables training-free and label-free adaptation of CLIP into downstream tasks, utilizing the text descriptions.

Training-free enhancement with CLIP. For single-label recognition, recent works has developed the training-free enhancement of CLIP. ZPE [4] weighted-averaged many prompts by automatically scoring the importance of each prompt in zero-shot manner for improving prompt ensemble technique. CALIP [19] designed a simple parameter-free attention module for zero-shot enhancement over CLIP without any tuning of model parameter. With few-shot samples, Tip-Adapter [51] proposed training-free approach for fast adaptation to target task, obtaining the weights of adapter using few-shot samples during inference. Since these methods were originally developed for single-label recognition, it is difficult to be directly applied to multi-label recognition. In multi-label recognition, our method enables training-free enhancement and demonstrated its effectiveness on the benchmark dataset.

3 Method

First of all, we propose *class concept* representation as a training-free approach for multi-label recognition instead of *class name* by exploiting pre-trained VLM and rich contextual text descriptions. Secondly, we also propose context-guided visual feature that can enhance the alignment of the visual feature of VLM with our novel class concept. Our proposed methods are label-free as well as training-free so that they can be applicable *synergetically* for most existing VLM-based multi-label recognition methods. The overall pipeline of our method is illustrated in Figure 2.

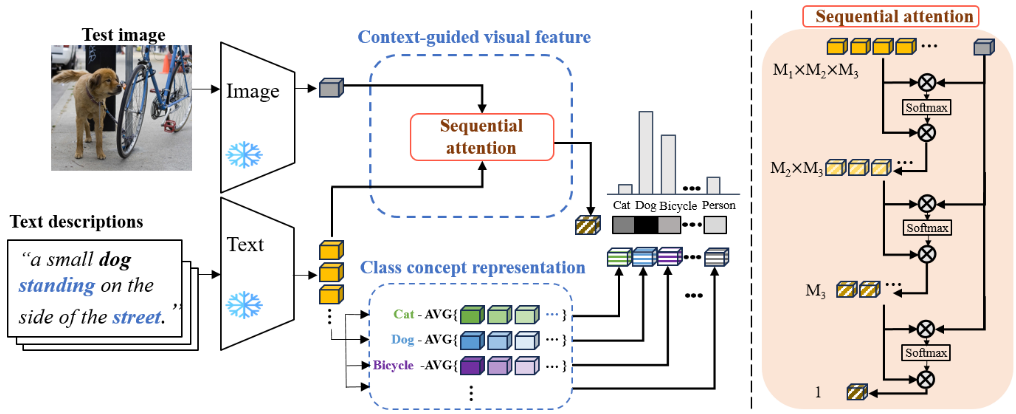
3.1 Class Concept Representation

Humans form concepts on words from past experience, often using their patterns of co-occurrence with *other words* [5, 29, 20]. For example, the word "apple" does not exist alone, but often comes with the verb "eat" or the noun "basket." However, it may not well associate with other words such as "fly" or "space." Fortunately, we can easily obtain rich contextual text descriptions from various public sources, including captions from benchmark datasets [26, 23, 24, 30], web crawling and large language models [38, 7, 40, 28]. These text descriptions do not only contain *class names*, but also include *other words* like class-related verbs and nouns in real-world contexts.

Assume that rich contextual text descriptions were gathered from the public sources that include one or multiple class names. We denote the set of text descriptions as $Z^{all} = \{z_1, z_2, ..., z_M\}$ where z_i refers to an individual text description. M denotes the total number of text descriptions across all classes. Note that M can be dynamically changed at inference since our proposed method does not require additional training, thus can be seen as test-time adaptation. Assuming that the target task uses the class names of person, scissors, clock, building and cake, the examples of the contextual text descriptions from Z^{all} are as follows:

"A **person** holding a large pair of **scissors**."
"A **clock** mounted on top of a **building** in the city."
"Half of a white **cake** with coconuts on top."

TaI-DPT [18] used these descriptions with rich contextual information as a surrogate for images to propose a label-free prompt tuning. In this work, we propose to use these descriptions to form concepts on class names to compare with images, so that ways of using them are completely different.



(a) Overall pipeline of our method (b) Detail of sequential attention

Figure 2: (a) Overall pipeline of our method. 1) Class concept representation: VLM's text features from the rich contextual descriptions associated with each class name are used to construct the class concept. 2) Context-guided visual features: VLM's visual features are sequentially transformed onto the class concept representation space using (b) sequential attention mechanism.

We define the class concept as a vector in the space constructed by the text descriptions as follows. Firstly, the linear space $\mathcal{Z}$ can be constructed by spanning the VLM's text features from all text descriptions z_i in Z^{all} using the VLM's text encoder $\mathcal{E}_{\text{txt}}(z_i) \in R^{1\times D}$, leading to $\mathcal{Z} = \text{span}\{\mathcal{E}_{\text{txt}}(z_1), \mathcal{E}_{\text{txt}}(z_2), \dots, \mathcal{E}_{\text{txt}}(z_M)\}$. Secondly, we propose the class concept for a target class name c as a vector $t_c^{concept}$ in the space $\mathcal{Z}$ by defining it as follows:

$$t_c^{concept} = \sum_{i=1}^{M} w_{c,i} \mathbb{1}_c(z_i) \mathcal{E}_{\text{txt}}(z_i) \in R^{1\times D} \tag{1}$$

where $\mathbb{1}_c(z_i)$ an indicator function such that $\mathbb{1}_c(z_i) = 1$ if the text description z_i contains the class name c and $\mathbb{1}_c(z_i) = 0$ otherwise. The weight $w_{c,i}$ is assigned to the text feature of each text description within a class c and it is assumed to be normalized within the class. In this work, we set $w_{c,i} = 1/\sum_j^M \mathbb{1}_c(z_j)$ for $\forall i$, thus will be the same for all i for each class, which was guided by the prior work on prompt ensembling [4], demonstrated that the prompt ensembling with equal weights achieved significant performance gains that were comparable to weighted ensembling for single-label recognition. Each class concept can be stored individually or together as a matrix.

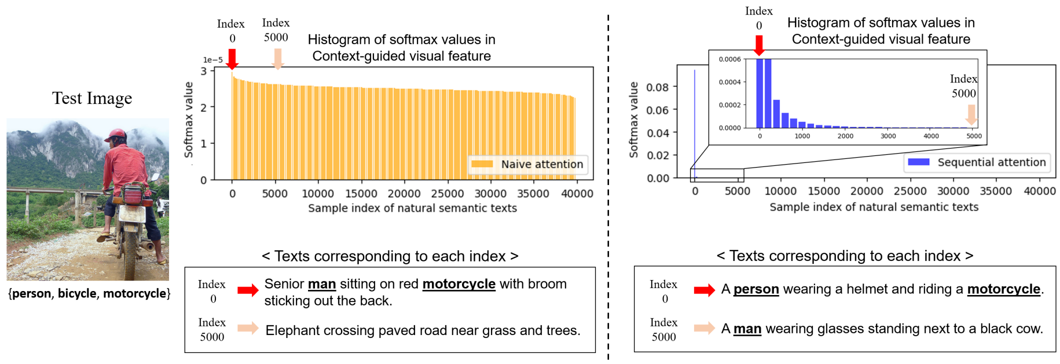
Our class concept representation thus consists of various text features including diverse contextual information related to the target class name. For instance, the descriptions for the class name "dog" should contain the target class name as the following examples of the text descriptions:

"A **dog** greets a sheep that is in a sheep pen."
"A woman walks her **dog** on a city sidewalk."
"A **dog** with goggles is in a motorcycle side car."

Note that the descriptions include the target class name (bold) as well as other related words in class-related contexts (underline) as intended. We expect that our novel class concepts will be beneficial for multi-label recognition due to other nouns (other class names) as well as other verbs to better explain the context where the target class name is used. In this work, we obtain the texts from two sources to collect the sufficient contextual text descriptions. The first source is the MS-COCO dataset [26] that is publicly available and the second source is large language model(*i.e.*, GPT-3.5[28]) that can generate the several sentences quickly if the set of class names related to the target task were provided.

3.2 Context-Guided Visual Feature

Our novel class concept representation forms new vectors for diverse class names in the linear space $\mathcal{Z}$ instead of the embedding space of the VLM where the text and image encoders were relatively well-aligned. Thus, it is expected that the class concept representation and the VLM's visual feature



(a) Naïve attention (b) Proposed sequential attention

Figure 3: Softmax values can be used to weigh the relevance with the given image. However, (a) naive attention mechanisms yielded almost equal softmax values, thus may include texts with low relevance. The proposed sequential attention method focuses on a subset of texts most relevant to the test image, thus can transforms visual features to context-guided visual features for multi-label recognition by assigning very high softmax value to the relevant text at index 0 while very low softmax value to the irrelevant text at index 5000.

may not be aligned well. Here, we propose context-guided visual feature by transforming the visual features of the VLM onto the same space as the class concept representation $\mathcal{Z}$ by using our sequential attention with the text descriptions Z^{all} that were used for class concept construction.

For the target image q and the VLM's visual encoder $\mathcal{E}_{\text{img}}(q)$, the L2-normalized global visual feature f is obtained by using $\mathcal{E}_{\text{img}}(q) \in R^{1\times D}$ and the flatten local visual feature $F \in R^{HW\times D}$ is also constructed by using $\mathcal{E}_{\text{img}}(P_{i,j}(q))$ where $P_{i,j}(\cdot)$ is an extractor of the (i,j)th patch of the input image. Then, we aim to transform both the global visual feature vector f and the local visual feature matrix F onto the same linear space $\mathcal{Z}$ as our class concept representation. One easy way is to "project" these visual features f and F onto the space $\mathcal{Z}$ by computing the cosine similarity between visual features (f and the column vectors of F) and all the text features $t_i = \mathcal{E}_{\text{txt}}(z_i) \in R^{1\times D}, i = 1,\ldots,M$ that constructed $\mathcal{Z}$. Unfortunately, when the softmax function is applied to the cosine similarity values, they tend to become similar, thus weigh both relevant and irrelevant texts almost equally as illustrated in Figure 3 (a). To address this challenge, we propose sequential attention, applying the softmax function to part of the cosine similarity values by dividing them into G groups. For the text feature matrix $T = [t_1\ t_2\ \cdots\ t_M] \in R^{M\times D}$, let us determine M_i for $i = 1,\ldots,G$ such that $M = \Pi_{i=1}^{G} M_i$ and reshape the text feature matrix to be $T \in R^{M_1\times\cdots\times M_G\times D}$. Then, propose to sequentially apply the following attention process for G iterations for estimating both global and local context-guided visual features $v^{(k)}$ and $V^{(k)}$, respectively, at the kth iteration:

$$v^{(k)} = \begin{cases} T & \text{if } k = 0, \\ \text{Softmax}_{\text{dim}_k}\left(\frac{f(v^{(k-1)})^t}{\alpha_f}\right) v^{(k-1)} & \text{if } k > 0, \end{cases} \tag{2}$$

$$V^{(k)} = \begin{cases} T & \text{if } k = 0, \\ \text{Softmax}_{\text{dim}_k}\left(\frac{F(V^{(k-1)})^t}{\alpha_F}\right) V^{(k-1)} & \text{if } k > 0, \end{cases} \tag{3}$$

where α_f and α_F denote the modulation parameters, Softmax_{M_k} refers to the softmax operation applied along the dimension corresponding to M_k. In this work, we utilize $v^{(3)}$ and $V_{(3)}$ to compute classification score. The sequential attention process is illustrated in Figure 2 (b). Figure 3 further demonstrates that our sequential attention is particularly effective in handling massive text descriptions. Without sequential attention, weighted averaging essentially becomes equal averaging.

3.3 Multi-Label Recognition with Class Concepts

Architecture of model. Two encoders of CLIP are denoted as $\mathcal{E}_{\text{img}}$ and $\mathcal{E}_{\text{txt}}$ for the visual encoder and text encoder, respectively. Following TaI-DPT [18], we adopt the structure of double-grained prompts (DPT), which has been shown effective for enhancing zero-shot multi-label recognition performance. To obtain visual representations at both coarse-grained and fine-grained levels, we extract the local visual feature map $F = \mathcal{E}_{\text{img}}(x) \in R^{HW \times D}$ is extracted before attention pooling layer, where H and W are spatial dimension of visual feature. After attention pooling layer, we obtain the global visual feature $f \in R^{1 \times D}$. Similarly, text features $t = \mathcal{E}_{\text{txt}}(z) \in R^{1 \times D}$are obtained by projecting the End-of-Sentence (EOS) token of the text prompt. Thus, we leverage both global and local visual features for multi-label recognition.

Inference. Through our sequential attention, we obtain the context-guided visual features $v^{(G)}$ and $V^{(G)}$ at both global and local levels, respectively. The similarity score S^{glo} and S^{loc} are calculated between the transformed context-guided visual features $v^{(G)}$, $V^{(G)}$ and the class concepts $t_c^{concept}$ using the cosine similarity $\Psi(\cdot,\cdot)$ as follows:

$$S_c^{tot} = S_c^{glo} + S_c^{loc} = \Psi(v^{(G)}, t_c^{concept}) + \sum_{j=1}^{HW} \text{Softmax}(s_{c,j}^{loc}) \cdot s_{c,j}^{loc} \tag{4}$$

where S_c^{tot} is the classification score for the class c and $s_{c,j}^{loc} = \Psi([V^{(G)}]_j, t_c^{concept})$ for the class c. For obtaining S_c^{loc}, we employ the spatial aggregation over HW [37].

Finally, we combined ZSCLIP[32] and other prompt tuning methods with our training-free approach through simple logit ensemble. In our experiments, we demonstrate the effectiveness of integrating of our method with existing methods, thereby boosting the performance of multi-label recognition.

4 Experiments

4.1 Implementation Details

Architecture. We empoly CLIP ResNet-50 in the Table. 2 and Table. 3 and ResNet-101 in other experiments as the visual encoders and the CLIP transformer as the text encoder for ZSCLIP[32], TaI-DPT [18], DualCoOP [37] and our method in the paper. In addition, ZSCLIP[32], TaI-DPT [18] and our method are based on the double-grained prompt [18] for both global and local features[1].

Datasets. For evaluation, we performed multi-label recognition experiments on 3 benchmark datasets. MS-COCO [26] consists of 80 classes with 82,081 images for training and 40,504 images for test. VOC2007[16] consists of 20 object classes with 5,011 image for training and 4,952 images for test. NUS-WIDE[12] consists of 81 concepts with 161,789 image for training and 107,859 image for test. For MS-COCO [26] and VOC2007 [26], text description source is from MS-COCO [26]. For NUS-WIDE[12], we gathered the text descriptions from GPT-3.5. Note that there is example of text template for extracting sentence from GPT-3.5 in supplementary.

Inference Details. In the paper, we set the total number of text descriptions, denoted as M, for the MSCOCO[26], VOC2007[16], and NUS-WIDE[12] at 40,000, 64,000, and 57,600, respectively. Note that we prepared the text embeddings of every text descriptions from CLIP text encoder in advance. We set values of modulation parameter α via validation.

4.2 Evaluation on Limited Data Setting

To evaluate our method, we conducted the experiments in limited data scenarios, including zero-shot and few-shot settings for data-limited cases and partially labeled setting for label-limited cases. Note that only our method provides training-free enhancement of CLIP without tuining cost for multi-label recognition. Therefore, our method can be easily combined with existing methods to improve their performance.

Evaluation on Zero-Shot Setting. We performed comparison studies for different zero-shot and fully supervised methods in multi-label image recognition. To evaluate the effectiveness of our method which, we combined our method with existing zero-shot methods, ZSCLIP[32] and TaI-DPT [18], for zero-shot setting, as shown in Table 1. Additionally, we utilized the fully supervised method, DualCoOp[37] with our method, for zero-shot learning setting (ZSL) as presented in Table 2.

Table 1 summarizes the results of the zero-shot experiment on benchmark datasets. In MS-COCO [26] and VOC2007 [16], TaI-DPT [18] and our method utilized the public language data from MS-COCO [26]. By applying our method to ZSCLIP[32] and TaI-DPT [18] during inference, we yield performance improvements without tuning costs. Especially, the performance of ZSCLIP[32] with

[1] `https://github.com/guozix/TaI-DPT`

Table 1: Multi-label recognition with zero-shot methods on MS-COCO [26], VOC2007 [16] and NUS-WIDE [12]. Without training, our method significantly enhances the performance of existing zero-shot methods. The evaluation is based on mAP.

Training-free	Methods	MS-COCO[26]	VOC2007[16]	NUS-WIDE[12]
✓	ZSCLIP[32]	57.4	82.8	37.3
✓	+Ours	70.0 (+12.6)	89.2 (+6.4)	46.6 (+9.3)
✗	TaI-DPT[18]	68.0	88.9	46.5
✓	+Ours	70.9 (+2.9)	90.1 (+1.2)	49.1 (+2.6)

Table 2: Multi-label recognition with 17 unseen classes on MS-COCO [26]. In zero-shot learning (ZSL , recognizing only unseen classes) and generalized ZSL (GZSL, recognizing both seen and unseen classes), our method effectively supplements the complementary information of unseen classes to the supervised DualCoOp[37] on 48 seen classes. The evaluation is based on mAP.

Methods	ResNet-50		ResNet-101	
	ZSL	GZSL	ZSL	GZSL
DualCoOp[37]	78.2	70.2	82.9	74.9
+Ours	82.9 (+4.7)	73.2 (+3.0)	87.6 (+4.7)	78.0 (+3.1)

our method is notably enhanced, achieving better and comparable performance to TaI-DPT [18], which requires mild tuning. In NUSWIDE [12], we incorporate contextual text descriptions from a large language model (GPT-3.5) to validate the potential of utilizing generated texts instead of well-curated caption data. With provided class name of NUSWIDE [12], we readily gathered the massive set of text descriptions within a short amount of time. TaI-DPT [18] is trained with the public caption data from OpenImages[23]. Our method exceeds the performance of ZSCLIP[32] and TaI-DPT [18] by a large margin, with improvements of 9.3 mAP and 2.6 mAP, respectively.

Table 2 shows the results of the zero-shot learning setting for unseen classes. In MS-COCO [26], we follow the DualCoOp[37] and split the dataset into 48 seen classes and 17 unseen classes. The evaluation is conducted in both zero-shot setting (ZSL, recognizing only unseen classes) and generalized zero-shot setting (GZSL, recognizing both seen and unseen classes). Based on prompt tuning, DualCoOp[37] trains learnable context tokens on 48 seen classes and achieves the state-of-the-art performance on both ZSL and GZSL. Our method was originally designed to handle novel classes (unseen classes) by leveraging text descriptions. As a result, our method significantly improved the ZSL and GZSL performance of the supervised DualCoOp[37] by providing complementary information. Table 1 and Table 2 demonstrate the effectiveness of our method performing training-free enhancement of CLIP with only text descriptions that are easily obtained.

Evaluation on Few-Shot Setting. We performed comparison study with few-shot methods in multi-label recognition. In TaI-DPT [18], they have investigate to confirm the effectiveness of their zero-shot method. Here, we further validate our method, which is zero-shot test-time task adaption without tuning costs.

Table 3 summarizes the results of the few-shot methods on MS-COCO dataset [26], especially using 1 and 5 shot samples for all classes. While existing few-shot methods [3, 35, 54, 51] demonstrated the performance enhancements with an increase of labeled samples, TaI-DPT [18] and our method are performed within the zero-shot setting. By applying our method with existing zero-shot methods (ZSCLIP[32] and TaI-DPT [18]), we consistently enhance performance, as already demonstrated in a zero-shot setting. In the absence of labeled samples and tuning, we achieved comparable performance with ML-FSL[35] and better results than other few-shot methods utilizing 5-shot samples.

Evaluation on Partially Labeled Setting. Due to high costs of annotation in multi-label image recognition, training with partially labeled samples [37, 21, 31, 6] has been studied. Following DualCoOp [37], we performed the evaluation of partially labeled setting. As shown in Table 4, our method supplements the decreased performance of DualCoOp [37] caused by partially labeled samples by providing complementary information during inference. Through zero-shot test time task adaptation without tuning costs, we consistently enhance the the performance of DualCoOp [37] on

Table 3: Comparison with few-shot methods on MS-COCO [26]. The evaluation is based on mAP with 16 novel classes. For each shot, we highlighted the best performance in bold.

Training-free	Methods	0-shot	1-shot	5-shot
✗	LaSO[3]	-	45.3	58.1
✗	ML-FSL[35]	-	**54.4**	**63.6**
✗	CoOp[54]	-	46.9	55.6
✓	Tip-Adapter[51]	-	53.8	59.7
✓	ZSCLIP[32]	49.7	-	-
✓	+Ours	58.5 (+8.8)	-	-
✗	TaI-DPT[18]	59.2	-	-
✓	+Ours	**61.4** (+2.2)	-	-

Table 4: Performance of multi-label recognition based on the partially labeled dataset [26, 16, 12]. Without training and labeled samples, our method consistently enhanced the performance of supervised DualCoOp [37] over all partial label ratio. DualCoOp [37] is reproduced and the evaluation is based on mAP.

Datasets	Method	Partial label									
		10%	20%	30%	40%	50%	60%	70%	80%	90%	**Avg.**
MS-COCO	SARB[31]	71.2	75.0	77.1	78.3	79.6	79.6	80.5	80.5	80.5	77.9
	DualCoOp[37]	80.8	82.2	82.8	83.0	83.5	83.8	83.9	84.1	84.2	82.7
	DualCoOp[37]+Ours	**81.5**	**82.8**	**83.3**	**83.5**	**84.0**	**84.2**	**84.4**	**84.5**	**84.6**	**83.6**
VOC2007	SARB[31]	83.5	88.6	90.7	91.4	91.9	92.2	92.6	92.8	92.9	90.7
	DualCoOp[37]	91.6	93.3	93.7	94.3	94.5	94.7	94.8	94.9	94.8	94.0
	DualCoOp[37]+Ours	**92.5**	**93.9**	**94.3**	**94.7**	**94.9**	**95.0**	**95.1**	**95.2**	**95.1**	**94.5**
NUS-WIDE	DualCoOp[37]	54.0	56.1	56.9	57.4	57.9	57.8	58.0	58.4	58.8	57.3
	DualCoOp[37]+Ours	**55.0**	**56.9**	**57.7**	**58.2**	**58.6**	**58.6**	**58.8**	**59.2**	**59.5**	**58.1**

all benchmark dataset. Furthermore, we achieved the performance of DualCoOp [37] trained with 90% labels by applying our method with DualCoOp trained with 60% labels from MS-COCO [26], 50% labels from VOC2007 [16], and 70% labels from NUSWIDE [12].

4.3 Ablation Study and Analysis

4.3.1 Effectiveness of our method

To verify the effectiveness of components of our method, we conducted an ablation study for analyzing our method. As shown in Table 5, we first proposed a novel class concept representation with text descriptions by class to ZSCLIP[32]. Since the text descriptions contain the semantic meaning among multiple class names and contextual information for multi-label recognition, the alignment between visual features of test image and text features are improved compared to the hand-crafted prompts as shown in the Fig.1. Thus, the performance is increased by 4.1 mAP and 1.1 mAP on MS-COCO [26] and VOC2007 [16], respectively. Then, we performed the context-guided visual feature using a large set of text descriptions, Z^{all}. Transforming the visual features into same text feature space as our class concept representation is essential to minimize the gap between visual feature from task-agnostic visual encoder and text features for each class. Constructing context-guided visual feature, our method yield remarkable performance gain by 8.5 mAP and 5.3 mAP on MS-COCO [26] and VOC2007 [16], respectively. Thus, we effectively designed our method that improves the alignment between visual and text features.

4.3.2 The Number of Text Descriptions

We investigate the effect of the number of text descriptions for our method. As shown in Table 6, we evaluated performance by increasing the number of randomly selected text descriptions from 1K to 32K texts. With only 1K text descriptions, our method enhances performance by approximately

Table 5: Effectiveness of our method on MS-COCO [26] and VOC2007 [16]. Each component of our method consistently improves performance, with significant enhancements achieved particularly in context-guided visual feature through narrowing the gap between visual and text features. The evaluation is based on mAP.

Method	MS-COCO [26]	VOC2007 [16]
Baseline (ZSCLIP[32])	57.4	82.8
+Class concept representation	61.5(+4.1)	83.9(+1.1)
+Context-guided visual feature	70.0(+8.5)	89.2(+5.3)

Table 6: Ablation studies in terms of the number of the text descriptions. As increasing the number of texts, we measured the performance of ZSCLIP[32] with our method in mAP on MS-COCO [26] and VOC2007 [16]. Note that ZSCLIP[32] achieves 57.4 mAP and 82.8 mAP for MS-COCO [26] and VOC2007 [16], respectively.

Dataset	Number of text descriptions					
	1K	2K	4K	8K	16K	32K
MS-COCO [26]	65.8	68.4	68.5	69.1	69.6	69.9
VOC2007 [16]	88.1	88.5	88.8	88.9	89.0	89.1

8 mAP on MS-COCO [26] and 5 mAP on VOC2007 [16], respectively. As the number of text descriptions ranges from 1K to 32K, the text embeddings of Z^{all} can cover the wider range of test dataset, resulting in increased performance gains. For adapting to novel classes during inference, our method not only achieves a significant performance improvement with only 1K texts but also further enhances performance as the quantity of texts increases.

4.3.3 Analysis of Inference Time

We analyzed the inference time of our method depending on the number of text descriptions. When extracting text embeddings from the text descriptions in advance, we measure the inference time as the number of text descriptions increases. ZSCLIP[32], as the baseline model, processes each sample for classification in 7.2ms. When the number of texts increases from 1K to 32K, integrating ZSCLIP[32] with our method only increases the inference time by 0.4-0.5ms, with tests conducted on the RTX3090. In addition, Our method (6.8GB) requires slightly more memory than ZSCLIP (6.5GB) on VOC2007 [16]. Therefore, our method presents a simple and efficient approach for training-free enhancement approach at inference.

5 Conclusion

In this paper, we propose a novel class concept representation from massive text descriptions for training-free multi-label recognition tasks. Inspired by how humans form concepts based on words, as studied in cognitive neuroscience, we replace single class name prompts with the class concept representation that capture various patterns of co-occurrence with other words. To further enhance alignment between multi-modal features of VLMs, we propose a context-guided visual representation that is transformed onto the same linear space as the class concept representation. Remarkably, our proposed method outperforms zero-shot prompt tuning methods such as TaI-DPT and achieves significant enhancements over ZSCLIP and other state-of-the-art prompt tuning methods without requiring parameter tuning or labeled samples, and with minimal inference time overhead.

Limitations. While our method achieved impressive results with training-free enhancement of CLIP, it exhibits limitations. First, a significant performance gap exists compared to prompt tuning methods with full samples, like DualCoOp [37]. Second, the computational memory demands of our method grow at a faster rate than ZSCLIP[32] as the batch size increases.

References

[1] Rabab Abdelfattah, Qing Guo, Xiaoguang Li, Xiaofeng Wang, and Song Wang. Cdul: Clip-driven unsupervised learning for multi-label image classification. In *ICCV*, 2023.

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *NeurIPS*, 2022.

[3] Amit Alfassy, Leonid Karlinsky, Amit Aides, Joseph Shtok, Sivan Harary, Rogerio Feris, Raja Giryes, and Alex M Bronstein. Laso: Label-set operations networks for multi-label few-shot learning. In *CVPR*, 2019.

[4] James Urquhart Allingham, Jie Ren, Michael W Dusenberry, Xiuye Gu, Yin Cui, Dustin Tran, Jeremiah Zhe Liu, and Balaji Lakshminarayanan. A simple zero-shot prompt weighting technique to improve prompt ensembling in text-image models. In *ICML*, 2023.

[5] Lawrence W Barsalou. Perceptual symbol systems. *Behavioral and brain sciences*, 22(4):577–660, 1999.

[6] Emanuel Ben-Baruch, Tal Ridnik, Itamar Friedman, Avi Ben-Cohen, Nadav Zamir, Asaf Noy, and Lihi Zelnik-Manor. Multi-label classification with partial annotations using class-aware selective loss. In *CVPR*, 2022.

[7] 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. In *NeurIPS*, 2020.

[8] Dolly Carrillo, Vivian F López, and María N Moreno. Multi-label classification for recommender systems. In *Trends in Practical Applications of Agents and Multiagent Systems: 11th International Conference on Practical Applications of Agents and Multi-Agent Systems*, pages 181–188. Springer, 2013.

[9] Tianshui Chen, Tao Pu, Hefeng Wu, Yuan Xie, and Liang Lin. Structured semantic transfer for multi-label recognition with partial labels. In *AAAI*, 2022.

[10] Tianshui Chen, Muxin Xu, Xiaolu Hui, Hefeng Wu, and Liang Lin. Learning semantic-specific graph representation for multi-label image recognition. In *CVPR*, pages 522–531, 2019.

[11] Zhao-Min Chen, Xiu-Shen Wei, Peng Wang, and Yanwen Guo. Multi-label image recognition with graph convolutional networks. In *CVPR*, 2019.

[12] Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yantao Zheng. Nus-wide: a real-world web image database from national university of singapore. In *CIVR*, pages 1–9, 2009.

[13] Elijah Cole, Oisin Mac Aodha, Titouan Lorieul, Pietro Perona, Dan Morris, and Nebojsa Jojic. Multi-label learning from single positive labels. In *CVPR*, 2021.

[14] Zixuan Ding, Ao Wang, Hui Chen, Qiang Zhang, Pengzhang Liu, Yongjun Bao, Weipeng Yan, and Jungong Han. Exploring structured semantic prior for multi label recognition with incomplete labels. In *CVPR*, 2023.

[15] Thibaut Durand, Nazanin Mehrasa, and Greg Mori. Learning a deep convnet for multi-label classification with partial labels. In *CVPR*, 2019.

[16] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010.

[17] Marian George and Christian Floerkemeier. Recognizing products: A per-exemplar multi-label image classification approach. In *ECCV*, 2014.

[18] Zixian Guo, Bowen Dong, Zhilong Ji, Jinfeng Bai, Yiwen Guo, and Wangmeng Zuo. Texts as images in prompt tuning for multi-label image recognition. In *CVPR*, 2023.

[19] Ziyu Guo, Renrui Zhang, Longtian Qiu, Xianzheng Ma, Xupeng Miao, Xuming He, and Bin Cui. Calip: Zero-shot enhancement of clip with parameter-free attention. In *AAAI*, 2023.

[20] Paul Hoffman, James L. McClelland, and Matthew A. Lambon Ralph. Concepts, control, and context: A connectionist account of normal and disordered semantic cognition. *Psychological Review*, 125(3):293–328, Apr.

[21] Ping Hu, Ximeng Sun, Stan Sclaroff, and Kate Saenko. Dualcoop++: Fast and effective adaptation to multi-label recognition with limited annotations. *arXiv preprint arXiv:2308.01890*, 2023.

[22] Chao Jia, Yinfei 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 *ICML*. PMLR, 2021.

[23] Ivan Krasin, Tom Duerig, Neil Alldrin, Vittorio Ferrari, Sami Abu-El-Haija, Alina Kuznetsova, Hassan Rom, Jasper Uijlings, Stefan Popov, Andreas Veit, et al. Openimages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://github. com/openimages*, 2017.

[24] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 2017.

[25] Yangguang Li, Feng Liang, Lichen Zhao, Yufeng Cui, Wanli Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. *arXiv preprint arXiv:2110.05208*, 2021.

[26] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[27] Yongcheng Liu, Lu Sheng, Jing Shao, Junjie Yan, Shiming Xiang, and Chunhong Pan. Multi-label image classification via knowledge distillation from weakly-supervised detection. In *ACMMM*, 2018.

[28] OpenAI. Gpt-4 technical report, 2023.

[29] Karalyn Patterson, Peter J Nestor, and Timothy T Rogers. Where do you know what you know? the representation of semantic knowledge in the human brain. *Nature reviews neuroscience*, 8(12):976–987, 2007.

[30] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*, 2015.

[31] Tao Pu, Tianshui Chen, Hefeng Wu, and Liang Lin. Semantic-aware representation blending for multi-label image recognition with partial labels. In *AAAI*, 2022.

[32] 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 *ICML*, 2021.

[33] Nikolaos Sarafianos, Xiang Xu, and Ioannis A Kakadiaris. Deep imbalanced attribute classification using visual attention aggregation. In *ECCV*, 2018.

[34] Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. Test-time prompt tuning for zero-shot generalization in vision-language models. *Advances in Neural Information Processing Systems*, 35:14274–14289, 2022.

[35] Christian Simon, Piotr Koniusz, and Mehrtash Harandi. Meta-learning for multi-label few-shot classification. In *WACV*, 2022.

[36] Josef Sivic and Andrew Zisserman. Video google: Efficient visual search of videos. *Toward category-level object recognition*, pages 127–144, 2006.

[37] Ximeng Sun, Ping Hu, and Kate Saenko. Dualcoop: Fast adaptation to multi-label recognition with limited annotations. *NeurIPS*, 2022.

[38] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Alpaca: A strong, replicable instruction-following model. *Stanford Center for Research on Foundation Models. https://crfm.stanford.edu/2023/03/13/alpaca.html*, 2023.

[39] Ivona Tautkute, Tomasz Trzciński, Aleksander P Skorupa, Łukasz Brocki, and Krzysztof Marasek. Deepstyle: Multimodal search engine for fashion and interior design. *IEEE Access*, 2019.

[40] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[41] Ao Wang, Hui Chen, Zijia Lin, Zixuan Ding, Pengzhang Liu, Yongjun Bao, Weipeng Yan, and Guiguang Ding. Hierarchical prompt learning using clip for multi-label classification with single positive labels. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 5594–5604, 2023.

[42] Shuai Wang, Daoan Zhang, Zipei Yan, Jianguo Zhang, and Rui Li. Feature alignment and uniformity for test time adaptation. In *CVPR*, 2023.

[43] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, and Ronald M Summers. Tienet: Text-image embedding network for common thorax disease classification and reporting in chest x-rays. In *CVPR*, 2018.

[44] Ya Wang, Dongliang He, Fu Li, Xiang Long, Zhichao Zhou, Jinwen Ma, and Shilei Wen. Multi-label classification with label graph superimposing. In *AAAI*, 2020.

[45] Xiu-Shen Wei, Quan Cui, Lei Yang, Peng Wang, and Lingqiao Liu. Rpc: A large-scale retail product checkout dataset. *arXiv preprint arXiv:1901.07249*, 2019.

[46] Ning Xu, Congyu Qiao, Jiaqi Lv, Xin Geng, and Min-Ling Zhang. One positive label is sufficient: Single-positive multi-label learning with label enhancement. *NeurIPS*, 2022.

[47] Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. Filip: Fine-grained interactive language-image pre-training. *arXiv preprint arXiv:2111.07783*, 2021.

[48] Vacit Oguz Yazici, Abel Gonzalez-Garcia, Arnau Ramisa, Bartlomiej Twardowski, and Joost van de Weijer. Orderless recurrent models for multi-label classification. In *CVPR*, 2020.

[49] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021.

[50] Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18123–18133, 2022.

[51] Renrui Zhang, Rongyao Fang, Wei Zhang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free clip-adapter for better vision-language modeling. *arXiv preprint arXiv:2111.03930*, 2021.

[52] Yong Zheng, Bamshad Mobasher, and Robin Burke. Context recommendation using multi-label classification. In *IEEE/WIC/ACM International Joint Conferences on WI and IAT*, volume 2, pages 288–295, 2014.

[53] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *CVPR*, 2022.

[54] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *IJCV*, 2022.

A Generation of Text Descriptions using LLMs

Our proposed method leverages the text descriptions for enhancing the alignment between the visual and text features. In practice, gathering the proper text descriptions is an essential process for replacing the hand-crafted prompts. As mentioned in the main paper, the text descriptions can be readily gathered from benchmark dataset, web crawling, or large language models. Recent advances in large language models (LLMs) enable to rapidly generate text descriptions that are similar to image captions in MS-COCO [26]. Therefore, we utilized the generated text descriptions from large language model. With provided class name of NUSWIDE [12], Fig. 6 illustrates the example of input prompt template and corresponding generated text descriptions using GPT3.5. We carefully designed the instruction of input prompt including main description, constraints, examples of bad and good cases, class names of target task and output format.

B Implementing Other Zero-Shot Training-Free Method

In single-label recognition, CALIP [19] proposed zero-shot alignment enhancement of CLIP for adapting target task without few-shot samples or additional training. The parameter-free attention module of cross-modal interaction effectively enhances the alignment of visual and text features. CALIP utilized the visual feature $F = \mathrm{Enc}_v(x_k) \in R^{HW \times D}$ via reshaping and the text feature $T = \mathrm{Enc}_t(P^h) \in R^{C \times D}$

Table 7: Ablation study of hyperparameter searching on validation set. We varied the modulation parameters $\alpha_{f,t}$ and $\alpha_{F,t}$ and searched the proper values for context-guided visual feature.

$1/\alpha_{f,t}$,$1/\alpha_{F,t}$	MS-COCO	VOC2007	NUS-WIDE
100, 50	69.45	87.62	47.32
80, 40	**69.51**	87.94	48.31
60, 30	69.25	**88.06**	49.05
40, 20	67.47	87.37	**49.82**
20, 10	64.13	85.04	47.33

where P^h is a hand-crafted description and C denotes the number of classes. The parameter-free attention module is formulated as follows:

$$F^a = \text{Softmax}(A/\alpha_t)T, \quad (5)$$

$$T^a = \text{Softmax}(A^T/\alpha_v)F \quad (6)$$

where the attention matrix is A=$FT^T \in R^{HW \times C}$, α_t and α_v are the modulation parameters of textual and visual features, respectively, and T^a and F^a are bidirectionally updated textual and visual features. After pooling the updated visual feature $F_v^a \in R^{1\times D}$ and the global visual feature $F_v \in R^{1\times D}$, the classification logit S is obtained as below:

$$S = \beta_1 \cdot F_v T^T + \beta_2 \cdot F_v T^{aT} + \beta_3 \cdot F_v^a T^T, \quad (7)$$

where $\beta_1, \beta_2, \beta_3$ are the weights for the three logits.

CALIP [19] tuned the hyperparameters β_2, β_3 for each dataset while fixed β_1 to be 1 for simplicity. As shown in Fig. 4, we have explored the value of β_2, β_3 for multi-label recognition setting on MS-COCO [26] and have observed that the parameter-free attention module consistently decreases the mAP performance since multi-label recognition covers the identification of multiple objects within an image, involving complex scene and diverse objects.

C Exploring Modulation Parameters

For hyperparameter searching, following existing methods for classification tasks, such as zero-shot [18, 19], training-free [51, 19], and test-time adaptation [42], we explore the modulation parameters α_t by conducting ablation studies on validation set. For simplicity, we set the value of $\alpha_{f,t}$ to be half of $\alpha_{F,t}$. As shown in Table 7, the value of ($1/\alpha_{f,t}$, $1/\alpha_{F,t}$) is suitable in the range of (40∼80,20∼40). In the experiments of main paper, we set the ($1/\alpha_{f,t}$, $1/\alpha_{F,t}$) as (80,40) for MS-COCO [26], (60,30) for VOC2007 [16] and (40,20) for NUSWIDE [12].

D Examples of Local Alignment Enhancement

In Fig. 5, we visualized the examples of local alignment enhancement by applying our method. Enhancing local alignment is important to recognize multiple objects in a test image [37]. Our proposed method enhances the local alignment between the visual features of test image and the text features of each class name, thereby suppressing the false-positive prediction. Therefore, Fig. 5 demonstrates the effectiveness of our method.

E Positive and Negative Societal Impacts

As a positive societal impact, our method can allow people with limited computing resources to achieve better performance in multi-label classification using existing vision-language models. This is because it does not require extensive training or labeled data. However, as a negative societal impact, the failure of classification could produce the negative side effects. For example, in security applications, incorrect classification of objects could lead to false alarms or missed detections, potentially compromising safety and security.

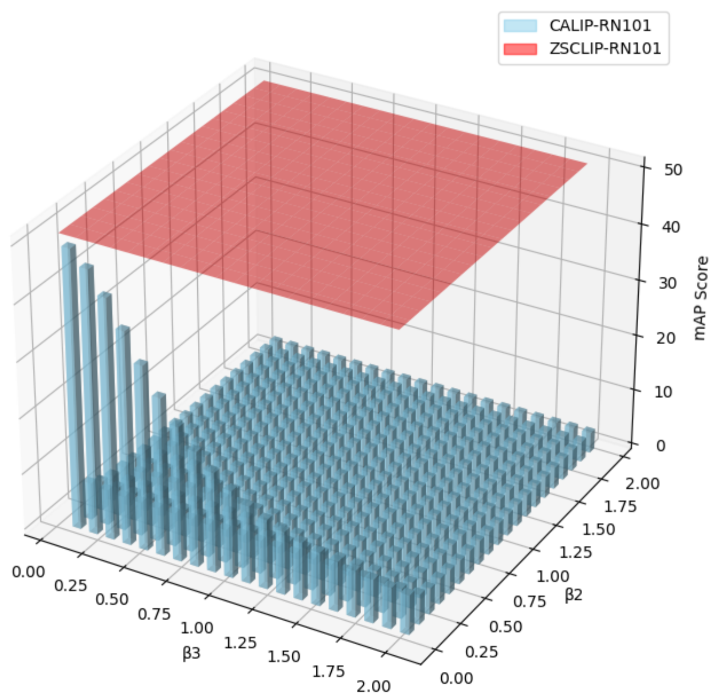


Figure 4: Results of hyperparameter searching of CALIP [19] on MS-COCO [26] on β_2 and β_3. Applying the parametric-free attention module of CALIP consistently decreases performance as compared to the zero-shot CLIP (ZSCLIP) [32].

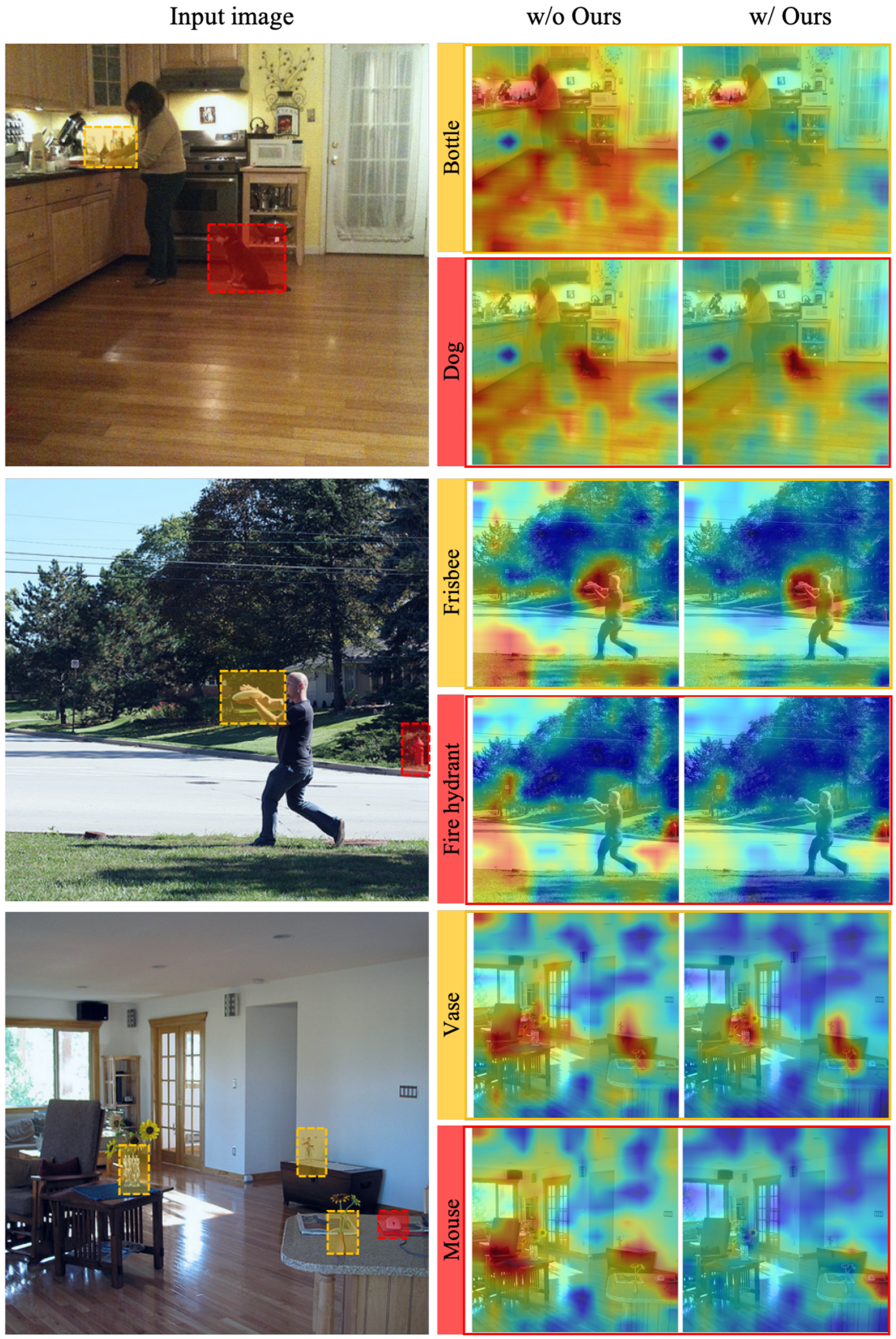


Figure 5: Additional examples of local alignment enhancement via our method. We visualized the test image in the left column and its corresponding spatial similarity map of each class name in the right column. The yellow and red boxes refer to the bounding boxes for different labels in a multi-label setting. By applying our method, the local alignment is enhanced across multiple objects in a test image, thereby suppressing false-positive predictions.

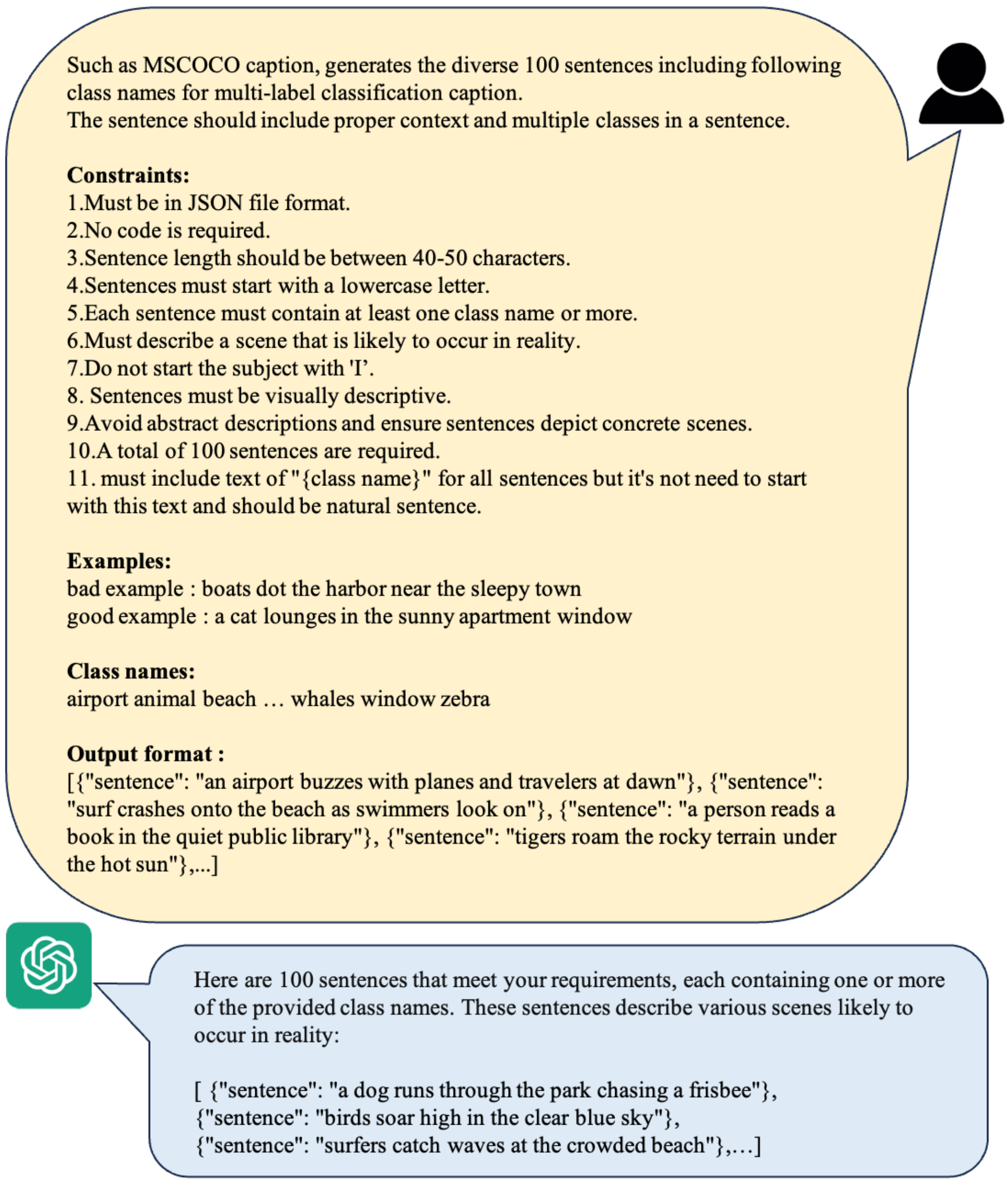


Figure 6: Example of text description generation using GPT3.5 for contextual text descriptions of NUSWIDE [12]. We carefully designed the input prompt to ensure that the generated sentences include the class name of the target task. The elements considered in designing the input prompt include the main description, constraints, examples, class names, and the desired output format.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We clearly state our contributions in both the abstract and introduction. Especially, we summarize our contributions in the last part of the introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of our method in conclusion.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: Our paper does not include theoretical results, assumptions and proof.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide the details of the used model, hyperparameters, source of datasets and proposed algorithm for reproducing main experimental results.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We provide open access to the code for our proposed method. In our experiments, we utilize a publicly accessible benchmark dataset described in Section **??**.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide the details of the experimental setting, including hyperparameters, in Sections 4.1 and C. The generation details of the texts used in our method are also provided in Section A.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We do not report the statistical significance of the experimental results, as our method does not rely on statistical variables for inference.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide the information of types of compute worker (GPU model), memory usage and inference time in Section. 4.3.3.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We abide by the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the positive and negative societal impacts of our paper in the supplementary material.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: Our paper does not pose a high risk for misuse in terms of model and dataset.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite the papers that provide datasets, code and models in the Section. 4.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We provide the detail documentation for the code in our submission.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: Our paper does not involve neither crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Our paper does not involve neither crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.