Prompt, Generate, then Cache: 
Cascade of Foundation Models makes Strong Few-shot Learners

Renrui Zhang∗2,3, Xiangfei Hu∗3, Bohao Li3, Siyuan Huang3, Hanqiu Deng3
Yu Qiao3, Peng Gao†1,3, Hongsheng Li2,4

1Shenzhen Institute of Advanced Technology, Chinese Academy of Science  
2CUHK MMLab  
3Shanghai Artificial Intelligence Laboratory  
4CPII under InnoHK

{zhangrenrui, huxiangfei, gaopeng}@pjlab.org.cn  
hсли@ee.cuhk.edu.hk

Abstract

Visual recognition in low-data regimes requires deep neural networks to learn generalized representations from limited training samples. Recently, CLIP-based methods have shown promising few-shot performance benefited from the contrastive language-image pre-training. We then question, if the more diverse pre-training knowledge can be cascaded to further assist few-shot representation learning. In this paper, we propose CaFo, a Cascade of Foundation models that incorporates diverse prior knowledge of various pre-training paradigms for better few-shot learning. Our CaFo incorporates CLIP’s language-contrastive knowledge, DINO’s vision-contrastive knowledge, DALL-E’s vision-generative knowledge, and GPT-3’s language-generative knowledge. Specifically, CaFo works by ‘Prompt, Generate, then Cache’. Firstly, we leverage GPT-3 to produce textual inputs for prompting CLIP with rich downstream linguistic semantics. Then, we generate synthetic images via DALL-E to expand the few-shot training data without any manpower. At last, we introduce a learnable cache model to adaptively blend the predictions from CLIP and DINO. By such collaboration, CaFo can fully unleash the potential of different pre-training methods and unify them to perform state-of-the-art for few-shot classification. Code is available at https://github.com/ZrrSkywalker/CaFo.

1. Introduction

Convolutional neural networks [44] and transformers [68] have attained great success on a wide range of vision tasks with abundant datasets [15]. Instead, for some data-deficient and resource-finite scenarios, few-shot learning [63,70] also becomes a research hotspot, where the networks are constrained to learn from limited images with annotations. Many previous works have been proposed in this field to enhance model’s generalization capability by meta learning [20,71], metric learning [74], and data augmentation [31,73]. Recently, CLIP [57] pre-trained by large-scale language-image pairs shows favorable zero-shot transfer ability for open-vocabulary visual recognition. The follow-up CoOp [83], CLIP-Adapter [24] and Tip-Adapter [76] further extend it for few-shot classification and achieve superior performance on various downstream datasets. This indicates that, even if the few-shot training data is insufficient, the large-scale pre-training has endowed the network with strong representation ability, which highly benefits the few-shot learning on downstream domains. Now that there exist various self-supervisory paradigms besides CLIP, could we adaptively integrate their pre-learned knowledge and collaborate them to be a better...
few-shot learner?

To tackle this issue, we propose CaFo, a Cascade of Foundation models blending the knowledge from multiple pre-training paradigms with a ‘Prompt, Generate, then Cache’ pipeline. As shown in Figure 1, we integrate CLIP [57], DINO [7], DALL-E [58], and GPT-3 [4] to provide four types of prior knowledge for CaFo. Therein, CLIP [57] is pre-trained to produce paired features in the embedding space for every image and its descriptive text. Guided by texts with different categorical semantics, CLIP [57] can well classify the images aided by language-contrastive knowledge. DINO follows contrastive self-supervised learning [7] to match the representations between two transformations of one same image, which is expert at distinguishing different images with vision-contrastive knowledge. Similar to CLIP [57], DALL-E [58] is also pre-trained by image-text pairs but learns to predict the encoded image tokens based on the given text tokens. Conditioned on the input text, DALL-E [58] could leverage the vision-generative knowledge to create high-quality synthetic images in a zero-shot manner. Pre-trained by large-scale language corpus, GPT-3 [4] takes a few hand-written templates as input, and autoregressively generates human-like texts, which contain rich language-generative knowledge. Therefore, the four models have distinctive pre-training goals and can provide complementary knowledge to assist the few-shot visual recognition.

In detail, we cascade them by three steps: 1) Prompt. We adopt GPT-3 [4] to produce textual prompts for CLIP based on a few hand-written templates. These prompts with richer language knowledge are fed into CLIP’s textual encoder. 2) Generate. We adopt DALL-E [58] to generate additional training images for different categories based on the domain-specific semantics, which enlarges the few-shot training data, but costs no extra manpower for collection and annotation. 3) Cache. We utilize a cache model to adaptively incorporate the predictions from both CLIP [57] and DINO [7]. Referring to Tip-Adapter [76], we build the cache model with two keys of keys respectively for the two pre-trained models. Regarding zero-shot CLIP as the distribution baseline, we adaptively ensemble the predictions of two cached keys as the final output. By only fine-tuning the lightweight cache model via expanded training data, CaFo can learn to fuse diverse prior knowledge and leverage their complementary characteristics for better few-shot visual recognition.

Our main contributions are summarized as follows:

- We propose CaFo to incorporate the prior knowledge learned from various pre-training paradigms for better few-shot learning.
- By collaborating CLIP, DINO, GPT-3 and DALL-E, CaFo utilizes more semantic prompts, enriches the limited few-shot training data, and adaptively ensembles diverse predictions via the cache model.
- We conduct thorough experiments on 11 datasets for few-shot classification, where CaFo achieves state-of-the-art without using extra annotated data.

2. Related Work

Pre-training of Vision Models. With the breakthroughs in deep learning models [18, 35, 48], most modern vision models are based on the paradigm of pre-training on ImageNet [15] and fine-tuning on downstream tasks [34]. Pre-trained models have shown promising adaptability for various downstream tasks, such as object detection [46], semantic segmentation [9], and 3D recognition [8, 22, 23, 29, 77, 80, 81]. To improve the representation capability by overcoming the constraints of annotation, self-supervised pre-training has attracted wide attention using large-scale unlabeled datasets [42]. Self-supervised learning is initialized by pretext tasks, such as image restoration from corruption [25, 32, 55, 69], pseudo labels [17, 53] and clustering [5]. Recently, contrast learning, which learns representations by contrasting positive pairs against negative pairs, has gotten well studied for diverse visual representation learning [7, 10, 11, 28, 33, 67]. Besides, language-supervised visual pre-training emerges as a novel paradigm closer to natural visual understanding [2, 51, 59, 62], among which CLIP [57] obtains powerful zero-shot transferability by contrastive pre-training on image-text pairs. In addition, vision-language pre-training can also promote the zero-shot image generation from text. Open generative models, such as DALL-E [58] and CogView [16] pre-trained on large-scale image-text pairs are able to generate images with diverse contents by given texts. In this paper, CaFo cascades three visual pre-training models, CLIP, DINO [7], and DALL-E-mini [14], which contributes to better few-shot learning capacity.

Language-assisted Vision Models. As different form of data, linguistic knowledge normally contains complementary knowledge to images. For vision-language models, several works [4, 26, 57] have showed the format of prompts would highly affect the accuracy on vision tasks. Thus, prompt engineering is worth putting in great effort. Some efforts [39, 60, 79, 84] utilize learnable textual inputs and optimize them during training. Other works [50, 56] propose to leverage linguistic knowledge pre-trained from large language models to generate prompts for each visual category, which enhances vision-language models without any additional training or labeling. Our CaFo refers to CuPL [56] to produce semantic-rich texts by GPT-3 [4] to prompt CLIP for better text-image alignment.
Few-shot Learning. Few-shot learning highly relies on the transferability of the trained neural networks. From the perspective of distance measurement, some metric learning methods learn a metric space by computing the distances from the instances to new categories [63, 65, 70]. Also, meta-learning is proposed to improve the few-shot adaptation ability of the models by finding a set of initialized parameters that can rapidly adapt to novel domains [12, 21, 40, 45]. More recently, with the vision-language pre-training model CLIP [57] exhibiting strong zero-shot adaptation performance, several efforts have started to find efficient strategies to adapt it to downstream few-shot datasets. CoOp [83] is proposed as a prompt tuning adaptation method by optimizing a set of learnable prompt tokens. Subsequently, to inject textual branch with visual signals, CoCoOp [85] and VT-CLIP [79] propose to train an intermediate network to generate image tokens as conditional inputs for the textual vectors. Referring to adapters [38] in natural language processing, CLIP-Adapter [24] is introduced to fine-tune CLIP by applying lightweight residual-style adapters. Tip-Adapter [76] is then proposed as a training-free adaptation method with a constructed key-value cache model. It can also be regarded as a better initialization of CLIP-Adapter with much faster convergence when fine-tuning. CALIP [50] proposes a parameter-free attention to enhance CLIP in a zero-shot manner, and its parametric solution further attains higher few-shot accuracy. SuS-X [66] constructs a dynamic support set and extends Tip-Adapter by leveraging image-text distances. Besides, many follow-up works [39, 47, 66, 75, 78, 82, 86] have also been proposed for further adapting CLIP to various vision tasks. Different from all existing methods, we integrate other powerful pre-training paradigms with CLIP and collaborate them with customized pipelines.

3. Cascade of Foundation Models

In this section, we first briefly revisit four types of pre-training paradigms in CaFo. Then, we specifically introduce how we cascade them by ‘Prompt, Generate, then Cache’.

3.1. Different Pre-training Paradigms

Contrastive Vision-Language Pre-training. The series [10] of contrastive learning between vision and language learn to map the two modalities into the same embedding space via a contrastive loss. Driven by web-scale datasets, e.g., 400 million for CLIP [57] and 1.8 billion for ALIGN [41], the basic pre-training target is to minimize the embedding distances of images and their textual descriptions, while maximize those unpaired ones. By the cross-modal alignment, we can discriminate images of different categorizes by the texts with different semantics. We denote such learned prior as language-contrastive knowledge and adopt CLIP as the representative model for such pre-training method.

Contrastive Vision Pre-training. As the traditional self-supervised learning methods, vision-contrastive models [10] focus on the discrimination between different images. Normally, the positive pairs to be drawn close are two transformations of the same image, while the optimization of negative pairs [27] is optional, which can be replaced by a momentum encoder [33] or cluster assignments [6]. Recent works reveal that we can learn self-supervised features without negative pairs between images [7, 28]. Given the strong linear classification capacity, the pre-trained DINO [7] is adopted here to provide vision-contrastive knowledge for collaboration.

Generative Language Pre-training. With 175 billion parameters, the large-scale pre-trained GPT-3 [4] is powerful to produce human-like texts with diverse contents and incredible quality. Taking as input a few designed language commands, GPT-3 is able to output prompts with rich linguistic semantics for vision-language models. CLIP utilizes handcrafted templates as prompts, e.g., “a photo of a [CLASS]”, which however lacks sufficient textual semantics to align with input images. We thus leverage GPT-3 to produce CLIP’s prompts to better align with visual information from images.

Generative Vision-Language Pre-training. Learned from millions of image-caption pairs, the DALL-E series can generate language-conditioned images in a zero-shot manner. They are pre-trained to autoregressively predict the encoded image tokens from the textual tokens of the captions. With such language-generative knowledge, the pre-trained DALL-E can be viewed as a free lunch to enlarge the training data without any manpower. Considering publicity, we select DALL-E-mini [14] as the representative among DALL-E models.
3.2. Prompt, Generate, then Cache

To cascade different pre-training paradigms, we introduce CaFo with a pipeline of ‘Prompt, Generate, then Cache’, which respectively unleashes the powers of different self-supervised knowledge.

Prompt with GPT-3. Under the $N$-way $K$-shot settings, we have the few-shot training images $I_{N,K}$ with labels $L_{N,K}$ that contain $K$ samples for each $N$ categories. As shown in Figure 2 (1), for $N$ categories, we adopt a unified series of templates as the language command for GPT-3 [4], e.g., “What a [CLASS] looks like?”, “How can you identify a [CLASS]?” and “A caption of an image of a [CLASS]”. We denote the created prompts for $N$ categories as $P_N$, formulated as

$$P_N = \text{GPT-3}(\text{Commands}).$$

Then, we adopt $P_N$ as the input of CLIP’s textual encoder. Further, for some downstream data with specialized categories, we can customize the language commands for producing prompts with more domain-specific semantics. For example, in OxfordPets [54] dataset of pet images, we adopt the input of GPT-3 as “This is a pet bulldog, it has thin neck, short face, floppy ears. It’s coat is short, straight, and in brindle color. This is a pet [CLASS].”. Based on that, GPT-3 continues to describe more details of the [CLASS] pet.

Generate via DALL-E. Via the zero-shot DALL-E [14], we generate synthesis images to enrich our limited training images $I_{N,K}$, as shown in Figure 3 (2). For different categories, we adopt a simple template, e.g., “a photo of a [CLASS]”. After the generation, we utilize CLIP to filter the top-$K'$ best-quality images as the newly-expanded training samples for each category. Then, we obtain the $N$-category $(K + K')$-sample training images, formulated as

$$I_{N,(K+K')} = \{\text{DALL-E}(T_N), I_{N,K}\},$$

where $T_N$ denotes the $N$-category textual inputs. We keep $K'$ comparable with $K$ to ensure the synthesis quality and also preserve the low-data regimes. By the pre-trained language-generative knowledge, the data expansion is totally zero-shot, which requires no manpower to collect or annotate the data, and alleviates the data deficiency issue inherently for few-shot learning.

Cache by CLIP and DINO. We construct a key-value cache model for adaptive knowledge ensemble. Different from Tip-Adapter [76] only adapting CLIP, our cache model contains the pre-learned knowledge from both CLIP and DINO by caching two kinds of keys. Specifically in Figure 3 (3), we first utilize CLIP and DINO to independently extract visual features of the few-shot training images, formulated as

$$F_{\text{CLIP}} = \text{CLIP}_{\text{vis}}(I_{N,(K+K')});$$

$$F_{\text{DINO}} = \text{DINO}(I_{N,(K+K')}),$$

where CLIP$_{\text{vis}}$ denotes the CLIP’s visual encoder and $F_{\text{CLIP}}, F_{\text{DINO}} \in \mathbb{R}^{N(N+K') \times C}$. Besides the two keys, we convert the few-shot training labels into one-hot encodings $L_{\text{onehot}} \in \mathbb{R}^{N(N+K') \times N}$, and regard them as the same values for both keys. During training, we follow Tip-Adapter that only enables the cached keys in the adapter to be learnable and keeps the pre-trained models frozen.
3.3. Adaptive Inference

For a test image in Figure 4, we first extract its two visual features
\( f_{\text{CLIP}}, f_{\text{DINO}} \in \mathbb{R}^{1 \times C} \) and regard them as queries to retrieve diverse knowledge from the cache model. Then, we could acquire three predicted classification logits \( p_{\text{ZS}}, p_{\text{CLIP}}, p_{\text{DINO}} \in \mathbb{R}^{1 \times N} \), which are respectively from CLIP’s zero-shot alignment and the two keys of cache model. We formulate them as

\[
\begin{align*}
   p_{\text{ZS}} &= f_{\text{CLIP}} \cdot \text{CLIP}_{\text{tex}}(P_N)^T; \\
   p_{\text{CLIP}} &= \varphi(f_{\text{CLIP}}F_{\text{CLIP}}^T) \cdot \text{onehot}(i); \\
   p_{\text{DINO}} &= \varphi(f_{\text{DINO}}F_{\text{DINO}}^T) \cdot \text{onehot}(i),
\end{align*}
\]

where \( \text{CLIP}_{\text{tex}} \) represents CLIP’s textual encoder, \( P_N \) denotes GPT-3’s created prompts, and \( f_{\text{CLIP}}F_{\text{CLIP}}^T \) denotes the query-key affinity matrix of the CLIP’s keys, analogous to DINO’s. \( \varphi(x) = \exp(-\beta \cdot (1 - x)) \) serves as a non-linear modulator to control the sharpness of affinity matrix.

As the language-contrastive \( p_{\text{ZS}} \) is pre-trained by 400 million data and can perform strong zero-shot transfer ability, we regard \( p_{\text{ZS}} \) as the prediction baseline and calculate the weights of \( p_{\text{CLIP}}, p_{\text{DINO}} \) for ensemble based on their distribution similarity with \( p_{\text{ZS}} \). By this, we can suppress some obviously false category possibilities in \( p_{\text{CLIP}}, p_{\text{DINO}} \) and also amplify the moderately correct ones during ensemble. Firstly, we respectively normalize the scales of three classification logits into -1~1 by their each mean and standard deviation. We then calculate the distribution similarities as the ensemble weights for the two logits of the cache as

\[
\begin{align*}
   w_{\text{CLIP}} &= p_{\text{CLIP}} p_{\text{ZS}}^T, \quad w_{\text{DINO}} = p_{\text{DINO}} p_{\text{ZS}}^T.
\end{align*}
\]

Finally, we adopt the softmax function to normalize the weights and obtain the final ensemble logits as

\[
p_{\text{en}} = p_{\text{ZS}} + \sum_i p_i \cdot \text{softmax}(w_i),
\]

where \( i \in \{\text{CLIP}, \text{DINO}\} \). By such similarity-based ensemble, \( p_{\text{en}} \) can adaptively fuse the prior knowledge learned by CLIP and DINO’s pre-training and achieve stronger few-shot image classification.

4. Experiments

4.1. Settings

Datasets. We conduct few-shot experiments on 11 publicly available datasets: ImageNet [15], Standford-Dogs [43], UCF101 [64], Caltech101 [19], Flowers102 [52], SUN397 [72], DTD [13], EuroSAT [36], FGVCAir- craft [49], OxfordPets [54], and Food101 [3]. We follow Tip-Adapter [76] to train CaFo with 1, 2, 4, 8, 16 shots and test on the full test set. As we adopt DALL-E to generate training images in a zero-shot manner, we can train CaFo only by the generated images and report its zero-shot performance without few-shot training set.

Implementation. Our CaFo integrates the knowledge from pre-trained CLIP [57], DINO [7], DALL-E [58], and GPT-3 [4]. For CLIP, we utilize ResNet-50 [35] as the visual encoder and its aligned transformer as the textual encoder. To align with the visual representation from CLIP, we also adopt DINO pre-trained upon ResNet-50. For DALL-E, we adopt different domain-specific textual templates as the input for different datasets, which correspond to the original textual prompts for CLIP’s textual encoder. For GPT-3, we adopt five simple templates as the language commands shared by different categories. Each command outputs ten prompts, which obtains fifty prompts in total. For each category, we simply ensemble the features of different prompts following CuPL [4]. During training, we only set the two kinds of keys in cache model to be learnable and utilize the data augmentation following Tip-Adapter-F. We train CaFo using batch size 64 only for 20 epochs, and adopt AdamW optimizer with the initial learning rate.
Figure 5. **Performance (%) Comparison on ImageNet.** We compare CaFo with other methods for different few-shot settings.

<table>
<thead>
<tr>
<th>Models</th>
<th>Epochs</th>
<th>Time</th>
<th>Accuracy</th>
<th>Gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zero-shot CLIP</td>
<td>0</td>
<td>0</td>
<td>60.33</td>
<td>-</td>
</tr>
<tr>
<td>Zero-shot CALIP</td>
<td>0</td>
<td>0</td>
<td>60.57</td>
<td>-</td>
</tr>
<tr>
<td>Linear-probe CLIP</td>
<td>-</td>
<td>13min</td>
<td>56.13</td>
<td>-4.20</td>
</tr>
<tr>
<td>CoOp</td>
<td>200</td>
<td>14h 40min</td>
<td>62.95</td>
<td>+2.62</td>
</tr>
<tr>
<td>CLIP-Adapter</td>
<td>200</td>
<td>50min</td>
<td>63.59</td>
<td>+3.26</td>
</tr>
<tr>
<td>Tip-Adapter-F</td>
<td>20</td>
<td>5min</td>
<td>65.51</td>
<td>+5.18</td>
</tr>
<tr>
<td>CALIP-FS</td>
<td>200</td>
<td>1h</td>
<td>65.81</td>
<td>+5.48</td>
</tr>
<tr>
<td>CaFo</td>
<td>20</td>
<td>10min</td>
<td><strong>68.79</strong></td>
<td><strong>+8.46</strong></td>
</tr>
</tbody>
</table>

Table 1. **Efficiency Comparison on ImageNet.** We test the training time with a single A100 GPU under 16-shot setting.

0.0001 with a cosine scheduler. Note that, we tune the hyperparameters in CaFo by the official validation sets.

### 4.2. Performance

**On ImageNet.** We compare CaFo with other CLIP-based adaption methods on the most representative ImageNet [15]: CALIP [30], Linear-probe CLIP [57], CoOp [83], CLIP-Adapter [24], Tip-Adapter-F [76], and CALIP-FS [30]. All these methods are based on the pre-trained CLIP [57] with ResNet-50 visual encoders. As reported in Figure 5 and Table 2, CaFo surpasses all existing methods for different shot settings. Remarkably, CaFo with 1 shot even outperforms the 8-shot Linear-probe CLIP and CoOp, and CaFo with 8 shots is better than all methods with 16 shots. For zero-shot learning, CaFo significantly surpasses CLIP and CALIP, demonstrating the importance of DALL-E’s generation. In Table 1, we present the efficiency of CaFo concerning training epochs and time. Our CaFo achieves the best performance-efficiency trade-off with 68.79% accuracy and only 10 minutes training.

<table>
<thead>
<tr>
<th>Shot</th>
<th>Zero-shot CLIP</th>
<th>Zero-shot CALIP</th>
<th>Linear-probe CLIP</th>
<th>CoOp</th>
<th>CLIP-Adapter</th>
<th>VT-CLIP</th>
<th>Tip-Adapter-F</th>
<th>CALIP-FS</th>
<th>CaFo</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>60.33</td>
<td>60.57</td>
<td>-22.17 31.90 41.20 49.52 56.13</td>
<td>57.15 57.81 59.99 61.56 62.95</td>
<td>61.20 61.52 61.84 62.68 63.59</td>
<td>60.53 61.29 62.02 62.81 63.92</td>
<td>61.32 61.69 62.52 64.00 65.51</td>
<td>61.35 62.03 63.13 64.11 65.81</td>
<td>62.99 63.80 64.34 65.64 66.86 68.79</td>
</tr>
</tbody>
</table>

Table 2. **Quantitative Performance (%) Comparison on ImageNet.** For zero-shot performance, CaFo is trained with images generated by DALL-E without any few-shot data.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Source</th>
<th>Target</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zero-shot CLIP</td>
<td>60.33</td>
<td>53.27</td>
</tr>
<tr>
<td>Zero-shot CALIP</td>
<td>60.57</td>
<td>53.70</td>
</tr>
<tr>
<td>CoOp</td>
<td>62.95</td>
<td>54.58</td>
</tr>
<tr>
<td>CLIP-Adapter</td>
<td>63.59</td>
<td>55.69</td>
</tr>
<tr>
<td>CALIP-FS</td>
<td>65.81</td>
<td>55.98</td>
</tr>
<tr>
<td>Tip-Adapter-F</td>
<td>65.51</td>
<td>57.11</td>
</tr>
<tr>
<td>CaFo</td>
<td><strong>68.79</strong></td>
<td><strong>57.99</strong></td>
</tr>
</tbody>
</table>

Table 3. **Distribution Shift (%) Comparison.** We train the models on “Source” dataset and test on “Target” datasets.

**On Other Datasets.** To further assess the robustness in different scenarios, we test CaFo on extra 10 datasets in Figure 6. For different semantic domains including real-world scenes, detailed textures, and satellite-captured landscapes, CaFo consistently shows leading performance and indicates excellent robustness via the collaboration of diverse knowledge. Notably, on some datasets, e.g., Caltech101 and OxfordPets, the zero-shot CaFo perform even comparably to other methods with 4 shots, demonstrating the effectiveness of zero-shot DALL-E for few-shot data expansion.

**Distribution Shift.** We further evaluate the robustness of CaFo to distribution shift by training on “Source” dataset and testing on “Target” datasets. In Table 3, we select the “Source” as ImageNet and the “Target” as ImageNet-V2 [61] and ImageNet-Sketch [37]. As we can utilize some prior knowledge of the target domain for GPT-3 and DALL-E for prompting and generation, CaFo achieves the best out-of-distribution performance on the two “Target” datasets, surpassing the second-best Tip-Adapter-F by +3.28%, +0.88%, and +3.43%, respectively.
Figure 6. Performance (%) Comparison on 10 Datasets. Our method shows state-of-the-art performance for all few-shot settings on different datasets, which indicates superior generalization capacity.

Table 4. Ablation Study (%) of Cascaded Models. We ablate different pre-trained models on ImageNet with 1, 4, and 16 shots.

<table>
<thead>
<tr>
<th>Pre-trained Models</th>
<th>Shot</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLIP</td>
<td>DINO</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Table 5. Ablation Study (%) of Adaptive Inference. We conduct different ensemble methods of cache model on ImageNet.

<table>
<thead>
<tr>
<th>Method</th>
<th>Shot</th>
<th>1</th>
<th>2</th>
<th>4</th>
<th>8</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLIP</td>
<td>61.36</td>
<td>61.78</td>
<td>62.83</td>
<td>64.04</td>
<td>65.53</td>
<td></td>
</tr>
<tr>
<td>DINO</td>
<td>34.13</td>
<td>34.44</td>
<td>41.12</td>
<td>45.01</td>
<td>53.63</td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td>60.70</td>
<td>60.72</td>
<td>60.99</td>
<td>61.47</td>
<td>61.97</td>
<td></td>
</tr>
<tr>
<td>Maximum</td>
<td>61.64</td>
<td>62.45</td>
<td>62.95</td>
<td>63.60</td>
<td>64.97</td>
<td></td>
</tr>
<tr>
<td>$p_{\text{CLIP}}$ Base.</td>
<td>62.36</td>
<td>63.22</td>
<td>64.11</td>
<td>65.50</td>
<td>67.40</td>
<td></td>
</tr>
<tr>
<td>$p_{\text{DINO}}$ Base.</td>
<td>62.61</td>
<td>63.39</td>
<td>64.31</td>
<td>65.83</td>
<td>67.73</td>
<td></td>
</tr>
<tr>
<td>$p_{\text{DS}}$ Base.</td>
<td>63.80</td>
<td>64.34</td>
<td>65.64</td>
<td>66.86</td>
<td>68.79</td>
<td></td>
</tr>
</tbody>
</table>

4.3. Ablation study

Cascaded Models. In Table 4, we explore how each pre-trained model contributes to the collaboration on different shots of ImageNet. Therein, “CLIP” denotes the zero-shot CLIP with cache model containing only CLIP’s keys, and “DINO” denotes only the cache model with DINO’s keys. As shown in the first three rows, the CLIP’s language-contrastive knowledge performs stronger than DINO’s vision-contrastive knowledge, which might benefit from millions of pre-training data. Their adaptive ensemble by cache model can bring larger improvement when the shot number increases. For the next two rows, DALL-E and GPT-3 can independently boost both CLIP and DINO for nearly all shots with the prompts and generated synthetic images. The last row represents our final solution, CaFo that incorporates all three pre-trained models with the best performance for all shots.

Generated Number via DALL-E. We utilize DALL-E to generate synthetic images as the expanded few-shot training data. In Table 6, we explore the best synthetic number $K'$ for each category of different shots on ImageNet. We observe that the larger $K'$ does not lead to better few-shot performance. As we adopt pre-trained CLIP to select the top-$K'$ generated images, which are scored by the similarities between CLIP-encoded images and category texts, the larger $K'$ would contain more low-quality images and adversely affect the cache model. Furthermore, the amount of expanded data is comparable to the original $K$ shots and thus preserves the characteristic of few-shot learning.

Adaptive Inference. In Table 5, we ablate different ensemble methods of CLIP and DINO’s predictions during inference on ImageNet. The first two rows represent the cache model with one type of keys respectively for two pre-trained models without ensemble. Then, we adopt average and maximum pooling between the two predictions and ensemble the result with $p_{\text{DS}}$. However, such naive integration without adaptive weights causes accuracy degradation. In the last three rows, we calculate the distribution similarities for adaptive ensemble and respectively select the three logits as the baseline. As shown, using $p_{\text{DS}}$ as the distribution base-
Figure 7. Visualizations of DALL-E’s Generated Images. Examples are from ImageNet, OxfordPets and Caltech101 datasets.

<table>
<thead>
<tr>
<th>DALL-E</th>
<th>1</th>
<th>2</th>
<th>4</th>
<th>8</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>63.29</td>
<td>64.06</td>
<td>65.11</td>
<td>66.48</td>
<td>68.64</td>
</tr>
<tr>
<td>2</td>
<td>63.66</td>
<td>64.34</td>
<td>65.37</td>
<td>66.86</td>
<td>68.79</td>
</tr>
<tr>
<td>4</td>
<td>63.71</td>
<td>64.33</td>
<td>65.35</td>
<td>66.75</td>
<td>68.61</td>
</tr>
<tr>
<td>8</td>
<td><strong>63.80</strong></td>
<td>64.26</td>
<td><strong>65.64</strong></td>
<td>66.68</td>
<td>68.76</td>
</tr>
<tr>
<td>16</td>
<td>63.68</td>
<td>64.16</td>
<td>65.40</td>
<td>66.57</td>
<td>68.41</td>
</tr>
</tbody>
</table>

Table 6. Ablation Study (%) of Generated Number via DALL-E. We compare different shot numbers on ImageNet.

Table line performs the best, since $p_{ZS}$ itself shows strong transfer ability and can effectively suppress the wrong predictions of other logits.

4.4. Visualization

DALL-E’s Generated Images. In Figure 7, we visualize the synthetic images generated by DALL-E on ImageNet [15], OxfordPets [54] and Caltech101 [19] datasets. As shown, benefited from the vision-generative knowledge, the generated images are realistic and well highlight the downstream semantics of target category. This effectively expands the few-shot training set in low-data regimes for better network training.

GPT-3’s Prompts for CLIP. In Figure 8, We present a rectified example in ImageNet [15] dataset aided by GPT-3’s prompts in our CaFo. As shown, the prompting by GPT-3 (Left) produces texts with rich semantics compared to CLIP’s handcrafted templates (Right), and better depicts the visual appearances in the image. In this way, CaFo predicts the correct category of ‘Goldfish’, other than ‘Coral Reef’.

5. Conclusion

We propose CaFo, a cascade of foundation models that comprehends diverse knowledge from different pre-training and follows the ‘Prompt, Generate, then Cache’ pipeline. We first incorporate the generative language model, GPT-3, for prompting CLIP with more semantic texts, and adopt DALL-E to expand the few-shot training data. Then, we adaptively fuse DINO with CLIP via a unified cache model. By collaboration, CaFo achieves state-of-the-art performance for few-shot learning on 11 datasets. Although CaFo has unified four types of pre-training, our future direction will focus on integrating more existing pre-trained knowledge, e.g., the masked-generative MAE [32], the 3D-contrastive CrossPoint [1], and 3D-generative I2P-MAE [80].

Acknowledgement. This project is funded in part by National Key R&D Program of China Project 2022ZD0161100, by the Centre for Perceptual and Interactive Intelligence (CPII) Ltd under the Innovation and Technology Commission (ITC)’s InnoHK, by General Research Fund of Hong Kong RGC Project 14204021, by the National Natural Science Foundation of China (Grant No. 62206272), and by Shanghai Committee of Science and Technology (Grant No. 21DZ1100100). Hongsheng Li is a PI of CPII under the InnoHK.
References


[38] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duering. Scaling up visual and vision-language representation learning with noisy text supervision. In International Conference on Machine Learning, pages 4904–4916. PMLR, 2021. 3


[46] Bohao Li, Boyu Yang, Chang Liu, Feng Liu, Rongrong Ji, and Qixiang Ye. Beyond max-margin: Class margin equi-


earth mover’s distance and structured classifiers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12203–12213, 2020. 1


