Supporting Vision-language Model Few-shot Inference with Confounder-pruned Knowledge Prompt

Jiangmeng Li[a,1], Wenyi Mo[b,1], Fei Song[a,c,1], ✉Chuxiong Sun[a], Wenwen Qiang[a], Bing Su[b], Changwen Zheng[a,c]

[a]*National Key Laboratory of Space Integrated Information System, Institute of Software Chinese Academy of Sciences, Beijing, China.*
[b]*Renmin University of China, Beijing, China.*
[c]*University of Chinese Academy of Sciences, Beijing, China.*

Abstract

Vision-language models are pre-trained by aligning image-text pairs in a common space to deal with open-set visual concepts. Recent works adopt fixed or learnable prompts, i.e., classification weights are synthesized from natural language descriptions of task-relevant categories, to reduce the gap between tasks during the pre-training and inference phases. However, how and what prompts can improve inference performance remains unclear. In this paper, we explicitly clarify the importance of incorporating semantic information into prompts, while existing prompting methods generate prompts *without* sufficiently exploring the semantic information of textual labels. Manually constructing prompts with rich semantics requires domain expertise and is extremely time-consuming. To cope with this issue, we propose a knowledge-aware prompt learning method, namely **C**onfounder-**p**runed **K**nowledge **P**rompt (CPKP), which retrieves an ontology knowledge graph by treating the textual label as a query to extract task-relevant semantic information. CPKP further introduces a double-tier confounder-pruning procedure to refine the derived semantic information. Adhering to the individual causal effect principle, the graph-tier confounders are gradually identified and phased out. The feature-tier confounders are eliminated by following the maximum entropy principle in information theory. Empirically, the evaluations demonstrate the effectiveness of CPKP in few-shot inference, e.g., with only two shots, CPKP outperforms the manual-prompt method by 4.64% and the learnable-prompt method by 1.09% on average.

Keywords: Multi-modal model, Large-scale pre-training, Prompt learning, Maximum entropy, Knowledge graph

1. Introduction

As a promising and tractable surrogate for large-scale supervised visual representation learning methods, large-scale self-supervised vision-language pre-training methods, e.g., CLIP Radford et al. (2021) and ALIGN Jia et al. (2021), jointly learn image and text representations with two modality-specific encoders by aligning the corresponding image-text pairs, which is achieved by adopting contrastive loss in pre-training. Benefiting from pre-training on large-scale data, models learn numerous visual concepts so that the learned representations have a strong generalization and can be transferred to various tasks.

Zhou et al. (2022) observes that the zero-shot generalization performance of the pre-trained vision-language model heavily relies on the form of the text input. Feeding pure labels, i.e., textual names of categories, into the text encoder leads to degenerate performance. To tackle this issue, recent works adopt various prompts to augment the textual labels Zhou et al. (2022); Derakhshani et al. (2023); Khattak et al. (2023). In the inference stage, the classification weights, i.e., textual label features, are obtained by providing the text encoder with prompts describing candidate categories. The image feature generated by the image encoder is compared with these label features for classification.

Figure 1 summarizes the typical prompt generation paradigms. The paradigm in Figure 1 *a* rigidly applies the fixed prompt template, which suffers from a dilemma that a specific prompt template has inconsistent boosts for different tasks. A motivating example, proposed by Zhou et al. (2022), shows that using "a photo of a [Y]" as a prompt for CLIP achieves an accuracy of 60.86% on Flowers102 Nilsback et al. (2008), and using a more describing prompt, i.e., "a flower photo of a [Y]", can improve the performance to 65.81%, where "[Y]" presents the label text. However, such an improvement is reversed on Caltech101 Li et al. (2004), where the accuracy of using "a [Y]" is 82.68%, while using "a photo of [Y]" only achieves 80.81%.

To tackle this dilemma, several works Zhu et al. (2023); Zhou et al. (2022); Gao et al. (2021); Jin et al. (2022); Lin et al. (2023) explore learning prompts from *limited* downstream labeled data, e.g., few-shot scenarios, which is shown in Figure 1 *b*. Generally, these methods rely on empirical risk loss to optimize the learnable prompt, while both the meaning of the learned prompts and why they work remain unclear. We attribute this in part to the fact that the semantic information of text labels is not explicitly explored, and such a deficiency further degenerates the performance of visual-language models in few-shot scenarios. We argue that the label-related semantic information is critical for improving the performance of pre-trained models. To further confirm our hypothesis, we conduct a motivating comparison. Figure 2 demonstrates that both the semantic prompt

Email address: chuxiong2016@iscas.ac.cn (✉Chuxiong Sun)
[1]Equal contribution.

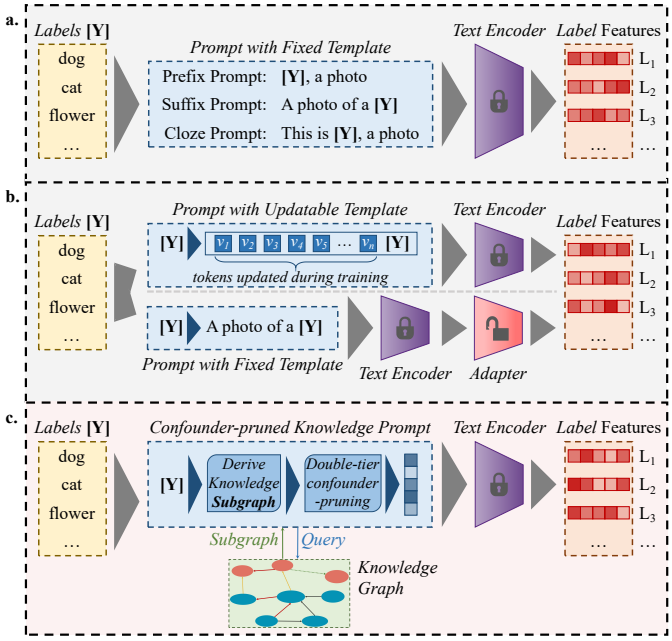

Figure 1: Comparison of different prompt generation paradigms. *a*: The paradigm of using the prompt with *fixed* templates Schick and Schütze (2021); Radford et al. (2021). *b*: The learning paradigms of recent benchmark works, including two major categories: the upper paradigm adopts a certain number of *updatable* tokens to generate adaptive prompts, and the tokens are learnable during training Zhou et al. (2022); Rao et al. (2021); the lower paradigm uses the same prompt with *fixed* templates as in *a*, but further injects an adapter after the fixed text encoder of the pre-trained vision-language model, and the adapter is trainable during inference on downstream tasks, including the adapter training and prediction Gao et al. (2021); Rao et al. (2021). *c*: The proposed learning paradigm, which directly *learn* a prompt from the labels by leveraging the *effective* semantic information from an ontology knowledge graph.

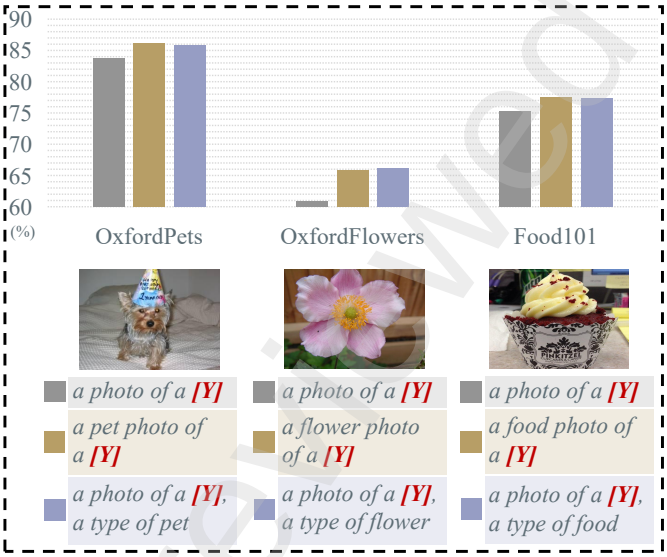

Figure 2: Comparison of different prompt forms for CLIP. The results of vision-language model inference experiments are shown in the histogram, where grey bars denote the prompt without semantic information which CLIP uses, brown bars denote the prompt with simple coarse-grained semantic information, and purple bars denote the prompt using more words to describe similar semantic information.

and a longer semantic prompt further improve the performance of CLIP. In contrast, the improvement of a longer semantic prompt over the semantic prompt is limited, which proves that the improvement of CLIP's performance relies on the addition of *semantic information* rather than simply adding *more words*.

To this end, we propose an innovative knowledge-aware prompt learning approach for improving few-shot inference of pre-trained vision-language models, namely, **C**onfounder-**p**runed **K**nowledge **P**rompt (CPKP). As illustrated in Figure 1 *c*, CPKP explores the semantic information associated with the label text by using labels as queries to retrieve an *ontology knowledge graph*. In practice, we observe that certain derived knowledge is redundant for downstream tasks, which may degenerate the performance of our method, e.g., specific relation types may negatively affect the prediction of the graph. The over-redundant information contained by the learned feature exacerbates acquiring discriminative information. Therefore, CPKP introduces a *double-tier confounder-pruning* procedure to refine the derived label-related knowledge representation. In graph-tier, inspired by the principle in benchmark works Granger (1969); Lin et al. (2021), CPKP gradually prunes the task-irrelevant relation types, which is treated as graph-level confounders[2]. In feature-tier, CPKP reduces feature-level confounders, i.e., the redundant information in features, by following the principle of *maximum entropy* Nakamura (2000); Liu et al. (2022a) in information theory. Empirically, the evaluations demonstrate that CPKP is effective for few-shot inference. The **contributions** of this paper are four-fold:

- We present a motivational study on the improvement of pre-trained visual-language models' few-shot inference in downstream applications through prompting learning methods, and identify the importance of exploring the semantic information of label text.
- For effectively mining semantic information from the label text, we propose a confounder-pruned knowledge prompt, which derives label-related semantic information by retrieving an ontology knowledge graph.
- We propose a double-tier confounder-pruning approach to remove task-redundant information from the label-related knowledge representation.
- Empirically, we conduct comprehensive comparisons to prove the superiority of our method in few-shot inference.

2. Related Work

Vision-Language Models. Recent development of joint learning on vision and language representations achieves impressive success in various fields Anderson et al. (2018); Antol et al. (2015); Gao et al. (2019); Kim et al. (2018); Huang et al. (2019); You et al. (2016). A critical issue is that few high-quality annotated multi-modal data are available. Therefore, Lu et al. (2019); Tan and Bansal (2019); Chen et al. (2019); Li et al. (2020) are designed to be pre-trained on massive unannotated data by taking

[2]The term *confounder* is used in its idiomatic sense, which is orthogonal to the existing statistical sense in Structural Causal Models Pearl (2009); Glymour et al. (2016) or other specific fields.

advantage of Transformer Vaswani et al. (2017). Such large-scale pre-trained vision-language models have great potential in the adaptation to various downstream tasks by learning universal representations via prompting Jia et al. (2021); Zhang et al. (2020). A representative approach is CLIP Radford et al. (2021), which pre-trains modality-specific encoders using 400 million image-text pairs and achieves impressive performance across a wide range of downstream tasks.

Prompt Design. Since directly applying pre-trained models to downstream tasks often leads to degenerate performance, template-based prompting methods Radford et al. (2021); Schick and Schütze (2021); Shin et al. (2020); Jiang et al. (2020) are proposed to address this issue. However, such template-based prompts have a critical issue: the optimal prompt may be excluded despite the large-scale candidate template library. To perform effective and data-efficient improvement on downstream tasks, simple yet effective adapter-based approaches Houlsby et al. (2019); Gao et al. (2021); Zhang et al. (2021) are introduced, which can be treated as a *post-model* prompt. Additionally, some methods Zhou et al. (2022); Rao et al. (2021); Derakhshani et al. (2023); Khattak et al. (2023) are developed to automatically learn prompts without relying on the template library. However, these approaches do not explore the *semantic information* of the label text during the inference stage. In this paper, we prove the importance of incorporating label-relevant semantic information into prompts and propose to derive such information by leveraging an ontology knowledge graph.

Knowledge Graph. Knowledge graphs include general domain knowledge graphs Vrandecic and Krötzsch (2014); Carlson et al. (2010); Xu et al. (2017); Speer et al. (2017) and specific domain knowledge graphs Harland (2012); Ferrucci et al. (2010); Tang et al. (2008). Specifically, ontology knowledge graphs Geng et al. (2021) only have the ontology entities, i.e., conceptual types, for instance, Wikidata-ZS and NELL-ZS Qin et al. (2020). To understand the graph-based information from knowledge graphs, Graph embedding Goyal and Ferrara (2018); Wang et al. (2017); Glorot et al. (2013); Socher et al. (2013) is proposed, which maps the high-dimensional graph data into the low-dimensional vector, e.g., TransE Bordes et al. (2013), TransR Lin et al. (2015), RESCAL Nickel et al. (2011), and KG-BERT Yao et al. (2019). Besides, Graph Neural Network (GNN)-based methods are proposed to mine graph structure information, e.g., KGCN Wang et al. (2019). For our approach, we refine the knowledge graph representation by eliminating the task-irrelevant and redundant information.

3. Preliminaries

Vision-Language Pre-training. CLIP Radford et al. (2021) introduces a pre-training approach to learning semantic knowledge from large amounts of image-text data and consists of an image encoder and a text encoder. Both encoders are trained jointly using a contrastive loss Chen et al. (2020). Suppose $\boldsymbol{h}$ denotes the image features extracted by the image encoder $f^I(\cdot)$ for an image $\boldsymbol{x}$ and $\{\boldsymbol{l}_i\}_{i=1}^K$ denotes a set of label features extracted by the text encoder $f^T(\cdot)$ from prompts $\{\boldsymbol{p}_i\}_{i=1}^K$ with a form of "a photo of a **[Y]**.", where K is the number of classes, and **[Y]** presents a specific class name, e.g., "dog", "cat", or "flower". The prediction probability is computed by

$$\mathcal{P}(y=i|\boldsymbol{h}) = \frac{\exp\left(\frac{<\boldsymbol{l}_i,\boldsymbol{h}>}{\tau}\right)}{\sum_{j=1}^{K}\exp\left(\frac{<\boldsymbol{l}_j,\boldsymbol{h}>}{\tau}\right)}, \tag{1}$$

where y denotes the semantically correct category for $\boldsymbol{x}$, τ is the temperature hyper-parameter in CLIP, and $<\cdot,\cdot>$ denotes the cosine similarity.

Graph Representation Learning. Let $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ be an attributed graph, where $\boldsymbol{V}$ is the node set and $\boldsymbol{E}$ is the edge set. Given a graph dataset $\mathcal{G} = \boldsymbol{G}_i, i \in [\![1, N^G]\!]$, where $\boldsymbol{G}_i$ is sampled *i.i.d* from the distribution $\mathcal{P}(\mathcal{G})$, and N^G represents the number of graphs in $\mathcal{G}$. The objective of graph representation learning is to learn an encoder $f^G(\cdot) : \mathcal{G} \rightarrow \mathbb{R}^{d^G}$, where $\mathbb{R}^{d^G}$ denotes a d^G-dimensional embedding space and $f^G(\boldsymbol{G}_i)$ is the representation of $\boldsymbol{G}_i$.

4. Methodology

The overall architecture of CPKP is illustrated in Figure 3[3]. CPKP consists of two stages: 1) ontology-enhanced knowledge embedding derives the label-related subgraph from an ontology knowledge graph by using the label token as a query; 2) double-tier confounder-pruning removes the *task-irrelevant* and *redundant* information from graph representations.

4.1. Learnable Prompt with Ontology-enhanced Knowledge Embedding.

We propose to retrieve an ontology knowledge graph by treating an input label as the query and further capture the corresponding high-order knowledge representation through a GNN. Given an input label $\boldsymbol{[Y]}_i$, we start by locating the 1-hop label-relevant subgraph $\boldsymbol{G}_i$, which is performed by obtaining the knowledge graph entity with the largest semantic similarity to $\boldsymbol{[Y]}_i$ and retrieving all neighbor entities that are directly connected to it by an edge.

Label-specific Prompt. From the experiments in Figure 2, we derive a common assumption for prompting pre-trained vision-language models:

Assumption 4.1. *(Semantic information in prompts). Introducing label-relevant semantic information in prompts boosts the performance of the pre-trained vision-language model in downstream inference tasks.*

Holding Assumption 4.1, we propose to effectively add label-relevant semantic information into learnable prompts. The label-specific prompt set $\{\boldsymbol{p}_i\}_{i=1}^K$ is generated by

$$\boldsymbol{p}_i = (\boldsymbol{\mu} + \lambda \cdot \varphi(\boldsymbol{[Y]}_i)) \oplus b(\boldsymbol{[Y]}_i), \tag{2}$$

where $\boldsymbol{\mu}$ is a set of learnable feature vectors, which are randomly initialized by Gaussian distributions. $\varphi(\cdot)$ denotes the function

[3]We are aware of the drawbacks of reusing notations. "i"s, used in $\boldsymbol{G}_i$ and $\{\boldsymbol{l}_i\}_{i=1}^K$, are two irrelevant indexes of random variables for simplicity.

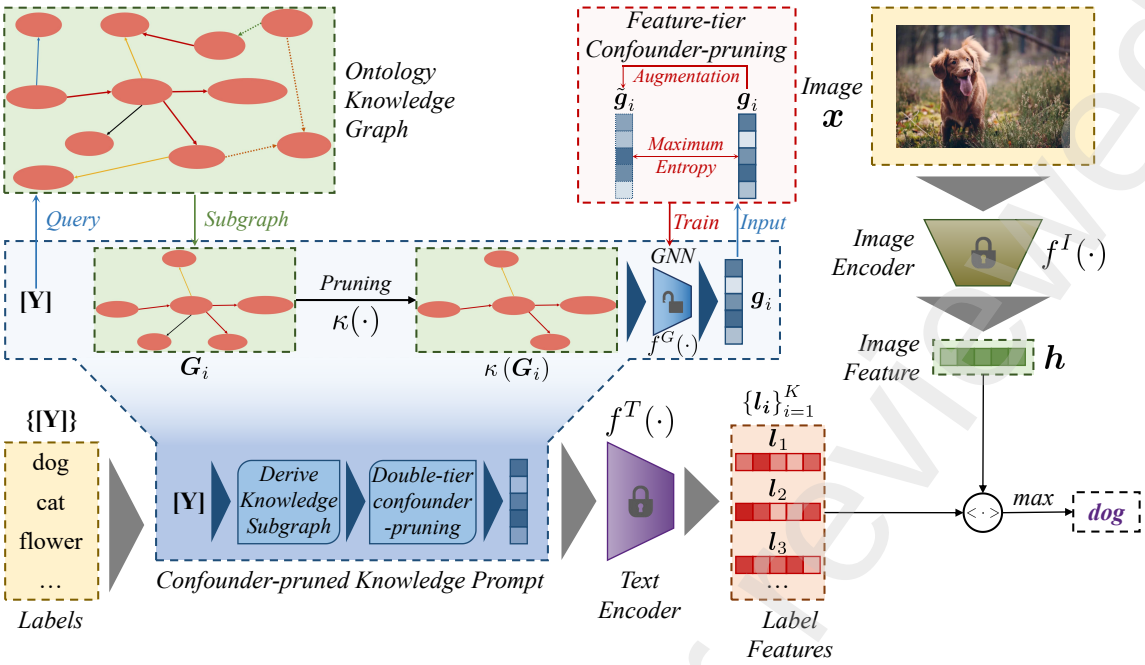

Figure 3: The architecture of CPKP. The intuition behind our method is to directly *learn* a prompt with label-related semantic information, which is achieved by introducing refined knowledge from an external knowledge graph.

of our proposed CPKP for encoding a label with rich semantic information, which will be detailed presented in Section 4.2, and λ is the coefficient that controls the balance between $\boldsymbol{\mu}$ and $\varphi(\cdot)$. $b([\boldsymbol{Y}]_i)$ denotes the lower-cased byte pair encoding representation of label $[\boldsymbol{Y}]_i$, and $\oplus$ is a concatenation function. Note that the output of $\varphi(\cdot)$ is a vector with the same dimension as $b([\boldsymbol{Y}]_i)$, e.g., 512 for CLIP. Feeding prompts $\{\boldsymbol{p}_i\}_{i=1}^K$ to the text encoder $f^T(\cdot)$, we obtain the label features $\{\boldsymbol{l}_i\}_{i=1}^K$, and the prediction probability is computed by Equation 1.

Label-shared Prompt. From the perspective of revisiting the training data for the vision-language model, we observe that the input text does not focus on describing the label-specific and discriminative semantic information; on the contrary, words with semantic information shared by different labels appear in a large body of descriptive text. For the examples "a [golden retriever] runs on the grass with its tail wagging" and "an [Alaskan] sits on a couch with a floppy tail", there only exists the label-shared information, i.e., "tail", but no label-specific information. Such a phenomenon is common in the description of fine-grained labels, and we thus hold an extended assumption:

Assumption 4.2. *(Generalized semantic information in prompts). Label-specific semantic information could be task-redundant to prompt pre-trained vision-language models, while generalized label-shared semantic information is crucial for generating effective prompts.*

We thus propose a label-shared prompt form by

$$\boldsymbol{p}_i = \left(\boldsymbol{\mu} + \lambda \cdot \psi\left(\left\lfloor \left\{\varphi\left([\boldsymbol{Y}]_j\right)\right\}_{j=1}^K \right\rceil\right)\right) \oplus b([\boldsymbol{Y}]_i), \tag{3}$$

where $\lfloor\cdot\rceil$ presents a cascade concatenation function, detailed by $\left\lfloor \left\{\varphi\left([\boldsymbol{Y}]_j\right)\right\}_{j=1}^K \right\rceil = \varphi([\boldsymbol{Y}]_1) \oplus \varphi([\boldsymbol{Y}]_2) \oplus \ldots \oplus \varphi([\boldsymbol{Y}]_K)$, and $\psi(\cdot)$ presents a linear mapping function in CPKP. We provide the interpretation of learned prompts and the case study of *label-shared* and *label-specific* prompts in Section 6.

4.2. Confounder-pruned Graph Representation

As mentioned previously, we employ a double-tier confounder-pruning procedure to achieve the desired task-relevant graph representation, which includes: 1) graph-tier confounder-pruning; and 2) feature-tier confounder-pruning.

Graph-tier Confounder-pruning. We refine the subgraph by performing the graph-tier confounder-pruning to remove the task-irrelevant information and encode the pruned subgraph into a vector by the function $\varphi(\cdot)$ in Equation 2 and Equation 3, which is defined by

$$\varphi([\boldsymbol{Y}]_i) = \boldsymbol{g}_i = f^G(\kappa(\boldsymbol{G}_i)), \tag{4}$$

where $\kappa(\cdot)$ denotes the proposed graph-tier confounder-pruning function, and $f^G(\cdot)$ is implemented by GNN.

We demonstrate confounder-pruned graph representation in Figure 4. Given the label-relevant knowledge subgraph $\boldsymbol{G}_i$ and the set of relation-types $\{\boldsymbol{r}_m\}_{m=1}^{N^R}$ in the subgraph, where N^R denotes the number of relation-types, we aim to remove the relation-types that are *decoupled* from the prediction of $\boldsymbol{G}_i$. To this end, we capture the individual causal effect Goldstein et al. (2015); Lin et al. (2021) of the knowledge subgraph $\boldsymbol{G}_i$ with the *relation-type* $\boldsymbol{r}_m$ on the label feature $\boldsymbol{l}_i$. *Graph Rule* denotes the process of quantitatively computing a score to ascertain whether the prediction of the graph is related to a specific pruned relation-type. Specifically, we quantify the contribution of a relation-type $\boldsymbol{r}_m$ to the prediction of the whole model, e.g., the output of the sequential model $f^G(\cdot)$ and $f^T(\cdot)$, by measuring the reduction in joint model error, formulated as

$$\Delta_{\epsilon,\boldsymbol{r}_m} = \epsilon_{\kappa^{(\neg r_m)}(\boldsymbol{G}_i)} - \epsilon_{\boldsymbol{G}_i}, \tag{5}$$

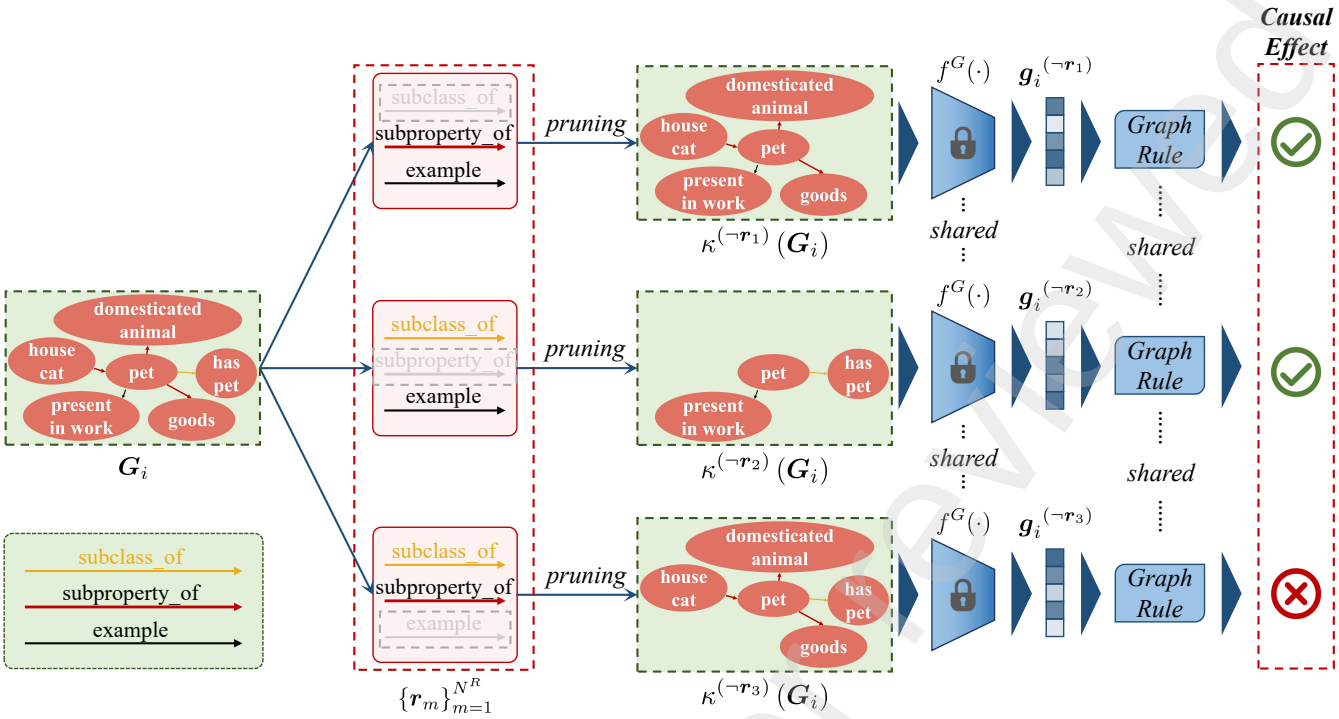

Figure 4: An example of the rationale of the graph-tier confounder-pruning for graph representations. We refine the derived knowledge subgraph $\boldsymbol{G}_i$ by pruning the edges that are causally decoupled from the downstream task. We determine whether a *relation-type* $\boldsymbol{r}_m$ is predictive of the graph by iteratively removing the edges related to the relation-type $\boldsymbol{r}_m$ and then checking the oscillation of the result, which is computed by following a specific graph rule. Only causally related edges are kept, and others are pruned. Note that the graph encoder $f^G(\cdot)$ is fixed throughout the process.

where $\epsilon_{\boldsymbol{G}_i}$ denotes the joint model error of the $f^G(\cdot)$ and $f^T(\cdot)$, i.e., the cross-entropy loss defined by $\mathcal{L}_{CE}(\{< \boldsymbol{l}_j, \boldsymbol{h} >\}_{j=1}^K, Y)$, and Y is the ground-truth label. $\epsilon_{\kappa^{(\neg r_m)}(\boldsymbol{G}_i)}$ denotes the joint model error excluding the relation-type $\boldsymbol{r}_m$.

Considering that the determination of whether a variable is predictive of the output requires sufficiently trained joint models, we adopt a truncated exponential moving weighted average approach, i.e., the exponential moving weighted average approach is adopted from the β-th epoch from the last epoch. Therefore, $\Delta_{\epsilon,\boldsymbol{r}_m}$ is transformed into $\bar{\Delta}_{\epsilon,\boldsymbol{r}_m}$ as follows:

$$\begin{aligned}\bar{\Delta}_{\epsilon,\boldsymbol{r}_m} =& \Big((1-\alpha)\cdot \Delta_{\epsilon,\boldsymbol{r}_m}^{N^t} + \alpha(1-\alpha)\cdot \Delta_{\epsilon,\boldsymbol{r}_m}^{N^t-1} + \ldots + \\ & \alpha^{\beta-1}(1-\alpha)\cdot \Delta_{\epsilon,\boldsymbol{r}_m}^{N^t-\beta+1}\Big) \Big/ \Big(1-\alpha^{\beta}\Big),\end{aligned} \tag{6}$$

where α is the balancing coefficient, and N^t denotes the total training epoch number.

We determine the relation between $\boldsymbol{r}_m$ and predicting the graph by

$$\begin{cases} \boldsymbol{r}_m \;\; is\ predictive\ of \;\; \Upsilon, & \bar{\Delta}_{\epsilon,\boldsymbol{r}_m} > 0 \\ \boldsymbol{r}_m \;\; is\ NOT\ predictive\ of \;\; \Upsilon, & \bar{\Delta}_{\epsilon,\boldsymbol{r}_m} \leq 0, \end{cases} \tag{7}$$

where Υ represents the classification based on the image representation $\boldsymbol{h}$ and the text representations $\{\boldsymbol{l}_j\}_{j=1}^K$.

$\bar{\Delta}_{\epsilon,\boldsymbol{r}_m}$ measures the contribution of a relation type $\boldsymbol{r}_m$. We prune those relation types with negative effects and visualize the process of confounder-pruning in Section 6. However, representations learned conventionally contain certain irremovable redundancy. To remedy this deficiency, we further introduce the feature-tier confounder-pruning technique.

Feature-tier Confounder-pruning. The feature-tier redundancy in the learned representations, dubbed the feature-tier confounder, explicitly degenerates the information entropy contained by representations, further resulting in the over-fitting and representation collapse issues. To cope with such problems, we propose to optimize the GNN to acquire the maximum entropy of representations, thereby explicitly reducing the risk of over-redundancy. Inspired by the data coding theory Liu et al. (2022b), we adopt a computationally tractable surrogate that measures the minimal coding length in lossy data coding to estimate the entropy. Specifically, suppose there exist a batch of K knowledge subgraph instances $\boldsymbol{G} = \{\boldsymbol{G}_i\}_{i=1}^K$ and the corresponding representations with D dimensions $f^G(\boldsymbol{G}) \in \mathbb{R}^{K\times D}$, the minimal coding length can be defined as Liu et al. (2022b); Yi et al. (2007):

$$MCL \triangleq \left(\frac{K+D}{2}\right)\log\det\left(\boldsymbol{I}_D + \frac{K}{D\epsilon^2} f^G(\boldsymbol{G})^\top f^G(\boldsymbol{G})\right), \tag{8}$$

where $\boldsymbol{I}_D$ presents a $D\times D$ identity matrix, and ϵ presents the distortion upper-bound of the encoding procedure. In practice, we conduct the Taylor series expansion to Equation 8 and derive

$$\begin{aligned} & MCL^{r\leq 2} \\ &= \mathrm{Tr}\left(\frac{K+D}{2}\left(\frac{K}{D\epsilon^2}\overline{f^G(\boldsymbol{G})}^\top \overline{f^G(\boldsymbol{G})} - \frac{1}{2}\left(\frac{K}{D\epsilon^2}\overline{f^G(\boldsymbol{G})}^\top \overline{f^G(\boldsymbol{G})}\right)^2\right)\right) \\ &= \frac{K+D}{2}\left(\sum_{i=1}^{D}\left(\boldsymbol{\xi}_{ii} - \frac{1}{2}\boldsymbol{\xi}_{ii}^2\right) - \frac{1}{2}\sum_{i=1}^{D}\sum_{j\neq i}^{D}\boldsymbol{\xi}_{ij}^2\right), \end{aligned} \tag{9}$$

and

$$\boldsymbol{\xi} = \frac{K}{D\epsilon^2}\overline{f^G(\boldsymbol{G})}^\top \overline{f^G(\boldsymbol{G})}, \tag{10}$$

Algorithm 1 CPKP(SPE) training

Input: The annotated image datasets X^{tr} for the training phase of few-shot learning. The corresponding label set $\left\{[\boldsymbol{Y}]_j\right\}_{j=1}^{K}$. Batch size n. Total training epoch number N^t. Coefficients λ, α, β and γ.
Initialize: The learnable neural network parameters θ for the graph encoder $f_\theta^G(\cdot)$ and ϑ for the learnable feature vectors $\boldsymbol{\mu}$, which share a learning rate ℓ. The fixed pre-trained parameters for the text encoder $f^T(\cdot)$ and the image encoder $f^I(\cdot)$.
repeat
  # *training phase of few − shot learning*
  for t-th training iteration **do**
    Sample a batch $\bar{X}^{tr}, \bar{Y}^{tr} = \{\boldsymbol{x}_i, \boldsymbol{y}_i\}_{i=(t-1)n}^{tn} \in X^{tr}$
    # *generate label features without pruning*
    $\left\{\boldsymbol{l}_j\right\}_{j=1}^{K} = f^T\left(\left(\boldsymbol{\mu}_\vartheta + \lambda \cdot f_\theta^G\left(\boldsymbol{G}_j\right)\right) \oplus b\left([\boldsymbol{Y}]_j\right)\right)$
    $\theta \leftarrow \theta - \ell \cdot \Delta_\theta(\mathcal{L}_{CE}(\left\{< \boldsymbol{l}_j, f^I\left(\bar{X}^{tr}\right) >\right\}_{j=1}^{K}, \bar{Y}^{tr}) + \gamma \mathcal{L}_{FTCP})$
    $\vartheta \leftarrow \vartheta - \ell \cdot \Delta_\vartheta(\mathcal{L}_{CE}(\left\{< \boldsymbol{l}_j, f^I\left(\bar{X}^{tr}\right) >\right\}_{j=1}^{K}, \bar{Y}^{tr}) + \gamma \mathcal{L}_{FTCP})$
    # *computing joint model error differentiation*
    if $t > (N^t - \beta)$ **then**
      # *fixing* $f^I(\cdot)$, $f^T(\cdot)$, *and* $f^G(\cdot)$
      for m-th relation type iteration **do**
        Init $\mathcal{L}_m^{\Delta^t} = 0$
        for $1 \le t' \le t$ **do**
          Sample $\bar{X}^{tr}, \bar{Y}^{tr} = \{\boldsymbol{x}_i, \boldsymbol{y}_i\}_{i=(t'-1)n}^{t'n} \in X^{tr}$
          # *cross − entropy loss without pruning*
          $\left\{\boldsymbol{l}_j\right\}_{j=1}^{K} = f^T\left(\left(\boldsymbol{\mu}_\vartheta + \lambda f_\theta^G\left(\boldsymbol{G}_j\right)\right) \oplus b\left([\boldsymbol{Y}]_j\right)\right)$
          $\mathcal{L}_m = \mathcal{L}_{CE}(\left\{< \boldsymbol{l}_j, f^I\left(\bar{X}^{tr}\right) >\right\}_{j=1}^{K}, \bar{Y}^{tr})$
          # *cross − entropy loss with pruning*
          # *remove* $\boldsymbol{r}_m$ *and derive* $\kappa^{(\neg \boldsymbol{r}_m)}\left(\boldsymbol{G}_j\right)$
          $\left\{\boldsymbol{l}_j\right\}_{j=1}^{K} = f^T\left(\left(\boldsymbol{\mu}_\vartheta + \lambda f_\theta^G(\kappa^{(\neg \boldsymbol{r}_m)}(\boldsymbol{G}_j))\right) \oplus b([\boldsymbol{Y}]_j)\right)$
          $\mathcal{L}_m^{(\neg \boldsymbol{r}_m)} = \mathcal{L}_{CE}(\left\{< \boldsymbol{l}_j, f^I\left(\bar{X}^{tr}\right) >\right\}_{j=1}^{K}, \bar{Y}^{tr})$
          $\mathcal{L}_m^{\Delta^t} += \mathcal{L}_m^{(\neg \boldsymbol{r}_m)} - \mathcal{L}_m$
        end for
      end for
      $\bar{\mathcal{L}}_m^{\Delta^t} = \left((1-\alpha) \cdot \mathcal{L}_m^{\Delta^t} + \alpha(1-\alpha) \cdot \mathcal{L}_m^{\Delta^{t-1}} + \ldots + \alpha^{t-N^t+\beta-1}(1-\alpha) \cdot \mathcal{L}_m^{\Delta^{N^t-\beta+1}}\right) \Big/ \left(1 - \alpha^{t-N^t+\beta}\right)$
    end if
  end for
until θ and ϑ converge
confirming relation types
Confirm the relation between the graph prediction and $\boldsymbol{r}_m$ by considering $\bar{\mathcal{L}}_m^{\Delta^{N^t}}$

Algorithm 2 CPKP(SPE) test

Input: The annotated image datasets X^{te} for the test phase of few-shot learning. The corresponding label set $\left\{[\boldsymbol{Y}]_j\right\}_{j=1}^{K}$. Batch size n. Coefficient λ.
Initialize: The learned neural network parameter θ for the graph encoder $f_\theta^G(\cdot)$. The fixed pre-trained parameters for the text encoder $f^T(\cdot)$ and the image encoder $f^I(\cdot)$.
# *test phase of few − shot learning*
for t-th test iteration **do**
  Sample a batch $\bar{X}^{te}, \bar{Y}^{te} = \{\boldsymbol{x}_i, \boldsymbol{y}_i\}_{i=(t-1)n}^{tn} \in X^{te}$
  # *generate label features with*
  # *graph − tier confounder − pruning*
  # *perform confounder − pruning and derive* $\kappa\left(\boldsymbol{G}_j\right)$
  $\left\{\boldsymbol{l}_j\right\}_{j=1}^{K} = f^T\left(\left(\boldsymbol{\mu}_\vartheta + \lambda \cdot f_\theta^G\left(\kappa\left(\boldsymbol{G}_j\right)\right)\right) \oplus b\left([\boldsymbol{Y}]_j\right)\right)$
  $Y^{predict} = \max_j \left\{< \boldsymbol{l}_j, f^I\left(\bar{X}^{te}\right) >\right\}_{j=1}^{K}$
end for

where $\mathrm{Tr}(\cdot)$ denotes the trace of a matrix, r denotes the order of the expanded Taylor series, and $\overline{f^G(\boldsymbol{G})}$ represents the dimensional normalized representations. In practice, we propose to compute $\boldsymbol{\xi}$ in a double-view manner to provide a sufficient regularization of representations, and Equation 10 is re-written by

$$\boldsymbol{\xi} = \frac{K}{D\epsilon^2} \overline{f^G(\boldsymbol{G})}^\top \overline{f^G(\boldsymbol{G})'}, \tag{11}$$

where $f^G(\boldsymbol{G})'$ denotes representations of another view, which is generated by combining distortions with $f^G(\boldsymbol{G})$. Under sufficient optimization, considering $\frac{K}{D\epsilon^2}$ is a constant, we derive that the trace of $\boldsymbol{\xi}$ is constant, and each element of $\{\boldsymbol{\xi}_{ii}\}_{i=1}^{D}$ is a constant value.

To acquire the objective of maximizing the minimum coding length, we simplify Equation 9 by eliminating the constant values $\{\boldsymbol{\xi}_{ii}\}_{i=1}^{D}$ and coefficients, i.e., K and D. The well-refined loss function of the feature-tier confounder-pruning is implemented as follows:

$$\mathcal{L}_{FTCP} = \sum_{i=1}^{D} \sum_{j \ne i}^{D} {\boldsymbol{\xi}_{ij}}^2. \tag{12}$$

Our complete method is called CPKP [4]. Considering two forms of prompts that are discussed in Section 4.1, we abbreviate label-specific prompt and label-shared prompt as SPE and SHR, respectively. We take CPKP(SPE) as an example to demonstrate the pipeline in Algorithm 1 and Algorithm 2.

5. Experiments

Datasets. The experiments are conducted on 11 publicly available image classification datasets: ImageNet Deng et al. (2009), Caltech101 Li et al. (2004), StandfordCars Krause et al. (2013), FGVCAircraft Maji et al. (2013), Flowers102 Nilsback et al. (2008), OxfordPets Parkhi et al. (2012), Food101 Bossard et al. (2014), SUN397 Xiao et al. (2010), UCF101 Soomro et al. (2012), DTD Cimpoi et al. (2014), and EuroSAT Helber et al. (2019). We collect the details of datasets in Table 1. Note that for Caltech101, the "BACKGROUND Google" and "Faces easy"

[4] Unless otherwise specified, CPKP refers to CPKP using the label-shared prompt, i.e., CPKP(SHR).

Table 1: The details of benchmark datasets for few-shot learning experiments.

Datasets	Classes	Train	Val	Test
ImageNet	1,000	1.28M	N/A	50,000
Caltech101	100	4,128	1,649	2,465
OxfordPets	37	2,944	736	3,669
StanfordCars	196	6,509	1,635	8,041
Flowers102	102	4,093	1,633	2,463
Food101	101	50,500	20,200	30,300
FGVCAircraft	100	3,334	3,333	3,333
SUN397	397	15,880	3,970	19,850
DTD	47	2,820	1,128	1,692
EuroSAT	10	13,500	5,400	8,100
UCF101	101	7,639	1,898	3,783

Table 2: Statistics of the adopted ontology knowledge graphs. *# Ent.* denotes the number of entities. *# Triples* denotes the amount of relation triples. *# Train/Dev/Test* denotes the number of relations for training/validation/test.

Dataset	# Ent.	# Triples	# Train/Dev/Test
Nell-ZS	1,186	3,055	139/10/32
Wikidata-ZS	3,491	10,399	469/20/48

classes are discarded. For the video dataset, UCF101, the middle frame of each video is used as input to the image encoder. These datasets cover general object classification tasks, scene recognition tasks, action recognition tasks, fine-grained classification tasks, and specialized tasks such as texture recognition and satellite image recognition, which constitute a comprehensive benchmark.

Baselines. We compare our approach with three major baseline models: 1) CLIP Radford et al. (2021), which is based on manual prompts, and we follow the instructions for prompt ensembling in Radford et al. (2021) and input seven corresponding prompt templates into the CLIP text encoder; 2) CLIP using linear probing Radford et al. (2021), which is implemented by following Radford et al. (2021); Tian et al. (2020), and we train a linear classifier on top of high-quality pre-trained CLIP's features; 3) CoOp Zhou et al. (2022), which automatically designs the prompt templates, and for fair comparisons, we adopt the best variants of CoOp.

Ontology Knowledge Graphs. We provide detailed descriptions of the candidate knowledge graphs in Table 2. Nell-ZS is constructed based on NELL Carlson et al. (2010), while Wikidata-ZS is based on Wikidata Vrandecic and Krötzsch (2014). Both Nell and Wikidata are well-configured, large-scale knowledge graphs with official relation descriptions, and the textual descriptions of Nell-ZS and Wikidata-ZS contain abundant information.

Training Details. We set the maximum epoch to 200, 100, and 50 for 16/8 shots, 4/2 shots, and 1 shot, respectively, while the maximum epoch on ImageNet is fixed to 50 for all shots. Unless otherwise specified, ResNet-50 He et al. (2016), and Transformer Vaswani et al. (2017) are used as the corresponding image and text encoders. We initially adopt Wikidata-ZS Qin et al. (2020) as the target ontology knowledge graph, while we also

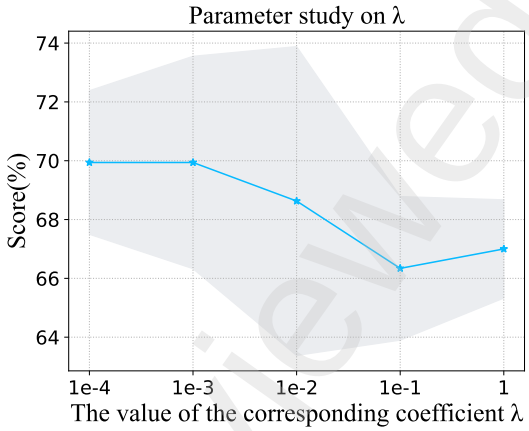

Figure 5: Parameter study on λ. We choose the best coefficient value of λ as 10^{-3} in benchmark experiments. The shade denotes the range of experimental results.

conduct experiments to evaluate our method using Nell-ZS Qin et al. (2020) in Section 5. We randomly sample half of the training data to confirm the graph prediction-related relation types. The set of learnable feature vectors $\boldsymbol{\mu}$ is randomly initialized by zero-mean Gaussian distributions with a standard deviation of 0.02. We set $\alpha = 0.8$, $\beta = 5.0$ and $\gamma = 1.0$. Figure 5 reports the results of the model with different λ values based on Flowers102 at 1 shot. The parameter study is conducted on the validation set. To explore the influence of λ, we fix other experimental settings and select λ from the range of $\{10^{-4}, 10^{-3}, 10^{-2}, 0.1, 1\}$. We can observe that the score reaches the maximum when the λ is 10^{-3}, indicating that an appropriate tuning of the impact of the knowledge embedding to guide the training of learnable label features, i.e., $\boldsymbol{\mu}$, can indeed promote the performance of CLIP on downstream tasks. However, overemphasizing the impact of knowledge embedding on training may degenerate the ability of the learnable features $\boldsymbol{\mu}$ to fit appropriate prompts needed for downstream tasks by using gradient back-propagation, so that the performance of CLIP is weakened. The setting of λ is shared among different downstream tasks. Table 3 shows the settings of λ for different datasets.

Few-shot Inference. We train our model with 1, 2, 4, 8, and 16 shots respectively, and evaluate it on test sets. The experimental results on 11 benchmark datasets are demonstrated in Figure 6, and the average results over three runs are shown in the top-left subfigure. We observe that CPKP achieves superior results under settings of different shots. With the increase of shots, each compared method achieves better performance, while CPKP still outperforms benchmark methods. Table 4 shows the performance gap between CPKP using 16 shots and CLIP using different manual prompts on all datasets. The results demonstrate that CPKP outperforms the best CLIP variant by a significant margin on most datasets, further proving our proposed Assumption 4.1 and the effectiveness of CPKP. Table 5 demonstrates the improvements achieved by the knowledge-aware learnable prompt (klp) over the conventional learnable prompt (clp). Specifically, the improvements reach 2.58%, 1.68%, and 1.90% on fine-grained image classification datasets, including DTD, Flowers102, and Food101, respectively.

Ontology-enhanced Knowledge Prompt. Figure 7 reports

Table 3: The Settings of λ for different datasets.

	Pets	Flowers	Aircraft	DTD	EuroSAT	Cars	Food	SUN	Cal	UCF	IN
λ	1e-1	1e-3	1e-2	1e-2	1e-3	1e-3	1e-1	1e-1	1e-1	1e-3	1e-1

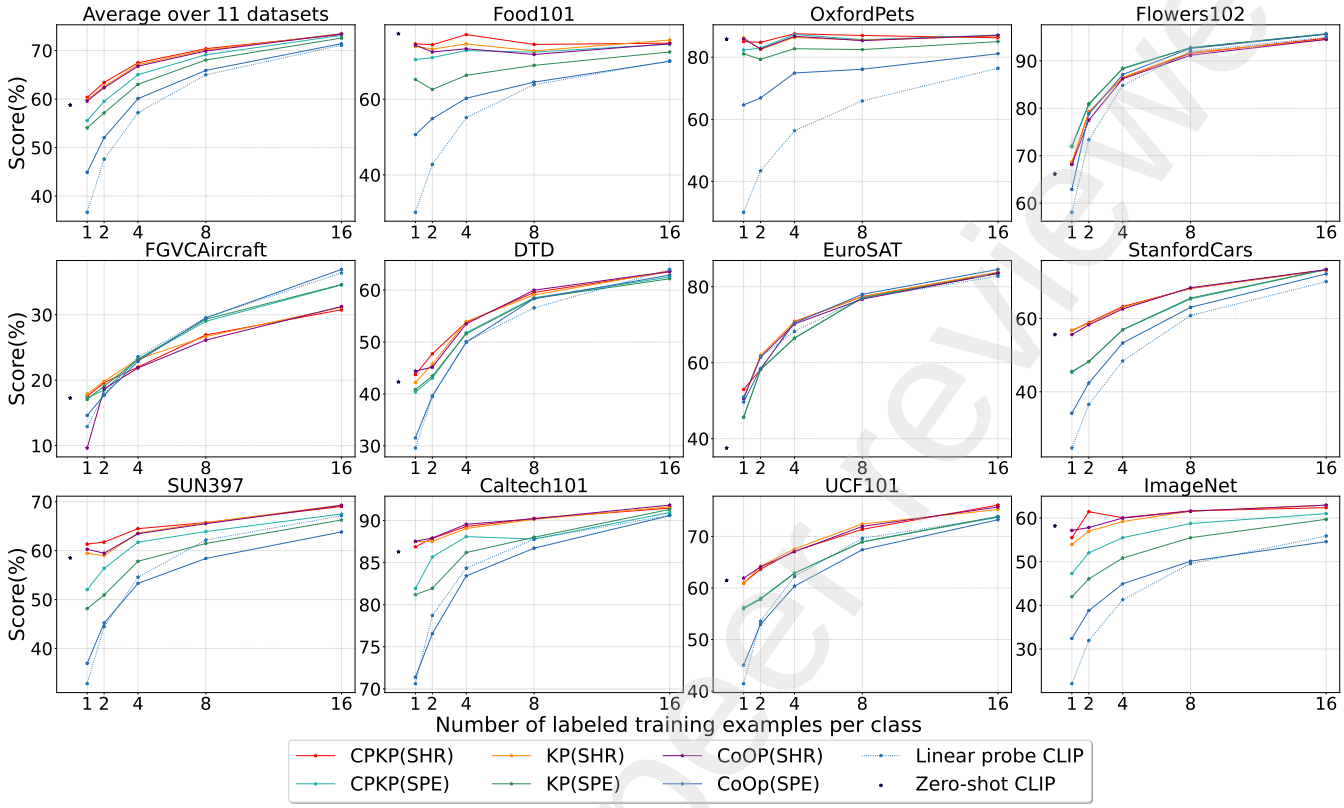

Figure 6: The performance of few-shot inference on 11 datasets. KP represents the variant of CPKP using no pruning.

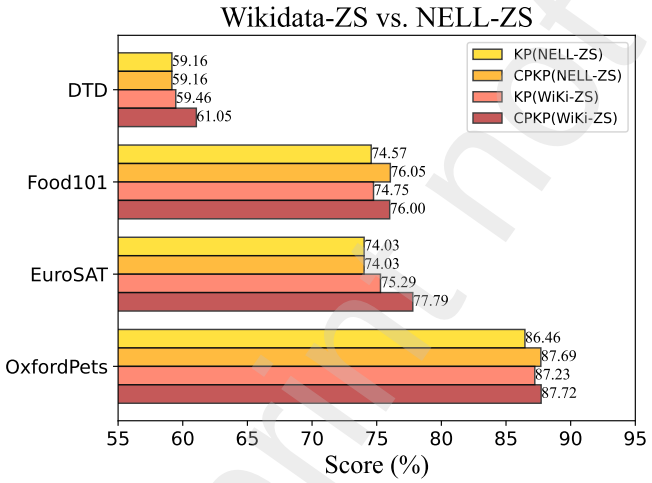

Figure 7: Comparison of leveraging two different knowledge graphs, i.e., Nell-ZS and Wikidata-ZS.

the results of the model trained on four datasets with 8 shots using Wikidata-ZS or NELL-ZS ontology knowledge graph. We observe that CPKP and the variant KP achieve better performance when using the Wikidata-ZS knowledge graph compared to using NELL-ZS. We reckon the reason is that Wikidata-ZS has more detailed relations and entities, empowering our method to locate label-specific knowledge subgraphs, which is consistent with our proposed Assumption 4.1. However, we also observe that the difference between the performance of CPKP using Wikidata-ZS and using Nell-ZS is not extremely large on some benchmark datasets, e.g., Food101 and OxfordPets. According to Assumption 4.2, we speculate that although Nell-ZS lacks enough label-specific knowledge, it contains sufficient *generalized* label-related knowledge for certain datasets. For example, Nell-ZS does not include entities such as "chocolate" and "potato", but it contains "concept:food", enabling the knowledge subgraph of "concept:food" can be used for many labels. This further demonstrates that the important content of prompts may *not* contain label-specific and discriminative information, and generalized label-shared semantic information is crucial for generating effective prompts.

Graph-tier Confounder-pruning. To support the superiority of the proposed graph-tier confounder-pruning, we compare the CPKP using label-specific prompt with CPKP using random pruning, as shown in Table 6. Specifically, the corresponding last two columns are the performance gaps between the proposed methods and CPKP using random pruning instead of the proposed graph-tier confounder-pruning, e.g., the accuracy achieved by CPKP using random pruning minus the accuracy achieved by our proposed methods. We observe that the complete CPKP outperforms CPKP using random pruning on all downstream tasks, and KP can even outperform CPKP using random pruning on most downstream tasks. We reckon that the random pruning

Table 4: Comparison with hand-crafted prompts. "Δ" denotes the performance gap between CPKP and the best CLIP model using the coarse-grained semantic prompt.

Datasets	Hand-crafted Prompts		CPKP	Δ
OxfordFlowers	a photo of a [Y]	60.86	94.76	**+28.62**
	a flower photo of a [Y]	65.85		
	a photo of a [Y], a type of flower	66.14		
FGVCAircraft	a photo of a [Y]	15.72	30.76	**+13.48**
	an aircraft photo of a [Y]	16.65		
	a photo of a [Y], a type of aircraft	17.28		
OxfordPets	a photo of a [Y]	83.73	86.23	**+0.02**
	a pet photo of a [Y]	86.21		
	a photo of a [Y], a type of pet	85.77		
DTD	a photo of a [Y]	39.83	63.53	**+21.10**
	a photo of a [Y] texture	40.25		
	[Y] texture	42.43		
EuroSAT	a photo of a [Y]	24.12	83.67	**+46.29**
	a satellite photo of [Y]	37.38		
	a photo of a [Y], a type of sate	31.41		
StanfordCars	a photo of a [Y]	55.61	73.27	**+17.28**
	a car photo of a [Y]	55.86		
	a photo of a [Y], a type of car	55.99		
Food101	a photo of a [Y]	75.20	74.82	-2.68
	a food photo of [Y]	77.50		
	a photo of [Y], a type of food	77.31		
ImageNet	a photo of a [Y]	58.19	62.36	**+4.07**
	an object photo of a [Y]	57.99		
	a photo of a [Y], a type of object	58.29		
Caltech101	a photo of a [Y]	86.25	91.47	**+4.13**
	an object photo of a [Y]	85.92		
	a photo of a [Y], a type of object	87.34		
SUN397	a photo of a [Y]	58.49	69.05	**+8.35**
	a scene photo of a [Y]	60.70		
	a photo of a [Y], a type of scene	60.48		
UCF101	a photo of a [Y]	58.37	76.08	**+14.59**
	an action photo of a [Y]	61.49		
	a photo of a [Y], a type of action	60.40		

may incorrectly remove the task-relevant semantic relations in the acquired knowledge graph, and some relations containing task-irrelevant noise information could be preserved. Therefore, the performance of CPKP using random pruning could be degraded.

Feature-tier Confounder-pruning. We visualize the representations learned by CPKP variants on the ImageNet dataset in Figure 8. CPKP w/ FTCP indicates CPKP with feature-tier confounder-pruning, while CPKP w/o FTCP indicates CPKP without feature-tier confounder-pruning. Concretely, the learned prompt feature representations are projected into an RGB-styled color image. Different colors indicate different types of information in features. The abscissa axis denotes the feature dimensions, and the ordinate axis presents various categories. The more different colors represent the less similar feature dimensions. The two left plots illustrate the contributions of dimensions to a specific category classification, and the right plots demonstrate the similarities between feature dimensions. These visualizations substantiate the superiority of the feature-tier confounder-pruning technique in addressing the feature redundancy issue.

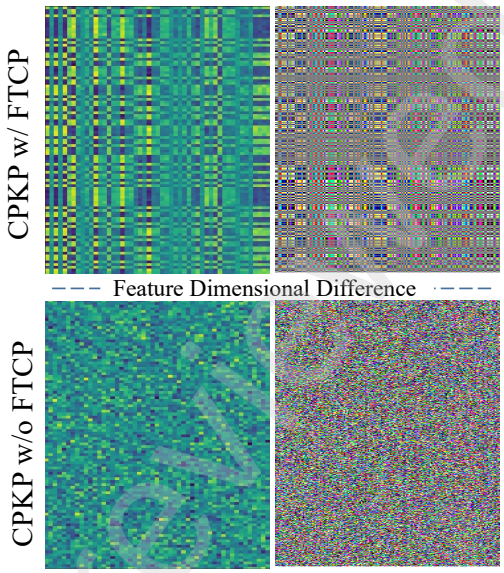

Figure 8: The visualization of representations learned by variants of CPKP in ImageNet.

6. Further Analysis

Interpreting the Learned Prompts. Figure 9 shows that the original input text of CLIP indeed contains several words with rich semantic information. Such a fact proves that our proposed assumptions are reliable. We interpret the learned prompt by transforming the learned feature vector into the word closest to the corresponding vector in the hidden space. Table 7 shows the visualized feature vectors of $\boldsymbol{\mu}$ learned by CPKP on benchmark datasets. We observe that there exist words that are task-relevant, e.g., "cat", "cag" and "furry" for OxfordPets, "ford" and "electr" for StanfordCar. From the experimental results demonstrated in Table 8, we observe that the vectors learned by CoOp Zhou et al. (2022) are basically ambiguous words, such as, "mul", "leng", "vish", "traveled", "check", "c", "darwin", ":]", "un", "ldnt", ""/", and "@", etc. This substantiates that CoOp can hardly learn task-relevant lexical features, since its training is only based on the gradient back-propagation without sufficiently exploring the *task-relevant semantic* information. Concretely, our proposed CPKP empowers the model to learn task-relevant feature vectors with rich semantic information.

Case Study of Label-shared Prompt and Label-specific Prompt. We describe the label-shared and label-specific information in Section 4.1. Specifically, label-shared information denotes the information shared by all alternative categories; for instance, for the classification of dogs and cats, the semantic information shared by all categories is that they are animals. Label-specific information denotes the information only belonging to a specific category; for instance, for the classification of dogs and cats, cats are felines and quiet, while dogs are canines and active.

As shown in Table 9, we demonstrate the examples of label-shared semantic information or label-specific semantic information in the ontology knowledge graph WIKI-ZS. Each row

Table 5: Comparison between the conventional learnable prompt (abbreviated as "clp") and the knowledge-aware learnable prompt (abbreviated as "klp") under the setting of 2 shots. △ denotes the gain of "klp" over "clp".

	OxfordPets	Flowers102	FGVCAircraft	DTD	EuroSAT	StanfordCars	Food101	SUN397	Caltech101	UCF101	ImageNet	Average
CLIP + clp	82.64	77.51	18.68	45.15	61.50	58.28	72.49	59.48	87.93	64.09	57.81	62.32
CLIP + klp	84.76	79.19	19.62	47.73	58.24	58.88	74.39	61.75	87.88	63.65	61.44	63.41
△	+2.12	+1.68	+0.94	+2.58	-3.26	+0.60	+1.90	+2.27	-0.05	-0.44	+3.63	+1.09

Table 6: The performance achieved by CPKP using random pruning.

Example num	Datasets											Avg	$\triangle_{KP}$	$\triangle_{CPKP}$
	Pets	Flowers	Aircraft	DTD	EuroSAT	Cars	Food	SUN	Cal	UCF	IN			
1-shot	81.22	71.88	17.07	40.92	45.61	45.59	65.65	48.10	81.45	56.00	42.32	54.16	+0.11	-1.38
2-shots	79.35	80.97	19.05	43.20	58.23	48.22	63.37	50.28	81.87	57.91	46.11	57.14	-0.00	-2.37
4-shots	82.73	88.32	23.25	51.46	66.46	56.97	67.03	56.36	85.53	62.75	50.65	62.86	-0.17	-2.15
8-shots	82.43	92.77	29.36	58.35	77.10	65.63	68.69	60.44	87.62	68.94	54.73	67.82	-0.21	-1.29
16-shots	84.99	95.71	34.47	62.17	83.52	73.40	71.64	64.82	91.22	73.88	58.95	72.25	-0.30	-0.87

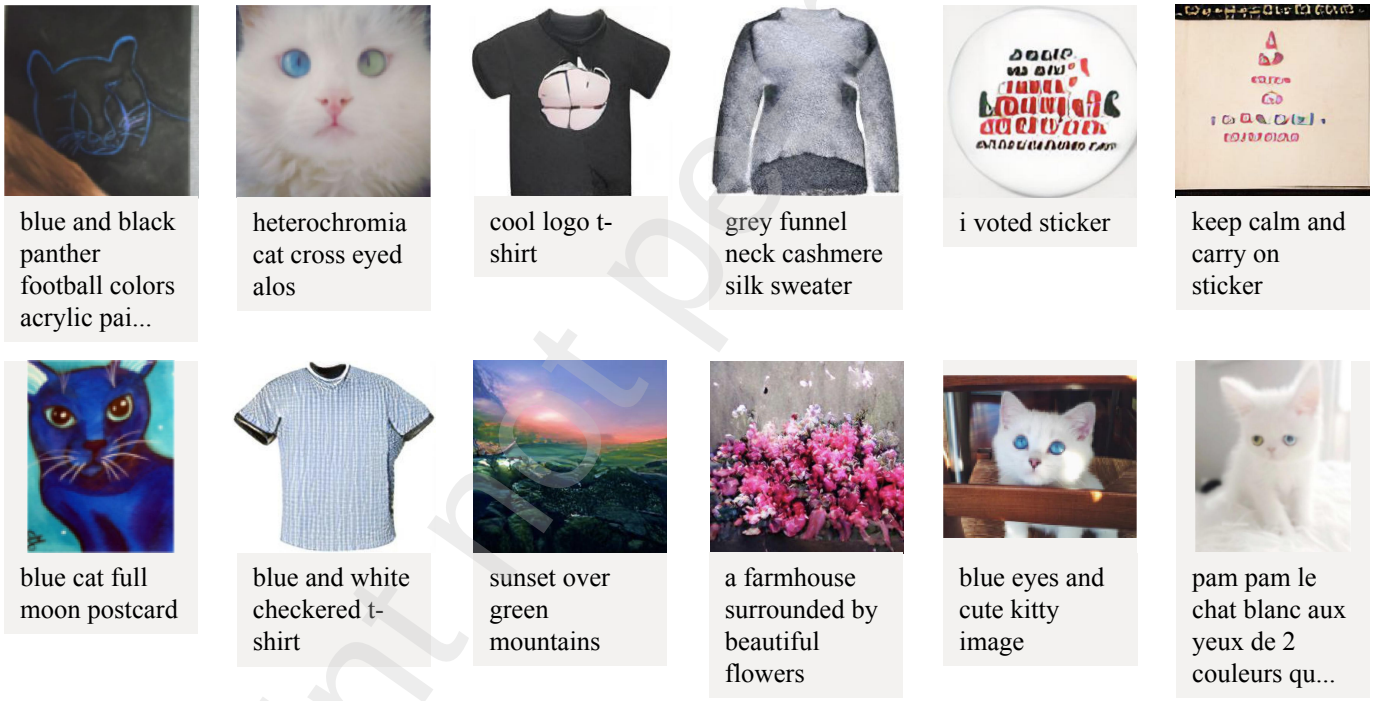

Figure 9: Real-world examples of input pairs for CLIP in the pre-training phase, including descriptive text and images.

represents a triple in the knowledge graph, i.e., (head entity, relation, tail entity). Accordingly, the label-shared semantic information of all examples is "vehicle", and the label-specific semantic information includes "crew member" for "Soyuz TMA-8", "use" for "Tesla Model S", "powered by" for "electric vehicle", and several semantic words (including "commanded by", "location of landing", "part of the series", and "watercraft") for "vehicle". Such a case study can prove that compared with the label-specific semantic information, the label-shared semantic information "vehicle" is more meaningful to prompt the pre-trained CLIP, because in the pre-training phase of CLIP, the input text descriptions corresponding to such input images are more likely to contain the label-shared semantic information "vehicle" than the label-specific words.

Figure 10 provides the direct head-to-head comparison between the label-specific and label-shared prompts. We observe that, on average, our method using the label-shared prompt leads to better performance, consistent with the main results demonstrated in Figure 6. In terms of when the label-shared prompt or the label-specific prompt may be more effective, we have the following suggestions. For generic objects (ImageNet and Caltech101), scenes (SUN397), actions (UCF101), and most fine-grained objects (Food101, OxfordPets, StanfordCars, and DTD), using the label-shared prompt achieves better performance. But

Table 7: Visualization of feature vectors $\boldsymbol{\mu}$ with the length of 16 learned by CPKP. We derive the words by measuring the Euclidean distances between word embeddings and each specific feature vector of $\boldsymbol{\mu}$, and the quantified distances are shown in parentheses. N/A represents non-Latin characters. The task-relevant words are marked in **BOLD**.

#	OxfordPets	SUN397	StanfordCar	UCF101	EuroSAT
1	**cat** (1.5036)	picked (2.5679)	**ford (1.3172)**	,& (1.2602)	py (2.1087)
2	brightest (2.3180)	N/A (2.2843)	hun (1.3301)	**dot (1.4643)**	**contrasting** (2.0325)
3	rj (3.1393)	though (1.6125)	**electr** (2.0140)	support (1.0559)	glau (1.0764)
4	minat (1.9015)	on (2.4797)	N/A (0.9406)	**patients** (1.3442)	qadri (1.8790)
5	**cag** (2.1252)	wolff (2.8140)	N/A (1.7079)	zhu (2.0469)	un (0.9153)
6	imo (1.3375)	can (2.2054)	parades (1.6918)	n (1.1065)	poignant (0.7527)
7	ulties (1.9407)	, (1.7601)	exemp (1.6267)	spani (2.0404)	asin (0.9094)
8	finds (1.1166)	][(1.3146)	ofa (1.9606)	vais (1.2858)	akh (0.7184)
9	gas (1.0581)	crazy (1.6973)	e (1.9894)	**vacancies** (1.3342)	almost (0.7762)
10	N/A (0.9488)	**front (1.8723)**	safetyfirst (1.7781)	exempt (1.5754)	uploading (0.9065)
11	**furry (1.0785)**	beth (3.8847)	cki (1.5057)	**sang** (1.3391)	lower (1.0842)
12	N/A (1.7903)	allthe (1.5069)	ils (1.9784)	N/A (1.3065)	watch (1.0168)
13	txwx (0.9706)	bel (1.5420)	ot (1.9794)	won (1.7255)	**montene (1.5863)**
14	ulty (3.1569)	third (1.7776)	digits (1.9339)	N/A (1.8748)	moy (1.1838)
15	dders (1.2218)	maid (2.8479)	1 (1.9641)	vivian (1.7552)	inindia (1.1385)
16	kha (1.1789)	..... (2.8771)	kes (1.7112)	although (1.5254)	define (1.1644)

Table 8: Visualization of context vectors with the length of 16 learned by CoOp.

#	UCF101	SUN397	StanfordCars	Eurosat	OxfordPets	Flowers102
1	pewdie (1.6189)	][(2.3275)	y (1.4562)	ow (0.7612)	tosc (2.5952)	mul (1.4018)
2	N/A	N/A	flips (1.3660)	wba (0.7127)	judge (1.2635)	leng (1.3333)
3	beh (1.6185)	appears (1.6177)	$ (1.6816)	N/A	fluffy (1.6099)	vish (1.5693)
4	ern (1.0291)	imprisonment (1.3888)	N/A	longtime (0.6693)	cart (1.3958)	N/A
5	runner (1.6778)	private (2.3183)	N/A	ff (0.5972)	harlan (2.2948)	traveled (1.6146)
6	sc (1.5346)	indigen (2.1600)	thats (1.5860)	6 (0.6079)	paw (1.3055)	check (1.1094)
7	N/A	_((1.8676)	N/A	arre (0.5725)	incase (1.2215)	c (1.3797)
8	jel (1.2151)	antly (2.1406)	N/A	prou (0.8177)	bie (1.5454)	N/A
9	frustr (1.2002)	chiev (1.9715)	=> (1.4286)	kp (0.7543)	snuggle (1.1578)	darwin (1.9828)
10	safe (1.1404)	clut (1.7267)	ails (1.9450)	eling (0.5951)	along (1.8298)	:] (1.2083)
11	fill (1.7019)	eck (1.7848)	u (2.2869)	op (0.7235)	enjoyment (2.3495)	un (2.0066)
12	ar (1.4778)	+(2.314)	ty (2.2074)	pap (0.7339)	jt (1.3726)	ldnt (1.8293)
13	yyyyyy (1.8229)	islam (2.0727)	th (1.9125)	shelter (0.7196)	improving (1.3198)	temperature (1.4219)
14	pple (2.2329)	kest (2.2443)	size (1.6790)	ak (0.8664)	srsly (1.6759)	/ (1.2637)
15	im (1.9048)	lucrative (2.2234)	N/A	N/A	asteroid (1.3395)	N/A
16	bourne (1.3172)	kz (2.4315)	fanfest (1.7902)	mar (0.9001)	N/A	@ (1.4321)

on two specific fine-grained datasets (Flowers102 and FGVCAircraft) and a satellite image dataset (EuroSAT), the label-specific prompt is preferred. In addition to differences in categorical objects, we also observe that using the label-specific prompt cannot achieve comparable performance to using the label-shared prompt in challenging few-shot scenarios, e.g., fewer than eight shots. We reckon the reason behind such an observation is that the label-specific prompt version has more parameters than the label-shared prompt version (the label-specific prompt has the same number of learnable feature vectors $\boldsymbol{\mu}$ as the number of categories, while the label-shared prompt only has a fixed number of $\boldsymbol{\mu}$, e.g., 16). Therefore, using the label-specific prompt requires more data for training.

Visualization of Graph-tier Confounder-pruning. CPKP can effectively remove several task-irrelevant relations. We visualize the process of the proposed confounder-pruning. Figure 11 illustrates two examples on Food101 and StanfordCars, demonstrating different relation-types correlated with predicting the graph on different datasets. The results in Figure 6 further support the effectiveness of the proposed confounder-pruning.

7. Conclusion

In this paper, we find the importance of the textual label's semantic information for prompting the pre-trained vision-language model through empirical observation. To explore such

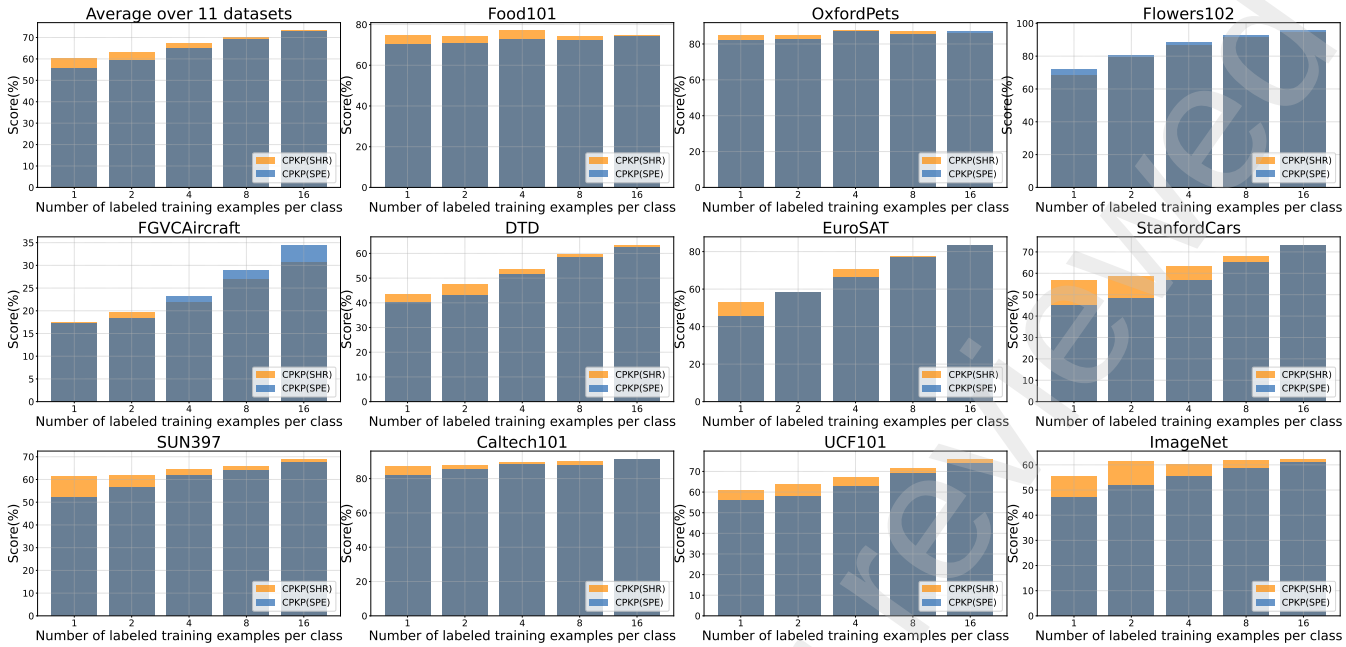

Figure 10: Detailed head-to-head comparisons between label-specific and label-shared prompts on 11 datasets.

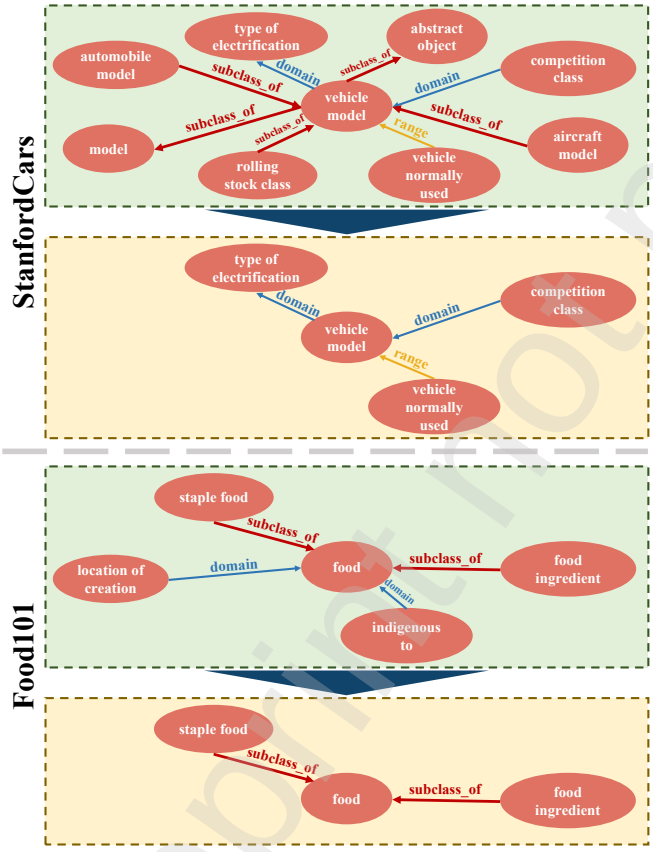

Figure 11: Visualization of the graph-tier confounder-pruning, demonstrating that CPKP removes task-irrelevant relations.

semantic information, we propose a knowledge-aware prompt learning approach called CPKP, which complements semantic information for the input label text by leveraging an ontology knowledge graph and further refining the derived label-relevant subgraph by the proposed double-tier confounder pruning. The extensive experimental comparisons prove the superiority of CPKP over benchmark manual prompt methods and conventional learnable prompt methods in few-shot inference.

Table 9: The case study of label-shared and label-specific semantic information in WIKI-ZS with the appropriate link complementation.

Target entity	Triple		
	Head	Relation	Tail
Soyuz	vehicle	example	Soyuz TMA-8
TMA-8	crew member	example	Soyuz TMA-8
Tesla	use	example	Tesla Model S
Model S	electric vehicle	example	Tesla Model S
electric	powered by	example	electric vehicle
vehicle	electric vehicle	range	vehicle
	electric vehicle	example	Tesla Model S
	vehicle	instance_of	Wikidata property related to transport
	vehicle	relationship	vessel
	vehicle	example	Soyuz TMA-8
vehicle	commanded by	domain	vehicle
	location of landing	domain	vehicle
	electric vehicle	range	vehicle
	part of the series	domain	vehicle
	watercraft	subclass_of	vehicle

Acknowledgements

We thank the reviewers for their efforts in revising this article. This work is supported by the China Postdoctoral Science Foundation, Grant No. 2024M753356, the Fundamental Research Program, China, Grant No. JCKY2022130C020, the National Funding Program for Postdoctoral Researchers, Grant No. GZC20232812, and 2023 Special Research Assistant Project.

References

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering. In *CVPR*. Computer Vision Foundation / IEEE Computer Society, 6077–6086.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. VQA: Visual Question Answering. In *ICCV*. IEEE Computer Society, 2425–2433.

Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating Embeddings for Modeling Multi-relational Data. In *NIPS*. 2787–2795.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. 2014. Food-101 - Mining Discriminative Components with Random Forests. In *ECCV (6) (Lecture Notes in Computer Science)*. Springer.

Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka Jr., and Tom M. Mitchell. 2010. Toward an Architecture for Never-Ending Language Learning. In *AAAI*. AAAI Press.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. 2020. A Simple Framework for Contrastive Learning of Visual Representations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event (Proceedings of Machine Learning Research)*. PMLR.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2019. UNITER: Learning UNiversal Image-TExt Representations. *CoRR* abs/1909.11740 (2019).

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. 2014. Describing Textures in the Wild. In *CVPR*. IEEE Computer Society, 3606–3613.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *CVPR*. IEEE Computer Society, 248–255.

Mohammad Mahdi Derakhshani, Enrique Sanchez, Adrian Bulat, Victor G Turrisi da Costa, Cees GM Snoek, Georgios Tzimiropoulos, and Brais Martinez. 2023. Bayesian prompt learning for image-language model generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 15237–15246.

David A. Ferrucci, Eric W. Brown, Jennifer Chu-Carroll, James Fan, David Gondek, Aditya Kalyanpur, Adam Lally, J. William Murdock, Eric Nyberg, John M. Prager, Nico Schlaefer, and Christopher A. Welty. 2010. Building Watson: An Overview of the DeepQA Project. *AI Mag.* 31, 3 (2010), 59–79.

Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. 2021. CLIP-Adapter: Better Vision-Language Models with Feature Adapters. *CoRR* (2021).

Peng Gao, Zhengkai Jiang, Haoxuan You, Pan Lu, Steven C. H. Hoi, Xiaogang Wang, and Hongsheng Li. 2019. Dynamic Fusion With Intra- and Inter-Modality Attention Flow for Visual Question Answering. In *CVPR*. Computer Vision Foundation / IEEE, 6639–6648.

Yuxia Geng, Jiaoyan Chen, Zhuo Chen, Jeff Z. Pan, Zhiquan Ye, Zonggang Yuan, Yantao Jia, and Huajun Chen. 2021. OntoZSL: Ontology-enhanced Zero-shot Learning. In *WWW*. ACM / IW3C2, 3325–3336.

Xavier Glorot, Antoine Bordes, Jason Weston, and Yoshua Bengio. 2013. A Semantic Matching Energy Function for Learning with Multi-relational Data. In *ICLR (Workshop Poster)*.

Madelyn Glymour, Judea Pearl, and Nicholas P Jewell. 2016. *Causal inference in statistics: A primer*. John Wiley & Sons.

A. Goldstein, A. Kapelner, J. Bleich, and E. Pitkin. 2015. Peeking Inside the Black Box: Visualizing Statistical Learning with Plots of Individual Conditional Expectation.. In *Journal of Computational and Graphical Statistics, 24(1):44âĂŞ65, 2015*.

Palash Goyal and Emilio Ferrara. 2018. Graph embedding techniques, applications, and performance: A survey. *Knowl. Based Syst.* 151 (2018), 78–94.

Clive Granger. 1969. Investigating Causal Relations by Econometric Models and Cross-Spectral Methods. *Econometrica* 37 (02 1969), 424–38. `https://doi.org/10.2307/1912791`

Lee Harland. 2012. Open PHACTS: A Semantic Knowledge Infrastructure for Public and Commercial Drug Discovery Research. In *EKAW (Lecture Notes in Computer Science, Vol. 7603)*. Springer, 1–7.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 770–778.

Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. 2019. EuroSAT: A Novel Dataset and Deep Learning Benchmark for Land Use and Land Cover Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.* 12, 7 (2019), 2217–2226.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-Efficient Transfer Learning for NLP. In *ICML (Proceedings of Machine Learning Research, Vol. 97)*. PMLR, 2790–2799.

Lun Huang, Wenmin Wang, Jie Chen, and Xiaoyong Wei. 2019. Attention on Attention for Image Captioning. In *ICCV*. IEEE, 4633–4642.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How Can We Know What Language Models Know. *Trans. Assoc. Comput. Linguistics* (2020).

Woojeong Jin, Yu Cheng, Yelong Shen, Weizhu Chen, and Xiang Ren. 2022. A Good Prompt Is Worth Millions of Parameters: Low-resource Prompt-based Learning for Vision-Language Models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*.

Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. 2023. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 19113–19122.

Jin-Hwa Kim, Jaehyun Jun, and Byoung-Tak Zhang. 2018. Bilinear Attention Networks. *CoRR* abs/1805.07932 (2018).

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 2013. 3D Object Representations for Fine-Grained Categorization. In *ICCV Workshops*. IEEE Computer Society, 554–561.

FF Li, R. Fergus, and P. Perona. 2004. Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories. In *Conference on Computer Vision & Pattern Recognition Workshop*.

Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. 2020. Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks. In *ECCV (30) (Lecture Notes in Computer Science, Vol. 12375)*. Springer, 121–137.

Wanyu Lin, Hao Lan, and Baochun Li. 2021. Generative Causal Explanations for Graph Neural Networks. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event (Proceedings of Machine Learning Research)*. PMLR.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning Entity and Relation Embeddings for Knowledge Graph Completion. In *AAAI*. AAAI Press, 2181–2187.

Zhiqiu Lin, Samuel Yu, Zhiyi Kuang, Deepak Pathak, and Deva Ramanan. 2023. Multimodality Helps Unimodality: Cross-Modal Few-Shot Learning With Multimodal Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 19325–19337.

Xin Liu, Zhongdao Wang, Yali Li, and Shengjin Wang. 2022a. Self-Supervised Learning via Maximum Entropy Coding. *CoRR* abs/2210.11464 (2022). `https://doi.org/10.48550/arXiv.2210.11464` arXiv:2210.11464

Xin Liu, Zhongdao Wang, Ya-Li Li, and Shengjin Wang. 2022b. Self-Supervised Learning via Maximum Entropy Coding. In *Advances in Neural Information Processing Systems*, Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (Eds.). `https://openreview.net/forum?id=nJt27NQffr`

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. In *NeurIPS*. 13–23.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew B. Blaschko, and Andrea Vedaldi. 2013. Fine-Grained Visual Classification of Aircraft. *CoRR* abs/1306.5151 (2013).

Tadas K Nakamura. 2000. Statistical mechanics of a collisionless system based on the maximum entropy principle. *The Astrophysical Journal* 531, 2 (2000), 739.

Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A Three-Way Model for Collective Learning on Multi-Relational Data. In *ICML*.

Omnipress, 809–816.

Nilsback, ME, and Zisserman. 2008. Automated flower classification over a large number of classes. *ICVGIP* (2008), 722–729.

Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. 2012. Cats and dogs. In *CVPR*. IEEE Computer Society, 3498–3505.

Judea Pearl. 2009. Causal inference in statistics: An overview. *Statistics surveys* (2009), 96–146.

Pengda Qin, Xin Wang, Wenhu Chen, Chunyun Zhang, Weiran Xu, and William Yang Wang. 2020. Generative Adversarial Zero-Shot Relational Learning for Knowledge Graphs. In *AAAI*. AAAI Press, 8673–8680.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*.

Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. 2021. DenseCLIP: Language-Guided Dense Prediction with Context-Aware Prompting. *CoRR* (2021).

Timo Schick and Hinrich Schütze. 2021. Exploiting Cloze-Questions for Few-Shot Text Classification and Natural Language Inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*. https://doi.org/10.18653/v1/2021.eacl-main.20

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*. https://doi.org/10.18653/v1/2020.emnlp-main.346

Richard Socher, Danqi Chen, Christopher D. Manning, and Andrew Y. Ng. 2013. Reasoning With Neural Tensor Networks for Knowledge Base Completion. In *NIPS*. 926–934.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. 2012. UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild. *CoRR* abs/1212.0402 (2012).

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. ConceptNet 5.5: An Open Multilingual Graph of General Knowledge. In *AAAI*. AAAI Press, 4444–4451.

Hao Tan and Mohit Bansal. 2019. LXMERT: Learning Cross-Modality Encoder Representations from Transformers. In *EMNLP/IJCNLP (1)*. Association for Computational Linguistics, 5099–5110.

Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. 2008. ArnetMiner: extraction and mining of academic social networks. In *KDD*. ACM, 990–998.

Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B. Tenenbaum, and Phillip Isola. 2020. Rethinking Few-Shot Image Classification: A Good Embedding is All You Need?. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XIV (Lecture Notes in Computer Science)*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*.

Denny Vrandecic and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Commun. ACM* 57, 10 (2014), 78–85.

Hongwei Wang, Miao Zhao, Xing Xie, Wenjie Li, and Minyi Guo. 2019. Knowledge Graph Convolutional Networks for Recommender Systems. In *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*. ACM.

Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. 2017. Knowledge Graph Embedding: A Survey of Approaches and Applications. *IEEE Trans. Knowl. Data Eng.* 29, 12 (2017), 2724–2743.

Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. 2010. SUN database: Large-scale scene recognition from abbey to zoo. In *CVPR*. IEEE Computer Society, 3485–3492.

Bo Xu, Yong Xu, Jiaqing Liang, Chenhao Xie, Bin Liang, Wanyun Cui, and Yanghua Xiao. 2017. CN-DBpedia: A Never-Ending Chinese Knowledge Extraction System. In *IEA/AIE (2) (Lecture Notes in Computer Science, Vol. 10351)*. Springer, 428–438.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. KG-BERT: BERT for Knowledge Graph Completion. *CoRR* abs/1909.03193 (2019).

Ma Yi, Derksen Harm, Hong Wei, and Wright John. 2007. Segmentation of Multivariate Mixed Data via Lossy Data Coding and Compression. *IEEE Trans. Pattern Anal. Mach. Intell.* 29, 9 (2007), 1546–1562. https://doi.org/10.1109/TPAMI.2007.1085

Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. 2016. Image Captioning with Semantic Attention. In *CVPR*. IEEE Computer Society, 4651–4659.

Renrui Zhang, Rongyao Fang, Wei Zhang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. 2021. Tip-Adapter: Training-free CLIP-Adapter for Better Vision-Language Modeling. *CoRR* abs/2111.03930 (2021).

Yuhao Zhang, Hang Jiang, Yasuhide Miura, Christopher D. Manning, and Curtis P. Langlotz. 2020. Contrastive Learning of Medical Visual Representations from Paired Images and Text. *CoRR* abs/2010.00747 (2020).

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2022. Learning to Prompt for Vision-Language Models. *Int. J. Comput. Vis.* 130, 9 (2022), 2337–2348. https://doi.org/10.1007/s11263-022-01653-1

Xiangyang Zhu, Renrui Zhang, Bowei He, Aojun Zhou, Dong Wang, Bin Zhao, and Peng Gao. 2023. Not All Features Matter: Enhancing Few-shot CLIP with Adaptive Prior Refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2605–2615.