

Learning Textual Prompts for Open-World Semi-Supervised Learning

Appendix

A. Analysis of the Effectiveness of Each Term of the Loss Function

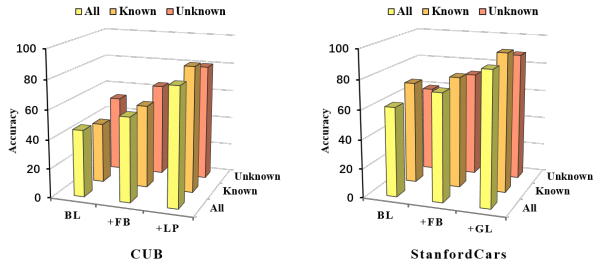


Figure 1. Ablation studies on each term of the loss function on the different datasets.

We perform ablation studies on each term of the loss function to assess its impact on the final model performance. The experimental results from the Flowers and Oxford Pets datasets are detailed within the main text. Additionally, we complement these with results from the CUB and Stanford Cars datasets. Initially, we establish a baseline by training the model with only the L_l^{GP} term. We then enhance this baseline by integrating the forward-and-backward strategy, which includes the L_u^{GP} term in the training process. Finally, we use the complete loss function that encompasses both the forward-and-backward strategy and the global-and-local textual prompt learning strategy. During inference, we use $Result_i^g$. The detailed results of these ablation studies are depicted in Figure 1. In the figure, “BL” denotes the baseline, “+FB” represents the integration of the forward-and-backward strategy based on baseline, and “+GL” signifies the further introduction of the global-and-local textual prompt learning strategy. The results indicate that each term is instrumental in improving model accuracy, especially in datasets with numerous classes, where classification tasks are more complex.

B. Analysis of the Effectiveness of Local Visual Features and Local Textual Prompts

Global visual features often contain irrelevant information from the background, which can introduce interference and

reduce the precision of the model’s classification performance. To address the challenge of background noise, we incorporate local visual features and local textual prompts into the training process, which strengthens the alignment between images and text, allowing the model to better focus on features pertinent to the classification. During the inference phase, we employ a strategy that combines both global and local textual prompts for classification. We conduct a series of experiments to evaluate the impact of these local features and prompts. The results of our experiments on the Flowers and Oxford Pets datasets are detailed in the main text. In this section, we supplement those findings with additional results from the CUB and Stanford Cars datasets. Initially, we train the model with L_l^{GP} and L_u^{GP} . During the inference phase, we use $Result_i^g$. Subsequently, we train the model with the complete loss function, and we use $Result_i$ for the inference phase. The findings are presented in Figure 2. In the figure, “GP” represents the first experimental setup, while “+LP” indicates the second experimental setup. The experimental results suggest that integrating local visual features and local textual prompts enhances the model’s capability to identify and distinguish key features across different classes.

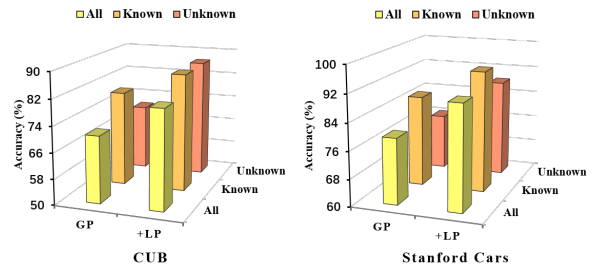


Figure 2. Validation of the effectiveness of local visual features and local textual prompts.

C. Analysis of Hyperparameter Sensitivity in Multiscale Local Textual Prompts Loss

In the multiscale local textual prompts loss function, several hyperparameters are defined: the symbol s represents the number of local textual prompts, with a default setting of

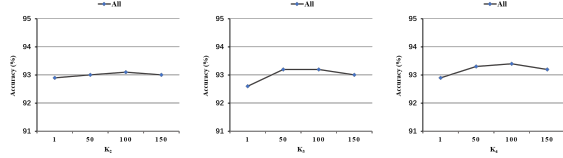


Figure 3. Validating K Sensitivity in Oxford Pets Dataset.

4; K_1 , K_2 , K_3 , and K_4 are parameters for different scales, with default values of 5, 10, 15, and 20, respectively. To investigate the specific impact of these hyperparameters on model performance, two related experiments are conducted in the main text, and additional experiments are supplemented here. Figure 3 demonstrates the model performance when s is fixed at 4 and other K values remain unchanged, with varying values for K_2 , K_3 , and K_4 . The experimental results indicate that the model performance remains relatively stable. These findings confirm that our method is robust to the hyperparameters considered.

D. Analysis of Learnable Parameters Quantity in the Proposed Method

The learnable parameters of TP-OWSSL include the textual prompts and the fully connected layer. In contrast, TextGCD requires fine-tuning of the image encoder, text encoder, and two classifiers, a process that involves approximately 10.34M learnable parameters for a classification task with 100 classes. The number of learnable parameters in TP-OWSSL is only 5% of that in TextGCD, yet it achieves superior performance.