# Unknown Text Learning for CLIP-based Few-shot Open-Set Recognition

Supplementary Material

## S1. Detailed criteria for U$^2$WO

**Construction of Base Classes.** As per the standard few-shot setting [3, 11, 12, 42, 51], we assume access to base classes. Unlike previous FSOR works, our U$^2$WO collects class descriptions (i.e., class label name) only. We collect 1000 label names for constructing the word embedding space and use the known class name and outlier text as the base classes. Specifically, to avoid unknown class names in downstream tasks, we first use the known class names as the base classes. For the large-scale benchmarks (i.e., ImageNet), the abundance of known class names suffices to construct the word embedding space. However, the word embedding space is constrained to small benchmarks (i.e., CIFAR-based benchmarks) with only access to a few known class names. To address this limitation, we select 1000 outlier text from WordNet [26].

**Outlier Text Filtering.** We filter 1,000 distinct class names as the outlier text. Specifically, we employ a subset of WordNet (a complete labeling system) as an initial and general word source, i.e., the label names from ImageNet-21K [35]. To ensure that we do not leverage any linguistic information about the unknown classes, we exclude words related to these unknown classes by using the Synset IDs from WordNet [26]. This strategy guarantees that any labels associated with unknown classes are entirely omitted. It's important to note that we apply the same word source across all tasks, and our experiments demonstrate that this word source generalizes well across various applications.

**Known Basis and Open Basis.** We explore two variants of U$^2$WO: U$^2$WO with Known Basis (UKB), where the base class names are drawn from known classes; and U$^2$WO with Open Basis (UOB), where the base class names are sourced from outlier text. First, we clarify that the core principle of UTL is to construct the semantic space based primarily on known class names. The incorporation of WordNet labels is only necessary when the number of known classes is insufficient, for example, in the C10 benchmark, which contains only six known class names. Alternatively, the universal basis can be constructed directly from the textual representations of the known classes, and the use of WordNet-based outlier text is not a strict requirement. Furthermore, for small-scale benchmarks with relatively few unknown categories, it is often easy to collect a sufficient number of semantically diverse class names from external sources that are explicitly disjoint from the predefined unknown categories. This makes the construction of a universal semantic basis both practical and effective in real-world open-set scenarios.

**Extraction of Word Embedding.** We feed the 1000 label names to the CLIP and extract the corresponding word embeddings. Then, we extract the 1000 embeddings with a fixed token length. In our experiments, we fix the length to 30.

## S2. Experimental Setting

**Datasets.** We conduct experiments on benchmark datasets for FSOR. Firstly, we construct the standard benchmark for CLIP-based FSOR based on the traditional OSR standard benchmarks [3, 5, 6, 15, 24, 27, 40, 42, 47, 53]. Specifically, it covers six datasets: **CIFAR10 (C10) [16]:** Six known classes and four unknown classes are randomly sampled. **CIFAR10 + 10 (C10 + 10), CIFAR10 + 50 (C10 + 50):** For the CIFAR10 + N experiments, four classes from CIFAR10 are used for training, while $N$ classes from CIFAR100 [16] are used for evaluation, where $N$ denotes either 10 or 50 classes. **CIFAR100 (C100) [16]:** Twenty known classes and eighty unknown classes are randomly sampled. **TinyImageNet (TINY) [17]:** Twenty known classes and one hundred eighty unknown classes are randomly sampled for evaluation. **TinyImageNet-Hard (TINY-H) [17]:** Forty known classes and one hundred sixty unknown classes are randomly sampled for evaluation. The above six benchmarks are evaluated over five random splits. Then, we construct two benchmarks with the two widely used datasets in FSOR [3, 11, 12, 20, 29, 42, 51]. **MiniImageNet (MiniIN) [41]:** Sixty-four known classes and thirty-six unknown classes are used for evaluation. MiniImageNet contains one hundred classes, and the classes are split as (64, 16, 20) for meta-training, meta-validation, and meta-testing, respectively. Each class has six hundred images. We extended the split as (64, 16+20) for known and unknown classes, respectively. **TieredImageNet (TieredIN) [34]:** Three hundred fifty-one known classes and two hundred fifty-seven unknown classes are used for evaluation. TieredImageNet contains six hundred eight classes, split into (351, 97, and 160). We extended the split as (351, 97+160) for evaluation. Finally, we employ the more challenging large-scale benchmark datasets, which involve long-tail datasets with semantic shift benchmarks. **ImageNet-200 (IN-200) [43]:** Following the dataset preparation [5, 43], we selected the first 200 classes of the ImageNet-1k dataset [7] as known and the remaining 800 ones as the unknown. **iNaturalist (iNa) [39]:** We simplify the long-tail setting [47], we randomly select 200 classes as known classes and the remaining 800 classes as unknown. **ImageNet-Easy (IN-Easy) and ImageNet-Hard (IN-Hard) [40]:** A large-scale eval-

Table S6. Comparison (%) of our UTL with different methods on various benchmarks under 16-shot setting in terms of **OSCR**. The best and second-best results are highlighted in **bold** and underline, respectively.

| Methods(ViT-B/16) | C10 | C10+10 | C10+50 | C100 | TINY | TINY_H | Average* | Average |
|---|---|---|---|---|---|---|---|---|
| | | | | Training-free | | | | |
| CLIP [32] | 88.08±2.23 | 89.33±2.56 | 89.89±1.78 | 72.40±4.73 | 79.16±3.39 | 71.68±3.57 | 86.62 | 81.76 |
| CLIP-PE [32] | 89.48±2.22 | 92.02±2.09 | 92.06±1.47 | 71.99±4.91 | 80.58±3.01 | 73.68±3.23 | 88.54 | 83.30 |
| LMC [31] | **93.60±1.50** | **96.80±0.70** | **96.40±0.40** | - | 80.60±3.40 | - | 91.85 | - |
| | | | Few-shot close-set methods (16-shot setting) | | | | | |
| CoOP [54] | 89.29±2.40 | 95.2±0.90 | 93.28±1.30 | 73.42±2.08 | 81.51±3.45 | 75.47±2.13 | 89.82 | 84.70 |
| CoCoOp [13] | 90.78±2.66 | 93.87±1.75 | 92.65±1.60 | 73.45±4.36 | 82.42±2.61 | 76.12±2.26 | 89.93 | 84.88 |
| | | | Few-shot OSR methods (16-shot setting) | | | | | |
| M-Tuning [19] | 91.72±1.72 | 95.74±0.72 | 93.78±1.41 | 74.02±4.00 | 82.02±3.26 | 75.89±2.40 | 90.81 | 85.53 |
| **UTL** (Our) | 93.44±1.20 | 96.7±0.58 | 95.32±0.85 | **77.60±3.65** | **85.19±2.29** | **78.31±2.00** | **92.66** | **87.76** |

Table S7. Comparison (%) of our UTL with different methods on challenging benchmarks under 16-shot setting in terms of **close-set performance (ACC)**. The best and second-best results are highlighted in **bold** and underline, respectively.

| Methods(ViT-B/16) | Mini-IN | Tiered-IN | IN200 | iNa | Places | IN | IN_LT | Average |
|---|---|---|---|---|---|---|---|---|
| | | | Training-free | | | | | |
| CLIP [32] | 91.60 | 69.00 | 74.50 | 29.90 | 36.40 | 66.80 | 66.80 | 62.14 |
| CLIP-PE [32] | 92.30 | 70.70 | 75.50 | 29.80 | 57.50 | 68.80 | 68.80 | 66.20 |
| | | | Few-shot close-set methods | | | | | |
| CoOp (16-shot setting) [54] | 94.07 | 75.57 | 76.77 | 65.40 | 62.00 | 71.70 | 71.83 | 73.91 |
| CoCoOp (16-shot setting) [13] | **94.53** | 74.50 | **77.33** | 42.30 | 61.20 | 71.10 | 71.10 | 70.29 |
| | | | Few-shot OSR methods (16-shot setting) | | | | | |
| M-Tuning (16-shot setting) [19] | 94.33 | 75.50 | 76.62 | 64.80 | **61.70** | 71.60 | **83.26** | **75.40** |
| **UTL (16-shot setting) (Our)** | 94.50 | **75.67** | 76.90 | **67.60** | 61.60 | **72.10** | 71.70 | 74.30 |

uation for category shift and open-set splits based on semantic distances to the ImageNet. The known is ImagNet-1k. The unknown samples are chosen from the disjoint set of ImageNet-21K-P [35]. The total semantic distances to the categories of ImageNet split the 'Easy' and 'Hard' categories. Each unknown set includes 1000 categories. **ImageNet-LT (IN-LT) [21]:** As a longtailed dataset, includes 1000 known classes from ImageNet-2012, and 360 unknown classes from the validation set of ImageNet-2010. **Places365-Standard [23]:** 365 scene classes split as (100, 265). 100 randomly selected classes as known, and the other 265 as unknown.

**Evaluation Metrics.** Following previous works [3, 11, 12, 42, 51], we evaluate the model's closed-set performance with accuracy (ACC) and evaluate the open performance with the area under the ROC curve (AUROC) for unknown class detection. The closed-set performance is the classification capacity via known samples, and the open performance is the unknown detection capacity via both known and unknown samples. As the previous few-shot methods [10, 13, 45, 46, 50, 54, 55] have already demonstrated promising closed-set performance, the challenges lie in open performance.

**Implementation Details.** In our experiments, we employ the available vision backbones in CLIP, including RN50 and ViT-B/16. The "`[CLASS]`" is placed at the end of the prompts. The number of the unknown word tokens ($M$) is set to 20, and the number of context tokens ($P$) is set to 16. The number of universal word bases ($F$) is set to 100. The weights matrix $\boldsymbol{W}$ is initialized with random values drawn from a zero-mean Gaussian distribution with a standard deviation of 0.02. We employ the SGD optimizer with an initial learning rate of 0.002, decayed according to the cosine annealing rule, following the default setup in CoOp. The maximum number of epochs is set to 50, and the batch size is set to 32. All experiments are conducted on a single NVIDIA Tesla V100 GPU. The source code will be publicly available.

## S3. Compared Methods

To evaluate the performance of our UTL in a few-shot setting, we reconstruct some baselines for comparison. First, we compare UTL with zero-shot prompt learning methods. **CLIP [32]:** Zero-shot learning (ZSL) prediction with the

Table S8. Comparison (%) of different methods on all-data settings in terms of AUROC. The best and second-best results are highlighted in **bold** and <u>underline</u>, respectively.

| Methods (ViT-B/16) | Publication | C10 | C + 10 | C + 50 | TINY | IN-200 | IN-LT | Average |
|---|---|---|---|---|---|---|---|---|
| Traditional OSR methods | | | | | | | | |
| CPN (all-data) [43] | TPAMI'20 | 82.80 | 88.10 | 87.90 | 63.90 | 79.56 | – | – |
| RPL (all-data) [5] | ECCV'20 | 86.10 | 85.60 | 85.00 | 70.20 | 91.70 | 55.20 | 78.97 |
| ARPL (all-data) [6] | TPAMI'21 | 91.00 | 97.10 | 95.10 | 78.20 | 94.90 | – | – |
| PMAL (all-data) [24] | AAAI'22 | 95.10 | 97.80 | 96.90 | 83.10 | 93.90 | 71.70 | 89.75 |
| All-You-Need (all-data) [40] | ICLR'22 | 93.60 | 97.90 | 96.50 | 83.00 | 95.71 | 59.18 | 87.65 |
| (ARPL+CS) (all-data) [40] | ICLR'22 | 93.90 | <u>98.10</u> | 96.70 | 82.50 | 96.16 | 62.55 | 88.32 |
| CLIP-based OSR methods | | | | | | | | |
| M-Tuning (all-data) [19] | arXiv'23 | <u>96.29</u> | 96.28 | 96.17 | <u>87.30</u> | <u>96.47</u> | <u>78.89</u> | <u>91.90</u> |
| VP-CK (all-data) [14] | AAAI'24 | 95.20 | 97.90 | <u>97.10</u> | - | 83.10 | - | - |
| **UTL (all-data)** | Ours | **96.84** | **99.34** | **97.79** | **91.15** | **98.50** | **87.64** | **95.21** |

Table S9. Comparison (%) of different methods on all-data settings in terms of the closed-set ACC. The best and second-best results are highlighted in **bold** and <u>underline</u>, respectively. Average$^{*}$ is the mean ACC across datasets excluding IN-LT.

| Methods (ViT-B/16) | Publication | C10 | C + N | TINY | IN-200 | IN-LT | Avarage | Average$^{*}$ |
|---|---|---|---|---|---|---|---|---|
| Traditional OSR methods | | | | | | | | |
| CPN (all-data) [43] | TPAMI'20 | 92.90 | 94.80 | 81.40 | 82.20 | 37.10 | 77.68 | 87.83 |
| RPL (all-data) [5] | ECCV'20 | 95.10 | 95.50 | 81.70 | 66.20 | 39.70 | 75.64 | 84.63 |
| ARPL (all-data) [6] | TPAMI'21 | 94.50 | 94.70 | 76.10 | <u>82.30</u> | 39.70 | 77.46 | 86.90 |
| PMAL (all-data) [24] | AAAI'22 | **97.50** | <u>97.80</u> | 84.70 | **84.10** | 42.90 | 81.40 | 91.03 |
| All-You-Need (all-data) [40] | ICLR'22 | 94.10 | 95.92 | 71.20 | 71.30 | – | – | – |
| ARPL+CS (all-data) [40] | ICLR'22 | 94.25 | 97.07 | 76.84 | 79.02 | – | – | – |
| CLIP-based OSR methods | | | | | | | | |
| M-Tuning (all-data) [19] | arXiv'23 | 96.30 | 96.23 | <u>90.70</u> | 81.90 | **81.74** | **89.37** | <u>91.28</u> |
| **UTL (all-data)** | Ours | <u>97.36</u> | **98.30** | **94.32** | 81.20 | <u>68.10</u> | <u>87.86</u> | **92.80** |

handcrafted templates, `"a photo of a [CLASS]."`. **CLIP with Prompt Engineering (CLIP-PE) [32]:** Zero-shot learning (ZSL) prediction with 7 ImageNet-select handcrafted templates. **LMC [31]:** Training-free method for open-set recognition. We report the performances from the corresponding sources.

Then, close-set few-shot prompt learning methods: **CoOp [54] and CoCoOp [13]:** These experiments are performed using the publicly released code. In CoOp, the class token is placed at the end, and we set the context length to 16 using random initialization. For CoCoOp, we fix the context length to 4 and initialize the context vectors using the pre-trained word embeddings of `"a photo of a"`. For the methods (i.e., CLIP, CLIP-PE, CoOp, and CoCoOp) that did not conduct experiments under the open-set setting, we re-implement them using publicly available code and report the results.

Next, the few-shot out-of-distribution (OOD) methods: **LoCoOp [28], ID-like [1], NegPromt [18]**. These methods employed the OOD setting and did not report the results on our compared benchmarks, so we reproduced them by using publicly available code. Particularly, the scoring function in MCM [27] was further adapted in LoCoOp. LoCoOp showed that LoCoOp-MCM is inferior to LoCoOp-GL, where the latter (LoCoOp-GL) was discussed and compared in our work. Following the few-shot open-setting methods: **M-Tuning [19]:** The `[class]` is placed in the middle of prompts, whose length L is set as 10. For performance evaluation, we use the maximum softmax probability (MSP). For M-tuning, we reset the context length to 16 and placed the `[class]` in the end. We reproduced it to compare it with other methods across all benchmarks. The above methods are compared in a few-shot setting. Finally, we compared UTL with traditional OSR methods: **CPN [43], RPL [5], ARPL [6], ARPL+CS [40], PMAL [24], and All-You-Need [40], VP-CK [14]**. These methods are compared under an all-data setting.

## S4. More results of 16-shot setting

**OSCR results.** As AUROC and ACC are two widely used metrics for OSR, we mainly report results in terms of them for comparison. To further evaluate the effect of

Table S10. Comparison(%) of CLIP-based methods with the backbone of RN50 on standard benchmark in terms of AUROC. The best results are highlighted in **bold**. Average$^*$ is the mean AUROC across datasets excluding IN-Easy and IN-Hard.

| Method (RN50) | C10 | C10+10 | C10+50 | TINY | Average* | IN-Easy | IN-Hard | Average |
|---|---|---|---|---|---|---|---|---|
| Train-free methods | | | | | | | | |
| CLIP [32] | 75.80 | 86.72 | 85.21 | 74.59 | 80.58 | 70.14 | 58.22 | 75.11 |
| Few-shot methods | | | | | | | | |
| CoOp (16-shot setting) [54] | 76.93 | 90.29 | 87.66 | 77.67 | 83.14 | 69.52 | 58.44 | 76.75 |
| Few-shot OSR methods | | | | | | | | |
| **UTL (16-shot setting) (Our)** | **85.31** | **96.60** | **93.85** | **81.01** | **89.19** | **78.64** | **69.70** | **84.19** |
| **UTL (all-data setting) (Our)** | **89.52** | **98.40** | **96.44** | **85.12** | **92.37** | - | - | - |

our UTL on standard benchmarks under the 16-shot setting, we also present the OSCR results in Tab. S6, where our UTL method still achieves the best average results across all benchmarks. Besides, all results in terms of OSCR show a similar trend to those in terms of AUROC.

**ACC results.** The close-set performance of different methods is presented in Tab. S7. Our UTL achieves competitive performance on the large-scale benchmarks. Compared with the CLIP-based methods, our UTL achieves the second-best performance on average and is slightly inferior to R-Tuning. The reason mainly lies in that R-Tuning introduces a CTT strategy to divide the known classes into many small groups. Such a strategy significantly improves closed-set accuracy on the IN-LT dataset. When the results on the IN-LT dataset are excluded, our UTL achieves the best performance under an all-data setting. These results above clearly demonstrate the effectiveness of UTL again.

## S5. More results of all-data Settings

To evaluate the effect of our UTL on all-data settings, we conduct experiments on various benchmarks with the backbone of VIT-B/16. The results of AUROC and ACC are presented in Tab. S8 and Tab. S9. As described in Tab. S8, our UTL achieves the best open-set performance on average. Particularly, our UTL performs an average 7.56% gain over the best traditional OSR method, All-You-Need. Compared with the CLIP-based OSR methods, our UTL performs a 3.31% improvement over M-Tuing. Similar to the 16-shot setting, our UTL achieves the second-best closed-set performance on average under the all-data setting as demonstrated in Tab. S9. Our UTL performs inferior to M-Tuning on close-set performance. Our UTL achieves the best close-set performance across benchmarks, excluding IN-LT. The above results again verify the effectiveness of our UTL in both known classification and unknown detection tasks.

## S6. Additional Results with Backbone of RN50

To verify the effectiveness with different backbones, we compare CLIP-based methods with the backbone of RN50 on standard OSR benchmarks. The open-set performance is presented in Tab. S10. Our UTL performs best on standard benchmarks and outperforms the CLIP-based methods by a large margin. Compared with CLIP and CoOp, UTL obtains an average of 8.61% and 6.05% improvement. Note that CoOp achieves 2.56% improvement over Zero-shot CLIP, which indicates context optimization under the closed-set setting improves the open set performance with RN50. UTL achieves 3.18% improvement gains in the all-data setting over the 16-shot setting. The above results demonstrate the effectiveness of UTL with backbone RN50, which shows strong generalization and good flexibility on both 16-shot and all-data settings with various backbones on standard benchmarks. To further evaluate the effectiveness of RN50, we compare it with several CLIP-based methods on IN-Easy and IN-Hard. UTL outperforms existing CLIP-based methods significantly, indicating remarkable effectiveness in extensive semantic shift open set benchmarks. For instance, UTL surpasses CLIP and CoOp by 9.08% and 7.44% on average, respectively.

Table S11. The AUROC and ACC results of UTL with various $\lambda$ on TINY.

| $\lambda$ | 0.1 | 0.3 | 0.5 | 0.7 | 0.9 |
|---|---|---|---|---|---|
| ACC | **91.59** | 91.48 | 91.19 | 91.17 | 91.13 |
| AUROC | 89.43 | 89.24 | 89.71 | 90.06 | **90.64** |

Table S12. The AUROC results of UTL with various numbers of universal word basis $F$ on IN-Easy.

| $F$ | 20 | 30 | 70 | 100 | 200 | 300 |
|---|---|---|---|---|---|---|
| AUROC | 81.46 | 81.97 | 82.05 | **82.33** | 82.32 | 82.04 |

## S7. Additional Ablation Study

**Effect of balance coefficient ($\lambda$).** To evaluate the effect of universal word basis $\lambda$, we conduct experiments with the backbone of ViT-B/16 by varying $\lambda \in$

Table S13. Comparison (%) of MSP with MLS on UTL across various benchmarks.

| ViT-B/16 | C10 | C10 + 10 | C10 + 50 | TINY |
|---|---|---|---|---|
| MSP | **96.37** | **98.73** | **97.27** | **89.74** |
| MLS | 94.64 | 98.25 | 96.57 | 88.26 |
| IN-200 | IN-LT | IN-Easy | IN-Hard | Average |
| 92.69 | 75.10 | 72.37 | 61.92 | 82.90 |
| **98.22** | **89.11** | **82.33** | **76.51** | **88.93** |

Table S14. Comparison (%) of FSOR methods on MiniIN.

| 5-way $N$-shot | 1-shot | | 5-shot | |
|---|---|---|---|---|
| (RN50) | ACC | AUROC | ACC | AUROC |
| SEMAN-G [11] | 68.24 | 72.85 | 83.48 | 82.07 |
| GEL [42] | 68.26 | 73.70 | 83.05 | 82.29 |
| OSLO [3] | 71.73 | 74.92 | 83.40 | 82.59 |
| UTL (Our) | **89.93** | **87.05** | **94.03** | **89.59** |

$\{0.1, 0.3, 0.5, 0.7, 0.9\}$ and set $\varepsilon$ as 0.1, the number of unknown words $N$ as 1 on TINY. As shown in Tab. S11, we observe that the variation of $\lambda$ has a limited impact on closed performance, and the AUROC slightly declines with the increase of $helambda$, we set $\lambda$ as 0.9 on the various benchmarks.

**Number of universal word basis $\boldsymbol{F}$.** After randomly selecting $T = 1000$ words, U$^2$WO employs Top-$F$ bases of PCA as universal word basis, and we assess the effect of $F$ by varying $F \in \{20, 30, 70, 100, 200, 300\}$ with $\varepsilon = 0.001$ and $N = 5$ on IN-Easy dataset. As shown in the third part of Tab. S12, we observe that $F$ has little impact on the final performance, and $F = 100$ generally leads to the best performance. We set $F$ to 100 throughout all the experiments.

**MSP *vs.* MLS**. Taking the Maximum Logits Score (MLS) as a confidence score, we can extend the classifier as:

$$\hat{y} = \begin{cases} \underset{k\in[1,K]}{\arg\max}\, \mathrm{sim}\left(\mathcal{T}\left(\boldsymbol{t_k}\right), \mathcal{I}(\boldsymbol{x})\right), & \text{if similarity} \geq \theta \\ \text{unknown}, & \text{otherwise}, \end{cases} \tag{9}$$

where $\theta$ is the preset threshold. To evaluate the performance of the different score rules for our UTL, we conduct experiments with a backbone of ViT-B/16 on various benchmarks. The open performance with different score functions is presented in Tab. S13; the MLS score achieves better performance on average. Particularly, the MLS score is more suitable for large-scale benchmarks, while the MSP score shows better performance on small benchmarks. Therefore, we adapt the MSP score on the standard FSOR benchmark and the MLS score on the challenging benchmark. To classify the unknown sample, as illustrated in Eq. (4), the MLS score measures the similarity between the text feature$[\boldsymbol{V}, \boldsymbol{C}_k]$ and the image feature $\boldsymbol{x}$, where $\boldsymbol{V}$ is the learned open context, which directly impacts the MLS score. The open context is learned based on unknown words. Therefore, the learned unknown words and the learned open contexts will help classify 'unknown' samples. The MSP score in Eq. (2) is defined as the maximum softmax probability of known classes, whereas the softmax probability is defined in Eq. (4), where it contains the unknown words $U$ and their open context $V$ in the denominator.

**Comparison with methods under meta-learning setting for FSOR.** We compare two FSOR methods (i.e., SEMAN-G [11] and GEL [42], OSLO [3]) on MiniImageNet by following their settings. As shown in Tab. S14, our UTL outperforms them by a large margin, owing to strong pretrained models and the technical superiority of our UTL.