# SegPrompt: Boosting Open-world Segmentation
# via Category-level Prompt Learning

Muzhi Zhu[1],   Hengtao Li[1],   Hao Chen[1]*,   Chengxiang Fan[1],   Weian Mao[2,1],   Chenchen Jing[1],
Yifan Liu[2],   Chunhua Shen[1,3]

[1] Zhejiang University, China     [2] The University of Adelaide, Australia     [3] Ant Group

## Abstract

*Current closed-set instance segmentation models rely on pre-defined class labels for each mask during training and evaluation, largely limiting their ability to detect novel objects. Open-world instance segmentation (OWIS) models address this challenge by detecting unknown objects in a class-agnostic manner. However, previous OWIS approaches completely erase category information during training to keep the model's ability to generalize to unknown objects. In this work, we propose a novel training mechanism termed* **SegPrompt** *that uses category information to improve the model's class-agnostic segmentation ability for both known and unknown categories. In addition, the previous OWIS training setting exposes the unknown classes to the training set and brings information leakage, which is unreasonable in the real world. Therefore, we provide a new open-world benchmark closer to a real-world scenario by dividing the dataset classes into known-seen-unseen parts. For the first time, we focus on the model's ability to discover objects that never appear in the training set images.*

*Experiments show that SegPrompt can improve the overall and unseen detection performance by 5.6% and 6.1% in AR on our new benchmark without affecting the inference efficiency. We further demonstrate the effectiveness of our method on existing cross-dataset transfer and strongly supervised settings, leading to 5.5% and 12.3% relative improvement. Code and data are released at:*
  *https://github.com/aim-uofa/SegPrompt*

## 1. Introduction

Datasets are one of the most important driving force for deep learning. For general foundation models, massive paired text-image data harvested from the internet is a convenient and valuable resource. However, for instance-level

perception tasks such as detection and segmentation, it is still very challenging to collect datasets at similar scales because every instance requires pixel-level annotations. Increasing the number of categories will introduce ambiguity and cause unrealistic manual labor. Category-level annotations prevail in these tasks, and an object can be discovered only if it is classified into one of the known classes in the training set. One promising solution is to decouple detection and classification tasks to increase the semantic complexity. For example, in open-vocabulary segmentation [3, 8, 28], a class-agnostic segmentation network is often adopted to locate objects, and the classification part relies on a vision language model. Under this framework, the ability to discover novel objects becomes crucial. In other words, we need to figure out how to discover more novel objects given the appropriate annotations. This task fits into the definition of open-world instance segmentation (OWIS) [23], where the model is required to not only segment the known categories but also the unknown categories.

Recent works [18] have shown that class-agnostic training encourages cross-dataset generalization. Previous works on OWIS also have agreed that class-aware training harms the model's ability to generalize to unknown objects. Thus, they all completely discard category information in the training phase. Consistent with this, our preliminary experimental results indicate that a stronger closed-world model generalizes better to unseen classes and class-agnostic training further improves the results. Specifically, Mask2Former [1] surpasses Mask R-CNN by 3.9% AR and class-agnostic training leads to another 2.5% AR improvement. All phenomena show that the direct inclusion of category information damages the generalization ability of the model.

In this work, we explore an interesting question: *is the category information really useless*? To make use of the category information while maintaining the generalization ability, we propose a prompt learning mechanism— **SegPrompt**—which serves as a training time auxiliary supervision to improve class-agnostic segmentation quality in

---

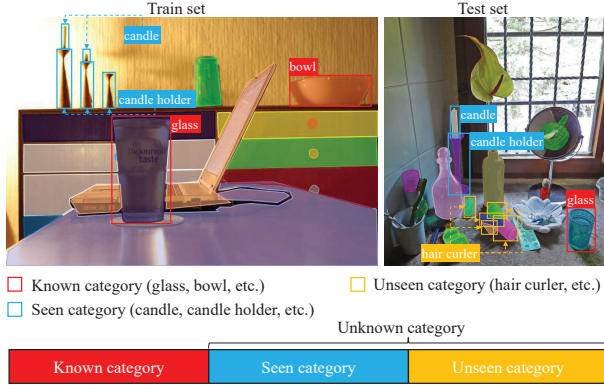*HC is the corresponding author. WM was visiting Zhejiang University.

Figure 1: **Our new benchmark, LVIS-OW.** We divide the object categories into three parts and mark them with different colored boxes. 'Known' represents categories that are labeled in the training set, 'Seen' represents categories that appear in the training set but are not labeled, and 'Unseen' represents categories that never appear in the training set.

general. The idea is to enhance category-level appearance representation learning while keeping the class-agnostic branch unaffected. The loose analogy can be drawn between our method and visual prompt tuning (VPT) [10]. While VPT is an efficient tuning protocol to transfer pre-trained models to downstream tasks, SegPrompt can be viewed as an efficient representation learning strategy that extracts category-level representations directly from images and corresponding masks as prompts and encourages the model to better learn the appearance differences between different categories. In this process, the model is able to better extract instance-level representations and facilitate class-agnostic segmentation. Finally, we demonstrate for the first time that *category information is actually beneficial for OWIS.*

In order to better evaluate OWIS models, we propose a more refined and reasonable benchmark. The existing cross-categories benchmarks [9, 11, 22] divide the categories of the dataset into known and unknown parts (*e.g.*, for COCO, 20 base categories and 60 target categories). However, in their setup, almost all of the unknown categories appear in the images of the training set, which can lead to inevitable information leakage. They also ignore the fact that in the real open world, there are many classes of objects that never appear in the training set. The cross-dataset benchmarks [22, 23] also face the same problem. What's worse is that the improvement on those benchmarks does not even reflect stronger open-world generalization, as it may also come from performing better on known categories (due to large categories overlap between datasets).

To address the above issues, we provide a new open-world benchmark that is closer to a real-world scenario by dividing the dataset classes into *known-seen-unseen* parts

and re-organizing the train-validation split for COCO and LVIS datasets. See Figure 1. Specifically, the 'seen' represents objects that appear in the training set but are not annotated, and the 'unseen' represents real-world long-tail categories that never appear in the training images. It is worth mentioning that the existing approaches [17, 22, 29] often rely on pseudo-labels, which only improve the seen categories, and no work in literature has explored whether these methods are indeed effective on unseen categories. We are the first to emphasize the model's ability to detect unseen objects that never appear in training images.

Experiments show that our approach SegPrompt can improve the overall/unseen detection by 5.6% and 6.1% in AR. On existing cross-dataset transfer and strongly supervised setting, it yields 5.5% and 12.3% relative improvement. We show that these prompts can be generated from word embeddings and example masks, extending our methods to open vocabulary and few-shot segmentation tasks. Our main contributions can be summarised as follows:

1. An auxiliary training mechanism, SegPrompt, which effectively uses category information to improve class-agnostic segmentation on various benchmarks including open-world, cross-dataset transfer, and strongly supervised segmentation.

2. A new benchmark, LVIS-OW, for open-world segmentation which explicitly separates "known", "seen" and "unseen" categories and aligns with real-world long-tail object discovery challenge. For the first time, we focus on the "unseen" categories that never appear in training images.

3. Experiments demonstrate that category-level prompts do have the ability to encode corresponding category appearance representations and control mask generation, and have the potential to extend to open-vocabulary and few-shot segmentation.

## 2. Related Work

**Open-vocabulary segmentation.** Recently, many works focus on open-vocabulary segmentation [2, 3, 4, 8, 12, 13, 28]. ZSSeg [28] establishes a simple baseline for classifying class-agnostic mask proposals via CLIP [19] image encoder and text encoder. OpenSeg [4] further exploits image-level supervision, but its category information is only input in the region word grounding part. In addition, XPM [8] and Ding et al. [3] further extend the task of semantic segmentation to instance segmentation and panoptic segmentation. Existing works [2, 3, 4, 8, 28] tend to build on the assumption that class-agnostic segmentation networks are already able to find all objects on the image. These methods usually require additional annotations, *e.g.*, captions, for unknown classes and focus more on improving classification rather than enhancing the quality of segmentation. They
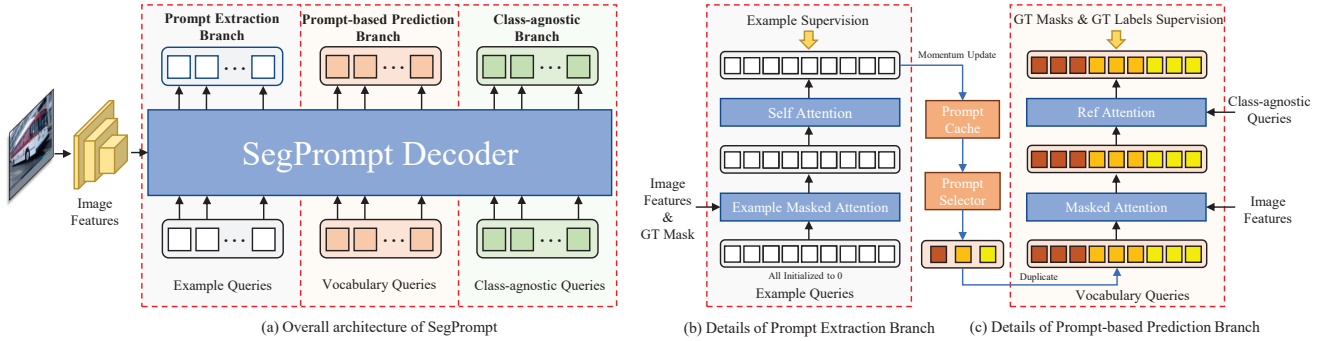
Figure 2: **The framework of SegPrompt**. (a) The transformer decoder of our framework consists of three branches, a class-agnostic branch, a prompt extraction branch, and a prompt-based prediction branch. All three branches share the same weight. The class-agnostic branch performs inference while the two other prompt branches are auxiliary supervision branches and can be removed during inference. Specifically, (b) the prompt extraction branch is used to perform prompt extraction and updates the prompt cache. (c) The prompt-based prediction branch performs prediction based on the prompt selected by the prompt selector as an auxiliary training module.

also tend to introduce too many classes in the training set, so the model focuses on semantic relations between categories and does not really reflect the ability to find unknown objects. Here, we want to argue that discovery capability is currently the main bottleneck of the task. Our work, on the other hand, is based only on limited annotations and is more concerned with how to make better use of the category information to improve class-agnostic segmentation.

**Open-world instance segmentation.** Open-world segmentation aims to segment all objects in an image, regardless of whether their classes have been seen during training. Wang et al. [23] first propose this task and designs the corresponding benchmark and UVO dataset. Kim et al. [11] and Qi et al. [18] show that forcing the model to learn classification easily hampers the model's ability to generalize across categories.

Thus, a class-agnostic segmentation network is actually more suitable for the open-world setting. Qi et al. [18] propose a new task called entity segmentation, which requires all things and stuff to be segmented out, and its model is improved based on fully-convolutional architecture. GGN [22] learns pairwise affinity to generate pseudo labels, further improving the performance of the model. SOIS [29] proposes a cross-task consistency loss to mitigate the impact of noise in the labels, and also performs semi-supervised training. All of the above works have removed the category information on the training sets, and most of them [17, 22, 29] have used pseudo-labels for self-training. Our work shows that designing a suitable network to decouple the mask and category can effectively improve the network's generalization ability to unseen classes by simply adding limited category information.

**Text-supervised/unsupervised segmentation.** Training based on limited pixel-level annotations inevitably affects the ability of the model to generalize in the open world.

Thus, many methods [7, 24, 27, 30] have emerged that are not based on pixel-level annotations at all. GroupViT [27], which groups semantic regions through image-text comparison learning, can be successfully transferred to the task of semantic segmentation without fine-tuning. Zhou et al. [30], directly choose to use a large-scale image-text pre-training model with a slight adjustment and the model is able to perform semantic segmentation without any training. FreeSOLO [24], successfully achieves unsupervised instance segmentation built upon SOLO [26] instance segmentation framework and pre-trained features from DenseCL [25].

However, not introducing pixel-level labeling at all will undoubtedly lead to very poor segmentation accuracy, which is hardly usable for practical applications in the open world. Therefore, we provide a method that can use both mask strong supervision and image-level labels for fine-granularity representation learning, which can guarantee the accuracy of segmentation and make full use of the information of image-level labels to improve the generalizability of the model and make real open world segmentation possible.

## 3. Methodology

### 3.1. Overview

Open-world instance segmentation aims to segment all instances in the image, including those belonging to categories that were not present in the training phase. The existing works [22, 29] use a class-agnostic segmentation model to accomplish this task. In this work, we propose a prompt learning mechanism where only few prompt-related parameters are specifically supervised by categorical labels in a separated branch so that instance-level representation can be improved without affecting the class-agnostic branch.

Specifically, as shown in Figure 2, our prompt learn-

ing mechanism consists of a prompt extraction branch and a prompt-based prediction branch. The prompt extraction branch provides category-level instance features as prompts and the prompt-based prediction branch then uses these prompts to predict corresponding per-class instance masks. Except for few prompt-related embeddings, all parameters are shared with the class-agnostic segmentation network. This allows the class-agnostic branch to better model instance-level representations and only marginal overhead is added in the training. Furthermore, the whole process is auxiliary and does not affect the original inference process.

## 3.2. Class-agnostic Baseline Branch

Our auxiliary prompt learning mechanism is flexible to be applied to any transformer-based perception decoders since the prompts are only appended to the decoder queries and the model structure is kept. We choose masked attention based framework Mask2Former [1]. Thus the prompt features can be naturally extracted with mask guidance. The $l$-th decoder layer consists of two parts. Guided by the mask predictions from the previous layer $\mathbf{M}^{l-1}$, a masked attention module aggregates information from the image feature $\mathbf{X}$ to refine the queries $\mathbf{q} \in \mathbb{R}^{N \times D}$:

$$\mathbf{q}_{\text{mask}}^l = \text{MaskAttn}\left(\mathbf{q}^{l-1}, \mathbf{X}, \mathbf{M}^{l-1}\right). \tag{1}$$

Then the queries are updated by a following self-attention layer:

$$\mathbf{q}^l = \text{SelfAttn}\left(\mathbf{q}_{\text{mask}}^l\right). \tag{2}$$

Queries from the last layer $\mathbf{q}^L$ are used to predict the final masks and corresponding binary scores. This branch has exactly the same supervision as the original Mask2Former except that we treat all ground-truth as one class.

## 3.3. Prompt Extraction Branch

We maintain a prompt cache $\mathbf{P} = \{\mathbf{p}_c | c = 1, ..., C\}$ for the $C$ known categories where $\mathbf{p} \in \mathbb{R}^D$ has the same dimension as the instance queries. To extract the corresponding prompt feature $\mathbf{p}_c$ for each instance from class $c$ in the training sample, a set of zero-initialized example queries $\mathbf{e}^0 = \{\mathbf{e}_c^0\}$ is added to the baseline decoder and use the output queries $\mathbf{e}^L$ as the extracted prompts. Differently from the baseline branch, the ground-truth masks $\mathbf{M}_{\text{gt}}$ are used in the masked attention.

$$\mathbf{e}_{\text{mask}}^l = \text{MaskAttn}\left(\mathbf{e}^{l-1}, \mathbf{X}, \mathbf{M}_{\text{gt}}\right). \tag{3}$$

We call this module **example masked attention** because it aggregates the information within a given mask and enables the model to segment unseen objects with the extracted prompts with only a few example masks. The self-attention module of the baseline branch is reused to refine the example queries,

$$\mathbf{e}^l = \text{SelfAttn}\left(\mathbf{e}_{\text{mask}}^l\right). \tag{4}$$

Notice that example queries $\mathbf{e}$ do not share the attention map with the baseline queries $\mathbf{q}$.

We also supervise each example query using mask loss, requiring the segmentation result of each query to be as consistent as possible with its corresponding ground-truth mask used in example masked attention, which we refer to as **example supversion**.

Finally, the example queries from the last layer $\mathbf{e}_c^L$ update the corresponding prompt $\mathbf{p}^c$ if they belong to the same category $c$ in a moving average fashion. For each image, we use at most 5 example queries, and we prioritize different classes:

$$\mathbf{p}_c \leftarrow m\mathbf{p}_c + (1 - m)\mathbf{e}_c^L, \tag{5}$$

where the momentum $m$ is set to 0.9.

## 3.4. Prompt-based Prediction Branch

By feeding the extracted prompts $\mathbf{p}^c$ to the decoder, we enforce the model to detect objects from the corresponding class $c$ in the image, namely prompt-based predictions. During training, it acts as a parallel branch and provides auxiliary class-aware supervision without affecting the main branch.

For each training sample, a prompt selector selects and duplicates the prompts to create candidate queries $\mathbf{v} \in R^{C_{\text{max}} K \times D}$ for the prompt-based predictions, where $C_{\text{max}}$ is the maximum number of categories can be predicted per image and $K$ is the maximum predictions per category. If the sample contains fewer categories $C_{\text{gt}} < C_{\text{max}}$, $C_{\text{neg}} = C_{\text{max}} - C_{\text{gt}}$ negative categories are uniformly selected from all negative classes. For each selected class $c$, the $k$-th candidate query $\mathbf{v}_{ck}$ is initializes as follows,

$$\mathbf{v}_{ck}^0 = \mathbf{p}_c + \mathbf{r}_k + \mathbf{s}_c, \tag{6}$$

where $\mathbf{r}_k \in R^D$ and $\mathbf{s}_c \in R^D$ are learnable embeddings. $\mathbf{r}$, the intra-class embedding, is shared across categories and provides an intra-class imbalance allowing queries belonging to the same category to focus on different objects, while $\mathbf{s}$ is class-specific embedding. The prompt-based prediction branch shares masked attention with the baseline branch but uses a modified self-attention module, termed reference attention $\text{RefAttn}$. In reference attention, every class candidate query $\mathbf{v}_c$ is separately encoded by the same multi-head attention layer whose weight is shared with the self-attention layer, $\text{SelfAttn}$, in the baseline model. Differently from the $\text{SelfAttn}$, $\text{RefAttn}$ not only encodes the relationship between the queries in $\mathbf{v}_c$, but also leaks the information of baseline queries $\mathbf{q}$ into $\mathbf{v}_c$. Specifically, $\mathbf{v}_c$ is taken as the query for the multi-head attention layer, and $[\mathbf{q}; \mathbf{v}_c]$ is taken as the key and value. The $\text{RefAttn}$ can be demonstrated by:

$$\mathbf{v}^{updated} = \text{RefAttn}\left(\mathbf{v}\right) = \text{Concat}\left(\mathbf{v}_0', ..., \mathbf{v}_{C_{max}}'\right),$$
$$\text{where } \mathbf{v}_c' = \text{MultiHead}\left(\mathbf{v}_c, [\mathbf{q}; \mathbf{v}_c], [\mathbf{q}; \mathbf{v}_c]\right) \tag{7}$$

where $\mathbf{v}^{updated}$ are the candidate quires updated by RefAttn. By doing so, the candidate queries can fully utilize the information of the class-agnostic branch. Meanwhile, the candidate queries are invisible to the candidate queries whose class are different, in order to prevent the model from learning trivial solutions such as some categories always appearing in pairs, and to avoid unnecessary interference from negative sample categories at the same time.

For the loss calculation, we adopt the intra-class many-to-one matching strategy. For queries corresponding to a category $\mathbf{c}$, if $\mathbf{c}$ is a negative sample, only the classification loss is calculated and it is not involved in the matching, and its ground-truth label is set to 0. If $\mathbf{c}$ is a positive sample, those queries are matched to the ground-truth instances whose categories are $\mathbf{c}$. After the matching, the ground-truth labels of matched queries are set to 1 and the mask loss is calculated with the matched ground-truth mask. The ground-truth labels of unmatched queries are set to 0, and the mask loss is not calculated.

## 4. Benchmark

### 4.1. Task Formulation

Open-world instance segmentation is to segment all the object instances of any class, regardless of whether the class appears in the training phase [18, 23, 29]. Formally, in open-world instance segmentation, the training set $\mathcal{D}_{train}$ contains various classes of densely annotated objects, denoted as $C_{train}$. Models trained on $\mathcal{D}_{train}$ are supposed to segment all the objects that appear in the images of the test set $\mathcal{D}_{test}$, which contains a set of classes $\mathcal{C}_{test}$. Generally, we have $\mathcal{C}_{train} \subset \mathcal{C}_{test}$.

Previous works [11, 22] divide the categories of the dataset into known and unknown. As mentioned in section 1, by splitting a specific dataset or using a cross-dataset setting, they guarantee $C_{train} = C_{known}$ and $C_{test} = C_{known} + C_{unknown}$. However, almost all of the unknown categories appear in the images of the training set, which leads to information leakage.

To address this issue, we divide the object categories into three parts: 'known' $\mathcal{C}_{known} = \{c_k\}_{k=1}^{K}$, 'seen' $\mathcal{C}_{seen} = \{c_s\}_{s=1}^{S}$ and 'unseen' $\mathcal{C}_{unseen} = \{c_u\}_{u=1}^{U}$.

The known objects in $\mathcal{C}_{known}$ are those with dense annotations and category information in the training set $\mathcal{D}^{train}$, while the objects in $\mathcal{C}_{seen}$ appear on the image of the training set but are not annotated. By contrast, the unseen objects do not appear in the training set at all. Thus, we have $C_{train} = C_{known}$ and $\mathcal{C}_{test} = \mathcal{C}_{known} \cup \mathcal{C}_{seen} \cup \mathcal{C}_{unseen}$.

### 4.2. Dataset Construction

**Base datasets.** We construct our dataset based on the COCO dataset [14] and the LVIS dataset [5]. The COCO

| | Train | | Test | | |
| | Known | Seen | Known | Seen | Unseen |
|---|---|---|---|---|---|
| $N^{cls}$ | 123 | 743 | 123 | 738 | 337 |
| $N^{ins}$ | 1,511,257 | 452,524 | 168,798 | 99,230 | 6,457 |

Table 1: **The statistics of the proposed dataset**. The $N^{cls}$ and the $N^{ins}$ are the number of classes and instances.

dataset is a widely used large-scale instance segmentation benchmark. COCO contains 118k images for training, 5k images for validation, and 41k images for testing. 80 categories of objects are annotated in the dataset. LVIS is recently proposed for large vocabulary instance segmentation. Sharing the same images with COCO, LVIS has 1203 categories. LVIS re-splits the images of COCO and contains 100k/20k images for training/validation, respectively. The categories of LVIS are divided into three groups: frequent, common, rare, based on the number of training images.

**Data organization.** Previous works [11, 22] based on both COCO and LVIS use the images of the training/validation split of LVIS as the training/testing set, but use annotations of COCO/LVIS for the training/testing images. To divide the unknown classes into the seen and unseen, we further remove the images containing objects of unseen classes of the training set and add them to the testing set. In the following, we illustrate how we determine the three groups of classes.

For the $\mathcal{C}_{known}$, we first choose the 80 categories of COCO because of the high quality of annotations of COCO. Considering there are obvious annotation granularity differences between COCO and LVIS, we add annotations of some frequent categories in LVIS to enable the model to segment objects at a finer granularity. Specifically, for an object with the label "person" in COCO, some parts of it will be labeled as "apron", "hat", etc., in LVIS. If only COCO is used for training, the model can locate a whole person, but can not segment the apron or hat of the person, which makes the OWIS task intractable. Thus we select the most frequent 64 categories of LVIS and the 80 categories of COCO to compose the known classes $\mathcal{C}_{known}$.

For the $\mathcal{C}_{unseen}$, we conduct experiments to explore the impact of removing images from the training set, and select the 337 classes of objects, which belongs to the rare split of LVIS, as the unseen classes $\mathcal{C}_{unseen}$. Only less than 1% images of the training set are removed. The details can be found in the supplementary material. The other classes (except the known and unseen classes) of LVIS are regarded as seen classes $\mathcal{C}_{seen}$ of our dataset.

**Dataset analyses.** After the re-splitting of COCO and LVIS, the training/testing set of the LVIS-OW dataset contains 98708/21271 images. Table 1 shows detailed statistics. Note that the known subset contains 123 rather than 144 classes due to the overlap between COCO and LVIS.

Besides, some classes belonging to the seen subset of the training set do not appear in the test set. But this does not affect the model training and evaluation.

## 4.3. Metrics

To fully evaluate the ability of models to segment unseen objects, we compute the class-wise mean Average Recall (AR) on different sets as $AR_{known}$, $AR_{seen}$, $AR_{unseen}$, and $AR_{all}$. For a particular class, we use all class-agnostic predictions and ground-truth instances belonging to the class to compute the AR, as [20]. Subsequently, the ARs of classes of a specific subset are then averaged to obtain the AR on the subset. Considering the number of instances in each subset is of different orders of magnitude, this division also helps to alleviate the impact of class imbalance.

We do not use the class-agnostic AR and the mean Average Precision (AP) as previous works [22, 29], because the two metrics are not suitable to evaluate open-world models on datasets with a large vocabulary such as LVIS. Firstly, AP is unable to properly measure the ability of models to discover unseen classes. Objects that are not in the test set annotation but found by the model will be counted as false positives, and result in a lower AP. Secondly, class-agnostic AR will be dominated by common classes with more instances, and thus can not evaluate the ability of models on long-tailed categories. For simplicity, the class-wise mean $AR_{cls}$ is referred to as $AR$ in later sections.

## 5. Experiments

### 5.1. Implementation Details

All experiments are conducted on eight Nvidia A100 GPUs. ResNet50 is used as the backbone for all models. Most of experimental settings follow Mask2Former [1], including the learning rate, weight decay, data augmentation and batch size. Unless specified, the number of object queries for the class-agnostic branch is 100, while the total number of object queries for the prompt-based prediction branch is 300.

| Method | $AR_{all}$ | $AR_{kn}$ | $AR_{se}$ | $AR_{un}$ | $AR_s$ | $AR_m$ | $AR_l$ |
|---|---|---|---|---|---|---|---|
| Mask R-CNN | 36.5 | 46.5 | 32.8 | 41.0 | 24.2 | 44.3 | 57.0 |
| Mask R-CNN† | 39.9 | 46.6 | 36.8 | 44.0 | 28.3 | 49.3 | 57.3 |
| Mask2Former | 40.8 | 50.2 | 37.4 | 44.9 | 25.5 | 50.0 | 67.9 |
| Mask2Former† | 42.5 | 49.9 | 39.0 | 47.4 | 27.2 | 51.5 | 68.1 |
| LDET [20] | 39.3 | 40.7 | 37.6 | 42.6 | 27.4 | 49.2 | 55.7 |
| GGN [22] | 42.3 | 42.6 | 40.3 | 46.6 | 28.3 | 52.9 | 62.1 |
| Ours | **44.9** | **52.2** | **41.2** | **50.3** | **29.4** | **54.5** | **70.4** |

Table 2: **Class-wise mean AR comparison of different methods.** † denotes class-agnostic training. Our prompt learning mechanism improves the performance on known and unseen categories by 2.3 AR and 2.6 AR for free.

## 5.2. Main Results

**Experimental results on our benchmark.** In Table 2, we compare our method with the two most representative strong baselines Mask R-CNN [6] and Mask2Former [1] on our benchmark. Notably, Mask2Former outperforms Mask R-CNN in the vast majority of metrics but is significantly weaker than Mask R-CNN in $AR_s$ instead. We also demonstrate that class-agnostic training does significantly improve generalization to unknown categories, with about 2% to 4% improvements in both $AR_{seen}$ and $AR_{unseen}$, but for known categories, the difference is actually very small, and can be considered as a fluctuation. In contrast, our method further improves generalizability by effectively using the category information instead, with very significant improvements in all metrics, especially $AR_{seen}$ and $AR_{unknown}$. It is worth mentioning that many of the works [2, 3, 8, 28] can actually be attributed to the class-agnostic Mask2Former [1] model in Table 2 or a weaker network structure. Our experiments show that, even under the same amount of data, there is still a lot of room for improvement in the performance of Mask2Former. How to use limited annotations to improve the performance of the model is still a topic worth studying.

**Evaluation on other benchmark.** To verify the effectiveness and universality of our method, we perform experiments following the cross-dataset evaluation fashion proposed by the work [20, 29], as shown in Table 4. All the models are trained on COCO dataset but evaluated in LVIS dataset, denote as COCO→ LVIS. The experiment results show that our method outperforms the current the-state-of-the-art method SOIS [29] in this setting. Both SOIS and our method are based on Mask2Former [1] methods, but SOIS also performs pseudo labeling and self-training methods, while our method does not need to perform self-training so if the number of training iterations is the same for both rounds, our training overhead is only half. We also show the results of whether to introduce example supervision or not, as we find that while introducing example supervision leads to an overall performance improvement, $AP_l$ decreases. This is consistent with the results in Table 5, where we speculate that example supervision allows the model to focus better on small objects, but not on large objects.

**Training on the full LVIS training set.** We further validate the effectiveness of our method when the fully supervised training on the full LVIS dataset is performed, as shown in Table 3. Compared to Mask2Former, Our method achieves a leap in all metrics when all the settings are strictly aligned, including the training data, the number of training iterations and so on. This shows that our method has a very obvious advantage in performing class-agnostic segmentation training for such a complex scenario in terms of object types and numbers.

Figure 3: **Qualitative results on OWIS**. We compare our method with Mask2Former [1], GGN [22], and X-Decoder [31] (X-Decoder is an open-vocabulary model, so we add LVIS class texts as guidance) and select top 30 masks for visualization. The additional objects detected by our method are highlighted with red boxes. All models except X-Decoder are trained using COCO annotations only.

| Method | $AP@100$ | $AR@100$ | $AR_s$ | $AR_m$ | $AR_l$ | $AR_{all}$ |
|---|---|---|---|---|---|---|
| Mask2Former[†] | 20.3 | 40.9 | 26.2 | 61.5 | 78.5 | 50.8 |
| Mask2Former | 20.1 | 41.7 | 26.4 | 63.0 | 80.8 | 53.8 |
| Ours | **22.8** | **45.7** | **30.2** | **68** | **84.2** | **57.4** |

Table 3: **Comparison on standard LVIS dataset.** [†] denotes class-agnostic training. Except $AR_{all}$, which is our metric, all other metrics are from the original LVIS class-agnostic evaluation. Our method significantly improves all the metrics by at least 2%.

| Method | $AR_{100}$ | $AP_{100}$ | $AP_s$ | $AP_m$ | $AP_l$ |
|---|---|---|---|---|---|
| Mask R-CNN [29] | 22.4 | 6.5 | 3.2 | 10.4 | 17.6 |
| LDET [29] | 25.1 | 6.5 | 2.8 | 10.6 | 18.4 |
| SOLO2 [29] | 21.7 | 7.5 | 3.1 | 12.4 | 22.4 |
| SOLO2 + SOIS [29] | 22.8 | 7.9 | 3.8 | 13.1 | 23.8 |
| M2F + SOIS [29] | 25.2 | 8.5 | 3.4 | 13.8 | 26.4 |
| GGN [22] | 27 | 6.5 | 3.5 | 12.1 | 14.9 |
| Ours★ | 26.5 | 9.3 | 4.0 | **15.1** | **28.0** |
| Ours | **28.5** | **9.4** | **4.6** | **15.1** | 24.5 |

Table 4: **Comparison on cross-dataset evaluation setting, COCO→LVIS.** The models are trained on COCO dataset but evaluated on LVIS dataset. The original LVIS evaluation metric is employed. Our method performs better over all metrics. For GGN, the original paper did not report the results under this setting, so we used their model trained under COCO to re-evaluate. ★, means we do not use example supervision.

## 5.3. Ablation Study

In the ablation study, we carefully investigate the design of each component we proposed. The number of the maximum prediction per category $K$ is set to 10 and the maximum number of categories $C_{max}$ that can be predicted per image is set to 30, if not specified. More ablation experiments are available in the supplementary material.

**Auxiliary supervision methods.** The proposed prompt learning is used as an auxiliary supervision to supervise the model with label information. However, there are many different auxiliary supervision methods to introduce the category information to the model. Here, we explore a straightforward auxiliary supervision method to compare with SegPrompt, as shown in Table 5. The straightforward

method is that a vanilla Mask2Former is used in the auxiliary supervision branch for classification, similar to the prompt-based prediction branch, while the weights of the vanilla Mask2Former are shared by the class-agnostic baseline branch except for the classification head.

The main difference between this straightforward method and SegPrompt is that each query in this method does

| Aux. Sup. | $AR_{all}$ | $AR_{kn}$ | $AR_{se}$ | $AR_{un}$ | $AR_s$ | $AR_m$ | $AR_l$ |
|---|---|---|---|---|---|---|---|
| None | 42.5 | 49.9 | 39.0 | 47.4 | 27.2 | 51.5 | 68.1 |
| Vanilla M2F | 43.0 | 51.4 | 39.3 | 47.9 | 27.6 | 52.0 | 69.7 |
| Ours⋆ | 44.1 | 51.9 | 40.6 | 49.1 | 28.9 | 53.3 | **70.6** |
| Ours | **44.9** | **52.2** | **41.2** | **50.3** | **29.4** | **54.5** | 70.4 |

Table 5: **Investigating the auxiliary supervision methods.** For 'None', the auxiliary supervision is removed from the model. For 'Vanilla M2F', a standard Mask2Former is used as an auxiliary supervision and shares weights with the baseline model. ⋆ means that we do not use example supervision.

not specify a corresponding category, but matches with the ground-truth of all categories directly in the matching phase for multi-category prediction. The experiment results demonstrate that our method outperforms the vanilla Mask2Former significantly and the prompt learning is the key to take advantage of category information without losing unknown class generalizability. In Table 5, we also compare the impact of introducing example supervision or not, and the results show that adding example supervision can further improve the model's performance.

**The way to generate the candidate queries.** As mentioned in Section 3.4, both the prompt $\mathbf{p}$ and class-specific embedding $\mathbf{s}$ are important components to generate the candidate queries $\mathbf{v}$. Here, we conduct experiments to verify the necessity of the class-specific embedding $\mathbf{s}$ and the prompt $\mathbf{p}$ by removing them from the candidate queries $\mathbf{v}$, separately.

As shown in Table 6, without any of the class-specific embedding $\mathbf{s}$ or the prompt $\mathbf{p}$, the performance will drop significantly which indicates both of $\mathbf{s}$ and $\mathbf{p}$ are necessary for our model.

| Method | $AR_{all}$ | $AR_{kn}$ | $AR_{se}$ | $AR_{un}$ | $AR_s$ | $AR_m$ | $AR_l$ |
|---|---|---|---|---|---|---|---|
| Baseline | **44.1** | **51.9** | **40.6** | **49.1** | **28.9** | **53.3** | **70.6** |
| Baseline w/o s | 43.3 | **51.9** | 39.6 | 48.3 | 28.0 | 52.7 | 70.1 |
| Baseline w/o p | 42.8 | 50.8 | 39.2 | 47.7 | 27.5 | 52.0 | 69.0 |

Table 6: **Verifying the necessity of the class-specific embedding s and the prompt p.** 'Baseline' represents our model without example supervision. 'w/o s' and 'w/o p' represent removing the class-specific embedding $\mathbf{s}$ and $\mathbf{p}$ from the candidate query $\mathbf{v}$.

| Query Attention | $AR_{all}$ | $AR_{kn}$ | $AR_{sn}$ | $AR_{un}$ | $AR_s$ | $AR_m$ | $AR_l$ |
|---|---|---|---|---|---|---|---|
| SelfAttn | 35.7 | 44.1 | 31.6 | 41.4 | 22.6 | 43.2 | 58.5 |
| SelfAttn (independent) | 41.0 | 47.5 | 37.3 | 46.5 | 26.0 | 49.6 | 67.8 |
| RefAttn (inter-class visible) | 44.0 | 51.8 | 40.5 | 48.9 | 28.2 | **53.7** | 70.1 |
| RefAttn (inter-class invisible) | **44.1** | **51.9** | **40.6** | **49.1** | **28.9** | 53.3 | **70.6** |

Table 7: **Comparison of different attention designs**. SelfAttn (independent) represents the prompt-based prediction branch and class-agnostic baseline branch conduct independent self-attention, and RefAttn (inter-class visible/invisible) represents whether to make queries between classes visible when reference-attention is performed.
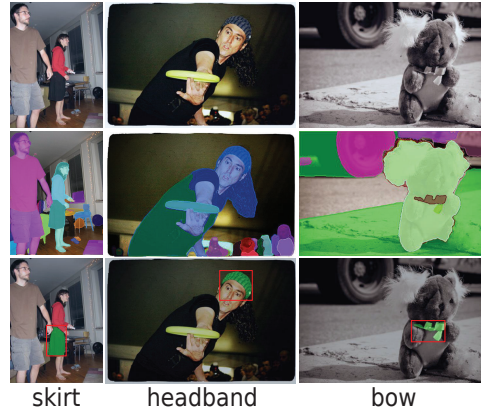


skirt  headband  bow

Figure 4: **Qualitative results on open vocabulary segmentation.** The first row is the original input image; the second row is the result of the class-agnostic branch segmentation; the third row is the result of using CLIP text embedding as a prompt.

**Reference attention.** Experiments in Table 7 are conducted to verify the effectiveness of reference attention RefAttn. In the experiments, we compare the proposed approach with three other designs. First, we use the global self-attention making the class-agnostic query and vocabulary query fully visible to each other. This definitely causes information leakage and impairs the segmentation ability of category-independent branches. Second, we let the two branches perform completely independent self-attention, which also does not facilitate class-agnostic segmentation. Next, we explore the impact of visibility between different classes of candidate queries when doing the reference attention and find no significant difference.
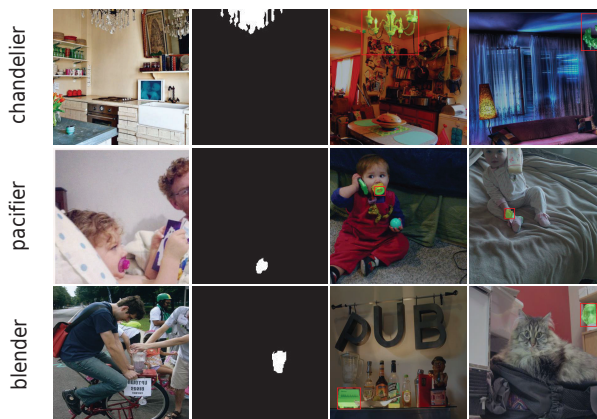


Figure 5: **Qualitative results on few-shot segmentation**. The prompts on the left side are the support images and their corresponding masks (we use the 5-shot setting, only one is shown here), followed by the segmentation results on test images.

## 5.4. Qualitative Results

We first compare our model with other methods on OWIS, see Figure 3. To verify that our framework is indeed able to encode category-level appearance representation in the prompt, we show some qualitative results when the setting is extended to few-shot segmentation [15, 16, 21] and open-vocabulary segmentation [4, 12].

**Open vocabulary segmentation.** Our framework is able to easily extend to the open-vocabulary segmentation task without changing the training process. We set the prompt of the prompt-based prediction branch to fixed CLIP [19] text embedding, without introducing the image features of CLIP. By adding the image-level labels of the seen part during training, our model is able to segment the new classes specified by the text without any pixel annotation, see Figure 4.

**Few-shot instance segmentation.** Our framework can also be naturally transferred to the few-shot segmentation task. With the help of the prompt extraction branch, we can input a limited number of new classes of masks and images to the prompt extraction branch to extract the prompt of the corresponding class. The prompt is then used to inference on other images containing this new class to obtain the mask. The visualization results are shown in Figure 5.

## 6. Conclusion

In this work, we have presented an LVIS-OW benchmark for open-world instance segmentation. We introduced **Seg-Prompt**, a novel prompt learning mechanism, for OWIS. It enables segmentation models to predict masks specified by the prompt and significantly improves the class-agnostic segmentation capability of the model. Extensive experiments demonstrated that the proposed SegPrompt can effectively discover objects of categories beyond the training phase.

## Acknowledgements

## References

[1] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 1290–1299, 2022.

[2] Jian Ding, Nan Xue, Gui-Song Xia, and Dengxin Dai. Decoupling zero-shot semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 11583–11592, 2022.

[3] Zheng Ding, Jieke Wang, and Zhuowen Tu. Open-vocabulary panoptic segmentation with maskclip. *arXiv: Comp. Res. Repository*, 2022.

[4] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *Eur. Conf. Comput. Vis.*, pages 540–557. Springer, 2022.

[5] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 5356–5364, 2019.

[6] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *Int. Conf. Comput. Vis.*, pages 2961–2969, 2017.

[7] Olivier J Hénaff, Skanda Koppula, Evan Shelhamer, Daniel Zoran, Andrew Jaegle, Andrew Zisserman, João Carreira, and Relja Arandjelović. Object discovery and representation networks. *Eur. Conf. Comput. Vis.*, 2022.

[8] Dat Huynh, Jason Kuen, Zhe Lin, Jiuxiang Gu, and Ehsan Elhamifar. Open-vocabulary instance segmentation via robust cross-modal pseudo-labeling. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 7020–7031, 2022.

[9] Jaedong Hwang, Seoung Wug Oh, Joon-Young Lee, and Bohyung Han. Exemplar-based open-set panoptic segmentation network. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 1175–1184, 2021.

[10] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. *arXiv: Comp. Res. Repository*, 2022.

[11] Dahun Kim, Tsung-Yi Lin, Anelia Angelova, In So Kweon, and Weicheng Kuo. Learning open-world object proposals without learning to classify. *IEEE Robotics and Automation Letters*, 7(2):5453–5460, 2022.

[12] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and René Ranftl. Language-driven semantic segmentation. *arXiv: Comp. Res. Repository*, 2022.

[13] Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-vocabulary semantic segmentation with mask-adapted clip. *arXiv: Comp. Res. Repository*, 2022.

[14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Eur. Conf. Comput. Vis.*, pages 740–755. Springer, 2014.

[15] Yang Liu, Muzhi Zhu, Hengtao Li, Hao Chen, Xinlong Wang, and Chunhua Shen. Matcher: Segment anything with one shot using all-purpose feature matching. *arXiv: Comp. Res. Repository*, 2023.

[16] Juhong Min, Dahyun Kang, and Minsu Cho. Hypercorrelation squeeze for few-shot segmentation. In *Int. Conf. Comput. Vis.*, pages 6941–6952, 2021.

[17] Lu Qi, Jason Kuen, Zhe Lin, Jiuxiang Gu, Fengyun Rao, Dian Li, Weidong Guo, Zhen Wen, Ming-Hsuan Yang, and Jiaya Jia. Ca-ssl: Class-agnostic semi-supervised learning for detection and segmentation. In *Eur. Conf. Comput. Vis.*, pages 59–77. Springer, 2022.

[18] Lu Qi, Jason Kuen, Yi Wang, Jiuxiang Gu, Hengshuang Zhao, Zhe Lin, Philip Torr, and Jiaya Jia. Open-world entity segmentation. *arXiv: Comp. Res. Repository*, 2021.

[19] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Proc. Int. Conf. Mach. Learn.*, pages 8748–8763. PMLR, 2021.

[20] Kuniaki Saito, Ping Hu, Trevor Darrell, and Kate Saenko. Learning to detect every thing in an open world. *arXiv: Comp. Res. Repository*, 2021.

[21] Amirreza Shaban, Shray Bansal, Zhen Liu, Irfan Essa, and Byron Boots. One-shot learning for semantic segmentation. *arXiv: Comp. Res. Repository*, 2017.

[22] Weiyao Wang, Matt Feiszli, Heng Wang, Jitendra Malik, and Du Tran. Open-world instance segmentation: Exploiting pseudo ground truth from learned pairwise affinity. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 4422–4432, 2022.

[23] Weiyao Wang, Matt Feiszli, Heng Wang, and Du Tran. Unidentified video objects: A benchmark for dense, open-world segmentation. In *Int. Conf. Comput. Vis.*, pages 10776–10785, 2021.

[24] Xinlong Wang, Zhiding Yu, Shalini De Mello, Jan Kautz, Anima Anandkumar, Chunhua Shen, and Jose M Alvarez. Freesolo: Learning to segment objects without annotations. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 14176–14186, 2022.

[25] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 3024–3033, 2021.

[26] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. SOLO: A simple framework for instance segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2021.

[27] Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. In *IEEE Conf. Comput. Vis. Pattern Recogn.*, pages 18134–18144, 2022.

[28] Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, and Xiang Bai. A simple baseline for zero-shot semantic segmentation with pre-trained vision-language model. *Eur. Conf. Comput. Vis.*, 2022.

[29] Xizhe Xue, Dongdong Yu, Lingqiao Liu, Yu Liu, Ying Li, Zehuan Yuan, Ping Song, and Mike Zheng Shou. Single-stage open-world instance segmentation with cross-task consistency regularization. *arXiv: Comp. Res. Repository*, 2022.

[30] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. In *Eur. Conf. Comput. Vis.*, pages 696–712. Springer, 2022.

[31] Xueyan Zou, Zi-Yi Dou, Jianwei Yang, Zhe Gan, Linjie Li, Chunyuan Li, Xiyang Dai, Harkirat Behl, Jianfeng Wang, Lu Yuan, et al. Generalized decoding for pixel, image, and language. *arXiv: Comp. Res. Repository*, 2022.