REVISIT THE OPEN NATURE OF OPEN VOCABULARY SEMANTIC SEGMENTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

In Open Vocabulary Semantic Segmentation (OVS), we observe a consistent drop in model performance as the query vocabulary set expands, especially when it includes semantically similar and ambiguous vocabularies, such as *'sofa'* and *'couch'*. The previous OVS evaluation protocol, however, does not account for such ambiguity, as any mismatch between predicted and human-annotated pairs is simply treated as incorrect on a pixel-wise basis. This contradicts the open nature of OVS, where ambiguous categories may both be correct from an open-world perspective. To address this, in this work, we study the open nature of OVS and propose a mask-wise evaluation protocol that is based on matched and mismatched mask pairs between prediction and annotation respectively. Extensive experimental evaluations demonstrate that the proposed mask-wise protocol provides a more effective and reliable evaluation framework for OVS models compared to the previous pixel-wise approach. Moreover, analysis of mismatched mask pairs reveals that a large amount of ambiguous categories exist in commonly used OVS datasets. Interestingly, we find that reducing these ambiguities during both training and inference enhances capabilities of OVS models. These findings and the new evaluation protocol encourage further exploration of the open nature of OVS, as well as broader open-world challenges.

1 INTRODUCTION

Open-world learning aims to address the problem of learning with novel and unknown categories or data distributions that are often encountered in the real world. With the development of large language-visual (LLV) models, such as CLIP (Radford et al., 2021), open vocabulary tasks are proposed to utilise the strong language visual alignment capability from LLV to identify objects from the data, including new entities not in the training data. Particularly, open vocabulary semantic segmentation (OVS) is a task where models trained on a close-set semantic segmentation dataset perform zero-shot inference on an unseen dataset by providing a vocabulary set for any object.

In the open world, category boundaries are often not clearly defined. For instance, when describing visual objects using text, vocabulary with high semantic ambiguity may be employed. While this works in closed-set settings, where labels are assumed to be mutually exclusive. However, in the case of open vocabulary semantic segmentation (OVS), where any vocabulary, including those with significant semantic ambiguity, can be introduced. For example, as illustrated in Fig. 1, under the current OVS evaluation protocol, predictions such as *'flower'* and *'chair'* are considered incorrect, despite appearing reasonable to humans from an open-world perspective. In this paper, we aim to address the challenges posed by such ambiguous categories by revisiting the *open nature* of OVS, trying to answer: *whether we should treat the ambiguous categories as incorrect or correct ones, and how to encourage OVS to be more open?*

Specifically, we first revisit the existing OVS evaluation process from the perspective of open-world learning, and find that it follows a closed-set approach, where predictions are considered incorrect if they do not match with the predefined category. We observe that as the number of inference categories increases, the performance of existing OVS models significantly declines, indicating issues caused by category ambiguity. To study this, we find that expanding model predictions from pixel-wise argmax to category-wise mask-wise predictions effectively mitigates such ambiguity problems. To this end, we propose an open-set prediction approach and a corresponding generalised category-

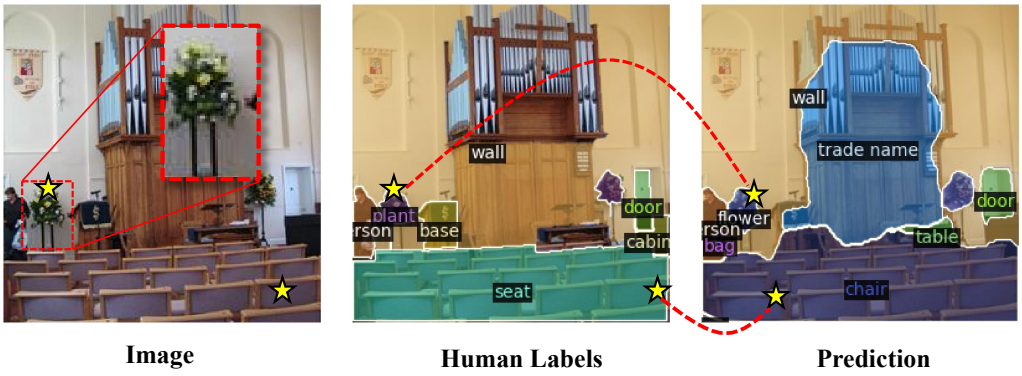

Figure 1: Category ambiguity in open vocabulary semantic segmentation. One object can be assigned multiple possible labels while the human label is only one of those plausible labels. For example, the area on the left with a yellow star was annotated as *'plant'* by humans, but predicted to be *'flower'* by the OVS model; the bottom part annotated as *'seat'* was predicted as *'chair'* by OVS model.

preserving evaluation metric. We find that the proposed new prediction and evaluation framework significantly improves the performance of existing OVS methods. Additionally, based on our proposed evaluation framework, we construct an ambiguous vocabulary graph between model predictions and human annotations, revealing clear community structures where vocabularies within the same community correspond to visually similar objects. The main contributions of this study can be summarised as follows: 1) We revisit existing OVS paradigms from an open-world perspective, offer insightful observations, and propose feasible solutions to encourage more openness in OVS. 2) Our proposed mask-wise evaluation protocol effectively addresses the issue of ambiguous categories in open-world evaluation. 3) Based on the proposed evaluation framework, we show the way to construct a confusion vocabulary graph for existing OVS datasets, highlighting the significant presence of ambiguous category annotations. 4) Extensive experimental analysis and comparisons validate the effectiveness of the proposed method, which achieves state-of-the-art performance with a new paradigm for OVS in the open world.

2 Related Works

2.1 Close-set segmentation

This task aims to segment images into regions with predefined categories. Fully Convolutional Networks (FCNs) (Long et al., 2015) marked the beginning of the deep learning era in image segmentation. Subsequently, convolution-based (Li et al., 2023) and Transformer-based (Liu et al., 2021) approaches further enhanced the model's performance in semantic segmentation (Zhao et al., 2017; Wang et al., 2020; Shen et al., 2022; Chen et al., 2017; Ronneberger et al., 2015; Xie et al., 2021), instance segmentation (He et al., 2017; Lin et al., 2014; Chen et al., 2019a; Liu et al., 2018; Qi et al., 2021), and panoptic segmentation (Kirillov et al., 2019b;a; Cheng et al., 2021; Zhang et al., 2021). Despite the continuous advancements in closed-set segmentation methods, predefined category sets are inadequate for open-world vision applications where the number of object categories is vast and constantly evolving. In this work, we focus on adapting closed-set evaluation metrics to accommodate open-world scenarios, especially for dealing with the case of ambiguous categories.

2.2 Open vocabulary semantic segmentation

Close-set segmentation aims to train the model to segment predetermined categories while OVS aims to segment the objects with arbitrary vocabulary queries (Cho et al., 2023; Xie et al., 2023; Xu et al., 2023) which enables the open ability for model prediction. There has been some recent exploratory work in this direction. LSeg (Li et al., 2022) utilised CLIP (Radford et al., 2021)

to train a visual encoder that generates pixel-level visual embeddings from an image, which are aligned with the corresponding textual embeddings learned from the training labels within the CLIP embedding space. OpenSeg (Ghiasi et al., 2022) employed a class-agnostic segmentation module utilising region-to-image cross-attention to detect local regions in images. Two-stage frameworks, ZegFormer (Ding et al., 2022) and ZSseg (Xu et al., 2022), also extract class-agnostic region proposal similar to (Ghiasi et al., 2022) at the first stage, then utilised pretrained vision-language models like CLIP to classify masked regions. Liang et al. (2023) improves CLIP's performance on masked images by finetuning CLIP on image-text pairs. CAT-Seg (Cho et al., 2023) proposed a cost aggregation method to optimise the image-text similarity map by fine-tuning the CLIP encoder and obtaining accurate pixel-level predictions. MaskCLIP (Dong et al., 2023) integrates a novel masked self-distillation technique into contrastive language-image pretraining, aiming to derive pixel-level embeddings from CLIP for immediate application in segmentation tasks. SED (Xie et al., 2023) introduced an encoder-decoder architecture, which consists of a hierarchical encoder-based cost map generation and a gradual fusion decoder with category early rejection to obtain pixel-level image-text cost map prediction. Despite the progress achieved by the previous OVS models, the existing training and inference of OVS still adhere to the pipeline of close-set recognition, *i.e.* a fixed training vocabulary set is used during training, and a particular dataset-specific vocabulary set is given during inference.

2.3 The Evaluation of OVS

The current evaluation methods for open vocabulary semantic segmentation primarily rely on the mean Intersection over Union (mIoU) metric, which assesses classification accuracy at the pixel level. Some studies (Zhou et al., 2023; Liu et al., 2023) point out that this involves strict pixel matching, where two pixels are considered correct only if they belong to the same class, which is not suitable for an open-world setting. Therefore, these works aim to account for the similarity between class vocabularies by assigning weights to each pixel when calculating IoU. In this approach, when the model predicts a class that is textually similar, a partial accuracy score is assigned based on the degree of similarity. However, textual similarity alone cannot reliably assess the similarity between two visual-semantic objects, and thus the effectiveness of weight assignment is uncertain. In this work, we focus on distinguishing visually similar classes based on visual similarity (i.e. the overlap between segmentation predictions and human annotations) and evaluating them separately.

3 Revisiting open vocabulary semantic segmentation

Definition of OVS. open vocabulary semantic segmentation (OVS) addresses the task of training a segmentation model capable of using textual descriptions to segment arbitrary objects. Given two category sets, C_{train} and C_{test}, where C_{train} and C_{test} are not equal in terms of object categories ($C_{train} \neq C_{test}$), the model is trained on C_{train} and directly tested on C_{test}. Typically, C_{train} and C_{test} are described using noun phrases (e.g. sky, ocean, mountains, etc.). During the testing stage, the previous assumption held by OVS is that it is known which dataset the test data originates from, and the whole vocabulary set corresponding to the dataset is provided as the inference vocabulary, $C_{D_1}, C_{D_2}, \ldots, C_{D_n}$.

3.1 OVS Framework

Training objective. Let's consider the Maximum A Posterior (MAP) estimate for training a deep learning model with given data X and a certain vocabulary set $\mathcal{V}$, let the prior distribution of Θ be $g(\Theta)$. We want to find a parameter Θ that maximises:

$$\Theta_{MAP} = \arg\max_{\Theta} P(\Theta|X, \mathcal{V}). \tag{1}$$

Since the training vocabulary set $\mathcal{V}$ is considered a fixed set in most previous OVS works (Cho et al., 2023; Xie et al., 2023; Xu et al., 2023), we apply Bayes' theorem to get:

$$\Theta_{MAP} \propto \arg\max_{\Theta} \log P_{\mathcal{V}}(X|\Theta) + \log g(\Theta). \tag{2}$$

Although OVS intends to incorporate vocabulary during training, the objective above fails to consider the vocabulary during the model optimisation. Here we propose to turn the training vocabulary

into a random variable represented by $P(\mathcal{V})$, as aforementioned, we have our MAP estimate for training an OVS model:

$$\begin{aligned}\hat{\Theta}_{MAP} &= \arg\max_{\Theta} P(\Theta|X,\mathcal{V}) \\ &= \arg\max_{\Theta} \frac{P(X|\Theta,\mathcal{V})P(\mathcal{V}|\Theta)P(\Theta)}{P(X|\mathcal{V})P(\mathcal{V})} \\ &\propto \arg\max_{\Theta} \underbrace{\log P(X|\Theta,\mathcal{V})}_{likelihood} + \underbrace{\log P(\mathcal{V}|\Theta)}_{language\ likelihood} + \underbrace{\log P(\Theta)}_{prior}. \end{aligned} \tag{3}$$

We notice that compared to the original objective in Eq. 2, the current optimisation incorporates another term $P(\mathcal{V}|\Theta)$ that relates to the vocabulary distribution (for a detailed mathematical derivation, please refer to the supplementary material). Considering vocabulary $\mathcal{V}$ as a random variable during training, the model parameters Θ are dependent on both the observed training data X and the vocabulary $\mathcal{V}$ through maximising the $P(\mathcal{V}|\Theta)$ and $P(X|\Theta,\mathcal{V})$ terms. This implicitly constructs the relationship between model parameters and vocabulary distribution, and could be beneficial to OVS in open world scenarios.

Zero-shot inference capability. open vocabulary semantic segmentation models can perform zero-shot inference on unseen datasets while providing customisable vocabulary. Given an image $I_i \in \mathcal{R}^{B,C,W,H}$ and vocabulary candidate set $\mathcal{V} \in \mathcal{R}^{B,D}$, the OVS model takes I_i and $\mathcal{V}$ as input, and generates a class posterior $P(y_i|X,\Theta,\mathcal{V})$. Usually the consequent semantic segmentation predictions are obtained by applying the argmax operation:

$$\hat{Y} = \arg\max_{y_i \in \mathcal{V}} P(y_i|X,\Theta,\mathcal{V}). \tag{4}$$

In zero-shot inference, a spatial posterior $\hat{Y} \in \mathbb{R}^{C,W,H}$ is generated, where C denotes the number of classes, similar to the vocabulary in open vocabulary semantic segmentation (OVS). Each pixel is assigned to the class in $\mathcal{V}_{test}$ with the highest posterior probability. Existing OVS methods assume that the test images come from a specific dataset (e.g. ADE20K (Zhou et al., 2019) or PASCAL-Context (Mottaghi et al., 2014)), and they use the corresponding dataset-specific vocabulary to restrict all predictions to the dataset's vocabulary.

The open nature of OVS. In an Open-World scenario, the open nature refers to a visual semantic object that may belong to multiple labels (can be described using different vocabularies or captions) which is overlooked. The "open nature" of vocabulary segmentation refers to the idea that the classification of objects or concepts is not rigid or strictly predefined, allowing for more flexibility in how categories are assigned and understood. It suggests that 1) Multiple labels: An object can be associated with multiple category labels simultaneously, rather than being constrained to just one label. This reflects the complexity and richness of real-world concepts. 2) Semantic similarity: The predicted category by a model may not match the ground truth label exactly, but if the predicted result is semantically similar to the true label, it shouldn't be considered entirely incorrect. For instance, if a model predicts "vehicle" instead of "car", this prediction may still be valid, as both are semantically related, and the broader category captures the essence of the object.

4 Mask-wise evaluation protocol

4.1 Notations and Definitions

For the sake of understanding, we define the main symbols used in the proposed evaluation framework and their meanings. For image $\mathcal{X}_i$ from a testing dataset $\mathcal{D}$, we have the class probability distribution predicted by the model, referred to as logits, denoted by $\mathbf{L}_i \in \mathbb{R}^{C\times W\times H}$, where C represents the total number of classes, and W and H denote the width and height of the image, respectively. Pixel-wise semantic segmentation annotations are represented as $\mathbf{Y}_i \in \{0,1,\ldots,C-1\}^{W\times H}$, where each value corresponds to the class index of the respective pixel. The one-hot encoded form of these annotations is given by $\mathbf{M}_i \in \{0,1\}^{C\times W\times H}$, where each pixel is either 0 or 1, $\mathbf{M}_i^{true} \in \{0,1\}^{C_{true}\times W\times H}$ is the mask list with all annotated category mask only, $C_{true} < C$. Binary predicted masks, $\mathbf{B}(\tau) \in \{0,1\}^{C\times W\times H}$, are obtained by thresholding $\mathbf{L}_i$ with a threshold τ, such that:

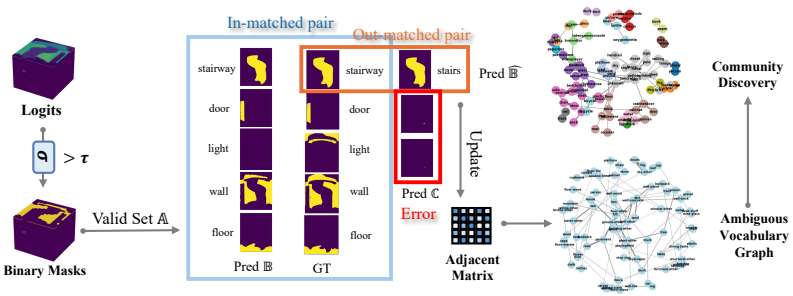

Figure 2: The proposed mask-wise evaluation protocol. The Valid Set $\mathbb{A}$ consists of all masks where $\mathbb{A} = \{M_i \mid M_i \in \{M_1, M_2, \ldots, M_K\}\}$. $\mathbb{B}$ represents the list of masks where the predicted category matches the category annotated in the ground truth (GT). $\hat{\mathbb{B}}$ is the set of masks obtained by performing bipartite matching between $(\mathbb{A} \setminus \mathbb{B})$ and the GT, where the IoU of the matched pairs exceeds the threshold τ_{AV}. For example, *'stairs'* belongs to $\hat{\mathbb{B}}$ in this figure. $\mathbb{C}$ is defined as $\mathbb{C} = \mathbb{A} \setminus (\hat{\mathbb{B}} \cup \mathbb{B})$.

$$\mathbf{B}(\tau)_i = \begin{cases} 1, & \text{if } \mathbf{L}_i \geq \tau, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

The $\mathbf{CM} \in \mathbb{N}^{C\times 2\times 2}$ is a binary confusion matrix for dataset $\mathcal{D}$, where each class c has:

$$\mathbf{CM}_c = \begin{bmatrix} \mathrm{TP}_c & \mathrm{FP}_c \\ \mathrm{FN}_c & \mathrm{TN}_c \end{bmatrix}. \tag{6}$$

$\mathbf{EM} \in \mathbb{R}^{C\times 1}$ is a vector used to quantify the error proportion for each class in the model's predictions. The error proportion is calculated as the ratio of the number of pixels with a value of 1 in the corresponding binary mask to the total number of pixels in the image i as follow:

$$err_{i,c} = \frac{\text{Number of pixels with a value of 1 in the binary mask}}{\text{Total number of pixels}} \tag{7}$$

4.2 Evaluation Protocol

Our mask-wise evaluation protocol provides CM, AV, and EM results under different thresholds as shown in Algorithm 1. Here, CM represents the binary confusion matrix between the predicted categories and the annotated categories. AV captures predictions with high overlap ($> \tau_{\text{AV}}$) between the predicted and annotated masks but with mismatched categories. Only CM and EM are used to evaluate the model's performance, as they are based on comparisons with the ground-truth annotations. AV, however, is utilized for subsequent ambiguous vocabulary graph analysis (i.e., analysing ambiguous vocabulary in the model's predictions) rather than directly assessing the model's performance.

Based on our proposed evaluation protocol, we define three metrics $front$, $back$ and err to evaluate OVS model performance cross different thresholds, defined as:

$$front_\tau = \frac{1}{|C|} \sum_{c\in C} \frac{\mathbf{CM}_\tau[TP_c]}{\mathbf{CM}_\tau[TP_c] + \mathbf{CM}_\tau[FP_c] + \mathbf{CM}_\tau[FN_c]} \tag{8}$$

$$back_\tau = \frac{1}{|C|} \sum_{c\in C} \frac{\mathbf{CM}_\tau[TN_c]}{\mathbf{CM}_\tau[TN_c] + \mathbf{CM}_\tau[FP_c] + \mathbf{CM}_\tau[FN_c]} \tag{9}$$

$$err_\tau = \frac{1}{|C|} \sum_{c\in C} \mathbf{EM}_{\tau,c} \tag{10}$$

Here, $front_{\tau,c}$ and $back_{\tau,c}$ represent the recognition IoU of the foreground and background classes, respectively, for each category c, where $c \in C$ under threshold τ. The foreground class refers to the pixels labeled as relevant to the target in category c (i.e., the regions belonging to that category), the background class refers to the pixels labeled as irrelevant to the target in category c (i.e., regions outside that category). The err_τ represents the average proportion of incorrectly predicted pixels across all categories under the threshold τ.

Inspired by best F1 thresholding (Lipton et al., 2014), the best threshold $\tau^\star$ can be automatically determined by:

$$\tau^\star = \underset{\tau \in \{0.1, 0.2, \ldots, 0.9\}}{\arg\max} \left(\sqrt{\text{front}_i^2 + (1 - \text{err}_i)^2} \right) \tag{11}$$

Algorithm 1 Mask-Wise Evaluation Protocol

Require: Annotated masks list $\{\mathbf{M}_i^{true}\}$, predicted masks list $\{\mathbf{B}(\tau)_i\}$, threshold range $\{0.1, 0.2, \ldots, 0.9\}$, and threshold τ_{AV} for ambiguous vocabulary matrix.
Ensure: Results for each threshold τ: $\{ \mathbf{CM}_\tau, \mathbf{EM}_\tau, \mathbf{AV}_\tau \}$

Initialize results dictionary **Results** $\leftarrow \emptyset$
for $\tau \in \{0.1, 0.2, \ldots, 0.8, 0.9\}$ **do** ▷ Iterate over all threshold values
  Initialize confusion matrix $\mathbf{CM}_\tau \leftarrow 0$
  Initialize ambiguous vocabulary matrix $\mathbf{AV}_\tau \leftarrow 0$
  Initialize error matrix $\mathbf{EM}_\tau \leftarrow 0$
  for $i \in \mathcal{D}$ **do** ▷ Iterate over all images in the dataset
    Obtain binary predicted masks $\mathbf{B}(\tau)_i$
    for $c \in C^{\text{true}}$ **do** ▷ Iterate over true categories
      Compute binary confusion matrix for $\mathbf{M}_{i,c}^{true}$ and $\mathbf{B}(\tau)_{i,c}$
      Update confusion matrix $\mathbf{CM}_{\tau,c}$ with computed values
    end for
    Remove masks not in C^{true} and zero masks from predictions:

$$\mathbf{B}(\tau)_i^{\text{rem}} \leftarrow \{\mathbf{B}(\tau)_{i,c} \mid c \notin C^{\text{true}}, \mathbf{B}(\tau)_{i,c} \neq 0\}$$

    Solve bipartite graph matching between $\mathbf{B}(\tau)_i^{\text{rem}}$ and $\mathbf{M}_i^{\text{true}}$:

$$\text{Matches} \leftarrow \{(b, m) \mid b \in \mathbf{B}(\tau)_i^{\text{rem}}, m \in \mathbf{M}_i^{\text{true}}, \text{IoU}(b, m) > \tau_{AV}\}$$

    for each matched pair $(b, m) \in$ Matches **do**
      Update $\mathbf{AV}_{\tau,m,b} \leftarrow \mathbf{AV}_{\tau,m,b} + 1$
      Remove matched mask b from $\mathbf{B}(\tau)_i^{\text{rem}}$
    end for
    Let $\hat{\mathbf{B}(\tau)}_i^{\text{rem}}$ be the remaining masks after removal
    for $c \in$ Categories in $\hat{\mathbf{B}(\tau)}_i^{\text{rem}}$ **do** ▷ Iterate over remaining categories
      Compute error $err_{i,c}$ for image i
      Update error matrix $\mathbf{EM}_{\tau,c} \leftarrow \mathbf{EM}_{\tau,c} + err_{i,c}$
    end for
  end for
  Store results for current τ:

$$\mathbf{Results}[\tau] \leftarrow \{\mathbf{CM}_\tau, \mathbf{AV}_\tau, \mathbf{EM}_\tau\}$$

end for
return Results

4.3 Ambiguous Graph in Out-Matched Pair

Building an ambiguous vocabulary graph. A graph is a mathematical structure used to model pairwise relationships between objects. It is commonly described using an adjacency matrix, where each entry in the matrix represents the connection or interaction between two nodes (in this case, vocabularies). The confusion graph is constructed based on the model's predictions and the manually annotated classes. The graph is used for analysing the model's performance in classification task (Jin et al., 2017).

The adjacency matrix $\mathbf{AV}$ for ambiguous vocabulary graph is the ambiguous vocabulary matrix in Algorithm 1. Each element $\mathbf{AV}_{i,j}$ represents the number of times the model predicts class j, while the ground truth is class i. For each out-matched pair, the adjacency matrix is updated. For example, if there is an out-matched pair such as *"couch"-"sofa"* where the ground truth class is *"sofa"* and the predicted class is *"couch"*, we update the corresponding entry in the adjacency matrix, $\mathbf{AV}_{\text{sofa,couch}}$, by incrementing it by 1. This reflects the number of times the model confused "sofa" with "couch" with high overlap in IoU.

Community discover over ambiguous graph. Given the confusion graph represented by the corresponding adjacency matrix $\mathbf{A} \in \mathbb{R}^{C \times C}$, we can perform community discovery to identify groups of classes that are frequently confused with each other. This process involves partitioning the nodes (classes) into communities such that nodes within the same community have stronger connections, as reflected by higher values in the adjacency matrix, than those across different communities.

One common approach for community discovery is modularity maximisation. The modularity Q of a given partition of the graph is defined as:

$$Q = \frac{1}{2m} \sum_{i,j} \left[A_{i,j} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j) \tag{12}$$

where $A_{i,j}$ is the weight of the edge between nodes i and j in the adjacency matrix, k_i and k_j are the degrees (total edge weights) of nodes i and j, respectively, m is the total weight of all edges in the graph, i.e. $m = \frac{1}{2} \sum_{i,j} A_{i,j}$. $\delta(c_i, c_j)$ is the Kronecker delta function, which is 1 if nodes i and j belong to the same community and 0 otherwise. The goal of community discovery is to maximise Q in order to find the optimal partition of nodes into communities. In the ambiguous vocabulary graph, if two categories are often confused by the model, they are likely to be in the same community.

5 Experiment

5.1 Datasets and Implementation Details

Following previous OVS works (Cho et al., 2023; Xie et al., 2023; Xu et al., 2023), we train the models on the COCO-Stuff171 (Caesar et al., 2018) dataset with 171 categories and perform zero-shot evaluation on ADE20K (Zhou et al., 2019) and PASCAL-Context (Mottaghi et al., 2014) datasets. ADE20K has two types of annotations namely ADE150 with 150 classes, and ADE847 with 847 classes. PASCAL-Context has the most frequent 59 classes annotation version PC59, and fully annotated version PC459 with 459 categories.

In this work, we utilise the following benchmark models for evaluation: EBSeg (Shan et al., 2024), CAT-Seg (Cho et al., 2023), SED (Xie et al., 2023), and MAFT+ (Jiao et al., 2024). In addition to the default setting, which uses dataset-specific vocabulary during inference for each dataset, we also created a joint-dataset inference vocabulary set, denoted by $\star$. This joint-dataset vocabulary set is disjoint from PC59, ADE150, PC459, and ADE847, resulting in a total of 1,086 vocabularies. It is important to note that no vocabularies were merged, for instance, terms like *'airplane'* and *'aeroplane'* were treated as distinct entries as they are predefined. The threshold $\hat{\tau}$ is set to 0.7.

We follow exactly the same configuration for experiments with the benchmark models. The experiments were conducted on one NVIDIA A100 GPU.

5.2 Re-benchmarking

Here we compare the commonly used pixel-wise mIoU metric that uses the argmax operation and our proposed mask-wise metric incorporating soft set prediction. The results are shown in Table 1. Our observations are as follows: 1) Simply replacing the pixel-wise argmax-based mIoU with the proposed mask-wise metric, *front* (target), leads to a performance improvement in existing OVS models. The OVS model achieves high accuracy (above 90%) for *back* (non-target) across all datasets, except for the PC459 dataset. 2) As the inference vocabulary increases (using the joint-dataset vocabulary set during testing), our proposed evaluation method maintains relatively stable

Table 1: Quantitative results of our proposed mask-wise evaluation protocol. The symbol ⋆ indicates using a joint-dataset vocabulary set during testing. NULL denotes non-out-matched mask is predicted. The quantitative results of the previous argmax pixel-wise are listed in the first four rows.

Method	Venue	PC59			ADE150			PC459			ADE847		
EBSeg	CVPR'24		60.20			32.80			21.00			13.70	
CAT-Seg	CVPR'24		63.30			37.90			23.80			16.00	
MAFT+	ECCV'24		59.40			36.10			21.60			15.10	
SED	CVPR'24		60.90			34.30			22.10			9.70	
		front↑	back↑	err↓	front↑	back↑	err↓	front↑	back↑	err↓	front↑	back↑	err↓
EBSeg	CVPR'24	65.91	93.75	9.99	43.53	93.12	7.32	29.30	70.87	4.54	22.18	92.46	4.33
CAT-Seg	CVPR'24	68.46	94.24	Null	51.46	94.61	Null	12.77	68.96	Null	14.77	93.66	Null
MAFT+	ECCV'24	64.95	93.57	9.10	45.61	93.10	6.71	30.41	70.82	5.60	28.74	92.15	7.43
SED	CVPR'24	66.29	94.21	6.43	44.90	93.50	4.73	32.54	70.72	4.10	29.15	92.61	4.64
EBSeg⋆	CVPR'24	64.32	91.83	10.99	42.18	91.50	8.32	27.85	69.06	6.20	21.01	91.04	5.10
CAT-Seg⋆	CVPR'24	66.35	92.24	2.19	50.04	92.68	2.30	11.56	67.32	2.00	12.83	91.20	2.20
MAFT+⋆	ECCV'24	62.05	91.55	8.56	44.30	91.32	6.70	29.04	69.01	4.40	26.01	90.50	6.40
SED⋆	CVPR'24	63.35	91.32	5.31	42.65	91.28	4.52	30.04	68.40	3.23	27.45	90.05	4.10

Table 2: The quantitative results of argmax (i.e. pixel-wise) evaluation using a joint-dataset vocabulary set during testing.

Method	PC59	ADE150	PC459	ADE847
EBSeg	40.15	22.50	10.50	3.20
CAT-Seg	42.90	25.60	12.30	7.00
SED	43.70	24.10	11.00	5.20
MAFT+	41.30	23.80	10.80	6.50

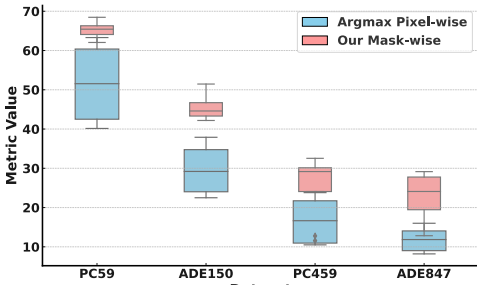

Figure 3: The stability of evaluation methods in comparison when evaluated with more vocabulary (*i.e. in a more open setting*).

performance for both *front* and *back*, whereas the performance of the previous argmax-based pixel-wise evaluation method drops significantly, as shown in Table 2. The performance gap here is caused by the ambiguous prediction while they have a high overlap in IoU with ground truth masks. Furthermore, with the increase in the number of inference words, the model's error rate also increases.

6 Discussion

6.1 The effectiveness of the proposed mask-wise evaluation protocol

To demonstrate the effectiveness of the proposed mask-wise evaluation protocol, we provide several key arguments that support its feasibility and practical utility, particularly in the open-world setting.

The effectiveness. Previous methods cannot quantitatively measure model performance in open-world settings because they usually rely on category-level matching and evaluate only when the predictions belong to a predefined set of categories. This approach often draws incorrect conclusions when faced with new categories or synonyms (such as *"sofa"* vs. *"long couch"*).

Our proposed evaluation method effectively eliminates the impact of ambiguous category labels via mask matching, focusing on the mask overlap between the predictions and manual annotations. Even in the case of joint datasets, where different datasets use different vocabularies to describe the same or similar categories, our method is still able to evaluate through accurate mask overlap (e.g. IoU), avoiding evaluation instability caused by differences in category terms. This mask-based evaluation method allows our scheme to seamlessly adapt to the fusion of multiple datasets without unifying or standardising the category labels of each dataset. In open-world settings, despite the limited number of manually annotated categories, the OVS model is still able to predict as many new concepts as possible, further demonstrating the applicability of the evaluation method.

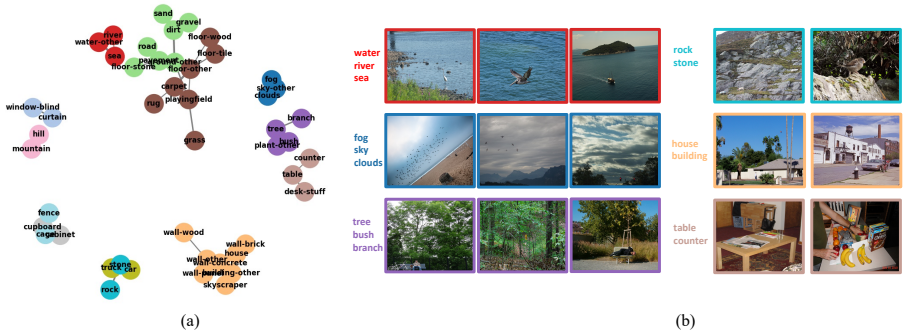

Figure 4: (a). The community extracted from the COCO-Stuff171 dataset (only 50 classes shown). (b). Example images from the same community, where the vocabularies share visually similar semantics (best viewed in color).

The stability. Our method is evaluated based on mask-wise matching, which means that its main evaluation criterion is the spatial overlap (such as IoU) between the predicted mask and the true annotated mask, rather than relying solely on the matching of class labels. When the vocabulary increases, traditional evaluation methods may experience large fluctuations due to the complexity of class label matching, especially when there are synonyms or inter-class ambiguities between different vocabulary sets. However, since our method only relies on matching the geometry and position of the mask, when the vocabulary set expands, the evaluation results will not be significantly affected by the expansion or change of class labels.

Assume that we have a universal vocabulary set $\mathcal{V}_{open}$, given a fixed testing dataset with k ground truth categories and corresponding $\mathcal{V}_k$ vocabularies, a suitable vocabulary set $\mathcal{V}_{test}$ for OVS testing on this dataset needs to satisfy: $\mathcal{V}_k \subseteq \mathcal{V}_{test}$. If we set the scale of testing vocabulary set to N, there exists C possible subset vocabulary $\mathcal{V}_{sub}$ to satisfy as a good testing vocabulary set for the given dataset, where $C = \frac{\mathcal{V}_{test}!}{(N-k)!(\mathcal{V}_{test}-(N-k))!}$. We then can use the Monte-Carlo method to approximate the upper bound and lower bound of test accuracy (Chen et al., 2019b) given various point estimations $(ACC_1, ..., ACC_C|\mathcal{V}_{open})$, for both the argmax with pixel-wise mIoU and our mask-wise evaluation as presented in Fig. 3. Specifically, the dataset-specific vocabulary set and joint vocabulary set are just two of many point estimations. We can observe that the proposed mask-wise evaluation protocol maintains a stable measurement across different datasets compared to the commonly used argmax-based and pixel-wise mIoU.

6.2 Analysis of ambiguous vocabulary graph

Many ambiguous annotations in existing datasets. Using community discovery methods, we can partition the ambiguous graph into communities as shown in Fig. 4 left. Each community represents a cluster of classes that are often confused with each other. For example, in an object detection or segmentation dataset, we might observe that the categories *"sofa"*, *"couch"*, and *"armchair"* form a tightly connected community, indicating that these classes are frequently misclassified or confused by the model.

This insight suggests that the dataset may contain ambiguous annotations where these objects are not clearly distinguishable or where multiple terms are used interchangeably in different regions or contexts. By visualising the labels of the same community, as shown in Fig. 4 right, we find that these labels are extremely similar visually and difficult to distinguish through subtle visual differences. From a human perspective, these labels are likely to be classified as the same thing, showing extremely high similarity, which may indicate that they share some core features or attributes.

Additionally, the community discovery process helps reveal systematic biases in the dataset. For example, if certain categories (such as "sedan" "SUV" and "truck") are often clustered in the same community, this may indicate that the dataset lacks diversity in the representation of different vehicle types, or that the model is inadequate in distinguishing between these categories. Details of the community discovery results for the datasets used in this paper can be found in Appendix A.2.

Table 4: Quantitative results of reducing the ambiguous vocabularies by non-target vocabulary removal with our proposed evaluation protocol. The *MAFT+* represents the two-stage OVS method while *SED* represents the one-stage OVS method. The one-stage method of OVS is performed through category aggregation, that is, by aggregating visual features and text features to train the model, thereby completing the detection and classification tasks at the same time. The two-stage method first generates class-agnostic masks and then classifies these masks.

Method	PC59			ADE150			PC459			ADE847		
	front↑	back↑	error↓	front↑	back↑	error↓	front↑	back↑	error↓	front↑	back↑	error↓
MAFT+	-0.87	-0.23	-1.98	-1.56	-0.63	-2.30	-1.12	-0.08	-1.24	-1.81	-1.33	-1.96
SED	+2.03	+0.61	-2.15	+5.28	+0.46	-0.21	+0.50	-0.04	-1.57	+1.51	-0.62	-2.00
SED w. 0.7	+1.10	+0.40	-1.80	+3.40	+0.30	-0.70	+0.25	-0.02	-1.50	+1.00	-0.30	-1.80
SED w. 0.5	+1.40	+0.50	-1.70	+3.90	+0.40	-0.60	+0.35	-0.01	-1.40	+1.30	-0.40	-1.70
SED w. 0.3	+1.70	+0.55	-1.60	+4.20	+0.42	-0.50	+0.40	+0.01	-1.30	+1.40	-0.50	-1.60
SED w. 0.1	+2.00	+0.60	-1.50	+4.50	+0.45	-0.40	+0.45	+0.02	-1.20	+1.50	-0.55	-1.50

We also calculate the confusion score for datasets commonly used in open set vision (OVS) tasks, which is calculated by dividing the number of communities formed through community discovery from the ambiguous graph by the total number of categories in the dataset, as shown in Table. 3. We found that there are a large number of ambiguous categories in the datasets currently used, which further highlights the importance of our proposed evaluation method. By effectively handling these ambiguous categories, our evaluation method can more accurately reflect the performance of the model in complex data environments.

Table 3: Dataset statistics of ambiguous vocabularies in existing datasets. The Cls Num. denotes the number of categories. The rate indicates the proportion of the ambiguous categories to the overall category.

Dataset	Cls Num.	Rate (%)
COCO171	171	9.94
PC59	59	10.16
ADE150	150	8.66
PC459	459	3.92
ADE847	847	3.30

Experiments show that reducing ambiguous vocabulary helps the one-stage model focus on meaningful semantic distinctions and avoid distractions from subtle differences between similar categories. This non-target vocabulary dropout strategy brings the language likelihood closer to real-world distributions (Eq. 3), improving training performance. In contrast, the two-stage model's reliance on mask classification means that reducing vocabulary weakens its classification ability, hindering generalisation.

6.3 Summary of the Observations

Based on the discussion above, we summarise the key findings as follows: 1) Through the analysis of our ambiguous vocabulary graph, we identified the presence of numerous ambiguous or synonymous vocabularies in commonly used OVS datasets. 2) After conducting community discovery analysis on the ambiguous vocabulary graph established from the model's predictions, we found that the categories the model tends to confuse often belong to the same community, and their corresponding images are visually similar. 3) We further proposed to remove such ambiguous vocabularies during the training stage, by simply randomly discarding non-target vocabularies, and found it led to performance improvements for OVS model.

7 Conclusion

In conclusion, we proposed a mask-wise evaluation protocol for Open-Vocabulary Segmentation (OVS) to address the issue of ambiguous vocabulary in evaluation. These ambiguities often arise under open-world conditions, where multiple interpretations of labels can be valid. Our experiments validated the effectiveness of the proposed evaluation approach. Moreover, using our evaluation protocol, we can construct an ambiguous vocabulary graph for OVS models, revealing a significant presence of confusing annotations in current OVS datasets. The experiments further showed that reducing such ambiguities can enhance the generalisation capability of OVS models, leading to improved performance. In addition, a further discussion provided insights for follow-up research. We hope our study could encourage the community to think more about the openness of open-world problems and hopefully inspire new research questions.

REFERENCES

Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1209–1218, 2018.

Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, et al. Hybrid task cascade for instance segmentation. In *CVPR*, 2019a.

Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4): 834–848, 2017.

Wenhu Chen, Yu Su, Yilin Shen, Zhiyu Chen, Xifeng Yan, and William Wang. How large a vocabulary does text classification need? a variational approach to vocabulary selection. *arXiv preprint arXiv:1902.10339*, 2019b.

Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. *arXiv:2112.01527*, 2021.

Seokju Cho, Heeseong Shin, Sunghwan Hong, Seungjun An, Seungjun Lee, Anurag Arnab, Paul Hongsuck Seo, and Seungryong Kim. Cat-seg: Cost aggregation for open-vocabulary semantic segmentation. *arXiv preprint arXiv:2303.11797*, 2023.

Jian Ding, Nan Xue, Gui-Song Xia, and Dengxin Dai. Decoupling zero-shot semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11583–11592, 2022.

Xiaoyi Dong, Jianmin Bao, Yinglin Zheng, Ting Zhang, Dongdong Chen, Hao Yang, Ming Zeng, Weiming Zhang, Lu Yuan, Dong Chen, et al. Maskclip: Masked self-distillation advances contrastive language-image pretraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10995–11005, 2023.

Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *European Conference on Computer Vision*, pp. 540–557. Springer, 2022.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017.

Siyu Jiao, Hongguang Zhu, Jiannan Huang, Yao Zhao, Yunchao Wei, and Humphrey Shi. Collaborative vision-text representation optimizing for open-vocabulary segmentation. *arXiv preprint arXiv:2408.00744*, 2024.

Ruochun Jin, Yong Dou, Yueqing Wang, and Xin Niu. Confusion graph: Detecting confusion communities in large scale image classification. In *IJCAI*, pp. 1980–1986, 2017.

Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *CVPR*, 2019a.

Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *CVPR*, 2019b.

Boyi Li, Kilian Q. Weinberger, Serge Belongie, Vladlen Koltun, and René Ranftl. Language-driven semantic segmentation, 2022.

Xiangtai Li, Henghui Ding, Wenwei Zhang, Haobo Yuan, Guangliang Cheng, Pang Jiangmiao, Kai Chen, Ziwei Liu, and Chen Change Loy. Transformer-based visual segmentation: A survey. *arXiv pre-print*, 2023.

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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7061–7070, 2023.

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 *ECCV*, 2014.

Zachary C Lipton, Charles Elkan, and Balakrishnan Naryanaswamy. Optimal thresholding of classifiers to maximize f1 measure. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2014, Nancy, France, September 15-19, 2014. Proceedings, Part II 14*, pp. 225–239. Springer, 2014.

Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *CVPR*, 2018.

Yong Liu, Sule Bai, Guanbin Li, Yitong Wang, and Yansong Tang. Open-vocabulary segmentation with semantic-assisted calibration. *arXiv preprint arXiv:2312.04089*, 2023.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 891–898, 2014.

Lu Qi, Yi Wang, Yukang Chen, Ying-Cong Chen, Xiangyu Zhang, Jian Sun, and Jiaya Jia. Pointins: Point-based instance segmentation. *TPAMI*, 2021.

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 *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

Xiangheng Shan, Dongyue Wu, Guilin Zhu, Yuanjie Shao, Nong Sang, and Changxin Gao. Open-vocabulary semantic segmentation with image embedding balancing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 28412–28421, 2024.

Tiancheng Shen, Yuechen Zhang, Lu Qi, Jason Kuen, Xingyu Xie, Jianlong Wu, Zhe Lin, and Jiaya Jia. High quality segmentation for ultra high-resolution images. In *CVPR*, 2022.

Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. In *TPAMI*. IEEE, 2020.

Bin Xie, Jiale Cao, Jin Xie, Fahad Shahbaz Khan, and Yanwei Pang. Sed: A simple encoder-decoder for open-vocabulary semantic segmentation. *arXiv preprint arXiv:2311.15537*, 2023.

Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *NeurIPS*, 2021.

Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2955–2966, 2023.

Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, and Xiang Bai. A simple baseline for open-vocabulary semantic segmentation with pre-trained vision-language model. In *European Conference on Computer Vision*, pp. 736–753. Springer, 2022.

Wenwei Zhang, Jiangmiao Pang, Kai Chen, and Chen Change Loy. K-net: Towards unified image segmentation. *NeurIPS*, 2021.

Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, 2017.

Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127:302–321, 2019.

Hao Zhou, Tiancheng Shen, Xu Yang, Hai Huang, Xiangtai Li, Lu Qi, and Ming-Hsuan Yang. Rethinking evaluation metrics of open-vocabulary segmentaion. *arXiv preprint arXiv:2311.03352*, 2023.

A Appendix

A.1 The evaluation protocol

In addition to determining the best threshold for model performance, we also introduce an roc-auc-like metric, analogous to the area under the ROC curve, to comprehensively evaluate performance across all thresholds, as illustrated in the Figre 5.

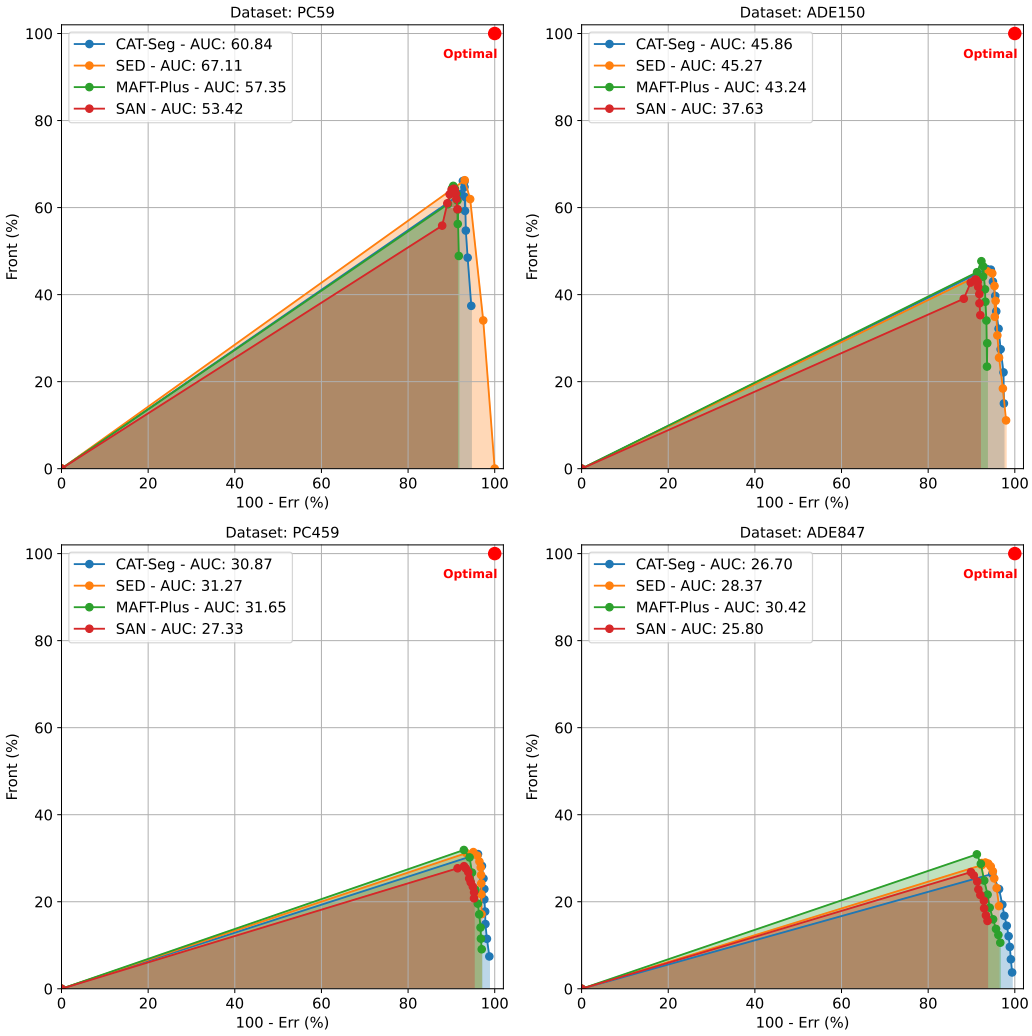

Figure 5: AUC evaluation metric based on $front$ and err.

A.2 THE COMMUNITY DISCOVERY RESULTS ACROSS THE DATASETS.

COCO-Stuff171

```
({'clouds', 'fog', 'sky-other'},
 {'building-other',
  'curtain',
  'house',
  'skyscraper',
  'wall-brick',
  'wall-concrete',
  'wall-other',
  'wall-panel',
  'wall-stone',
  'wall-tile',
  'wall-wood',
  'window-blind',
  'window-other',
  'wood'},
 {'carpet',
  'dirt',
  'floor-marble',
  'floor-other',
  'floor-stone',
  'floor-tile',
  'floor-wood',
  'gravel',
  'ground-other',
  'pavement',
  'platform',
  'playingfield',
  'road',
  'rug',
  'sand'},
 {'river', 'sea', 'water-other'},
 {'branch',
  'bush',
  'flower',
  'grass',
  'hill',
  'mountain',
  'plant-other',
  'potted plant',
  'straw',
  'tree'},
 {'cabinet', 'cupboard', 'shelf'},
 {'bus', 'car', 'train', 'truck'},
 {'rock', 'stone'},
 {'cage', 'fence', 'railing', 'structural-other'},
 {'counter', 'desk-stuff', 'dining table', 'table'},
 {'cloth', 'textile-other'},
 {'clothes', 'person'},
 {'furniture-other', 'metal', 'plastic', 'stop sign'},
 {'backpack', 'handbag'},
 {'cup', 'wine glass'},
 {'ceiling-other', 'roof'},
 {'hot dog', 'sandwich'})
```

PC59

```
({'bench',
  'building',
  'cabinet',
  'ceiling',
  'chair',
```

```
  'curtain',
  'diningtable',
  'door',
  'fence',
  'floor',
  'flower',
  'grass',
  'ground',
  'mountain',
  'platform',
  'pottedplant',
  'road',
  'rock',
  'shelves',
  'sidewalk',
  'track',
  'tree',
  'wall',
  'window',
  'wood'},
 {'bus', 'car', 'truck'},
 {'bag', 'bed', 'bedclothes', 'cloth', 'dog', 'sofa'},
 {'boat', 'water'},
 {'computer', 'tvmonitor'},
 {'bicycle', 'motorbike'})
```

ADE150

```
({'awning',
  'blind',
  'booth',
  'building',
  'ceiling',
  'chandelier',
  'curtain',
  'door',
  'fence',
  'hill',
  'house',
  'lamp',
  'light',
  'mountain',
  'railing',
  'rock',
  'sconce',
  'screen door',
  'skyscraper',
  'tower',
  'wall',
  'windowpane'},
 {'canopy',
  'dirt track',
  'earth',
  'field',
  'floor',
  'flower',
  'grass',
  'land',
  'path',
  'plant',
  'road',
  'rug',
  'runway',
  'sand',
  'sidewalk',
```

```
  'tree'},
 {'cushion', 'pillow'},
 {'car', 'truck', 'van'},
 {'armchair', 'chair', 'seat', 'sofa', 'stool', 'swivel chair'},
 {'bookcase',
  'cabinet',
  'chest of drawers',
  'coffee table',
  'counter',
  'countertop',
  'desk',
  'kitchen island',
  'pool table',
  'shelf',
  'table'},
 {'oven', 'stove'},
 {'lake', 'river', 'sea', 'water'},
 {'crt screen', 'monitor', 'television receiver'},
 {'ashcan', 'pot', 'vase'},
 {'stairs', 'stairway'},
 {'poster', 'signboard'})
```

PC459

```
({'brick',
  'bridge',
  'building',
  'cabinet',
  'cabinetdoor',
  'cage',
  'ceiling',
  'closet',
  'concrete',
  'counter',
  'door',
  'fence',
  'floor',
  'footbridge',
  'ground',
  'handrail',
  'mat',
  'metal',
  'patio',
  'platform',
  'pole',
  'road',
  'rug',
  'sand',
  'shed',
  'shelves',
  'sidewalk',
  'sign',
  'sky',
  'table',
  'tableware',
  'unknown',
  'wall'},
 {'car', 'toycar', 'truck'},
 {'light', 'lightbulb'},
 {'bag',
  'bedclothes',
  'chair',
  'cloth',
  'clothestree',
  'cushion',
```

```
  'flower',
  'grass',
  'leaves',
  'mountain',
  'pack',
  'pillow',
  'plant',
  'plastic',
  'pot',
  'pottedplant',
  'sofa',
  'stool',
  'straw',
  'towel',
  'tree',
  'wood'},
 {'curtain', 'window', 'windowblinds'},
 {'rock', 'stone'},
 {'dolphin', 'water', 'wharf'},
 {'box', 'paperbox'},
 {'bicycle', 'tricycle'},
 {'beer', 'bottle', 'oxygenbottle'},
 {'person', 'player'},
 {'coffee', 'cup', 'glass'},
 {'screen', 'tvmonitor', 'videogameconsole', 'videoplayer'},
 {'picture', 'poster'},
 {'bird', 'duck'},
 {'rail', 'track'},
 {'dog', 'fox'},
 {'book', 'paper'})
```

ADE847

```
({'baseboard',
  'central reservation',
  'curb',
  'floor',
  'footpath',
  'mat',
  'path',
  'road',
  'rug',
  'sidewalk',
  'skirting board'},
 {'balcony',
  'building',
  'building materials',
  'cabin',
  'first floor',
  'house',
  'pane',
  'porch',
  'shop',
  'shops',
  'skyscraper',
  'street number',
  'windowpane'},
 {'cover curtain', 'curtain'},
 {'flower',
  'forest',
  'plant',
  'plant pots',
  'pot',
  'tree',
  'trunk',
```

```
  'vase',
  'weeds'},
 {'bed', 'beds', 'eiderdown'},
 {'door', 'door bars', 'doorframe', 'double door'},
 {'buffet',
  'cabinet',
  'chest of drawers',
  'coffee table',
  'desk',
  'table',
  'table cloth',
  'tables',
  'television stand'},
 {'booth',
  'brick',
  'hill',
  'mountain',
  'mountain pass',
  'rock',
  'rocky formation',
  'shower room',
  'temple',
  'wall'},
 {'apparel', 'dummy', 'person', 'trouser'},
 {'car', 'truck', 'van'},
 {'cushion', 'pillow'},
 {'earth', 'field', 'grass', 'land', 'sand'},
 {'counter', 'countertop', 'work surface'},
 {'armchair', 'chair', 'rocking chair', 'seat', 'stool', 'swivel chair'},
 {'fireplace', 'fireplace utensils'},
 {'lake', 'river', 'sea', 'shore', 'water'},
 {'awning', 'blind'},
 {'sofa', 'sofa bed'},
 {'light', 'light bulb'}, {'lamp', 'sconce'},
 {'screen', 'television receiver'},
 {'ceiling', 'eaves', 'roof'},
 {'cooker', 'stove'},
 {'clock', 'watch'},
 {'games table', 'pool table'},
 {'ashcan', 'recycling bin'},
{'barrier', 'fence', 'railing'}, {'stairs', 'step'})
```