Improving Accuracy in Open-Vocabulary Object Detection

Rabia Şevval Aydın
Istanbul Technical University
aydinr25@itu.edu.tr

Abstract

The aim of open-vocabulary object detection is detecting objects beyond a fixed set of predefined categories by leveraging vision–language representations. Recent approaches achieve strong performance but often rely on heavy architectures. This work focuses on YOLO-World, a real-time open-vocabulary detector, and investigate accuracy-oriented modifications to its vocabulary construction and vision–language fusion mechanisms.

1. Introduction

Object detection is a fundamental task in computer vision. Modern object detectors have achieved remarkable performance by training on large-scale datasets with predefined category sets. However, these detectors are limited by their fixed vocabularies and they are unable to recognize objects outside the categories they have seen during training. These limitations restrict their application capabilities in open-world scenarios, where the objects that are needed to be detected may change dynamically or include previously unseen categories.

To address this issue, open-vocabulary object detection has emerged as a promising research topic. Using vision-language representations, open-vocabulary detectors aim to recognize objects using textual descriptions instead of fixed class labels. Recent approaches incorporate large vision language models to align image regions with text embeddings, enabling zero-shot detection. Although these works achieve strong accuracy scores, many of these methods rely on heavy transformer based architectures and require online text encoding during inference, resulting in high computational cost and limited performance for real-world deployment.

YOLO-World [1] takes an important step toward solving this issue by extending the YOLO detection framework to the open-vocabulary setting. By integrating vision language modeling into a lightweight, single-stage detector, YOLO-World achieves a balance between efficiency and generalization. A key feature of YOLO-World is its offline vocabulary mechanism, which allows textual prompts to be encoded once and re-parameterized into the detector, enabling fast inference without repetition in text encoding. In addition, YOLO-World rearranges object detection as a region-text matching problem, allowing the detector to generalize beyond a fixed category set.

Although YOLO-World shows strong performance, most of its design choices are primarily motivated by efficiency. In this project, prompt formats are analyzed to see which changes might improve detection accuracy. The investigation was about how vocabulary construction strategies and region-text interaction mechanisms influence open-vocabulary detection performance. Evaluation of the approach was done on COCO dataset under a zero-shot setting, which provides a challenging benchmark due to its large number of categories and long-tailed distribution.

2. Method

2.1. Baseline: YOLO-World

YOLO-World extends the YOLO object detection framework to the open-vocabulary setting by redesigning object detection as a region-text matching problem. Instead of predicting fixed class labels, the detector predicts object embeddings for each bounding box, which are then matched to text embeddings representing the target vocabulary.

Given an input image, the YOLO backbone extracts multi-scale visual features. In parallel, textual prompts describing object categories or phrases are encoded into text embeddings using a pretrained vision-language text encoder, in this case it's the CLIP. The detector outputs bounding boxes along with the corresponding object embeddings, and detection scores are computed on based on the similarity between object embeddings and text embeddings.

2.2. Online-Vocabulary

Vocabulary is the set of words or phrases that model is allowed to detect. DUring training process, YOLO-World uses online vocabulary. In online vocabulary during training each batch has different words that comes from category names, noun phrases and captions. For every batch

text encoder is runned again which allows the vocabulary to change dynamically which is good for open-vocabulary generalization and aligning regions with words. However for inference because it requires said steps to be done over and over again it causes computational issues. This is the problem with other open-vocabulary models. In YOLO-World offline vocabulary is used. At the beggining, textual prompts are encoded once and then YOLO-World absorbs the text embeddings into network weights. After this the text encoder is no longer needed and since embedding are no longer inputs the model becomes pure CNN again. This method is called prompt-then-detect.

2.3. Vocabulary Construction

YOLO-World aligns image regions and text descriptions in a shared embedding space. For each predicted bounding box, the detector produces an object embedding. Given a set of text embeddings representing the vocabulary, similarity scores are computed using cosine similarity. Detection is performed by selecting the text prompt with the highest similarity score for each region.

3. Experimental Setup

In this section, datasets, evaluation metrics, and implementation details used to assess the performance are mentioned. Experiments are designed to evaluate open-vocabulary detection accuracy under a zero-shot setting, with particular emphasis on large-vocabulary and long-tailed scenarios.

3.1. Datasets

Primary evaluation dataset is the COCO dataset. COCO contains 80 object categories. Results are reported on the COCO val2017 split using standard detection metrics including mean Average Precision (AP), AP at IoU thresholds of 0.50 (AP50) and 0.75 (AP75). These metrics provide a view of detection accuracy on a fixed-vocabulary benchmark.

3.2. Implementation Details

All experiments are conducted using YOLO-World-L as the baseline detector with 37.1 million parameters. The detector is evaluated under a zero-shot setting, where textual prompts corresponding to target categories are provided in offline vocabulary. First baseline performance is established using the original YOLO-World configuration. Then each modification is evaluated independently, reporting detection accuracy under identical conditions.

4. Results

In this section, the detection performance of baseline configuration and a prompt-based variant using descriptive class labels are reported. Both settings are evaluated using standard COCO-style metrics, including AP across IoU thresholds and scale-based AP for small/medium/large objects.

4.1. Baseline vs. Descriptive Class Labels

Table 1 summarizes the full evaluation metrics for the baseline model using raw class labels and for the variant using descriptive class labels.

To show the comparison results more clearly, Table 2 reports the absolute differences between the two settings (descriptive labels minus baseline).

4.2. Discussion

Across all reported metrics, using descriptive class labels reduces detection performance relative to the baseline raw-label setting. The drop is consistent across IoU thresholds (AP50 and AP75) and across object scales (small, medium, and large). This suggests that, under current configuration, the prompt template "a photo of a ..." does not yield better region-text alignment for detection, and may introduce a mismatch between how the model encodes category semantics and how the vocabulary was represented during training and evaluation.

These results motivate exploring alternative prompt strategies (e.g., class-specific templates, synonym expansion, or prompt ensembling) or training-time adaptations

Metric	Baseline (raw labels)	Descriptive labels
AP (0.50:0.95)	0.459	0.411
AP50	0.626	0.565
AP75	0.499	0.447
AP_{small}	0.357	0.315
AP_{medium}	0.510	0.467
AP_{large}	0.542	0.504
AR@100	0.702	0.680

Table 1. Evaluation results comparing the baseline model using raw class labels against the prompt-based setting using descriptive class labels.

Metric	Difference
AP (0.50:0.95)	−0.048
AP50	−0.061
AP75	−0.052
AP_{small}	−0.042
AP_{medium}	−0.043
AP_{large}	−0.038
AR@100	−0.022

Table 2. Absolute performance change when switching from raw class labels to descriptive class labels. Negative values indicate a drop in performance.

(e.g., prompt-aware vocabulary sampling) to improve semantic matching without degrading standard detection performance.

5. Conclusion

In this project, accuracy-oriented improvements for open-vocabulary object detection by building upon YOLO-World detector was tried. Rather than focusing on inference speed or architectural scaling, vocabulary construction's influence on detection accuracy in large-vocabulary settings was investigated.

Investigation was done on the use of more descriptive prompting strategies, such as class labels formulated as natural language phrases. Contrary to expectations, this modification did not lead to performance improvements and in fact resulted in a consistent drop in detection accuracy. Hypothesis is that this behavior is due to the fact that the detector was pretrained and optimized using raw category labels, and that introducing descriptive prompts only at inference time creates a mismatch between the training and inference distributions. This observation suggests that prompt-based formulations may be more effective if incorporated during training rather than applied solely at inference. Training the model from scratch or fine-tuning it with descriptive prompts could allow the detector to better adapt its region-text alignment, potentially leading to improved open-vocabulary detection performance.

References

[1] Tianheng Cheng, Lin Song, Yixiao Ge, et al. Yolo-world: Real-time open-vocabulary object detection. *arXiv preprint arXiv:2401.17270*, 2024. 1