OmniLabel: A Challenging Benchmark for Language-Based Object Detection

Samuel Schulter[1, †] Vijay Kumar B G[1] Yumin Suh[1] Konstantinos M. Dafnis[2,*]
Zhixing Zhang[2,*] Shiyu Zhao[2,*] Dimitris Metaxas[2]

[†] project lead [*] equal technical contribution, alphabetic order

[1] NEC Laboratories America [2] Rutgers University

Abstract

Language-based object detection is a promising direction towards building a natural interface to describe objects in images that goes far beyond plain category names. While recent methods show great progress in that direction, proper evaluation is lacking. With OmniLabel, we propose a novel task definition, dataset, and evaluation metric. The task subsumes standard- and open-vocabulary detection as well as referring expressions. With more than 28K unique object descriptions on over 25K images, OmniLabel provides a challenging benchmark with diverse and complex object descriptions in a naturally open-vocabulary setting. Moreover, a key differentiation to existing benchmarks is that our object descriptions can refer to one, multiple or even no object, hence, providing negative examples in free-form text. The proposed evaluation handles the large label space and judges performance via a modified average precision metric, which we validate by evaluating strong language-based baselines. OmniLabel indeed provides a challenging test bed for future research on language-based detection. Visit the project website at `https://www.omnilabel.org`

1. Introduction

A nuanced understanding of the rich semantics of the world around us is a key ability in the visual perception system of humans. Identifying objects from a description like "person wearing blue-and-white striped T-shirt standing next to the traffic sign" feels easy, because humans understand the composition of object category names, attributes, actions, and spatial or semantic relations between objects. When automated, this same ability can improve and enable a plethora of applications in robotics, autonomous vehicles, navigation, retail, etc.

With the recent advances in vision & language models [17, 19, 37], along with extensions towards object localization [15, 18, 24, 27, 49], a comprehensive evaluation benchmark is needed. However, existing ones fall short in various aspects. While object detection datasets significantly increased the label space over time (from 20 in Pascal [12] to 1200 in LVIS [16]), a fixed label space is assumed. The zero-shot [4] and open-vocabulary detection [15, 46] settings drop the fixed-labelspace assumption, but corresponding benchmarks only evaluate simple category names, neglecting more complex descriptions. Referring expression datasets [33, 45] probe models with free-form text descriptions of objects. However, the corresponding dataset annotations and metrics do not allow for a comprehensive evaluation of models.

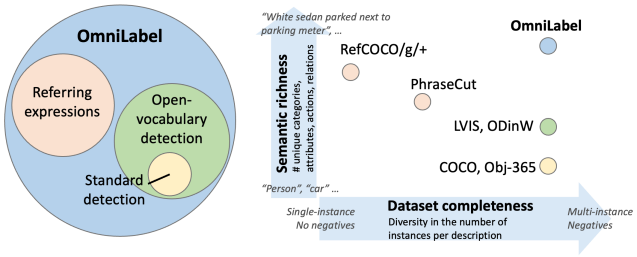

Figure 1: **(Left)** OmniLabel extends upon standard detection, open-vocabulary detection, as well as referring expressions, subsuming these tasks as special cases. **(Right)** The OmniLabel dataset is semantically rich with diverse free-form text descriptions of objects. Moreover, with object descriptions referring to multiple instances, dedicated negative descriptions, and a novel evaluation metric, our benchmark poses a challenging task for language-based detectors.

We introduce a novel benchmark called OmniLabel with the goal to comprehensively probe models for their ability to understand complex, free-form textual descriptions of objects and to locate the corresponding instances. This requires a novel task definition and evaluation metric, which we propose in Sec. 3. Our evaluation benchmark does not assume a fixed label space (unlike standard detection), uses

Dataset	# images	Free-form	Descr. length	# unique nouns	Open-vocabulary	Multi-Instance	Negative	Evaluation
LVIS [16]	5K	✗	–	1.2K	✗	✓	✓	AP
ODinW [25]	27.3k	✗	–	0.3K	✓	✓	✓	AP
RefCOCO [33, 45]	4.3K	✓	4.5	3.5K	✓	✗	✗	P
Flickr30k [36]	1.0K	✓	2.4	1.9K	✓	✗	✗	R
PhraseCut [42]	2.9K	✓	2.0	1.5K	✓	✓	✗	IoU
OmniLabel	12.2K	✓	5.6	4.6K	✓	✓	✓	AP

Table 1: Comparing OmniLabel to existing benchmarks: On 12.2K images, OmniLabel provides free-form text descriptions of objects with an average description length of 5.6 words, covering 4.6K unique nouns. Each description can refer to multiple objects, or no object, *i.e.*, a negative example, an important factor in our evaluation. (Numbers are computed on validation sets. P: Precision, R: Recall, AP: Average Precision, IoU: Intersection over Union)

complex object descriptions beyond plain category names (unlike open-vocabulary detection), and evaluates true detection ability with descriptions referring to zero, one or more instances in a given image (unlike referring expressions). A unique aspect of our benchmark are the descriptions that refer to zero instances, which pose a challenge to existing methods as hard negative examples. Fig. 1 positions our OmniLabel benchmark.

To build this evaluation benchmark, we collected a set of novel annotations upon existing object detection datasets. We augment the existing plain category names with novel free-form text descriptions of objects. Our specific annotation process (Sec. 4) increases the difficulty of the task by ensuring that at least one of the following conditions is true:

(a) Multiple instances of the same underlying object category are present in the same image
(b) One object description can refer to multiple objects
(c) An image contains a negative object description, which refers to no object but is related to the image's semantics
(d) Descriptions do not use the original category name

Tab. 1 highlights the key differences of OmniLabel to existing benchmarks: The diversity in the free-form text descriptions and the evaluation as an object detection task, including multiple instances per description as well as negative object descriptions. The numbers in the table reflect our *public validation set*, which is roughly the same size as our *private test set*. Fig. 2 provides examples of the dataset.

We also evaluate recent language-based detectors on

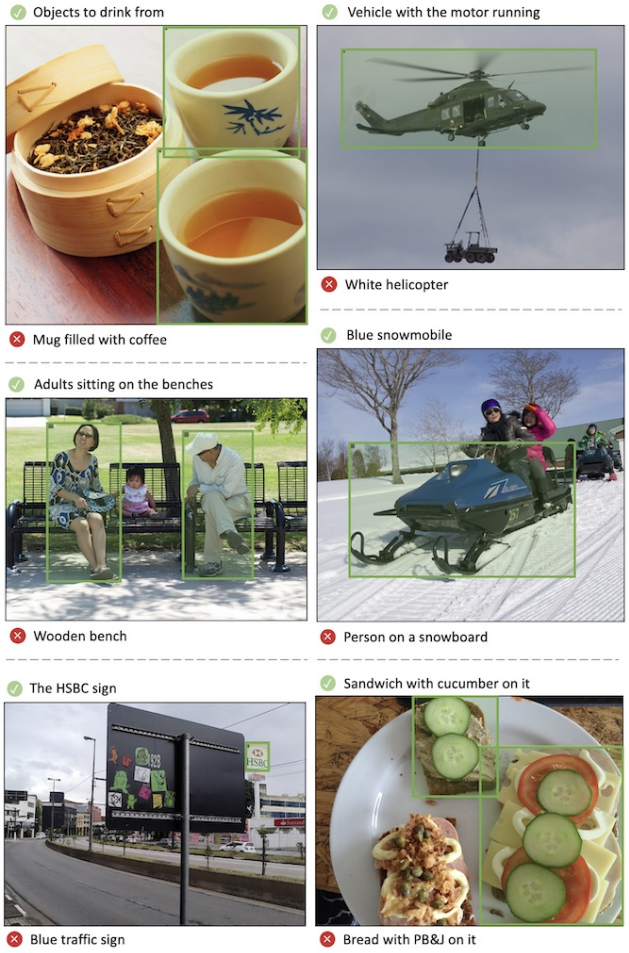

Figure 2: Examples of the OmniLabel ground truth annotations. Positive descriptions (above each image) can refer to one or multiple instances. Negative descriptions (below each image) are semantically related to the image but refer to no object.

our benchmark, including RegionCLIP [51], Detic [53], MDETR [18], GLIP [27] and FIBER [11]. Summarized in Sec. 6.2, our key observation is that the proposed benchmark is difficult for all methods, and the evaluation metric is more stringent than in prior benchmarks. Negative object descriptions pose the biggest challenge to current methods.

We summarize our contributions as follows:

(a) A novel benchmark to unify standard detection, open-vocabulary detection and referring expressions
(b) A data annotation process to collect diverse and complex free-form text descriptions of objects, including negative examples
(c) A comprehensive evaluation metric that handles the virtually infinite label space

2. Related Work

To position our proposed OmniLabel benchmark, we relate it to existing tasks and focus on the corresponding benchmark datasets.

Object Detection: Localizing and categorizing objects is a long-standing and important task with many applications. An enormous amount of datasets fueled this research. Besides datasets for specific use cases, like face [43], pedestrians [3, 10] or driving scenes [9, 13, 34, 44], the most popular ones are general-purpose: Pascal VOC [12], MS COCO [30], Objects-365 [40], OpenImages [21] or LVIS [16]. These datasets also reflect the evolution of size, both in number of images and, more relevant here, the number of categories. In the same order as above, the label space sizes are 20, 80, 365, 600 and 1203. These datasets lead to significant progress in the past years on neural network architectures [6, 29, 31, 38, 39, 47, 56, 57] as well as robustness and scaling [50, 53, 55]. Still, the limitations over OmniLabel are obvious: All detection datasets assume a fixed labelspace, do not provide an open-vocabulary setting or free-form object detections.

Referring Expressions: Instead of a limited and fixed set of category names, the motivation in referring expressions is to refer to objects with natural language. The most popular benchmark is the series of RefCOCO/g/+ [33, 45]. While RefCOCO/g [33] often contains long and redundant descriptions, RefCOCO/+ [45] limited the referring phrases with a specific annotation process involving a two-player game. The RefCOCO+ extension restricted annotators to use spatial references (*e.g.*, "man on left"), which was likely over-used because all of RefCOCO/g/+ assume each phrase to refer to exactly one instance. In contrast, OmniLabel explicitly asks annotators to pick two or more instances to describe in many images. PhraseCut [42] also collects templated expressions that refer to multiple instances and also provides segmentation masks. However, OmniLabel still has more instances per object description and uses free-form descriptions. Moreover, none of the existing referring expression datasets provides negative examples.

Visual Grounding: While the task of referring expressions is to localize the main subject of the phrase, visual grounding aims at localizing each object of the phrase, *i.e.*, grounding the text in the image. Benchmarks include Flickr30k [36] or Visual Genomes [20], which have often been used also for general object-centric pre-training for vision & language models like GLIP [27], MDETR [18], SIMLA [19], ALBEF [26]. OmniLabel addresses a different task that is more related to referring expressions. The annotation costs for grounding are also typically higher since all objects mentioned in a phrase need an associated bounding box, which often leads to noisy ground truth. In contrast, the annotation process for OmniLabel can easily be built upon existing detection datasets with high-quality bounding boxes.

Open-Vocabulary Object Detection: Aside from using natural language as object descriptions, scaling the label space of object detectors becomes infeasible with a standard supervised approach. This sparked work on the zero-shot setting [1, 2, 22, 23], where a set of base categories is available at training time, but novel (or unseen) categories are added at test time. While Bansal *et al*. [4] introduced the first work on zero-shot detection, later works relaxed the setting to open-vocabulary [46], where annotations other than bounding boxes can be leveraged that may include the novel categories, *e.g.*, image captions [7, 8, 41]. The recent success of large V&L models [17, 19, 37] surged interest in open-vocabulary detection [5, 14, 27, 15, 28, 48, 49]. However, benchmarks for this setting are lacking. Most existing work evaluates on standard detection datasets, COCO [30] and LVIS [16], by separating categories into base and novel. Most recently, [25] introduces the ODinW benchmark which combines 35 standard detection datasets to setup an open-vocabulary challenge. Still, all benchmarks use a rather limited set of simple category names. In contrast, OmniLabel provides higher complexity with object descriptions being free-form text and, with this, a larger number of unique words (and nouns) which poses a naturally open-vocabulary setting since every description is effectively unique.

3. Benchmark and Evaluation Metric

This section provides a formal definition of the benchmark task and the corresponding evaluation metric. An illustration of both is given in Fig. 3.

3.1. Benchmark Task

Input: Given a regular RGB image I_i along with a label space D_i, the task for model M is to output object predictions P_i according to the label space D_i. The subscript in D_i indicates that both content and size $|D_i|$ vary for each image I_i. The label space $D_i = \left[d_i^k\right]_{k=1}^{|D_i|}$ consists of $|D_i|$ elements, $d_i^k \in D_i$, each of which is called an "object description". Our object descriptions comprise a combination of plain category names (as in detection) as well as newly-collected free-form text descriptions of objects, see Sec. 4. Being free-form text effectively makes each description unique. While we could define a common label space as the union of all descriptions, this results in a huge label space and poses hard computational challenges on models that tightly fuse image and text, like MDETR [18]. Instead, we vary the label space and each D_i contains both positive (referring to an object in the image) and negative (related to image content but no related objects) descriptions. Examples of free-form object descriptions are given in Fig. 2.

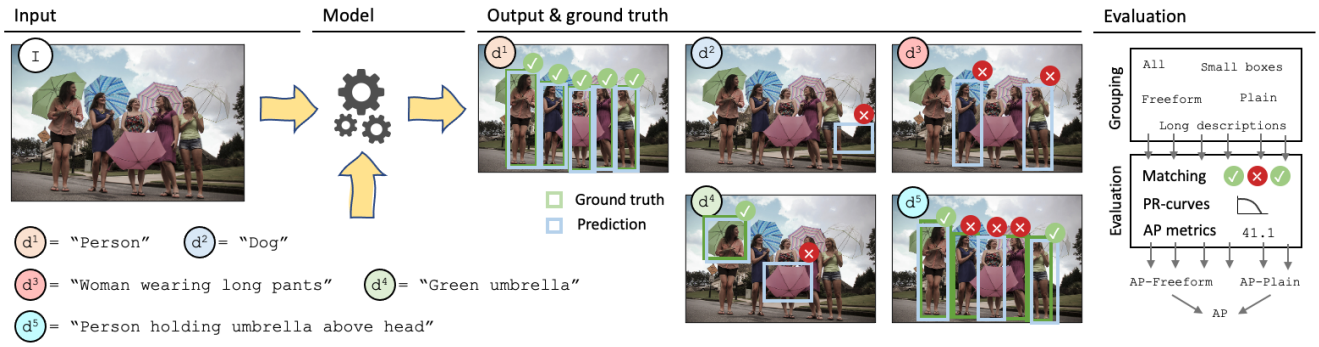

Figure 3: Illustration of the proposed task and evaluation. Given an image I and a set of object descriptions $\left[d^1, d^2, \ldots, d^5\right]$, the model outputs a set of predictions. Each prediction p^l is a triplet, consisting of a bounding box b^l (blue boxes), a confidence score s^l (not shown) and an index g^l, which links the prediction to one of the object descriptions d^* (the figure groups predicted boxes by g^l). For evaluation, predictions and ground truth (green boxes) are matched based on location (intersection-over-union) as well as the indices g^l. In contrast to standard detection, note that one object can be matched with multiple (but different!) descriptions. For example, the woman holding the green umbrella is matched with both d^1 and d^5. While there is no category-wise grouping as in standard detection, average precision (AP) metrics are computed for other groups, like "plain categories", "free-form object descriptions", and others. The final metric is the harmonic mean between AP values for "plain categories" and "free-form object descriptions".

Output: Model M must output a set of triplets $P_i = \left[p_i^l\right]_{l=1}^{|P_i|}$ for image I_i and label space D_i. Each triplet $p_i^l = (b_i^l, s_i^l, g_i^l)$ consists of a bounding box b, a confidence score s, and an index g linking the prediction to an object description in D_i. A bounding box b_i^l consists of 4 coordinates in the image space to define the extent of an object, as in standard object detection. The confidence of a model's prediction is expressed by the real-valued scalar s_i^l. Finally, the index g_i^l is in the range of $[1, |D_i|]$ and indicates that the prediction p_i^l localizes one object instance of the description g_i^l of the label space D_i. Note that multiple predictions p_i^l can point to the same object description d_i^k.

Difference to object detection benchmarks: The main difference is the label space, which is more complex (with natural text object descriptions, often unseen during training) as well as dynamic (size of label space changes for every test image). Standard object detectors fail this task because of their fixed label space assumption.

Difference to referring expression benchmarks: While the task definition is similar, the key difference is in the corresponding data. First, object descriptions D_i in our benchmark range from plain categories (like in standard detection) to highly specific descriptions. Second, each description can refer to zero, one, or multiple instances in the image. All referring expression datasets assume the presence of the object described by the text and, hence, do not contain negative examples that refer to zero instances, an important aspect of standard detection evaluation. Moreover, only one referring expression dataset ([42]) refers to more than a single instance per image.

3.2. Evaluation Metric

To evaluate a model M on our task, we propose a modified version of the object detection evaluation metric, average precision (AP) [30]. This modification is necessary to account for our novel object descriptions that make the label space virtually infinite in size, and that are different for each image. The following list summarizes the changes:

- While AP is computed for each category separately (and then averaged) in standard detection, this initial grouping is omitted in OmniLabel. Due to the high specificity of the object descriptions, many of these "groups" would then consist of only a single object instance in the whole dataset. This can make the metric less robust. However, to ensure that our metric considers the predicted semantic categories, we adjust the matching between prediction and ground truth. While in standard detection the matching is based purely on the bounding boxes via intersection-over-union (since categories are already grouped), we include the index g_i^l that links a prediction with the object descriptions in D_i, see above in Sec. 3.1. Specifically, a prediction is matched to a ground truth only if the prediction and the ground truth point to the same object description (semantics) and the predicted and ground truth bounding boxes overlap sufficiently (localization).
- Standard detection ground truth exclusively assigns each object instance one semantic category. In contrast, our task requires multi-label predictions. For instance, "person" and "woman in red shirt" can refer to the same object. This needs to be considered in the matching process

of the evaluation. In contrast to standard detection, one ground truth box can be correctly matched with multiple predictions if the match happens via different object descriptions (recall index g_i^l above).

- Our object descriptions D_i contain both plain category names (like "car" or "person" from standard detection) as well as complex free-form text (like "blue sports car parked near left sidewalk"). We want our metric to give equal importance to both types. Due to the different number of ground truth instances, we first compute AP for both types separately and then take the harmonic mean. Different from the arithmetic mean, the harmonic mean requires good results on both types to achieve a high number on the final metric.

We implemented this evaluation protocol in Python and released it at `https://github.com/samschulter/omnilabeltools`

4. Dataset Collection

To establish our novel evaluation benchmark, we need images that annotate objects with bounding boxes and corresponding free-form text descriptions. To do so, we define a multi-step annotation and verification process. Fig. 4 and the following paragraphs describe the process.

Existing datasets: We start with the validation/test sets of COCO [30], Objects-365 [40], and OpenImages-V5 [21], which not only saves annotation cost for obtaining bounding boxes, but also helps collecting *diverse* object descriptions. By leveraging the (super-)category information when sampling images, we force annotators to provide descriptions also for rare categories. Otherwise, annotators will quickly pick simple and common categories to describe. And although reusing datasets may exclude some categories that were not annotated, the object descriptions we collect often include additional categories. For example, while a "bottle cap" is not part of the original categories, a description like "bottle with red cap" requires to understand "bottle cap".

Sample image / (super)category pairs: To encourage a diverse distribution of categories and images, we propose a strategy to randomly sample pairs of images and (super) categories. We first filter all possible pairs based on the following criteria: (a) At least two instances of a (super) category need to be present in the image. (b) For super-categories, at least two different sub-categories need to be present in the image. (c) To collect descriptions that focus on the object's appearance, relations and actions, we reject pairs with more than 10 instances, if the larger side of any instances' bounding box is smaller than 80 pixels, if the average over the bounding box's largest overlap with any other box is larger than 50%, or if any instance is flagged as covering a crowd of objects ("iscrowd"). Finally, we pick a random subset of the filtered pairs for annotation with free-form descriptions.

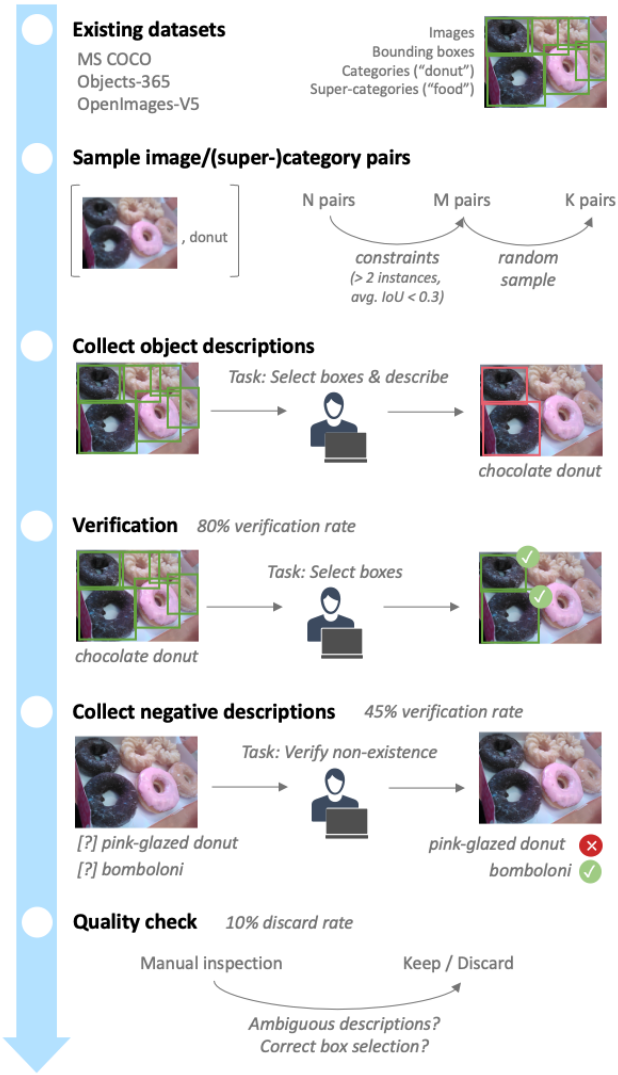

Figure 4: A step-by-step summary of our data collection and annotation process. Sec. 4 describes each step in detail. The key benefits of our annotation process are: (1) reuse of existing object detection datasets with annotated categories allows for a balanced sampling and consequently a set of diverse object description. (2) Contrary to existing free-form text benchmarks, we collect negative descriptions, which are related to an image but do not refer to any object.

Collect object descriptions: All initial object descriptions are collected with Amazon Mechanical Turk (AMT). Given an image/(super-)category pair, we draw the bounding boxes of that (super-)category's instances and request annotators to pick a subset of the instances and provide a text description that only matches their selection. Specifically, for *i*mage/category pairs with $N = 2$ possible instances, we ask to pick exactly one. If $N > 2$, we ask to select at least 2 but at most $N - 1$ instances. This ensures

that if an object description uses the category name itself, additional text is needed to distinguish the instances. For *i*mage/super-category pairs, we ask to select at least one instance, but avoid using the category names themselves in the descriptions. This encourages higher-level descriptions like "edible item" for all objects of the super-category "food". Finally, we ran a semi-automatic profanity check [52] on the collected text. We manually inspected 500 descriptions with the highest probability of containing profane language, but did not need to discard any description.

Verification: To ensure high quality descriptions, we again use AMT to verify the selection of bounding boxes from the previous step. We provide annotators the originally highlighted bounding boxes and the newly collected description and request to select the objects for which the description applies. We only keep descriptions for which both selections (initial and verification) are equal, which is about 80% of all descriptions.

Collect negative descriptions: As described earlier, a key aspect of our benchmark are negative object descriptions. These descriptions are related to an image, but do not actually refer to any object. To collect such descriptions, we leverage the already-collected free-form object descriptions with their underlying (super-)category information. Hence, a sample & verify approach is suitable, where, for each image/(super-)category pair, we randomly sample 5 object descriptions from the same (super-)category but a different image and ask 2 AMT annotators to confirm that the given description does not refer to any object. We then only keep negative descriptions with 2 confirmations, which was about 30% in our case.

Quality check: Finally, we perform a manual quality check. We fix misspellings and ambiguous descriptions when possible. If the meaning of the description changed, we keep the positive description, but discard all negative associations to other images. If an object description is entirely wrong, we discard it, which was the case for about 10% of the remaining descriptions.

Annotators: In total, 263 different annotators from AMT provided inputs for our annotations. For the three tasks using AMT (generating descriptions, verifying descriptions, and verifying negative descriptions), we had 54, 71, and 235 annotators, respectively.

5. Dataset Analysis

This section analyzes various statistics of OmniLabel and compares them with other related datasets. For all datasets, we analyze the corresponding validation sets.

5.1. Basic statistics

Tab. 2 summarizes key numbers of our OmniLabel dataset in comparison with prior benchmarks on referring expressions or visual grounding, specifically, the combination of RefCOCO/g/+ [33, 45], Flickr30k [36] and PhraseCut [42]. The key takeaways are: (a) The existence of negative objet descriptions (# neg). Like in standard detection, where categories not present in an image are considered negative and are still evaluated, OmniLabel provides free-form object descriptions that are related to the image but do not refer to any object. (b) The number of bounding boxes per description is higher than for any other dataset, which adds to the difficulty of the benchmark.

	RefCOCO/g/+	Flickr30k	PhraseCut	**OmniLabel**
# images	4.3K	1.0K	2.9K	12.2K
# descr.	26.5K	11.3K	19.5K	15.8K (16.8K)
# pos	26.5K	11.3K	19.5K	11.7K
# neg	0	0	0	9.4K
# boxes	10.2K	4.6K	32.1K	20.4K (165.7K)
# boxes/descr	1.0±0.0	1.0±0.0	1.6±1.6	1.7±1.0

Table 2: Basic statistics of the validation sets of OmniLabel, the combination of RefCOCO/g/+ [33, 45], Flickr30k [36] and PhraseCut [42]. All numbers are based only on free-form object descriptions. Numbers in parenthesis in the last column indicate statistics with plain categories included

5.2. Analysis of free-form object descriptions

While OmniLabel also contains plain categories like in standard object detection, our focus for this analysis is on the free-form object descriptions.

Part-Of-Speech (POS) tagging: To analyze the content and the diversity of our object descriptions, Fig. 5 shows an analysis of the words when grouped by part-of-speech tagging. On the left, we have the number of unique words (not counting multiple occurrences) based on a random subset of 10K descriptions. For adjectives, verbs and particularly nouns, OmniLabel covers more unique words, attributing to its diversity. On the right, we have the distribution of (non-unique) words among the different POS tags. We observe a more uniform distribution than other datasets, indicating longer descriptions (see below) that are closer to sentences, rather than short phrases or single words. On average, we have 2.04 ± 0.90 nouns, 0.62 ± 0.71 adjectives and 0.43 ± 0.61 verbs per object description.

Description lengths: Fig. 6 confirms our assumption from above that object descriptions in OmniLabel contain more words than other datasets.

6. Baselines

Beyond statistics of the collected annotations, we also evaluate recent language-based object detection models on OmniLabel with our novel evaluation metric.

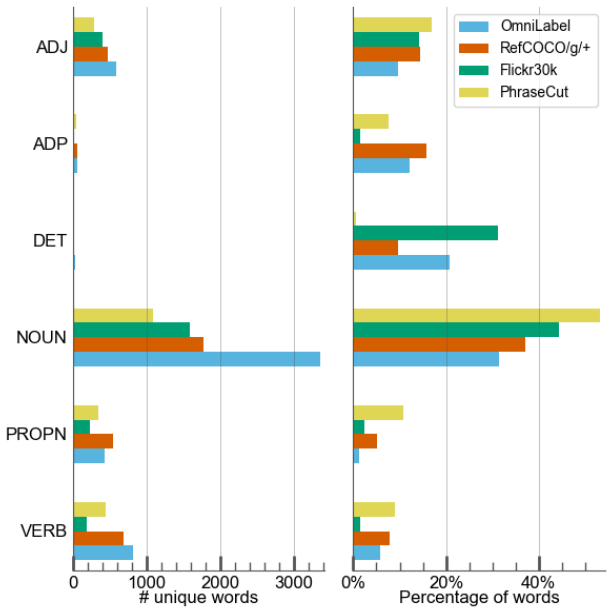

Figure 5: Grouping of the words in our object descriptions into relevant part-of-speech tags. We show the count of unique words (left) and the distribution of words (right). The statistics are computed from a random subset of 10K descriptions of each dataset.

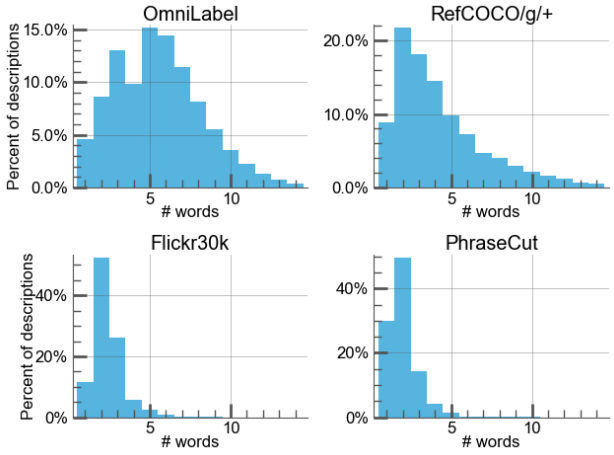

Figure 6: Histograms of description lengths (in number of words) for OmniLabel and three other datasets.

6.1. Models

Our evaluation aims to encompass a wide range of models, and we select them based on their performance on (a) standard detection benchmarks (like LVIS [16] and COCO [30]), and (b) tasks like Phrase Grounding and Referring Expression Compression. For models that primarily focus on open-vocabulary detection via large-scale pretraining, we utilize RegionCLIP [51] and Detic [53]. For models that are designed for text-conditioned detection with state-of-the-art performance on visual grounding, we use MDETR [18], GLIP [27], and FIBER [11]. We present a brief summary of each of these models.

RegionCLIP is an open-vocabulary object detector based on Faster RCNN [39]. It adopts pretrained CLIP's visual encoder (ResNet-50) [37] as the backbone and is finetuned with image-text pairs from the Internet (e.g. CC3M [41]). Thus, RegionCLIP is expected to get lower performance on our benchmark, compared to other baselines trained with detection and visual grounding datasets.

Detic is an open-vocabulary object detector that relies on CLIP [37] embeddings to encode class names. It utilizes a combination of box-level and image-level annotations, with a loss function that is weakly-supervised (modified Federated Loss [54]). For the results presented, we utilized Swin-Base [32] as the backbone.

MDETR is an end-to-end modulated detector that can detect objects for a given free-form text query, with a tight coupling between image and text modalities. The model is based on DETR [6] and trained with a combination of different visual grounding datasets (GoldG).

GLIP is a large scale visual grounding model that is trained with a combination of detection annotations, visual grounding data and image-text pairs. We evaluate its two variants, GLIP-T and GLIP-L. GLIP-T adopts Swin-Tiny [32] as the backbone and is trained with Objects365 [40], GoldG [18], CC3M, and SBU [35]. GLIP-L adopts Swin-Large [32] as the backbone and is trained with several detection datasets (including Objects365, OpenImages [21], and Visual Genome [20]), GoldG, CC12M [7], SBU, and additional 24M image-text pairs collected from the Internet.

FIBER-B introduces a two-stage pretraining strategy, from coarse- to fine-grained data, with image-text and image-text-box annotations, respectively. It generally follows the model design and the training protocol of GLIP, but adopts Swin-Base [32] as the backbone.

6.2. Results

We run two experiments with the above described models. The first one focuses on a detailed analysis of our new metric (see Sec. 3.2) on the OmniLabel dataset. The second experiment compares our metric on three different datasets.

Analysis on the OmniLabel dataset – Tab. 3: The first observation we make is that nearly all methods achieve higher accuracy on plain object categories (AP-categ) compared to free-form text descriptions (AP-descr). One can also clearly see the effect of using the geometric mean for the final metric (AP), when averaging over plain categories (AP-categ) and free-form descriptions (AP-descr). This effect is more pronounced for COCO images.

Images	Method	AP	AP-categ	AP-descr	AP-descr-pos	AP-descr-S	AP-descr-M	AP-descr-L
All	RegionCLIP [51]	2.7	2.7	2.6	3.2	3.6	2.7	2.3
	Detic [53]	8.0	15.6	5.4	8.0	5.7	5.4	6.2
	MDETR [18]	-	-	4.7	9.1	6.4	4.6	4.0
	GLIP-T [27]	19.3	23.6	16.4	25.8	29.4	14.8	8.2
	GLIP-L [27]	25.8	32.9	21.2	33.2	37.7	18.9	10.8
	FIBER-B [11]	25.7	30.3	22.3	34.8	38.6	19.5	12.4
COCO	RegionCLIP	4.1	5.1	3.5	5.1	6.1	3.3	4.1
	Detic	8.3	43.1	4.6	9.9	10.2	3.5	7.2
	MDETR	-	-	13.2	31.6	15.4	13.5	12.4
	GLIP-T	18.7	45.7	11.7	31.2	27.0	10.9	10.2
	GLIP-L	21.8	50.4	13.9	36.8	28.9	12.9	11.5
	FIBER-B	22.2	49.6	14.3	38.8	31.3	12.7	14.2
Objects-365	RegionCLIP	3.6	3.6	3.6	4.1	5.0	3.5	3.0
	Detic	9.1	21.6	5.7	8.4	6.6	5.9	6.9
	MDETR	-	-	3.2	5.9	3.0	3.2	2.7
	GLIP-T	22.6	30.0	18.1	26.9	34.2	16.0	9.1
	GLIP-L	29.3	37.5	24.0	35.2	44.5	20.5	11.8
	FIBER-B	30.8	37.9	25.9	38.2	44.7	22.5	14.1
OpenImages v5	RegionCLIP	2.3	2.1	2.7	2.9	3.4	2.7	2.0
	Detic	6.4	8.1	5.4	6.9	5.4	5.6	5.8
	MDETR	-	-	6.1	10.6	9.6	5.7	4.1
	GLIP-T	17.6	20.0	15.7	24.4	25.8	14.9	7.5
	GLIP-L	25.7	35.8	20.1	31.2	33.3	18.7	10.3
	FIBER-B	22.0	24.4	20.1	30.9	34.1	18.5	10.5

Table 3: Evaluation of language-based detection baselines on the OmniLabel benchmark with the metric described in Sec. 3.2. The final AP value is the geometric mean of only plain categories (AP-categ) and free-form descriptions (AP-descr). AP-descr-pos evaluates on only positive descriptions, clearly showcasing the impact of negative descriptions. AP-descr-S/M/L evaluate descriptions of different length (up to 3 words, 4 to 8, and more than 8)

A key takeaway message from Tab. 3 is the impact of negative descriptions. The performance gap between including negative descriptions in the label space (AP-descr) and excluding them (AP-descr-pos) is significant. The biggest gap can be observed for COCO images, which is because this part of the dataset contains the most negative descriptions relative to the number of images (due to our data collection process), see supplement). Another observation we get from Tab. 3 is that accuracy correlates negatively with description length. AP values are in general higher for shorter descriptions (AP-descr-S, up to three words) than for longer descriptions (AP-descr-L, more than 8 words).

Finally, we can see that GLIP-T/L and FIBER-B achieve the best results on OmniLabel. MDETR achieves reasonable results when only considering positive descriptions (AP-descr-pos) but fails when negatives are added (AP-descr), likely due to the specific training algorithm. Also, we did not report results of MDETR for AP or AP-categ due to the significant runtime induced by the large labelspace and MDETR's model design. As expected, Detic is good on plain categories (AP-categ) but underperforms on free-form descriptions (AP-descr). RegionCLIP's lower performance is likely due to a combination of a weaker backbone and the training data.

Method	OmniLabel		RefCOCOg	PhraseCut
	descr	descr-pos	descr	descr
RegionCLIP [51]	2.6	3.2	1.1	2.2
Detic [53]	5.4	8.0	6.8	6.8
GLIP-T [27]	16.4	25.8	32.1	23.9
GLIP-L [27]	21.2	33.2	33.4	29.3
FIBER-B [11]	22.3	34.8	33.0	27.4

Table 4: Comparing our evaluation metric (Sec. 3.2) for all models on three different datasets. OmniLabel poses a more difficult challenge, specifically when negative descriptions are included (descr vs. descr-pos)

Evaluation metric across different datasets – Tab. 4: We compare all models on three datasets (OmniLabel, RefCOCOg [33] and PhraseCut [42]). We make two main observations: First, OmniLabel is a more difficult benchmark, particularly because of negative descriptions. Second, the proposed evaluation metric from Sec. 3.2 is more stringent than the one used in RefCOCO/g/+. For instance, FIBER-B on RefCOCOg (val) achieves 87.1% accuracy [11] compared to the 33.0 from Tab. 4.

7. Conclusions

OmniLabel presents a novel benchmark for evaluating language-based object detectors. A key innovations is the annotation process, which (a) encourages free-form text descriptions of objects that are complex and diverse, (b) ensures collecting difficult examples with multiple instances of the same underlying category present in the images, and (c) provides negative free-form descriptions that are related but not present in an image. Moreover, OmniLabel defines a novel task setting and a corresponding evaluation metric. Our analysis of the dataset shows that we could indeed collect object descriptions that are diverse and contain more unique nouns, verbs and adjectives than existing benchmarks. Also, evaluating recent language-based object detectors confirmed the level of difficulty that OmniLabel poses to these models. We hope that our contributions in providing a challenging benchmark help progress the field towards robust object detectors that understand semantically rich and complex descriptions of objects.

References

[1] Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for attribute-based classification. In *CVPR*, 2013.

[2] Zeynep Akata, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. Evaluation of output embeddings for fine-grained image classification. In *CVPR*, 2015.

[3] Mykhaylo Andriluka, Stefan Roth, and Bernt Schiele. People-tracking-by-detection and people-detection-by-tracking. In *CVPR*, 2008.

[4] Ankan Bansal, Karan Sikka, Gaurav Sharma, Rama Chellappa, and Ajay Divakaran. Zero-shot object detection. In *ECCV*, pages 384–400, 2018.

[5] Zhaowei Cai, Gukyeong Kwon, Avinash Ravichandran, Erhan Bas, Zhuowen Tu, Rahul Bhotika, and Stefano Soatto. X-DETR: A versatile architecture for instance-wise vision-language tasks. In *ECCV*, 2022.

[6] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020.

[7] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 1M: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *CVPR*, 2021.

[8] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015.

[9] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016.

[10] Piotr Dollár, Christian Wojek, Bernt Schiele, and Pietro Perona. Pedestrian detection: An evaluation of the state of the art. *IEEE TPAMI*, 34(4):743–761, 2012.

[11] Zi-Yi Dou, Aishwarya Kamath, Zhe Gan, Pengchuan Zhang, Jianfeng Wang, Linjie Li, Zicheng Liu, Ce Liu, Yann LeCun, Nanyun Peng, et al. Coarse-to-Fine Vision-Language Pre-training with Fusion in the Backbone. In *NeurIPS*, 2022.

[12] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn., and Andrew Zisserman. The Pascal Visual Object Classes (VOC) Challenge. *IJCV*, 88(2):303–338, June 2010.

[13] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *CVPR*, 2012.

[14] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling Open-Vocabulary Image Segmentation with Image-Level Labels. In *ECCV*, 2022.

[15] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary Object Detection via Vision and Language Knowledge Distillation. In *ICLR*, 2022.

[16] Agrim Gupta, Piotr Dollár, and Ross Girshick. LVIS: A Dataset for Large Vocabulary Instance Segmentation. In *CVPR*, 2019.

[17] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision. In *ICML*, 2021.

[18] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. MDETR–Modulated Detection for End-to-End Multi-Modal Understanding. In *ICCV*, 2021.

[19] Zaid Khan, Vijay Kumar B.G., Xiang Yu, Samuel Schulter, Manmohan Chandraker, and Yun Fu. Single-Stream Multi-Level Alignment for Vision-Language Pretraining . In *ECCV*, 2022.

[20] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 123:32–73, 2017.

[21] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 2020.

[22] Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. Learning To Detect Unseen Object Classes by Between-Class Attribute Transfer. In *CVPR*, 2009.

[23] Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-Based Classification for Zero-Shot Visual Object Categorization. *IEEE TPAMI*, 36(3):453–465, 2014.

[24] Boyi Li, Kilian Q. Weinberger, Serge Belongie, Vladlen Koltun, and René Ranftl. Language-driven Semantic Segmentation. In *ICLR*, 2022.

[25] Chunyuan Li, Haotian Liu, Liunian Harold Li, Pengchuan Zhang, Jyoti Aneja, Jianwei Yang, Ping Jin, Houdong Hu, Zicheng Liu, Yong Jae Lee, and Jianfeng Gao. ELEVATER: A Benchmark and Toolkit for Evaluating Language-Augmented Visual Models. In *NeurIPS*, 2022.

[26] Junnan Li, Ramprasaath R. Selvaraju, Akhilesh Deepak Gotmare, Shafiq Joty, Caiming Xiong, and Steven Hoi. Align before fuse: Vision and language representation learning with momentum distillation. In *NeurIPS*, 2021.

[27] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, Kai-Wei Chang, and Jianfeng Gao. Grounded Language-Image Pre-training. In *CVPR*, 2022.

[28] Chuang Lin, Peize Sun, Yi Jiang, Ping Luo, Lizhen Qu, Gholamreza Haffari, Zehuan Yuan, and Jianfei Cai. Learning object-language alignments for open-vocabulary object detection. In *ICLR*, 2023.

[29] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature Pyramid Networks for Object Detection. In *CVPR*, 2017.

[30] 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.

[31] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: Single shot multibox detector. In *ECCV*, 2016.

[32] 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 *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.

[33] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L. Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *CVPR*, 2016.

[34] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulò, and Peter Kontschieder. The Mapillary Vistas Dataset for Semantic Understanding of Street Scenes. In *ICCV*, 2017.

[35] Vicente Ordonez, Girish Kulkarni, and Tamara Berg. Im2text: Describing images using 1 million captioned photographs. *Advances in neural information processing systems*, 24, 2011.

[36] Bryan A. Plummer, Liwei Wang, Christopher M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. *IJCV*, 123(1):74–93, 2017.

[37] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[38] Joseph Redmon and Ali Farhadi. YOLO9000: Better, Faster, Stronger. In *CVPR*, 2017.

[39] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *NIPS*, 2015.

[40] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Jing Li, Xiangyu Zhang, and Jian Sun. Objects365: A Large-scale, High-quality Dataset for Object Detection. In *ICCV*, 2019.

[41] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*, 2018.

[42] Chenyun Wu, Zhe Lin, Scott Cohen, Trung Bui, and Subhransu Maji. PhraseCut: Language-based Image Segmentation in the Wild. In *CVPR*, 2020.

[43] Shuo Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. WIDER FACE: A Face Detection Benchmark. In *CVPR*, 2016.

[44] Fisher Yu, Wenqi Xian, Yingying Chen, Fangchen Liu, Mike Liao, Vashisht Madhavan, and Trevor Darrell. BDD100K: A Diverse Driving Video Database with Scalable Annotation Tooling. *arXiv:1805.04687*, 2018.

[45] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C. Berg, and Tamara L. Berg. Modeling Context in Referring Expressions. In *ECCV*, 2016.

[46] Alireza Zareian, Kevin Dela Rosa, Derek Hao Hu, and Shih-Fu Chang. Open-Vocabulary Object Detection Using Captions. In *CVPR*, 2021.

[47] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M. Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. In *ICLR*, 2023.

[48] Haotian Zhang, Pengchuan Zhang, Xiaowei Hu, Yen-Chun Chen, Liunian Harold Li, Xiyang Dai, Lijuan Wang, Lu Yuan, Jenq-Neng Hwang, and Jianfeng Gao. GLIPv2: Unifying Localization and Vision-Language Understanding. In *NeurIPS*, 2022.

[49] Shiyu Zhao, Zhixing Zhang, Samuel Schulter, Long Zhao, Vijay Kumar B. G, Anastasis Stathopoulos, Manmohan Chandraker, and Dimitris Metaxas. Exploiting Unlabeled Data with Vision and Language Models for Object Detection. In *ECCV*, 2022.

[50] Xiangyun Zhao, Samuel Schulter, Gaurav Sharma, Yi-Hsuan Tsai, Manmohan Chandraker, and Ying Wu. Object Detection with a Unified Label Space from Multiple Datasets . In *ECCV*, 2020.

[51] Yiwu Zhong, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, et al. RegionCLIP: Region-based Language-Image Pretraining. In *CVPR*, 2022.

[52] Victor Zhou. profanity-check. `https://github.com/vzhou842/profanity-check`. Accessed: 2023-03-06.

[53] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting Twenty-thousand Classes using Image-level Supervision. In *ECCV*, 2022.

[54] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Probabilistic two-stage detection. *arXiv preprint arXiv:2103.07461*, 2021.

[55] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Simple multi-dataset detection. In *CVPR*, 2022.

[56] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. In *arXiv preprint arXiv:1904.07850*, 2019.

[57] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: Deformable Transformers for End-to-End Object Detection. In *ICLR*, 2021.