Discriminative Learning of Open-Vocabulary Object Retrieval and Localization by Negative Phrase Augmentation

Ryota Hinami[1,2] **and Shin'ichi Satoh**[2,1]
[1]The University of Tokyo, [2]National Institute of Infomatics
hinami@nii.ac.jp, satoh@nii.ac.jp

Abstract

Thanks to the success of object detection technology, we can retrieve objects of the specified classes even from huge image collections. However, the current state-of-the-art object detectors (such as Faster R-CNN) can only handle pre-specified classes. In addition, large amounts of positive and negative visual samples are required for training. In this paper, we address the problem of open-vocabulary object retrieval and localization, where the target object is specified by a textual query (e.g., a word or phrase). We first propose Query-Adaptive R-CNN, a simple extension of Faster R-CNN adapted to open-vocabulary queries, by transforming the text embedding vector into an object classifier and localization regressor. Then, for discriminative training, we then propose negative phrase augmentation (NPA) to mine hard negative samples which are visually similar to the query and at the same time semantically mutually exclusive of the query. The proposed method can retrieve and localize objects specified by a textual query from one million images in only 0.5 seconds with high precision.

1 Introduction

Our goal is to retrieve objects from large-scale image database and localize their spatial locations given a textual query. The task of object retrieval and localization has many applications such as spatial position-aware image searches (Hinami et al., 2017) and it recently has gathered much attention from researchers. While much of the previous work mainly focused on object instance retrieval wherein the query is an image (Shen et al., 2012; Tao et al., 2014; Tolias et al., 2016), recent approaches (Aytar and Zisserman, 2014; Hinami and Satoh, 2016) enable retrieval of more generic concepts such as an object category. Although such approaches are built on the recent successes of object detection including that of R-CNN (Girshick et al., 2014), object detection methods can generally handle only closed sets of categories (e.g., PASCAL 20 classes), which severely limits the variety of queries when they are used as retrieval systems. Open-vocabulary object localization is also a hot topic and many approaches are proposed to solve this problem (Plummer et al., 2015; Chen et al., 2017). However, most of them are not scalable to make them useful for large-scale retrieval.

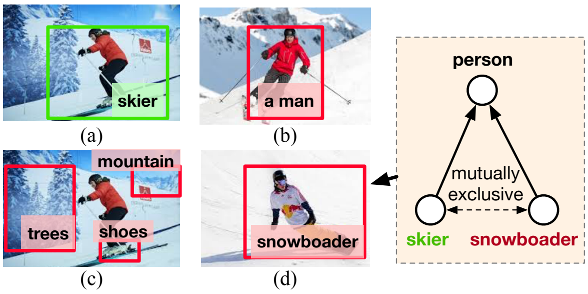


Figure 1: Training examples in open-vocabulary object detection. (a) positive example of skier classifier. (b) examples without positive annotation, which can be positive. (c) examples without positive annotation from an image that contains a positive example. (d) proposed approach to select hard and true negative examples by using linguistics knowledge.

We first describe *Query-Adaptive* R-CNN as an extension of the Faster R-CNN (Ren et al., 2015) object detection framework to open-vocabulary object detection simply by adding a component called a *detector generator*. While Faster R-CNN learns the class-specific linear classifier as learnable parameters of the neural network, we generate the weight of the classifier adaptively from text descriptions by learning the detector generator (Fig. 2b). All of its components can be trained in an end-to-end manner. In spite of its simple archi-

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2605–2615
Brussels, Belgium, October 31 - November 4, 2018. ©2018 Association for Computational Linguistics

tecture, it outperforms all state-of-the-art methods in the Flickr30k Entities phrase localization task. It can also be used for large-scale retrievals in the manner presented in (Hinami and Satoh, 2016).

However, training a discriminative classifier is harder in the open-vocabulary setting. Closed-vocabulary object detection models such as Faster R-CNN are trained using many negative examples, where a sufficient amount of good-quality negative examples is shown to be important for learning a discriminative classifier (Felzenszwalb et al., 2010; Shrivastava et al., 2016). While closed-vocabulary object detection can use all regions without positive labels as negative data, in open-vocabulary detection, it is not guaranteed that a region without a positive label is negative. For example, as shown in Fig. 1b, a region with the annotation `a man` is not always negative for `skier`. Since training data for open-vocabulary object detection is generally composed of images, each having region annotations with free descriptions, it is nearly impossible to do an exhaustive annotation throughout the dataset for all possible descriptions. Another possible approach is to use the regions without positive labels in the image that contains positive examples, as shown in Fig. 1c. Although they can be guaranteed to be positive by carefully annotating the datasets, negative examples are only limited to the objects that cooccur with the learned class.

To exploit negative data in open-vocabulary object detection, we use mutually exclusive relationships between categories. For example, an object with a label `dog` is guaranteed to be negative for the `cat` class because `dog` and `cat` are mutually exclusive. In addition, we propose an approach to select *hard negative* phrases that are difficult to discriminate (e.g., selecting `zebra` for `horse`). This approach, called *negative phrase augmentation (NPA)*, significantly improves the discriminative ability of the classifier and improves the retrieval performance by a large margin.

Our contributions are as follows. 1) We propose Query-Adaptive R-CNN, an extension of Faster R-CNN to open vocabulary, that is a simple yet strong method of open-vocabulary object detection and that outperforms all state-of-the-art methods in the phrase localization task. 2) We propose negative phrase augmentation (NPA) to exploit hard negative examples when training for open-vocabulary object detection, which makes the classifier more discriminative and robust to distractors in retrieval. Our method can accurately find objects amidst one million images in 0.5 second.

2 Related work

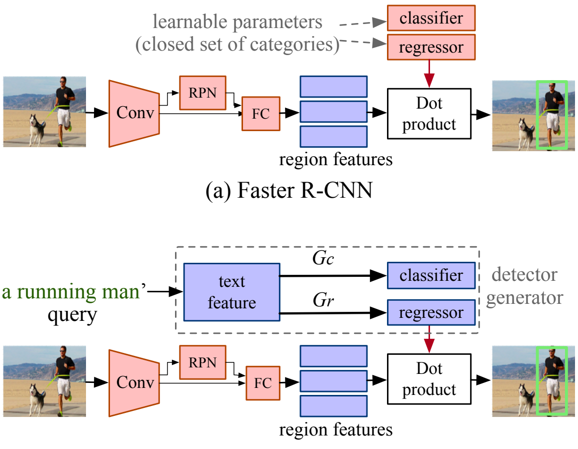
Phrase localization. Object grounding with natural language descriptions has recently drawn much attention and several tasks and approaches have been proposed for it (Guadarrama et al., 2014; Hu et al., 2016; Kazemzadeh et al., 2014; Mao et al., 2016; Plummer et al., 2015). The most related task to ours is the phrase localization introduced by Plummer et al. (Plummer et al., 2015), whose goal is to localize objects that corresponds to noun phrases in textual descriptions from an image. Chen et al. (Chen et al., 2017) is the closest to our work in terms of learning region proposals and performing regression conditioned upon a query. However, most phrase localization methods are not scalable and cannot be used for retrieval tasks. Some approaches (Plummer et al., 2017b; Wang et al., 2016a) learn a common subspace between the text and image for phrase localization. Instead of learning the subspace between the image and sentence as in standard cross-modal searches, they learn the subspace between a region and a phrase. In particular, Wang et al. (Wang et al., 2016a) use a deep neural network to learn the joint embedding of images and text; their training uses structure-preserving constraints based on structured matching. Although these approaches can be used for large-scale retrieval, their accuracy is not as good as recent state-of-the-art methods.

Object retrieval and localization. Object retrieval and localization have been researched in the context of particular object retrieval (Shen et al., 2012; Tao et al., 2014; Tolias et al., 2016), where a query is given as an image. Aytar et al. (Aytar and Zisserman, 2014) proposed retrieval and localization of generic category objects by extending the object detection technique to large-scale retrieval. Hinami and Satoh (Hinami and Satoh, 2016) extended the R-CNN to large-scale retrieval by using approximate nearest neighbor search techniques. However, they assumed that the detector of the category is given as a query and require many sample images with bounding box annotations in order to learn the detector. Several other approaches have used the external search engines (e.g., Google image search) to get training images from textual queries (Arandjelovi et al., 2012; Chatfield et al., 2015). Instead, we generate an object detector directly from the given textual query by using a neural network.

Parameter prediction by neural network. Query-Adaptive R-CNN generates the weights of the detector from the query instead of learning them by backpropagation. The dynamic filter network (De Brabandere et al., 2016) is one of the first methods that generate neural network parameters dynamically conditioned on an input. Several subsequent approaches use this idea in zero-shot learning (Ba et al., 2016) and visual question answering (Noh et al., 2016). Zhang et al. (Zhang et al., 2017) integrates this idea into the Fast R-CNN framework by dynamically generating the classifier from the text in a similar manner to (Ba et al., 2016). We extend this work to the case of large-scale retrieval. The proposed Query-Adaptive R-CNN generates the regressor weights and learn the region proposal network following Faster R-CNN. It enables precise localization with fewer proposals, which makes the retrieval system more memory efficient. In addition, we propose a novel hard negative mining approach, called negative phrase augmentation, which makes the generated classifier more discriminative.

3 Query-Adaptive R-CNN

Query-adaptive R-CNN is a simple extension of Faster R-CNN to open-vocabulary object detection. While Faster R-CNN detects objects of fixed categories, Query-Adaptive R-CNN detects any objects specified by a textual phrase. Figure 2 illustrates the difference between Faster R-CNN and Query-Adaptive R-CNN. While Faster R-CNN learns a class-specific classifier and regressor as parameters of the neural networks, Query-Adaptive R-CNN generates them from the query text by using a detector generator. Query-Adaptive R-CNN is a simple but effective method that surpasses state-of-the-art phrase localization methods and can be easily extended to the case of large-scale retrieval. Furthermore, its retrieval accuracy is significantly improved by a novel training strategy called negative phrase augmentation (Sec. 3.2).



(a) Faster R-CNN

(b) Query-Adaptive R-CNN

Figure 2: Difference in network architecture between (a) Faster R-CNN and (b) Query-Adaptive R-CNN. While Faster R-CNN learns the classifier of a closed set of categories as learnable parameters of neural networks, Query-Adaptive R-CNN generates a classifier and regressor adaptively from a query text by learning a detector generator that transforms the text into a classifier and regressor.

3.1 Architecture

The network is composed of two subnetworks: a *region feature extractor* and *detector generator*, both of which are trained in an end-to-end manner. The region feature extractor takes an image as input and outputs features extracted from sub-regions that are candidate objects. Following Faster R-CNN (Ren et al., 2015), regions are detected using a region proposal network (RPN) and the features of the last layer (e.g., fc7 in VGG network) are used as region features. The detector generator takes a text description as an input and outputs a linear classifier and regressor for the description (e.g., if `a dog` is given, `a dog` classifier and regressor are output). Finally, a confidence and a regressed bounding box are predicted for each region by applying the classifier and regressor to the region features.

Detector generator. The detector generator transforms the given text t into a classifier $\mathbf{w}^c$ and regressor $(\mathbf{w}^r_x, \mathbf{w}^r_y, \mathbf{w}^r_w, \mathbf{w}^r_h)$, where $\mathbf{w}_c$ is the weight of a linear classifier and $(\mathbf{w}^r_x, \mathbf{w}^r_y, \mathbf{w}^r_w, \mathbf{w}^r_h)$ is the weight of a linear regressor in terms of x, y, width w, and height h, following (Girshick et al., 2014). We first transform a text t of variable length into a text embedding vector $\mathbf{v}$. Other phrase localization approaches uses the Fisher vector encoding of word2vec (Klein et al., 2015; Plummer et al., 2015) or long-short term memory

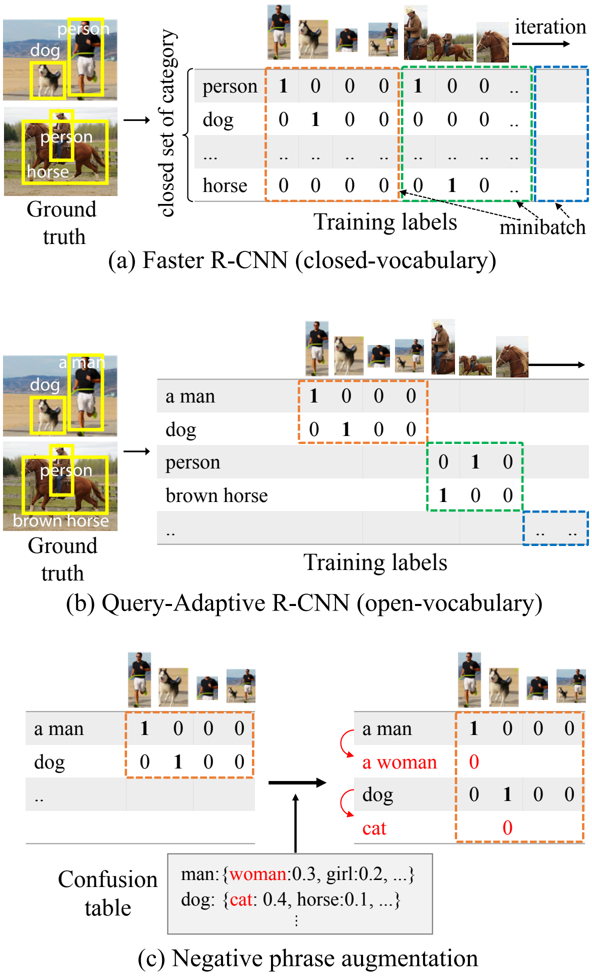
(LSTM) (Chen et al., 2017) for the phrase embedding. However, we found that the simple mean pooling of word2vec (Mikolov et al., 2013) performs better than these methods for our model (comparisons given in the supplemental material). The text embedding is then transformed into a detector, i.e., $\mathbf{w}_c = G_c(\mathbf{v})$ and $(\mathbf{w}_x^r, \mathbf{w}_y^r, \mathbf{w}_w^r, \mathbf{w}_h^r) = G_r(\mathbf{v})$. Here, we use a linear transformation for G_c (i.e., $\mathbf{w}_c = \mathbf{W}\mathbf{v}$, where $\mathbf{W}$ is a projection matrix). For the regressor, we use a multi-layer perceptron with one hidden layer to predict each of $(\mathbf{w}_x^r, \mathbf{w}_y^r, \mathbf{w}_w^r, \mathbf{w}_h^r) = G_r(\mathbf{v})$. We tested various architectures for G_r and found that sharing the hidden layer and reducing the dimension of the hidden layer (up to 16) does not adversely affect the performance, while at the same time it significantly reduces the number of parameters (see Sec. 5.2 for details).

3.2 Training with Negative Phrase Augmentation

All components of Query-Adaptive R-CNN can be jointly trained in an end-to-end manner. The training strategy basically follows that of Faster R-CNN. The differences are shown in Figure 3. Faster R-CNN is trained with the fixed closed set of categories (Fig. 3a), where all regions without a positive label can be used as negative examples. On the other hand, Query-Adaptive R-CNN is trained using the open-vocabulary phrases annotated to the regions (Fig. 3b), where sufficient negative examples cannot be used for each phrase compared to Faster R-CNN because a region without a positive label is not guaranteed to be negative in open-vocabulary object detection. We solve this problem by proposing negative phrase augmentation (NPA), which enables us to use good quality negative examples by using the linguistic relationship (e.g., mutually exclusiveness) and the confusion between the categories (Fig. 3c). It significantly improves the discriminative ability of the generated classifiers.

3.2.1 Basic Training

First, we describe the basic training strategy without NPA (Fig. 3b). Training a Query-Adaptive R-CNN requires the phrases and their corresponding bounding boxes to be annotated. For the ith image (we use one image as a minibatch), let us assume that C_i phrases are associated with the image. The C_i phrases can be considered as the classes to train in the minibatch. The labels $\mathbf{L}_i \in \{0,1\}^{C_i \times n_r}$



(a) Faster R-CNN (closed-vocabulary)

(b) Query-Adaptive R-CNN (open-vocabulary)

(c) Negative phrase augmentation

Figure 3: Difference in training between (a) closed-vocabulary and (b) open-vocabulary object detection. The approach of NPA is illustrated in (c).

are assigned to the region proposals generated by RPN (each of the dotted rectangles in Fig 3b); a positive label is assigned if the box overlaps the ground truth box by more than 0.5 in IoU and negative labels are assigned to other RoIs under the assumption that all positive objects of C_i classes are annotated (i.e., regions without annotations are negative within the image).[1] We then compute the classification loss by using the training labels and classification scores.[2] The loss in terms of RPN and bounding box regression is computed in the

[1] Although this assumption is not always true for datasets such as Flickr30k Entities, it nonetheless works well for them because exceptions are rare.

[2] Whereas Faster R-CNN uses the softmax cross entropy over the $C + 1$ (background) classes, where C is the number of closed sets of a category, we use the sigmoid cross entropy because the C_i classes are not always mutually exclusive and a background class cannot be defined in the context of open-vocabulary object detection.

same way as Faster R-CNN (Ren et al., 2015).

3.2.2 Negative Phrase Augmentation

Here, we address the difficulty of using negative examples in the training of open-vocabulary object detection. As shown in Fig. 1b, our generated classifier is not discriminative enough. The reason is the scarcity of negative examples when using the training strategy described in Sec. 3.2.1; e.g., the `horse` classifier is not learned with the `zebra` as a negative example except for the rare case that both a `zebra` and a `horse` are in the same image. Using hard negative examples has proven to be effective in the object detection to train a discriminative detector (Felzenszwalb et al., 2010; Girshick et al., 2014; Shrivastava et al., 2016). However, adding negative examples is usually not easy in the open-vocabulary setting, because it is not guaranteed that a region without a positive label is negative. For example, an object with the label `man` is not a negative of `person` even though `person` is not annotated. There are an infinite number of categories in open-vocabulary settings, which makes it difficult to exhaustively annotate all categories throughout the dataset.

How can we exploit hard examples that are guaranteed to be negative? We can make use of the mutually exclusive relationship between categories: e.g., an object with a `dog` label is negative for `cat` because `dog` and `cat` are mutually exclusive. There are two ways we can add to a minibatch: add negative images (regions) or negative phrases. Adding negative phrases (as in Fig. 3c) is generally better because it involves a much smaller additional training cost than adding images in terms of the both computational cost and GPU memory usage. In addition, to improve the discriminative ability of the classifier, we select only hard negative phrases by mining the confusing categories. This approach, called *negative phrase augmentation (NPA)*, is a generic way of exploiting hard negative examples in open-vocabulary object detection and leads to large improvements in accuracy, as we show in Sec. 5.3.

Confusion table. We create a confusion table that associates a category with its hard negative categories, from which negative phrases are picked as illustrated in Fig. 3c. To create the entry for category c, we first generate the candidate list of hard negative categories by retrieving the top 500 scored objects from all objects in the validation set of Visual Genome (Krishna et al., 2016) (using c as a query). After that, we remove the mutually non-exclusive category relative to c from the list. Finally, we aggregate the list by category and assign a weight to each category. Each of the registered entries becomes like `dog:{cat:0.5, horse:0.3, cow:0.2}`. The weight corresponds to the probability of selecting the category in NPA, which is computed based on the number of appearances and their ranks in the candidate list.[3]

Removal of mutually non-exclusive phrases. To remove non-mutually exclusive phrases from the confusion table, we use two approaches that estimate whether the two categories are mutually exclusive or not. 1) The first approach uses the *WordNet hierarchy*: if two categories have parent-child relationships in WordNet (Miller, 1995), they are not mutually exclusive. However, the converse is not necessarily true; e.g., `man` and `skier` are not mutually exclusive but do not have the parent-child relationship in the WordNet hierarchy. 2) As an alternative approach, we propose to use *Visual Genome annotation*: if two categories co-occur more often in the Visual Genome dataset (Krishna et al., 2016), these categories are considered to be not mutually exclusive.[4] These two approaches are complementary, and they improve detection performance by removing the mutually non-exclusive words (see Sec. 5.3).

The training pipeline with NPA is as follows:

(1) **Update the confusion table:** The confusion table is updated periodically (after every 10k iterations in our study). Entries were created for categories that frequently appeared in 10k successive batches (or the whole training set if the size of the dataset is not large).

(2) **Add hard negative phrases:** Negative phrases are added to each of the C_i phrases in a minibatch. We replace the name of the category in each phrase with its hard negative category (e.g., generate `a running woman` for `a running man`), where the category name is obtained by extracting nouns. A negative phrase is randomly selected from the confusion table on the basis of the assigned probability.

[3] We compute the weight of each category as the sum of 500 minus the rank for all ranked results in the candidate lists normalized over all categories in order to sum to one.

[4] We set the ratio at 1% of objects in either category. For example, if there are 1000 objects with the `skier` label and 20 of those objects are also annotated with `man` (20/1000=2%), we consider that `skier` and `man` are not mutually exclusive.

(3) **Add losses:** As illustrated in Fig. 3c, we only add negative labels to the regions where a positive label is assigned to the original phrase. The classification loss is computed only for the regions, which is added to the original loss.

4 Large-Scale Object Retrieval

Query-Adaptive R-CNN can be used for large-scale object retrieval and localization, because it can be decomposed into a query-independent part and a query-dependent part, i.e., a region feature extractor and detector generator. We follow the approach used in large-scale R-CNN (Hinami and Satoh, 2016), but we overcome its two critical drawbacks. First, a large-scale R-CNN can only predict boxes included in the region proposals; these are detected offline even though the query is unknown at the time; therefore, to get high recall, a large number of object proposals should be used, which is memory inefficient. Instead, we generate a regressor as well as a classifier, which enables more accurate localization with fewer proposals. Second, a large-scale R-CNN assumes that the classifier is given as a query, and learning a classifier requires many samples with bounding annotations. We generate the classifier from a text query directly by using the detector generator of Query-Adaptive R-CNN. The resulting system is able to retrieve and localize objects from a database with *one million images* in *less than one second.*

Database indexing. For each image in the database, the region feature extractor extracts region proposals and corresponding features. We create an index for the region features in order to speed up the search. For this, we use the IVFADC system (Jégou et al., 2011) in the manner described in (Hinami and Satoh, 2016).

Searching. Given a text query, the detector generator generates a linear classifier and bounding box regressor. The regions with high classification scores are then retrieved from the database by making an IVFADC-based search. Finally, the regressor is applied to the retrieved regions to obtain the accurately localized bounding boxes.

5 Experiments

5.1 Experimental Setup

Model: Query-Adaptive R-CNN is based on VGG16 (Simonyan and Zisserman, 2015), as in other work on phrase localization. We first initialized the weights of the VGG and RPN by using Faster R-CNN trained on Microsoft COCO (Lin et al., 2014); the weights were then fine-tuned for each dataset of the evaluation. In the training using Flickr30k Entities, we first pretrained the model on the Visual Genome dataset using the object name annotations. We used Adam (Kingma and Ba, 2015) with a learning rate starting from 1e-5 and ran it for 200k iterations.

Tasks and datasets: We evaluated our approaches on two tasks: phrase localization and open-vocabulary object detection and retrieval. The **phrase localization task** was performed on the Flickr30k Entities dataset (Plummer et al., 2015). Given an image and a sentence that describes the image, the task was to localize region that corresponds to the phrase in a sentence. Flickr30k datasets contain 44,518 unique phrases, where the number of words of each phrase is 1–8 (2.1 words on average). We followed the evaluation protocol of (Plummer et al., 2015). We did not use Flickr30k Entities for the retrieval task because the dataset is not exhaustively annotated (e.g., not all men appearing in the dataset are annotated with `man`), which makes it difficult to evaluate with a retrieval metric such as AP, as discussed in Plummer et al. (Plummer et al., 2017b). Although we cannot evaluate the retrieval performance directly on the phrase localization task, we can make comparisons with other approaches and show that our method can handle a wide variety of phrases.

The **open-vocabulary object detection and retrieval task** was evaluated in the same way as the standard object detection task. The difference was the assumption that we do not know the target category at training time in open-vocabulary settings; i.e., the method does not tune in to a specific category, unlike the standard object detection task. We used the Visual Genome dataset (Krishna et al., 2016) and selected the 100 most frequently object categories as queries among its 100k or so categories.[5] [6] We split the dataset into training, validation, and test sets following (Johnson et al., 2016). We also evaluated our approaches on the PASCAL VOC 2007 dataset, which is a widely used dataset

[5] Since the WordNet synset ID is assigned to each object, we add objects with labels of hyponyms as positives (e.g., `man` is positive for the `person` category).

[6] We exclude the background (e.g., `grass`, `sky`, `field`), multiple objects (e.g., `people`, `leaves`), and ambiguous categories (e.g, `top`, `line`).

Approach	People	Clothing	Body	Animals	Vehicles	Instruments	Scene	Other	All
Non-scalable methods									
GroundeR (Rohrbach et al., 2016)	61.00	38.12	10.33	62.55	68.75	36.42	58.18	29.08	47.81
Multimodal compact bilinear (Fukui et al., 2016)	-	-	-	-	-	-	-	-	48.69
PGN+QRN (Chen et al., 2017)	75.08	55.90	20.27	73.36	68.95	45.68	65.27	38.80	60.21
Non-scalable and joint localization methods									
Structured matching (Wang et al., 2016b)	57.89	34.61	15.87	55.98	52.25	23.46	34.22	26.23	42.08
SPC+PPC (Plummer et al., 2017a)	71.69	50.95	25.24	76.25	66.50	35.80	51.51	35.98	55.85
QRC net (Chen et al., 2017)	76.32	59.58	25.24	**80.50**	**78.25**	50.62	67.12	43.60	65.14
Scalable methods									
Structure-preserving embedding (Wang et al., 2016a)	-	-	-	-	-	-	-	-	43.89
CCA+Detector+Size+Color (Plummer et al., 2017b)	64.73	46.88	17.21	65.83	68.75	37.65	51.39	31.77	50.89
Query-Adaptive R-CNN (proposed)	**78.17**	**61.99**	**35.25**	74.41	76.16	**56.69**	**68.07**	**47.42**	**65.21**

Table 1: **Phrase localization** accuracy on Flickr30k Entities dataset.

		IoU				
Architecture	Params	0.5	0.6	0.7	0.8	0.9
w/o regression	-	**65.21**	53.19	35.70	14.32	1.88
300–16(–4096)	**0.3M**	64.14	57.66	48.22	33.04	9.29
300–64(–4096)	1.1M	63.87	57.43	**49.05**	33.84	**10.55**
300–256(–4096)	4.3M	63.84	57.70	48.71	33.87	10.05
300–1024(–4096)	17M	64.29	**58.05**	48.49	**33.94**	10.09
300(–256–4096)	4.5M	62.82	56.28	48.02	32.71	9.89
300–4096	1.2M	63.23	56.92	48.17	32.66	9.20

Table 2: Comparison of various **bounding box regressors** on Flickr30k Entities for different IoU thresholds. The number of parameters in G_r is also shown.

for object detection.[7] As metrics, we used top-k precision and average precision (AP), computed from the region-level ranked list as in the standard object detection task.[8]

5.2 Phrase localization

Comparison with state-of-the-art. We compared our method with state-of-the-art methods on the Flickr30k Entities phrase localization task. We categorized the methods into two types, i.e., non-scalable and scalable methods (Tab. 1). 1) *Non-scalable methods* cannot be used for large-scale retrieval because their query-dependent components are too complex to process a large amount of images online, and 2) *Scalable methods* can be used for large-scale retrieval because their query-dependent components are easy to scale up (e.g., the L_2 distance computation); these include common subspace-based approaches such as CCA. Our method also belongs to the scalable category. We used a simple model without a regressor and NPA in the experiments.

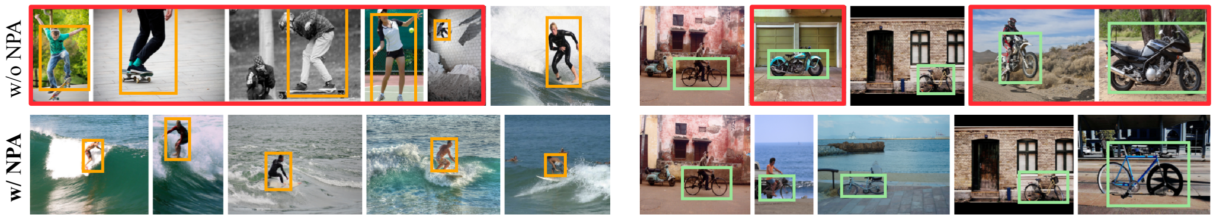
Table 1 compares Query-Adaptive R-CNN with the state-of-the-art methods. Our model achieved *65.21%* in accuracy and outperformed all of the previous state-of-the-art models including the non-scalable or joint localization methods. Moreover, it significantly outperformed the scalable methods, which suggests the approach of predicting the classifier is better than learning a common subspace for the open-vocabulary detection problem.

Bounding box regressor. To demonstrate the effectiveness of the bounding box regressor for precise localization, we conducted evaluations with the regressor at different IoU thresholds. As explained in Sec. 3.1, the regressor was generated using G_r, which transformed 300-d text embeddings x into 4096-d regressor weights $\mathbf{w}_x^r$, $\mathbf{w}_y^r$, $\mathbf{w}_w^r$, and $\mathbf{w}_h^r$. We compared three network architectures for G_r: 1) `300-n(-4096)` MLP having a hidden layer with n units that is shared across the four outputs, 2) `300(-n-4096)` MLP having a hidden layer that is not shared, and 3) `300(-4096)` linear transformation (without a hidden layer).

Table 2 shows the results with and without regressor. The regressor significantly improved the accuracy with high IoU thresholds, which demonstrates that the regressor improved the localization accuracy. In addition, the accuracy did not decrease as a result of sharing the hidden layer or reducing the number of units in the hidden layer. This suggests that the regressor lies in a very low-dimensional manifold because the regressor for one concept can be shared by many concepts (e.g., the `person` regressor can be used for `man`, `woman`, `girl`, `boy`, etc.). The number of parameters was significantly reduced by these tricks,

[7] We used the model trained on Visual Genome even for the evaluation on the PASCAL dataset because of the assumption that the target category is unknown.

[8] We did not separately evaluate the detection and retrieval tasks because both can be evaluated with the same metric.



Query: surfer

Query: bicycle

Figure 4: Qualitative results with and without NPA. Top-k retrieved results for two queries are shown (sorted by rank) and false alarms are depicted with a red border.

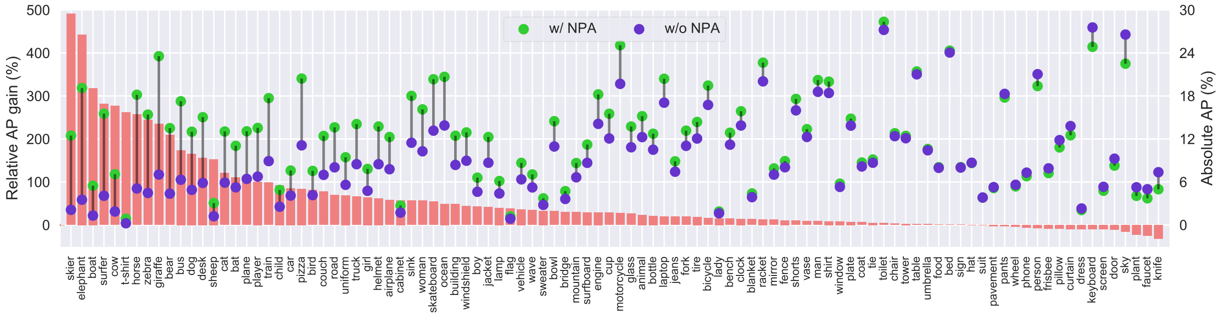


Figure 5: AP gain by **negative phrase augmentation** (NPA) for individual queries. The bars show the relative AP gain and points shows the absolute AP with and without NPA.

				Visual Genome			VOC
	NPA	WN	VG	mAP	PR@10	PR@100	mAP
CCA				3.18	20.40	15.64	28.23
Query-Adaptive R-CNN				9.15	52.60	36.85	29.14
	✓			10.90	60.10	43.21	36.74
	✓	✓		11.53	61.80	45.91	37.07
	✓		✓	11.65	65.40	46.85	41.32
	✓	✓	✓	**12.19**	**65.70**	**48.45**	**42.81**

Table 3: **Open-vocabulary object detection** performance on Visual Genome and PASCAL VOC 2007 datasets. WN and VG are the strategies to remove mutually non-exclusive phrases.

Query	Most confusing class				2nd most confusing class			
girl	man	19	→	3	boy	4	→	2
skateboard	surfboard	12	→	0	snowboard	11	→	0
train	bus	17	→	1	oven	3	→	0
helmet	hat	18	→	1	cap	6	→	4
elephant	bear	14	→	0	horse	6	→	0

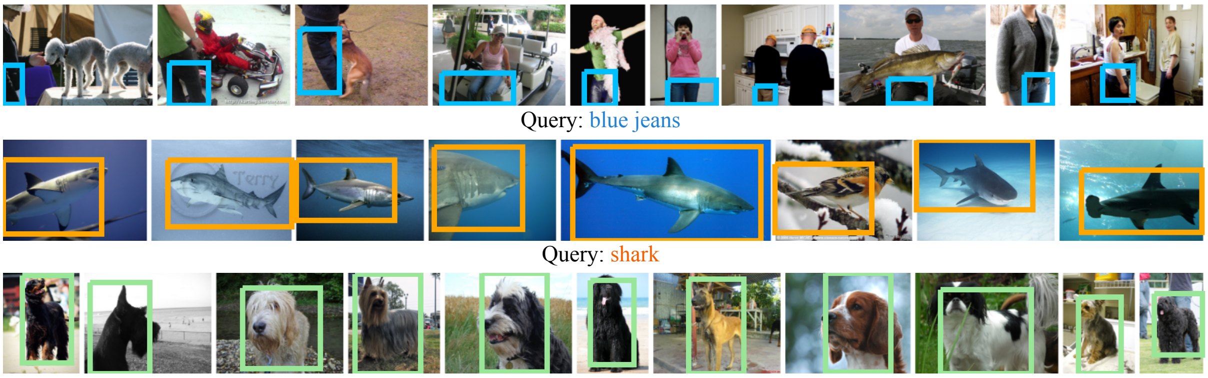
Table 4: Number of false alarms in top 100 results for five queries (**w/o NPA** → **w/ NPA**). The top 2 confusing categories are shown for each query.

to even fewer than in the linear transformation. The accuracy slightly decreased with a threshold of 0.5, because the regressor was not learned properly for the categories that did not frequently appear in the training data.

5.3 Open-Vocabulary Object Retrieval

Main comparison. Open-vocabulary object detection and retrieval is a much more difficult task than phrase localization, because we do not know how many objects are present in an image. We used NPA to train our model. As explained in Sec. 3.2.2, we used two strategies, *Visual Genome annotation (VG)* and *WordNet hierarchy (WN)*, to remove mutually non-exclusive phrases from the confusion table. As a baseline, we compared with region-based CCA (Plummer et al., 2017b), which is scalable and shown to be effective for phrase localization; for a fair comparison, the subspace was learned using the same dataset as ours. An approximate search was not used to evaluate the actual performance at open-vocabulary object detection.

Table 3 compares different training strategies. NPA significantly improved the performance: *more than 25% relative improvement* for all metrics. Removing mutually non-exclusive words also contributed the performance: WN and VG both improved performance (5.8% and 6.9% relative AP gain, respectively). Performance improved even further by combining them (11.8% relative AP gain), which shows they are complementary. AP was much improved by NPA for the PASCAL dataset as well (47% relative gain). However, the performance was still much poorer than those of the state-of-the-art object detection



Query: blue jeans

Query: shark

Query: black dog

Figure 6: Retrievals from one million images. Top-k results for three queries are shown.

methods (Redmon and Farhadi, 2017; Ren et al., 2015), which suggests that there is a large gap between open-vocabulary and closed-vocabulary object detection.

Detailed results of NPA. To investigate the effect of NPA, we show the AP with and without NPA for individual categories in Figure 5, which are sorted by relative AP improvement. It shows that AP improved especially for animals (`elephant`, `cow`, `horse`, etc.) and person (`skier`, `surfer`, `girl`), which are visually similar within the same upper category. Table 4 shows the most confused category and its total count in the top 100 search results for each query, which shows what concept is confusing for each query and how much the confusion is reduced by NPA.[9] This shows that visually similar categories resulted in false positive without NPA, while their number was suppressed by training with NPA. The reason is that these confusing categories were added for negative phrases in NPA, and the network learned to reject them. Figure 4 shows the qualitative search results for each query with and without NPA (and CCA as a baseline), which also showed that NPA can discriminate confusing categories (e.g., `horse` and `zebra`). These results clearly demonstrate that NPA significantly improves the discriminative ability of classifiers by adding hard negative categories.

Large-scale experiments. Finally, we evaluated the scalability of our method on a large image database. We used one million images from the ILSVRC 2012 training set for this evaluation. Table 5 show the speed and memory. The mean and standard deviation of speed are computed over 20 queries in PASCAL VOC dataset. Our system could retrieve objects from one million images in around 0.5 seconds. We did not evaluate accuracy because there is no such large dataset with bounding box annotations.[10] Figure 6 shows the retrieval results from one million images, which demonstrates that our system can accurately retrieve and localize objects from a very large-scale database.

Database size	10K	50K	100K	500K	1M
Time (ms)	183±16	196±21	242±28	314±90	484±165
Memory (GB)	0.46	1.23	2.19	9.87	19.47

Table 5: Speed/memory in **large-scale** experiments.

6 Conclusion

Query-Adaptive R-CNN is a simple yet strong framework for open-vocabulary object detection and retrieval. It achieves state-of-the-art performance on the Flickr30k phrase localization benchmark and it can be used for large-scale object retrieval by textual query. In addition, its retrieval accuracy can be further increased by using a novel training strategy called negative phrase augmentation (NPA) that appropriately selects hard negative examples by using their linguistic relationship and confusion between categories. This simple and generic approach significantly improves the discriminative ability of the generated classifier.

Acknowledgements: This work was supported by JST CREST JPMJCR1686 and JSPS KAKENHI 17J08378.

[9] For each query, we scored all the objects in the Visual Genome testing set and counted the false alarms in the top 100 scored objects.

[10] adding distractors would also be difficult, because we cannot guarantee that relevant objects are not in the images.

References

Relja Arandjelovi, Andrew Zisserman, Relja Arandjelovic, Andrew Zisserman, Relja Arandjelovi, Andrew Zisserman, Relja Arandjelovic, and Andrew Zisserman. 2012. Multiple queries for large scale specific object retrieval. In *BMVC*.

Yusuf Aytar and Andrew Zisserman. 2014. Immediate, scalable object category detection. In *CVPR*.

Jimmy Lei Ba, Kevin Swersky, Sanja Fidler, and Ruslan Salakhutdinov. 2016. Predicting deep zero-shot convolutional neural networks using textual descriptions. In *ICCV*.

Ken Chatfield, Relja Arandjelovi, Andrew Zisserman, Relja Arandjelović, Omkar Parkhi, and Andrew Zisserman. 2015. On-the-fly learning for visual search of large-scale image and video datasets. *International Journal of Multimedia Information Retrieval*, 4(2):75–93.

Kan Chen, Rama Kovvuri, and Ram Nevatia. 2017. Query-guided Regression Network with Context Policy for Phrase Grounding. In *ICCV*.

Bert De Brabandere, Xu Jia, Tinne Tuytelaars, and Luc Van Gool. 2016. Dynamic filter networks. In *NIPS*.

Pedro F Felzenszwalb, Ross B Girshick, David McAllester, Deva Ramanan, and David Forsyth. 2010. Object detection with discriminatively trained part-based models. *PAMI*, 32(9):1627–45.

Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. 2016. Multimodal compact bilinear pooling for visual question answering and visual grounding. In *EMNLP*.

Ross Girshick, Jeff Donahue, Trevor Darrell, U C Berkeley, and Jitendra Malik. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*.

Sergio Guadarrama, Erik Rodner, Kate Saenko, Ning Zhang, Ryan Farrell, Jeff Donahue, and Trevor Darrell. 2014. Open-vocabulary object retrieval. *Robotics: Science and Systems*, 2(5):1–9.

Ryota Hinami, Yusuke Matsui, and Shin'ichi Satoh. 2017. Region-based image retrieval revisited. In *ACMMM*.

Ryota Hinami and Shin'ichi Satoh. 2016. Large-scale r-cnn with classifier adaptive quantization. In *ECCV*.

Ronghang Hu, Huazhe Xu, Marcus Rohrbach, Jiashi Feng, Kate Saenko, and Trevor Darrell. 2016. Natural language object retrieval. In *CVPR*.

Hervé Jégou, Matthijs Douze, and Cordelia Schmid. 2011. Product quantization for nearest neighbor search. *PAMI*, 33(1):117–128.

Justin Johnson, Andrej Karpathy, and Li Fei-Fei. 2016. Densecap: fully convolutional localization networks for dense captioning. In *CVPR*.

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara L Berg. 2014. Referitgame: referring to objects in photographs of natural scenes. In *EMNLP*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: a method for stochastic optimization. In *ICLR*.

Benjamin Klein, Guy Lev, Gil Sadeh, and Lior Wolf. 2015. Fisher vectors derived from hybrid gaussian-laplacian mixture models for image annotation. In *CVPR*.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalanditis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, Li Fei-Fei, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Fei-Fei Li. 2016. Visual genome: connecting language and vision using crowdsourced dense image annotations. *IJCV*, page 44.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. 2014. Microsoft coco: common objects in context. In *ECCV*.

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

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*.

George A. Miller. 1995. Wordnet: a lexical database for English. *Communications of the ACM*, 38(11):39–41.

Hyeonwoo Noh, Paul Hongsuck Seo, and Bohyung Han. 2016. Image question answering using convolutional neural network with dynamic parameter prediction. In *CVPR*.

Bryan A Plummer, Christopher M Cervantes, and C V Aug. 2017a. Phrase localization and visual relationship Detection with Comprehensive Image-Language Cues. In *ICCV*.

Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2015. Flickr30k entities: collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*.

Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2017b. Flickr30k Entities: Collecting Region-to-Phrase Correspondences for Richer Image-to-Sentence Models. *International Journal of Computer Vision*, 123(1):74–93.

Joseph Redmon and Ali Farhadi. 2017. YOLO9000: better, faster, stronger. In *CVPR*.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: towards real-time object detection with region proposal networks. In *NIPS*.

Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele. 2016. Grounding of textual phrases in images by reconstruction. In *ECCV*.

Xiaohui Shen, Zhe Lin, Jonathan Brandt, Shai Avidan, and Ying Wu. 2012. Object retrieval and localization with spatially-constrained similarity measure and k-nn re-ranking. In *CVPR*.

Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. 2016. Training region-based object detectors with online hard example mining. In *CVPR*.

Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In *ICLR*.

Ran Tao, Efstratios Gavves, Cees G M Snoek, and Arnold W M Smeulders. 2014. Locality in generic instance search from one example. In *CVPR*.

Giorgos Tolias, Ronan Sicre, and Hervé Jégou. 2016. Particular object retrieval with integral max-pooling of cnn activations. In *ICLR*.

Liwei Wang, Yin Li, and Svetlana Lazebnik. 2016a. Learning deep structure-preserving image-text embeddings. In *CVPR*.

Mingzhe Wang, Mahmoud Azab, Noriyuki Kojima, Rada Mihalcea, and Jia Deng. 2016b. Structured matching for phrase localization. In *ECCV*.

Yuting Zhang, Luyao Yuan, Yijie Guo, Zhiyuan He, I-An Huang, and Honglak Lee. 2017. Discriminative Bimodal Networks for Visual Localization and Detection with Natural Language Queries. *CVPR*.