Object Detection Meets Knowledge Graphs

Yuan Fang, Kingsley Kuan, Jie Lin, Cheston Tan and **Vijay Chandrasekhar**
Institute for Infocomm Research, A*STAR, Singapore
yfang@i2r.a-star.edu.sg, kingsley.kuan@gmail.com, {lin-j,cheston-tan,vijay}@i2r.a-star.edu.sg

Abstract

Object detection in images is a crucial task in computer vision, with important applications ranging from security surveillance to autonomous vehicles. Existing state-of-the-art algorithms, including deep neural networks, only focus on utilizing features within an image itself, largely neglecting the vast amount of background knowledge about the real world. In this paper, we propose a novel framework of *knowledge-aware object detection*, which enables the integration of external knowledge such as knowledge graphs into any object detection algorithm. The framework employs the notion of *semantic consistency* to quantify and generalize knowledge, which improves object detection through a re-optimization process to achieve better consistency with background knowledge. Finally, empirical evaluation on two benchmark datasets show that our approach can significantly increase recall by up to 6.3 points without compromising mean average precision, when compared to the state-of-the-art baseline.

1 Introduction

Many computer vision tasks ultimately seek to interpret the world through images and videos. While significant progress has been made in the past decade, there still exists a striking gap between how humans and machines learn. Although current machine learning approaches, including state-of-the-art deep learning algorithms, can effectively find patterns from the training data, they fail to leverage what an average person has at his or her disposal—the vast amount of background knowledge about the real world. Given that images and videos are reflections of the world, exploiting background knowledge can have a tremendous advantage towards interpreting these data.

Task and insight

In this paper, we study the key computer vision task of object detection [Everingham *et al.*, 2010]. Given an image, the goal is to identify a set of regions or bounding boxes, and to further classify each bounding box with one of the pre-defined object labels, as illustrated in Figure 1.

(a) Detecting `cat` and `table` (b) Detecting `bear`

Figure 1: Object detection on images from MSCOCO15.

Recent advances in deep convolutional neural networks [Sermanet *et al.*, 2013; Girshick *et al.*, 2014], in particular Fast or Faster R-CNN [Girshick, 2015; Ren *et al.*, 2015], show great promise in object detection. However, like previous approaches, these methods only account for patterns present in the training images, without leveraging much of the knowledge an average person would have. For example, humans have the common sense or implicit knowledge that a domestic `cat` sometimes sits on a table, but a `bear` does not barring very rare circumstances. This background knowledge would natrually help reinforce the simultaneous detections of `cat` and `table` (e.g. in Figure 1a), even if none of the training images portrays a `cat` together with a `table`. On the other hand, if an image is predicted to contain both `bear` and `table`, which conflicts with our background knowledge, the detections are more prone to be false.

While such background knowledge appears random and difficult to organize, there have been extensive research and commercial efforts to encode it into machine readable forms often known as knowledge graphs [Paulheim, 2017]. A knowledge graph is a graph that models semantic knowledge, where each node is a real-world concept, and each edge represents a relationship between two concepts. For instance, Figure 2 showcases a toy knowledge graph. In particular, the relationship "`cat` sits on `table`" reinforces the detections of `cat` and `table` in Figure 1a. We note that knowledge graphs already demonstrate considerable success in other domains such as Web search and social networks [Dong *et al.*, 2014]. Beyond a toy graph, large-scale knowledge graphs are often constructed through crowdsourcing or automated extraction from semi-structured and unstructured data, which are beyond the scope of this paper.

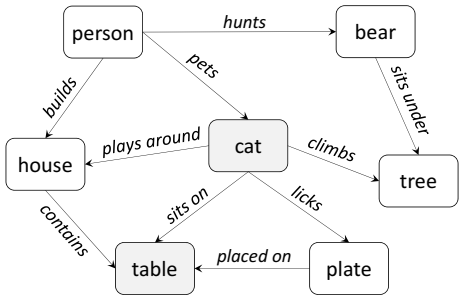


Figure 2: A toy knowledge graph modeling seven concepts as nodes (e.g., `cat` and `table`), as well as their relationships as edges (e.g., "`cat` sits on `table`").

Challenges and approach

Even with an existing knowledge graph, to effectively leverage the knowledge therein for object detection, two major technical challenges still remain.

First, how do we *quantify* and *generalize* knowledge? Quantification is necessary as knowledge graphs entail symbolic representations but most object detection algorithms operate over subsymbolic or numerical representations. Moreover, the quantification shall not only apply to images with contexts matching directly observed knowledge, but also generalize to images with new contexts. In our approach, for every pair of concepts on the knowledge graph, we compute a numerical degree of *semantic consistency* for them. For example, since the relationship "`cat` sits on `table`" is present on the knowledge graph, `cat` and `table` are semantically consistent concepts, but `bear` and `table` are not. Concepts can also be connected through a chain of indirect relationships, such as "`cat` licks `plate`" and "`plate` placed on `table`". This gives rise to the generalization ability—we can infer that `cat` and `table` tend to appear together without directly observing "`cat` sits on `table`".

Second, how do we incorporate semantic consistency to achieve *knowledge-aware object detection*? We hinge on the key constraint that more semantically consistent concepts are more likely to occur in an image with comparable probability. For instance, letting (`o`, p) denote a bounding box containing object `o` with probability p, it is more plausible to have two bounding boxes (`cat`, 0.8) and (`table`, 0.9) in the same image, than (`bear`, 0.8) and (`table`, 0.9). In particular, for the latter, it is more likely to have (`bear`, 0.8) and (`table`, 0.01) or (`bear`, 0.01) and (`table`, 0.9) instead. We cast such a constraint as an optimization problem.

Contribution

We make three major contributions in this paper. First, we advocate incorporating knowledge into object detection, an emerging paradigm still limited in visual tasks. Second, we formulate a knowledge-aware framework that quantifies semantic consistency based on knowledge graphs in a generalizable manner, and further re-optimizes object detection to achieve better consistency. Last, we conduct extensive experiments on two benchmark datasets, which significantly improves recall by up to 6.3 points while keeping the same level of mean average precision.

2 Related Work

In recent years, deep convolutional neural networks (CNNs) have become the de-facto baseline for computer vision tasks such as image classification and object detection. Their strong performance stems from the ability to learn high-level image features [Krizhevsky *et al.*, 2012; Simonyan and Zisserman, 2014; Szegedy *et al.*, 2015; He *et al.*, 2016]. For object detection, earlier research such as Regions with CNN features (R-CNN) [Girshick *et al.*, 2014] and its fast variant [Girshick, 2015] employs CNNs to classify objects, but depends on precomputed region proposals for object localization. Subsequently, networks such as Overfeat [Sermanet *et al.*, 2013] and Faster R-CNN [Ren *et al.*, 2015] leverages CNNs for not only object classification but also object localization. Faster R-CNN in particular introduces a region proposal network that efficiently shares convolutional features for both region proposal and classification. Furthermore, using contextual information from the entire image has also been explored to improve object detection, by generating a context feature to enhance the classification of individual regions [Bell *et al.*, 2016].

There is also an emerging trend to exploit information outside of images, i.e., external background knowledge such as texts and knowledge graphs, for certain computer vision tasks such as image classification [Deng *et al.*, 2014], visual motivation prediction [Vondrick *et al.*, 2016], visual question answering [Wu *et al.*, 2016] and visual relationship extraction [Lu *et al.*, 2016]. However, to date, using external knowledge has received limited attention for the task of object detection. An early work [Rabinovich *et al.*, 2007] introduces a conditional random field model to maximize the agreement of labels and semantic contexts from their own training data, as well as from an external online service called Google Sets[1] which returned a set of similar concepts based on a few input examples. One recent work [Hong *et al.*, 2017] uses co-occurrence statistics to re-weight the detection scores in indoor scenes. Note that both methods cannot generalize to images with contexts not observed in their training or external data, while our knowledge graph-based approach has a better generalization potential.

Finally, background knowledge can often be organized as a knowledge graph, which is a data structure capable of modeling both real-world concepts and their interactions. The use of knowledge graphs have become widespread and largely successful in many data-driven applications including Web search and social networks [Dong *et al.*, 2014]. Numerous research and commercial efforts have been spent to construct large-scale knowledge graphs [Paulheim, 2017], which often require continuous expansion and refinement. Typically, knowledge graphs are constructed through human curation [Lenat, 1995], crowdsourced contribution [Liu and Singh, 2004], as well as automatic extraction from semi-structured [Suchanek *et al.*, 2007] or unstructured data [Fang and Chang, 2011]. More recently, knowledge has also been systematically harvested from multimodal data including images [Krishna *et al.*, 2017].

[1] The product was discontinued in 2011.

3 Proposed Approach

We describe our knowledge-aware framework in this section, starting with the notations and problem statement, followed by the notion of semantic consistency, as well as the integration of knowledge into object detection.

3.1 Notations and Problem

Consider a set of pre-defined concepts or object labels $\mathcal{L} = \{1, 2, \ldots, L\}$.[2] We assume an existing object detection algorithm that outputs a set of bounding box $\mathcal{B} = \{1, 2, \ldots, B\}$ for each image, and assigns a label $\ell \in \mathcal{L}$ to each bounding box $b \in \mathcal{B}$ with probability $p(\ell|b)$. For each image, these probabilities can be encoded by a $B \times L$ matrix P, such that $P_{b,\ell} = p(\ell|b)$.

Our goal is to produce a new matrix $\widehat{P}$ based on not only the initial matrix P, but also the semantic consistency between concepts which are derived from given knowledge. In other words, $\widehat{P}$ is a knowledge-aware enhancement of P. Ultimately, the new matrix $\widehat{P}$ enables us to improve object detection, such that a bounding box b is assigned a potentially new label $\widehat{\ell} = \arg\max_\ell \widehat{P}_{b,\ell}$. The overall framework is summarized in Fig. 3.

3.2 Semantic Consistency

Knowledge is fundamentally symbolic and logical. However, most state-of-the-art algorithms function on subsymbolic or numerical representations. Thus, towards a knowledge-aware framework, the first step is to quantify such knowledge, especially in a manner that can generalize to images with unobserved contexts. To this end, we propose to measure a numerical degree of *semantic consistency* for each pair of concepts. A high degree of semantic consistency between two concepts implies that the two concepts are likely to appear together in the same image.

Formally, let S be an $L \times L$ matrix such that $S_{\ell,\ell'}$ is defined as the degree of semantic consistency between concepts ℓ and ℓ', $\forall(\ell, \ell') \in \mathcal{L}^2$. Naturally, S shall be symmetric, i.e., $S_{\ell,\ell'} = S_{\ell',\ell}$. Note that, when $\ell = \ell'$, $S_{\ell,\ell'}$ captures the self-consistency, which is meaningful since multiple instances of the same concept can appear in the same image.

In other words, additional background knowledge about various concepts can be quantified and modeled by the matrix S. In the following, we describe two alternatives of constructing S from additional knowledge: one using simple frequency, and the other based on a knowledge graph.

Frequency-based knowledge

To compute semantic consistency, one immediate approach is to utilize the frequency of co-occurrences for each pair of concepts. Such co-occurrences can be identified from given background data, which can be potentially multi-modal including text corpora and photo collections.

Let $n(\ell, \ell')$ denote the frequency of co-occurrences for concepts ℓ and ℓ', and $n(\ell)$ denote the frequency of ℓ. Let N be the total number of instances in the background data. Then, we define semantic consistency below, based on pointwise mutual information. Simply put, when ℓ and ℓ' occur independently, or they co-occur less frequently than if they were to occur independently, the value would be zero; otherwise, the value is positive. In particular, the more likely two concepts co-occur than if they were independent, the more positive the value, bounded by $\log N$ from the above.

$$S_{\ell,\ell'} = \max\left(\log \frac{n(\ell, \ell')N}{n(\ell)n(\ell')}, 0\right) \quad (1)$$

While it is straightforward to compute Eq. (1), there is two major drawbacks. First, collecting enough background data with high-quality annotations is often difficult and expensive, especially when the baseline detection model is given as a blackbox without its accompanying training data. Second, the resulting matrix S only works for known co-occurrences in the background data, but does not generalize to unseen co-occurrences in new images. In other words, if two concepts never co-occur in the background data, their semantic consistency would be exactly zero, and thus are not helpful to new images containing these two concepts.

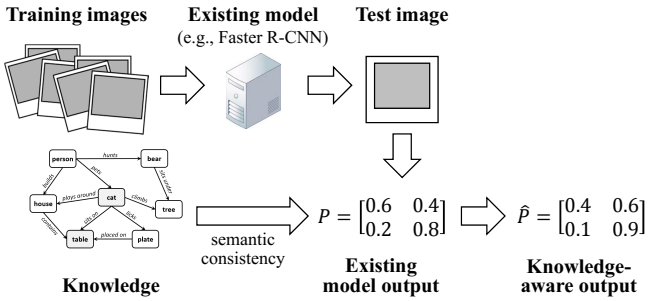


Figure 3: Overview of knowledge-aware framework.

Graph-based knowledge

Next, we consider a knowledge graph for modeling semantic consistency. Unlike the toy example in Figure 2, a typical off-the-shelf knowledge graph often captures at least millions of concepts and their complex relationships, providing immense background knowledge external to the images.

Using a large-scale knowledge graph has a significant advantage—it can better generalize to a pair of concepts even if they are not connected by any edge. In particular, when two concepts are not involved in a direct relationship, potentially we can still establish a chain of relationships between them. For instance, `people` and `plate` in Figure 2 are not directly connected. This does not necessarily mean that they are not semantically consistent. Quite the contrary, they should enjoy a fair degree of semantic consistency based on common sense. Nonetheless, despite a missing edge between them, there is still a chain of edges "`person` pets `cat`" and "`cat` licks `plate`" to indicate that they are semantically consistent to some extent. Furthermore, multiple direct relationships or chains of relationships can exist between two concepts. In Figure 2, `cat` and `table` can be related through the edge "`cat` sits on `table`", and a chain of edges "`cat` licks `plate`" and "`plate` placed on `table`". Each relationship or chain is called a *path* from `cat` to `table`. Different paths between the two concepts complement each other for increased robustness.

[2] In this paper, we use "concept" and "label" interchangeably.

To quantify semantic consistency on a knowledge graph, we employ *random walk with restart* [Tong *et al.*, 2006]. Starting from a node v_0 on the graph, we move to a random neighboring node of v_0, and record it as v_1. Once at v_1, we repeat this process. In general, when we are at v_t, we move to one of its neighbors randomly, and denote the new node we have just arrived as v_{t+1}. In addition, to avoid being trapped in a small locality, at each move, there is a probability of α to restart the random walk by "teleporting" to the starting node v_0, instead of moving to one of the neighbors. Formally, a random walk is a sequence of nodes $\langle v_0, v_1, v_2 \ldots, v_t \rangle$, and $p(v_t = \ell' | v_0 = \ell; \alpha)$ represents the probability of reaching the concept ℓ' in t steps given that we start from ℓ.

This probability can be used to formulate semantic consistency, such that a larger probability from ℓ to ℓ' implies that they are more semantically consistent. Intuitively, when the number of paths from ℓ to ℓ' increases or the length of these paths decreases, the semantic consistency between ℓ and ℓ' becomes larger, so does the probability of reaching ℓ' from ℓ. Interestingly, as we take longer random walks, this probability eventually converges to a unique steady state as follows.

$$R_{\ell,\ell'} = \lim_{t \to \infty} p(r_t = \ell' | r_0 = \ell; \alpha). \tag{2}$$

Note that $R_{\ell,\ell'}$ is not symmetric in general. Thus, in Eq. (3) we define a symmetric matrix S based on the geometric mean. The geometric mean has a roundtrip random walk interpretation, and has been shown to be superior than the arithmetic or harmonic means [Fang *et al.*, 2013]. The matrix S can be efficiently computed even on a very large knowledge graph [Zhu *et al.*, 2013].

$$S_{\ell,\ell'} = S_{\ell',\ell} = \sqrt{R_{\ell,\ell'} R_{\ell',\ell}} \tag{3}$$

One caveat is the huge effort required to build and refine a large-scale knowledge graph, which is an active research area itself. Fortunately, a suite of off-the-shelf solutions are available, many of which offer open datasets or APIs. For a thorough discussion on this matter, we refer the reader to a survey paper [Paulheim, 2017] and the citations therein. In our experiments, we adopt MIT ConceptNet [Liu and Singh, 2004], a crowdsourced knowledge graph with more than 4 million concepts and 9 million relationships.

3.3 Knowledge-Aware Re-optimization

Given a matrix that quantifies the semantic consistency between pairwise concepts, we need to further integrate it with an existing model to enable knowledge-aware detection through a re-optimization process. In the following, we formulate a cost function based on semantic consistency, and further discuss its efficient optimization.

Cost function

The key intuition is that two concepts with a higher degree of semantic consistency are more likely to appear in the same image with comparable probability. That is, for two different bounding boxes b and b' in one image, $P_{b,\ell}$ and $P_{b',\ell'}$ should not be too different when $S_{\ell,\ell'}$ is large. This constraint can be formalized by minimizing the cost function in Eq. (4), where $\{P_{b,\ell} : b \in \mathcal{B}, \ell \in \mathcal{L}\}$ represent the detections from any existing algorithm, and $\{\widehat{P}_{b,\ell} : b \in \mathcal{B}, \ell \in \mathcal{L}\}$ represent our proposed knowledge-aware detections.

$$E(\widehat{P}) = (1-\epsilon) \sum_{b=1}^{B} \sum_{\substack{b'=1 \\ b' \neq b}}^{B} \sum_{\ell=1}^{L} \sum_{\ell'=1}^{L} S_{\ell,\ell'} \left(\widehat{P}_{b,\ell} - \widehat{P}_{b',\ell'} \right)^2 + \epsilon \sum_{b=1}^{B} \sum_{\ell=1}^{L} B \|S_{\ell,*}\|_1 \left(\widehat{P}_{b,\ell} - P_{b,\ell} \right)^2 \tag{4}$$

On the one hand, the first term of Eq. (4) captures the constraint on the semantic consistency. For a pair of detected bounding boxes b and b', if $S_{\ell,\ell'}$ is large, minimizing the objective function would force $P_{b,\ell}$ and $P_{b',\ell'}$ to become smaller; if $S_{\ell,\ell'}$ is small, $P_{b,\ell}$ and $P_{b',\ell'}$ are less constrained and can become very different.

On the other hand, the second term requires that knowledge-aware detections should not depart too much from detections of existing algorithms. Existing algorithms use features specific to each image which form the basis of our knowledge-aware approach. Note that the squared error has a coefficient $B\|S_{\ell,*}\|_1$ in order to balance different concepts. Without this coefficient, the cost function would give more importance to the first term over summations involving $P_{b,\ell}, \forall b \in \mathcal{B}$ when $\|S_{\ell,*}\|$ is larger. The overall trade-off between the two terms is controlled by a hyperparameter $\epsilon \in (0, 1)$, which can be selected on a validation set.

Optimization

To minimize Eq. (4), we find its stationary point where its gradient w.r.t. $\widehat{P}_{b,\ell}$ is zero, $\forall b \in \mathcal{B}, \ell \in \mathcal{L}$.

$$\frac{\partial E(\widehat{P})}{\partial \widehat{P}_{b,\ell}} \propto (1-\epsilon) \sum_{\substack{b'=1 \\ b' \neq b}}^{B} \sum_{\ell'=1}^{L} S_{\ell,\ell'} \left(\widehat{P}_{b,\ell} - \widehat{P}_{b',\ell'} \right) + \epsilon B \|S_{\ell,*}\|_1 \left(\widehat{P}_{b,\ell} - P_{b,\ell} \right) \tag{5}$$

Setting the above to zero, we obtain below an equivalent configuration over optimal $\widehat{P}_{b,\ell}$.

$$\widehat{P}_{b,\ell} = (1-\epsilon) \frac{\sum_{b'=1, b' \neq b}^{B} \sum_{\ell'=1}^{L} S_{\ell,\ell'} \widehat{P}_{b',\ell'}}{\sum_{b'=1, b' \neq b}^{B} \sum_{\ell'=1}^{L} S_{\ell,\ell'}} + \epsilon P_{b,\ell} \tag{6}$$

It can be shown that the exact solution to Eq. (6) is the limit of the series in Eq. (7) for $i \in \{1, 2, \ldots\}$. In particular, for any arbitrary initialization $\widehat{P}_{b,\ell}^{(0)}$, $\widehat{P}_{b,\ell}^{(i)}$ always converges to the same solution as $i \to \infty$.

$$\widehat{P}_{b,\ell}^{(i)} = (1-\epsilon) \frac{\sum_{b'=1, b' \neq b}^{B} \sum_{\ell'=1}^{L} S_{\ell,\ell'} \widehat{P}_{b',\ell'}^{(i-1)}}{\sum_{b'=1, b' \neq b}^{B} \sum_{\ell'=1}^{L} S_{\ell,\ell'}} + \epsilon P_{b,\ell} \tag{7}$$

Note that the solution can be computed in polynomial time. The theoretical complexity is $O(B^2 L^2 I)$, where I is the number of iterations. Convergence typically happens very fast in fewer than 30 iterations. To further speed up the computation, we could apply an approximation using B_k nearest bounding boxes and L_k nearest concepts. That is, a pair of bounding boxes b and b' are considered only if either of them is among

the B_k bounding boxes with smallest distances to the other; a pair of concepts ℓ and ℓ' are considered only if either of them is among the L_k labels with largest semantic consistency to the other. Thus, the practical complexity is only $O(BL)$, assuming that I, B_k, L_k are small constants.

4 Evaluation

We empirically evaluate the proposed approach on two benchmark datasets. Results of our knowledge-aware detection is promising, significantly outperforming the baseline method in recall while maintaining the same level of mean average precision.

4.1 Experimental setup

Datasets

We use benchmark data MSCOCO15 [Lin *et al.*, 2014] and PASCAL07 [Everingham *et al.*, 2010], summarized in Table 1. For MSCOCO15, we combine their training and validation sets for training the baseline, except for a subset of 5000 images named "minival". We further split minival into 1000 and 4000 images, named "minival-1k" and "minival-4k" respectively. We use minival-1k to choose hyperparameter for our approach, and minival-4k for offline testing. Online evaluation on the MSCOCO15 server[3] is performed on the test set, since its ground truth is not publicly available. The test set contains two subsets of roughly equal size, namely "test-dev" and "test-std", where the latter only allows for limited submissions. For PASCAL07, we use their training set for training the baseline, validation set for choosing our hyperparameter, and test set for evaluation.

Model training

We employ the state-of-the-art Faster R-CNN and VGG-16 as the baseline [Simonyan and Zisserman, 2014; Ren *et al.*, 2015], using the public Python Caffe implementation[4]. We call this baseline **FRCNN** hereafter. Models are trained using stochastic gradient descent with a momentum of 0.9, a mini-batch size of 2 and a weight decay of 5e-4. Layer weights are initialized from a VGG-16 model pre-trained on ImageNet. New layers defined by Faster R-CNN are randomly initialized from a Gaussian distribution with a standard deviation of 0.01. We use a learning rate of 1e-3 for the first 350K/50K iterations on MSCOCO15/PASCAL07, followed by 1e-4 for another 140K/10K iterations.

For our knowledge-aware approach, we re-optimize the output of FRCNN. We only retain top 500 bounding boxes whose scores are at least 1e-5. On the validation data, we choose the hyperparameter ϵ in Eq. (4) from $\{0.1, 0.25, 0.5, 0.75, 0.9\}$. To speed up the computation of Eq. (7), we only consider 5 nearest neighbors for both the bounding boxes and labels as an approximation. The updates are performed for 10 iterations, which already show convergence. We compare several variants of our approach.

On the one hand, we adopt frequency-based knowledge, combining the training sets of both benchmarks as the background data. We name this variant **KF-All**. Furthermore, to demonstrate that the accuracy of the results depend on the quality of the background data, we consider a second variant named **KF-500**, by sampling only 500 images from the training sets as the background data.

On the other hand, for the graph-based knowledge, we employ MIT ConceptNet 5[5] as our knowledge graph. We only use its English subgraph, and filter out "negative" relationships (NotDesires, NotHasProperty, NotCapableOf, NotUsedFor, Antonym, DistinctFrom and ObstructedBy) and self-loops. The resulting graph has 1.3 million concepts and 2.8 million relationships. We set the random walk restarting probability $\alpha = 0.15$, a typical value known to be stable [Fang *et al.*, 2013]. We call this variant **KG-CNet**.

Dataset	# Concepts	# Images		
		training	validation	test
MSCOCO15	80	80K	40K	40K
PASCAL07	20	2.5K	2.5K	5.0K

Table 1: Summary statistics of benchmark datasets.

	mAP	Recall		Recall@100 by area		
	@100	@100	@10	small	medium	large
minival-4k						
FRCNN	24.5	35.9	35.2	14.2	41.5	55.6
KF-500	24.4	37.1	35.6	14.3	42.8	57.3
KF-All	24.5	37.9	36.2	**14.6**	43.9	58.6
KG-CNet	24.4	**38.9**	**36.6**	14.4	**45.2**	**60.0**
test-dev						
FRCNN	24.2	34.6	34.0	12.0	38.5	54.4
KF-500	24.3	37.4	35.9	13.7	42.1	58.0
KF-All	24.3	38.2	36.4	14.2	43.0	59.2
KG-CNet	24.2	**39.2**	**36.9**	**14.5**	**44.0**	**60.7**
test-std						
FRCNN	24.2	34.7	34.1	11.5	38.9	54.4
KG-CNet	24.1	**39.2**	**37.0**	**14.2**	**44.4**	**60.5**

Table 2: Comparison of our knowledge-aware variants with the baseline method on MSCOCO15.

Accuracy metrics

The main metrics are mean average precision (mAP) and recall at top 100. On MSCOCO15, we also report recall at top 10 and by object areas (small, medium and large); on PASCAL07, we further report recall by concepts. In particular, a bounding box is judged correct only if its intersection over union (IoU) w.r.t. the ground truth is above some threshold. We use the IoU threshold as standardized in each benchmark: On MSCOCO15, it is varied over $\{0.50, 0.55, \ldots, 0.95\}$ and their average results are reported; for PASCAL07, it is fixed at 0.5.

4.2 Main results

We report the results on MSCOCO15 in Table 2. Both KF-All and KG-CNet significantly increase recall@100 over the baseline method FRCNN by up to 3.6 and 4.6 points, respectively.

[3] http://mscoco.org/home/

[4] https://github.com/rbgirshick/py-faster-rcnn

[5] http://conceptnet-api-1.media.mit.edu/

	mAP @100	**Recall**@100 by concepts																				
		all	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
FRCNN	66.5	81.9	76.1	89.0	74.3	73.4	64.6	89.7	85.8	90.5	69.0	88.9	85.4	91.6	92.0	85.2	82.4	60.8	83.1	89.1	84.4	82.1
KF-500	66.6	83.8	80.0	91.7	**79.1**	**76.0**	67.0	89.7	88.8	92.5	69.7	92.6	85.9	90.8	**94.0**	86.8	82.0	59.6	87.2	90.0	89.7	82.8
KF-All	66.5	84.6	**80.7**	**93.5**	**79.1**	**76.0**	**67.6**	**90.1**	88.8	**93.6**	68.1	**93.0**	**86.9**	**94.1**	93.1	**89.5**	83.1	65.4	**88.0**	89.1	**90.1**	81.8
KG-CNet	66.6	**85.0**	80.4	92.3	78.6	**76.0**	**67.6**	**90.1**	**89.1**	92.2	**74.2**	**93.0**	86.4	93.0	92.2	88.6	**87.7**	**66.9**	87.6	**90.4**	89.7	**83.4**

Table 3: Comparison of our knowledge-aware variants with the baseline method on PASCAL07.

Other recall metrics, at top 10 and by areas, also show significant improvement up to 4.8 and 6.3 points, respectively. At the same time, both approaches do not compromise mAP. Moreover, comparing with KF-All, KF-500 attains smaller improvements across all recall metrics, which is not surprising given fewer background data.

Next, we present the results on PASCAL07 in Table 3. Likewise, both KF-All and KG-CNet beat FRCNN in recall@100 by 2.7 and 3.1 points, respectively, without affecting mAP. In particular, KF-All outperforms the baseline in 17 out of the 20 concepts, whereas KG-CNet outperforms the baseline in all 20 concepts. As usual, KF-500 shows smaller improvements than KF-All in most cases.

Note that KG-CNet generates consistently better results than KF-All on MSCOCO15, but to a much lesser extent on PASCAL07. We hypothesize that the discrepancies are caused by the complexity of the benchmarks. In particular, MSCOCO15 are more complex than PASCAL07 [Lin *et al.*, 2014]: The former contains an average of 3.5 concepts and 7.7 instances per image, whereas the latter has fewer than 2 concepts and 3 instances per image. The simpler scenes in PASCAL07 would thus require less generalization, and the frequency-based variant could benefit from this situation.

We also observe that both KF-All and KG-CNet deliver more significant improvements on MSCOCO15 than on PASCAL07. We believe that the underlying reason is similar in that the knowledge-aware variants are able to benefit more from the semantically richer scenes in MSCOCO15.

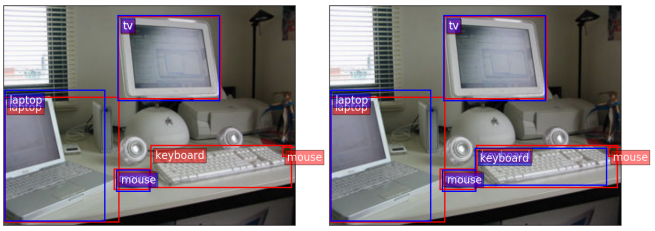
4.3 Case study

Finally, we showcase the ability of the knowledge graph-based variant in detecting additional objects and thus improving recall, on real images from MSCOCO15.

The example in Figure 4a depicts an office scene, containing ground truth objects `keyboard` and `laptop`, among others. Although the baseline misses the `keyboard`, it is picked up by KG-CNet after re-optimization. The reason is that the probability of `keyboard` is promoted given the presence of `laptop`, since the two concepts share very high semantic consistency (135 times of the median value among all pairwise concepts). Of course, other equipment like `mouse` may also have contributed.

Another example in Figure 4b depicts an outdoor scene with ground truth objects `person` and `surfboard`. Likewise, the baseline fails to detect `surfboard`, but KG-CNet identifies it correctly. In particular, the two concepts are also semantically consistent (5 times of the median value).

(a) Office scene: FRCNN (left) fails to detect `keyboard`, but KG-CNet (right) does due to the presence of `laptop`.



(b) Outdoor scene: FRCNN (left) fails to detect `surfboard`, but KG-CNet (right) does due to the presence of `person`.

Figure 4: Two scenes from MSCOCO15 (best viewed in color). In each scene, the *left* image contains the output of the baseline method FRCNN, whereas the *right* image contains the output of our proposed KG-CNet. Ground-truth objects are marked with orange boxes, and correct detections of IoU at least 0.75 in top 100 are marked with blue boxes.

5 Conclusion

In this paper, we study the problem of object detection in a novel knowledge-aware framework. Compared to existing algorithms which only focus on features within an image, we propose to leverage external knowledge such as knowledge graphs. Towards this goal, we derive and quantify semantic consistency from knowledge graphs that can generalize to new images with unobserved contexts. Next, we integrate knowledge into existing object detection algorithms, by re-optimizing the detections to attain better semantic consistency. Finally, we demonstrate the superior performance of our proposed approach through extensive experiments on two benchmark datasets. As future work, we plan to explore or construct knowledge graphs that are specifically tailored to visual tasks, instead of using a general-purpose knowledge graph without emphasizing on visual relationships.

References

[Bell *et al.*, 2016] Sean Bell, C. Lawrence Zitnick, Kavita Bala, and Ross B. Girshick. Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks. In *CVPR*, pages 2874–2883, 2016.

[Deng *et al.*, 2014] Jia Deng, Nan Ding, Yangqing Jia, Andrea Frome, Kevin Murphy, Samy Bengio, Yuan Li, Hartmut Neven, and Hartwig Adam. Large-scale object classification using label relation graphs. In *ECCV, Part I*, pages 48–64, 2014.

[Dong *et al.*, 2014] Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. Knowledge vault: a Web-scale approach to probabilistic knowledge fusion. In *KDD*, pages 601–610, 2014.

[Everingham *et al.*, 2010] Mark Everingham, Luc J. Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The PASCAL visual object classes (VOC) challenge. *IJCV*, 88(2):303–338, 2010.

[Fang and Chang, 2011] Yuan Fang and Kevin Chen-Chuan Chang. Searching patterns for relation extraction over the web: rediscovering the pattern-relation duality. In *WSDM*, pages 825–834, 2011.

[Fang *et al.*, 2013] Yuan Fang, Kevin Chen-Chuan Chang, and Hady Wirawan Lauw. RoundTripRank: Graph-based proximity with importance and specificity. In *ICDE*, pages 613–624, 2013.

[Girshick *et al.*, 2014] Ross B. Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, pages 580–587, 2014.

[Girshick, 2015] Ross B. Girshick. Fast R-CNN. In *ICCV*, pages 1440–1448, 2015.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[Hong *et al.*, 2017] Jongkwang Hong, Yongwon Hong, Youngjung Uh, and Hyeran Byun. Discovering overlooked objects: Context-based boosting of object detection in indoor scenes. *Pattern Recogn. Lett.*, 86:56–61, 2017.

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

[Krizhevsky *et al.*, 2012] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, pages 1106–1114, 2012.

[Lenat, 1995] Douglas B. Lenat. CYC: A large-scale investment in knowledge infrastructure. *Commun. ACM*, 38(11):32–38, 1995.

[Lin *et al.*, 2014] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *ECCV, Part V*, pages 740–755, 2014.

[Liu and Singh, 2004] Hugo Liu and Push Singh. ConceptNet—a practical commonsense reasoning tool-kit. *BT Technology Journal*, 22(4):211–226, 2004.

[Lu *et al.*, 2016] Cewu Lu, Ranjay Krishna, Michael S. Bernstein, and Fei-Fei Li. Visual relationship detection with language priors. In *ECCV, Part I*, pages 852–869, 2016.

[Paulheim, 2017] Heiko Paulheim. Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic Web*, 8(3):489–508, 2017.

[Rabinovich *et al.*, 2007] Andrew Rabinovich, Andrea Vedaldi, Carolina Galleguillos, Eric Wiewiora, and Serge J. Belongie. Objects in context. In *ICCV*, pages 1–8, 2007.

[Ren *et al.*, 2015] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *NIPS*, pages 91–99, 2015.

[Sermanet *et al.*, 2013] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *CoRR*, arXiv:1312.6229, 2013.

[Simonyan and Zisserman, 2014] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, arXiv:1409.1556, 2014.

[Suchanek *et al.*, 2007] Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. YAGO: a core of semantic knowledge. In *WWW*, pages 697–706, 2007.

[Szegedy *et al.*, 2015] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, pages 1–9, 2015.

[Tong *et al.*, 2006] Hanghang Tong, Christos Faloutsos, and Jia-Yu Pan. Fast random walk with restart and its applications. In *ICDM*, pages 613–622, 2006.

[Vondrick *et al.*, 2016] Carl Vondrick, Deniz Oktay, Hamed Pirsiavash, and Antonio Torralba. Predicting motivations of actions by leveraging text. In *CVPR*, pages 2997–3005, 2016.

[Wu *et al.*, 2016] Qi Wu, Peng Wang, Chunhua Shen, Anthony R. Dick, and Anton van den Hengel. Ask me anything: Free-form visual question answering based on knowledge from external sources. In *CVPR*, pages 4622–4630, 2016.

[Zhu *et al.*, 2013] Fanwei Zhu, Yuan Fang, Kevin Chen-Chuan Chang, and Jing Ying. Incremental and accuracy-aware personalized pagerank through scheduled approximation. *PVLDB*, 6(6):481–492, 2013.