Learning Unsupervised Visual Grounding Through Semantic Self-Supervision

Syed Ashar Javed
Robotics Institute
Carnegie Mellon University

Shreyas Saxena

Vineet Gandhi
Centre for Visual Information Technology
IIIT Hyderabad

Abstract

Localizing natural language phrases in images is a challenging problem that requires joint understanding of both the textual and visual modalities. In the unsupervised setting, lack of supervisory signals exacerbate this difficulty. In this paper, we propose a novel framework for unsupervised visual grounding which uses concept learning as a proxy task to obtain self-supervision. The simple intuition behind our idea is to encourage the model to localize to regions which can explain some semantic property in the data, in our case, the property being the presence of a concept in a set of images. We present quantitative and qualitative experiments to demonstrate the efficacy of our approach and show a 5.6% improvement over the state of the art on Visual Genome dataset, a 5.8% improvement on the ReferItGame dataset and comparable to state-of-art performance on the Flickr30k dataset.

1 Introduction

Utilizing arbitrary length phrases for visual grounding overcomes the limitation of using a restricted set of categories for localization and provides a more detailed description of the region of interest as compared to single-word nouns or attributes. The problem of supervised visual grounding (i.e localizing) of phrases has hence gathered a lot of interest in recent vision literature [4, 7, 2]. However, supervised approaches require expensive bounding box or pixel level annotations which are difficult to scale. In this paper we address the problem of visual grounding of textual phrases with an unsupervised approach where no bounding box annotation exists during training. Given the lack of supervision, we develop a self-supervised proxy task which can be used to guide the learning. The general idea behind self-supervision is to design a pretext task which involves explaining some regularity about the input data. Instead of directly optimizing the localization objective, the model is trained with a surrogate loss which tries to optimize for the proxy task. A good proxy improves performance on the final task when the surrogate loss is minimized. We propose concept-learning as a substitute task for visual grounding. During training, we create *concept batches* of size k, consisting of k different phrase-image pairs, all containing a common concept. We exploit this presence of semantic commonality within the concept batch to generate supervisory signals. We hypothesize that to predict these commonalities, the model must localize them correctly within each image of the set. We induce a parametrization in the form of attention which, given the input text and image, can localize the phrase. These localized regions are then used to predict the common concept. The entity to be grounded in most phrases is a single-word occurring in varying contexts and learning high-level semantic representations for these concept can improve visual grounding. In summary, the main contributions of our work are as follows:

- We propose a novel framework for visual grounding of phrases through semantic self-supervision where the proxy task is formulated as concept learning. We introduce the idea of a concept batch to aid learning.

32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.

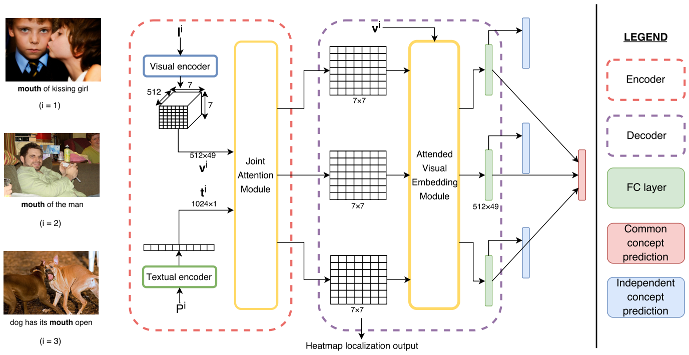

Figure 1: An overview of our model. The encoder takes in a set of image-phrase pairs, indexed by i, all sharing a common concept and embeds them to $\mathbf{V}^i$ and $\mathbf{t}^i$ respectively. These features are used to induce a parametrization for spatial attention. Next, the decoder uses the visual attention map to predict the common concept. In addition, the decoder also predicts the common concept independently for each pair (i). For details, see Section 2.

- We evaluate our approach on the Visual Genome and ReferIt dataset and achieve state-of-art performance with a gain of 5.6% and 5.8% respectively. We also get performance comparable to the state-of-art on Flickr30k dataset.
- We analyze the behavior of our surrogate loss and the concept batch which gives an insight into the functioning of our approach. We also analyze the correlation of our model's performance with factors like bounding box size and concept similarity.

2 Grounding through semantic self-supervision

Proxy task formulation. Our model is trained for the proxy task of concept-learning. A concept is defined as the entity which is to be grounded in the image. For example, in the phrase *'white **towel** on the counter'*, the highlighted word *'towel'* is the concept. We hypothesize that if we induce a parametrization for localization of the phrase and use the localized regions to predict the concept present in an image, the parametrization will converge to the ground truth localization of the phrase. Given this proxy task, we're faced with two main challenges: 1) How do we identify the concept in a phrase? and 2) How do we learn concept representations in an unsupervised setting? For the first part, we assume the concept is a single-word entity and exists within the phrase, and naively pick a random noun from the phrase. For the second problem, we introduce the notion of a concept batch. A concept batch is one training instance for our model, which itself consists of k phrase-image pairs, all containing a common concept. The proxy task is now formulated as jointly decoding the common concept using all k localized feature representations in addition to independently decoding the same concept. The intuition behind training with a concept batch is that for decoding the common concept, k phrase-image pairs should encode a localized representation which is invariant to the difference in context across the k pairs. On the other hand, the proxy task of decoding independent concept (for each image in the batch) ensures two things: a) Individual and common representations are consistent b) Model cannot find a shortcut by using only few inputs from the concept batch to decode the common concept (see section 4). Note that using a concept batch with random sampling of nouns can be interpreted as generating weak supervision, albeit noisy in nature. Since the same image-phrase pair can be chosen with different sampled concepts during training, it is this random sampling of nouns which ensures that the model doesn't only learn a simple concept-identifier, but also generates discriminative information about a concept in different contexts.

Encoder-Decoder model. We adopt an encoder-decoder architecture for learning to ground as illustrated in Figure 1. The encoder uses an attention mechanism similar to [10] using the joint features from visual and textual modalities. Similar to previous work, we use a pre-trained VGG16 and a language model [1] and freeze their weights during training. For the i^{th} index in the concept batch, given visual features $\mathbf{V}^i = f_{VGG}(I^i)$ and textual features $\mathbf{t}^i = f_{LM}(P^i)$, the attention over visual regions is given by:

$$\mathbf{f}^i_{attn} = softmax(\mathbf{f}_{joint}(\mathbf{V^i}, \mathbf{t^i})). \tag{1}$$

$$\mathbf{f}_{joint}(\mathbf{V}^i, \mathbf{t}^i) = \Phi_s(\Phi_r(\Phi_q(\Phi_p([\mathbf{V}^i, \mathbf{t}^i])))), \tag{2}$$

Method	**Accuracy**			
	Visual Genome	ReferIt (mask)	ReferIt (bbox)	Flickr30k
Random baseline	11.15	16.48	24.30	27.24
Center baseline	20.55	17.04	30.40	49.20
VGG baseline	18.04	15.64	29.88	35.37
Fang *et al.* [3]	14.03	23.93	33.52	29.03
Zhang *et al.* [11]	19.31	21.94	31.97	42.40
Ramanishka *et al.* [8]	-	-	-	**50.10**
Xiao *et al.* [9]	24.40	-	-	-
Semantic self-supervision	**30.03**	**29.72**	**39.98**	49.10

Loss Type	**Concept batch Size (k)**			
	$k = 3$	$k = 5$	$k = 7$	$k = 9$
Independent concept only	27.15	27.27	28.01	28.05
Common concept only	27.52	28.94	29.18	27.90
Independent and common concept	28.25	28.91	29.89	30.03

Table 1: Grounding evaluation using the pointing game (left) and the ablation analysis of different surrogate losses while varying the concept batch size (right).

where $\mathbf{V}^i \in \mathbb{R}^{m\times n}$, $\mathbf{t}^i \in \mathbb{R}^{l\times 1}$, $\mathbf{f}_{joint}(\mathbf{V}^i, \mathbf{t}^i) \in \mathbb{R}^{1\times n}$, $[\mathbf{V}^i, \mathbf{t}^i]$ is an index-wise concatenation operator (over the first dimension) between a matrix $\mathbf{V}^i$ and a vector $\mathbf{t}^i$ resulting in a matrix of size $((m+l)\times n)$. $\Phi(\cdot)$ corresponds to a hidden layer of a neural network and is defined as:

$$\Phi_p(\mathbf{X}) = ReLU(\mathbf{W}_p\mathbf{X} + \mathbf{b}_p), \tag{3}$$

where $ReLU(x) = max(x, 0)$, $\mathbf{W}_p \in \mathbb{R}^{p\times d}$, $\mathbf{b}_p \in \mathbb{R}^{p\times 1}$ and $\mathbf{X} \in \mathbb{R}^{d\times n}$. Here n is the number of regions over which attention is defined and d is the dimensionality of each region with respective to $\mathbf{X}$. Thus, we use a 4 layered non linear perceptron to calculate attention for each of the n regions (Since our attention is over VGG16 features, n = 7 $\times$ 7). The four $\Phi(\cdot)$ layers gradually decrease the dimensionality of the concatenated joint features from $(m+l) \to p \to q \to r \to s$ where $s = 1$. Note that the attention module is shared across all $\mathbf{V}^i$ and $\mathbf{t}^i$ and the encoder is common for all pairs in the concept batch. Given the attention weights $\mathbf{f}^i_{attn} \in \mathbb{R}^{1\times n}$, the visual attention for common concept prediction ($\mathbf{f}_{vac}$) is computed by taking the weighted sum with the original visual features.

$$\mathbf{f}_{vac} = \sum_{i=1}^{k} \mathbf{f}^i_{attn}\mathbf{V}^i \tag{4}$$

We find that aggregating the visual attention across regions, which is commonly done in the past attention literature degrades performance for our task. Therefore we retain the spatial information and only aggregate the features across the concept batch. Similarly, the visual attention for independent concept prediction, $\mathbf{f}^i_{vai}$ is given by the element-wise product of the attention weights and visual features.

$$\mathbf{f}^i_{vai} = \mathbf{f}^i_{attn}\mathbf{V}^i \tag{5}$$

Finally, both the attended features are flattened and separately connected to a fully connected layer, leading to a softmax over the concepts. In practice, we also down-sample the dimensionality of $\mathbf{f}^i_{vai}$ using $1 * 1$ convolutions before we aggregate and flatten the features.

$$\mathbf{y}_{common} = softmax(\mathbf{W}_{vac}\mathbf{f}_{vac} + \mathbf{b}_{vac}). \tag{6}$$

$$\mathbf{y}^i_{independent} = softmax(\mathbf{W}_{vai}\mathbf{f}^i_{vai} + \mathbf{b}_{vai}), \tag{7}$$

where $\mathbf{y}_{common}$ is the network prediction for the common concept and $\mathbf{y}^i_{independent}$ is the independent concept prediction for the i^{th} index in the concept batch. Our surrogate loss is the sum of common concept cross-entropy loss and the concept-batch averaged independent concept cross-entropy loss.

3 Experimental setup and evaluation

An ImageNet trained VGG16 and a Google 1 Billion trained language model are used for encoding the image and the phrase respectively. Before the attention module, both the features are normalized using a batch-normalization layer [5]. The concept vocabulary used for training with the softmax loss is taken from the most frequently occurring nouns. Since the frequency distribution follows the Zipf's Law, around 95% of the phrases are accounted for by top 2000 concepts, which is used as the softmax size. To handle imbalance in concept sampling, we create mini batches by uniformly sampling from the concept vocabulary instead of image-phrase samples during training. In the encoder, the values of p, q, r, s from Equation 2 are taken as $512, 128, 32, 1$ respectively. We test our model on three diverse datasets in terms of their phrase and image statistics, namely Visual Genome, Flickr30k Entities and Refer-It. Since our model generates localization in the form of a heatmap, we evaluate with the pointing game metric [11]. Out of the 49 regions in the attention map, we choose the center of the grid which is maximally activated, as the maximum pixel for the pointing game. We also test on three naive, but revealing baselines. The first is the random baseline which randomly picks one of the 49 grids. The second is the center baseline which always picks the center-most point of the image as the maximum pixel. The third is the VGG baseline which picks the maximum grid from the 7×7 feature map (which is the input to our model) of a pre-trained VGG16, hence acting like a visual-only baseline.

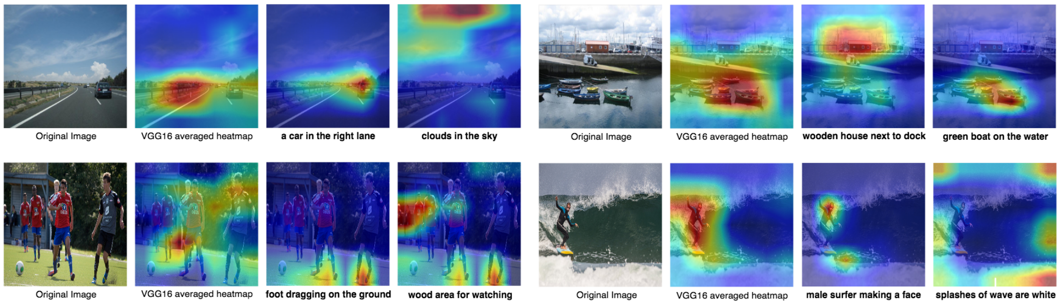

Figure 2: Qualitative results of our approach and its comparison with channel-averaged VGG16 features.

4 Results and analysis

Results. We report the comparison of our method with the baselines and previous methods in this section. Table 1 summarizes the performance of our best model on the three different datasets. To highlight our generalization, we train our model on Visual Genome since it's the largest dataset out of the three and directly evaluate on the other two without fine tuning. Surprisingly, the VGG16 baseline fares decently well even though it does not take any phrase-related information into account owing to the fact that many phrases refer to commonly occurring objects. The center baseline performs very well for Fickr30k showing the centered nature of the phrases with respect to spatial location. Our model outperforms all baselines on Visual Genome and Refer-It and is just 1% less than the state-of-art work of [8] on Flickr30k.

Concept batch size and surrogate loss. We perform ablative studies on the two loss terms and the concept batch size k. We use the shorthand IC (independent concept only), CC (common concept only) and ICC (independent and common concept) for the three loss types from Table 1. We train our model with the IC and CC loss separately, keeping everything else in the pipeline fixed. For all three settings, we vary the concept batch size k and observe some interesting trends. For a fixed loss type, the performance increases as we increase k, the CC loss being the exception to this trend. The performance for CC loss increases up to $k = 7$, but goes down with $k = 9$. This points to a very common problem with self-supervised techniques where the model finds a shortcut to reduce the loss value without improving on the performance. With only the common concept loss, the network can learn a majority voting mechanism such that not all k concept representations are consistent with the common concept and the loss can be optimized even though some instances aren't grounded correctly. This is corroborated with the fact that we also observe a faster convergence of CC loss for $k = 9$ than the lower concept batch sizes. These results empirically highlight the importance of the IC loss term as a regularizer and also highlight the usefulness of our concept batch formulation since it improves performance.

Performance variation across concepts. To better understand the variation in performance across the chosen concepts, we also compute the performance across each of the 2000 concept classes and observe variation in performance. We investigate two possible causes for this variability. The first is the average bounding box size associated with each of these concepts for which we expectedly find a strong positive correlation ($\rho = 0.85$), explaining the lower performance for concepts like *'screw'* and *'doorknob'*. The second is the existing knowledge of concept labels present in the ImageNet classes which our model obtains through the VGG16 based visual encoder. For computing the correlation of concept performance with the knowledge from ImageNet classes, we use a trained word2vec model [6] and compute the maximum similarity of a particular concept across all the ImageNet classes. We find no noticeable correlation between the ImageNet similarity of our concepts and their performance ($\rho = -0.02$). This further strengthens the case for our approach since our concept performance isn't biased towards the ImageNet labels.

Improvement over a noun-based concept detector. We also conduct a simple experiment to verify that the model isn't simply working as a noun-based concept detector instead of modeling the complete phrase. For this, we replace the full phrase with a single noun, randomly sampled from the phrase, as the input to the textual encoder. We note a 4.7% drop in performance on Visual Genome. Since training of the original model enforces only concept-level discrimination, it's interesting to see the presence of complete phrases being useful for model performance and shows that our model learns more than just word-level concepts.

References

[1] Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. One billion word benchmark for measuring progress in statistical language modeling. *arXiv preprint*, 2013.

[2] Kan Chen, Rama Kovvuri, and Ram Nevatia. Query-guided regression network with context policy for phrase grounding. In *ICCV*, 2017.

[3] Hao Fang, Saurabh Gupta, Forrest Iandola, Rupesh K Srivastava, Li Deng, Piotr Dollár, Jianfeng Gao, Xiaodong He, Margaret Mitchell, John C Platt, et al. From captions to visual concepts and back. In *CVPR*, 2015.

[4] Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding. *arXiv preprint*, 2016.

[5] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

[6] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013.

[7] Bryan A Plummer, Arun Mallya, Christopher M Cervantes, Julia Hockenmaier, and Svetlana Lazebnik. Phrase localization and visual relationship detection with comprehensive image-language cues. In *CVPR*, 2017.

[8] Vasili Ramanishka, Abir Das, Jianming Zhang, and Kate Saenko. Top-down visual saliency guided by captions. In *CVPR*, 2017.

[9] Fanyi Xiao, Leonid Sigal, and Yong Jae Lee. Weakly-supervised visual grounding of phrases with linguistic structures. In *CVPR*, 2017.

[10] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, 2015.

[11] Jianming Zhang, Zhe Lin, Jonathan Brandt, Xiaohui Shen, and Stan Sclaroff. Top-down neural attention by excitation backprop. In *ECCV*, 2016.