Learning Deep Structure-Preserving Image-Text Embeddings

Liwei Wang* Yin Li† Svetlana Lazebnik*
lwang97@illinois.edu yli440@gatech.edu slazebni@illinois.edu
*University of Illinois at Urbana-Champaign †Georgia Institute of Technology

Abstract

This paper proposes a method for learning joint embeddings of images and text using a two-branch neural network with multiple layers of linear projections followed by nonlinearities. The network is trained using a large-margin objective that combines cross-view ranking constraints with within-view neighborhood structure preservation constraints inspired by metric learning literature. Extensive experiments show that our approach gains significant improvements in accuracy for image-to-text and text-to-image retrieval. Our method achieves new state-of-the-art results on the Flickr30K and MSCOCO image-sentence datasets and shows promise on the new task of phrase localization on the Flickr30K Entities dataset.

1. Introduction

Computer vision is moving from predicting discrete, categorical labels to generating rich descriptions of visual data, for example, in the form of natural language. There is a surge of interest in image-text tasks such as image captioning [10, 22, 23, 25, 31, 43, 46, 50] and visual question answering [2, 12, 52]. A core problem for these applications is how to measure the semantic similarity between visual data (e.g., an input image or region) and text data (a sentence or phrase). A common solution is to learn a joint embedding for images and text into a shared latent space where vectors from the two different modalities can be compared directly. This space is usually of low dimension and is very convenient for cross-view tasks such as image-to-text and text-to-image retrieval.

Several recent embedding methods [14, 15, 26] are based on Canonical Correlation Analysis (CCA) [17], which finds linear projections that maximize the correlation between projected vectors from the two views. Kernel CCA [17] is an extension of CCA in which maximally correlated nonlinear projections, restricted to reproducing kernel Hilbert spaces with corresponding kernels, are found. Extensions of CCA to a deep learning framework have also been proposed [1, 33]. However, as pointed out in [30], CCA is hard to scale to large amounts of data. In particular, stochastic gradient descent (SGD) techniques cannot guarantee a good solution to the original generalized eigenvalue problem, since covariance estimated in each small batch (due to the GPU memory limit) is extremely unstable.

An alternative to CCA is to learn a joint embedding space using SGD with a ranking loss. WSABIE [49] and DeVISE [11] learn linear transformations of visual and textual features to the shared space using a *single-directional* ranking loss that applies a margin-based penalty to incorrect annotations that get ranked higher than correct ones for each training image. Compared to CCA-based methods, this ranking loss easily scales to large amounts of data with stochastic optimization in training. As a more powerful objective function, a few other works have proposed a *bi-directional* ranking loss that, in addition to ensuring that correct sentences for each training image get ranked above incorrect ones, also ensures that for each sentence, the image described by that sentence gets ranked above images described by other sentences [22, 23, 25, 43]. However, to date, it has proven frustratingly difficult to beat CCA with an SGD-trained embedding: Klein et al. [26] have shown that properly normalized CCA [14] on top of state-of-the-art image and text features can outperform considerably more complex models.

Another strand of research on multi-modal embeddings is based on deep learning [3, 24, 25, 31, 35, 44], utilizing such techniques as deep Boltzmann machines [44], autoencoders [35], LSTMs [8], and recurrent neural networks [31, 45]. By making it possible learn nonlinear mappings, deep methods can in principle provide greater representational power than methods based on linear projections [11, 15, 26, 49].

In this work, we propose to learn an image-text embedding using a two-view neural network with two layers of nonlinearities on top of any representations of the image and text views (Figure 1). These representations can be given by the outputs of two pre-trained networks, off-the-shelf feature extractors, or trained jointly end-to-end with the embedding. To train this network, we use a bi-directional loss function similar to [22, 23, 25, 43], combined with con-

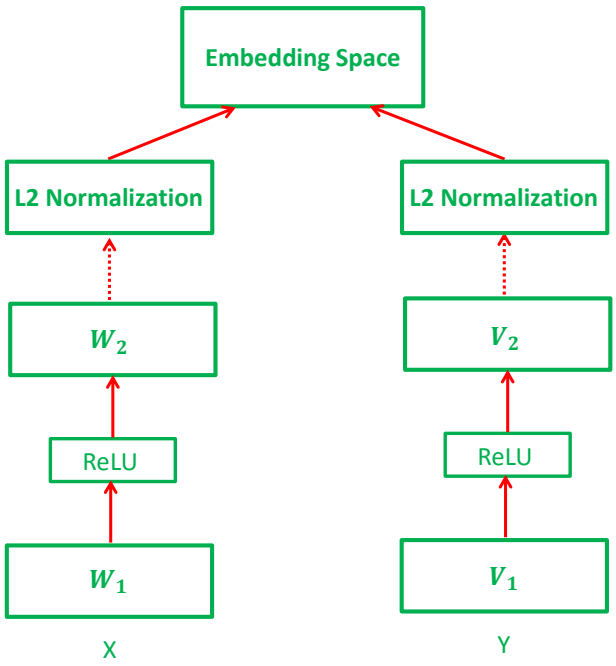

Figure 1. Our model structure: there are two branches in the network, one for images (X) and the other for text (Y). Each branch consists of fully connected layers with ReLU nonlinearities between them, followed by L2 normalization at the end.

straints that preserve neighborhood structure within each individual view. Specifically, in the learned latent space, we want images (resp. sentences) with similar meaning to be close to each other. Such within-view structure preservation constraints have been extensively explored in the metric learning literature [19, 32, 40, 41, 48, 53]. In particular, the Large Margin Nearest Neighbor (LMNN) approach [48] tries to ensure that for each image its target neighbors from the same class are closer than samples from other classes. As our work will show, these constraints can also provide a useful regularization term for the cross-view matching task.

From the viewpoint of architecture, our method is similar to the two-branch Deep CCA models [1, 33], though it avoids Deep CCA's training-time difficulties associated with covariance matrix estimation. Our network also gains in accuracy by performing feature normalization (L2 and batch normalization) before the embedding loss layer. Finally, our work is related to deep similarity learning [4, 6, 7, 16, 18, 39, 47], though we are solving a cross-view, not a within-view, matching problem. Siamese networks for similarity learning (e.g., [39]) can be considered as special cases of our framework where the two views come from the same modality and the two branches share weights.

Our proposed approach substantially improves the state of the art for image-to-sentence and sentence-to-image retrieval on the Flickr30K [51] and MSCOCO [28] datasets. We are also able to obtain convincing improvements over CCA on phrase localization for the Flickr30K Entities dataset [37].

2. Deep Structure-Preserving Embedding

Let X and Y denote the collections of training images and sentences, each encoded according to their own feature vector representation. We want to map the image and sentence vectors (which may have different dimensions initially) to a joint space of common dimension. We use the inner product over the embedding space to measure similarity, which is equivalent to the Euclidean distance since the outputs of the two embeddings are L2-normalized. In the following, $d(x, y)$ will denote the Euclidean distance between image and sentence vectors in the embedded space.

2.1. Network Structure

We propose to learn a nonlinear embedding in a deep neural network framework. As shown in Figure 1, our deep model has two branches, each composed of fully connected layers with weight matrices W_l and V_l. Successive layers are separated by Rectified Linear Unit (ReLU) nonlinearities. We apply batch normalization [20] right after the last linear layer. And at the end of each branch, we add L2 normalization.

In general, each branch can have a different number of layers, and if the inputs of the two branches X and Y are produced by their own networks, the parameters of those networks can be trained (or fine-tuned) together with the parameters of the embedding layers. However, in this paper, we have obtained very satisfactory results by using two embedding layers per branch on top of pre-computed image and text features (see Section 3.1 for details).

2.2. Training Objective

Our training objective is a stochastic margin-based loss that includes bidirectional cross-view ranking constraints, together with within-view structure-preserving constraints.

Bi-directional ranking constraints. Given a training image x_i, let Y_i^+ and Y_i^- denote its sets of matching (positive) and non-matching (negative) sentences, respectively. We want the distance between x_i and each positive sentence y_j to be smaller than the distance between x_i and each negative sentence y_k by some enforced margin m:

$$d(x_i, y_j) + m < d(x_i, y_k) \quad \forall y_j \in Y_i^+, \forall y_k \in Y_i^- \,. \quad (1)$$

Similarly, given a sentence $y_{i'}$, we have

$$d(x_{j'}, y_{i'}) + m < d(x_{k'}, y_{i'}) \quad \forall x_{j'} \in X_{i'}^+, \forall x_{k'} \in X_{i'}^-, \quad (2)$$

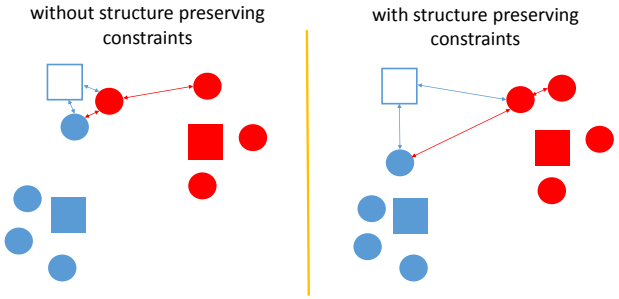

Figure 2. Illustration of the proposed structure-preserving constraints for joint embedding learning (see text). Rectangles represent images and circles represent sentences. Same color indicates matching images and sentences.

where $X_{i'}^{+}$ and $X_{i'}^{-}$ denote the sets of matching (positive) and non-matching (negative) images for $y_{i'}$.

Structure-preserving constraints. Let $N(x_i)$ denote the neighborhood of x_i containing images that share the same meaning. In our case, this is the set of images described by the same sentence as x_i. Then we want to enforce a margin of m between $N(x_i)$ and any point outside of the neighborhood:

$$d(x_i, x_j) + m < d(x_i, x_k) \quad \forall x_j \in N(x_i), \forall x_k \notin N(x_i), \tag{3}$$

Analogously to (3), we define the constraints for the sentence side as

$$d(y_{i'}, y_{j'}) + m < d(y_{i'}, y_{k'}) \quad \forall y_{j'} \in N(y_{i'}), \forall y_{k'} \notin N(y_{i'}), \tag{4}$$

where $N(y_{i'})$ contains sentences describing the same image.

Figure 2 gives an intuitive illustration of how within-view structure preservation can help with cross-view matching. The embedding space on the left satisfies the cross-view matching property. That is, each square (representing an image) is closer to all circles of the same color (representing its corresponding sentences) than to any circles of the other color. Similarly, for any circle (sentence), the closest square (image) has the same color. However, for the new image query (white square), the embedding space gives an ambiguous matching result since both red and blue circles are very close to it. This problem is mitigated in the embedding on the right, where within-view structure constraints are added, pushing semantically similar sentences (same color circles) closer to each other.

Note that our two image-sentence datasets, Flickr30K and MSCOCO, consist of images paired with five sentences each. The neighborhood of each image, $N(x_i)$, generally only contains x_i itself, since it is rare for two different images to be described by an identical sentence. Thus, the image-view constraints (eq. 3) are trivial, while the neighborhood of each sentence $N(y_{i'})$ has five members. However, for the region-phrase dataset of Section 3.3, many phrases have multiple region exemplars, so we get a non-trivial set of constraints for the image view.

Embedding Loss Function. We convert the constraints to our training objective in the standard way using hinge loss. The resulting loss function is given by

$$\begin{aligned} L(X, Y) = & \sum_{i,j,k} \max[0, m + d(x_i, y_j) - d(x_i, y_k)] \\ & + \lambda_1 \sum_{i',j',k'} \max[0, m + d(x_{j'}, y_{i'}) - d(x_{k'}, y_{i'})] \\ & + \lambda_2 \sum_{i,j,k} \max[0, m + d(x_i, x_j) - d(x_i, x_k)] \\ & + \lambda_3 \sum_{i',j',k'} \max[0, m + d(y_{i'}, y_{j'}) - d(y_{i'}, y_{k'})] \,, \end{aligned} \tag{5}$$

where the sums are over all triplets defined as in the constraints (1-4). The margin m could be different for different types of distance or even different instances. But to make it easy to optimize, we fix m for all terms across all training samples ($m = 0.1$ in the experiments). The weight λ_1 balances the strengths of both ranking terms. In other work with a bi-directional ranking loss [22, 23, 25, 43], this is always set to 1, but in our case, we found $\lambda_1 = 2$ produces the best results. The weights λ_2, λ_3 control the importance of the structure-preserving terms, which act as regularizers for the bi-directional retrieval tasks. We usually set both to small values like 0.1 or 0.2 (see Section 3 for details).

Triplet sampling. Our loss involves all triplets consisting of a target instance, a positive match, and a negative match. Optimizing over all such triplets is computationally infeasible. Therefore, we sample triplets within each mini-batch and optimize our loss function using SGD. Inspired by [21, 40], instead of choosing the most violating negative match in all instance space, we select top K most violated matches in each mini-batch. This is done by computing pairwise similarities between all (x_i, y_j), (x_i, x_j) and (y_i, y_j) within the mini-batch. For each positive pair (i.e., a ground truth image-sentence pair, two neighboring images, or two neighboring sentences), we then find at most top K violations of each relevant constraint (we use $K = 50$ in the implementation, although most pairs have many fewer violations). Theoretical guarantees of such a sampling strategy have been discussed in [40], though not in the context of deep learning. In our experiments, we observe convergence within 30 epochs on average.

In Section 3, we will demonstrate the performance of our method both with and without structure-preserving constraints. For training the network without these constraints,

we randomly sample 1500 pairs (x_i, y_i) to form our mini-batches. For the experiments with the structure-preserving constraints, in order to get a non-empty set of constraint triplets, we need a moderate number of positive pairs (i.e., at least two sentences that are matched to the same image) in each mini-batch. However, random sampling of pairs cannot guarantee this. Therefore, for each x_i in a given mini-batch, we add one more positive sentence distinct from the ones that may already be included among the sampled pairs, resulting in mini-batches of variable size.

3. Experiments

In this section, we analyze the contributions of different components of our method and evaluate it on image-to-sentence and sentence-to-image retrieval on popular Flickr30K [51] and MSCOCO [28] datasets, and on phrase localization on the new Flickr30K Entities dataset [37].

3.1. Features and Network Settings

In image-sentence retrieval experiments, to represent images, we follow the implementation details in [26, 37]. Given an image, we extract the 4096-dimensional activations from the 19-layer VGG model [42]. Following standard procedure, the original 256×256 image is cropped in ten different ways into 224×224 images: the four corners, the center, and their x-axis mirror image. The mean intensity is then subtracted from each color channel, the resulting images are encoded by the network, and the network outputs are averaged.

To represent sentences and phrases, we primarily use the Fisher vector (FV) representation [36] as suggested by Klein et al. [26]. Starting with 300-dimensional word2vec vectors [34] of the sentence words, we apply ICA as in [26] and construct a codebook with 30 centers using both first- and second-order information, resulting in sentence features of dimension $300 * 30 * 2 = 18000$. We only use the Hybrid Gaussian-Laplacian mixture model (HGLMM) from [26] for our experiments rather than the combined HGLMM+GMM model which obtained the best performance in [26]. To save memory and training time, we perform PCA on these 18000-dimensional vectors to reduce them to 6000 dimensions. PCA also makes the original features less sparse, which is good for the numerical stability of our training procedure.

Since FV is already a powerful hand-crafted nonlinear transformation of the original sentences, we are also interested in exploring the effectiveness of our approach on top of simpler text representations. To this end, we include results on 300-dimensional means of word2vec vectors of words in each sentence/phrase, and on tf-idf-weighted bag-of-words vectors. For tf-idf, we pre-process all the sentences with WordNet's lemmatizer [5] and remove stop words. For the Flickr30K dataset, our dictionary size (and descriptor dimensionality) is 3000, and for MSCOCO, it is 5600.

For our experiments using tf-idf or FV text features, we set the embedding dimension to be 512. On the image (X) side, when using 4096-dimensional visual features, W_1 is a 4096×2048 matrix, and W_2 is a 2048×512 matrix. That is, the output dimensions of the two layers are [2048, 512]. On the text (Y) side, the output dimensions of the V_1 and V_2 layers are [2048, 512]. For the experiments using 300-D word2vec features, we use a lower dimension (256) for the embedding space and the intermediate layers output are accordingly changed to [1024, 256].

We train our networks using SGD with momentum 0.9 and weight decay 0.0005. We use a small learning rate starting with 0.1 and decay the learning rate by 0.1 after every 10 epochs. To accelerate the training and also make gradient updates more stable, we apply batch normalization [20] right after the last linear layer of both network branches. We also use a Dropout layer after ReLU with probability = 0.5. We set the mini-batch size to 1500 ground truth image-sentence pairs and augment these pairs as necessary as described in the previous section. Compared with CCA-based methods, our method has much smaller memory requirements and is scalable to larger amounts of data.

3.2. Image-sentence retrieval

In this section, we report results on image-to-sentence and sentence-to-image retrieval on the standard Flickr30K [51] and MSCOCO [28] datasets. Flickr30K [51] consists of 31783 images accompanied by five descriptive sentences each. The larger MSCOCO dataset [28] consists of 123000 images, also with five sentences each.

For evaluation, we follow the same protocols as other recent work [22, 26, 37]. For Flickr30K, given a test set of 1000 images and 5000 corresponding sentences, we use the images to retrieve sentences and vice versa, and report performance as Recall@K ($K = 1, 5, 10$), or the percentage of queries for which at least one correct ground truth match was ranked among the top K matches. For MSCOCO, consistent with [22, 26], we also report results on 1000 test images and their corresponding sentences.

For Flickr30K, bidirectional retrieval results are listed in Table 1. Part (a) of the table summarizes the performance reported by a number of competing recent methods. In Part (b) we demonstrate the impact of different components of our model by reporting results for the following variants.

- Linear + one-directional: In this setting, we keep only the first layers in each branch with parameters W_1, V_1, immediately followed by L2 normalization. The output dimensions of W_1 and V_1 are changed to be the embedding space dimension. In the objective function (eq. 5), we set $\lambda_1 = 0, \lambda_2 = 0, \lambda_3 = 0$, only retaining

	Methods on Flickr30K	Image-to-sentence			Sentence-to-image		
		R@1	R@5	R@10	R@1	R@5	R@10
(a) State of the art	Deep CCA [33]	27.9	56.9	68.2	26.8	52.9	66.9
	mCNN(ensemble) [29]	33.6	64.1	74.9	26.2	56.3	69.6
	m-RNN-vgg [31]	35.4	63.8	73.7	22.8	50.7	63.1
	Mean vector [26]	24.8	52.5	64.3	20.5	46.3	59.3
	CCA (FV HGLMM) [26]	34.4	61.0	72.3	24.4	52.1	65.6
	CCA (FV GMM+HGLMM) [26]	35.0	62.0	73.8	25.0	52.7	66.0
	CCA (FV HGLMM) [37]	36.5	62.2	73.3	24.7	53.4	66.8
(b) Fisher vector	Linear + one-directional	33.5	61.7	73.6	21.0	47.4	60.5
	Linear + bi-directional	34.6	64.3	74.9	24.2	52.0	64.2
	Linear + bi-directional + structure	35.2	66.8	76.2	25.6	54.8	66.5
	Nonlinear + one-directional	37.5	65.6	76.9	22.4	50.9	63.3
	Nonlinear + bi-directional	39.3	68.0	78.3	28.1	59.2	71.2
	Nonlinear + bi-directional + structure	**40.3**	**68.9**	**79.9**	**29.7**	**60.1**	**72.1**
(c) Mean vector	Nonlinear + bi-directional	33.5	60.2	71.9	22.8	52.5	65.0
	Nonlinear + bi-directional + structure	35.7	62.9	74.4	25.1	53.9	66.5
(d) tf-idf	Nonlinear + bi-directional	38.7	66.6	76.9	27.6	57.0	69.0
	Nonlinear + bi-directional + structure	40.1	67.6	78.2	28.1	58.5	69.8

Table 1. Bidirectional retrieval results. The numbers in (a) come from published papers, and the numbers in (b-d) are results of our approach using different textual features. Note that the Deep CCA results in [33] were obtained with AlexNet [27]. The results of our method with AlexNet are still about 3% higher than those of [33] for image-to-sentence retrieval and 1% higher for sentence-to-image retrieval.

the image-to-sentence ranking constraints. This results in a model similar to WSABIE [49].

- Linear + bi-directional: The model structure is as above, and in eq. (5), we set $\lambda_1 = 2, \lambda_2 = 0, \lambda_3 = 0$. This form of embedding is similar to [22, 23, 25, 43] (though the details of the representations used by those works are quite different).
- Linear + bi-directional + structure: same linear model, eq. (5) with $\lambda_1 = 2, \lambda_2 = 0, \lambda_3 = 0.2$.
- Nonlinear + one-directional: Network as in Figure 1, eq. (5) with $\lambda_1 = 0, \lambda_2 = 0, \lambda_3 = 0$.
- Nonlinear + bi-directional: Network as in Figure 1, eq. (5) with $\lambda_1 = 2, \lambda_2 = 0, \lambda_3 = 0$.
- Nonlinear + bi-directional + structure: Network as in Figure 1, eq. (5) with $\lambda_1 = 2, \lambda_2 = 0, \lambda_3 = 0.2$.

Note that in all the above configurations we have $\lambda_2 = 0$, that is, the structure-preserving constraint associated with the image space is inactive, since in the Flickr30K and MSCOCO datasets we do not have direct supervisory information about multiple images that can be described by the same sentence. However, our results for the region-phrase dataset of Section 3.3 will incorporate structure-preserving constraints on both spaces.

From Table 1 (b), we can see that changing the embedding function from linear to nonlinear improves the accuracy by about 4% across the board. Going from one-directional to bi-directional constraints improves the accuracy by 1-2% for image-to-sentence retrieval and by a bigger amount for sentence-to-image retrieval. Finally, adding the structure-preserving constraints provides an additional improvement of 1-2% in both linear and nonlinear cases. The methods from Table 1 (a) most comparable to ours are CCA (HGLMM) [26, 37], since they use the same underlying feature representation with linear CCA. Our linear model with all the constraints of eq. (5) does not outperform linear CCA, but our nonlinear one does.

Finally, to check how much our method relies on the power of the input features, parts (c) and (d) of Table 1 report results for our nonlinear models with and without structure-preserving constraints applied on top of weaker text representations, namely mean of word2vec vectors of the sentence and tf-idf vectors, as described in Section 3.1. Once again, we can see that structure-preserving constraints give us an additional improvement. Our results with mean vector are considerably better than the CCA results of [26] on the same feature, and are in fact comparable with the results of [26, 37] on top of the more powerful FV representation. For tf-idf, we achieve results that are just below our best FV results, showing that we do not require a highly nonlinear feature as an input in order to learn a good embedding. Another possible reason why tf-idf performs so strongly may be that word2vec features are pre-trained on an unrelated text corpus, so they may not be as well adapted to our specific data.

For MSCOCO, results on 1000 test images are listed in Table 2. The trends are the same as in Table 1: adding structure-preserving constraints on the sentence space con-

	Methods on MSCOCO 1000 testing set	Image-to-sentence			Sentence-to-image		
		R@1	R@5	R@10	R@1	R@5	R@10
(a) State of the art	Mean vector [26]	33.2	61.8	75.1	24.2	56.4	72.4
	CCA (FV HGLMM) [26]	37.7	66.6	79.1	24.9	58.8	76.5
	CCA (FV GMM+HGLMM) [26]	39.4	67.9	80.9	25.1	59.8	76.6
	DVSA [22]	38.4	69.9	80.5	27.4	60.2	74.8
	m-RNN-vgg [31]	41.0	73.0	83.5	29.0	42.2	77.0
	mCNN(ensemble) [29]	42.8	73.1	84.1	32.6	68.6	82.8
(b) Fisher Vector	Nonlinear+bi-directional	47.5	77.6	88.3	36.8	72.2	85.6
	Nonlinear+bi-directional+structure	**50.1**	**79.7**	**89.2**	**39.6**	**75.2**	**86.9**
(c) Mean Vector	Nonlinear+bi-directional	39.6	74.0	84.8	32.0	67.3	81.6
	Nonlinear+bi-directional+structure	40.7	74.2	85.3	33.5	68.7	83.2
(d) tf-idf	Nonlinear+bi-directional	45.3	77.6	86.8	35.4	70.2	83.4
	Nonlinear+bi-directional+structure	46.7	77.9	87.7	36.2	72.3	84.7

Table 2. Bidirectional retrieval results on MSCOCO 1000-image test set.

sistently improves performance, and our results with the FV text feature considerably exceed the state of the art. We have also tried fine-tuning the VGG network by back-propagating our loss function through all the VGG layers, and obtained about 0.5% additional improvement.

3.3. Phrase Localization on Flickr30K Entities

The recently published Flickr30K Entities dataset [37] allows us to learn correspondences between phrases and image regions. Specifically, the annotations in this dataset provide links from 244K mentions of distinct entities in sentences to 276K ground truth bounding boxes (some entities consist of multiple instances, such as "group of people"). We are interested in this dataset because unlike the global image-sentence datasets, it provides many-to-many correspondences, i.e., each region may be described by multiple phrases and each phrase may have multiple region exemplars across multiple images. This allows us to take advantage of structure-preserving constraints on both the visual and textual spaces.

As formulated in [37], the goal of phrase localization is to predict a bounding box in an image for each entity mention (noun phrase) from a caption that goes with that image. For a particular phrase, we perform the search by extracting 100 EdgeBox [54] region proposals and scoring them using our embedding. To get good performance, the best-scoring box should have high overlap with the ground truth region. This can be considered as a ranking problem, and both CCA and our methods can be trained to match phrases and regions. On the other hand, we should realize that this problem is more like detection, where the algorithm should be able to distinguish foreground objects from boxes that contain only background or poorly localized objects. CCA and Deep CCA are not well suited to this scenario, since there is no way to add negative boxes into their learning stage. However, our margin-based loss function makes it possible.

Plummer et al. [37] reported baseline results for a region-phrase embedding using CCA on top of ImageNet-trained VGG features. Following Rohrbach et al. [38], who obtained big improvements on phrase localization using detection-based VGG features, we also use Fast R-CNN features [13] fine-tuned on a union of the PASCAL 2007 and 2012 train-val sets [9]. Consistent with [37], we do not average multiple crops for region features. For text, in this section we use only the FV feature. Thus, the input dimension of X is 4096 and the input dimension of Y is 6000 as before (reduced by PCA from the original 18000-D FV). We use the two-layer network structure with $[8192, 4096]$ as the intermediate layer dimensions on both the X and Y sides (note that on the X side, the intermediate layer actually doubles the feature dimension).

For our first experiment, we train our embedding without negative mining, using the same positive region-phrase pairs as CCA. For this, we use the same training set as [37], which is resampled with at most ten regions per phrase, for a total of 137133 region-phrase pairs, 70759 of which are unique. As in the previous section, we use initial mini-batch size of 1500. But now, for the full version of our objective (eq. 5), we augment the mini-batches by sampling not only additional positive phrases for regions, but also additional positive regions for phrases, to make sure that we have as many triplets as possible for structure-preserving constraints on the region side (eq. 3) and the phrase side (eq. 4).

The results of training our model without negative mining for 28 epochs are shown in the top part of Table 3. We use the evaluation protocol proposed by [37]. First, we treat phrase localization as the problem of retrieving instances of a query phrase from a set of region proposals extracted from test images, and report Recall@K, or the percentage of queries for which a correct match has rank of at most K (a region proposal is considered to be a correct match if it has IOU of at least 0.5 with the ground-truth bounding box

Methods	R@1	R@5	R@10	mAP(all)
CCA baseline	40.11	61.52	67.17	41.96
Our method without negative mining				
(a) $\lambda_1 = 2,\ \lambda_2 = 0,\ \lambda_3 = 0$	35.83	60.51	66.70	40.50
(b) $\lambda_1 = 2,\ \lambda_2 = 0,\ \lambda_3 = 0.1$	36.59	60.44	66.92	40.85
(c) $\lambda_1 = 2,\ \lambda_2 = 0.1,\ \lambda_3 = 0$	**36.74**	60.35	66.73	**41.22**
(d) $\lambda_1 = 2,\ \lambda_2 = 0.1,\ \lambda_3 = 0.1$	36.72	**61.14**	**67.21**	41.13
Fine-tuned with negative mining				
Fine-tuning (a) for 5 epochs	41.77	63.01	68.27	46.55
Fine-tuning (b) for 5 epochs	43.77	64.22	**68.84**	47.38
Fine-tuning (c) for 5 epochs	42.88	63.41	68.47	46.78
Fine-tuning (d) for 5 epochs	**43.89**	**64.46**	68.66	**47.72**

Table 3. Phrase localization results on Flickr30K Entities using Fast-RCNN features. We use 100 EdgeBox proposals, for which the recall upper bound is $R@100 = 76.91$.

for that phrase). Second, we report average precision (AP) of ranking bounding boxes for each phrase in the test images that contain that phrase, following nonmaximum suppression. The last column of Table 3 shows mAP over all unique phrases in the test set, with each unique phrase being treated as its own class label.

Table 3 (a-d) shows the performance of our bi-directional ranking objective with different combinations of structure terms. We can see that including the structure terms generally gives better results than excluding them, though the effects of turning on each term separately do not differ too much. In large part, this is because of the limited number of structure-preserving constraint triples for each view. In the Flickr30K Entities training set, for all 130K pairs, there are around 70K unique phrases and 80K regions described by a single phrase. This means, that, for most phrases/regions, there are no more than two corresponding regions/phrases. The top line of Table 3 gives baseline CCA results. For the pre-trained model without using negative mining, our deep embedding has comparable results with CCA on Recall@5 and Recall@10, but lower results on Recall@1. As mentioned earlier, in our past experience we have found CCA to be surprisingly hard to beat with more complex methods [15, 37].

In order to further improve the accuracy of our embedding, we need to refine it using negative data from background and poorly localized regions. To do this, we take the embedding trained without negative mining, and for each unique phrase in the training set, calculate the distance between this phrase and the ground truth boxes as well as all our proposal boxes. Then we record those "hard negative" boxes that are closer to the phrase than the ground truth boxes. For efficiency, we only sample at most 50 hard negative regions for each unique phrase. Next, we continue training our region-phrase model on a training set augmented with these hard negative boxes, using only the bi-directional ranking constraints (eqs. 1 and 2). We exclude the structure-preserving constraints because they would now be even more severely outnumbered by the bi-directional ranking constraints.

The last four lines of Table 3 show the results of fine-tuning the models from Table 3 (a-d) with hard negative samples. Compared to the best model trained with only positive regions, our Recall@1 and mAP have improved by almost 6%, and are now considerably better than CCA. Note that in absolute terms, Rohrbach et al. [38] get higher results, with a R@1 of over 47%, but they use a much more complex method that includes LSTMs with a phrase reconstruction objective.

Finally, Figure 3 shows examples of phrase localization in four images where our model improves upon the CCA baseline.

4. Conclusion

This paper has proposed an image-text embedding method in which a two-branch network with multiple layers is trained using a margin-based objective function consisting of bi-directional ranking terms and structure-preserving terms inspired by metric learning. Our architecture is simple and flexible, and can be applied to various kinds of visual and textual features. Extensive experiments demonstrate that the components of our system are well chosen and all the terms in our objective function are justified. To the best of our knowledge, our retrieval results on Flickr30K and MSCOCO datasets considerably exceed the state of the art, and we also demonstrate convincing improvements over CCA on the new problem of phrase localization on the Flickr30K Entities dataset.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant CIF-1302438, Xerox UAC, and the Sloan Foundation. We would like to thank Bryan Plummer for help with phrase localization evaluation.

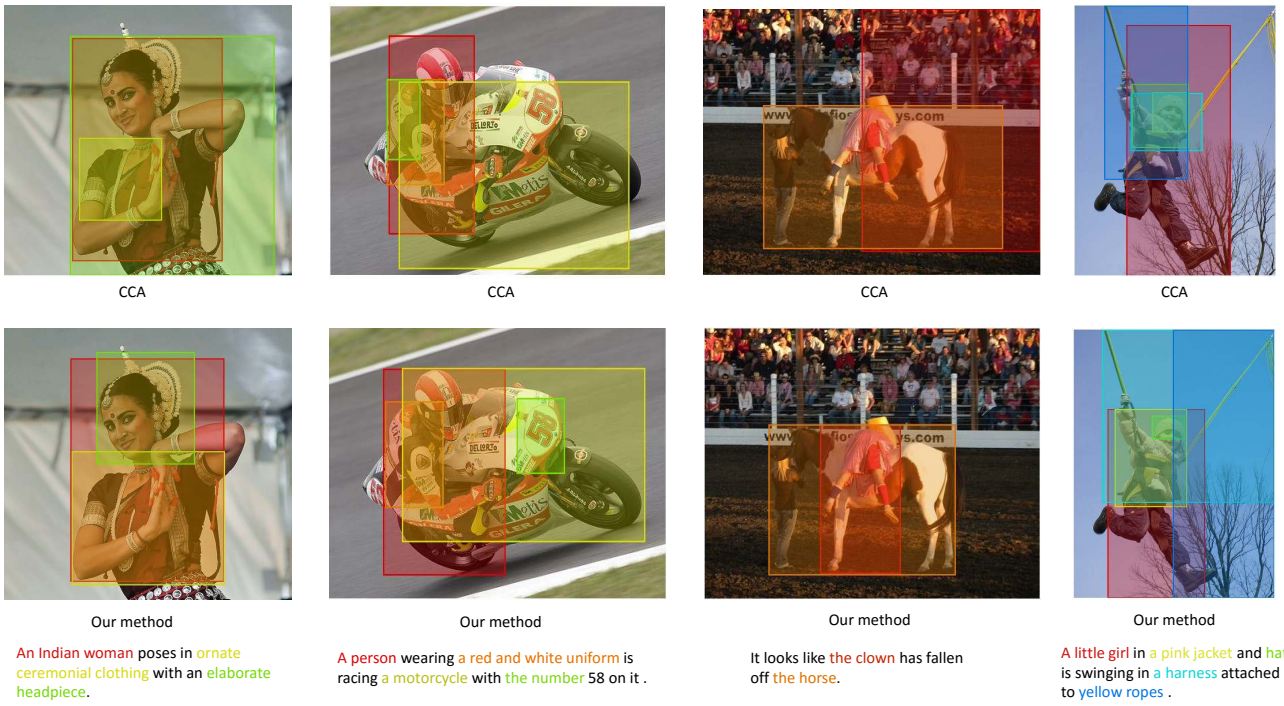

Figure 3. Example phrase localization results. For each image and reference sentence, phrases and best-scoring corresponding regions are shown in the same color. The first row shows the output of the CCA method [37] and the second row shows the output of our best model (fine-tuned model (d) in Table 3 with negative mining). For the first (left) example, our method gives more accurate bounding boxes for the clothing and headpiece. For the second example, our method finds the correct bounding box for the number 58 while CCA completely misses it; for the third column, our method gives much tighter boxes for the horse and clown; and for the last example, our method accurately locates the hat and jacket.

References

[1] G. Andrew, R. Arora, J. Bilmes, and K. Livescu. Deep canonical correlation analysis. In *ICML*, 2013.

[2] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh. Vqa: Visual question answering. *arXiv:1505.00468*, 2015.

[3] J. Ba, K. Swersky, S. Fidler, and R. Salakhutdinov. Predicting deep zero-shot convolutional neural networks using textual descriptions. *ICCV*, 2015.

[4] S. Bell and K. Bala. Learning visual similarity for product design with convolutional neural networks. *ACM Transactions on Graphics (TOG)*, 34(4):98, 2015.

[5] S. Bird. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*. Association for Computational Linguistics, 2006.

[6] J. Bromley, J. W. Bentz, L. Bottou, I. Guyon, Y. LeCun, C. Moore, E. Säckinger, and R. Shah. Signature verification using a siamese time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence*, 7(04):669–688, 1993.

[7] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*, 2005.

[8] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. *arXiv:1411.4389*, 2014.

[9] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge 2012, 2011.

[10] H. Fang, S. Gupta, F. Iandola, R. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. Platt, et al. From captions to visual concepts and back. *CVPR*, 2015.

[11] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, T. Mikolov, et al. Devise: A deep visual-semantic embedding model. In *NIPS*, 2013.

[12] H. Gao, J. Mao, J. Zhou, Z. Huang, L. Wang, and W. Xu. Are you talking to a machine? dataset and methods for multilingual image question answering. *NIPS*, 2015.

[13] R. Girshick. Fast r-cnn. In *ICCV*, 2015.

[14] Y. Gong, Q. Ke, M. Isard, and S. Lazebnik. A multi-view embedding space for modeling internet images, tags, and their semantics. *IJCV*, 106(2):210–233, 2014.

[15] Y. Gong, L. Wang, M. Hodosh, J. Hockenmaier, and S. Lazebnik. Improving image-sentence embeddings using large weakly annotated photo collections. In *ECCV*. 2014.

[16] X. Han, T. Leung, Y. Jia, R. Sukthankar, and A. C. Berg. Matchnet: Unifying feature and metric learning for patch-based matching. In *CVPR*, 2015.

[17] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor. Canonical correlation analysis: An overview with application to learning methods. *Neural computation*, 16(12):2639–2664, 2004.

[18] E. Hoffer and N. Ailon. Deep metric learning using triplet network. *ICLR*, 2015.

[19] J. Hu, J. Lu, and Y.-P. Tan. Discriminative deep metric learning for face verification in the wild. In *CVPR*, 2014.

[20] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *ICML*, 2015.

[21] T. Joachims, T. Finley, and C.-N. J. Yu. Cutting-plane training of structural svms. *Machine Learning*, 77(1):27–59, 2009.

[22] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. *CVPR*, 2015.

[23] A. Karpathy, A. Joulin, and F. F. Li. Deep fragment embeddings for bidirectional image sentence mapping. In *NIPS*, 2014.

[24] R. Kiros, R. Salakhutdinov, and R. Zemel. Multimodal neural language models. In *ICML*, 2014.

[25] R. Kiros, R. Salakhutdinov, and R. S. Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv:1411.2539*, 2014.

[26] B. Klein, G. Lev, G. Sadeh, and L. Wolf. Fisher vectors derived from hybrid gaussian-laplacian mixture models for image annotation. *CVPR*, 2015.

[27] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, 2012.

[28] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*. 2014.

[29] L. Ma, Z. Lu, L. Shang, and H. Li. Multimodal convolutional neural networks for matching image and sentence. *ICCV*, 2015.

[30] Z. Ma, Y. Lu, and D. Foster. Finding linear structure in large datasets with scalable canonical correlation analysis. *ICML*, 2015.

[31] J. Mao, W. Xu, Y. Yang, J. Wang, and A. Yuille. Deep captioning with multimodal recurrent neural networks (m-rnn). *ICLR*, 2015.

[32] T. Mensink, J. Verbeek, F. Perronnin, and G. Csurka. Metric learning for large scale image classification: Generalizing to new classes at near-zero cost. In *ECCV*. 2012.

[33] F. Y. K. Mikolajczyk. Deep correlation for matching images and text. In *CVPR*, 2015.

[34] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013.

[35] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng. Multimodal deep learning. In *ICML*, 2011.

[36] F. Perronnin, J. Sanchez, and T. Mensink. Improving the Fisher kernel for large-scale image classification. In *ECCV*. 2010.

[37] B. Plummer, L. Wang, C. Cervantes, J. Caicedo, J. Hockenmaier, and S. Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. *ICCV*, 2015.

[38] A. Rohrbach, M. Rohrbach, R. Hu, T. Darrell, and B. Schiele. Grounding of textual phrases in images by reconstruction. *arXiv:1511.03745*, 2016.

[39] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. *CVPR*, 2015.

[40] B. Shaw, B. Huang, and T. Jebara. Learning a distance metric from a network. In *NIPS*, 2011.

[41] B. Shaw and T. Jebara. Structure preserving embedding. In *ICML*, 2009.

[42] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*, 2014.

[43] R. Socher, A. Karpathy, Q. V. Le, C. D. Manning, and A. Y. Ng. Grounded compositional semantics for finding and describing images with sentences. *Transactions of the Association for Computational Linguistics*, 2:207–218, 2014.

[44] N. Srivastava and R. R. Salakhutdinov. Multimodal learning with deep boltzmann machines. In *NIPS*, 2012.

[45] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko. Translating videos to natural language using deep recurrent neural networks. *arXiv:1412.4729*, 2014.

[46] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. *CVPR*, 2015.

[47] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu. Learning fine-grained image similarity with deep ranking. In *CVPR*, 2014.

[48] K. Q. Weinberger, J. Blitzer, and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. In *NIPS*, 2005.

[49] J. Weston, S. Bengio, and N. Usunier. Wsabie: Scaling up to large vocabulary image annotation. In *IJCAI*, 2011.

[50] K. Xu, J. Ba, R. Kiros, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. *ICML*, 2015.

[51] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014.

[52] L. Yu, E. Park, A. C. Berg, and T. L. Berg. Visual madlibs: Fill in the blank image generation and question answering. *ICCV*, 2015.

[53] J. Žbontar and Y. LeCun. Computing the stereo matching cost with a convolutional neural network. *arXiv:1409.4326*, 2014.

[54] C. L. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*. 2014.