Visual-Semantic Matching by Exploring High-Order Attention and Distraction

Yongzhi Li[1], Duo Zhang[2], Yadong Mu[3*]
[1]Center for Data Science, [2]EECS, [3]Wangxuan Institute of Computer Technology, Peking University
{yongzhili,zhduodyx,myd}@pku.edu.cn

Abstract

Cross-modality semantic matching is a vital task in computer vision and has attracted increasing attention in recent years. Existing methods mainly explore object-based alignment between image objects and text words. In this work, we address this task from two previously-ignored aspects: high-order semantic information (e.g., object-predicate-subject triplet, object-attribute pair) and visual distraction (i.e., despite the high relevance to textual query, images may also contain many prominent distracting objects or visual relations). Specifically, we build scene graphs for both visual and textual modalities. Our technical contributions are two-folds: firstly, we formulate the visual-semantic matching task as an attention-driven cross-modality scene graph matching problem. Graph convolutional networks (GCNs) are used to extract high-order information from two scene graphs. A novel cross-graph attention mechanism is proposed to contextually reweigh graph elements and calculate the inter-graph similarity; Secondly, some top-ranked samples are indeed false matching due to the co-occurrence of both highly-relevant and distracting information. We devise an information-theoretic measure for estimating semantic distraction and re-ranking the initial retrieval results. Comprehensive experiments and ablation studies on two large public datasets (MS-COCO and Flickr30K) demonstrate the superiority of the proposed method and the effectiveness of both high-order attention and distraction.

1. Introduction

The rapid growth of various multimedia data such as text, images, and videos has brought great difficulties to users in accurate and effective search nowadays. Among them, cross-modal search between vision and language is of great importance in practical applications. Therefore, cross-modal retrieval between images and text has attracted plenty of attention from researchers in recent years [43]. This paper focuses on cross-modal retrieval of text and images with complex semantics. For this task, how to effectively eliminate the huge gap between data of two different modalities is the key to solve the problem.

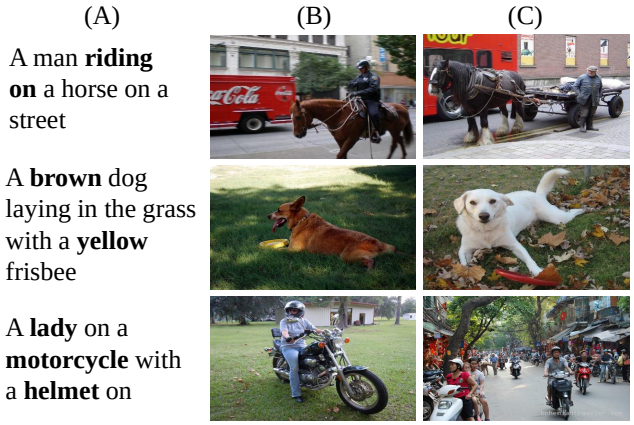

Figure 1. Illustration of the importance of predicate (top row), attributes (middle row), and semantic distraction (bottom row) in visual-semantic matching. From left to right, three columns represent the querying caption, ground-truth images, and the highly-ranked false matching returned by conventional methods, respectively. Key words in queries are highlighted in bold.

Thanks to advances in computer vision and natural language processing, some early developments [17, 5] have proposed to use pre-trained neural networks that encode multi-modality data into feature vectors, then project them into a common feature space and measure the similarity by computing the distance between their representations. Although such methods are capable of capturing global semantic information, they lack accurate modeling of high-level semantic information and are not clearly interpretable. Some modern methods have attempted to perform some more granular semantic learning to enhance the feature representation of data. For example, Karpathy *et al.* [14] aligned textual words and image regions for image captioning. The work in [10, 35] proposed using CNN to extract semantic concepts from pictures to enhance the representation of global features. Kuang *et al.* [20] introduced an attention mechanism to identify important parts of text and images.

A majority of these methods are based on first-order

*Corresponding author.

information like semantic concepts or objects. However, high-order information such as the relationships between objects and object attributes, are rarely explored in current literatures. When facing structural queries, these methods are often frustrated by ambiguous false matching. Some examples are provided in Figure 1. In the first example, even all key objects and scene concepts (man, horse and street) relevant to the querying sentence ("a man riding on a horse on a street") are precisely detected, one would still be unable to distinguish some confusing samples without considering high-order information in the query (*e.g.*, triplet relation *person-ride-horse* in this example). We regard *object-attribute* pair as another kind of informative high-order information. An example is found in the second row of Figure 1, which illustrates the importance of the color attribute.

To address this issue, we adopt scene graphs [13, 12] for representing highly-structural visual or textual semantics, and formulate visual-semantic matching as a heterogeneous graph matching problem. Figure 2 shows two exemplar scene graphs that encapsulate various pairwise or triple relationships, respectively. We propose to use Graph Convolutional Networks (GCNs) [16, 46] to contextualize each individual graph node. Both cross-graph attention and intra-graph self-attention mechanisms are developed to reweight each graph element and calculate the similarity between querying and reference scene graphs.

Additionally, we argue that the issue of semantic distraction still remains unexplored in the previous literatures. In specific, most of existing methods primarily concern the relevance between query and reference samples. However, whether matched semantics dominate a reference image or text is not considered. An example is provided in the third row of Figure 1. The rightmost image is highly-ranked since it contains all key words in the query, yet shall be actually classified as false matching given the co-occurrence of vast distracting contents (*e.g.*, the children, the pedestrians etc.). We are thus inspired to propose an information-theoretic metric to explicitly quantify visual distraction, which is used to re-rank the initial retrieved top matchings.

Our contributions can be summarized as below:

1) We aim to effectively explore high-order information in the visual-semantic matching task, particularly the object-predicate-subject and object-attribute types. Technically, GCNs are incorporated into our model for encoding above-mentioned high-order information. Multiple attention mechanisms are tailored for computing the similarity between querying and reference scene graphs.

2) To our best knowledge, we are the first to explicitly explore the visual distraction problem in structural visual-semantic matching. Informational entropy is novelly adapted to gauge the dominance of distracting factors in a reference image or text.

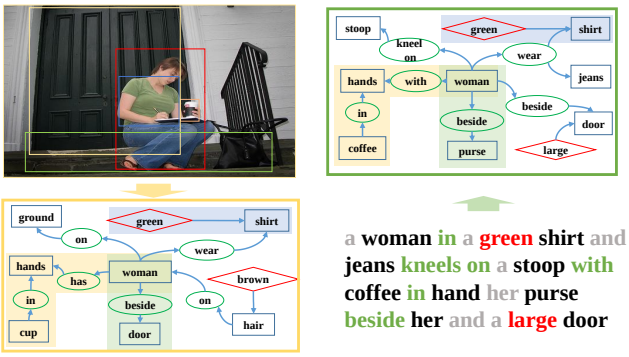

Figure 2. An image-text pair and their corresponding scene graphs. Graphical shapes represent different types of graph nodes.

2. Related Work

Embedding based methods. A widely used framework is to map the semantic embedding vectors of image and text into a common space and calculate the similarities according to the cosine or euclidean distance [37, 40, 18, 21, 4, 30, 3, 17, 7]. Kiros *et al*. [17] first used convolutional neural network (CNN) and recurrent neural network (RNN) to encode image and sentence features, and learned a cross-modal representation from triplet ranking loss. Gu *et al*. [7] proposed using generative object to enhance the fine-grained feature representations. Zheng *et al*. [51] suggested using a dual-task to embed the semantic features more discriminatively in the shared space. [39] introduced two-branch embeddings and proposed novel neighborhood constraints.

Semantic knowledge based methods. Several modern works explored the alignment of visual objects and textual words, as found in [31, 11, 29, 13]. The pioneering work [14] adopted an R-CNN [6] model to detect local regions from an image and aligned them with words in the sentence. Huang *et al*. [10] proposed learning semantics and orders to improve the image representations. A context-modulated attention scheme was developed in [9] to selectively attend instances appearing in both the image and sentence. Furthermore, Lee *et al.* proposed a method in [20], which used stacked cross attention to match two modalities in a finer-grained model. Some other research [38, 35] adopted external knowledge to further enhance the model capability.

Graph matching. Similarity-based graph search or matching has been a long-standing research task in a number of communities including data mining [45, 2, 28, 23] and natural language processing [44, 27]. Regarding the domain of computer vision, graph matching has been used for video re-identification [47] and 3D model retrieval [26] etc. With the development of graph convolutional networks (GCNs) [16] in recent years, the authors of [49] proposed a GMN network to align key-points in different images. [41] furthermore suggested an embedding based cross-graph affinity method to model the graph structure.

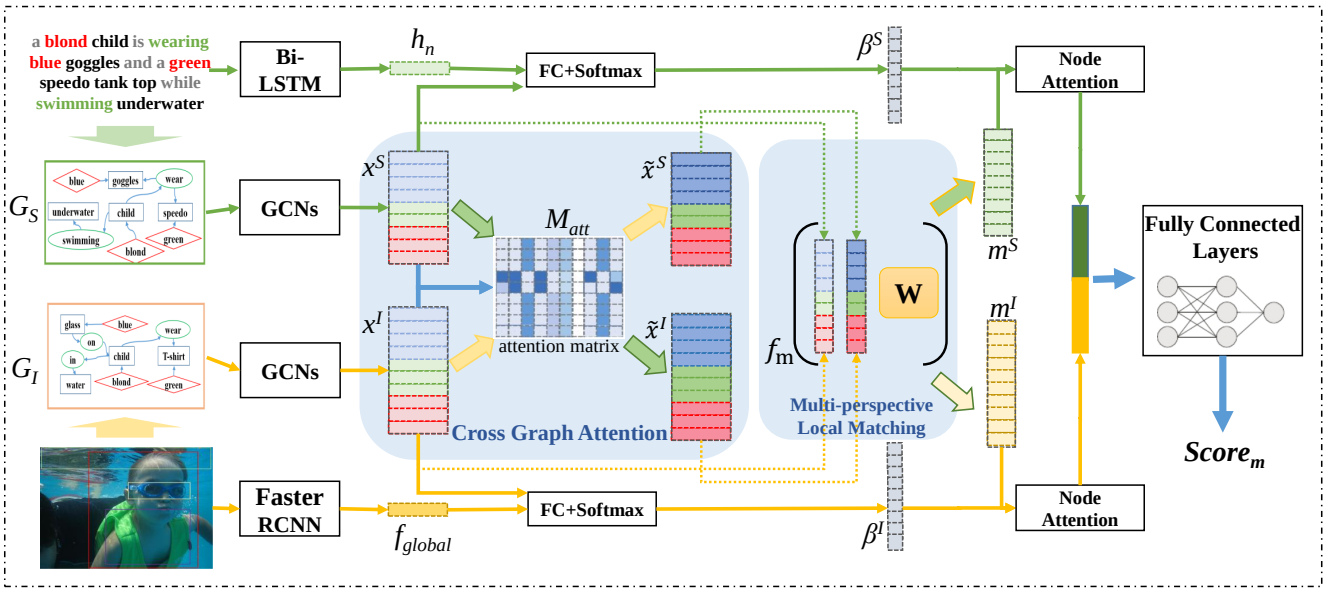

Figure 3. The overall architectural diagram of our model. Yellow arrows in the figure represent the data stream of visual information, and green arrows show the data stream of textual semantic information. The generation of two kinds of scene graph data is detailed in the main text. The final output of our proposed model is a similarity score for semantically matching these two heterogeneous scene graphs.

3. Approach

The image-text (or referred to as visual-semantic) matching problem is defined as follows: given an image-sentence pair, our model aims to calculate a similarity score between them, such that accurate cross-modality search is feasible.[1] As shown in Figure 3, our model utilize scene graphs to represent visual or textual semantic modalities. The extracted scene graphs first go through two graph convolutional networks (GCNs) to obtain contextualized embeddings. Intuitively, the importance of an object or relation in the reference heavily hinges on its relevance to the query. Inspired by this intuition, a cross-graph attention module is proposed for weighing graph nodes. Then, a multi-perspective local matching layer produces a matching vector for each node in the graphs, and a node attention mechanism is included to get global matching representation. Finally, the fully-connected layers predict the similarity score by taking the global representation as input.

3.1. Text Encoding and Sentence Scene Graph

To acquire the representation of a given sentence ***S***, a bi-directional LSTM [8] is adopted to model the long-range context information. For each word in ***S***, its index first passes an embedding layer to get a word embedding vector , which is then fed into the bi-LSTM to get a temporally-contextual representation h_i. The output of the last step h_n is used to represent the whole sentence.

To obtain a *sentence scene graph* (SSG), we use a fixed rule-based language parser [1] to translate the inputting sentences into scene graphs. Following the prior practice [46], we feed all the captions of an image into the parser and get a tuple $G_S = (N, E)$, where N and E are the sets of nodes and edges, respectively. we define three kinds of nodes in N: object node o, attribute node a, and relation node r that are represented as rectangles, diamonds and ellipses respectively in Figure 2. o_i represents the i_{th} object. The relationship between object o_i and o_j is denoted as r_{ij} , and $a_{i,l}$ is the l_{th} attribute of object o_i. Each node in N is represented by a d-dimensional vector, denoted as n_o^s, n_a^s, or n_r^s according to the node type. In particular, we use a trainable embedding layer to get the node features. As shown in Figure 2, the edges in E are formulated as follows:

- If an object o_i owns an attribute $a_{i,l}$, we assign a directed edge from $a_{i,l}$ to o_i;
- a relationship triplet $< o_i, r_{ij}, o_j >$ exists, two directed edges will be assigned from o_i to r_{ij} and from r_{ij} to o_j, respectively.

3.2. Image Encoding and Image Scene Graph

Given an input image, we first use a Faster-RCNN model [32] pretrained on MS-COCO [24] to get a number of object proposals and corresponding ROI features f_{roi}. Global average pooling is applied on the feature map to get a global image feature f_{global}.

To generate the *image scene graph* (ISG), we borrow the visual scene graph detector from [50] to predict the relationships between the object proposals. Another classifier pre-trained on Visual Genome [19] is used for predicting

[1]There are two settings: image-to-text or text-to-image matching. In each setting, one modality serves as the query and the other plays the role of reference data. Such asymmetry significantly affects the design of optimization objective, as later described in Section 3.7 and experiments.

the attributes of each object proposal. Similar to G_S, there are also three kinds of nodes in the ISG G_I. However, the object nodes n_o^I in G_I take the ROI features as the node features, while the relation nodes n_r^I and attribute nodes n_a^I take the class-label embeddings [46] as the original features.

3.3. Graph Convolutional Networks (GCNs)

To effectively explore the high-order information in G_S and G_I, GCNs are adopted to merge the local information from each node and its neighbors into a new set of context-aware embeddings at some d-dimensional feature space. We hereafter use variable x (with proper index) to denote such embedding vectors. There are three types of embeddings: relation embedding $x_{r_{ij}}$ for relation node r_{ij} , object embedding x_{o_i} for object node o_i, and attribute embedding x_{a_i} for attribute node a_i, respectively. Inspired by the previous work [46], we propose four spatial graph convolutions: g_r, g_a, g_s, and g_o to generate above-mentioned embeddings. In practice, these four functions are implemented as multi-layer perception (MLP) with same network architecture but independent parameters. For brevity, we only elaborate on SSG. The derivation of ISG is similar.

For each r_{ij} in SSG, the relation embedding $x_{r_{ij}}$ is processed according to Eqn. (1) as below to jointly represent a relation $< o_i, r_{ij}, o_j >$, shown in green color in Figure 2:

$$x_{r_{ij}} = g_r\left(concat(n^s_{o_i}, n^s_{o_j}, n^s_{r_{ij}})\right). \tag{1}$$

For the attribute nodes, the spacial convolution operation is depicted in blue color in Figure 2. The information of all attribute nodes connected to each object node o_i is merged to get a single contextual feature vector:

$$x_{a_i} = \frac{1}{N_{a_i}} \sum_{l=1}^{N_{a_i}} g_a\left(concat(n^s_{o_i}, n^s_{a_{i,l}})\right), \tag{2}$$

where $\{a_{i,l}\}$ forms o_i's attribute-related neighbor set with a cardinality of N_{a_i}.

To compute object embedding x_{o_i}, we collect information from all nodes that have a relationship with o_i, according to:

$$\begin{aligned} x_{o_i} = \frac{1}{N_{r_i}}[&\sum_{o_j \in sbj(o_i)} g_s(concat(n^s_{o_i}, n^s_{o_j}, n^s_{r_{i,j}})) + \\ &\sum_{o_k \in obj(o_i)} g_o(concat(n^s_{o_k}, n^s_{o_i}, n^s_{r_{k,i}}))], \end{aligned} \tag{3}$$

where $o_j \in sbj(o_i)$ implies that o_j acts as the subject in some relation with the object o_i, $o_k \in obj(o_i)$ implies a role of object. And $N_{r_i} = |sbj(o_i)| + |obj(o_i)|$. This operation is shown in yellow color in Figure 2.

3.4. Cross-Graph Attention

As stated before, it is crucial to calculate the asymmetric correlation between any two cross-graph nodes, either from G_S to G_I or from G_I to G_S. We design a cross heterogeneous graph attention mechanism, shown in Figure 3. For each node x_i^S in G_S, we calculate the cosine similarities with all nodes x_j^I in G_I to get an attention matrix M_{att}, sized $|G_S| \times |G_I|$. Specifically, $M_{att} = [\alpha_{i,j}]$ is calculated as below:

$$\alpha_{i,j} = cosine(x_i^S, x_j^I),\ i \in 1, \ldots, |G_S|, j \in 1, \ldots, |G_I|, \tag{4}$$

where $cosine(\cdot, \cdot)$ returns the cosine value in $[-1, 1]$ for its two inputting vectors.

These similarities are then used as weights to compute the attentive embedding for x_i^S by a weighted sum on all the node embeddings of G_I, namely

$$\tilde{x}_i^S = \frac{\sum_{j=1}^{|G_I|} [\alpha_{i,j}]_+ \cdot x_j^I}{\sum_{j=1}^{|G_I|} [\alpha_{i,j}]_+}, \tag{5}$$

where $[x]_+ \equiv max(x, 0)$. We can swtich the role of G_I, G_S in above procedure, obtaining attentive embedding $\tilde{x}_i^I$ for each x_i^I in G_I.

3.5. Local Graph Matching

Based on the attentive embedding of each node, we proceed to calculate local-matching vectors for all nodes in both G_S, G_I. The goal is to obtain intra-graph contextually-enhanced features from multiple perspective (*i.e.*, semantics and graph structure). For G_S, a multi-perspective cosine matching function is applied on each node x_i^S and its attentive embedding $\tilde{x}_i^S$. Similar treatment for G_I. In particular, local-matching vectors m_i^S and m_j^I are computed as below:

$$m_i^S = f_m(x_i^S, \tilde{x}_i^S; W), \quad m_j^I = f_m(x_j^I, \tilde{x}_j^I; W), \tag{6}$$

where f_m denotes the multi-perspective matching function. Let $W \in \mathbb{R}^{l \times d}$ be a learnable parameter matrix and W_k be the k-th row of W. l is pre-defined to specify how many perspectives are required. Given two d-dimensional vectors v_1 and v_2, the matching vector m is rendered as below:

$$\begin{aligned} &m_k = cosine(W_k \circ v_1, W_k \circ v_2), \quad k \in [1, 2, \ldots, l], \\ &m = [m_1, m_2, \ldots, m_l], \end{aligned} \tag{7}$$

where $\circ$ denotes element-wise product. This implicitly defines f_m.

3.6. Node Attention and Global Matching

Intuitively, different nodes in graphs shall not be equally weighed. Some nodes are more important according to common sense (*e.g.*, human-related objects are often more

likely to be relevant to queries). We thus further design a node attention mechanism to attach a weight β_i to each node x_i in graphs. Take SSG for instance,

$$\beta_i^S = \frac{exp(\phi(h_n, x_i^S))}{\sum_{i=1}^{|G_s|} exp(\phi(h_n, x_i^S))}, \tag{8}$$

where x_i^S is the context embedding in G_S as described in Equations (1)(2)(3). ϕ is a learnable sub-network that reads h_n, x_i^S. Recall that h_n is a global feature vector for the entire sentence. Replacing h_n by f_{global}, x_i^S by x_i^I, and $|G_s|$ by $|G_I|$ in Eqn. (8) leads to a new formula for ISG.

After getting the importance of each node in the graph, a global weighted sum is adopted to fuse all the matching vectors into a global representation:

$$\bar{m}^S = \sum_{i=1}^{|G_S|} \beta_i^S m_i^S, \quad \bar{m}^I = \sum_{j=1}^{|G_I|} \beta_j^I m_j^I. \tag{9}$$

Finally, $\bar{m}^S, \bar{m}^I$ are concatenated together and fed into a MLP followed by a sigmoid function to predict the matching similarity, denoted by $score_m$.

3.7. Distraction Based Re-Ranking

We regard that a good match should simultaneously satisfy two conditions: 1) the matched reference contains all key semantics of the query (*i.e.*, *maximal relevance*); 2) the contents in the reference irrelevant to the query should not be dominating, since they distract viewers (*i.e.*, *minimal distraction*). We argue that the second condition is insufficiently explored in previous studies. An example is presented in Figure 4.

This work adopts information entropy to quantify distraction and uses distraction scores to re-rank initial retrieved results. Take text-to-image matching for instance. The computations is purely based on the attention matrix M_{att} described in Section 3.4, For text-to-image matching, we estimate the distraction by asking each node in G_S to vote for each node in G_I. To ensure votes from each node in G_S are equal, we first perform L_1 normalization on the row corresponding to specific node in G_S. Namely $M_{att}(k,\cdot) \leftarrow M_{att}(k,\cdot)/\sum_j M_{att}(k,j)$. Next, by column sum we obtain how many votes each node in G_I receives from all nodes in G_S, termed as distraction vector $v_{dist} \in \mathbb{R}^{|G_I|}$. It is formally computed via $v_{dist}(j) = \sum_i M_{att}(i,j)$. We conduct L_1 normalization to ensure v_{dist} forms a valid probabilistic distribution. Finally, the distraction score, denoted by $score_d$, is computed by information entropy:

$$score_d = -\sum_{i=1}^{|G_I|} (v_{dist,i} + \epsilon) \cdot \log\left(v_{dist,i} + \epsilon\right), \tag{10}$$

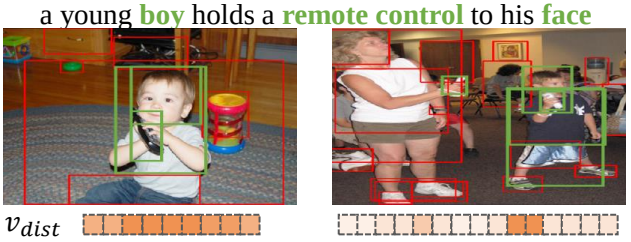

Figure 4. Illustration of the motivation of distraction based re-ranking. The left image is the ground-truth match to querying sentence. Green boxes indicate relevant objects, and red boxes indicate unrelated (*i.e.*, distracting) objects. v_{dist} is defined in Eqn. (10). Darker color in v_{dist} implies higher values.

where ϵ is a tiny constant introduced for numeric stability. For true matches, there are few distractions. Therefore most entries in its v_{dist} are large (see the left image in Figure 4). For false matches, v_{dist} tends to be sparse, with most zeros attribute to the distracting objects or relations. It is known that sparse distribution leads to smaller information entropy.

In practice, we let $score_f = score_m + \gamma \cdot score_d$ be a fused score for re-ranking initial results. γ is set to $4 * 10^{-3}$ in our experiments. For efficiency, we only calculate the distraction scores for top-10 results in the inference time. Similar derivation for image-to-text matching is omitted due to space limit.

3.8. Loss Function of Joint Learning

For image-to-text or text-to-image matching, we use a composite loss function with two components. One is the triplet loss L_t [34] to maximize the margin of positive sample and negative samples. The other binary cross entropy loss L_{ce} aims to effectively decrease the negative samples' scores. Definitions are:

$$\begin{aligned} L_t &= \sum_i [\, score'_{m,i} - score_m + \delta \,]_+ \,, \\ L_{ce} &= \sum_i y_i log(score_{m,i}) + (1 - y_i) log(1 - score_{m,i}) \,, \end{aligned} \tag{11}$$

where the δ is a margin hyper-parameter. $score'_{m,i}$ denotes the i-th negative sample's score and it's label y_i is set to 0, while $score_m$ is the positive sample score and it's label is set to 1. The entire loss is $\lambda_1 L_t + \lambda_2 L_{ce}$, where λ_1 and λ_2 are two hyper-parameters.

4. Experiments

4.1. Datasets

For experiments, following the previous work [20], we selected two large widely-used datasets:

MS-COCO [24] is a large-scale dataset which contains 123,287 images, and each image in it is annotated with five text descriptions. We follow [14] to prepare the training,

Methods	MS-COCO 1K						MS-COCO 5K					
	Sentence Retrieval			Image Retrieval			Sentence Retrieval			Image Retrieval		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
DVSA [14]	38.4	69.9	80.5	27.4	60.2	74.8	11.8	32.5	45.4	8.9	24.9	36.3
VQA-ICR [25]	50.5	80.1	89.7	37.0	70.9	82.9	23.5	50.7	63.6	16.7	40.5	53.8
DSPE [40]	50.1	79.7	89.2	39.6	75.2	86.9	-	-	-	-	-	-
VSE++ [4]	64.6	90.0	95.7	52.0	84.3	92.0	41.3	71.1	81.2	30.3	59.4	72.4
TBNN [39]	54.0	84.0	91.2	43.3	76.8	87.6	-	-	-	-	-	-
DPC [51]	65.6	89.8	95.5	47.1	79.9	90.0	41.2	70.5	81.1	25.3	53.4	66.4
DXN [7]	68.5	-	97.9	56.6	-	94.5	42.0	-	84.7	31.7	-	74.6
SCO [10]	69.9	92.9	97.5	56.7	87.5	94.8	42.8	72.3	83.0	33.1	62.9	75.5
SCAN [20]	72.7	94.8	98.4	58.8	88.4	94.8	50.4	**82.2**	**90.0**	38.6	69.3	80.4
SAEM [42]	71.2	94.1	97.7	57.8	88.6	94.9	-	-	-	-	-	-
VSRN [22]	76.2	94.8	98.2	62.8	89.7	95.1	**53.0**	81.1	89.4	**40.5**	70.6	81.1
Ours	77.0	**96.1**	**98.7**	65.1	**93.1**	**97.9**	51.2	81.7	89.1	39.4	72.5	**84.1**
+Dist	**77.8**	**96.1**	**98.7**	**66.2**	93.0	**97.9**	51.4	81.8	89.1	**40.5**	**73.5**	**84.1**

Table 1. Results on MS-COCO 1K and 5K. The sentence retrieval and image retrieval utilize images and sentences as queries respectively.

validation and testing set. All images are split into three parts which contain 113,287, 5,000, 5,000 samples respectively. For the evaluation of MS-COCO 5K setting, we used all these 5K testing images. We also used 1/5 of these testing samples as MS-COCO 1K setting for ablation study and compare with some algorithms which report their results only on MS-COCO 1K dataset.

Flickr30K [48] is a dataset that contains 31,783 images collected from the Flickr website also with five captions each. These 158,915 descriptions generated by web users are about events, activities, and scenes in the images. We followed the split in [4] and [14] that used 1,000 images for testing and 1,000 images for validation. The rest (28783 images) are used for training.

4.2. Evaluation Metrics

As a common practice in information retrieval, we measure the performance of sentence retrieval (image query) and image retrieval (sentence query) by recall at top-K (R@K), which is defined as the fraction of queries for which the correct item is retrieved in the closest K points to the query. Take the image retrieval as an example, given a query sentence, we rank the similarity scores of all the images in the test set and select top-K candidates. We regard the query as a "successful" one if the ground-truth image is in these candidates. And the R@K is the proportion of these successful queries in the whole test set. K is set to 1, 5 and 10 in all experiments.

4.3. Implementation Details

In all of our experiments, the hidden units of the LSTM are set to 1024 to encode the text semantic information. VGG16 [36] pre-trained on ImageNet [33] is used as the image feature extractor for the Faster-RCNN [32] and to get the global image feature. The dimension of all node embeddings in the scene graphs is set to 256. The perspective number l in Eqn. (7) is set to 64.

Since the visual scene graph detector may introduce some noise in the high-order information, we further add a first-order branch. Specifically, we remove the GCN module, and pool the cross attention matrix produced by the context vectors h_i and ROI features f_{roi} in a LogSumExp [20] manner to obtain a first-order part $score_m^1$. This is fused with the high-order $score_m^2$ which is detailed in previous section to get the final matching similarity $score_m = score_m^2 + 0.1\ score_m^1$.

We implement all models using the Pytorch framework. For the loss function, we set the hyper-parameters $\lambda_1 = 1.0$, $\lambda_2 = 0.5$, and $\delta = 1.0$. For each positive sample in the triplet loss, ten randomly selected samples are used to form the negative pairs. Adam optimizer [15] with default setting (learning rate=10^{-3}, momentum=0.9, weight-decay=10^{-4}) is applied to tune the model parameters. The learning rate attenuates by 1/10 for every 25 epochs. We adopt early stop strategy to avoid over-fitting.

4.4. Comparison with State-of-the-Art Models

In this section, we first present our quantitative results and comparison with other state-of-the-art methods on the MS-COCO dataset in Table 1. To make a more comprehensive comparison, we used two scales test sets (1K and 5K) on MS-COCO. It can be seen that our model exceeds the current best method VSRN [22] by most of the metrics on both image retrieval and sentence retrieval tasks. In particular, in the 1K setting, compared to the previous method our model has a significant improvement in the image retrieval scenario (by 2.3 on R@1 and 3.4 on R@5).

We also conducted the same experiments on the Flickr30K dataset, Table 2 shows the comparison results on the testing set. Obviously, we can find that our method still dominates other methods under most of the evaluation metrics, which strongly indicates the superiority of the proposed method.

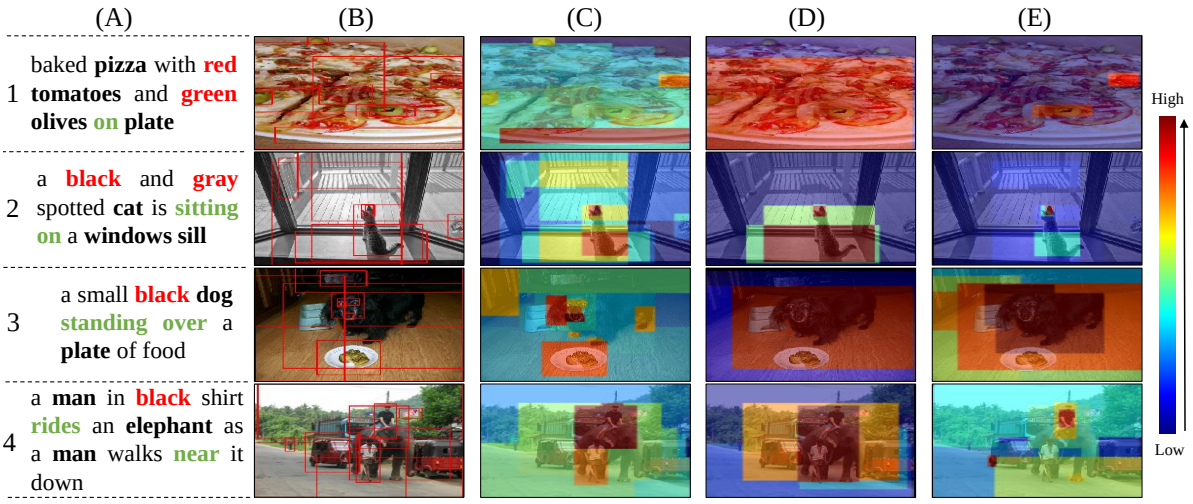

Figure 5. Visualization the cross graph attention mechanism. Each row is an example, and the column (A) shows the query sentences. Part of object proposals are shown in column (B). And columns (C), (D), and (E) show the attention results for the object, relation, and attribute nodes, respectively. The color of the mask reflects the attention value of the area corresponding to the node. The warmer red color represents greater attentive response. Best viewed in color.

Methods	Flickr30K					
	Sentence Retrieval			Image Retrieval		
	R@1	R@5	R@10	R@1	R@5	R@10
DVSA	22.2	48.2	61.4	15.2	37.7	50.5
VQA-ICR	33.9	62.5	74.5	24.9	52.6	64.8
DSPE	40.3	68.9	79.9	29.7	60.1	72.1
VSE++	41.3	69.0	77.9	31.4	59.7	71.2
TBNN	37.5	64.7	75.0	28.4	56.3	67.4
DPC	55.6	81.9	89.5	39.1	69.2	80.9
DXN	56.8	-	89.6	41.5	-	80.1
SCO	55.5	82.0	89.3	41.1	70.5	80.1
SCAN	67.4	90.3	95.8	48.6	77.7	85.2
SAEM	69.1	91.0	95.1	52.4	81.1	88.1
VSRN	**71.3**	90.6	96.0	54.7	81.8	88.2
Ours	70.8	**92.7**	**96.0**	59.5	85.6	**91.0**
+Dist	70.8	**92.7**	**96.0**	**60.9**	**86.1**	**91.0**

Table 2. Experimental results on Flickr30K.

In addition, we also applied the distraction based re-ranking strategy on the initial top-10 results, which is reported at the bottom of Tables 1 and 2 (denoted by '+Dist'). It is clear that the re-ranking strategy further improves the performance in most settings and metrics especially the R@1. In the image retrieval scenario, this brings about a 2.8% relative performance boost in the MS-COCO 5K test set on R@1. These results strongly demonstrate the effectiveness of the proposed distraction strategy. While the promotion on the sentence retrieval task is not so impressive, we think this is due to the node number in sentence graphs is relatively small, the entropy-based re-ranking strategy cannot play a big role in this situation. We have also tried to consider the region size of the object proposals as weights when calculating the distraction score, but the final improvement is trivial.

Methods	MS-COCO 1K					
	Sentence Retrieval			Image Retrieval		
	R@1	R@5	R@10	R@1	R@5	R@10
o	70.9	93.6	97.3	52.3	84.3	92.5
or	74.1	93.2	96.9	61.8	91.5	96.0
ora	**77.0**	**96.1**	**98.7**	**65.1**	**93.1**	**97.9**
$-crossatt$	46.9	91.1	97.9	47.5	84.3	93.9
$-nodeatt$	56.7	89.0	95.8	51.3	87.1	95.2

Table 3. Ablation studies on the MS-COCO 1K test set.

4.5. Ablation Studies

To explore the effect of high-order information (relationships and attributes) on visual semantic matching performance, we performed related ablation experiments. The quantitative results on MS-COCO 1K test set are shown in Table 3. First, we removed the relation and attribute information in the data, only used the object information when training and testing. This is denoted by 'o' in Table 3. On the basis of this, we added the relation information to the data, and the result is recorded as 'or'. Finally, we contained the complete object, relation and attribute information and showed the result on the row noted as 'ora'. It can be clearly noticed that with the addition of more information, the performance of the model has steadily improved in each indicator, which confirms the validity of the relation and attribute information in the visual semantic matching task.

We also explored the impact of the two attention mechanisms proposed above. We removed the node attention (noted as $-nodeatt$) and the cross graph attention (noted as $-crossatt$) separately and used mean pooling instead. The bottom part of Table 3 shows the results. It can be seen that the performance of the model has dropped signifi-

Figure 6. Demonstrate the effect of distraction based re-ranking. The upper part shows the image retrieval results while the bottom part shows the sentence retrieval results. The original ranking is shown in the upper right corner of each image and behind the sentences respectively. Green boxes indicate ground truth images and the red sentences represent negative samples.

cantly without the attention mechanism. For example, without the cross graph attention, the R@1 result of the image retrieval is reduced by 17.6. After removing the node attention mechanism, the performance on the sentence retrieval is attenuated by 26.4% relatively on R@1. This illustrates the essentiality of the proposed attention mechanisms.

4.6. Visualization and Analysis

To further demonstrate the interpretability of our model, we selected several examples in Figure 5 to visualize the cross graph attention components learned in our model. Given an attention matrix, we first applied a binary mask to get three sub-attention matrices for object, relation, and attribute nodes. Similar to the operation in Section 3.7, we first normalized the contribution of each image node on row, and then got the attention value of each image node by column summation. We assigned this value to the region corresponding to this node to get a colorful mask. For the object and attribute node, their corresponding region is the area of object proposal. The corresponding area of the relation node is the union of two related object areas. For the overlapping part, we take the maximum value. The warmer red color in the mask reflects larger attention response value.

In column (A) of Figure 5, we first showed the three kinds of nodes in SSG in bold fonts of different colors. Column (B) shows some of the object proposals extracted by Faster-RCNN. The next three columns show the attention effects of the object, relation, and attribute nodes, respectively. From the results, we can see that our model accurately detects the areas corresponding to the nodes in image that aligned to the sentence nodes. Take the result in the first line as an example, the warm color in 1 (C) reflects the object information of tomatoes, olives, and plate. The red area in 4 (D) contains the information of "ride" and "near" mentioned in the sentence, which shows that our network can effectively extract relevant relation information. For the attribute nodes, the regions with highest attention value in 3 (E) and 4 (E) indicate the "black" attribute of dog and shirt. These examples strongly illustrate that our cross-attention module has learned interpretable alignments between sentence nodes in G_S and image nodes in G_I.

4.7. Distraction Based Re-ranking

To demonstrate the effectiveness of our semantic distracting based re-ranking strategy, we showed several results corrected by the post-processing in Figure 6. Due to the limited space, only the top-3 images and top-5 sentences are presented, the original retrieval ranking is also provided. As shown, the distraction score effectively lowered the ranking of false matching samples. Take the first one for example, in the original top-1, the queried car only accounts for a tiny part of the picture, while the irrelevant bus station and buildings are dominant. In the second sample, the mentioned dolls appears in the top false matching but a storybook accounts for a larger part. As for the sentence retrieval examples, the "bicycles" and "decorations" do not appear in the query images, so the ranking of negative sentences are lowered.

5. Conclusion

We explored and confirmed the importance of higher-order information (relationships and attributes) and distraction-based re-ranking in the visual semantic matching task. Ablation and visualization experiments both confirmed the rationality and interpretability of our model design. **Acknowledgement**: This work is supported by National Key R&D Program of China (2018AAA0100702), Beijing Natural Science Foundation (Z190001) and National Natural Science Foundation of China (61772037).

References

[1] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Spice: Semantic propositional image caption evaluation. In *ECCV*, 2016. 3

[2] Remco Dijkman, Marlon Dumas, and Luciano García-Bañuelos. Graph matching algorithms for business process model similarity search. In *BPM*, 2009. 2

[3] Aviv Eisenschtat and Lior Wolf. Linking image and text with 2-way nets. In *CVPR*, 2017. 2

[4] Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler. Vse++: Improved visual-semantic embeddings. *arXiv preprint arXiv:1707.05612*, 2(7):8, 2017. 2, 6

[5] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In *NIPS*, 2013. 1

[6] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 2

[7] Jiuxiang Gu, Jianfei Cai, Shafiq R Joty, Li Niu, and Gang Wang. Look, imagine and match: Improving textual-visual cross-modal retrieval with generative models. In *CVPR*, 2018. 2, 6

[8] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 3

[9] Yan Huang, Wei Wang, and Liang Wang. Instance-aware image and sentence matching with selective multimodal lstm. In *CVPR*, 2017. 2

[10] Yan Huang, Qi Wu, Chunfeng Song, and Liang Wang. Learning semantic concepts and order for image and sentence matching. In *CVPR*, 2018. 1, 2, 6

[11] Xinyang Jiang, Fei Wu, Xi Li, Zhou Zhao, Weiming Lu, Siliang Tang, and Yueting Zhuang. Deep compositional cross-modal learning to rank via local-global alignment. In *ACM Multimedia*, 2015. 2

[12] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *CVPR*, 2018. 2

[13] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *CVPR*, 2015. 2

[14] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015. 1, 2, 5, 6

[15] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[16] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017. 2

[17] Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*, 2014. 1, 2

[18] Benjamin Klein, Guy Lev, Gil Sadeh, and Lior Wolf. Associating neural word embeddings with deep image representations using fisher vectors. In *CVPR*, 2015. 2

[19] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017. 3

[20] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching. In *ECCV*, 2018. 1, 2, 5, 6

[21] Guy Lev, Gil Sadeh, Benjamin Klein, and Lior Wolf. Rnn fisher vectors for action recognition and image annotation. In *ECCV*, 2016. 2

[22] Kunpeng Li, Yulun Zhang, Kai Li, Yuanyuan Li, and Yun Fu. Visual semantic reasoning for image-text matching. In *ICCV*, 2019. 6

[23] Yujia Li, Chenjie Gu, Thomas Dullien, Oriol Vinyals, and Pushmeet Kohli. Graph matching networks for learning the similarity of graph structured objects. *arXiv preprint arXiv:1904.12787*, 2019. 2

[24] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 3, 5

[25] Xiao Lin and Devi Parikh. Leveraging visual question answering for image-caption ranking. In *ECCV*, 2016. 6

[26] Anan Liu, Zhongyang Wang, Weizhi Nie, and Yuting Su. Graph-based characteristic view set extraction and matching for 3d model retrieval. *Information Sciences*, 320:429–442, 2015. 2

[27] Diego Marcheggiani and Ivan Titov. Encoding sentences with graph convolutional networks for semantic role labeling. In *EMNLP*, 2017. 2

[28] Giannis Nikolentzos, Polykarpos Meladianos, and Michalis Vazirgiannis. Matching node embeddings for graph similarity. In *AAAI*, 2017. 2

[29] Zhenxing Niu, Mo Zhou, Le Wang, Xinbo Gao, and Gang Hua. Hierarchical multimodal lstm for dense visual-semantic embedding. In *ICCV*, 2017. 2

[30] Yuxin Peng and Jinwei Qi. Cm-gans: Cross-modal generative adversarial networks for common representation learning. *TOMCCAP*, 15(1):22:1–22:24, 2019. 2

[31] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*, 2015. 2

[32] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 3, 6

[33] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 6

[34] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015. 5

[35] Botian Shi, Lei Ji, Pan Lu, Zhendong Niu, and Nan Duan. Knowledge aware semantic concept expansion for image-text matching. In *IJCAI*, 2019. 1, 2

[36] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 6

[37] Ivan Vendrov, Ryan Kiros, Sanja Fidler, and Raquel Urtasun. Order-embeddings of images and language. In *ICLR*, 2016. 2

[38] Huan Wang, Song Liu, and Liang-Tien Chia. Does ontology help in image retrieval?: a comparison between keyword, text ontology and multi-modality ontology approaches. In *ACM Multimedia*, 2006. 2

[39] Liwei Wang, Yin Li, Jing Huang, and Svetlana Lazebnik. Learning two-branch neural networks for image-text matching tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2):394–407, 2018. 2, 6

[40] Liwei Wang, Yin Li, and Svetlana Lazebnik. Learning deep structure-preserving image-text embeddings. In *CVPR*, 2016. 2, 6

[41] Runzhong Wang, Junchi Yan, and Xiaokang Yang. Learning combinatorial embedding networks for deep graph matching. *arXiv preprint arXiv:1904.00597*, 2019. 2

[42] Yiling Wu, Shuhui Wang, Guoli Song, and Qingming Huang. Learning fragment self-attention embeddings for image-text matching. In *ACM Multimedia*, 2019. 6

[43] Liang Xie, Jialie Shen, and Lei Zhu. Online cross-modal hashing for web image retrieval. In *AAAI*, 2016. 1

[44] Kun Xu, Liwei Wang, Mo Yu, Yansong Feng, Yan Song, Zhiguo Wang, and Dong Yu. Cross-lingual knowledge graph alignment via graph matching neural network. In *ACL*, 2019. 2

[45] Xifeng Yan, Philip S Yu, and Jiawei Han. Substructure similarity search in graph databases. In *SIGMOD*, 2005. 2

[46] Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. Auto-encoding scene graphs for image captioning. In *CVPR*, 2019. 2, 3, 4

[47] Mang Ye, Andy J Ma, Liang Zheng, Jiawei Li, and Pong C Yuen. Dynamic label graph matching for unsupervised video re-identification. In *ICCV*, 2017. 2

[48] Peter Young, Alice Lai, Micah Hodosh, and Julia 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. 6

[49] Andrei Zanfir and Cristian Sminchisescu. Deep learning of graph matching. In *CVPR*, 2018. 2

[50] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *CVPR*, 2018. 3

[51] Zhedong Zheng, Liang Zheng, Michael Garrett, Yi Yang, and Yi-Dong Shen. Dual-path convolutional image-text embedding with instance loss. *arXiv preprint arXiv:1711.05535*, 2017. 2, 6