Knowledge-Based Systems 263 (2023) 110280

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Generative label fused network for image–text matching

Guoshuai Zhao [a,*], Chaofeng Zhang [a], Heng Shang [a], Yaxiong Wang [b], Li Zhu [a], Xueming Qian [b]

[a] School of Software Engineering, Xi'an Jiaotong University, Xi'an 710049, China
[b] Key Laboratory for Intelligent Networks and Network Security, Ministry of Education, and the SMILES LAB, Xi'an Jiaotong University, Xi'an 710049, China

ARTICLE INFO

Article history:
Received 15 August 2022
Received in revised form 5 January 2023
Accepted 5 January 2023
Available online 10 January 2023

Keywords:
Image–text matching
Cross-modal retrieval
Cross-domain
Feature fusion

ABSTRACT

Although there is a long line of research on bidirectional image–text matching, the problem remains a challenge due to the well-known semantic gap between visual and textual modalities. Popular solutions usually first detect the objects and then find the association between visual objects and the textual words to estimate the relevance; however, these methods only focus on the visual object features while ignoring the semantic attributions of the detected regions, which is an important clue in terms of bridging the semantic gap. To remedy this issue, we propose a generative multiattribution tag fusion method that further includes region attribution to alleviate the semantic gap. In particular, our method comprises three steps: the extraction of image features, the extraction of text features, and the matching of image and text by an attention mechanism. We first divide the image into blocks to obtain the region image features and region attribute labels. Then, we fuse them to reduce the semantic gap between the image features and text features. Second, BERT and bi-GRU are used to extract text features, and third, we use the attention mechanism to match each area in the image with each word in the text with the same meaning. The quantitative and qualitative results on the public datasets Flickr30K and MS-COCO demonstrate the effectiveness of our method. The source code is released on Github https://github.com/smileslabsh/Generative-Label-Fused-Network.

1. Introduction

In the last decade, with the development of the internet and information technology, the information on the internet has changed changing from text-based single-mode data to multimodal data information composed of text, picture, video, audio, and data of other modes. These different modalities are often used to describe the same object, the same event, or the same subject. In the face of these huge and interrelated multimedia data, users urgently need to be able to use one of the modalities (such as text) to simultaneously retrieve the results of other related modalities (such as images), namely, cross-modal retrieval. Among them, image and text are the two most commonly used modalities. Consequently, image–text matching attracts much attention and is an important direction in the future development of information retrieval. At the same time, major internet companies (especially search engine companies) are also trying to provide better image and text search services for users. Therefore, cross-modal image–text matching has a wide range of application scenarios and research significance.

The goal of cross-modal image–text matching is to retrieve the text that best describes the image for a given image or to retrieve the image describing a given piece of text. The main difficulty in cross-modal image–text matching is that the features of different modalities are in different feature spaces, and they are heterogeneous at the bottom data structure and semantically related at the top semantics. For example, text encoding features and image encoding features that represent the same topic are in completely different feature spaces. Even though they both represent the same topic, their feature vectors are completely different.

Most of the existing methods of cross-modal image–text matching extract features of different modalities, such as text and images. Then, text features and image features are associated in different feature spaces through mapping, attention, and other methods. Finally, the similarity between text and image is estimated to perform cross-modal image–text retrieval. In the process of measuring the similarity between the image and the text, an increasing number of models have noticed that the global similarity between the image and the sentence largely depends

* Corresponding author.
E-mail addresses: guoshuai.zhao@xjtu.edu.cn (G. Zhao), surmount@stu.xjtu.edu.cn (C. Zhang), shangheng@stu.xjtu.edu.cn (H. Shang), wangyx15@stu.xjtu.edu.cn (Y. Wang), zhuli@mail.xjtu.edu.cn (L. Zhu), qianxm@mail.xjtu.edu.cn (X. Qian).

https://doi.org/10.1016/j.knosys.2023.110280
0950-7051/

on the local similarity between the object in the image and the word in the sentence. When people use a sentence to describe images, it is natural to describe the objects and actions in the pictures by using the corresponding word, and when retrieving images, users often expect the results to include the objects corresponding to the words in the sentence. Inspired by this phenomenon, the models used a variety of different methods to measure these local similarities. For example, Karpathy et al. [1] used the statistical method to infer the relationship between the regions in the image and the words. Lee et al. [2] used the attention mechanism to design the embedded network, which can capture the correspondence between the region in the image and the word in the sentence. Many researchers have proven that dividing images into regions and using region features can better contact text words [2–4]. It helps us to match the features of different modalities.

However, we note that the existing methods only focus on the visual features of the image region but ignore the attribute features and category features of these image regions. For image–text matching, the original input is only an image. If a model can detect the object in the image, it can not only obtain the visual features of the target but also obtain its properties and categories. The attributes and categories of these objects are complementary descriptions of visual features, and these attributes and categories are naturally expressed in text. If we can fuse this information into visual features, on the one hand, we can enhance the feature representation of these areas; on the other hand, these fused tag features can reduce the semantic gap with the text modal.

Therefore, based on the above analysis, we propose a GLFN (generative label fused network) model. On the one hand, the GLFN can fuse the image features and attribute and category features obtained from the image, and on the other hand, we use the pretraining model [5–7] to obtain the representation vector of different modalities. This model divides a picture into multiple areas and can describe the image in more detail. At the same time, since the pretraining model uses supervision data with labels in the training stage, we can obtain the attributes and category information while using the pretraining Faster R-CNN to obtain the features of each image area. BERT is then used to represent the attributes and category information as vectors and splice them with image feature vectors. Finally, we fuse generative tag features, region features and position features for image representation. The position feature is proposed in PFAN [3], which is our baseline model. By splitting the images into blocks, we can infer the relative position of the region in the image, and then, an attention mechanism is proposed to model the relations between the image region and blocks and generate valuable position features. Therefore, the motivation of the position feature is to make full use of the position information of the object in the image to improve the performance of the image retrieval model. The position feature allows the model to measure the importance of the object region based on the positional cues, thereby focusing on the salient regions in the image. For the text modality, we use BERT [8] and bi-GRU [9] to extract the text vector representation. Pretrained BERT is used to obtain the generic representation of each word in the text, and bi-GRU is used to further adjust the word vector of each word. Experiments show that this scheme has better effects than BERT or bi-GRU alone. Finally, we focus on finding the fine-grained interaction between the objects and words and estimate the relevance for the image–text pair.

To effectively assess the effectiveness of our proposed approach, we perform experiments on two commonly public datasets, Flickr30k [10] and MS-COCO [11]. On the Flickr30K dataset, our results reached 75.1 on top1 Recall, a 7.3% increase from 70.0 to 75.1 on baseline (PFAN [3]), and compared to SCAN [2], we obtained an improvement of 11.4%. On the MS-COCO dataset, we achieved a result of 78.4 on top 1 recall, which is an improvement of 1.9 over PFAN.

Our contribution can be summarized as follows:

- We propose a generative tag feature fusion method for image–text matching. The generative object tag can reduce the semantic gap between image and text because it is a mixture that bridges image object features and textual features.
- We fuse generative tag features, region features and position features for image representation. Furthermore, position attention combining generative tag features and region features is utilized to enhance the representation.
- We propose a method combining BERT and bi-GRU to represent text features. The overfitting problem caused by using two models at the same time is relieved by properly using one fully connected layer. Then, we utilize visual–textual attention to calculate the final similarity score. The experimental results demonstrate the effectiveness of our model.

2. Related work

At present, cross-modal image–text matching [12–18] has been researched extensively and in depth. The mainstream method of cross-modal image–text matching is the common subspace method; the premise of common subspace learning is to assume that data of different modalities have the same semantic distribution, so the data of different modalities with the same high-level semantics have a potential correlation in the semantic space. The main methods include the traditional method based on statistical correlation analysis, the DNN, cross-media graph regularization, measurement learning, and sorting.

2.1. Statistical correlation analysis methods

The first subspace learning method originated in statistics. The canonical-correlation analysis (CCA) proposed by Hotelling et al. [19] and Hardoon et al. [20] was the most famous method in subspace learning. This method was used to learn a common subspace for two sets of data, which can maximize the paired correlation between two sets of heterogeneous data. Rasiwasia et al. [21] first applied the CCA method to cross-modal retrieval. However, the CCA approach was unsupervised and did not use semantic tags, so many researchers have tried to extend the CCA approach. Pereira et al. [22] combined CCA and semantic tag categories which verifies the validity of semantic tags, and Gong et al. [23] and Ranjan et al. [24] considered cross-modal data with multiple labels. Sharma et al. [25] extended unsupervised CCA to generalized multiperspective discriminant analysis, making the projection of similar samples in the potential subspace as close as possible and the projection of nonsimilar samples as separate as possible. In addition to CCA, Sharma et al. [26] used partial least squares (PLS) to map features of different modalities to the common subspace, and Li et al. [27] used a cross-modal factor analysis method to measure and evaluate the similarity between two modalities. Mahadevan et al. [28] proposed the maximum covariance expansion (MCU) and used the manifold learning idea to reduce the dimension of high-dimensional data of different modalities. Wu et al. [29] proposed a Cross-Modal Online Low-Rank Similarity function learning (CMOLRS) method and used a low-rank bilinear similarity measurement to capture the multilevel semantic correlations among cross-modal data.

2.2. Method based on the DNN

With the great progress of deep learning in recent years, deep neural networks have made great breakthroughs in different

multimedia fields. Deep learning has a strong nonlinear learning ability. The basic idea of the cross-modal image–text matching method based on deep learning is to use deep learning's feature extraction ability to extract an effective representation of different modals and establish a semantic correlation between different modalities. For example, Ngiam et al. [30] proposed a cross-modal learning method based on a deep network. The model accounted for multimodal fusion learning, cross-modal learning, and shared representation learning and verified the effectiveness of the method through video and speech recognition. Some scholars have tried to combine deep learning with traditional statistical methods. Andrew et al. [31] and Yan et al. [32] carried out a nonlinear extension of CCA and proposed deep canonical correlation analysis (DCCA). They learned complex nonlinear projections through a multilayer deep network to maximize the correlation of common representations after projection. Feng et al. [33] proposed a deep learning model based on a cross-modal correspondence autoencoder. By minimizing the sum of the reconstruction errors of the single-modal autoencoder and the correlation errors of different modal representation layers, the model integrates single-modal representation learning and correlation learning between modalities into one framework. Some researchers have tried to use multiple automatic coders, such as ICMAE proposed by Zhang et al. [34] and DCCAE proposed by Wang et al. [35], which are based on label information. Wei et al. [36] proposed a method of deep semantic matching. Some researchers have used data-enhancement methods to improve model power. Gu et al. [37] tried to use the GAN [38] to enhance data and used GAN-generated data features to fuse with multimodal features. Some researchers have also tried to align image areas with words in sentences. For example, Karpathy et al. [1] used the R-CNN [39] to detect the regions in the image and then calculated the similarity score between the image regions and word pairs to infer the similarity of the image–text. Some researchers then improved this statistical approach by using attention mechanisms. Nam et al. [40] used a dual attention network to capture fine-grained interactions between different modals, and Ji et al. [41] introduced a Saliency-Guided Attention Network (SAN) that is characterized by building an asymmetrical link between vision and language to efficiently learn a fine-grained crossmodal correlation. Wei et al. [42] extracted semantic features for image regions by a pretrained bottom-up attention model. Zhao et al. [43,44] proposed an effective method to match the emojis and the user comments by using attention mechanism.

The similarity between Ref. [42] and our work is that we both use the attention mechanism. The differences are as follows: (1) Ref. [42] focuses more on the attention mechanism of stacking, whereas our model additionally utilizes the region tag information and position embedding for image representation. For sentence representation, Ref. [42] utilizes the CNN to extract sentence features, while our model leverages the bi-GRU module.

2.3. Methods based on graphs

Graph regularization [45]was a method widely used in semisupervised learning; this method used graphs to describe the data. The nodes of the graph represented every single piece of data and the properties of the nodes as attributes of the data. The edges of the graph were used to represent the relationships between the data, and the attributes of the remaining untagged nodes were predicted from the tagged data of a portion of the graph. Zhai et al. [46] proposed Joint Graph Regularization Isomerization Metric Learning (JGRHML), which combines graph regularization with cross-modal retrieval. Subsequently, this method was further extended to the Joint Representation Learning (JRL) [47] method, which supported more media types for multimodal retrieval. Several works [48,49] extended the graph to a hypergraph and used fine-grained information to get a more accurate result. Using graph regularization was useful for multimodal retrieval, but it often led to excessive time and space complexity during graph building.

In other words, the main purpose of the subspace method is to learn a discriminative shared subspace, and the main method is to maximize correlation. The deep learning method benefits from a large number of training samples and the excellent representation ability of the deep model, and it achieves a better retrieval effect. These methods [50] mainly focus on low-level feature learning and high-level network correlation. A common shortcoming is that they do not consider the local structure of the data. Our method divides the regions on the image and pays attention to both the features of each region and the features of each word in the text. The data's local structure is deconstructed and modeled in more detail. Using the attention mechanism to match each region and word can more effectively model the correlation of different modalities. For image recognition, GCT [51] proposed a lightweight channel-wise attention mechanism, which can generally improve the robustness of DNNs in image classification, detection, and instance segmentation. For video understanding, AOT [52] proposed the long-short term transformer to match visual patches and propagate object information across video sequences. In addition, our method also uses tags, but there is a significant difference from the previous method [22–24,53]. The labels they use are manually marked ahead of time, and these methods can only be applied to premarked data. Our method uses tags that are generated from the image. The generated tags include not only the area's category but also the area's color attribute, and an image contains multiple tags.

3. Our GLFN model

In this section, our model will be elaborated on in detail, and our workflow is displayed in Fig. 1. First, we need to map the text and image into a vector space. We adopt pretrained BERT [8] to encode text into a vector. For an image, we acquire an image region feature vector and tag it through bottom-up attention [7], and then, we combine the region vector and tag to describe the image. Finally, the image vector and the text vector are aligned by visual–textual attention [2]. We first introduce BERT in subsection A. Then, we describe the image and text representations in detail in subsections B and C, respectively. Finally, we present the image–text relevance calculation in subsection D.

3.1. BERT for text representation

BERT, which stands for Bidirectional Encoder Representations from Transformers, is based on transformers, an open-source machine learning model designed by Google.

Transformers were proposed by [54] and are known as a sequence-to-sequence architecture. Sequence-to-sequence (or Seq2Seq) is a neural network that transforms a given sequence of elements, such as the sequence of words in a sentence, into another sequence. It has been widely used in natural language processing applications such as translation. In transformers, every output element is connected to every input element, and the weights between them are dynamically calculated based on their connection. Therefore, the transformers handle any given input with all other words in the sentence rather than processing them one at a time. By looking at all surrounding words, the transformers can better understand the context of the input text.

We adopt pretrained BERT as a language model, and we calculate the word vector using pretrained BERT. The word vector representation calculated by pretrained BERT can better describe the word meaning and more accurately calculate the correlation

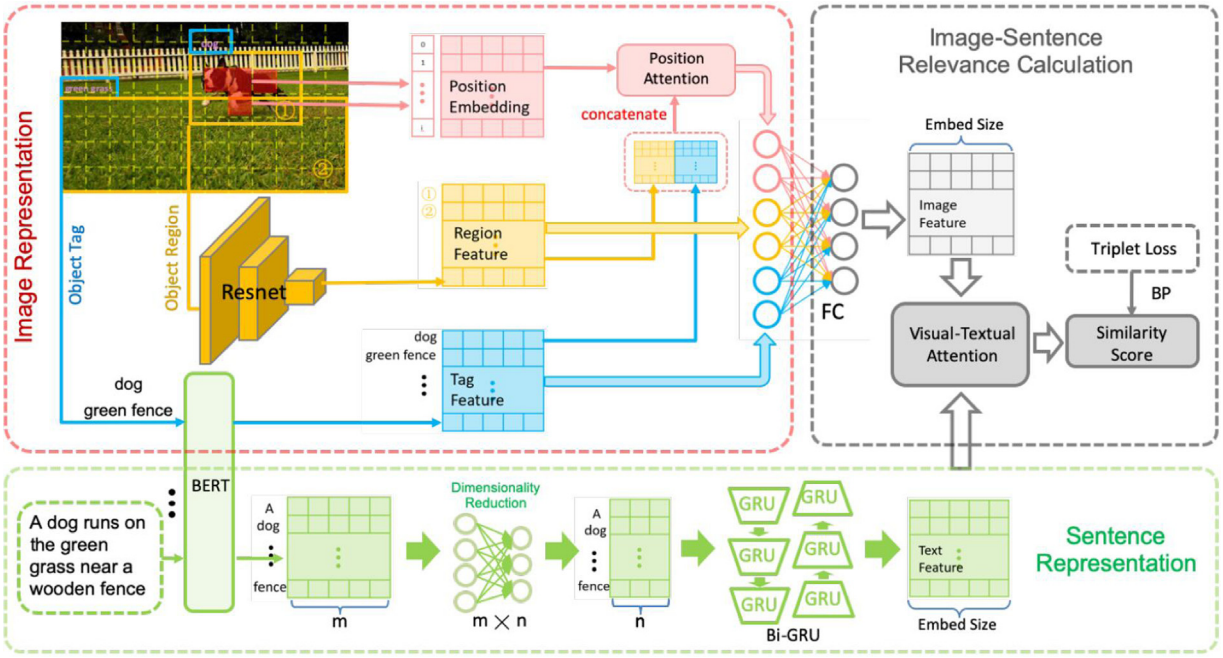

Fig. 1. This workflow shows our model in detail. The final representation of the image features consists of three parts. The red part represents the location-based position attention mechanisms, the yellow part represents the region features of the image extracted by ResNet, and the blue part represents the region tag features. For the text features, BERT is first used to obtain text features. Then, the dimension of the features is reduced through a linear layer. Finally, the features are input into the GRU to obtain the final text features, and the text features and image features are matched one by one using the visual–textual attention mechanism.

between any two words. In our model, both the image section and sentence section use word vector representations calculated by BERT to map them to the same representation spaces to construct a cross-modal shared representation of pictures and texts.

3.2. Image representation

There is a variety of information in an image. In the traditional method, people tend to represent a picture with one vector. However, one image contains many objects and potential relations. For example, in the picture in Fig. 1, there are dogs, fences, and grass. If the picture is described by a caption, the above words are most likely used. If we can detect these objects, the original image can be better represented by combining the features of the object in the image with the features of the original image.

To obtain object features from images, we use a Faster R-CNN model [6]. To obtain a broader category of objects, we feed images into Faster R-CNN pretrained on Visual Genomes [55] by [7]. For each image I, we obtain a set of features $I = \{i_1, \ldots, i_r\}$ and a set of tags $T = \{t_1, \ldots, t_r\}$, where r is the number of image objects. The image object feature i_i is a D_I-dimensional vector, and the tag t_i is a word or phrase. Then, we input the tag t_i into BERT, which was pretrained by [8], to obtain tag feature v_i^t, which is a D_T-dimensional vector. Subsequently, we concatenate the i_i vector and t_i vector as the object feature o_i as follows:

$$O = \{o_1, \ldots, o_i, \ldots, o_r\}, where\ o_i = [i_i, v_i^t] \tag{1}$$

In one image, the position of the object in an image is a very important clue that allows people to understand the image's emphasis. To understand the meaning of an image, we use an effective attention mechanism based on the position of objects in an image proposed by [3].

To mark the object position in the image, we divide the image into $k \times k$ blocks, and each block is initially represented by an index $m \in \left[1, k^2\right]$. Then, we calculate the pixel of overlap in each block and object box as follows:

$$s_{im}^{ob} = |s_i^o \cap s_m^b|, m = 1, 2, \ldots, k^2 \tag{2}$$

where s_i^o is the number of pixels for the ith object box, and s_m^b is the number of pixels for the jth block. s_{im}^{ob} represents the intersecting pixel number between the ith object box and the jth block. For every object box, we select the top L ranked blocks according to the most overlapping pixel as follows:

$$s_{im}^{ob} = \left|s_i^o \cap s_m^b\right|, m = 1, 2, \ldots, L \tag{3}$$

We use the proportion of these overlapping areas in the top L to represent the block weight as follows:

$$W_i^b = \left\{w_{i1}^b, \ldots, w_{im}^b, \ldots, w_{iL}^b\right\}, i \in [1, r], m \in [1, L] \tag{4}$$

where:

$$w_{im}^b = \frac{s_{im}^{ob}}{\sum_{m=1}^{L} s_{im}^{ob}}, i \in [1, r], m \in [1, L] \tag{5}$$

To obtain a more accurate description of the position, we embed the block index into a dense representation. The split blocks B are regarded as the position vocabulary, and each block $b_i \in B$ is represented by the one-hot vector, which indicates the index in the position vocabulary. We next apply an embedding layer to project the one-hot representation into a D_b dimensional vector. For the L blocks, we use L embedding vectors to represent these blocks as follows:

$$v_{im}^b = \{v_{i1}^b, \ldots, v_{im}^b, \ldots, v_{iL}^b\}, i \in [1, r], m \in [1, L] \tag{6}$$

where v_{im}^b is a D_b-dimensional vector. Attention machining can adaptively assign weight to each block as follows:

$$o_i' = f\left(o_i\right) \tag{7}$$

$$att_{im}' = \tanh\left(o_i' \times {v_m^b}^T\right), i \in [1, r], m \in [1, L] \tag{8}$$

$$att_{im} = \frac{exp(att_{im}')}{\sum_m exp\left(att_{im}'\right)} \tag{9}$$

where f is a linear layer that compresses the object vector into the D_b-dimensional vector, and att_{im}' is the weight matrix that

decides how much weight should be given to the block for the ith object vector. Then, we employ a softmax layer to process the att'_{im}.

Next, we calculate the position attention vector as follows:

$$p'_{im} = att_{im} \odot w^b_{im} \tag{10}$$

$$p_{im} = \frac{p'_{im}}{\sum_m p'_{im}} \tag{11}$$

$$p_i = p_{im} \times v^b_{im} \tag{12}$$

where $\odot$ is the Hadamard (elementwise) product, and p_i is the position attention vector.

Subsequently, we concatenate the object vector o_i, position attention vector p_i and tag vector t_i such that the object vector will carry position attention and tag information as follows:

$$\widetilde{v^o_i} = [o_i, p_i, t_i], i \in [1, r] \tag{13}$$

Finally, we use a linear layer to fuse the three features to obtain a D-dimension object vector v^o_i as follows:

$$v^o_i = f(\widetilde{v^o_i}) \tag{14}$$

3.3. Sentence representation

A sentence can be seen as a word sequence with a fixed length. In the traditional method, every word is represented by a one-hot vector because a word is a basic element in the sentence. These same words in different sentences are represented by the same vector. However, a word should have different meanings in a different sentence. For example:

- **Apple** sold fewer iPhones this quarter.
- **Apple** pie is delicious.

The word **Apple** is a company in sentence one, but it is a fruit in sentence two. Using the same vector to represent the word results in an incorrect contextual meaning; therefore, every word in a sentence should be represented as an individual vector, and the vector should be decided by the context of a sentence. Therefore, we use BERT to extract every word in a sentence because BERT uses all words in a sentence to calculate every word individually using an attention mechanism.

We input the whole sentence into BERT, which can export every word vector in a sentence. For a sentence with a length of n, we can represent it as follows:

$$S = \{w^b_1, \ldots, w^b_j, \ldots, w^b_n\}, i \in [1, n] \tag{15}$$

where w^b_j represents a D^b_w-dimensional word vector from BERT. The vector w^b_j is from BERT, which is pretrained on other datasets. Because there are some disparities between datasets, we still need to adjust the vector in a new dataset.

We use the RNN to adjust the vector since it can consider the context of a sentence. Before inputting the vector into the RNN, we first use a full connection to reduce the vector dimension to decrease the number of parameters, considering that too many parameters could lead the RNN to overfit the training dataset. Therefore, we input w^b_j into a fully connected layer to adjust the vector dimension from D^b_w to $\widetilde{D}^b_w$ as follows:

$$w_j = f\left(w^b_j\right), j \in [1, n] \tag{16}$$

where w_j is a $\widetilde{D}^b_w$-dimensional vector that is used to represent the word. We select a bidirectional GRU to process the vector; the bidirectional GRU can process a sequence from front to back and process the sequence from back to front. The forward GRU can be defined by the following set of functions:

$$z_j = \sigma\left(W_z w_j + U_z h_{j-1}\right) \tag{17}$$

$$r_j = \sigma\left(W_r w_j + U_r h_{j-1}\right) \tag{18}$$

$$\widetilde{h}_j = \tanh\left(W w_j + r_j \odot U h_{j-1}\right) \tag{19}$$

$$\overrightarrow{h}_j = z_j \odot h_{j-1} + \left(1 - z_j\right) \odot \widetilde{h}_j \tag{20}$$

$$j \in [1, n] \tag{21}$$

where z_j is the update gate, W and U are weight matrices, σ is the sigmoid function, w_j is the input word vector for time step t, h_{j-1} is the previous $t-1$ unit information, r_j is the reset gate, $\widetilde{h}_j$ represents the current memory content, $\odot$ is the Hadamard (elementwise) product and h_j is the final memory at the current time step.

The backward GRU can be defined by similar functions, which are expressed as follows:

$$z_j = \sigma\left(W_z w_j + U_z h_{j+1}\right) \tag{22}$$

$$r_j = \sigma\left(W_r w_j + U_r h_{j+1}\right) \tag{23}$$

$$\widetilde{h}_j = \tanh\left(W w_j + r_j \odot U h_{j+1}\right) \tag{24}$$

$$\overleftarrow{h}_j = z_j \odot h_{j-1} + \left(1 - z_j\right) \odot \widetilde{h}_j \tag{25}$$

$$j \in [1, n] \tag{26}$$

The final word vector v^w_j is the average of the hidden state $\vec{h}_j$ in the forward GRU and the hidden state h_j in the backward GRU:

$$v^j_i = \frac{(\overrightarrow{h}_j + \overleftarrow{h}_j)}{2}, j \in [1, n] \tag{27}$$

3.4. Image-sentence relevance calculation

To calculate the relevance between images and sentences, we need to map images and sentences into a common embedding space. We have already mapped the image and the sentence into a set of object vectors and a set of word vectors. Then, we use stack cross attention [2] to calculate the similarity between the image and sentence by aligning the object vector and word vector.

stack cross attention divides the calculation of similarity into two parts: text–image stack cross attention and image–text stack cross attention. Text–image stack cross attention calculates the similarity from querying image by text, denoted by $t2i$. In contrast, image–text stack cross attention calculates the similarity from querying text by image, denoted by $i2t$. Among them, the calculation processes of text–image stack cross attention and image–text stack cross attention are similar.

3.4.1. Text–image stack cross attention

The input to stack cross attention is a set of object vectors v^o_i and a set of word vectors v^w_j. We first calculate the similarity matrix between all object vectors and all word vectors as follows:

$$s'_{ij} = v^o_i \times v^{wT}_j, i \in [1, k], j \in [1, n] \tag{28}$$

Then, we obtain a weighted vector, where the weight is the similarity of the word corresponding to all object vectors:

$$w^{t2i}_j = s'_{ij} \times v^o_i \tag{29}$$

Next, we calculate the cosine similarity between the weighted vector representing w_j^{t2i} and the word vector v_j^w in the sentence as follows:

$$s_j^{w-o} = \frac{w_j^{t2i} \cdot v_j^w}{\left\| w_j^{t2i} \right\| \left\| v_j^w \right\|} \tag{30}$$

Finally, the Text–Image similarity s_{the2i} is the average of s_j^{w-o}:

$$s_{t2i} = \frac{\sum_{j=1}^{n} s_j^{w-o}}{n} \tag{31}$$

3.4.2. Image–text stack cross attention

The calculation of the image–text stack cross attention and text–image stack cross attention is similar, and only the order of the images and text are reversed. Finally, the Image–Text similarity s_{i2t} is the average of s_i^{w-o}:

$$s_{i2t} = \frac{\sum_{i=1}^{k} s_i^{w-o}}{k} \tag{32}$$

3.4.3. Final stack cross attention similarity

We obtain the final similarity as a linear combination of text–image stack cross attention and image–text stack cross attention as follows:

$$S_{i2t+t2i} = \alpha S_{i2t} + (1-\alpha) S_{t2i}\ ,\ \ \alpha \in [0,1] \tag{33}$$

$S_{i2t+t2i}$ is the final similarity matrix, which stores the similarity scores between all texts and images and arranges them in order of similarity scores from high to low.

3.5. Loss function

We employ a common ranking objective function, which is Triplet Loss, as our loss function. Triplet loss was first used in face recognition [56], and it could be used to learn good embeddings. In Text–Image Matching, the goal of the triplet loss is to ensure that a pair of similar images and text have their embedding features close together in the embedding space and that a pair of different images and text have their embedding features far away. In our model, we employ the hardest negatives in the mini-batch following [57]:

$$L = max_i[\beta - s_{ii} + s_{ij}]_+ + max_j[\beta - s_{jj} + s_{ji}]_+ \tag{34}$$

where β is a margin parameter, $[x]_+ = \max(x, 0)$, and s_{ii} is the similarity score between the matched ith image and ith sentence. s_{ij} is the similarity score between the mismatched ith image and jth sentence, and s_{ji} is the opposite. When i and j are the same, s_{ii} or s_{jj} is a positive sample, and when i and j are different, s_{ji} or s_{ij} is a negative sample.

4. Experiment

In this section, to demonstrate the effectiveness of our proposed method, we carry out extensive experiments on two public datasets. Compared with existing methods, the results prove the effectiveness of our method. We also conduct an ablation study and provide some discussions to incrementally verify our method.

4.1. Dataset

Flickr30K. Flickr30k [10] is a publicly available collection of sentence-based image descriptions. The dataset contains 31783 images and 158915 English sentences, and there are five sentences for each picture. In addition, the dataset contains 244k coreference chains and 276k manually annotated bounding boxes. It is widely used for evaluating cross-model retrieval. Similar to [1–3,57], we split the Flickr30k dataset into 1000 images for validation, 1000 images for testing and the remaining images for training.

MS-COCO. MS-COCO [11] is a large-scale object detection, segmentation, and caption dataset that contains 113682 images. There are also five sentences for each picture. MS-COCO defines 91 classes, but only 80 classes are used for data. The panorama annotation defines 200 classes, but only 133 classes are used. Similar to [1–3,57], we split the MS-COCO dataset into 5000 images for validation, 5000 images for testing and the remaining images for training.

4.2. Experimental details

In our experiments, the popular Adam algorithm, which used a learning rate of 0.0002 and a gradient clipping value of 2, was used in all experiments as a gradient update algorithm. The model was iteratively trained for 30 epochs to guarantee convergence, and the BERT output is fixed as a 768-dimensional vector. In the image representation part, the number of objects r is 36 in every image. Each object feature dimension d_i is 2048, and the dimensions of the corresponding tag feature D_t are 768. The number of blocks $k \times k$ is set to 16×16, and L is set to 15. The block index is embedded in a D_b-dimension space, and D_b is 200. The dimension of the final object vector D is 1024. In the sentence representation part, the vector length D_w^b of the word is 768, and the length $\widetilde{D}_w^b$ is 300. The hidden dimension of the bi-GRU is 1024, and the dropout of the bi-GRU is set to 0.5. The parameter α of the linear combination that calculates the similarity of $i2t + t2i$ is 0.5. The GPU used in our experiment platform is Nvidia 2080Ti, and the CPU is Intel (R) Xeon (R) E5-2620 v4.

During the training process, we used the text as the query term and the image as the content item to train and obtain an $t2i$ similarity matrix S_{t2i}. Similarly, we used the image as the query term and the text as the content item to train and obtain an $i2t$ similarity matrix S_{i2t}. The similarity of $i2t + t2i$ is a linear combination of the two similarity matrices of $i2t$ and $t2i$ that was used to obtain the final similarity matrix $S_{i2t+t2i}$.

4.3. Compared methods

To evaluate the performance of our method, we selected many strong baselines for comparison. A brief introduction to these compared methods is as follows:

DVSA [1] refers to deep visual-semantic alignments. This alignment model is based on a combination of convolutional neural networks over image regions, bidirectional recurrent neural networks over sentences, and a structured objective that aligns the two modalities through multimodal embedding.

HM-LSTM [58] refers to Hierarchical Multimodal LSTM, which proposes a hierarchical structured recurrent neural network without the need for any supervised labels that can automatically learn the fine-grained correspondences between phrases and image regions toward dense embedding.

SM-LSTM [59] refers to Selective Multimodal LSTM. It proposes a selective multimodal long short-term memory network for instance-aware image and sentence matching. The sm-LSTM includes a multimodal context-modulated attention scheme at each timestep that can selectively attend to a pair of instances of image and sentence by predicting pairwise instance-aware saliency maps for the image and sentence.

2WayNet [60] introduces a bidirectional neural network architecture. Their approach employs two tied neural network channels that project the two views into a common, maximally correlated space using the Euclidean loss. They show a direct link between the correlation-based loss and Euclidean loss, enabling the use of Euclidean loss for correlation maximization.

DAN [40] refers to the Dual Attention Networks, which jointly leverages visual and textual attention mechanisms to capture the

Table 1
Comparison of cross-modal retrieval on Flickr30K dataset with the competing methods.

Methods		Image-to-Text Retrieval			Text-to-Image Retrieval			mR
		R@1	R@5	R@10	R@1	R@5	R@10	
DVSA [1]		22.2	48.2	61.4	15.2	37.7	50.5	39.2
HM-LSTM [58]		38.1	–	76.5	27.7	–	68.8	–
SM-LSTM [59]		42.5	71.9	81.5	30.2	60.4	72.3	59.8
2WayNet [60]		49.8	67.5	–	36.0	55.6	–	–
DAN [40]		55.0	81.8	89.0	39.4	69.2	79.1	68.9
VSE++[57]		52.9	–	87.2	39.6	–	79.5	–
DPC [61]		55.6	81.9	89.5	39.1	69.2	80.9	69.4
SCO [62]		55.5	82.0	89.3	41.1	70.5	80.1	69.8
SAEM [63]		69.1	91.0	95.1	52.4	81.1	88.1	79.5
CAAN [64]		70.1	91.6	97.2	52.8	79.0	87.9	79.8
IMRAM [65]		74.1	93.0	96.6	53.9	79.4	87.2	80.7
MMCA [42]		74.2	92.8	96.4	54.8	81.4	87.8	81.2
SCAN [2]	t2i	61.8	87.5	93.7	45.8	74.4	83.0	74.4
	i2t	67.7	88.9	94.0	44.0	74.2	82.6	75.2
	i2t+t2i	67.4	90.3	95.8	48.6	77.7	85.2	77.5
PFAN [3]	t2i	66.0	89.6	94.3	49.6	77.0	84.2	76.8
	i2t	67.6	90.0	93.8	45.7	74.7	83.6	75.9
	i2t+t2i	70.0	91.8	95.0	50.4	78.7	86.1	78.7
PFAN++[4]	t2i	67.2	91.2	96.1	50.8	77.8	85.3	78.1
	i2t	67.3	88.6	93.7	45.7	75.4	83.9	75.7
	i2t+t2i	70.1	91.8	96.1	52.7	79.9	87.0	79.6
Ours	t2i	74.1	**94.1**	96.6	**55.2**	82.3	89.1	81.9
	i2t	72.1	92.4	95.9	45.9	77.1	85.6	78.2
	i2t+t2i	**75.1**	93.8	**97.2**	54.5	**82.8**	**89.9**	**82.2**
Improve		↑ 5.1	↑ 2.0	↑ 1.6	↑ 4.8	↑ 4.1	↑ 3.8	↑ 3.5
(Compare PFAN i2t+t2i)		↑ 7.3%	↑ 2.2%	↑ 2.1%	↑ 9.5%	↑ 5.2%	↑ 4.4%	↑ 4.4%

fine-grained interplay between vision and language. The reasoning model allows visual and textual attention to steer each other during collaborative inference.

VSE++ [57], Visual-Semantic Embeddings, introduces a simple change to common loss functions used for multimodal embeddings. That, combined with fine-tuning and the use of augmented data, yields significant gains in retrieval performance.

DPC [61] refers to the dual-path convolutional network, which constructs an end-to-end dual-path convolutional network to learn the image and text representations. They proposed instance loss, which explicitly considers the intramodal data distribution.

SCO [62] improves the image representation by learning semantic concepts and then organizing them into a correct semantic order. Given an image, it uses a multiregional multilabel CNN to predict its semantic concepts. Then, the model uses a context-gated sentence generation scheme for semantic order learning. Finally, it learns the sentence representation with a conventional LSTM and then jointly performs image and sentence matching and sentence generation for model learning.

SAEM [63] refers to Self-Attention Embeddings. It exploits fragment relations in images or texts by a self-attention mechanism and aggregates fragment information into visual and textual embeddings. SAEM extracts salient image regions based on bottom-up attention and uses WordPiece tokens as sentence fragments. The self-attention layers are built to model subtle and fine-grained fragment relations in images and text, respectively, which consist of a multihead self-attention sublayer and a positionwise feed-forward network sublayer.

CAAN [64] proposes a unified Context-Aware Attention Network that selectively focuses on critical local fragments (regions and words) by aggregating the global context. Specifically, it simultaneously utilizes global intermodal alignments and intramodal correlations to discover latent semantic relations.

IMRAM [65] proposes an Iterative Matching with Recurrent Attention Memory (IMRAM) method, in which correspondences between images and texts are captured with multiple steps of alignments. Specifically, it introduces an iterative matching scheme to explore such fine-grained correspondence progressively.

MMCA [42] proposes a novel Multi-Modality Cross Attention Network for image and sentence matching by jointly modeling the intramodality and intermodality relationships of image regions and sentence words in a unified deep model. It designs a novel cross-attention mechanism that is able to exploit not only the intramodality relationship within each modality but also the intermodality relationship between image regions and sentence words so that they complement and enhance each other for image and sentence matching.

SCAN [2] refers to the Stacked Cross Attention Network, which discovers the full latent alignments using both image regions and words in a sentence as context and then infers image–text similarity. It is also the benchmark for most recent models.

PFAN [3] refers to the position focused attention network. This model is our baseline, and it uses the object position clue to enhance the visual–text joint-embedding learning. We first split the images into blocks, by which we infer the relative position of the region in the image. Then, an attention mechanism is proposed to model the relations between the image region and blocks and generate the valuable position feature, which will be further utilized to enhance the region expression and model a more reliable relationship between the visual image and the textual sentence.

PFAN++ [4] integrates the prior object position to enhance visual-text joint-embedding learning. It introduces global features based on PFAN and achieves better results.

4.4. Performance comparison

4.4.1. Results on Flickr30K

Table 1 shows the results of different methods on Flickr30k. It can be seen from the results that our method achieves satisfactory performance on both the from-image-to-text retrieval tasks and the from-text-to-image retrieval tasks. The t2i on the left of the table indicates that only the text-to-image attention method was employed to train the network, and the i2t indicates that only the

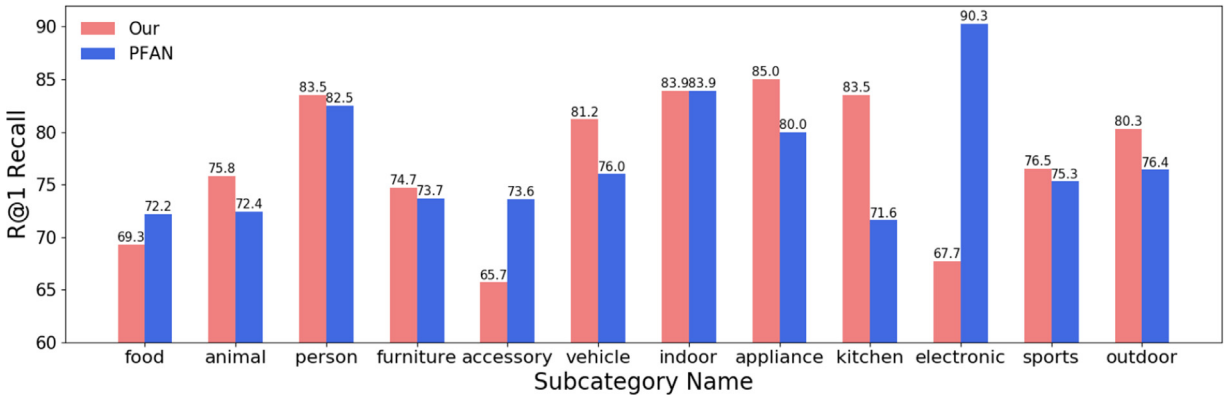

Fig. 2. Subcategory text retrieval results in MSCOCO.

Table 2
Comparison of cross-modal retrieval on MSCOCO dataset with the competing methods.

Methods		Image-to-Text Retrieval			Text-to-Image Retrieval			mR
		R@1	R@5	R@10	R@1	R@5	R@10	
DVSA [1]		38.4	69.9	80.5	27.4	60.2	74.8	58.5
HM-LSTM [58]		43.9	–	87.8	36.1	–	86.7	–
SM-LSTM [59]		53.2	83.1	91.5	40.7	75.8	87.4	72.0
2WayNet [60]		55.8	75.2	–	39.7	66.3	–	–
DAN [40]		55.0	81.8	89.0	39.4	69.2	79.1	69.0
VSE++[57]		64.6	–	95.7	52	–	92	–
DPC [61]		65.6	89.8	95.5	47.1	79.9	90.0	78.0
SCO [62]		69.9	92.9	97.5	56.7	87.5	94.8	83.2
SAEM [63]		71.2	94.1	97.7	57.8	88.6	94.9	84.1
CAAN [64]		75.5	95.4	98.5	61.3	89.7	95.2	85.9
IMRAM [65]		76.7	95.6	98.5	61.7	89.1	95.0	86.1
MMCA [42]		74.8	95.6	97.7	61.6	89.8	95.2	85.9
SCAN [2]	t2i	67.5	92.9	97.6	53.0	85.4	92.9	81.6
	i2t	69.2	93.2	97.5	54.4	86.0	93.6	82.3
	i2t+t2i	72.7	94.8	98.4	58.8	88.4	94.8	84.7
PFAN [3]	t2i	75.8	95.9	**99.0**	61.0	89.1	95.1	86.0
	i2t	70.7	94.1	97.8	53.0	84.5	92.6	82.1
	i2t+t2i	76.5	96.3	**99.0**	61.6	89.6	95.2	86.4
PFAN++[4]	t2i	75.4	95.5	98.2	60.9	88.9	94.7	85.6
	i2t	72.0	94.6	98.5	56.4	86.1	92.6	83.4
	i2t+t2i	77.1	**96.5**	98.3	62.5	**89.9**	95.4	86.7
Ours	t2i	77.5	95.6	98.4	60.7	88.4	94.7	85.9
	i2t	72.0	94.7	97.9	57.0	86.9	93.9	83.7
	i2t+t2i	**78.4**	96.0	98.5	**62.6**	89.6	**95.4**	**86.8**
Improve		↑ 1.9	↓ 0.3	↓ 0.5	↑ 1.0	0.0	↑ 0.2	↑ 0.4
(Compare PFAN i2t+t2i)		↑ 2.5%	↓ 0.3%	↓ 0.5%	↑ 1.6%	0.0%	↑ 0.2%	↑ 0.4%

image-to-text attention method was employed. In both i2t and t2i, the recall rate of our model exceeds the benchmark model, PFAN, in terms of querying text from the image. The best R@1 in terms of querying images from text is 55.0, which was achieved by t2i and is a 9.1% improvement over PFAN. The fused model, i2t+t2i, achieves better performance. The R@1 on querying text from the image even reaches 74.8, which is an improvement of 6.9% compared with PFAN. These results prove the effectiveness of our method.

4.4.2. Results on MSCOCO

Table 2 shows the results of the different methods on MS-COCO. The results show that our method achieves better performance on all important indicators. On the final fusion result of i2t+ t2i, the r@1 of our method for text retrieval is 78.4, which is 1.9 higher than the 76.5 of the PFAN method. Our method has an r@1 of 62.6 for image retrieval, which is a 1.0 improvement over the PFAN method of 61.6.

4.5. Subcategory results

In Figs. 2 and 3, we show the effects of our model and the PFAN model on the MS-COCO dataset for different categories of data. The MS-COCO dataset has 80 small categories and 12 large categories. We calculated the top-1 recall rate (R@1) for different categories of the two models on 12 large categories. Fig. 2 shows the results of the two models on text retrieval, and it can be seen that our model is superior to the PFAN model in 9 out of 12 categories. In addition to the 'Electronic' and 'Accessory' categories, our model works better on categories that are closely related to daily life. In the 'Electronic' and 'Accessory' categories, the key targets in these images are mostly proper nouns. Because these nouns were not marked by the pretraining target detection model, the target detection model could not obtain accurate labels for the proper nouns. Therefore, inaccurate labels lead to a decrease in accuracy. Fig. 3 shows the display results of the two models in terms of image retrieval. It can be seen that our model is superior to the PFAN model in 10 of the 12 categories. In the

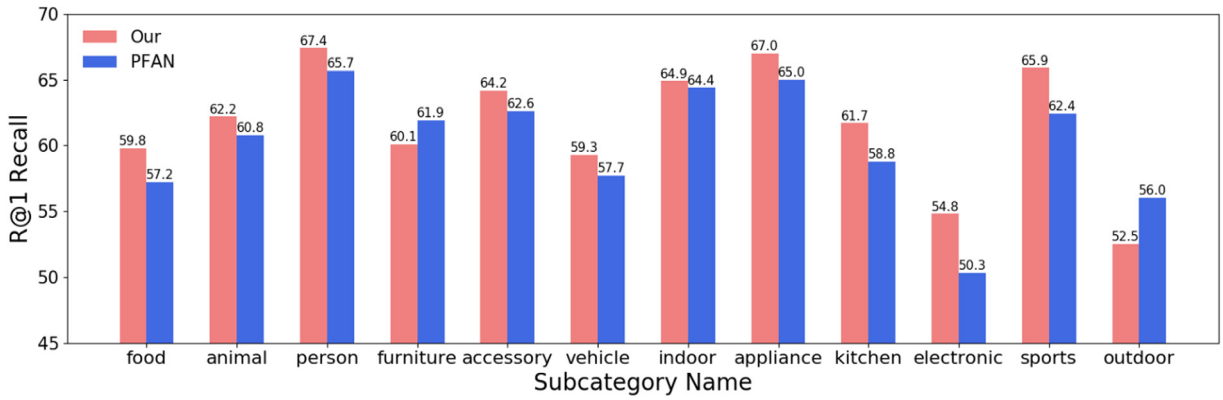

Fig. 3. Subcategory image retrieval results in MSCOCO.

Table 3
The results of the intact model are compared with those of the model with different parts removed.

on Flick30k		Image-to-Text Retrieval			Text-to-Image Retrieval			mR
		R@1	R@5	R@10	R@1	R@5	R@10	
Intact Model	t2i	74.1	94.1	96.6	55.2	82.3	89.1	81.9
	i2t	72.1	92.4	95.6	45.9	77.1	85.6	78.1
	i2t+t2i	75.1	93.8	97.2	54.5	82.8	89.9	82.2
No-Tag	t2i	71.5	93.5	96.9	55.2	81.7	89.0	81.3
	i2t	67.7	91.0	95.6	44.6	76.2	84.5	76.6
	i2t+t2i	73.8	92.7	96.9	53.9	82.0	88.8	81.4
Only-Gru	t2i	68.3	90.2	94.2	45.6	75.4	84.1	76.3
	i2t	67.2	89.7	94.4	49.5	76.7	84.7	77.0
	i2t+t2i	68.9	90.7	94.3	50.0	77.9	85.7	77.9
Only-Bert	t2i	45.2	78.6	87.7	44.7	72.4	81.8	68.4
	i2t	60.5	86.6	92.5	39.4	71.0	80.7	71.8
	i2t+t2i	63.5	88.4	93.3	48.4	77.1	85.3	76.0

other two categories, its scores are similar to those of PFANs. In general, our model is superior to the PFAN model.

4.6. Ablation study

To fully verify the validity of our proposed model, we analyzed the impact of all newly proposed parts on the model results on the Flickr30k dataset. As shown below, (1) 'no-tag' means that our model does not use detected tags, which eliminates the influence of tags on the model. (2) 'Only-GRU' means that the feature extracted by BERT is not used in the text representation. Only GRU models are used to extract text features. (3) 'Only-BERT' means that we only use the pretrained BERT model in the text representation but not the GRU model. (4) is our complete model in which the image feature representation is enhanced with a tag. In the next part, BERT+GRU is used to extract text features.

Tag enhancement: To verify the influence of generative region tag features added to the image features on the experimental results and based on the complete model, we remove the generated tag feature. After we train with t2i, the R@1 of the model without the generative tag was reduced from 73.9 to 71.5, a reduction of 2.4 compared with that of the complete model. After i2t training, the 'no-tag' model decreased from 70.1 to 67.7 in comparison with the complete model, which also decreased by 2.4. Finally, the results of the 't2i + i2t' model combining the similarity of the two training results decreased from 74.8 to 73.8. This result shows the effectiveness of the proposed generative region tag.

Text feature extraction: To verify the effectiveness of the BERT+GRU method in text feature extraction, we conducted ablation experiments using BERT only and GRU only in text feature extraction. In the case of Only-GRU, the r@1 of the final t2i + i2t fusion model is 68.9. In the case of using only pretrained BERT, the final t2i+i2t result R@1 is 63.5. The results of these two models are lower than that of the complete model. We did not pretrain the GRU model on large-scale datasets. In our GLFN model, we first use BERT for word embedding and then reduce the feature from m-dimension to n-dimension. Finally, we use the GRU for sentence representation. The results show that the BERT model performs better in downstream NLP tasks, but in our model, we only use BERT for word representation, not for sentence representation. If we do not use the bidirectional GRU, then the sentence representation only consists of the word representation without contextual information. The lack of the GRU model means the lack of contextual information, so the BERT model is less effective.

4.7. Parameter analysis

To further study our model, the influence of different parameters on the model is discussed, and the values of some important parameters in the model are analyzed and discussed.

In Fig. 4, we discuss the effect of the embed size on the result. The embed size parameter determines the final image feature dimension and text feature dimension in the model, which is the attention alignment input feature calculation between the image feature and text feature. We choose 256, 512, 1024, and 1536 as preselected parameter values. The experiment was carried out without changing the other parameters. The r@1 result is shown in the figure. The red line is the result of text retrieval, and the blue line is the result of image retrieval. When the embed size is

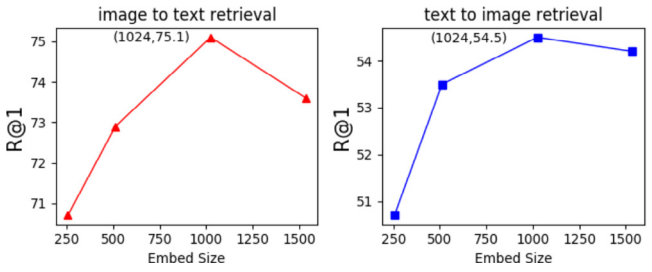

Fig. 4. Embed size parameter analysis.

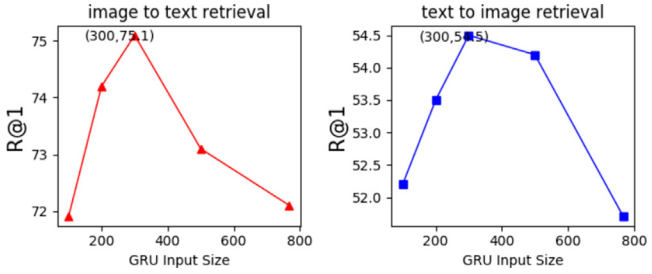

Fig. 5. GRU input size parameter analysis.

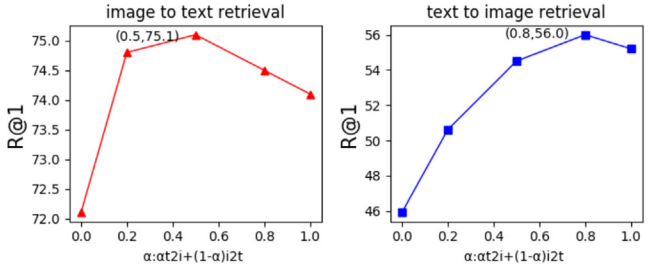

Fig. 6. α Parameter analysis.

1024, the optimal value is obtained in both image retrieval and text retrieval.

In Fig. 5, we discuss the effect of the value of the GRU input size on the result. In the extraction of text features, we first use BERT to obtain the features of each word. If the features obtained by BERT are directly input into the GRU, severe overfitting will occur. Therefore, a linear layer is added between BERT and GRU to compress BERT's output vector, and the compressed vector is input into the GRU. We choose 100, 200, 300, 500, and 768 as preselected parameter values for the compressed vectors of the linear layer. Among them, the BERT output has 768 dimensions; that is, the BERT output is directly used without compressing its features. The experiment was carried out without changing the other parameters. The r@1 result is shown in the figure. The red line is the result of text retrieval, and the blue line is the result of image retrieval. When the GRU input size is selected as 300, the optimal value is obtained in both image retrieval and text retrieval.

In Fig. 6, we discuss the parameter α and use formula (33) to calculate the i2t + t2i fusion model, where is the proportion of different models in the fusion model. The higher the value is, the greater the proportion of the i2t model, and the model tends to text retrieval. The lower the value is, the greater the proportion of the t2i model, which is inclined to image retrieval. We selected 0, 0.2, 0.5, 0.8, and 1.0 as candidate values and carried out experiments without changing the other parameters. The r@1 result is shown in Fig. 6. The red lines are for text retrieval, and it can be seen that the optimal value for text retrieval is achieved when α is 0.5. When α is 0.8, the optimal value of image retrieval is obtained. The results in Tables 1 and 3 both use an α of 0.5.

4.8. Position attention visualization

We design a position attention mechanism to adaptively determine the importance of the block position on the region, and the region feature and the generative tag feature are then concatenated and input into the image–text attention mechanism to investigate the interplay between the regions and tags. In this subsection, we visualize the attention results in this paper. An exemplary visualization result is shown in Fig. 7, where the green box indicates the image region, and the tag with the region is depicted in red text in each figure. The red frames indicate the blocks of the current region; we depict the blocks of the first 6 maximum weights for each region, and the brighter blocks have higher weights. We can observe that the brighter blocks reveal the more important part of the regions. For example, in the second image in the first row, the brightest block is located in the center of the region, which is one of the most semantically related parts.

4.9. Retrieval examples

In Fig. 8, we compared our model with the PFAN model on the Flickr30K dataset and display the text retrieval results. For each picture, we input the top 5 sentences sorted from top to bottom. Sentences with green $\surd$ checkmarks represent correct matches, and those with red marks $\times$ represent incorrect matches. For example, in the second picture, our model finds more suitable results than the PFAN model. For objects that do not appear in the picture, such as 'Apple' and 'Parents', our model can easily filter out sentences with these incorrect words. In the third picture, although our model also has incorrect sentences, the incorrect sentences retrieved by our model rank lower than those of PFAN. Meanwhile, compared with PFAN, we filter out the incorrect sentences with 'Man'. It can be seen that our model can retrieve more correct sentences than PFAN, and the correct results can be ranked higher.

In Fig. 9, we compare our model with the PFAN model on the Flickr30K dataset for image retrieval results presentation. For each sentence, we input a picture of the top 5 sorted from left to right. The green boxes represent correct matches, and the red boxes represent incorrect matches. For example, in the lower-right example, our model focuses on specific objects that appear in the sentence, such as 'Soccer'. Therefore, there is 'Soccer' in all results. PFAN, on the other hand, focuses on 'Jumping' and falsely detects several images that are different from the sentence, such as the first image of 'Skateboarding'. It can be seen that our model can rank the correct results higher than PFAN.

5. Conclusion

In this paper, we propose a GLFN model to solve the problem of text–image matching. On the one hand, we use the image tag generated from the image to enhance the expression ability of the region image features, and the added features reduce the gap between the image and the text. On the other hand, we use both the BERT and Bi-GRU models to represent the text. Using a linear layer solves the overfitting problem caused by the simultaneous

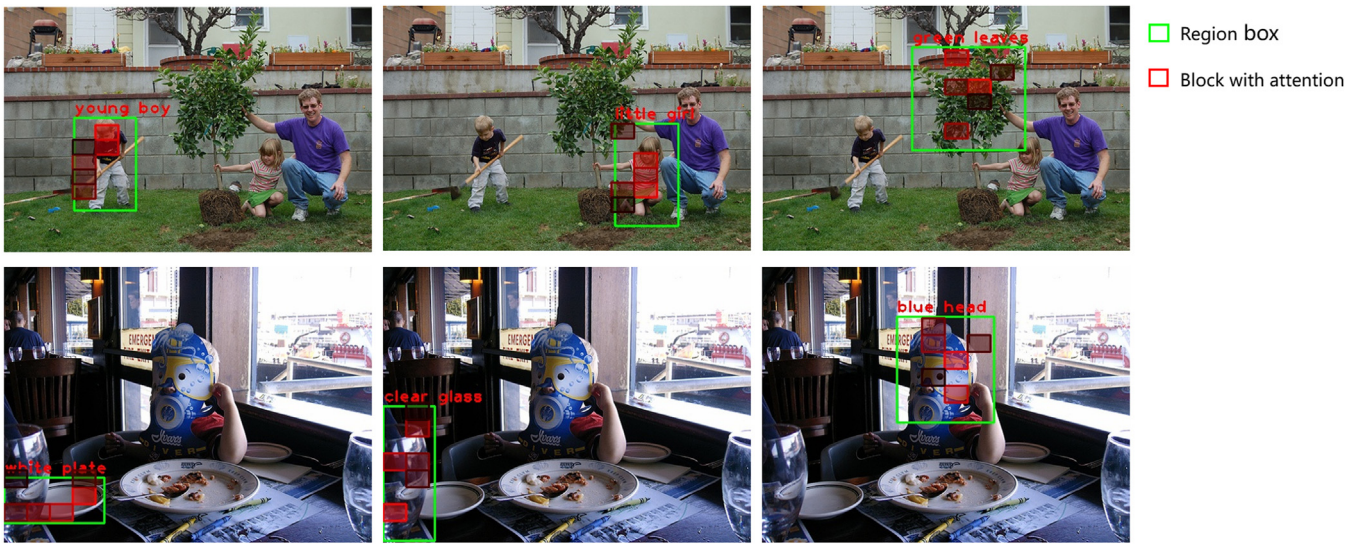

Fig. 7. Attention visualized map, with red text representing region's tag, green box representing the bounding box acquired by object detecting. The brighter blocks are, the higher weights of blocks are.

our	A truck driving along a beach near a flock of seagulls . ✔	A small child wearing a blue and white t-shirt happily holding a yellow plastic alligator . ✔	Woman wearing a yellow hat , pink shirt , and red apron is holding food in a kitchen . ✔
	Red white and blue SUV patrolling the beach and shoreline on a cloudy day while seagulls walk in the sand . ✔	A little boy in a light blue shirt playing outside with a toy crocodile . ✔	A woman wearing a pink shirt and red apron stands in her restaurant holding food and surrounded by bagged breads , a microwave and the chalkboard menu up on the wall . ✔
	A red , white , and blue security vehicle is driving down a rocky beach toward some birds on a cloudy day . ✔	A little boy is wearing a blue and white shirt while holding a toy animal that is either a crocodile or an alligator . ✔	Employee in pink shirt smiles for camera inside a restaurant kitchen . ✔
	A car parked at the beach . ✔ Orange SUV drive by the shore . ✔	Two children sit side by side while eating a treat . ✘ A little boy holding a baby reptile . ✔	A female barista dressed in black is making coffee behind a counter . ✘ A woman stands behind the counter in the open kitchen of a restaurant ✘
PFAN	A truck driving along a beach near a flock of seagulls . ✔	A small child wearing a blue and white t-shirt happily holding a yellow plastic alligator . ✔	A woman stands behind the counter in the open kitchen of a restaurant ✘
	Red white and blue SUV patrolling the beach and shoreline on a cloudy day while seagulls walk in the sand . ✔	A young , asian child sitting on its parents ' shoulders , clapping . ✘	A female barista dressed in black is making coffee behind a counter . ✘
	A red , white , and blue security vehicle is driving down a rocky beach toward some birds on a cloudy day . ✔	Two children sit side by side while eating a treat . ✔	Woman wearing a yellow hat , pink shirt , and red apron is holding food in a kitchen . ✔
	A car parked at the beach . ✔	A little boy is wearing a blue and white shirt while holding a toy animal that is either a crocodile or an alligator . ✔	man standing at a counter in a convenience store with a woman on the other side of the counter wearing a white shirt looking at him very expectantly . ✘
	Two cars are on a racetrack . ✘	A young boy is eating an apple . ✘	Employee in pink shirt smiles for camera inside a restaurant kitchen . ✔

Fig. 8. Comparing our model with the PFAN model in the Flickr30K dataset text retrieval results display.

use of two models. We have verified through experiments that for text feature extraction, the combination of the BERT and GRU models is better than a single model. Ultimately, the results of experiments on the Flickr30K and MS-COCO datasets demonstrate

Fig. 9. Comparing our model with the PFAN model in the Flickr30K data set image retrieval results display.

the effectiveness of our proposed method. In future work, we will continue to explore how to use fewer data to train the model and facilitate a more practical direction for cross-modal retrieval.

CRediT authorship contribution statement

Guoshuai Zhao: Conceptualization, Methodology, Formal analysis, Writing – original draft, Writing – review & editing, Supervision. **Chaofeng Zhang:** Methodology, Software, Data curation, Writing – original draft, Formal analysis. **Heng Shang:** Validation, Visualization. **Yaxiong Wang:** Methodology, Formal analysis. **Li Zhu:** Writing – review & editing. **Xueming Qian:** Resources, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grants 61902309, in part by China Postdoctoral Science Foundation under Grant 2020M683496 and BX20190273, in part by the Humanities and Social Sciences Foundation of Ministry of Education, China under Grant 16XJAZH003, and in part by the Science and Technology Program of Xi'an, China under Grant 21RGZN0017. (G. Zhao is the corresponding author.)

References

[1] A. Karpathy, L. Fei-Fei, Deep visual-semantic alignments for generating image descriptions, in: IEEE Conf. Computer Vision and Pattern Recognition, 2015, pp. 3128–3137.

[2] K.-H. Lee, X. Chen, G. Hua, H. Hu, X. He, Stacked cross attention for image-text matching, in: 15th European Conf. Computer Vision, Vol. 11208, 2018, pp. 212–228.

[3] Y. Wang, H. Yang, X. Qian, L. Ma, J. Lu, B. Li, X. Fan, Position focused attention network for image-text matching, in: Proc. Twenty-Eighth Int. Joint Conf. Artificial Intelligence, 2019, pp. 3792–3798.

[4] Y. Wang, H. Yang, X. Bai, X. Qian, L. Ma, J. Lu, B. Li, X. Fan, PFAN++: Bi-directional image-text retrieval with position focused attention network, IEEE Trans. Multim. (2020).

[5] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: IEEE Conf. Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[6] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards real-time object detection with region proposal networks, in: Neural Information Processing Systems, 2015, pp. 91–99.

[7] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, L. Zhang, Bottom-up and top-down attention for image captioning and visual question answering, in: IEEE Conf. Computer Vision and Pattern Recognition, 2018, pp. 6077–6086.

[8] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, in: Proc. Conf. the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2018, pp. 4171–4186.

[9] J. Chung, C. Gulcehre, K. Cho, Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, 2014.

[10] B.A. Plummer, L. Wang, C.M. Cervantes, J.C. Caicedo, J. Hockenmaier, S. Lazebnik, Flickr30K entities: Collecting region-to-phrase correspondences for richer image-to-sentence models, Int. J. Comput. Vis. 123 (1) (2017) 74–93.

[11] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, Microsoft coco: Common objects in context, in: 13th European Conf. Computer Vision, Vol. 8693, 2014, pp. 740–755.

[12] S. Karaoglu, R. Tao, T. Gevers, A.W.M. Smeulders, Words matter: Scene text for image classification and retrieval, IEEE Trans. Multim. 19 (5) (2017) 1063–1076.

[13] X. Li, C. Xu, X. Wang, W. Lan, Z. Jia, G. Yang, J. Xu, COCO-CN for cross-lingual image tagging, captioning, and retrieval, IEEE Trans. Multim. 21 (9) (2019) 2347–2360.

[14] E. Yu, J. Sun, J. Li, X. Chang, X. Han, A.G. Hauptmann, Adaptive semi-supervised feature selection for cross-modal retrieval, IEEE Trans. Multim. 21 (5) (2019) 1276–1288.

[15] L. Zhang, B. Ma, G. Li, Q. Huang, Q. Tian, Generalized semi-supervised and structured subspace learning for cross-modal retrieval, IEEE Trans. Multim. 20 (1) (2017) 128–141.

[16] L. Zhang, B. Ma, G. Li, Q. Huang, Q. Tian, Cross-modal retrieval using multiordered discriminative structured subspace learning, IEEE Trans. Multim. 19 (6) (2017) 1220–1233.

[17] Y. He, S. Xiang, C. Kang, J. Wang, C. Pan, Cross-modal retrieval via deep and bidirectional representation learning, IEEE Trans. Multim. 18 (7) (2016) 1363–1377.

[18] C. Kang, S. Xiang, S. Liao, C. Xu, C. Pan, Learning consistent feature representation for cross-modal multimedia retrieval, IEEE Trans. Multim. 17 (3) (2015) 370–381.

[19] H. Hotelling, Relations between two sets of variates, in: Breakthroughs in Statistics, Springer, 1992, pp. 162–190.
[20] D.R. Hardoon, S. Szedmak, J. Shawe-Taylor, Canonical correlation analysis: An overview with application to learning methods, Neural Comput. 16 (12) (2004) 2639–2664.
[21] N. Rasiwasia, J. Costa Pereira, E. Coviello, G. Doyle, G.R. Lanckriet, R. Levy, N. Vasconcelos, A new approach to cross-modal multimedia retrieval, in: Proc. 18th ACM Int. Conf. on Multimedia, 2010, pp. 251–260.
[22] J.C. Pereira, E. Coviello, G. Doyle, N. Rasiwasia, G.R. Lanckriet, R. Levy, N. Vasconcelos, On the role of correlation and abstraction in cross-modal multimedia retrieval, IEEE Trans. Pattern Anal. Mach. Intell. 36 (3) (2013) 521–535.
[23] Y. Gong, Q. Ke, M. Isard, S. Lazebnik, A multi-view embedding space for modeling internet images, tags, and their semantics, Int. J. Comput. Vis. 106 (2) (2014) 210–233.
[24] V. Ranjan, N. Rasiwasia, C. Jawahar, Multi-label cross-modal retrieval, in: IEEE Int. Conf. on Computer Vision, 2015, pp. 4094–4102.
[25] A. Sharma, A. Kumar, H. Daume, D.W. Jacobs, Generalized multiview analysis: A discriminative latent space, in: IEEE Conf. Computer Vision and Pattern Recognition, 2012, pp. 2160–2167.
[26] A. Sharma, D.W. Jacobs, Bypassing synthesis: PLS for face recognition with pose, low-resolution and sketch, in: IEEE Conf. Computer Vision and Pattern Recognition, 2011, pp. 593–600.
[27] D. Li, N. Dimitrova, M. Li, I.K. Sethi, Multimedia content processing through cross-modal association, in: Eleventh ACM Int. Conf. Multimedia, 2003, pp. 604–611.
[28] V. Mahadevan, C.W. Wong, J.C. Pereira, T. Liu, N. Vasconcelos, L.K. Saul, Maximum covariance unfolding: Manifold learning for bimodal data, in: 25th Conf. Neural Information Processing Systems, 2011, pp. 918–926.
[29] Y. Wu, S. Wang, Q. Huang, Online fast adaptive low-rank similarity learning for cross-modal retrieval, IEEE Trans. Multim. 22 (5) (2020) 1310–1322.
[30] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, A.Y. Ng, Multimodal deep learning, in: Proc. Int. Conf. Machine Learning, 2011, pp. 689–696.
[31] G. Andrew, R. Arora, J. Bilmes, K. Livescu, Deep canonical correlation analysis, in: Proc. 30th Int. Conf. on Machine Learning, 2013, pp. 1247–1255.
[32] F. Yan, K. Mikolajczyk, Deep correlation for matching images and text, in: IEEE Conf. Computer Vision and Pattern Recognition, 2015, pp. 3441–3450.
[33] F. Feng, X. Wang, R. Li, Cross-modal retrieval with correspondence autoencoder, in: Proc. ACM Int. Conf. Multimedia, 2014, pp. 7–16.
[34] H. Zhang, Y. Yang, H. Luan, S. Yang, T.-S. Chua, Start from scratch: Towards automatically identifying, modeling, and naming visual attributes, in: Proc. ACM Int. Conf. Multimedia, 2014, pp. 187–196.
[35] W. Wang, R. Arora, K. Livescu, J. Bilmes, On deep multi-view representation learning, in: Proc.32nd Int. Conf. Machine Learning, Vol. 37, 2015, pp. 1083–1092.
[36] Y. Wei, Y. Zhao, C. Lu, S. Wei, L. Liu, Z. Zhu, S. Yan, Cross-modal retrieval with CNN visual features: A new baseline, IEEE Trans. Cybern. 47 (2) (2016) 449–460.
[37] J. Gu, J. Cai, S.R. Joty, L. Niu, G. Wang, Look, imagine and match: Improving textual-visual cross-modal retrieval with generative models, in: IEEE Conf. Computer Vision and Pattern Recognition, 2018, pp. 7181–7189.
[38] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Conf. Neural Information Processing Systems, 2014, pp. 2672–2680.
[39] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: IEEE Conf. Computer Vision and Pattern Recognition, 2014, pp. 580–587.
[40] H. Nam, J.-W. Ha, J. Kim, Dual attention networks for multimodal reasoning and matching, in: IEEE Conf. Computer Vision and Pattern Recognition, 2017, pp. 299–307.
[41] Z. Ji, H. Wang, J. Han, Y. Pang, Saliency-guided attention network for image-sentence matching, in: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019, IEEE, 2019, pp. 5753–5762.
[42] X. Wei, T. Zhang, Y. Li, Y. Zhang, F. Wu, Multi-modality cross attention network for image and sentence matching, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020, Computer Vision Foundation / IEEE, 2020, pp. 10938–10947.
[43] G. Zhao, Z. Liu, Y. Chao, X. Qian, CAPER: Context-aware personalized emoji recommendation, IEEE Trans. Knowl. Data Eng. 33 (9) (2021) 3160–3172.
[44] X. Zheng, G. Zhao, L. Zhu, X. Qian, PERD: Personalized emoji recommendation with dynamic user preference, in: SIGIR, 2022, ACM, 2022, pp. 1922–1926.
[45] M. Belkin, I. Matveeva, P. Niyogi, Regularization and semi-supervised learning on large graphs, in: 17th Conf. on Learning Theory, 2004, pp. 624–638.
[46] X. Zhai, Y. Peng, J. Xiao, Heterogeneous metric learning with joint graph regularization for cross-media retrieval, in: Proc. AAAI Conf. Artificial Intelligence, 2013.
[47] X. Zhai, Y. Peng, J. Xiao, Learning cross-media joint representation with sparse and semisupervised regularization, IEEE Trans. Circuits Syst. Video Technol. (2013) 965–978.
[48] Y. Peng, X. Zhai, Y. Zhao, X. Huang, Semi-supervised cross-media feature learning with unified patch graph regularization, IEEE Trans. Circuits Syst. Video Technol. (2015) 583–596.
[49] H. Tang, G. Zhao, X. Bu, X. Qian, Dynamic evolution of multi-graph based collaborative filtering for recommendation systems, Knowl.-Based Syst. 228 (2021) 107251.
[50] Y. Hu, L. Zheng, Y. Yang, Y. Huang, Twitter100k: A real-world dataset for weakly supervised cross-media retrieval, IEEE Trans. Multim. 20 (4) (2018) 927–938.
[51] Z. Yang, L. Zhu, Y. Wu, Y. Yang, Gated channel transformation for visual recognition, in: CVPR, 2020, pp. 11791–11800.
[52] Z. Yang, Y. Wei, Y. Yang, Associating objects with transformers for video object segmentation, in: NeurIPS, 2021, pp. 2491–2502.
[53] D. Mandal, P. Rao, S. Biswas, Semi-supervised cross-modal retrieval with label prediction, IEEE Trans. Multim. 22 (9) (2020) 2345–2353.
[54] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Neural Information Processing Systems, 2017, pp. 5998–6008.
[55] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D.A. Shamma, et al., Visual genome: Connecting language and vision using crowdsourced dense image annotations, Int. J. Comput. Vis. 123 (1) (2017) 32–73.
[56] F. Schroff, D. Kalenichenko, J. Philbin, Facenet: A unified embedding for face recognition and clustering, in: IEEE Conf. Computer Vision and Pattern Recognition, 2015, pp. 815–823.
[57] F. Faghri, D.J. Fleet, J.R. Kiros, S. Fidler, VSE++: Improving visual-semantic embeddings with hard negatives, in: British Machine Vision Conf., 2018, p. 12.
[58] Z. Niu, M. Zhou, L. Wang, X. Gao, G. Hua, Hierarchical multimodal lstm for dense visual-semantic embedding, in: IEEE Int. Conf. Computer Vision, 2017, pp. 1881–1889.
[59] Y. Huang, W. Wang, L. Wang, Instance-aware image and sentence matching with selective multimodal lstm, in: IEEE Conf. Computer Vision and Pattern Recognition, 2017, pp. 2310–2318.
[60] A. Eisenschtat, L. Wolf, Linking image and text with 2-way nets, in: IEEE Conf. Computer Vision and Pattern Recognition, 2017, pp. 4601–4611.
[61] Z. Zheng, L. Zheng, M. Garrett, Y. Yang, M. Xu, Y.-D. Shen, Dual-path convolutional image-text embeddings with instance loss, ACM Trans. Multim. Comput. Commun. Appl. 16 (2) (2020) 1–23.
[62] Y. Huang, Q. Wu, C. Song, L. Wang, Learning semantic concepts and order for image and sentence matching, in: IEEE Conf. Computer Vision and Pattern Recognition, 2018, pp. 6163–6171.
[63] Y. Wu, S. Wang, G. Song, Q. Huang, Learning fragment self-attention embeddings for image-text matching, in: L. Amsaleg, B. Huet, M.A. Larson, G. Gravier, H. Hung, C. Ngo, W.T. Ooi (Eds.), Proceedings of the 27th ACM International Conference on Multimedia, MM 2019, Nice, France, October 21-25, 2019, ACM, 2019, pp. 2088–2096.
[64] Q. Zhang, Z. Lei, Z. Zhang, S.Z. Li, Context-aware attention network for image-text retrieval, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020, Computer Vision Foundation / IEEE, 2020, pp. 3533–3542.
[65] H. Chen, G. Ding, X. Liu, Z. Lin, J. Liu, J. Han, IMRAM: Iterative matching with recurrent attention memory for cross-modal image-text retrieval, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020, Computer Vision Foundation / IEEE, 2020, pp. 12652–12660.