Learning Semantic Structure-preserved Embeddings for Cross-modal Retrieval

Yiling Wu[1,2], Shuhui Wang[1,*], Qingming Huang[1,2]
[1] Key Lab of Intell. Info. Process., Inst. of Comput. Tech., CAS, Beijing, 100190, China.
[2] University of Chinese Academy of Sciences, Beijing, 100049, China.
yiling.wu@vipl.ict.ac.cn,wangshuhui@ict.ac.cn,qmhuang@ucas.ac.cn

ABSTRACT

This paper learns semantic embeddings for multi-label cross-modal retrieval. Our method exploits the structure in semantics represented by label vectors to guide the learning of embeddings. First, we construct a semantic graph based on label vectors which incorporates data from both modalities, and enforce the embeddings to preserve the local structure of this semantic graph. Second, we enforce the embeddings to well reconstruct the labels, *i.e.*, the global semantic structure. In addition, we encourage the embeddings to preserve local geometric structure of each modality. Accordingly, the local and global semantic structure consistencies as well as the local geometric structure consistency are enforced, simultaneously. The mappings between inputs and embeddings are designed to be nonlinear neural network with larger capacity and more flexibility. The overall objective function is optimized by stochastic gradient descent to gain the scalability on large datasets. Experiments conducted on three real world datasets clearly demonstrate the superiority of our proposed approach over the state-of-the-art methods.

KEYWORDS

cross-modal retrieval, graph embeddings, semantic embeddings

ACM Reference Format:
Yiling Wu, Shuhui Wang, Qingming Huang. 2018. Learning Semantic Structure-preserved Embeddings for Cross-modal Retrieval. In *MM'18: 2018 ACM Multimedia Conference, Oct. 22–26, 2018, Seoul, Republic of Korea.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3240508.3240521

1 INTRODUCTION

The past couple of decades witnessed an explosive increase in the amount of multi-modal data on the Internet, which describes information from complementary modalities. Cross-modal retrieval [24] for the multi-modal data has received considerable attention. Given a query in one modality, a list of documents in another modality is returned by cross-modal retrieval, *e.g.*, images retrieved with a text query or vice versa. The returned list of documents are to be ordered according to the semantic relevances between database documents and the query document.

As a standard solution of cross-modal retrieval, subspace learning [2, 20, 23, 24, 30] aims to find a low dimensional latent common space that can well preserve or capture the cross-modal relationship among data objects. For example, CCA [13] and PLS [25] aim to learn a low dimensional common subspace to make cross-modal data pairs maximally correlated. Following similar paradigm, significant research efforts have been devoted afterworth [2, 20, 26, 31] to build modality-specific mapping functions by preserving intra- and inter-modal similarities. Nevertheless, the similarity information is either employed as the observed information to fit, or simply considered as a set of binary class indicators to identify if cross-modal data objects belong to the same class.

Real-world cross-modal data contains rich semantic information. For example, a natural image can be associated with multiple category labels, and the keywords of these categories also describe the topics in the corresponded textual description of the image. A label vector [23] can be used to represent the multi-label information of each image or text. Although cross-modal data have different feature spaces, they share a common label representation. In existing frameworks, the semantic labels are either used as an auxiliary view [9], or to calculate the weights of cross-modal training pairs [23], or used as the back constraints on the learned representation [9]. Despite of their simplicity, the semantic similarities modeled on inter-modal data pairs provide only partial information on the whole cross-modal data corpus. For large scale similarity-based information retrieval, techniques of compact embeddings (codes) [10, 21, 22, 31] becomes increasing popular and important in recent years. However, for data with multiple modalities, the full semantic structure reflected by intra-modal structure, inter-modal correlation and label similarity has not been considered.

We address the cross-modal representation learning by fully exploring the semantic structure of the multi-label information. Specifically, to fully represent the cross-modal semantic similarity structure, we construct a semantic graph based on the corresponding label vectors, where each vertex in the graph represents an image or a text document, and

[*] Corresponding author.

MM '18, October 22–26, 2018, Seoul, Republic of Korea
© 2018 Association for Computing Machinery.
ACM ISBN 978-1-4503-5665-7/18/10...$15.00
https://doi.org/10.1145/3240508.3240521

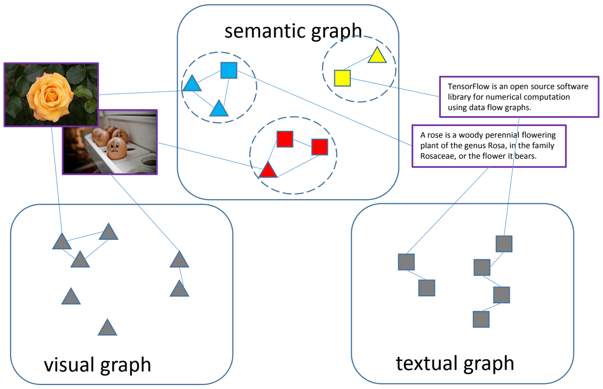

semantic graph
TensorFlow is an open source software library for numerical computation using data flow graphs.
A rose is a woody perennial flowering plant of the genus Rosa, in the family Rosaceae, or the flower it bears.
visual graph
textual graph

Figure 1: Every image or text is a node on the semantic graph and its modality-specific feature similarity graph. Triangles denote images and rectangles denote texts. On semantic graph, different colors represent different semantic clusters.

each edge connecting two vertices denotes the similarity score using their corresponding label vectors, see Figure 1. The semantic graph reflects the semantic affinity structure in a more comprehensive way, *i.e.*, one can easily identify all the semantically similar data objects (image and text) given a specific vertex (image or text) by some efficient affinity search techniques [10, 22] on the semantic graph. According to the spectral graph theory [5], the affinities provide essential information on the graph spectral structure. Therefore, compared to randomized training data selection strategy widely used by previous approaches, a set of training data pairs can be sampled from the database which contain more complete set of instances with fine-grained semantic relevance scores rather than the coarse similar/dissimilar ones.

Accordingly, we propose to learn the **S**emantic-**S**tructure-**P**reserved cross-modal **E**mbedding (SSPE) which encourages both local and global semantic structure consistencies. The local semantic structure is encoded by the semantic graph, and the global structure consistency [19] is enforced by reconstruction of the original label vectors. As shown in Figure 2, the embeddings in hidden space are connected to the label space using a mapping g learned by least-square fitting, then the label reconstruction error passes back to guide the embedding learning.

Moreover, nearby points in the feature spaces are likely to have the same labels [35, 37]. To enforce smoothness and semantic information propagation along the data manifolds, the embeddings should be learned towards the consistency with local geometric structure in the original visual space and textual space. As shown in Figure 1, we build two affinity graphs on the original visual and textual feature spaces, respectively. To avoid time-consuming similarity matrix storage and eigen-analysis on the graph Laplacians, we sample neighborhood vertices of the query vertices on each graph according to the edge weights, and maximizes the likelihood of neighborhood preservation of all the query vertices. The scalable negative sampling algorithm [21] is applied to approximate the likelihood. Based on the three graphs and neighborhood sampling strategy, the most informative sets of both intra-modal and inter-modal training data pairs can be quickly identified for model training.

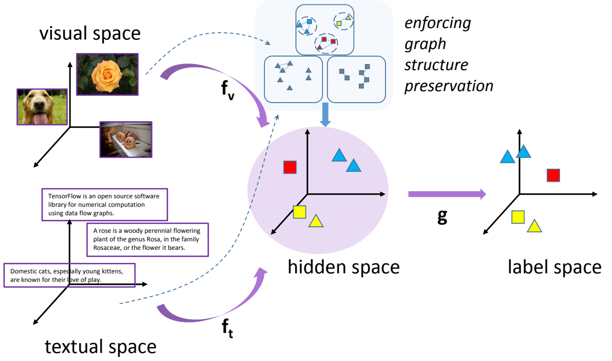

visual space
f_v
enforcing graph structure preservation
TensorFlow is an open source software library for numerical computation using data flow graphs.
A rose is a woody perennial flowering plant of the genus Rosa, in the family Rosaceae, or the flower it bears.
Domestic cats, especially young kittens, are known for their love of play.
hidden space
g
label space
f_t
textual space

Figure 2: Illustration of our method. Starting by inputs of images and texts, the model outputs their predicted labels in a feed-forward manner. The hidden space indicates the semantic structure-preserved embedding space to learn, which bridges the inputs and outputs, and is then used for cross-modal retrieval.

The functions f_v and f_t to project images and texts to the hidden space are neural networks with stacked nonlinear mappings, which endow our model with large capacity and more flexibility. The overall objective function, including terms of **local semantic structure preserving**, **global semantic structure preserving** and **local geometric structure preserving**, is optimized by stochastic gradient descent and standard back-propagation on large datasets. Finally, we apply the proposed method to cross-modal retrieval tasks. Experiments conducted on real-world datasets well demonstrate its efficiency and effectiveness. In summary, our key contributions are summarized as follows.

(1) We propose SSPE, a semantic structure-preserved embedding learning method for multi-label cross-modal retrieval. The embeddings are learned towards local and global semantic structure consistencies as well as local geometric structure consistency.
(2) By using neighborhood sampling and negative sampling, our model does NOT require similarity matrix storage for training, and it can scale well to large real-world cross-modal datasets.
(3) Experimental results on three public cross-modal datasets show significant improvement over existing state-of-the-art methods on the cross-modal retrieval tasks.

2 RELATED WORK

2.1 Cross-modal retrieval

Cross-modal retrieval methods can be generally divided into two categories: traditional methods and deep learning based methods.

Traditional methods. Traditional methods usually learn a latent space based on linear projections. CCA [13], as a classical method, aims at learning a latent space by maximizing the correlation between the projected data of two modalities. PLS [25] is another classical method which creates orthogonal score vectors by maximizing the covariance between the projected data of two modalities. As labels contain rich semantic information, CCA is extended by using label information. Sharma *et al.* [2] combine popular supervised and unsupervised feature extraction techniques with CCA. 3view CCA [9] extends CCA by incorporating a third view, *i.e.*, the label vectors, to capture high level semantics. ml-CCA [23] extends CCA with the high level multi-label semantic information to establish correspondences between different modalities.

Besides, some cross-modal learning methods focusing on relative order of the retrieved documents employ learning-to-rank techniques. Yao *et al.* [36] propose RCCA which jointly explores subspace learning and pairwise learning-to-rank techniques. It initially finds a common subspace by CCA, and further simultaneously learns a bilinear similarity function and adjusts the subspace to preserve the preference relations. Wu *et al.* [33] propose an online learning algorithm which learns a similarity function with bi-directional pairwise learning-to-rank loss. Wu *et al.* [32] propose Bi-CMSRM which optimizes a bi-directional listwise ranking loss by structural risk minimization.

There are some cross-modal learning methods that directly use label space as the common space and see the retrieval task as a classification task. LCFS [30] performs linear regression to label space with $\ell 21$-norm constraints on the projection matrices and a low-rank constraint on the projected data. LGCFL [17] performs ε-dragging on the label space and imposes group sparse constraints in regression process. Deng *et al.* [6] proposed a discriminative dictionary learning method augmented with common label alignment. However, directly using label space as the common hidden space can't discover the correlation between labels and fixes the dimension of common space to the dimension of label space.

Deep learning based methods. Since deep methods provide better representation learning power, neural network is adopted by many cross-modal methods recently. Masci *et al.* [20] propose MMNN to train a two-branch neural network to map images and texts into a common space. A loss function involving intra-modality and inter-modality similarity is optimized to learn the mappings. Wang *et al.* [31] propose to train a two-branch neural network using a large-margin-based objective that combines cross-view ranking constraints with within-view neighborhood structure preservation constraints. He *et al.* [12] apply two convolution based networks to map images and texts into a common space in which the cross-modal similarity is measured by cosine distance. Subsequently, a bi-directional loss involving the matched and unmatched image-text pairs is designed. Yan *et al.* [34] propose to match images and texts in a joint latent space learnt with deep canonical correlation analysis (DCCA) [3]. Corr-AE [8] is constructed by correlating hidden representations of two autoencoders.

Although many cross-modal methods have been proposed, there is a need to capture the nonlinear relation between cross-modal data and label information. To achieve this goal, our method uses neural network to gain the nonlinear representation power and uses label information to guide the learning of semantic embeddings.

2.2 Learning Graph Embedding

Graphs are common data representations used in many real-world problems. Earlier works like Graph embedding [35] first construct the affinity graph based on the feature vectors and then solve the eigenvectors of the largest eigenvalues to approximate the original graph. Graph-based cross-modal methods [18, 27] use LPP [11] style approach to learn linear mappings. However, they adopt shallow models which are difficult to capture the highly nonlinear nature in the underlying graph and need inefficient eigen-value decomposition. Deep methods have been proposed to learn graph representations. For example, DeepWalk proposed by Perozzi *et al.* [22] applies local information obtained from truncated random walks to learn representations of vertices in a graph. Grover *et al.* [10] generalize DeepWalk by designing a biased random walk procedure to efficiently explore diverse neighborhoods of vertices. As we focus on cross-modal data, we build three graphs and utilize a deep model to learn the unified embeddings from the three graphs.

3 PROPOSED METHOD

3.1 Problem Formulation

Assume we have n_v images $\mathcal{V} = \{v_i, \cdots, v_{n_v}\}$ and n_t texts $\mathcal{T} = \{t_1, \cdots, t_{n_t}\}$, where each image is represented as a data point $v_i \in \mathbb{R}^{d_v}$, and each text is represented as a data point $t_i \in \mathbb{R}^{d_t}$. Along with the images and texts are their corresponding label vectors. Let $z_{v_i} \in \mathbb{R}^c$ denote the label vector of v_i and $z_{t_i} \in \mathbb{R}^c$ denote the label vector of t_i. Label vector indicates what labels an image or a text has, where more than one entry in z_{v_i} and z_{t_i} could be nonzero. Our goal is to learn mapping functions to project images and texts into a common hidden space that can well preserve or capture their cross-modal semantic relationship. Let $f_v : \mathbb{R}^{d_v} \rightarrow \mathbb{R}^d$ be the mapping function from image feature space to the hidden space. Here d is a parameter specifying the dimensionality of the hidden space. Similarly, let $f_t : \mathbb{R}^{d_t} \rightarrow \mathbb{R}^d$ be the mapping function from text feature space to the hidden space.

Local semantic structure preserving. In the learned hidden space, images and texts with similar meaning should be close. As images and texts share a label space, we build a graph to incorporate all images and texts using their label vectors to encode their high level semantic relations. This is shown as semantic graph in Figure 1. In the following, we use n_i to denote an image or a text, z_{n_i} to denote its label vector, and $f(n_i)$ to denote its embedded representation after the mapping, where f can be f_v or f_t depending on the modality of the input. The semantic graph $G_s = (V_s; E_s)$ is built as follows. First, we set vertex set $V_s = \mathcal{V} \bigcup \mathcal{T}$, *i.e.*, every image or text is a vertex. Then two vertices are connected if they share at least one label. The weight of edge is defined by a similarity function in label space as follows:

$$w_{ij} = \begin{cases} s(z_{n_i}, z_{n_j}), & \text{if } n_i \text{ and } n_j \text{ are connected,} \\ 0, & \text{otherwise.} \end{cases}$$

The similarity function $s(z_{n_i}, z_{n_j})$ assigns a high value to similar label pair, and assigns a low value to dissimilar label pair. Cosine similarity function $s(z_{n_i}, z_{n_j}) = \frac{<z_{n_i}, z_{n_j}>}{\|z_{n_i}\| \|z_{n_j}\|}$ or squared exponential distance based similarity function $s(z_{n_i}, z_{n_j}) = \exp(-\frac{\|z_{n_i} - z_{n_j}\|_2^2}{\sigma})$ can be used [23], where σ is a constant factor. Thus, images and texts are on a large graph containing both intra- and inter-modal semantic relations.

The local semantic structure preserving quality is particularly important as similarity search in the hidden space is performed in the retrieval phase. Therefore, we encourage the learned embeddings to capture the local semantic graph structure. Motivated by Node2vec [10] and Deepwalk [22] which learn latent representations of vertices in a network, we use skip-gram model [21] to learn the local structure of G_s. The skip-gram model seeks to minimize the negative log-probability of observing a neighborhood $P(n_i)$ for each vertex n_i:

$$-\sum_{n_i} \log Pr(P(n_i)|n_i). \tag{1}$$

Assuming that neighborhood vertices are independent of each other given one vertex, and the given vertex and its neighborhood have a symmetric effect over each other, we have

$$Pr(P(n_i)|n_i) = \Pi_{n_j \in P(n_i)} \frac{\exp(f(n_i)^T f(n_j))}{\sum_{n_k \in V_s} \exp(f(n_i)^T f(n_k))}.$$

However, the softmax function requires exhaustive computation of $\exp(f(n_i)^T f(n_k))$ for every training example. As we focus on learning semantically consistent embeddings, we use negative sampling [21] to approximate the softmax function. The aim of negative sampling is to distinguish the target vertex from vertices drawn from a noise distribution using logistic regression. For vertex n_i, the objective becomes to minimize the following loss:

$$-\sum_{n_j \in P(n_i)} \{\log \sigma(f(n_i)^T f(n_j)) + \sum_{n_k \in N(n_i)} \log \sigma(-f(n_i)^T f(n_k))\},$$

where $N(n_i)$ defines the negative samples for vertex n_i. Since our semantic graph incorporate all images and texts and there are intra-modal and inter-modal semantic relations, inner product term $f(n_i)^T f(n_j)$ can be either $f_v(v_i)^T f_t(t_j)$, or $f_v(v_i)^T f_v(v_j)$, or $f_t(t_i)^T f_t(t_j)$.

Now the problem is how to sample neighborhood vertices and negative vertices. Taking account of intra- and inter-modal relations, for each vertex, we sample k_1 vertices from the same modality and k_2 vertices from the other modality to form its neighborhood. When sampling neighborhood vertices for n_i, we sample a set of modality-specific vertices from multinomial distribution with probability in proportion to their edge weights to n_i. Thus vertices with high label correlation to n_i are more likely to be sampled. Since our objective is to enlarge the similarity of relevant pairs while suppressing the similarity of irrelevant pairs, we randomly sample its unconnected vertices as negative samples.

Taking an image v_i as an example, the set of neighborhood images is denoted as P_v, and the set of negative images is denoted as N_v. Similar meanings are applied on notations P_t and N_t. Specifically, Eq. 2 becomes the following loss for v_i,

$$\begin{aligned} l_{G_s}(v_i) = -\sum_{v_j \in P_v(v_i)} &\{\log \sigma(f_v(v_i)^T f_v(v_j)) + \\ \sum_{v_k \in N_v(v_i)} &\log \sigma(-f_v(v_i)^T f_v(v_k))\} - \\ \sum_{t_j \in P_t(v_i)} &\{\log \sigma(f_v(v_i)^T f_t(t_j)) + \\ \sum_{t_k \in N_t(v_i)} &\log \sigma(-f_v(v_i)^T f_t(t_k))\}. \end{aligned} \tag{2}$$

Global semantic structure preserving. If the embeddings contain enough semantic information, they should be able to recover their corresponding label information. The spirit is similar to an Autoencoder [29] which uses reconstruction ability to get hidden representations. As shown in Figure 2, we define a prediction function $g : \mathbb{R}^d \to \mathbb{R}^c$ from the hidden space to the label space which predicts label vector for each point in the hidden space. We aim to minimize the label reconstruction error of embeddings, and the label reconstruction loss term is defined as:

$$l_z(n_i) = \|g(f(n_i)) - z_{n_i}\|_F^2,$$

where $\| \cdot \|_F$ denotes the Frobenius-norm, which means a least-square error is adopted and minimized. The consistency between the recovered labels and ground-truth labels provides global consistency of semantic structure [19].

Local geometric structure preserving. In addition to label information, the geometry in the original feature spaces is also important for propagating the semantic information. Considering that nearby points in the feature space are likely to have the same label [37], we enforce the learned embeddings to preserve local geometric property as well.

As show in Figure 1, we build a graph $G_v = (V_v; E_v)$ on the original visual features and a graph $G_t = (V_t; E_t)$ on the original textual features. Taking G_v as an example, it is defined on $\mathcal{V}$, where the vertex set V_v is $\mathcal{V}$. We connect v_i and v_j if one of them is among the other's k-nearest neighbor by Euclidean distance and define the corresponding weight on the edge as follows:

$$w_{ij} = \begin{cases} 1, & \text{if } v_i \text{ and } v_j \text{ are connected,} \\ 0, & \text{otherwise.} \end{cases}$$

As in semantic graph, our model preserves local structure of the visual graph by predicting nearby vertices of every vertex. But different from semantic graph, this visual graph only involves images. Given a vertex, sampling nearby vertices is done based on the probabilities in proportion to edge weights to the given vertex. Negative vertices are sampled randomly

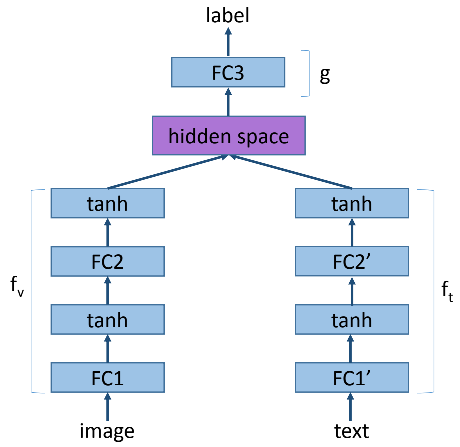


Figure 3: The proposed architecture. Below the hidden space, there are two branches for image and text respectively. These two branches are modality-specific so that different branch has different parameters. Above the hidden space, a mapping connects embeddings in hidden space to label vectors.

from those unconnected to the given vertex. The loss with respect to image v_i in the visual graph is defined as follows:

$$l_{G_v}(v_i) = - \sum_{v_j \in P(v_i)} \{\log \sigma(f_v(v_i)^T f_v(v_j)) + \sum_{v_k \in N(v_i)} \log \sigma(-f_v(v_i)^T f_v(v_k))\}. \tag{3}$$

The local geometric structure preserving term enforces the mapping function to be smooth, and thus can also be considered as a regularization.

The loss function. Combining the local semantic structure consistency, global semantic structure consistency and local geometry structure consistency, the final loss for an image v_i is:

$$l_{G_s}(v_i) + \alpha_1 l_z(v_i) + \alpha_2 l_{G_v}(v_i), \tag{4}$$

where α_1 and α_2 are trade-off parameters determining the relative importance of each loss term. Similarly, the loss for a text t_i is:

$$l_{G_s}(t_i) + \alpha_1 l_z(t_i) + \alpha_2 l_{G_t}(t_i). \tag{5}$$

3.2 Network Structure

In general, there are two mappings (f_v, f_t) projecting images and texts into a hidden space and one mapping g connecting the embeddings to label space. We model the mapping functions with neural networks because of their ability of modeling nonlinearity. As shown in Figure 3, our model has two different branches, one for image mapping and the other for text mapping. Each is composed of fully connected layers followed by tanh nonlinearities. Other nonlinearities can also be used, but we find that tanh which can give positive and negative values leads to better results.

3.3 Training

The model can be easily learned using back-propagation and stochastic gradient descent based technique. The loss function consists of logistic regression terms and linear regression terms, so the gradient can be easily derived. It is worth mentioning that since every image (text) is mapped through the same branch, summation of gradient passed back from every embedding is needed. Algorithm 1 summarizes the proposed algorithm. The outer loop specifies the number of times we sample the neighborhood for each vertex. In the inner loop, we iterate over all images and texts.

Algorithm 1 The SSPE algorithm.

```
Input: images, texts and their corresponding labels,
  sampling times $r$,
  neighborhood size $|P|$ of $G_s$, $G_v$ and $G_t$, negative sample
  size $|N|$ for each neighborhood vertex,
  trade-off parameters $\alpha_1$ and $\alpha_2$
Output: $f_v$, $f_v$ and $g$
  Build graph $G_s$, $G_v$ and $G_t$
  for $i = 1, \dots, r$ do
    for $j = 1, \dots, n_v$ do
      Sample neighbor vertices and negative vertices for
      image $v_j$ in $G_s$ and in $G_v$
    end for
    for $j = 1, \dots, n_t$ do
      Sample neighbor vertices and negative vertices for text
      $t_j$ in $G_s$ and in $G_t$
    end for
    Perform mini-batch stochastic gradient descent to optimize the sum of image and text losses in Eq. 4 and 5
  end for
```

4 EXPERIMENTS

4.1 Datasets

Pascal VOC 2007 [7], collected from Flickr, contains 9963 images. For label representations, we use the ground-truth annotation of the images which have 20 classes. For text representations, we use the 399-dim tag frequency features provided by Hwang and Grauman [15]. For image representations, we use the 4,096-dim CNN image features in 'fc7' extracted using Caffe[16] with the pre-trained CaffeNet learned on ImageNet. The original train-test split provided in the dataset is used for training and testing. After removing images without tags, we get a training set with 5,000 pairs and a test set with 4,919 pairs.

MIRFLICKR dataset [14] contains 25,000 images along with the user assigned tags. Each image-text pair is annotated with some of 38 classes. We use the 2,000-dim tag frequency text features [28], and extract 4,096-dim CNN image features in 'fc7' as on Pascal dataset. Following the training-testing split [14] and removing images without tags, we have 12,144 pairs for training and 7,958 pairs for testing.

NUS-WIDE dataset [4], crawled from the Flickr website, contains 269,648 images associated with their tags. Each image is labeled with 81 underlying semantic concepts. We take the 1,000-dim tag frequency features as text representations, and take the 4,096-dim output of 'fc7' layer from CaffeNet as image features. Original train-test split provided in the dataset is used. By selecting pairs that belong to the 20 largest classes for a reasonable multi-label setting, we get 52,830 pairs for training and 35,216 pairs for testing. The 20 largest classes are used for label representations.

4.2 Compared Approaches

Eight approaches are chosen as the baselines of the evaluations. Here we briefly introduce the eight methods: **CCA** [13] aims at learning a latent space by maximizing the correlation between two modality data; **PLS** [25] creates orthogonal score vectors by maximizing the covariance between two modality data; **GMLDA** [2] combines LDA with CCA to obtain directions which achieve closeness between multi-view samples of the same class; **MMNN** [20] uses a coupled siamese neural network architecture which allows unified treatment of intra- and inter-modality similarity learning to map data into a common space; **ml-CCA** [23] extends CCA by using multi-label information to establish correspondences between two modalities; **LGCFL** [17] performs ε-dragging on the label space, and then performs linear regression to the modified label space with group sparsity constraint. **DSPE** [31] is a two-branch neural network using a large-margin-based objective that combines cross-view ranking constraints with within-view neighborhood structure preservation constraints. **CMOS** [33] is an online learning algorithm which learns a similarity function with bi-directional loss.

4.3 Evaluation

To evaluate the performance of our method, we perform two kinds of two cross-modal retrieval tasks, *i.e.*, the image-to-text task (text retrieval using an image query) and the text-to-image task (image retrieval using a text query). The test set is used as both query set and database set. Here the images and texts used for testing are not present during training which allows us to evaluate the generalization ability of the proposed approach. In the evaluation process, documents that share at least one label with the query are considered to be relevant to the query. The search quality is evaluated with Mean Average Precision (MAP) which is a widely used metric in retrieval task. Given a query and a set of R retrieved documents, Average Precision (AP) is defined as AP= $\frac{\sum_{r=1}^{R} P(r)\delta(r)}{\sum_{r=1}^{R} \delta(r)}$, where $P(r)$ denotes the precision of the top r retrieved documents, and $\delta(r) = 1$ if the r-th retrieved document is relevant with the query and $\delta(r) = 0$ otherwise. We average the AP values over all queries in the query set to obtain the MAP measure. Besides, we present precision-scope curve [30] to further show the performance of different methods. The precision-scope curve shows precision values as a function of the number of returned documents.

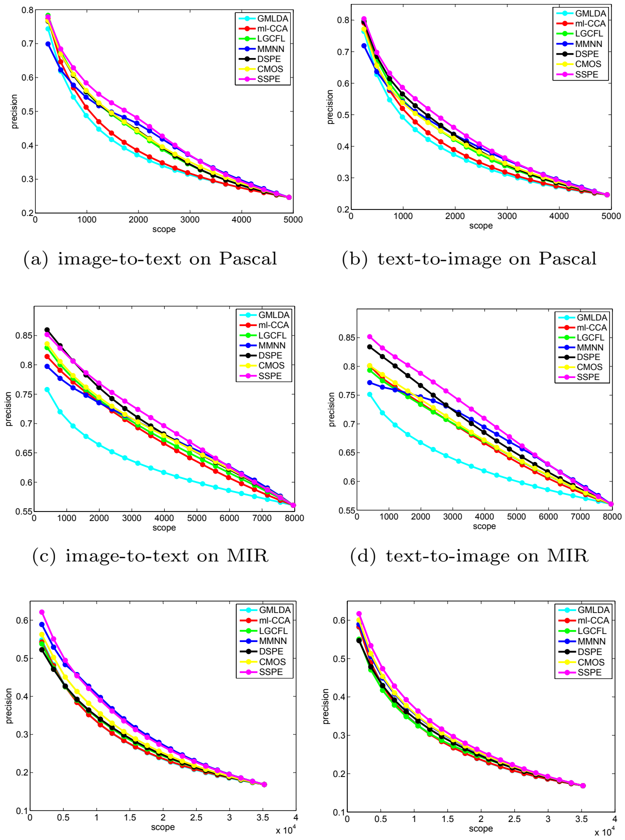
4.4 Implementation Details

All datasets were mean-centered and unit-length normalized. Principle component analysis (PCA) is performed by preserving 95% energy additionally for CCA, GMLDA and ml-CCA, leading to better results [30]. For GMLDA and LGCFL, the category of every training pair is selected randomly from its multiple labels. For fair comparison, we ignore the binarization step for hashing method MMNN. In the testing phase, cosine distance function is adopted for all methods except MMNN that uses Euclidean distance and CMOS that learns similarity function itself. The dimensions of common hidden space for all methods are set to 32 on all datasets, except LGCFL that fixes label space as its common space and CMOS that does not have hidden space. The network configuration of image branch and text branch of MMNN, DSPE and our method are the same. The output dimensions of the first layer and the second layer of image branch and text branch are 64 and 32, respectively. For the compared methods, we use the parameters optimal settings tuned by a parameters validation process except for the specified values.

We implement our model by tensorflow [1]. For our method, the configuration is the same on all the three datasets. We use cosine similarity function to build semantic graph, and select 8 nearest neighbors on feature graphs. The trade-off parameters α_1 and α_2 are set to 0.1. We sample 8 neighborhood vertices for each vertex in each graph. Specifically, on G_s, the eight neighborhood vertices consist of six inter-modal neighborhood vertices and two intra-modal neighborhood vertices. Only one negative vertex for each neighborhood vertex is sampled, since sampling more negative vertices can not improve the performance. The sampling is performed five times to get the final training data. We train our networks using mini-batch stochastic gradient descent with momentum 0.9 and weight decay 0.001. Every batch contains five images and five texts, and their corresponding neighborhood and negative samples. We use a small learning rate starting with 0.01 and decay the learning rate by 0.95 after every 2000 batches.

4.5 Comparison Results

The performance comparison on all three datasets is shown in Table 1 and Figure 4. From the table, we can see that SSPE outperforms other methods with remarkable performance gains. The promising results confirm the effectiveness of SSPE. The performance of SSPE is the best on all three datasets, but the performance of other methods is not stable. DSPE is the second best method on MIRFLICKR, but does not perform well on NUS-WIDE. MMNN performs well on NUS-WIDE but does not perform well on Pascal. This may be caused by their cursory random sampling strategy and the structure difference of different datasets. ml-CCA and GMLDA extend CCA, but ml-CCA performs better than GMLDA, which shows that multi-label information is important. Moreover, by projecting images and texts to label space, LGCFL achieves good results. This verifies the effectiveness of global semantic preserving term. It is easy to see from the results that label vector is very useful for cross-modal retrieval. From the precision-scope curves we can observe that SSPE still outperforms other methods on the whole. Since SSPE considers preserving structure of three graphs simultaneously and can model nonlinear relations between inputs and embeddings, it achieve better results.



(a) image-to-text on Pascal (b) text-to-image on Pascal

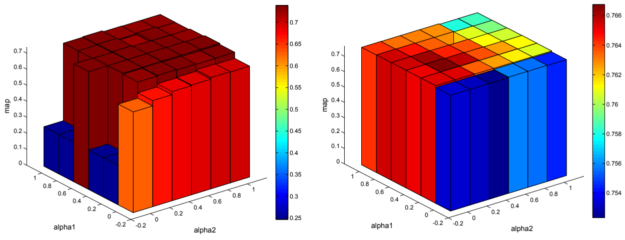
(c) image-to-text on MIR (d) text-to-image on MIR

(e) image-to-text on NUS (f) text-to-image on NUS

Figure 4: Performance of different methods on all benchmark datasets based on precision-scope curve.

4.6 Analysis of Parameters

As α_1 and α_2 adjust the contributions of global semantic structure preserving term and local geometric structure preserving term, respectively, here we take Pascal dataset and MIRFLICKR dataset as examples to analyze the effect of these two parameters. The evaluation is conducted by changing one parameter while fixing the other. The values of α_1 and α_2 are varied in the range $\{0, 0.2, 0.4, 0.6, 0.8, 1\}$. Figure 5 shows the average cross-modal retrieval performance of SSPE in terms of MAP@all for different values of α_1 and α_2. It can be observed from Figure. 5(a) that when $\alpha_2 = 0$, the model is instable. Sometimes the model collapses as it gets NaN value when the training procedure proceeds. The observation indicates that the local geometric structure preserving terms encourage the mapping functions to be smoother, and work like regularization terms. Except for the collapsed situation, the performance of SSPE is stable as the average performance does not vary much. This demonstrates that our method is not sensitive to the trade-off parameters. But if we carefully compare the relative color and height of each bin, we can find that global semantic structure preserving term can improve the performance. But the maximum value is obtained by setting α_1 and α_2 to middle value.



(a) performance on Pascal. (b) performance on MIRFLICKR.

Figure 5: Average cross-modal retrieval performance of SSPE with different values of α_1 and α_2 on Pascal and MIRFLICKR.

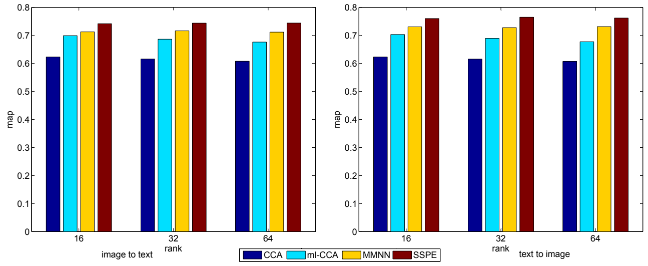


Figure 6: Performance of different methods with different ranks (the number of the latent dimensions) on MIRFLICKR.

4.7 Effect of hidden space dimensionality

Furthermore, we show the MAP scores of different methods on MIRFLICKR dataset with varying ranks (*i.e.*, the dimensionality of the latent space) in Figure 6. We can see that the performance of CCA and ml-CCA is slightly worse as the dimension increasing, while the performance of SSPE and MMNN is stable with varied ranks. These results show that the dimension of the latent space has little influence on the performance of our method. Moreover, our method outperforms other methods on all the ranks.

4.8 Effect of neighborhood size

We report the MAP scores when sampling different numbers of neighborhood vertices in Figure 7. On the semantic graph, the proportion of intra-modal neighborhood vertices is fixed to 25%. It can be seen that with the increasing number of neighborhood vertices, the quality of the learned embeddings first increases and then decreases. Small number of neighborhood vertices may not provide enough information to learn,

Table 1: Performance comparison in terms of MAP (%) on Pascal VOC 2007, MIRFLICKR and NUS-WIDE. The best results are shown in boldface.

Dataset	Direction	CCA	PLS	GMLDA	MMNN	ml-CCA	LGCFL	DSPE	CMOS	SSPE
Pascal	img-to-txt	65.9±0.0	63.3±0.0	67.3±0.1	69.7±0.2	69.8±0.0	73.2±0.4	72.7±0.2	72.4±0.1	**74.5±0.1**
	txt-to-img	64.5±0.0	50.1±0.0	66.4±0.1	64.7±0.2	69.7±0.0	72.3±0.1	71.7±0.1	70.0±0.2	**74.1±0.2**
MIR-FLICKR	img-to-txt	65.2±0.0	66.2±0.0	65.0±0.1	71.2±0.5	70.8±0.0	72.3±0.4	72.8±0.1	72.7±0.1	**74.9±0.3**
	txt-to-img	65.3±0.0	66.1±0.0	65.1±0.1	73.1±0.4	71.3±0.0	72.5±0.5	73.9±0.3	72.8±0.4	**76.6±0.2**
NUS-WIDE	img-to-txt	40.9±0.0	50.3±0.0	50.7±0.1	57.7±0.1	51.6±0.0	51.6±0.1	48.2±0.2	53.3±0.1	**58.5±0.4**
	txt-to-img	37.7±0.0	45.6±0.0	45.9±0.1	47.3±0.5	43.5±0.0	44.0±0.7	43.7±0.2	48.5±0.1	**50.4±0.3**

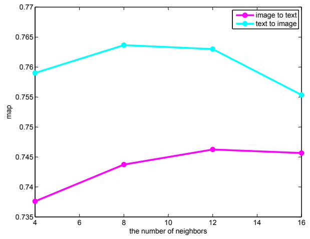
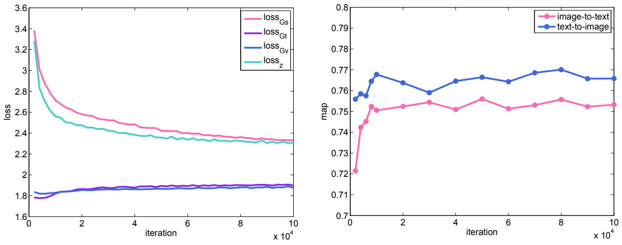


Figure 7: Performance of SSPE with different numbers of neighborhood vertices on MIRFLICKR.



(a) Changes of losses. (b) Changes of MAP scores.

Figure 8: Changes of losses and MAP scores as a function of training iterations on MIRFLICK.

while large number of neighborhood vertices may lead to overfitting to specific vertices. We also conducted experiment to see how the number of negative vertices for each neighborhood vertex influences the performance of our method. We find that increasing the number of negative samples cannot boost the performance. Therefore, in our experiment, only one negative sample is sampled for each neighborhood vertex.

4.9 Analysis of losses

As our method consists of three terms, *i.e.*, local semantic structure preserving term, global semantic structure preserving term and local geometric structure preserving term, it is interesting to see how the losses of the three terms vary as the training procedure proceeding and how the retrieval performance changes. We show experiment results of losses and MAP scores as a function of training iterations on MIRFLICKR dataset. The experiment result is shown in Figure 8. The losses shown are averaged over 2000 iterations.

It is apparent from the Figure 8(a) that local semantic structure preserving loss ($i.e.$, $loss_{G_s}$) and global semantic structure preserving loss ($i.e.$, $loss_z$) decrease dramatically at first and decline slowly after $10,000$ iterations. However, local geometric structure preserving losses ($i.e.$, $loss_{G_v}$ and $loss_{G_t}$) increase a little and flatten out then. This indicates that original visual space and textual space contain noise not in accordance with semantic information. But as shown in subsection 4.6, information contained in original visual space and textual space is important. The local geometric structure preserving terms work like regularization terms.

It can be clearly seen from Figure 8(b) that the retrieval performance increases fast at first and then stays stable. The result indicates that our method can achieve good results by only training modest iterations ($i.e.$, $10,000$ iterations). Combined with Figure 8(a), we can see that the retrieval performance remains steady even though the local semantic structure preserving loss and global semantic structure preserving loss decrease.

5 CONCLUSION

In this paper, we propose SSPE, which learns semantic structure-preserved embeddings for multi-label cross-modal retrieval. The local and global semantic structure consistencies as well as the local geometric structure consistency are enforced simultaneously. The three terms fully exploits the intrinsic relevance between cross-modal data. Nonlinear neutral network is applied to model the mapping functions between inputs and embeddings. The overall objective function is optimized by stochastic gradient descent to gain the scalability on large datasets. Comprehensive experiments are conducted on three public cross-modal datasets. Experimental results compared with eight methods have demonstrated that the proposed method outperforms state-of-the-art approaches. In future work, we will study efficient cross-modal embedding learning method with a finer-scale visual-textual association, *e.g.*, the object-word and region-phrase association.

ACKNOWLEDGMENT

This work was supported in part by National Natural Science Foundation of China: 61672497, 61332016, 61620106009, 61650202 and U1636214, in part by National Basic Research Program of China (973 Program): 2015CB351802 and in part by Key Research Program of Frontier Sciences of CAS: QYZDJ-SSW-SYS013.

REFERENCES

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. 2016. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467* (2016).

[2] S. Abhishek, K. Abhishek, H. Daume, and D. W. Jacobs. 2012. Generalized multiview analysis: A discriminative latent space. In *CVPR*. 2160–2167.

[3] Galen Andrew, Raman Arora, Jeff Bilmes, and Karen Livescu. 2013. Deep canonical correlation analysis. In *ICML*. 1247–1255.

[4] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng. 2009. NUS-WIDE: a real-world web image database from National University of Singapore. In *CIVR*. 48.

[5] Dragoš M Cvetković, Peter Rowlinson, and Slobodan Simic. 1997. *Eigenspaces of graphs*. Number 66. Cambridge University Press.

[6] C. Deng, J. Tang, X.and Yan, W. Liu, and X. Gao. 2016. Discriminative Dictionary Learning With Common Label Alignment for Cross-Modal Retrieval. *TMM* 18, 2 (2016), 208–218.

[7] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. 2010. The pascal visual object classes (voc) challenge. *IJCV* 88, 2 (2010), 303–338.

[8] Fangxiang Feng, Xiaojie Wang, and Ruifan Li. 2014. Cross-modal retrieval with correspondence autoencoder. In *ACM MM*. 7–16.

[9] Yunchao Gong, Qifa Ke, Michael Isard, and Svetlana Lazebnik. 2014. A multi-view embedding space for modeling internet images, tags, and their semantics. *IJCV* 106, 2 (2014), 210–233.

[10] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *KDD*. 855–864.

[11] Xiaofei He and Partha Niyogi. 2002. Locality Preserving Projections (LPP). *NIPS* 16, 1 (2002), 186–197.

[12] Yonghao He, Shiming Xiang, Cuicui Kang, Jian Wang, and Chunhong Pan. 2016. Cross-modal retrieval via deep and bidirectional representation learning. *TMM* 18, 7 (2016), 1363–1377.

[13] H. Hotelling. 1936. Relations between two sets of variates. *Biometrika* 28, 3/4 (1936), 321–377.

[14] Mark J Huiskes and Michael S Lew. 2008. The MIR flickr retrieval evaluation. In *Proceedings of the 1st ACM international conference on Multimedia information retrieval*. 39–43.

[15] S. J. Hwang and K. Grauman. 2010. Accounting for the Relative Importance of Objects in Image Retrieval.. In *BMVC*, Vol. 1. 5.

[16] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. 2014. Caffe: Convolutional architecture for fast feature embedding. In *ACM MM*. 675–678.

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

[18] Shaishav Kumar and Raghavendra Udupa. 2011. Learning hash functions for cross-view similarity search. In *IJCAI*, Vol. 22. 1360.

[19] Zechao Li, Jing Liu, Jinhui Tang, and Hanqing Lu. 2015. Robust structured subspace learning for data representation. *TPAMI* 37, 10 (2015), 2085–2098.

[20] J. Masci, M. M. Bronstein, A. M. Bronstein, and J. Schmidhuber. 2014. Multimodal similarity-preserving hashing. *TPAMI* 36, 4 (2014), 824–830.

[21] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*. 3111–3119.

[22] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations. In *KDD*. 701–710.

[23] V. Ranjan, N. Rasiwasia, and CV Jawahar. 2015. Multi-Label Cross-modal Retrieval. In *ICCV*. 4094–4102.

[24] N. Rasiwasia, J. Costa Pereira, E. Coviello, G. Doyle, G. RG Lanckriet, R. Levy, and N. Vasconcelos. 2010. A new approach to cross-modal multimedia retrieval. In *ACM MM*. 251–260.

[25] R. Rosipal and N. Krämer. 2006. Overview and recent advances in partial least squares. In *Subspace, latent structure and feature selection*. Springer, 34–51.

[26] Guoli Song, Shuhui Wang, Qingming Huang, and Qi Tian. 2015. Similarity gaussian process latent variable model for multi-modal data analysis. In *ICCV*. 4050–4058.

[27] Jingkuan Song, Yang Yang, Yi Yang, Zi Huang, and Heng Tao Shen. 2013. Inter-media hashing for large-scale retrieval from heterogeneous data sources. In *SIGMOD*. 785–796.

[28] Nitish Srivastava and Ruslan R Salakhutdinov. 2012. Multimodal learning with deep boltzmann machines. In *NIPS*. 2222–2230.

[29] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. 2008. Extracting and composing robust features with denoising autoencoders. In *ICML*. 1096–1103.

[30] K. Wang, R. He, W. Wang, L. Wang, and T. Tan. 2013. Learning coupled feature spaces for cross-modal matching. In *ICCV*. 2088–2095.

[31] Liwei Wang, Yin Li, and Svetlana Lazebnik. 2016. Learning deep structure-preserving image-text embeddings. In *CVPR*. 5005–5013.

[32] Fei Wu, Xinyan Lu, Zhongfei Zhang, Shuicheng Yan, Yong Rui, and Yueting Zhuang. 2013. Cross-media semantic representation via bi-directional learning to rank. In *ACM MM*. 877–886.

[33] Yiling Wu, Shuhui Wang, and Qingming Huang. 2017. Online Asymmetric Similarity Learning for Cross-Modal Retrieval. In *CVPR*. 4269–4278.

[34] Fei Yan and Krystian Mikolajczyk. 2015. Deep correlation for matching images and text. In *CVPR*. 3441–3450.

[35] Shuicheng Yan, Dong Xu, Benyu Zhang, and Hong-Jiang Zhang. 2005. Graph embedding: A general framework for dimensionality reduction. In *CVPR*, Vol. 2. 830–837.

[36] Ting Yao, Tao Mei, and Chong-Wah Ngo. 2015. Learning query and image similarities with ranking canonical correlation analysis. In *ICCV*. 28–36.

[37] Dengyong Zhou, Olivier Bousquet, Thomas Navin Lal, Jason Weston, and Bernhard Schölkopf. 2003. Learning with local and global consistency.. In *NIPS*, Vol. 16. 321–328.