Ranking-Based Deep Cross-Modal Hashing

Xuanwu Liu,[1] **Guoxian Yu,**[1,2*] **Carlotta Domeniconi,**[3] **Jun Wang,**[1] **Yazhou Ren,**[4] **Maozu Guo**[5]

[1]College of Computer and Information Sciences, Southwest University, Chongqing, China
[2]Hubei Key Laboratory of Intelligent Geo-Information Processing, China University of Geosciences, Wuhan, China
[3]Department of Computer Science, George Mason University, Fairfax, USA
[4]SMILE Lab, School of Computer Science and Engineering,
University of Electronic Science and Technology of China, Chengdu, China
[5]School of Electrical and Information Engineering, Beijing University of Civil Engineering and Architecture, Beijing, China
Email: {alxw1007,gxyu,kingjun}@swu.edu.cn, carlotta@cs.gmu.edu, yazhou.ren@uestc.edu.cn, guomaozu@bucea.edu.cn

Abstract

Cross-modal hashing has been receiving increasing interests for its low storage cost and fast query speed in multi-modal data retrievals. However, most existing hashing methods are based on hand-crafted or raw level features of objects, which may not be optimally compatible with the coding process. Besides, these hashing methods are mainly designed to handle simple pairwise similarity. The complex multilevel ranking semantic structure of instances associated with multiple labels has not been well explored yet. In this paper, we propose a ranking-based deep cross-modal hashing approach (RDCMH). RDCMH firstly uses the feature and label information of data to derive a semi-supervised semantic ranking list. Next, to expand the semantic representation power of hand-crafted features, RDCMH integrates the semantic ranking information into deep cross-modal hashing and jointly optimizes the compatible parameters of deep feature representations and of hashing functions. Experiments on real multi-modal datasets show that RDCMH outperforms other competitive baselines and achieves the state-of-the-art performance in cross-modal retrieval applications.

Introduction

With the explosive growth of data, how to efficiently and accurately retrieve the required information from massive data becomes a hot research topic and has various applications. For example, in information retrieval, approximate nearest neighbor (ANN) search (Andoni and Indyk 2006) plays a fundamental role. Hashing has received increasing attention due to its low storage cost and fast retrieval speed for ANN search (Kulis and Grauman 2010). The main idea of hashing is to convert the high-dimensional data in the ambient space into binary codes in the low-dimensional Hamming space, while the proximity between data in the original space is preserved in the Hamming space(Wang et al. 2016; 2018; Shao et al. 2016). By using binary hash codes to represent the original data, the storage cost can be dramatically reduced. In addition, we can use hash codes to construct an index and achieve a constant or sub-linear time complexity for ANN

search. Hence, hashing has become more and more popular for ANN search on large scale datasets.

In many applications, the data can have multi-modalities. For example, a web page can include not only a textual description but also images and videos to illustrate its contents. These different types (views) of data are called *multi-modal* data. With the rapid growth of multi-modal data in various applications, multi-modal hashing has recently been widely studied. Existing multi-modal hashing methods can be divided into two main categories: mutli-source hashing (MSH) and cross-modal hashing (CMH)(Zhu et al. 2013). The goal of MSH is to learn hash codes by utilizing all the information from multiple modalities. Hence, MSH requires all the modalities observed for all data points, including query points and those in the database. In practice, it is often difficult or even infeasible to acquire all data points across all the modalities, as such the application of MSH is limited. On the contrary, the application scenarios of CMH are more flexible and practical. In CMH, the modality of a query point can be different from the modality of the points in the database. In addition, the query point typically has only one modality and the points in the database can have one or more modalities. For example, we can use text queries to retrieve images from the database, and we can also use image queries to retrieve texts from the database. Due to its wide application, CMH has attracted increasing attention (Kumar and Udupa 2011; Zhang and Li 2014).

Many CMH methods have been proposed recently, existing CMH methods can be roughly divided into two categories: supervised and unsupervised. Unsupervised approaches seek hash coding functions by taking into account underlying data structures, distributions, or topological information. To name a few, Canonical correlation analysis (Rasiwasia et al. 2010) maps two modalities, such as visual and textual, into a common space by maximizing the correlation between the projections of the two modalities. Inter-media hashing (Song et al. 2013) maps view-specific features onto a common Hamming space by learning linear hash functions with intra-modal and inter-modal consistencies. Supervised approaches try to leverage supervised information (i.e., semantic labels) to improve the performance. Cross-modal similarity sensitive hashing (CMSSH) (Bronstein et al. 2010) regards the hash

codes learning as binary classification problems, and efficiently learns the hash functions using a boosting method. Co-regularized hashing (Yi and Yeung 2012) learns a group of hash functions for each bit of binary codes in every modal. Semantic correlation maximization (SCM) (Zhang and Li 2014) optimizes the hashing functions by maximizing the correlation between two modalities with respect to the semantic labels. Semantics Preserving Hashing (SePH)(Lin et al. 2017) generates one unified hash code for all observed views of any instance by considering the semantic consistency between views.

Most supervised hashing are pairwise supervised methods, which leverage labels of instances and pairwise labels of instance-pairs to train the coding functions, such that the label information can be preserved in the Hamming space (Chang et al. 2012). Their objectives, however, may be suboptimal for ANN search, because they do not fully explore the high-order ranking information (Song et al. 2015). For example, a triplet rank contains a query image, a positive image, and a negative image, where the positive image is more similar to the query image than the negative image (Lai et al. 2015). High-order ranking information carries relative similarity ordering in the triplets and provides richer supervision, it often can be more easily obtained than pairwise ranking. Some hashing methods consider the high-order ranking information for hashing learning. For example, deep semantic ranking based hashing (Zhao et al. 2015) learns deep hash functions based on CNN (Convolutional neural network)(Krizhevsky, Sutskever, and Hinton 2012), which preserves the semantic structure of multi-label images. Simultaneous feature learning and hash coding (Lai et al. 2015) generates bitwise hash codes for images via a carefully designed deep architecture and uses a triplet ranking loss to preserve relative similarities.

However, these semantic ranking methods just consider one modality, and cannot apply to cross-modal retrieval. Besides, the ranking lists are just simply computed by the number of shared labels, which could not preserve the integral ranking information of labels. Furthermore, the ranking lists adopted by these methods ask for sufficient labeled training data, and cannot make use of abundant unlabeled data, whose multi-modal feature information can boost the cross-modal hashing performance. Semi-supervised hashing methods were introduced to leverage both labeled and unlabeled samples (Wang, Kumar, and Chang 2012), but these methods cannot directly be applied on multi-modal data. Almost all these CMH methods are based on hand-crafted (or raw-level) features. One drawback of these hand-crafted feature based methods is that the feature extraction procedure is isolated from the hash-code learning procedure, or the original raw-level features can not reflect the semantic similarity between objects very well. The hand-crafted features might not be optimally compatible with the hash-code learning procedure(Cao et al. 2017). As a result, these CMH methods can not achieve satisfactory performance in real applications.

Recently, deep learning has also been utilized to perform feature learning from scratches with promising performance. Deep cross-modal hashing(DCMH)(Jiang and Li 2017) combines the deep feature learning with cross-modal retrieval and guides deep learning procedure with multi-labels of multi-modal objects. Correlation auto-encoder hashing (Cao et al. 2016) adopts deep learning for uni-modal hashing. Their studies show that the end-to-end deep learning architecture is more compatible for hashing learning. However, they still ask for sufficient label information of training data, and treat the parameters of the hash quantization layer and those of deep feature learning layers as the same, which may reduce the discriminative power of the quantification process.

In this paper, we propose a ranking-based deep cross-modal hashing (RDCMH), for cross-modal retrieval applications. RDCMH firstly uses the feature and label information of data to derive a semi-supervised semantic ranking list. Next, it integrates the semantic ranking information into deep cross-modal hashing and jointly optimizes the ranking loss and hashing codes functions to seek optimal parameters of deep feature representations and those of hashing functions. The main contributions of RDCMH are outlined as follows:

1. A novel cross-modal hash function learning framework (RDCMH) is proposed to integrate deep feature learning with semantic ranking to address the problem of preserving semantic similarity between multi-label objects for cross-modal hashing; and a label and feature information induced semi-supervised semantic ranking metric is also introduced to leverage labeled and unlabeled data.

2. RDCMH jointly optimizes the deep feature extraction process and the hash quantization process to make feature learning procedure being more compatible with the hash-code learning procedure, and this joint optimization indeed significantly improves the performance.

3. Experiments on benchmark multi-modal datasets show that RDCMH outperforms other baselines (Bronstein et al. 2010; Zhang and Li 2014; Lin et al. 2017; Jiang and Li 2017; Cao et al. 2016) and achieves the state-of-the-art performance in cross-modal retrieval tasks.

The Proposed Approach

Suppose $\mathbf{X} = \{x_1, x_2, \cdots, x_n\} \in \mathbb{R}^{n \times d_X}$ and $\mathbf{Y} = \{y_1, y_2, \cdots, y_n\} \in \mathbb{R}^{n \times d_Y}$ are two data modalities, n is the number of instances (data points), $d_X(d_Y)$ is the dimensionality of the instances in the respective modality. For example, in the Wiki-image search application, x_i is the image features of the entity i, and y_i is the tag features of this entity. $\mathbf{Z} \in \mathbb{R}^{n \times m}$ stores the label information of n instances in $\mathbf{X}$ and $\mathbf{Y}$ with respect to m distinct labels. $z_{ik} \in \{0, 1\}$, $z_{ik} = 1$ indicates that x_i is labeled with the k-th label; $z_{ik} = 0$ otherwise. Without loss of generality, suppose the first l samples have known labels, whereas other $u = n - l$ samples lack label information. To enable cross-modal hashing, we need to learn two hashing functions, $F_1 \colon \mathbb{R}^{d_1} \to \{0, 1\}^c$ and $F_2 \colon \mathbb{R}^{d_2} \to \{0, 1\}^c$, where c is the length of binary hash codes. These two hashing functions are expected to map the feature vectors in the respective modality onto a common Hamming space and to preserve the proximity of the original data.

RDCMH mainly involves with two steps. It firstly measures the semantic ranking between instances based on the label and feature information. Next, it defines an objective function to simultaneously account for semantic ranking, deep feature learning and hashing coding functions learning;

and further introduces an alternative optimization procedure to jointly optimize these learning objectives. The overall workflow of RDCMH is shown in Fig. .

Semi-supervised Semantic Ranking

To preserve the semantic structure, we can force the ranking order of neighbors computed by the Hamming distance being consistent with that derived from semantic labels in terms of ranking evaluation measures. Suppose q is a query point, the semantic similarity level of a database point x with respect to q can be calculated based on ranking order of label information. Then we can obtain a ground-truth ranking list for q by sorting the database points in decreasing order of their similarity levels (Zhao et al. 2015; Song et al. 2015). However, this similarity level is just simply derived from the number of shared labels and these semantic ranking-based methods ignore that the labels of training data are not always readily available. Furthermore, these methods work on one modality and can not directly apply on multi-modal data.

To alleviate the issue of insufficient labeled data, we introduce a semi-supervised semantic measure that takes into account both the label and feature information of training data. The labels of an instance depend on the features of this instance, and the semantic similarity is positively correlated with the feature similarity of respective instances (Zhang and Zhou 2010; Wang et al. 2009). The semi-supervised semantic measure is defined as follows:

$$s_{ij}^{xx} = \begin{cases} s_{ij}^1 e^{(s_{ij}^2 - s_{ij}^1)}, & |z_i|^2 \neq 0 \text{ and } |z_j|^2 \neq 0 \\ s_{ij}^1, & \text{otherwise} \end{cases} \quad (1)$$

where s_{ij}^1 is the feature similarity of x_i and x_j, while s_{ij}^2 is the label similarity, both of them are computed by the cosine similarity. Note, s_{ij}^{xx} is always in the interval [0,1] and other similarity metrics can also be used. Eq. (1) can account for both the labeled and unlabeled training data. Specifically, for two unlabeled data, the similarity between x_i and x_j is directly computed from the feature information of the respective data. For labeled data, we consider that the label similarity s_{ij}^2 is a supplement to s_{ij}^1. The larger the s_{ij}^2 is, the larger the s_{ij}^{xx} is. In this way, we leverage the label and feature information of training data to account for insufficient labels.

Extending the ranking order to the cross-modal case, we should keep the semantic structure both in the inter-modality and intra-modality. Based on $\mathbf{S}^{xx} \in \mathbb{R}^{n \times n}$, we can obtain a ranking list $\{x_{q_k}\}_{k=1}^n$ for q by sorting the database points in decreasing order of $s_{q_k}^{xx}$. Similarly, we can define the semi-supervised semantic similarity $\mathbf{S}^{yy} \in \mathbb{R}^{n \times n}$ for the data modality $\mathbf{Y}$. To balance the inconsistence of ranking list between two modalities, the semi-supervised semantic similarity is averaged as: $\mathbf{S}^{xy} = \mathbf{S}^{yx} = (\mathbf{S}^{xx} + \mathbf{S}^{yy})/2$. Finally, we can obtain three different ranking lists: $\{r_i^x\}_{i=1}^n$, $\{r_i^y\}_{i=1}^n$, $\{r_i^{xy}\}_{i=1}^n$ for each query point.

Unified Objective Function

Deep Feature Representation   Most existing hashing methods first extract hand-crafted visual features (like GIST

and SIFT) from images and then learn 'shallow' (usually linear) hash functions upon these features (Bronstein et al. 2010; Zhang and Li 2014; Lin et al. 2017). However, these hand-crafted features have limited representation power and may lose key semantic information, which is important for similarity search. Here we consider designing deep hash functions using CNN (Krizhevsky, Sutskever, and Hinton 2012) to jointly learn feature representations and their mappings to hash codes. This non-linear hierarchical hash function has more powerful learning capability than the shallow one based on features crafted in advance, and thus is able to learn more suitable feature representations for multilevel semantic similarity search. Other representation learning models (i.e., AlexNet) can also be used to learn deep features of images and text for RDCMH. The feature learning part contains two deep neural networks, one for image modality and the other for text modality.

The adopted deep neural network for image modality is a CNN, which includes eight layers. The first six layers are the same as those in CNN-F(Chatfield et al. 2014). The seventh and eighth layer is a fully-connected layer with the outputs being the learned image features. As to the text modality, we first represent each text as a vector with bag-of-words (BOW) representation. Next, the bag-of-words vectors are used as the inputs for a neural network with two fully-connected layers, denoted as "full1 - full2". The "full1" layer has 4096 neurons, and the second layer "full2" has c (hashing codes) neurons, The activation function for the first layer is ReLU, and that for the second layer is the identity function.

For presentation, we represent the learnt deep feature representations of x and y as $\varphi(x)$ and $\phi(y)$. The non-linear mapping parameters of these two representations will be discussed later.

Triplet Ranking Loss and Quantitative Loss   Directly optimizing the ranking criteria for cross-modal hashing is very hard. Because it is very difficult to compare the ranking lists and stringently comply with the lists. To circumvent this problem, we use a triplet ranking loss as the surrogate loss. Given a query q and a ranking list $\{r_{q_i}^x\}_{i=1}^n$ for q, we can define a ranking loss on a set of triplets of hash codes as follows:

$$L(h(q), h(\{r_{q_i}^x\}_{i=1}^n)) = \sum_{i=1}^n \sum_{j:s_{qi}>s_{qj}} [\delta dH(h(q), h(i), h(j))]_+ \quad (2)$$

where n is the length of the ranking list, s_{qi} and s_{qj} are the similarity between query q and x_i and x_j, respectively. $h(x)$ represents the learnt hash codes of x, $[x]_+ = max(0, x)$, $\delta dH(a_1, a_2, a_3) = dH(a_1, a_2) - dH(a_1, a_3)$, $dH(\cdot)$ is the Hamming distance. This triplet ranking loss is a convex upper bound on the pairwise disagreement, it counts the number of incorrectly ranked triplets.

Eq. (2) equally treats all triplets, but two samples (x_i and x_j) of a triplet may have different similarity levels to the query q. So we introduce the weighted ranking triplet loss based on the ranking list as follows:

$$L(h(q), h(\{r_{q_i}^x\}_{i=1}^n)) = \sum_{i=1}^n \sum_{j:s_{qi}>s_{qj}} (1 - s_{ij}^{xx})[\delta dH(h(q), h(i), h(j))]_+ \quad (3)$$
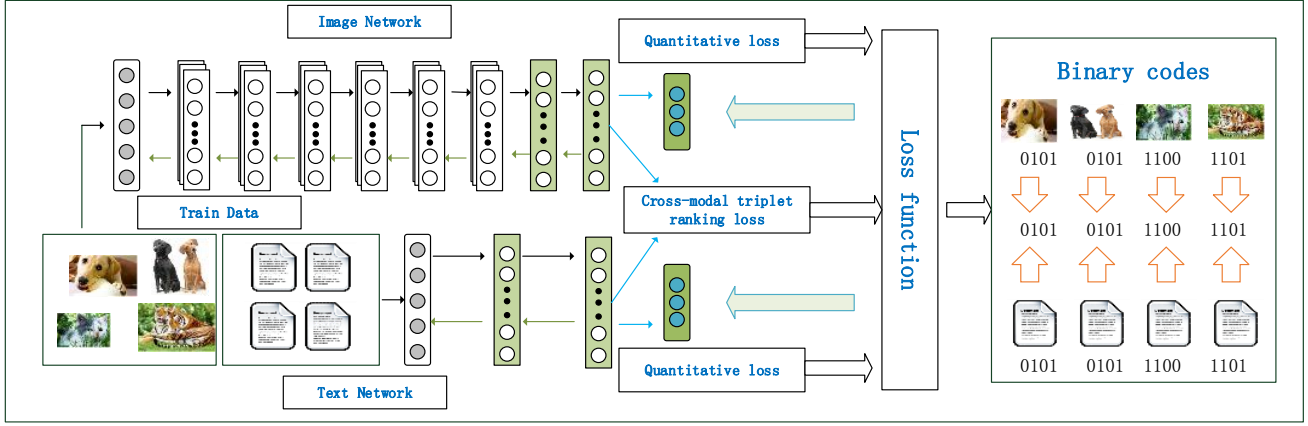
Figure 1: Workflow of the proposed Rank based Deep Cross-Modal Hashing (RDCMH). RDCMH encompasses two steps: (1) an Image CNN network for learning image representations and a Text two-layer Network for learning text representations. (2) Jointly optimize the cross-modal triplet ranking loss and the quantitative loss to seek optimal parameters of deep feature representations and those of hashing functions.

The larger the relevance between x_i and q than that between x_j and q is, the larger the ranking loss results in, if x_i is ranked behind x_j for q.

As to the cross-modal case, we should balance the inconsistence of ranking lists between two modalities. To this end, we give the unified objective function that simultaneously account for triplet ranking loss and quantitative loss as follows:

$$
\min_{\mathbf{W}_x, \mathbf{W}_y, \mathbf{B}} L = \sum_{q \in \mathcal{Q}} \sum_{i,j=1}^{n} [(1 - s_{ij}^{xx})(\delta dH_{xx}) + (1 - s_{ij}^{yy})
$$
$$
(\delta dH_{yy}) + (1 - s_{ij}^{yx})(\delta dH_{yx}) + (1 - s_{ij}^{xy})(\delta dH_{xy})]
$$
$$
+ \frac{\lambda}{2}(||\mathbf{B}^x - \mathbf{F}||_F^2 + ||\mathbf{B}^y - \mathbf{G}||_F^2)
$$

(4)

where

$$
\delta dH_{xx} = dH(h(x_q), h(x_i)) - dH(h(x_q), h(x_j))
$$
$$
\delta dH_{yy} = dH(h(y_q), h(y_i)) - dH(h(y_q), h(y_j))
$$
$$
\delta dH_{xy} = dH(h(x_q), h(x_i)) - dH(h(y_q), h(y_j))
$$
$$
\delta dH_{yx} = dH(h(y_q), h(y_i)) - dH(h(x_q), h(x_j))
$$

(5)

$$
h(x) = h(\varphi(x); \mathbf{W}_x) = sign(\mathbf{W}_x^T \varphi(x))
$$
$$
h(y) = h(\phi(y); \mathbf{W}_y) = sign(\mathbf{W}_y^T \phi(y))
$$

(6)

$$
\mathbf{F}_{*i} = (\varphi(x_i); \mathbf{W}_x), \quad \mathbf{G}_{*i} = (\phi(y_i); \mathbf{W}_y)
$$
$$
\mathbf{B}^x \in \{-1, +1\}^{n \times c}, \quad \mathbf{B}^y \in \{-1, +1\}^{n \times c}
$$

(7)

$\mathcal{Q}$ is the set of query points, $\varphi(x)$ and $\phi(y)$ are the deep features of images and texts, $\mathbf{W}_x$ and $\mathbf{W}_y$ are the coefficient matrices of two modalities, respectively. λ is the scalar parameter to balance the triplet ranking loss and quantitative loss. $\mathbf{B}^x$ and $\mathbf{B}^y$ are the binary hash codes for image and text modality, respectively. In the training process, since different modality data of the same sample share the same label set, and they actually represent the same sample from different viewpoints, we fix the binary codes of same training points from two modalities as the same, namely $\mathbf{B}^x = \mathbf{B}^y = \mathbf{B}$.

Eq. (4) simultaneously accounts for the triplet ranking loss and the quantitative loss. The first term enforces the consistency of cross-modal ranking list by minimizing the number of incorrectly ranked triplets, and the second term (weighted by λ) measures the quantitative loss of hashing. $\mathbf{F}$ and $\mathbf{G}$ can preserve the cross-modal similarity in $\mathbf{S}^{xx}$, $\mathbf{S}^{yy}$ and $\mathbf{S}^{xy}$, as a result, binary hash codes $\mathbf{B}^x$ and $\mathbf{B}^y$ can also preserve these cross-modal similarities. This exactly coincides with the goal of cross-modal hashing.

Optimization   We can solve Eq. (4) via the Alternating Direction Method of Multipliers (ADMM) (Boyd et al. 2011), which alternatively optimizes one of $\mathbf{W}_x$, $\mathbf{W}_y$, and $\mathbf{B}$, while keeping the other two fixed.

Optimize $\mathbf{W}_x$ with $\mathbf{W}_y$ and $\mathbf{B}$ fixed: We observe that the loss function in Eq. (4) is actually a summation of weighted triplet losses and the quantitative loss. Like most existing deep learning methods, we utilize stochastic gradient descent (SGD) to learn $\mathbf{W}_x$ with the back-propagation (BP) algorithm. In order to facilitate the gradient computation, we rewrite the Hamming distance as the form of inner product: $dH(h(a), h(b)) = \frac{c - h(a)^T h(b)}{2}$, where c is the number of hash bits.

More specifically, in each iteration we sample a mini-batch of points from the training set and then carry out our learning algorithm based on the triplet data. For any triplet $(q; x_i; x_j)$, the derivative of Eq. (4) with respect to coefficient matrix $\mathbf{W}_x$ in the data modality $\mathbf{X}$ is given by:

$$
\frac{\partial L}{\partial \mathbf{F}_{*q}} = \frac{1}{2}[(1 - s_{ij}^{xx})(h(\mathbf{F}_{*i}) - h(\mathbf{F}_{*j})) + (1 - s_{ij}^{xy})
$$
$$
(h(\mathbf{F}_{*i})) + (1 - s_{ij}^{yx})(-h(\mathbf{F}_{*j}))] + \lambda(\mathbf{F}_{*q} - \mathbf{B}_{*q})
$$

(8)

$$
\frac{\partial L}{\partial \mathbf{F}_{*i}} = -\frac{1}{2}[(1 - s_{ij}^{xx})(h(\mathbf{F}_{*q}))
$$
$$
+ (1 - s_{ij}^{xy})(h(\mathbf{F}_{*q}))] + \lambda(\mathbf{F}_{*i} - \mathbf{B}_{*i})
$$

(9)

$$\frac{\partial L}{\partial \mathbf{F}_{*j}} = \frac{1}{2}[(1 - s_{ij}^{xx})(h(\mathbf{F}_{*q}))$$
$$+ (1 - s_{ij}^{xy})(h(\mathbf{F}_{*q}))] + \lambda(\mathbf{F}_{*j} - \mathbf{B}_{*j}) \quad (10)$$

We can compute $\frac{\partial L}{\partial \mathbf{W}_x}$ with $\frac{\partial L}{\partial \mathbf{F}_{*q}}$, $\frac{\partial L}{\partial \mathbf{F}_{*i}}$ and $\frac{\partial L}{\partial \mathbf{F}_{*j}}$ using the chain rule. These derivative values are used to update the coefficient matrix $\mathbf{W}_x$, which is then fed into six layers CNN to update the parameters of $\varphi(x)$ in each layer via the BP algorithm.

Similar to the optimization of $\mathbf{W}_x$, we optimize $\mathbf{W}_y$ on the data modality $\mathbf{Y}$ with $\mathbf{W}_x$ and $\mathbf{B}$ fixed. The derivative values are similarly used to update the coefficient matrix $\mathbf{W}_y$, which is then fed into the adopted two-layer network to update the parameters of $\phi(y)$ in each layer via the BP algorithm.

Optimize B with $\mathbf{W}_x$ and $\mathbf{W}_y$ fixed: When $\mathbf{W}_x$ and $\mathbf{W}_y$ are optimized and fixed, $\mathbf{F}$ and $\mathbf{G}$ are also determined, then the minimization problem in Eq. (4) is equal to a maximization as follows:

$$\max_{\mathbf{B}} tr(\lambda \mathbf{B}^T(\mathbf{F}+\mathbf{G})) = tr(\mathbf{B}^T \mathbf{U}) = \sum_{i,j} \mathbf{B}_{ij}\mathbf{U}_{ij} \quad (11)$$

where $\mathbf{B} \in \{-1, +1\}^{n \times c}$, $\mathbf{U} = \lambda(\mathbf{F} + \mathbf{G})$. It is easy to observe that the binary code $\mathbf{B}_{ij}$ should keep the same sign as $\mathbf{U}_{ij}$. Therefore, we have:

$$\mathbf{B} = sign(\mathbf{U}) = sign(\lambda(\mathbf{F} + \mathbf{G})) \quad (12)$$

The whole procedure of RDCMH and entire iterative process for solving Eq. (4) are summarized in Algorithm 1.

Algorithm 1 RDCMH: Ranking based Deep Cross-Modal Hashing

Input: Two modality data matrix $\mathbf{X}$ and $\mathbf{Y}$, and the corresponding label matrix $\mathbf{Z}$
Output: Hashing coefficient matrices $\mathbf{W}_x$ and $\mathbf{W}_y$, the binary code matrix $\mathbf{B}$.
1: Initialize neural network parameters of $\varphi(x)$ and $\phi(y)$, mini-batch size $n_x = n_y = 128$, and the number of iterations $iter$, $t = 1$.
2: Calculate the similarity matrix $\mathbf{S}^{xx}$, $\mathbf{S}^{yy}$, $\mathbf{S}^{xy}$ and $\mathbf{S}^{yx}$.
3: **while** $t < iter$ or not converged **do**
4:     Randomly sample n_x (n_y) triplets from $\mathbf{X}$ ($\mathbf{Y}$) to construct a mini-batch.
5:     For each sampled triplet (x_q, x_i, x_j) (or (y_q, y_i, y_j)) in the mini-batch, compute $\mathbf{F}$ and $\mathbf{G}$ in Eq. (7) by forward propagation;
6:     Update coefficient matrix $\mathbf{W}_x$,$\mathbf{W}_y$ using Eqs. (8-10);
7:     Update the network parameters of $\varphi(x)$ ($\phi(y)$) based on $\mathbf{W}_x$ ($\mathbf{W}_y$) and back propagation;
8:     Update $\mathbf{B}$ according to Eq. (12);
9:     $t = t + 1$.
10: **end while**

Experiment

Datasets

We use three benchmark datasets: Nus-wide, Pascal VOC, and Mirflicker to evaluate the performance of RDCMH. Each dataset include two modalities (image and text), but RDCMH can also be applied to other data modalities. For ≥ 3 modalities, we just need to compute the ranking lists for each modality and optimize it by minimizing the inconsistency of each ranking list between any pairwise modality.

Nus-wide[1] contains 260,648 web images, and some images are associated with textual tags. It is a multi-label dataset where each point is annotated with one or several labels from 81 concept labels. The text for each point is represented as a 1000-dimensional bag-of-words vector. The hand-crafted feature for each image is a 500-dimensional bag-of-visual words (BOVW) vector.

Wiki[2] is generated from a group of 2866 Wikipedia documents. Each document is an image-text pair labeled with 10 semantic classes. The images are represented by 128-dimensional SIFT feature vectors. The text articles are represented as probability distributions over 10 topics, which are derived from a Latent Dirichlet Allocation (LDA) model.

Mirflickr[3] originally contains 25,000 instances collected from Flicker. Each instance consists of an image and its associated textual tags, and is manually annotated with one or more labels, from a total of 24 semantic labels. The text for each point is represented as a 1386-dimensional bag-of-words vector. For the hand-crafted feature based method, each image is represented by a 512-dimensional GIST feature vector.

Evaluation metric and Comparing Methods

We use the widely used Mean Average Precision (MAP) to measure the retrieval performance of all cross-view hashing methods. A larger MAP value corresponds to a better retrieval performance.

Seven state-of-the-art and related cross-modal hashing methods are used as baselines for comparison, including Cross-modal Similarity Sensitive Hashing (CMSSH) (Bronstein et al. 2010), Semantic Correlation Maximization (SCM-seq and SCM-orth) (Zhang and Li 2014), Semantics Preserving Hashing (SePH) (Lin et al. 2017), Deep Cross-modal Hashing (DCMH) (Jiang and Li 2017), Correlation Hashing Network (CHN) (Cao et al. 2016) and Collective Deep Quantization(CDQ)(Cao et al. 2017). Source codes of these baselines are kindly provided by the authors and the input parameters of these baselines are specified according to the suggestion of the papers. As to RDCMH, we set the mini-batch size for gradient descent to 128, and set dropout rate as 0.5 on the fully connected layers to avoid overfitting. The regularization parameter λ in Eq. (4) is set to 1, and the number of iterations for optimizing Eq. (4) is fixed to 500. We empirically found RDCMH generally converges in no more than 500 iterations on all these datasets. The length of the

---

semi-supervised sematic ranking list used for training is set to 5. Namely, we divide the ranking list (i.e., $\{r_j^{xy}\}_{j=1}^n$ into 5 bins and randomly pick three points from three different bins to form a triplet for training. By doing so, we can not only capture different levels of semantic similarity, but also avoid optimizing too much triplets, whose maximum number is cubic to the number of samples. Our preliminary study shows that DRCMH holds relatively stable performance when the number of bins ≥ 4.

Results and Analysis

Search Accuracies

The MAP results for RDCMH and other baselines with hand-crafted features on MIRFLICKR, NUS-WIDE and Wiki datasets are reported in Table . Here, 'Image vs. Text' denotes the setting where the query is an image and the database is text, and 'Text vs. Image' denotes the setting where the query is a text and the database is image.

From Table , we have the following observations.

(1) RDCMH outperforms almost all the other baselines, which demonstrate the superiority of our method in cross-modal retrieval. This superiority is because RDCMH integrates the semantic ranking information into deep cross-modal hashing to preserve better semantic structure information, and jointly optimizes the triplet ranking loss and quantitative loss to obtain more compatible parameters of deep feature representations and of hashing functions. SePH achieves better results for text to image retrieval on Wiki. That is possible because its adaptability of probability-based strategy on small datasets.

(2) An unexpected observation is that the performance of CMSSH and SCM-Orth decreases as the length of hash codes increase. This may be caused by the imbalance between bits in the hash codes learnt by singular value decomposition or eigenvalue decomposition, these two decompositions are adopted these two approaches.

(3) Deep hashing methods (DCMH, CHN, CDQ and DR-CMH) have an improved performance than the others. This proves that deep feature learned from raw data is more compatible for hashing learning than hand-crafted features in cross-modal retrieval. DRCMH still outperforms DCMH, CDQ and CHN. This observation corroborates the superiority of ranking-based loss and the necessity of jointly learning deep feature presentations and hashing functions.

To further verify the effectiveness of RDCMH in semi-supervised situation, we randomly mask all the labels of 70% training samples. All the comparing methods then use the remaining labels to learn hash functions. Table reports the results under different hash bits on three datasets. All these methods manifest sharply reduced MAP values. RDCMH have higher MAP values than all the other baselines, and also outperforms SePH on the Wiki dataset. RDCMH is less affected by the insufficient labels than other methods. For example, the average MAP value of the second best performer CHN is reduced by 81.9%, and that of RDCMH is 70.2%. This is because label integrity has a significant impact on the effectiveness of supervised hashing methods. In practice, the pairwise semantic similarity between labeled data is reduced

to 9% ($3/10 \times 3/10$) in this setting. As a result, RDCMH also has a sharply reduced performance. All these comparing methods ask for sufficient label information to guide the hashing code learning. Unfortunately, these comparing methods disregard unlabeled data, which contribute to more faithfully explore the structure of data and to reliable cross-modal hashing codes. This observation proves the effectiveness of the introduced semi-supervised semantic measure in leveraging unlabeled data to boost the hashing code learning.

We conducted additional experiments on multi-label datasets with 30% missing labels by randomly masking the labels of training data. The recorded results show that RDCMH again outperforms the comparing methods. Specifically, the average MAP value of the second best performer (CDQ) is 4% less than that of RDCMH. Due to space limitation, the results are not reported here. Overall, we can conclude that RDCMH is effective in weakly-supervised scenarios.

Sensitivity to Parameters

We further explore the sensitivity of the scalar parameter λ in Eq. (4), and report the results on Mirflickr and Wiki in Fig. , where the code length fixed as 16 bits. We can see that RDCMH is slightly sensitive to λ with $\lambda \in [10^{-3}, 10^3]$, and achieves the best performance when $\lambda = 1$. Over-weighting or under-weighting the quantitative loss have a negative impact to the performance, but not so significant. In summary, an effective λ can be easily selected for RDCMH.
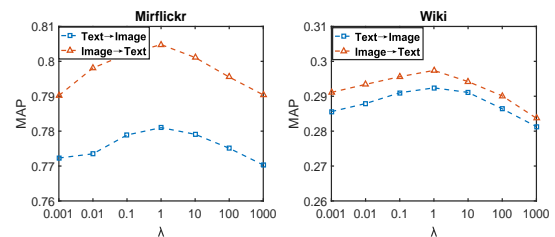

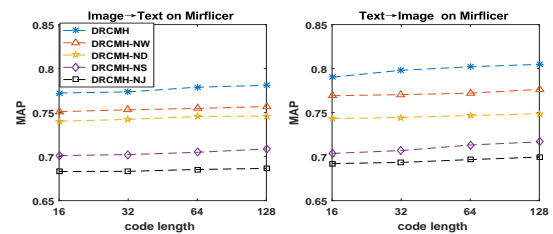Figure 2: MAP vs. λ on Mirfilcker and Wiki datasets.

Figure 3: The results of different variants on Mirflicker.

Further Analysis

To investigate the contribution components of RDCMH, we introduce four variants of RDCMH, namely RDCMH-NW, RDCMH-ND, RDCMH-NS and RDCMH-NJ. RDCMH-NW disregards the weight $(1 - s(i, j))$ and equally treats all the

Table 1: Results (MAP) on Mirflickr , Nus-wide and Wiki dataset.

	Methods	Mirflickr				Nus-wide				Wiki			
		16bits	32bits	64bits	128bits	16bits	32bits	64bits	128bits	16bits	32bits	64bits	128bits
Image vs. Text	CMSSH	0.5616	0.5555	0.5513	0.5484	0.3414	0.3336	0.3282	0.3261	0.1694	0.1523	0.1447	0.1434
	SCM-seq	0.5721	0.5607	0.5535	0.5482	0.3623	0.3646	0.3703	0.3721	0.1577	0.1434	0.1376	0.1358
	SCM-orth	0.6041	0.6112	0.6176	0.6232	0.4651	0.4714	0.4822	0.4851	0.2341	0.2411	0.2443	0.2564
	SePH	0.6573	0.6603	0.6616	0.6637	0.4787	0.4869	0.4888	0.4932	0.2836	0.2859	0.2879	0.2863
	DCMH	0.7411	0.7465	0.7485	0.7493	0.5903	0.6031	0.6093	0.6124	0.2673	0.2684	0.2687	0.2748
	CHN	0.7438	0.7485	0.7511	0.7595	0.6012	0.6028	0.6059	0.6121	0.2534	0.2677	0.2681	0.2684
	CDQ	0.7604	0.7631	0.7745	0.7738	0.6203	0.6253	0.6274	0.6284	0.2873	0.2834	0.2831	0.2901
	RDCMH	**0.7723**	**0.7735**	**0.7789**	**0.7810**	**0.6231**	**0.6236**	**0.6273**	**0.6302**	**0.2943**	**0.2968**	**0.3001**	**0.3042**
Text vs. Image	CMSSH	0.5616	0.5551	0.5506	0.5475	0.3392	0.3321	0.3272	0.3256	0.1578	0.1384	0.1331	0.1256
	SCM-seq	0.5694	0.5611	0.5544	0.5497	0.3412	0.3459	0.3472	0.3539	0.1521	0.1561	0.1371	0.1261
	SCM-orth	0.6055	0.6154	0.6238	0.6299	0.437	0.4428	0.4504	0.1235	0.2257	0.2459	0.2482	0.2518
	SePH	0.6481	0.6521	0.6545	0.6534	0.4489	0.4539	0.4587	0.4621	**0.5345**	**0.5351**	**0.5471**	**0.5506**
	DCMH	0.7827	0.7901	0.7932	0.7956	0.6389	0.6511	0.6571	0.6589	0.2712	0.2751	0.2812	0.2789
	CHN	0.7402	0.7435	0.7463	0.7481	0.6415	0.6426	0.6435	0.6478	0.2416	0.2456	0.2483	0.2512
	CDQ	0.7856	0.7841	0.7892	0.7931	0.6531	0.6579	0.6613	0.6658	0.2901	0.2847	0.3001	0.3021
	RDCMH	**0.7931**	**0.7924**	**0.8001**	**0.8024**	**0.6641**	**0.6685**	**0.6694**	**0.6703**	0.2931	0.2956	0.3012	0.3035

Table 2: Results (MAP) on Mirflickr , Nus-wide and Wiki dataset with 70% unlabeled data.

	Methods	Mirflickr				Nus-wide				Wiki			
		16bits	32bits	64bits	128bits	16bits	32bits	64bits	128bits	16bits	32bits	64bits	128bits
Image vs. Text	CMSSH	0.1384	0.1363	0.1332	0.1293	0.0731	0.0723	0.0721	0.0716	0.0146	0.0141	0.0126	0.0118
	SCM-seq	0.1419	0.1391	0.1358	0.1331	0.0741	0.0734	0.0721	0.0711	0.0162	0.0146	0.0148	0.0126
	SCM-orth	0.1321	0.1345	0.1386	0.1413	0.1056	0.1062	0.1074	0.1089	0.0174	0.0158	0.0136	0.0109
	SePH	0.1511	0.1532	0.1541	0.1538	0.1256	0.1249	0.1289	0.1291	0.0674	0.0671	0.0684	0.0681
	DCMH	0.1423	0.1435	0.1452	0.1468	0.1055	0.1056	0.1059	0.1064	0.0541	0.0514	0.0553	0.0584
	CHN	0.1344	0.1357	0.1402	0.1431	0.1125	0.1134	0.1151	0.1156	0.0584	0.0602	0.0608	0.0611
	CDQ	0.1431	0.1423	0.1462	0.1433	0.1242	0.1241	0.1195	0.1162	0.0623	0.0637	0.0645	0.0681
	RDCMH	**0.1842**	**0.1861**	**0.1875**	**0.1889**	**0.1634**	**0.1656**	**0.1674**	**0.1705**	**0.1026**	**0.1058**	**0.1073**	**0.1106**
Text vs. Image	CMSSH	0.1297	0.1343	0.1368	0.1392	0.0744	0.0751	0.0754	0.0758	0.0119	0.0119	0.0118	0.0117
	SCM-seq	0.1331	0.1366	0.1395	0.1429	0.0756	0.0759	0.0765	0.0763	0.0127	0.0122	0.0121	0.0118
	SCM-orth	0.1321	0.1376	0.1393	0.1414	0.0961	0.1008	0.1034	0.1047	0.0105	0.0108	0.0113	0.0113
	SePH	0.1434	0.1455	0.1462	0.1471	0.1194	0.1204	0.1228	0.1264	0.0651	0.0631	0.0635	0.0629
	DCMH	0.1284	0.1256	0.1284	0.1351	0.1033	0.1065	0.1069	0.1075	0.0564	0.0572	0.0585	0.0591
	CHN	0.1384	0.1342	0.1384	0.1432	0.1134	0.1142	0.1148	0.1156	0.0534	0.0542	0.0554	0.0561
	CDQ	0.1324	0.1362	0.1358	0.1402	0.1131	0.1156	0.1163	0.1189	0.0546	0.0584	0.0563	0.0573
	RDCMH	**0.1769**	**0.1786**	**0.1821**	**0.1833**	**0.1549**	**0.1569**	**0.1553**	**0.1764**	**0.1038**	**0.1045**	**0.1072**	**0.1089**

triplets; RDCMH-ND denotes the variant without deep feature learning, it directly uses the hand-crafted features to learn hashing functions during training. RDCMH-NS simply obtains the ranking list by the number of shared labels, as done by (Song et al. 2015; Zhao et al. 2015). RDCMH-NJ isolates deep feature learning and hashing functions learning, it first learns deep features and then generates hash codes based on the learnt features. Fig. shows the results of these variants on the Mirfilcker dataset. The results on other datasets provide similar observations and conclusions, and are omitted here for space limit.

We can see RDCMH outperforms RDCMH-NW. This means the triplet ranking loss with adaptive weights can improve the cross-modal retrieval quality, since it assigns larger weights to more relevant points and smaller weights to the less relevant ones. RDCMH also outperforms RDCMH-NS, which indicates that dividing the ranking lists into different levels based on the semi-supervised semantic similarity $\mathbf{S}$ is better than simply dividing by the number of shared labels, which was adopted by (Zhao et al. 2015;

Lai et al. 2015). Moreover, we can find that RDCMH achieves a higher accuracy than RDCMH-ND and RDCMH-NJ, which shows not only the superiority of deep features than hand-crafted features in cross-modal retrieval, but also the advantage of simultaneous hash-code learning and deep feature learning.

Conclusion

In this paper, we proposed a novel cross-modal hash function learning formwork (RDCMH) to seamlessly integrate deep feature learning with semantic ranking based hashing. RDCMH can preserve multi-level semantic similarity between multi-label objects for cross-modal hashing, and it also introduces a label and feature information induced semi-supervised semantic measure to leverage labeled and unlabeled data. Extensive experiments demonstrate that RDCMH outperforms other state-of-the-art hashing methods in cross-modal retrieval. The code of RDCMH is available at mlda.swu.edu.cn/codes.php?name=RDCMH.

Acknowledgments

References

Andoni, A., and Indyk, P. 2006. Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. In *FOCS*, 459–468.

Boyd, S.; Parikh, N.; Chu, E.; Peleato, B.; and Eckstein, J. 2011. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning* 3(1):1–122.

Bronstein, M. M.; Bronstein, A. M.; Michel, F.; and Paragios, N. 2010. Data fusion through cross-modality metric learning using similarity-sensitive hashing. In *CVPR*, 3594–3601.

Cao, Y.; Long, M.; Wang, J.; and Zhu, H. 2016. Correlation autoencoder hashing for supervised cross-modal search. In *ICMR*, 197–204.

Cao, Y.; Long, M.; Wang, J.; and Liu, S. 2017. Collective deep quantization for efficient cross-modal retrieval. In *AAAI*, 3974–3980.

Chang, S. F.; Jiang, Y. G.; Ji, R.; Wang, J.; and Liu, W. 2012. Supervised hashing with kernels. In *CVPR*, 2074–2081.

Chatfield, K.; Simonyan, K.; Vedaldi, A.; and Zisserman, A. 2014. Return of the devil in the details: Delving deep into convolutional nets. In *BMVC*, 1–12.

Jiang, Q. Y., and Li, W. J. 2017. Deep cross-modal hashing. In *CVPR*, 3270–3278.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *NIPS*, 1097–1105.

Kulis, B., and Grauman, K. 2010. Kernelized locality-sensitive hashing for scalable image search. In *ICCV*, 2130–2137.

Kumar, S., and Udupa, R. 2011. Learning hash functions for cross-view similarity search. In *IJCAI*, 1360–1365.

Lai, H.; Pan, Y.; Liu, Y.; and Yan, S. 2015. Simultaneous feature learning and hash coding with deep neural networks. In *CVPR*, 3270–3278.

Lin, Z.; Ding, G.; Han, J.; and Wang, J. 2017. Cross-view retrieval via probability-based semantics-preserving hashing. *IEEE Transactions on Cybernetics* 47(12):4342–4355.

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

Shao, W.; He, L.; Lu, C.-T.; Wei, X.; and Philip, S. Y. 2016. Online unsupervised multi-view feature selection. In *ICDM*, 1203–1208.

Song, J.; Yang, Y.; Yang, Y.; Huang, Z.; and Shen, H. T. 2013. Inter-media hashing for large-scale retrieval from heterogeneous data sources. In *SIGMOD*, 785–796.

Song, D.; Liu, W.; Ji, R.; Meyer, D. A.; and Smith, J. R. 2015. Top rank supervised binary coding for visual search. In *ICCV*, 1922–1930.

Wang, C.; Yan, S.; Zhang, L.; and Zhang, H.-J. 2009. Multi-label sparse coding for automatic image annotation. In *CVPR*, 1643–1650.

Wang, J.; Liu, W.; Kumar, S.; and Chang, S.-F. 2016. Learning to hash for indexing big data – a survey. *Proceedings of the IEEE* 104(1):34–57.

Wang, J.; Zhang, T.; Sebe, N.; and Shen, H. T. 2018. A survey on learning to hash. *TPAMI* 40(4):769–790.

Wang, J.; Kumar, S.; and Chang, S. F. 2012. Semi-supervised hashing for large-scale search. *TPAMI* 34(12):2393–2406.

Yi, Z., and Yeung, D. Y. 2012. Co-regularized hashing for multimodal data. In *NIPS*, 1376–1384.

Zhang, D., and Li, W. J. 2014. Large-scale supervised multimodal hashing with semantic correlation maximization. In *AAAI*, 2177–2183.

Zhang, Y., and Zhou, Z.-H. 2010. Multilabel dimensionality reduction via dependence maximization. *TKDD* 4(3):14.

Zhao, F.; Huang, Y.; Wang, L.; and Tan, T. 2015. Deep semantic ranking based hashing for multi-label image retrieval. In *CVPR*, 1556–1564.

Zhu, X.; Huang, Z.; Shen, H. T.; and Zhao, X. 2013. Linear cross-modal hashing for efficient multimedia search. In *ACM MM*, 143–152.