Cross-Media Hashing with Neural Networks

Yueting Zhuang[†], Zhou Yu[†], Wei Wang[‡], Fei Wu[†], Siliang Tang[†], Jian Shao[†]
[†]College of Computer Science, Zhejiang University, China
[‡]School of Computing, National University of Singapore, Singapore
[†]{yzhuang, yuz, wufei, jshao}@zju.edu.cn, [†]siliang.tang@gmail.com,
[‡]wangwei@comp.nus.edu.sg

ABSTRACT

Cross-media hashing, which conducts cross-media retrieval by embedding data from different modalities into a common low-dimensional hamming space, has attracted intensive attention in recent years. This is motivated by the facts a) the multi-modal data is widespread, e.g., the web images on Flickr are associated with tags, and b) hashing is an effective technique towards large-scale high-dimensional data processing, which is exactly the situation of cross-media retrieval. Inspired by recent advances in deep learning, we propose a cross-media hashing approach based on multi-modal neural networks. By restricting in the learning objective a) the hash codes for relevant cross-media data being similar, and b) the hash codes being discriminative for predicting the class labels, the learned Hamming space is expected to well capture the cross-media semantic relationships and to be semantically discriminative. The experiments on two real-world data sets show that our approach achieves superior cross-media retrieval performance compared with the state-of-the-art methods.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information Search and Retrieval

Keywords

Cross-media hashing; Neural networks

1. INTRODUCTION

With the rapid development of Internet and social networks, huge amount of multi-modal data (e.g., images and texts) is being generated at every moment. For example, one uploaded image on the Flickr web site is usually tagged with some related descriptions or labels. It is desirable to support cross-media retrieval across different modalities. In terms of the large-scale property of the multi-modal web data, hashing techniques that have been intensively investigated for the large-scale retrieval applications, become the natural choice. Consequently, cross-media hashing which incorporates hashing techniques into cross-media retrieval, is a hot research focus recently.

Many cross-media hashing approaches have been proposed in recent years [2, 3, 9, 8]. The first one was proposed by Bronstein *et al.* in CMSSH [2]. Specifically, given two modalities of data sets, CMSSH learns two groups of hash functions to ensure that if two data points (with different modalities) are relevant, their corresponding hash codes are similar and otherwise dissimilar. However, CMSSH only preserves the inter-modal correlation but ignores the intra-modal similarity. Kumar *et al.* extended Spectral Hashing [7] from the traditional uni-modal setting to the multi-modal scenario and proposed CVH [3]. CVH attempts to generate the hash codes by minimizing the distance of hash codes for the similar data and maximizing the distance for the dissimilar data, which preserves the inter-modal and intra-modal similarities at the same time. MLBE employs a generative probabilistic model to encode the intra-similarity and inter-similarity of data across multiple modalities according to the estimation of maximum a posteriori [9].

Most of the existing cross-media hashing approaches exploit the symbiosis of multi-modal data when learning hash functions. However, they do not consider the discriminative capability of the learned hash codes, which is significantly important for cross-media retrieval. Yu *et al.* propose DCDH which incorporate discriminative capability into cross-media hashing for the first time and achieve significant improvement over the existing cross-media hashing approaches [8]. However, DCDH has a limitation that it requires each data point in the training set to be discriminative to a unique class label thus can not handle the multi-class scenario, which weakens its applicability to the real-world data sets.

In this paper, we propose to learn the hash functions with the data symbiosis and discriminative capability into consideration. Specifically, we minimize two loss functions in our learning objective: a) the distance of hash codes for symbiosis data to make semantic relevant data have similar hash codes; b) the inconsistency of hash codes and class labels to make the hash codes discriminative. Motivated by the recent remarkable advances of deep learning for multi-modal data [5, 4], we exploit the NN to learn the hash functions, referred as Cross-Media Neural Network Hashing (CMNNH). Due to the flexibility of designing the NN, we can easily apply the above two loss functions on different layers of NN and optimize them through back propagation.

MM'14, November 3–7, 2014, Orlando, Florida, USA.

http://dx.doi.org/10.1145/2647868.2655059.

2. THE FRAMEWORK OF CMNNH

2.1 Notations

To simplify our presentation, we assume that the data points come from two modalities (e.g., images and texts): $X = [x_1, ..., x_{n_x}] \in \mathbb{R}^{d_x \times n_x}$, $Y = [y_1, ..., y_{n_y}] \in \mathbb{R}^{d_y \times n_y}$, where X and Y are the matrix representation of the data points, d_x and d_y denote the dimensionality of the data points from the two modalities, respectively (usually, $d_x \neq d_y$), n_x and n_y denote the number of the data points in the data sets X and Y, respectively. In our scenario, the data from different modality has pairwise relationship, i.e., each data point x_i is associated with a data point y_i, which means $n_x = n_y = n$. Therefore, we use n instead of n_x and n_y in the following sections. Besides, $T = [t_1, ..., t_n] \in \mathbb{R}^{c \times n}$ is the class label matrix for the training set, where each $t_i \in \mathbb{R}^c$ is the ground-truth class label vector for the i-th training pair, and c is the total number of labels. $t_{ij} = 1$ means the i-th training pair is labeled with the j-th label and 0 otherwise. Each class label vector t_i is normalized by $t_i = t_i / \|t_i\|_1$.

The objective of CMNNH is to learn two hash functions H^x and H^y to project the data points from different modalities into a shared Hamming space: $H^x : \mathbb{R}^{d_x} \rightarrow \{-1, 1\}^k$ and $H^y : \mathbb{R}^{d_y} \rightarrow \{-1, 1\}^k$, where k is the dimensionality of the shared space (i.e., the length of hash codes).

2.2 The Network Structure of CMNNH

The network structure of CMNNH can be seen as a combination of two modality-specific NNs as shown in Figure 1. Each NN consists of L layers: one layer for the input data, one layer for the hash codes, one layer for the output (class label) predictions, and the rest $L - 2$ layers for the hash functions, respectively. For simplification, we assume all the NNs have the same number of layers in this paper. In practice, the number of hash function layers for each NN can be different.

Denote the two NNs corresponding to X and Y as NN^x and NN^y, respectively. For each $x \in X$ ($y \in Y$ in a similar way), we forward x layer-by-layer through NN^x to generate the representation of each layer, i.e., $x^{(1)}, ..., x^{(L)}$ (for simplification, we directly use x to represent $x^{(1)}$). The l^{th} layer takes $x^{(l)}$ as the input and uses a projection function to transform it to $x^{(l+1)}$ in the next layer:

$$x^{(l+1)} = f^{(l)}(W^{(l)} x^{(l)}) \tag{1}$$

where $x^{(l)}$ and $x^{(l+1)}$ are the feature representation in the l^{th} and $l + 1^{th}$ layer, respectively; W^l is the projection matrix. $f^{(l)}(\cdot)$ is the activation function, which is usually the *sigmoid* or *tanh* function for $l = 1$ to $L - 2$, and is the *softmax* function for $l = L - 1$.

From the perspective of the hash function H^x, it takes x as input, forwards x to the hash code layer (the $L - 1^{th}$ layer), and outputs the k-dimensional binary hash codes:

$$H^x(x) = sign(x^{(L-1)}) \tag{2}$$

where $x^{(L-1)} \in \mathbb{R}^k$ is a k-dimensional real-value vector, and we use the *sign* function to convert $x^{(L-1)}$ to a binary hash code. The hash function H^y is formulated by analogy.

Nevertheless, the *sign* function is not differentiable, and thus is hard to optimize directly. We simply remove the *sign* function at the hash function learning stage and add it at the testing stage similar with most of other hashing approaches [7, 2, 3].

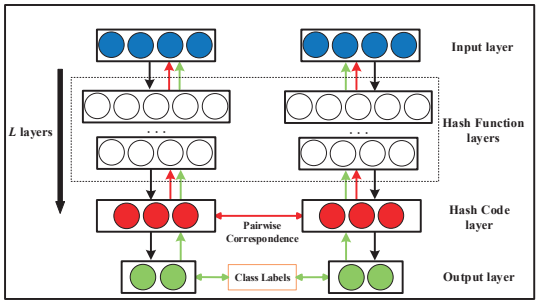


Figure 1: The illustration of hashing with multi-modal data (e.g., the pairs of images and texts) by a deep model in CMNNH, which not only disentangles the intrinsical structures of uni-modal data, but also faithfully preserves modality-specific correlation and discriminative cues (best viewed in color).

So far, the two networks NN^x and NN^y are still independent, which does not exploit any prior symbiosis of the data from different modalities. To associate NN^x with NN^y, we add two constraints:

- **Inter-modal pairwise correspondence:** the hash code layer should preserve the prior inter-modal correspondence. For each paired x_i and y_i, their hash codes $x_i^{(L-1)}$ and $y_i^{(L-1)}$ should be equal or similar.
- **Intra-modal discriminative capability:** the feature representation for x on the output layer, i.e., x^L, should be consistent with its ground-truth class labels.

The first constraint aims at preserving the inter-modal similarity which is of crucial importance in cross-media retrieval. However, merely preserving the inter-modal correlation often results in poor performance since the shared embedding space is not semantically discriminative. Therefore, we introduce additional supervised information, i.e., the class label side information, to make hash codes discriminative for predicting the class labels. The flowchart of CMNNH is given in Figure 1.

2.3 Learning Algorithm of CMNNH

The learning of CMNNH consists of two stages, namely pre-training and fine-tuning.

Pre-training is a commonly used technique for providing a good parameters initialization and preventing the learned neural network from being trapped in a bad local optimum. In CMNNH, we choose the stacked autoencoder (SAE) to pre-train each layer of the NN^x and NN^y sequentially [1].

After NN^x and NN^y are well initialized, we fine-tune the parameters by optimizing an loss function according to the two constraints. First, to preserve inter-modal pairwise correspondence, we define a loss function based on the least square error of pairwise inter-modal correspondence:

$$\ell_1(x, y) = \frac{1}{2} \|x^{(L-1)} - y^{(L-1)}\|_F^2 \tag{3}$$

Second, to preserve the intra-modal discriminative capability, we use the commonly used *softmax* regression function as the loss function on the output layer as follows.

$$\ell_2(x, y, t) = \text{KL}(x^{(L)}, t) + \text{KL}(y^{(L)}, t) \tag{4}$$

where t is the class label for x and y, KL$(\cdot)$ is the KL-divergence function. The loss of the output layer will be back propagated to its former layers, so the hash code layer is discriminative. Finally, we integrate the two loss functions for all the data points in X and Y and minimize the overall loss function as follows:

$$\mathcal{J} = \sum_{i=1}^{n} \ell_1(x_i, y_i) + \lambda \sum_{i=1}^{n} \ell_2(x_i, y_i, t_i) \quad (5)$$

where λ is a hyper-parameter to balance the two losses.

With the collaborative effect of the two loss functions, the learned hash functions H^x and H^y are expected to embed the data sets X and Y into the same space with the hash codes: a) preserving the inter-modal correlation; b) semantically discriminative.

The training of CMNNH is conducted by the classical back-propagation method. After NN^x and NN^y are trained, we obtain the hash functions H^x and H^y using Eq.(2). The overall procedures of CMNNH is given in Algorithm 1.

Algorithm 1 CMNNH

Input: data sets X,Y,T,λ
Output: The hash functions H^x, H^y

```
NN^x ← SAE(X)
NN^y ← SAE(Y)
repeat
    Pick a random pair (x_i, y_i) and their corresponding label vector t_i
    Make a gradient step for λℓ2(x_i, y_i, t_i)
    Update NN^x and NN^y, respectively
    Make a gradient step for ℓ1(x_i, y_i)
    Update NN^x and NN^y, respectively
until stopping criteria is met
```

3. EXPERIMENTS

3.1 Experimental Setup

We use two real-world data sets Wikipedia feature articles (abbreviated as Wiki) [1] and NUS-WIDE[2]. Both data sets are bi-modal containing images and texts.

The Wiki data set consists of 2,866 Wikipedia documents. Each document contains one text-image pair and is labeled by one of 10 semantic categories. For the image modality, we extract 1,000-D Bag-of-Visual-Words (BoVW) for each image. For the text modality, we calculate the frequency of all words and select the most representative words to quantize all texts into 5,000-D Bag-of-Words (BoW).

The NUS-WIDE data set contains 269,648 images and is manually annotated with 81 categories. Each image with its annotated tags can be taken as a pair of image-text data. We select those pairs that belong to one of the 10 largest categories. For the image modality, 500-D BoVW are extracted for each image. For the text modality, the corresponding tags of each image are represented by a 1,000-D BoW.

For both data sets, we setup the NNs with the the number of layers $L = 3$, i.e., the input layer is directly connected to the hash code layer. The parameters between the input layer and hash code layer are initialized by modality-specific SAE. The reason for this setting is to give a fair comparison since all the compared methods adopt shadow models. Besides, even using the 3-layers structure, we achieve good results.

[1] `http://www.svcl.ucsd.edu/projects/crossmodal/`
[2] `http://lms.comp.nus.edu.sg/research/NUS-WIDE.htm`

Table 1: The details of the data sets used in the experiments

Data Set	Wiki	NUS-Wide
Image NN structure	(1,000-k-10)	(500-k-10)
Text NN structure	(5,000-k-10)	(1,000-k-10)
Data set size	2,866	186,577
Training set size	1,000	10,000
Validation set size*	866/866	5,000/20,000
Testing set size*	866/1,000	5,000/146,577

** k in the NN structure indicates the hash code length. Partitions are ordered by query/database set respectively, and the query set are randomly sampled from the database set.*

The hyper-parameter λ is set to 10, which achieves the optimal results on the validation set and the activation function for each layer (except the output layer) is set to the sigmoid function.

We conduct two retrieval schemes in the experiments : 1) Image query vs. Text database ($\mathbf{I} \rightarrow \mathbf{T}$): use image queries to retrieve relevant texts. 2) Text query vs. Image Database ($\mathbf{T} \rightarrow \mathbf{I}$): use text queries to retrieve relevant images. For the two retrieval schemes, we compare CMNNH with the state-of-the-art cross-media hashing methods: CMSSH [2], CVH [3], MLBE [9] and LCMH [10]. However MLBE fails to learn the hash function on the NUS-Wide data set due to its high complexity on both training and testing stages.

The details of the data sets are summarized in Table 1. To evaluate the performance of the cross-media retrieval results, we adopt the Mean Average Precision (MAP) criterion.

3.2 Performance Comparison

We evaluate the cross-media retrieval performance with code length varying from 16 to 48 and report results in terms of MAP in Table 2 and 3.

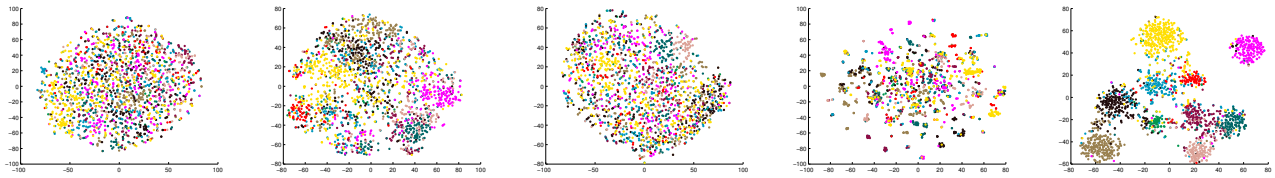
It can be noted that CMNNH significantly outperforms the counterparts over different code lengths. The is because that none of the compared methods consider the discriminative capability, while the hash codes generated by CMNNH are discriminative and well represent the semantic information. Besides, with the increasing of the code length, the performance of all the counterparts have a setback. A possible reason for this observation is that the learned hash functions is farther from the optimal solutions when the code length gets larger. In contrast, CMNNH does not suffer from the it which reveals the fact that incorporating discriminative capability facilitate the performance of cross-media retrieval.

3.3 Discussion of Discriminative Capability

To show the discriminative capability of the hash codes, we investigate the embedding Hamming space. We use the text modality on the Wiki data set since it has rich textual information and is convenient for illustrations.

First, we use the t-SNE method [6] to map the hash codes of the texts into 2-dimensional space as shown in Figure 2. From Figure 2, we can see that the hash codes of CMNNH are grouped according to their class labels, and explicit margins are observed among different classes. By contrast, the hash codes distribution of the other approaches do not have similar discriminative effect.

Second, we can also demonstrate the topic words by using the weight matrix $W^{(1)} \in \mathbb{R}^{k \times d_y}$ from the textual NN. For each row in $W^{(1)}$, the row weights indicate the contribution



(a) CVH (b) CMSSH (c) MLBE (d) LCMH (e) CMNNH

Figure 2: 2D t-SNE feature visualization of the hash codes on the textual modality of the Wiki data set. The same color indicates the hash codes have the same class label (best viewed in color).

Table 2: The MAP performance comparison on the Wiki data set.

Task	Methods	Hash code length		
		$k = 16$	$k = 32$	$k = 48$
I → T	CVH	0.1436	0.1382	0.1363
	CMSSH	0.1446	0.1384	0.1396
	MLBE	0.1393	0.1371	0.1358
	LCMH	0.1256	0.1269	0.1287
	CMNNH	**0.1917**	**0.2172**	**0.2186**
Task	**Methods**	**Hash code length**		
		$k = 16$	$k = 32$	$k = 48$
T → I	CVH	0.1380	0.1340	0.1331
	CMSSH	0.1391	0.1340	0.1353
	MLBE	0.1351	0.1367	0.1322
	LCMH	0.1242	0.1238	0.1245
	CMNNH	**0.1523**	**0.1672**	**0.1658**

Table 3: The MAP performance comparison on the NUS-Wide data set.

Task	Methods	Hash code length		
		$k = 16$	$k = 32$	$k = 48$
I → T	CVH	0.3626	0.3552	0.3522
	CMSSH	0.3608	0.3603	0.3602
	MLBE	-	-	-
	LCMH	0.3417	0.3420	0.3426
	CMNNH	**0.4071**	**0.4266**	**0.4271**
Task	**Methods**	**Hash code length**		
		$k = 16$	$k = 32$	$k = 48$
T → I	CVH	0.3621	0.3546	0.3516
	CMSSH	0.3566	0.3657	0.3426
	MLBE	-	-	-
	LCMH	0.3404	0.3407	0.3414
	CMNNH	**0.3958**	**0.4094**	**0.4171**

of all words to the corresponding "topic" in the Hamming space. The larger the weight value $W_{ij}^{(1)}$ is, the more positive correlation between i-th topic and j-th word. The top 5 words for four topics are given in Table 4, and we manually assign the most probable class labels for the four topics. From Table 4, we can find that the topic words in each line reflect a certain topic (i.e., class label).

4. CONCLUSIONS

In this paper, we propose a cross-media hashing approach based on neural networks named CMNNH, which learns the hash functions taking both the data symbiosis and discriminative capability into consideration. The hash functions are pre-trained using SAE and fine-tuned using the traditional back-propagation algorithm. The experimental results on the two data sets demonstrate the effectiveness of CMNNH.

Table 4: The representative topic words and its most probable class labels.

Class Labels	Representative Topic Words
History	Lords Rome Reign Augustus Crown
Sport	Champion Player Football Coach Stadium
Geography	Creek Park Ridge Valley Forest
Art	Theater Fiction Actress Film Poem

5. ACKNOWLEDGEMENT

This work is supported in part by National Basic Research Program of China (2012CB316400), NSFC (No.61105074), 863 program (2012AA012505), the Fundamental Research Funds for the Central Universities and Chinese Knowledge Center of Engineering Science and Technology (CKCEST) and Program for New Century Excellent Talents in University, Zhejiang Provincial Natural Science Foundation of China (No.LQ14F010004, No.LY14F020027)

6. REFERENCES

[1] Y. Bengio. Learning deep architectures for ai. *Foundations and trends in Machine Learning*, 2(1):1–127, 2009.

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

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

[4] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng. Multimodal deep learning. In *ICML*, pages 689–696, 2011.

[5] N. Srivastava and R. Salakhutdinov. Multimodal learning with deep boltzmann machines. In *NIPS*, pages 2222–2230, 2012.

[6] L. Van der Maaten and G. Hinton. Visualizing data using t-sne. *JMLR*, 9(11), 2008.

[7] Y. Weiss, A. Torralba, and R. Fergus. Spectral hashing. In *NIPS*, 2008.

[8] Z. Yu, F. Wu, Y. Yang, Q. Tian, J. Luo, and Y. Zhuang. Discriminative coupled dictionary hashing for fast cross-media retrieval. In *SIGIR*, pages 395–404, 2014.

[9] Y. Zhen and D. Yeung. A probabilistic model for multimodal hash function learning. In *SIGKDD*, 2012.

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