Cross-Media Semantic Representation via Bi-directional Learning to Rank

Fei Wu
College of Computer Science
Zhejiang University, China
wufei@cs.zju.edu.cn

Xinyan Lu
College of Computer Science
Zhejiang University, China
xinyanlu@zju.edu.cn

Zhongfei Zhang
Department of Information Science & Electronic Engineering
Zhejiang University, China
zhongfei@zju.edu.cn

Shuicheng Yan
ECE Department
National University of Singapore, Singapore
eleyans@nus.edu.sg

Yong Rui
Microsoft Research Asia, China
yongrui@microsoft.com

Yueting Zhuang
College of Computer Science
Zhejiang University, China
yzhuang@zju.edu.cn

ABSTRACT

In multimedia information retrieval, most classic approaches tend to represent different modalities of media in the same feature space. Existing approaches take either one-to-one paired data or uni-directional ranking examples (i.e., utilizing only text-query-image ranking examples or image-query-text ranking examples) as training examples, which do not make full use of bi-directional ranking examples (*bi-directional ranking* means that both text-query-image and image-query-text ranking examples are utilized in the training period) to achieve a better performance. In this paper, we consider learning a cross-media representation model from the perspective of optimizing a listwise ranking problem while taking advantage of bi-directional ranking examples. We propose a general cross-media ranking algorithm to optimize the bi-directional listwise ranking loss with a latent space embedding, which we call Bi-directional Cross-Media Semantic Representation Model (Bi-CMSRM). The latent space embedding is discriminatively learned by the structural large margin learning for optimization with certain ranking criteria (mean average precision in this paper) directly. We evaluate Bi-CMSRM on the Wikipedia and NUS-WIDE datasets and show that the utilization of the bi-directional ranking examples achieves a much better performance than only using the uni-directional ranking examples.

Categories and Subject Descriptors

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

MM '13, October 21–25, 2013, Barcelona, Spain.

http://dx.doi.org/10.1145/2502081.2502097.

Keywords

Cross-Media Representation; Bi-directional Learning to Rank; Latent Space Embedding

1. INTRODUCTION

Nowadays, many real-world applications involve multi-modal data. Cross-media retrieval is imperative to many applications of practical interest, such as finding relevant textual documents of a tourist spot that best match a given image of the spot or finding a set of images that visually best illustrate a given text description [32, 29]. However, the *heterogeneity-gap* between multi-modal data has been widely understood as a fundamental barrier to successful cross-media retrieval. To reduce this gap, one way is to map the multi-modal data into a common feature space, with then the retrieval procedure conducted in the newly mapped space. For example, automatic annotation translates the images from the image space into the text space to support the image retrieval from text queries.

In recent years, there has been a great deal of research devoted to the development of algorithms for learning an optimal common representation of different modalities. These popular approaches map the data of multiple modalities into a common (or shared) space such that the distance between two similar objects is minimized, while the distance between two dissimilar objects is maximized. One kind of approaches exploits the symbiosis of multiple-modality data which are strictly one-to-one paired. Data with multi-modal symbiosis are pervasive to describe the rich literal and visual semantics, such as a web image with loosely related narrative text descriptions, and a news report with collateral text and images. Methods like Canonical Correlation Analysis (CCA) [12] and its extensions as well as the extensions of Latent Dirichlet Allocation (LDA) [5] fall into this category.

Different from the aforementioned category of approaches which do not maximize a criterion related to the ultimate retrieval performance, another direction of approaches is based on the techniques of learning to rank. These approaches (e.g., [10, 23, 2]) are supervised but do not enforce a strict assumption that the trained multi-modal data

must be paired (e.g., one image is in pair-correspondence with its collateral text). They actually require some rankings of the data related to the queries for training, where the training examples can be easily obtained from the abundance of users' clickthrough data with little overhead [18, 9]. In this way, the learned representation for ranking multi-modal data is generally optimized based on a ranking-based loss function (evaluation criterion) to preserve the order of the relevance instead of the purely absolute similarity (dis-similarity) values between multi-modal data. However, these approaches either project the images into the text space which cannot be applied to the text document retrieval from image queries, e.g., [10, 23], or they learn a common latent space for both text and imagery while only considering uni-directional ranking examples, e.g., [2]. All the above ranking-based methods optimize only one direction of the retrieval task (image-query-text or text-query-image); thus, when bi-directional training examples are available, two different models must be individually learned, each optimizing one direction of training examples. We argue that such approaches not only add unnecessary complexity and overhead, but more importantly lead to a worse performance than directly considering and optimizing bi-directional training examples.

We focus on the retrieval of cross-media data in this paper. Moreover, we aim to learn a latent space which can be applied to both image-query-text retrieval and text-query-image retrieval, assuming bi-directional ranking examples available for training. We also consider the learned space as a latent semantic space, in which two data objects with similar semantics are close to each other. The latent space is constrained to be a low-dimensional space since the intrinsic dimensionality of a semantic space is usually much lower than that of original feature space. We note that learning a latent semantic space is particularly appropriate for retrieval with long queries/documents, where a long search query (e.g., a whole document) is beneficial as users' intents can be described in detail [30].

This paper aims to bridge the gap between learning a latent space and the retrieval of cross-media data, especially taking bi-directional ranking examples into account, which can be seen as an extension of [22]. We consider the problem of learning a latent cross-media representation from the perspective of a listwise ranking problem in this paper. We propose a general cross-media ranking algorithm to optimize the listwise ranking loss while considering a latent space embedding and bi-directional ranking examples, called Bi-directional Cross-Media Semantic Representation Model (Bi-CMSRM). Bi-CMSRM employs the structural SVM [28] to support the optimization of various ranking evaluation measures (e.g., MAP [31] and NDCG [6]) under a unified algorithmic framework. Bi-CMSRM also incorporates a latent space embedding in the learning procedure in which the latent aspect space is induced to address the curse of dimensionality and to discover the correlations between different modalities. Moreover, Bi-CMSRM takes bi-directional ranking examples into account by which two directions of retrieval tasks are optimized simultaneously, yielding a better representation for multi-modal data.

It is worthwhile to highlight the main differences between the proposed method and the existing methods. The proposed method benefits from both the latent space embedding and the most recent advances in learning to rank techniques.

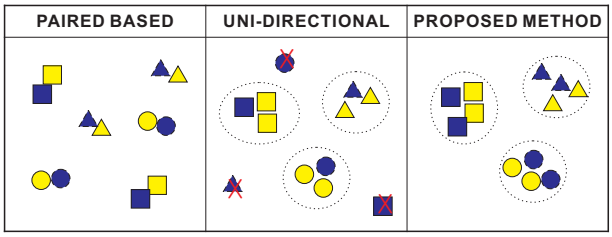


Figure 1: A simple demonstration of the latent spaces learned by different approaches. The same shape indicates relevant semantics. Colors represent modalities (i.e., text and imagery). The paired-based methods like CCA try to unite paired samples only. The uni-directional-ranking-based methods like PAMIR and SSI only capture the relationship between two modalities from one direction of retrieval but their generalization performances are limited since they do not capture the latent structure of the query modality, which is represented as blue queries with red cross in the figure. The proposed method Bi-CMSRM is trained with bi-directional training examples by which it can be applied to both directions of retrieval and the generalization performance is improved.

Moreover, the learned model considers the bi-directional ranking examples. As shown in Figure 1, the paired based methods like CCA try to unite paired samples only, which do not optimize the ultimate retrieval performance; the uni-directional-ranking based methods like PAMIR [10] and SSI [2] train asymmetric models, which cannot capture the latent structure of the query modality. To see this, assuming that we are given a corresponding image set related to a text query, the relevant images are pairwise close in semantics since they are all relevant to the same text query; thus, the latent structure of the image modality is explored. However, there is not much information about the semantics of the text query. When a new text query comes, it may not be mapped to the place near the semantics-similar text queries, and the generalization performance is limited. By this consideration, Bi-CMSRM is trained with bi-directional ranking examples such that not only both text and imagery are projected into the same semantic space such that a single learned model can be applied to both directions of retrieval, but also the performances of the two directions of retrieval are both improved.

We show experimental results on the retrieval performance of cross-media data obtained from two real-world datasets. The proposed Bi-CMSRM outperforms the existing cross-media retrieval approaches, especially in the case of the text modality with a lot of words. We also compare Bi-CMSRM with Uni-CMSRM which is trained only with uni-directional training examples, demonstrating that the utilization of bi-directional training examples does help achieve a better cross-media representation.

The rest of this paper is organized as follows. Section 2 discusses related work. In Section 3, we describe the method in detail and show its feasibility. We compare the proposed Bi-CMSRM with the existing cross-media retrieval approaches on two real-world datasets in Section 4. Conclusions are given at the end.

2. RELATED WORK

To perform cross-media retrieval, the typical method of bridging the semantic gap is through the automatic image annotation. The approaches of image annotation can be roughly classified into the generative models and the discriminative models. The generative models learn a joint distribution over image features and annotation tags. To annotate a new image, a learned generative model computes the conditional probability over tags given the visual features [4]. On the other hand, the discriminative models train a separate classifier from visual features for each tag. These classifiers are used to predict particular tags for test image samples [7].

As one of the most popular approaches to find a pair of linear transformations to maximize the correlations between two variables, Canonical Correlation Analysis (CCA) [15] and its extensions are applied in cross-media retrieval. For example, after the maximally correlated subspaces of text and image features are obtained by CCA, a logistic regression is employed to cross-media retrieval in [25]. As a supervised kernelizable extension of CCA, Generalized Multiview Analysis [27] is conducted to map data in different modality spaces to a single (non)linear subspace. Motivated by the fact that dictionary learning (DL) methods have the intrinsic power of capturing the heterogeneous features by generating different dictionaries for multi-modal data, multi-modal dictionary learning has been recently applied to cross-media retrieval [16, 24]. Following the seminal work of Blei *et al.* [5], LDA has been extended to learn the joint distribution of multi-modal data (e.g., text and imagery) such as Multimodal Document Random Field [17]. LDA-based methods assume that the paired multi-modal data should share the same latent topic proportion.

The aforementioned approaches, either optimizing the similarity (distance) between pairs of samples or optimizing the likelihood of the topic models, do not optimize the ultimate retrieval performance directly. While bearing a resemblance to multi-modal representation learning which aims at preserving the similarity or the distance measure from multimodal data, multi-modal ranking functions are generally optimized by an evaluation criterion or a loss function defined over the *permutation* space induced by the scoring function over the target documents.

Different from the classical *uni*-modal learning to rank techniques, to the best of our knowledge, Passive-Aggressive Model for Image Retrieval (PAMIR) is the first attempt to address the problem of ranking images by text queries directly [10]. PAMIR formulates the cross-media retrieval problem in a way similar to RankSVM and derives an efficient training procedure by adapting the Passive-Aggressive algorithm.

The authors of [23] studied metric learning as a problem of learning to rank. They presented a general metric learning algorithm MLR based on the structural SVM, to learn a metric such that the ranking of the data induced by the distance from a query can be optimized against various ranking measures. Different from Bi-CMSRM, both PAMIR and MLR optimize only uni-directional ranking examples and do not induce an intermediate latent space (e.g., the images are translated from the image space to the text space) such that the trained model is not applicable for the reverse direction of cross-media retrieval.

The text and imagery are usually represented as BoW and BoVW in a high-dimensional vector space respectively. However, the high-dimensional vector space representation suffers from its inability to cope with two classic problems, i.e., synonymy and polysemy. To capture the latent semantic associations of data and to address these problems, embedding words in a low-dimensional latent space to capture the semantics is a classic approach in text retrieval such as Latent Semantic Indexing (LSI) [8] and pLSA [14]. The idea of latent space embedding is also introduced into Supervised Semantic Indexing (SSI) originally proposed for cross-lingual retrieval [2]. SSI defines a set of linear low-rank models to take into account correlations between words (synonymy and polysemy). Related to SSI, Polynomial Semantic Indexing (PSI) [3] generalizes and extends the SSI approach to general polynomial models which could be used to capture the higher-order relationships among words. When SSI and PSI are applied to cross-media retrieval, they suffer from the uni-directional training examples despite inducing the latent space as mentioned before: first, the latent space is fully optimized for one direction of retrieval while the other direction is ignored; second, the generalization performance is limited. The Latent Semantic Cross-Modal Ranking algorithm [22], that optimizes for latent space embedding and direct ranking loss which we will build upon, still suffers from the above problem, i.e., only one direction of retrieval is optimized.

3. THE ALGORITHM OF BI-CMSRM

The proposed method Bi-CMSRM learns a general cross-media representation in the sense that it maps the two types of cross-media data into the same common space in which both directions of image-query-text retrieval and text-query-image retrieval can be applied. The training examples of Bi-CMSRM may cover the two directions of retrieval simultaneously, i.e., some examples are text queries with corresponding rankings of images while the other examples are image queries with corresponding rankings of text documents.

3.1 Notation

In this work, all vectors are assumed to be column vectors and a superscript T denotes the transpose of a matrix or vector. Denote m as the dimension of the text feature space (e.g., vocabulary size of bag-of-words (BoW)) and n as the dimension of the image feature space (e.g., vocabulary size of bag-of-visual-words (BoVW) quantized by clustering the low-level visual features such as SIFT [21]). We are given a training set of $N + M$ examples, with N text-query examples and M image-query examples. A query q here may be either an image p or a text document t. Similarly, the set of retrieved documents $\mathbf{d}$ can be either an image set $\mathbf{p}$ or a text document set $\mathbf{t}$. Each text-query example contains a text query $t_i \in \mathbb{R}^m$ $(i = 1, \ldots, N)$, a set of corresponding retrieved images $\mathbf{p}_i$, as well as the true rankings over the image set $\mathbf{y}_i^* \in \mathcal{Y}$, where $\mathcal{Y}$ denotes the set of all possible permutations (rankings). Similarly, each image-query example contains an image query $p_j \in \mathbb{R}^n$ $(j = N+1, \ldots, N+M)$, a set of corresponding retrieved text documents $\mathbf{t}_j$, as well as the true rankings over the text document set $\mathbf{y}_j^* \in \mathcal{Y}$. For simplicity, we omit the subscripts i and j denoting the order of a training example $(q,\mathbf{d},\mathbf{y})$ in the case where the formulation can be applied to every training example.

We denote a ranking as a matrix of pair orderings as is done in [31], $\mathcal{Y} \subset \{-1, 0, +1\}^{|\mathbf{d}|\times|\mathbf{d}|}$ where the operator $|\cdot|$ denotes the number of the elements in a set. For any $\mathbf{y} \in \mathcal{Y}$, $y_{ij} = +1$ if document d_i is ranked ahead of document d_j, and $y_{ij} = -1$ if d_j is ranked ahead of d_i, and $y_{ij} = 0$ if d_i and d_j have an equal rank. We consider only matrices corresponding to valid rankings, i.e., obeying antisymmetry and transitivity. In this paper, we assume that the true rankings are weak rankings with two rank values (*relevant* and *irrelevant*). For any query q, let $\mathbf{d}^+$ and $\mathbf{d}^-$ denote the set of relevant and irrelevant documents in $\mathbf{d}$, respectively. For example, the set of relevant text documents is denoted as $\mathbf{t}^+$ and the set of irrelevant documents as $\mathbf{t}^-$.

3.2 The Linear Mapping Functions

Motivated by the idea of latent space embedding, we would like to learn the linear mapping functions which map the text and imagery into a common latent space respectively, in which a text document and an image with similar semantics are close to each other.

Given a text $t \in \mathbb{R}^m$ and an image $p \in \mathbb{R}^n$, we consider a linear similarity function to measure the relevance between t and p:

$$f(t, p) = (Ut)^T Vp \tag{1}$$

where $U \in \mathbb{R}^{k\times m}$ and $V \in \mathbb{R}^{k\times n}$. U refers to mapping the text t from the m-dimensional text space to the k-dimensional latent space by a liner mapping, and V refers to mapping the image p from the n-dimensional image space to the k-dimensional latent space. Therefore, the text and the image are mapped to a common k-dimensional latent *aspect* space, and then their similarity is measured by a dot product of the two vectors in the k-dimensional space, which is commonly used to measure the matching between textual vectors [1].

Intuitively, the linear model in Equation (1) helps deal with the problem of textual/visual *synonymy* and *polysemy* which particularly occur in both the text space and the image space. Note that Latent Semantic Indexing (LSI) [8] takes into account of the correlations between textual words (synonym and polysemy) in a single modality in an unsupervised manner, while the linear model in Equation (1) attempts to capture the correlation across two different modalities from a supervised manner. By constraining the form of Equation (1), the benefits are similar to LSI: U and V not only induce a k-dimensional latent aspect space but are also faster to compute and lead to much smaller storage by representing the imagery and text in the k dimensions than their original dimensions (k is chosen much smaller than m or n). Similar to [2], here U and V are different and there is no assumption that the text and the imagery should be embedded to the latent space in the same way. This is appealing to cross-media representation since the distributions of the text and the imagery are inherently different due to the heterogeneity-gap.

The rest is to learn U and V. We consider learning U and V from a supervised manner, especially from both directions of training examples, i.e., the ranking of images corresponding to a given text query and the ranking of text documents corresponding to a given image query. Note that the similarity function f can also be considered a ranking function: given a text query t and a set of images $\mathbf{q}$ in the other modality, the ranking prediction $\mathbf{y}$ is derived simply by sorting the documents in $\mathbf{q}$ by descending values of $f(t, p)$: $y_{ij} = 1$ if $f(t, p_i) > f(t, p_j)$ and $y_{ij} = -1$ otherwise. Thus, we aim to obtain the values of U and V by minimizing the following empirical ranking risk,

$$R^{\Delta}(f) = \frac{1}{N}\sum_{i=1}^{N} \Delta(\mathbf{y}_i^*, \mathbf{y}_i) + \frac{1}{M}\sum_{j=N+1}^{N+M} \Delta(\mathbf{y}_j^*, \mathbf{y}_j)\,, \tag{2}$$

where the non-negative loss function $\Delta : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ quantifies the penalty for making prediction $\mathbf{y}$ if the correct output is $\mathbf{y}^*$. The value of Δ is typically bounded in $[0, 1]$. For example, we define the loss function Δ with the average precision (AP, defined in Equation (15)) loss as follows:

$$\Delta_{ap}(\mathbf{y}^*, \mathbf{y}) = 1 - \mathrm{AP}(\mathrm{rank}(\mathbf{y}^*), \mathrm{rank}(\mathbf{y}))\,,$$

and then to minimize the empirical risk is to maximize the Mean Average Precision (MAP). Consequently, we can solve the problem of *learning to rank* to learn the cross-media representation which optimizes certain ranking criteria. Furthermore, note that one can optimize different ranking criteria by considering different ranking loss functions Δ.

3.3 The Formulation of Bi-CMSRM

In this section, we present the formulation of Bi-CMSRM in details. The proposed Bi-CMSRM is based on the structural SVM framework [28]. The motivation of Bi-CMSRM is to learn a cross-media ranking function $h : \mathcal{X} \to \mathcal{Y}$ between an input space $\mathcal{X}$ (a query q as well as all target retrieved documents $\mathbf{d}$) and an output space $\mathcal{Y}$ (rankings over the retrieved document set). Similar to the structural SVM, we derive a prediction by finding the ranking $\mathbf{y}$ that maximizes the following discriminant function h:

$$h(q, \mathbf{d}) = \mathrm{argmax}_{\mathbf{y}\in\mathcal{Y}} F(q, \mathbf{d}, \mathbf{y}; U, V)\,, \tag{3}$$

where F is considered a compatibility function parameterized by U, V that measures how compatible the triple $(q, \mathbf{d}, \mathbf{y})$ is.

We first consider only one direction of the retrieval, i.e., ranking images from text queries. By adapting the most commonly used feature representation combined with *partial order* in [19] to the cross-media ranking, we define the compatible function as:

$$F(t, \mathbf{p}, \mathbf{y}) = \sum_{i\in\mathbf{p}^+}\sum_{j\in\mathbf{p}^-} y_{ij}\frac{(Ut)^T V(p_i - p_j)}{|\mathbf{p}^+|\cdot|\mathbf{p}^-|}\,, \tag{4}$$

where for any $\mathbf{y} \in \mathcal{Y}$, $y_{ij} = +1$ if image p_i is preferred (more relevant to the text query t) to image p_j, and $y_{ij} = -1$ otherwise since we assume that the predicted rankings are complete (thus for the true ranking $\mathbf{y}^*$, y_{ij} are all $+1$). Note that the summation is over all the relevant/irrelevant document pairs since we assume that the true rankings are weak rankings and we only care about the relative ranking position between a relevant document and an irrelevant document.

One attractive property of F is that for the fixed U and V, the ranking $\mathbf{y}$ which maximizes function F (then the predicted ranking) is simply sorted by descending $f(t, p) = (Ut)^T Vp$. To see this, recall that F is a summation over all the relevant/irrelevant document pairs since we assume weak rankings with two rank values. Since F decomposes linearly over the pairwise representation, we maximize F by optimizing each y_{ij} individually: if $(Ut)^T Vp_i > (Ut)^T Vp_j$, y_{ij} is set to be 1, and $y_{ij} = -1$ otherwise. This is the same procedure as sorting documents by descending $f(t, p)$.

More details can be obtained from [19]. We note that this simple prediction rule establishes a connection between the compatibility function F and the aforementioned similarity function f.

Since U and V are independent of the summation in Equation (4), we rewrite F as a linear function of U^TV:

$$F(t, \mathbf{p}, \mathbf{y}) =< U^TV, \Psi(t, \mathbf{p}, \mathbf{y}) > \quad (5)$$

where

$$\Psi(t, \mathbf{p}, \mathbf{y}) = t \sum_{i \in \mathbf{p}^+} \sum_{j \in \mathbf{p}^-} y_{ij} \frac{p_i^T - p_j^T}{|\mathbf{p}^+| \cdot |\mathbf{p}^-|} . \quad (6)$$

Here the combined feature function $\Psi(t, \mathbf{p}, \mathbf{y})$ is a summation over the vector differences of all the relevant/irrelevant image pairs. By representing the scoring F as a Frobenius inner product of U^TV and Ψ, we see that it is straightforward to extend the idea of the structural SVM to learn the cross-media ranking function F.

For the purpose of learning to rank, the structural SVM takes a set of vector-valued features which characterize the relationship between the input query and a set of target documents as the input, and predicts a ranking $\mathbf{y} \in \mathcal{Y}$ of the target documents. The structural SVM is applied to maximize the margins between the true ranking $\mathbf{y}^*$ and all the other possible rankings $\mathbf{y}$. In this paper, Bi-CMSRM takes cross-media ranking into consideration, for $i = 1, \ldots, N$:

$$\forall \mathbf{y} \in \mathcal{Y} : \delta F(t_i, \mathbf{p}_i, \mathbf{y}) \geq \Delta(\mathbf{y}_i^*, \mathbf{y}) - \xi_{1,i} \quad (7)$$

where for compactness, we define

$$\delta F(t_i, \mathbf{p}_i, \mathbf{y}) = F(t_i, \mathbf{p}_i, \mathbf{y}_i^*) - F(t_i, \mathbf{p}_i, \mathbf{y}) .$$

Similarly, consider the other direction of the retrieval, i.e., ranking text documents from image queries. To apply structural SVM, the process is analogous. Define the compatible function:

$$F(p, \mathbf{t}, \mathbf{y}) = \sum_{i \in \mathbf{t}^+} \sum_{j \in \mathbf{t}^-} y_{ij} \frac{(Vp)^T U(t_i - t_j)}{|\mathbf{t}^+| \cdot |\mathbf{t}^-|} , \quad (8)$$

rewrite F as a linear function of V^TU:

$$F(p, \mathbf{t}, \mathbf{y}) =< V^TU, \Psi(p, \mathbf{t}, \mathbf{y}) > \quad (9)$$

where

$$\Psi(p, \mathbf{t}, \mathbf{y}) = p \sum_{i \in \mathbf{t}^+} \sum_{j \in \mathbf{t}^-} y_{ij} \frac{t_i^T - t_j^T}{|\mathbf{t}^+| \cdot |\mathbf{t}^-|} \quad (10)$$

and consider maximizing the margins for $j = N+1, \ldots, N+M$:

$$\forall \mathbf{y} \in \mathcal{Y} : \delta F(p_j, \mathbf{t}_j, \mathbf{y}) \geq \Delta(\mathbf{y}_j^*, \mathbf{y}) - \xi_{2,j} \quad (11)$$

where for compactness, we define

$$\delta F(p_j, \mathbf{t}_j, \mathbf{y}) = F(p_j, \mathbf{t}_j, \mathbf{y}_j^*) - F(p_j, \mathbf{t}_j, \mathbf{y}) .$$

Since we assume that the text and the imagery are embedded into a common latent space, respectively, Bi-CMSRM adapts the original structural SVM to learn the optimal U^* and V^* which maximize the margins between the true ranking and all the other possible rankings of the target documents for each query in the other modality. Hence, we replace the standard quadratic regularization $\frac{\lambda}{2}\|w\|_2^2$ with $\frac{\lambda}{2}\|U\|_F^2 + \frac{\lambda}{2}\|V\|_F^2$ where $\|\cdot\|_F$ denotes the Frobenius norm. Intuitively, this extension simplifies the model complexity, thereby promoting a better generalization performance.

The optimization problem is then presented as follows:

Optimization Problem 1.

$$\min_{U,V,\xi_1,\xi_2} \frac{\lambda}{2}\|U\|_F^2 + \frac{\lambda}{2}\|V\|_F^2 + \frac{1}{N}\sum_{i=1}^{N} \xi_{1,i} + \frac{1}{M}\sum_{j=N+1}^{N+M} \xi_{2,j}$$

$$s.t. \quad \forall i \in \{1, \ldots, N\}, \forall \mathbf{y} \in \mathcal{Y} :$$
$$\delta F(t_i, \mathbf{p}_i, \mathbf{y}) \geq \Delta(\mathbf{y}_i^*, \mathbf{y}) - \xi_{1,i} \quad (12)$$
$$\forall j \in \{N+1, \ldots, N+M\}, \forall \mathbf{y} \in \mathcal{Y} :$$
$$\delta F(p_j, \mathbf{t}_j, \mathbf{y}) \geq \Delta(\mathbf{y}_j^*, \mathbf{y}) - \xi_{2,j} . \quad (13)$$

For each text-query example $(t, \mathbf{p}, \mathbf{y})$ and each image-query example $(p, \mathbf{t}, \mathbf{y})$ in the training set, a set of constraints (12) and (13) are added to the optimization problem. To see how these constraints indeed work, note that during the prediction, the model chooses the ranking $\bar{\mathbf{y}}$ which maximizes $F(q, \mathbf{d}, \mathbf{y})$ given the fixed U and V. If the predicted ranking is an incorrect ranking $\bar{\mathbf{y}}$, i.e., $F(q, \mathbf{d}, \bar{\mathbf{y}}) > F(q, \mathbf{d}, \mathbf{y}^*)$ where $\mathbf{y}^*$ is the true ranking, the corresponding slack variable ξ must be at least $\Delta(\mathbf{y}^*, \bar{\mathbf{y}})$ to satisfy the constraint. Considering all the triples $(t_i, \mathbf{p}_i, \mathbf{y}_i)$ for $i = 1, \ldots, N$ and $(p_j, \mathbf{t}_j, \mathbf{y}_j)$ for $j = N+1, \ldots, N+M$, the weighted sum of slacks (i.e., $\frac{1}{N}\sum_{i=1}^{N} \xi_{1,i} + \frac{1}{M}\sum_{j=N+1}^{N+M} \xi_{2,j}$) upper-bounds the empirical risk $R^{\Delta}(f)$ defined in Equation (2). This is stated formally in Proposition 1.

Proposition 1. *Denote by $\xi_1^*(U, V)$ and $\xi_2^*(U, V)$ the optimal solution of the slack variables in Optimization Problem 1 for the given parameters U and V. Then the weighted sum of slacks $\frac{1}{N}\sum_{i=1}^{N} \xi_{1,i}^* + \frac{1}{M}\sum_{j=N+1}^{N+M} \xi_{2,j}^*$ is an upper bound on the empirical risk $R^{\Delta}(f)$.*

Similar to SVM, to avoid overfitting, the objective function of Optimization Problem 1 to be minimized is a tradeoff between the model complexity and a hinge loss relaxation of Δ loss. A pre-chosen value of parameter λ controls this tradeoff and can be tuned to achieve a good performance via the validation procedure over a validation set. Similarly, a pre-chosen row number k of U and V representing the dimensionality of the latent semantic space is determined by the validation procedure.

Note that by exploring the latent semantics property, the optimization problem is not convex. The well-known *kernel trick* is difficult to be applied to Optimization Problem 1, while the kernel trick is considered one of the main benefits of the traditional support vector machine. Fortunately, a linear-SVM without using kernels has been shown to give competitive performances for textual document classification [13]. On the other hand, according to the cross-media retrieval approach PAMIR [10], a linear mapping of BoVW yields the highest performance of the other kernel mappings. As a result, with the multi-modal data under a certain feature representation, we argue that the model can indeed capture the linear structures of the multi-modal data to learn a cross-media semantic representation.

3.4 Algorithm and Implementation

Since $|\mathcal{Y}|$ is super-exponential in the size of the training set, our algorithm for learning U and V is adapted

from the 1-slack margin-rescaling cutting-plane algorithm of Joachims *et al.* [20]. The algorithm alternates between two steps, one optimizing the model parameters (U and V in our case) and the other updating the constraints set with a new batch of rankings $(\hat{\mathbf{y}}_1, \ldots, \hat{\mathbf{y}}_N, \hat{\mathbf{y}}_{N+1}, \ldots, \hat{\mathbf{y}}_{N+M})$ that are violated the most by the current model, where $\hat{\mathbf{y}}_i$ $(i = 1, \ldots, N)$ is one ranking for one text-query example, and $\hat{\mathbf{y}}_j$ $(j = N+1, \ldots, N+M)$ is one ranking for one image-query example. Once reaching a stopping criterion based on the accuracy of the empirical risk, the algorithm terminates, where the new constraint batch's empirical risk is no more than that of the current set of constraints within a tolerance $\epsilon > 0$.

Algorithm 1 Bi-directional Cross-Media Semantic Representation Model (Bi-CMSRM).

Input: text-query examples $(t_i, \mathbf{p}_i, \mathbf{y}_i^*), i = 1, \ldots, N$, image-query examples $(p_j, \mathbf{t}_j, \mathbf{y}_j^*), j = N+1, \ldots, N+M$, trade-off control parameter $\lambda > 0$, accuracy tolerance threshold $\epsilon > 0$

Output: mapping parameters U and V, slack variables $\xi_1 \geq 0$ and $\xi_2 \geq 0$

1: $\mathcal{W}_1 \leftarrow \emptyset$, $\mathcal{W}_2 \leftarrow \emptyset$

2: **repeat**

3: Solve for the optimal U, V and slack ξ_1, ξ_2:

$$\begin{aligned} \min_{U,V,\xi_1,\xi_2} \quad & \frac{\lambda}{2}\|U\|_F^2 + \frac{\lambda}{2}\|V\|_F^2 + \xi_1 + \xi_2 \\ \text{s.t.} \quad & \forall (\mathbf{y}_1, \ldots, \mathbf{y}_N) \in \mathcal{W}_1 : \\ & \frac{1}{N}\sum_{i=1}^{N} \delta F(t_i, \mathbf{p}_i, \mathbf{y}_i) \geq \frac{1}{N}\sum_{i=1}^{N} \Delta(\mathbf{y}_i^*, \mathbf{y}_i) - \xi_1 \\ & \forall (\mathbf{y}_{N+1}, \ldots, \mathbf{y}_{N+M}) \in \mathcal{W}_2 : \\ & \frac{1}{M}\sum_{j=N+1}^{N+M} \delta F(p_j, \mathbf{t}_j, \mathbf{y}_j) \geq \\ & \qquad \frac{1}{M}\sum_{j=N+1}^{N+M} \Delta(\mathbf{y}_j^*, \mathbf{y}_j) - \xi_2 \end{aligned}$$

4: **for** $i = 1$ **to** N **do**

5: $\hat{\mathbf{y}}_i \leftarrow \operatorname{argmax}_{\mathbf{y} \in \mathcal{Y}} \Delta(\mathbf{y}_i^*, \mathbf{y}) + F(t_i, \mathbf{p}_i, \mathbf{y})$

6: **end for**

7: $\mathcal{W}_1 \leftarrow \mathcal{W}_1 \cup (\hat{\mathbf{y}}_1, \ldots, \hat{\mathbf{y}}_N)$

8: **for** $j = N+1$ **to** $N+M$ **do**

9: $\hat{\mathbf{y}}_j \leftarrow \operatorname{argmax}_{\mathbf{y} \in \mathcal{Y}} \Delta(\mathbf{y}_j^*, \mathbf{y}) + F(p_j, \mathbf{t}_j, \mathbf{y})$

10: **end for**

11: $\mathcal{W}_2 \leftarrow \mathcal{W}_2 \cup (\hat{\mathbf{y}}_{N+1}, \ldots, \hat{\mathbf{y}}_{N+M})$

12: **until**

$$\frac{1}{N}\sum_{i=1}^{N} \Delta(\mathbf{y}_i^*, \hat{\mathbf{y}}_i) - \frac{1}{N}\sum_{i=1}^{N} \delta F(t_i, \mathbf{p}_i, \hat{\mathbf{y}}_i) \leq \xi_1 + \epsilon$$

and

$$\frac{1}{M}\sum_{j=N+1}^{N+M} \Delta(\mathbf{y}_j^*, \hat{\mathbf{y}}_j) - \frac{1}{M}\sum_{j=N+1}^{N+M} \delta F(p_j, \mathbf{t}_i, \hat{\mathbf{y}}_i) \leq \xi_2 + \epsilon$$

13: **return** U, V, ξ_1, ξ_2;

The general optimization procedure of Bi-CMSRM is listed in Algorithm 1. To solve the optimization problem in Algorithm 1, there are two key issues to resolve. One is searching for the most violated constraints, the so-called *separation oracle*, in Step 5 and Step 9. For different loss functions $\Delta(\mathbf{y}^*, \mathbf{y})$, different methods are proposed to address this issue, for example, [19] for AUC loss (defined as $1 - \text{AUC}(\mathbf{y}^*, \mathbf{y})$) and [31] for MAP loss. Recalling that F can be written as a Frobenius inner product, their work [19, 31] can be easily applied to this algorithm with minor modifications in the implementation to reduce the computational complexity.

The other key issue is to solve the optimization problem in Step 3. Since the problem is not a convex problem, the parameters U and V are initialized with their previous (local) optimal values while in the beginning they are randomly initialized using a normal distribution with mean zero and standard deviation one. We have implemented a subgradient descent solver adapted from Pegasos algorithm [26] originally proposed for solving a traditional support vector machine. The Pegasos algorithm is a simple iterative algorithm which alternates between stochastic subgradient descent and projection steps, and is shown to be effective to solve the primal problem of SVM. In the problem, the subgradient descent is performed iteratively where each iteration picks the most violated ranking tuple $(\hat{\mathbf{y}}_1, \hat{\mathbf{y}}_2, \ldots, \hat{\mathbf{y}}_N)$ from the set $\mathcal{W}_1$ and $(\hat{\mathbf{y}}_{N+1}, \hat{\mathbf{y}}_{N+2}, \ldots, \hat{\mathbf{y}}_{N+M})$ from the set $\mathcal{W}_2$ simultaneously to minimize the slack variables.

On iteration t, the update for U is given by:

$$\begin{aligned} U_{t+\frac{1}{2}} \leftarrow (1 - \eta_t\lambda)U_t \quad & + \quad \frac{\eta_t}{N}\sum_{i=1}^{N} V_t(\delta\Psi(t_i, \mathbf{p}_i, \hat{\mathbf{y}}_i))^T \\ & + \quad \frac{\eta_t}{M}\sum_{j=N+1}^{N+M} V_t \delta\Psi(p_j, \mathbf{t}_j, \hat{\mathbf{y}}_j) \end{aligned}$$

where $\delta\Psi(t_i, \mathbf{p}_i, \hat{\mathbf{y}}_i)$ is defined as $\Psi(t_i, \mathbf{p}_i, \mathbf{y}_i^*) - \Psi(t_i, \mathbf{p}_i, \hat{\mathbf{y}}_i)$, $\delta\Psi(p_j, \mathbf{t}_j, \hat{\mathbf{y}}_j)$ defined as $\Psi(p_j, \mathbf{t}_j, \mathbf{y}_j^*) - \Psi(p_j, \mathbf{t}_j, \hat{\mathbf{y}}_j)$, and η_t is the learning rate on iteration t which is adjustable. U_{t+1} is obtained by projecting $U_{t+\frac{1}{2}}$ onto the set for acceleration (see [26] and [11]):

$$B = \{U : \|U\|_F \leq 1/\sqrt{\lambda}\}. \tag{14}$$

The update for V can be derived similarly except for the most violated ranking tuple which is computed using the updated U_{t+1}. The update is calculated exactly as given by:

$$\begin{aligned} V_{t+\frac{1}{2}} \leftarrow (1 - \eta_t\lambda)V_t \quad & + \quad \frac{\eta_t}{N}\sum_{i=1}^{N} U_{t+1}\delta\Psi(t_i, \mathbf{p}_i, \hat{\mathbf{y}}_i) \\ & + \quad \frac{\eta_t}{M}\sum_{j=N+1}^{N+M} U_{t+1}(\delta\Psi(p_j, \mathbf{t}_j, \hat{\mathbf{y}}_j))^T \end{aligned}$$

followed by the projection step (14).

Moreover, our problem is a bit different from [26]: the objective function is penalized by $\frac{\lambda}{2}(\|U\|_F^2 + \|V\|_F^2)$ to control the model perplexity. It should be noted that the optimal U and V must satisfy the condition $\|U\|_F = \|V\|_F$ since the prediction rule uses the product $U^T V$ only. Thus after each subgradient descent, the updated U and V are forced to be multiplied by a constant respectively to ensure $\|U\|_F =$

Table 1: The statistics of the datasets used.

	Wikipedia	NUS-WIDE
BoVW vocabulary size	1,000	500
BoW vocabulary size	5,000	1,000
Avg. # of words/image	**117.5**	**7.73**
Documents	1,500/500	2,664/23,977
Partition[a]	866	106,567
Queries	1,500/500	2,664/2,000
Partition[a]	866	2,000

[a] Partitions are ordered by training/validation/test.

$\|V\|_F$ while keeping $\|U^T V\|_F$ fixed. Let $\alpha = \sqrt{\|U\|_F \|V\|_F}$,

$$U \leftarrow \alpha U / \|U\|_F \,,$$
$$V \leftarrow \alpha V / \|V\|_F \,.$$

The experiments show that this strategy yields a much faster convergence rate. For fixing tolerance $\epsilon = 0.01$, the loop in Algorithm 1 usually terminates within 200 iterations.

4. EXPERIMENTS AND RESULTS

The main goal of the experiments is to evaluate the effectiveness of the proposed Bi-CMSRM approach. To show its competitive performance, Bi-CMSRM is compared with the other three state-of-the-art approaches (CCA, PAMIR and SSI) for cross-media retrieval. These comparative methods are elaborately chosen for fair comparisons. Comparing with the classical CCA method aims to test Bi-CMSRM's ability to learn a useful latent space; PAMIR has been shown to outperform PLSA and SVM [10]; however it is designed specifically for one direction of the retrieval; SSI introduces the similar parameterizations to that of Bi-CMSRM while it takes only uni-directional training examples as the input. We further highlight the advantages of considering the bi-directional ranking examples by comparing the proposed Bi-CMSRM with Uni-CMSRM (i.e., only the uni-directional ranking examples are available with only one direction of retrieval optimized).

4.1 Experimental Setup

4.1.1 Datasets

Two public real-world datasets are used in the comparative experiments. They are the largest available multi-modal datasets that are fully paired and labeled (tagged), to the best of our knowledge. Both datasets are bi-modal with the image and the associated text modalities. The statistics of the two datasets are summarized in Table 1.

The first dataset, Wikipedia feature articles[1], consists of 2,866 images, each with a short paragraph describing the image. The images are labeled with exactly one of the 10 different semantic classes, such as art and geography. In the originally provided dataset, the text comes with a 10 dimensional feature vector representing the probabilistic proportions over the 10 topics, which is derived from a LDA model [5]. We note that Bi-CMSRM and the comparative methods all resort to the raw low-level features rather than the high-level semantic features. For the training text, we extract 5,000-dimensional feature vectors using the bag of words (BoW) representation with the TF-IDF weighting scheme. For images, we first extract SIFT points from each image in the dataset. The randomly selected SIFT points are clustered by k-means to generate 1,000 centers as the visual dictionary. Then each image is quantized into a 1,000 dimensional histogram feature vector using the bag-of-visual-words (BoVW) model.

The second dataset, NUS-WIDE[2], contains 133,208 images with 1,000 tags and 81 concepts, which are pruned from the NUS dataset by keeping the images that have at least one tag and one concept. For the feature representation, we use the publicly available 1,000 dimensional text feature vector (namely tags) and 500 dimensional image feature vector based on SIFT BoVW kindly provided by the authors.

Another reason why we choose the two datasets is due to the large difference in the average number of the textual words per image and the dimensionality of the text space. In the Wiki dataset the textual descriptions are based on Wikipedia surrounding paragraphs which yield a 5,000 dimension text space with an average of 117.5 surrounding words per image. The NUS dataset, on the other hand, is based on user-provided tags which yield a 1,000 dimension text space and in average there are 7.73 words (tags) per image. A manual examination reveals that the synonymy and polysemy problem may occur more frequently in the Wiki dataset than in the NUS dataset. For this difference, first we want to examine our algorithm's ability to learn a latent semantic representation to address the problem of synonymy and polysemy for the Wiki dataset, and second we want to see whether our algorithm decays rapidly with the NUS dataset by learning a latent semantic representation.

4.1.2 Ranking Example Generation

Note that the two datasets are both presented by pairs of text and imagery where CCA can be trained by this setting. For the other three methods (PAMIR, SSI and Bi-CMSRM), the restriction of the paired correspondence between a text document and an image is not needed. On the contrary, the queries and the corresponding rankings over the documents are needed as training examples for the three methods. These training examples originate from both directions of text-query-image and image-query-text retrieval. For this purpose, we first define the relevance assessment. For the Wiki dataset, we define that a target document d is relevant to a query q if d and q belong to the same semantic class. Similarly, for the NUS dataset, a target document d is relevant if it shares at least one concept with query q. The query lists are generated as follows: for each text (image) query, we randomly select 40 images (text documents) in the other modality in the training set as candidates and then the selected target documents are automatically labeled as relevant or irrelevant to form a ranking example. For all the 2,866 generated ranking examples in the Wiki dataset, we randomly sample 1,500 to form the training set, of which 500 form the validation set. The rest are used to form the testing set. For the NUS dataset, 2,664 ranking examples are randomly selected to be the training examples and 2,000 to be validation examples (see Table 1).

4.1.3 Performance Evaluation

Note that the proposed Bi-CMSRM is different from the other comparative methods in the way that Bi-CMSRM requires bi-directional training examples. Thus for fairness,

[1] http://www.svcl.ucsd.edu/projects/crossmodal/

[2] http://lms.comp.nus.edu.sg/research/NUS-WIDE.htm

the retrieval performance evaluations are conducted with the following settings: the comparative methods PAMIR and SSI train two individual models optimized for the image-query-text and text-query-image retrieval tasks respectively, while Bi-CMSRM trains a unified model for both directions of retrieval.

For both datasets, performance evaluations are conducted using standard information retrieval metrics. We use *Mean Average Precision* (MAP) as the performance measures. Let $r^* = \text{rank}(\mathbf{y}^*)$ (true ranking with two rank value $+1$ and -1) and $r = \text{rank}(\mathbf{y})$ (predicted ranking with a total order). Given a query and a set of R retrieved target text documents or images, the *Average Precision* (AP) is defined as

$$\text{AP}(r^*, r) = \frac{1}{L} \sum_{j=1}^{R} Prec(j) \cdot Rel(j)\,, \tag{15}$$

where L is the number of the relevant documents in the retrieved set, $Prec(j)$ is the percentage of the relevant text documents (images) in the top j text documents (images) in predicted ranking p and $Rel(j)$ is an indicator function equaling 1 if the item at rank j in predicted ranking p is relevant to the query, zero otherwise. We then average the AP values from all the queries in the query set to obtain the MAP score. The larger the MAP, the better the performance. In the experiments, R is the number of the retrieved text documents or images to be examined, where we set $R = 50$ for the top 50 retrieved text documents (images) or $R = all$ for all the documents (images). Recalling that our model can be optimized for various ranking measures, we implement the greedy algorithm for optimizing the average precision proposed in [31].

We report the performance results on both directions of ranking images from text queries (*text-query-image*) and ranking text documents from image queries (*image-query-text*). Besides, to give a pictorial demonstration of an algorithm's performance, the *Precision-Recall* curves are also reported for all the approaches.

4.1.4 Parameter Tuning

To achieve the best performance, all the comparative methods need manual parameter tuning. For the proposed method Bi-CMSRM, we tune the value of parameters k (the dimensionality of the latent space) and λ (the tradeoff between the model complexity and the empirical risk) of Optimization Problem 1 on the random sampled validation data. The respective range of k and λ are: {5, 10, 25, 50, 100, 200} and {0.01, 0.1, 1, 10}. We choose those parameters which perform best for the validation data and then the values are fixed throughout the experiments.

For the other comparative methods, the procedures to tune the parameters are analogous except that the parameters needed to be tuned are different: the dimensionality of the latent space in CCA, the aggressiveness parameter in PAMIR, and the dimensionality of the latent space & the pre-chosen fixed learning rate in SSI.

4.2 Results on the Wiki Dataset

Table 2 reports the performance of Bi-CMSRM and the other comparative models on the test set of the Wiki dataset, showing that Bi-CMSRM outperforms all the comparative methods on both directions of the retrieval.

Table 2: The performance comparison in terms of MAP@R scores on the Wiki dataset. Each text document is represented as 5000-D BoW and each image is represented as 1000-D BoVW. Both directions of retrieval tasks are reported. The results shown in boldface are the best results.

Methods	Text Query		Image Query	
	R=50	R=all	R=50	R=all
CCA	0.2343	0.1433	0.2208	0.1451
PAMIR	0.3093	0.1734	0.1797	0.1779
SSI	0.2821	0.1664	0.2344	0.1759
Uni-CMSRM	0.3663	0.2021	0.2570	0.2229
Bi-CMSRM	**0.3981**	**0.2123**	**0.2599**	**0.2528**

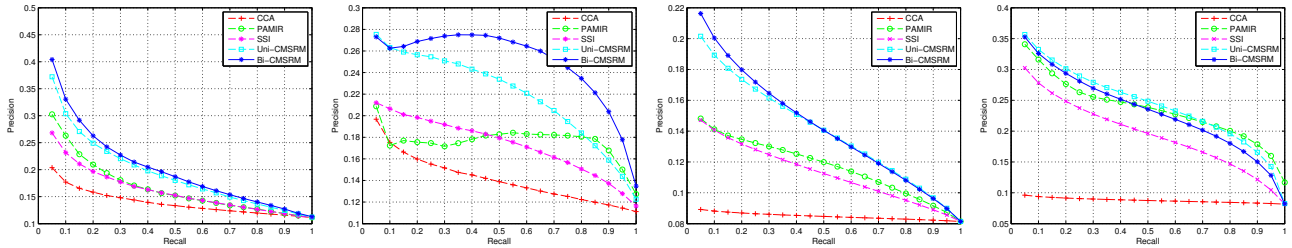
First, this improvement is due to the latent semantic space. Recall that for the Wiki dataset, the text documents contain about 117 words in average and learning a latent semantic space is particularly appropriate for the cross-media retrieval with long queries/documents. To verify this, we see that the SSI also outperforms PAMIR in the image-query-text retrieval while PAMIR even controls the model complexity by optimizing an adapted cross-media RankSVM model. Further, both Uni-CMSRM and Bi-CMSRM outperform SSI due to the structural large margin that regularizes the model and optimizes MAP ranking loss directly. Second, though Uni-CMSRM outperforms other comparative methods except for Bi-CMSRM, the bi-directional training leads to a better performance of Bi-CMSRM than that of Uni-CMSRM.

The Precision-Recall curves on both directions are reported in Figure 2(a) and 2(b). The Precision-Recall curves further validate the superiority of Bi-CMSRM for the cross-media retrieval.

4.3 Results on the NUS Dataset

The improvement of Bi-CMSRM on the NUS dataset is not as significant as that on the Wiki dataset. The MAP scores of all the methods are shown in Table 3 and the Precision-Recall curves are reported in Figure 2(c) and Figure 2(d). For text-query-image retrieval, Bi-CMSRM outperforms the other comparative methods again, while for image-query-text retrieval, both PAMIR and Uni-CMSRM have a slightly better overall performance than Bi-CMSRM when $R = 50$ and $R = all$.

Recall that in the NUS-WIDE dataset, one image is associated with about seven annotated words in average. The latent space embedding does not help much for querying short text in the task of image-query-text retrieval. PAMIR and Bi-CMSRM both train a regularized model, and therefore, the performances of PAMIR and Bi-CMSRM are undoubtedly superior to that of SSI in the image-query-text retrieval. The reason why PAMIR and Uni-CMSRM even beat Bi-CMSRM in the case of image-query-text retrieval may be as follows. As mentioned above, the assumption of latent space embedding does not help much for querying short text. With adding another direction of ranking constraints, which still assumes a latent space embedding, the assumption is violated even more, resulting in a poorer generalization performance of Bi-CMSRM than that of Uni-CMSRM. We also note that though Bi-CMSRM does not achieve the best performance, the performance differences



(a) Wiki Text-Query-Image (b) Wiki Image-Query-Text (c) NUS Text-Query-Image (d) NUS Image-Query-Text

Figure 2: Precision-Recall curves on the two datasets.

Table 3: The performance comparison in terms of MAP@R scores on the NUS dataset. Each text document is represented as 1000-D BoW and each image is represented as 500-D BoVW. Both directions of retrieval tasks are reported. The results shown in boldface are the best results.

Methods	Text Query		Image Query	
	R=50	R=all	R=50	R=all
CCA	0.1497	0.0851	0.1523	0.0883
PAMIR	0.2046	0.1184	**0.5003**	0.2410
SSI	0.2156	0.1140	0.4101	0.1992
Uni-CMSRM	0.2781	0.1424	0.4997	**0.2491**
Bi-CMSRM	**0.3224**	**0.1453**	0.4950	0.2380

among PAMIR, Uni-CMSRM and Bi-CMSRM are not significant.

5. CONCLUSIONS

In this paper, we have presented a new approach to solving the problem of learning a cross-media representation model for cross-media retrieval by casting the problem as a problem of learning to rank taking the bi-directional ranking examples into account. We have demonstrated the effectiveness of our proposed method Bi-CMSRM and shown significant improvements over the comparative methods on two datasets.

6. ACKNOWLEDGEMENTS

This work is supported in part by National Basic Research Program of China (2012CB316400), NSFC (No. 61070068, 90920303), 863 program(2012AA012505), the Fundamental Research Funds for the Central Universities and Chinese Knowledge Center of Engineering Science and Technology (CKCEST). ZZ is also supported in part by US NSF (IIS-0812114, CCF-1017828) and Zhejiang Provincial Engineering Center on Media Data Cloud Processing and Analysis.

7. REFERENCES

[1] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley, 1999.

[2] B. Bai, J. Weston, D. Grangier, R. Collobert, K. Sadamasa, Y. Qi, O. Chapelle, and K. Weinberger. Learning to rank with (a lot of) word features. *Information Retrieval*, 13(3):291–314, 2010.

[3] B. Bai, J. Weston, D. Grangier, R. Collobert, K. Sadamasa, Y. Qi, C. Cortes, and M. Mohri. Polynomial semantic indexing. *Advances in Neural Information Processing Systems*, 22:64–72, 2009.

[4] K. Barnard, P. Duygulu, D. Forsyth, N. De Freitas, D. M. Blei, and M. I. Jordan. Matching words and pictures. *The Journal of Machine Learning Research*, 3:1107–1135, 2003.

[5] D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *the Journal of Machine Learning Research*, 3:993–1022, 2003.

[6] S. Chakrabarti, R. Khanna, U. Sawant, and C. Bhattacharyya. Structured learning for non-smooth ranking losses. In *Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 88–96, 2008.

[7] Y. Chen and J. Z. Wang. Image categorization by learning and reasoning with regions. *The Journal of Machine Learning Research*, 5:913–939, 2004.

[8] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[9] J. Gao, W. Yuan, X. Li, K. Deng, and J. Nie. Smoothing clickthrough data for web search ranking. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 355–362, 2009.

[10] D. Grangier and S. Bengio. A discriminative kernel-based approach to rank images from text queries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(8):1371–1384, 2008.

[11] N. Gupta et al. *Learning to reformulate long queries*. PhD thesis, Massachusetts Institute of Technology, 2010.

[12] D. Hardoon, S. Szedmak, and J. Shawe-Taylor. Canonical correlation analysis: An overview with application to learning methods. *Neural Computation*, 16(12):2639–2664, 2004.

[13] C. Ho and C. Lin. Large-scale linear support vector regression. Technical report, National Taiwan University, 2012.

[14] T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 50–57, 1999.

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

[16] Y. Jia, M. Salzmann, and T. Darrell. Factorized latent spaces with structured sparsity. *Advances in Neural Information Processing Systems*, 23:982–990, 2010.

Figure 3: Exemplar retrieval comparison between the proposed Bi-CMSRM and Uni-CMSRM on the Wiki dataset. For text-query-image direction, the query text is shown with its corresponding image and selected words. For the image-query-text direction, the retrieved documents are shown with their corresponding images.

[17] Y. Jia, M. Salzmann, and T. Darrell. Learning cross-modality similarity for multinomial data. In *IEEE International Conference on Computer Vision*, pages 2407–2414, 2011.

[18] T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 133–142, 2002.

[19] T. Joachims. A support vector method for multivariate performance measures. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 377–384, 2005.

[20] T. Joachims, T. Finley, and C. Yu. Cutting-plane training of structural svms. *Machine Learning*, 77(1):27–59, 2009.

[21] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[22] X. Lu, F. Wu, S. Tang, Z. Zhang, X. He, and Y. Zhuang. A low rank structural large margin method for cross-modal ranking. In *Proceedings of the 36th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 433–442, 2013.

[23] B. McFee and G. Lanckriet. Metric learning to rank. In *27th International Conference on Machine Learning, Haifa, Israel*. Citeseer, 2010.

[24] G. Monaci, P. Jost, P. Vandergheynst, B. Mailhe, S. Lesage, and R. Gribonval. Learning multimodal dictionaries. *IEEE Transactions on Image Processing*, 16(9):2272–2283, 2007.

[25] N. Rasiwasia, J. Costa Pereira, E. Coviello, G. Doyle, G. Lanckriet, R. Levy, and N. Vasconcelos. A new approach to cross-modal multimedia retrieval. In *Proceedings of the International Conference on Multimedia*, pages 251–260, 2010.

[26] S. Shalev-Shwartz, Y. Singer, and N. Srebro. Pegasos: Primal estimated sub-gradient solver for svm. In *Proceedings of the 24th International Conference on Machine Learning*, pages 807–814, 2007.

[27] A. Sharma, A. Kumar, H. Daume, and D. Jacobs. Generalized multiview analysis: A discriminative latent space. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2160–2167, 2012.

[28] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, 6(2):1453–1484, 2006.

[29] F. Wu, H. Zhang, and Y. Zhuang. Learning semantic correlations for cross-media retrieval. In *2006 IEEE International Conference on Image Processing*, pages 1465–1468, 2006.

[30] Y. Yang, N. Bansal, W. Dakka, P. Ipeirotis, N. Koudas, and D. Papadias. Query by document. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, pages 34–43, 2009.

[31] Y. Yue, T. Finley, F. Radlinski, and T. Joachims. A support vector method for optimizing average precision. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 271–278, 2007.

[32] Y.-T. Zhuang, Y. Yang, and F. Wu. Mining semantic correlation of heterogeneous multimedia data for cross-media retrieval. *IEEE Transactions on Multimedia*, 10(2):221–229, 2008.