IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 2 Issue 2, February 2015.
www.ijiset.com
ISSN 2348 – 7968

Rank Learning Model of Cross-media Retrieval based on Structured SVM

Ying Xia, Tingting Xiao

Research Center of Spatial Information System, Chongqing University of Posts & Telecommunications, Chongqing, 400065, China

Abstract

The diversity of Internet information increases the users' retrieval requirements for the multimedia data across different modals. Compared with existing one-across-one retrieval approaches to get results in only one media modal, we focus on the one-across-all retrieval model to acquire related results from multiple media modals. Ranking is crucial for the quality of information retrieval. A rank learning model for cross-media retrieval is proposed, it takes advantage of listwise rank learning method, utilizes structured SVM framework construct optimization with ranking criteria to learn relevance mapping functions. Application prototype shows that this rank learning method is effective to gain relevant cross-media retrieval results.

Keywords: *Cross-media Retrieval, Rank Learning, Structured SVM.*

1. Introduction

With the development and popularization of the Internet, various information service platforms such as social networks, audio and video sharing sites are emerged. These platforms gather vast amounts of data in text, images, audios, videos and other modals. Users expect to make cross-media retrieval to get relevant objects in not only the same media modal as query but also other modals [1]. For example, if given one movie poster, user can retrieve the related brief introduction, theme songs and trailers [2]. Cross-media retrieval allows retrieving the semantic relevant multimedia data, and thus expands the method and scope of information acquisition to some extents [3]. However, heterogeneity gap [4] existing in multimedia data is one of the basic questions for cross-media retrieval. One approach to solve this problem is to map various high-dimensional features to a common low-dimensional latent space and then perform similarity retrieval, such as CCA [5], LDA [6], etc.

The results returned from the cross-media retrieval is an ordered list, which reflect the relevance between query inputs and retrieval results, the topper the results locate the greater possibility the users may browse, and the results of the sequence may affect the user experience and final evaluation directly. Therefore, rank learning method for cross-media retrieval has been widely studied. For example, PAMIR [7] introduces pairwise [8] learning uses the machine learning algorithm to study ranking functions, and calculates the correlation between text and images. LSCMR [9] optimizes the MAP [10] retrieval evaluation criterion to learn the relevance mapping function, and solves the ranking problem by employing listwise [11] model. Bi-CMSRM [12] uses bi-directional training data to learn ranking model.

Most cross-media retrieval methods based on rank learning are in one-across-one style, that is, to get results in only one media modal, this cannot satisfy users' real requirements sometimes. Assume that the database contains text and images, the one-across-one retrieval such as text-query-image retrieval is able to achieve that the query is text and results are images, as shown in Fig. 1. However, the real intention of cross-media retrieval is that query input could be any type of text or image, and the retrieval results are both of them, which could be represented as text-query-image/text or image-query-text/image retrieval, as shown in Fig. 2.

Fig.1 Text-query-image one-across-one retrieval

Fig.2 Text-query-image/text one-across-all retrieval

In order to achieve a one-across-all retrieval, this paper proposes a rank learning model for cross-media retrieval (RLMCR), which employs the query input and ordered list of it as training data to learn cross-media ranking model. For different queries and documents, use different relevance mapping functions to map them into low-dimensional latent space. Combined with the characteristics of rank learning, this model utilizes structured SVM framework to learn the relevance mapping functions. The structured SVM framework

ISSN 2348 – 7968

regards listwise ranking lists as structured output to support the optimization of evaluation criterion.

The remainder of this paper is organized as follows: Section 2 describes rank learning method and the effect for information retrieval. Section 3 introduces the processing details. Section 4 presents a cross-media retrieval application to verify the effectiveness of this method. Finally, section 5 gives some conclusions and future works.

2. Rank Learning in Information Retrieval

Rank learning [13] is a research point comes from information retrieval combined with machine learning. It aims to use machine learning algorithm to learn ranking functions, calculate the correlation of queries and documents to rank documents. The core issue of rank learning is how to construct a proper function or model to discover the relevance of queries and documents [14].

Based on training sets, rank learning could be classified into pointwise, pairwise and listwise [15]. The pointwise regards each individual training data as a learning unit, uses classification or regression method to learn ranking model, such as PankProp[16], Prank[17], etc. The pairwise employs training data with pair partial order as a learning unit, uses supervised learning to decrease the number of partial order pair, such as RankingSVM[8], RankNet[18], RankBoost[19], etc. The listwise utilizes ranking list of each query as a learning unit which maximally retain the information of ranking list, such as SVM_{MAP} [11], SoftRank [20], etc.

The original rank learning algorithm has already been explored more in text retrieval. With the advent of plenty of multimedia data, the works in multimedia retrieval such as image retrieval [21], video retrieval [22] has been carried out. PAMIR [7] tries to rank images from text queries, formulate cross-media retrieval via projecting the image into the text spaces. PAMIR uses pairwise method with Passive-Aggressive algorithm to generate efficient training procedure. But the ranking performance of PAMIR is restricted by the distributions of document pairs, and the offset data may even worsen ranking results. Inspired by PAMIR, LSCMR [9] takes advantage of structured SVM to optimize MAP evaluation criterion with low rank embedding in learning procedure to realize the semantic representation of different modals. Bi-CMSRM[12] can be seen as an extension of LSCMR, other than LSCMR's unidirectional method, Bi-CMSRM taking bi-directional ranking examples into account to train model and optimize bi-directional retrieval task to achieve better representation of multi-modal data.

3. Rank Learning Model for Cross-media Retrieval

3.1 Basic Concepts and Terminology

Query q: an image p, a text document t, or audio/video query example.

Retrieval documents $\mathbf{d}$: a documents list of query q with relevant documents $\mathbf{d}^+$ and irrelevant documents $\mathbf{d}^-$.

Ranking $\mathbf{y}$: for query q, the ranking list of retrieval documents $\mathbf{d}$. It shows relevance between query q and each document in $\mathbf{d}$.

Let us denote the query examples in training set as $S=\{s_i\}_{i=1}^{M+N}$, with M text-query examples and N image-query examples. Define m and n as the dimension of the text feature space and image feature space, respectively. For each query example triple $s_i=(q_i,\mathbf{d}_i,\mathbf{y}_i)$, q_i refers to the i-th query. Each text-query example contains a text-query $t_i \in \mathbb{R}^m (i=1,...,M)$ and each image-query example contains an image-query $p_i \in \mathbb{R}^n (i=M+1,...,M+N)$. $\mathbf{d}_i=\{d_1,d_2,...,d_l\}$ denotes all documents of the i-th query. For any query q_i , there are l relevant text and images. $\mathbf{y}_i=\{y_1,y_2,...,y_l\}$ denotes rankings determined by sorting the documents $\mathbf{d}_i$ based on their relevance with the query q_i , in which y_j refers to the ranking of j-th d_j for query q_i .

We defines $Y \subset \{-1,0,+1\}^{|\mathbf{d}|\times|\mathbf{d}|}$, for $\forall \mathbf{y} \in Y$, if document d_i is ranked ahead of d_j , $y_{ij}=+1$; and if d_j is ranked ahead of d_i , $y_{ij}=-1$; and if d_i and d_j have an equal ranking, $y_{ij}=0$.

3.2 The Relevance Mapping Functions

RLMCR studies the relevance mapping functions to map the text and image into a low-dimensional latent space. For a query q and a document d , the mapping function can be written as follows:

$$f(q,d)=q^T W d \quad . \tag{1}$$

In Eq. (1), $f(q,d)$ can also be referred to as ranking scoring function. W is the weight matrix which weights the correlation between q and d. Since the q and d may have different modals, we give the following discussion.

For text-query q_t , if q and d have the same modal, define $W=U^T U$ in Eq. (1) based on latent space embedding:

ISSN 2348 – 7968

$$f(q_t, d_t) = q_t^T (U^T U) d_t = (Uq_t)^T U d_t .$$
(2)

If query q and document d are in different modals, define $W = U^T V$:

$$f(q_t, d_p) = q_t^T (U^T V) d_p = (Uq_t)^T V d_p .$$
(3)

Here, $U \in \mathbb{R}^{k \times m}$ and $V \in \mathbb{R}^{k \times n}$. U refers to mapping the text from the m-dimensional text space to the k-dimensional latent space, and V refers to mapping the image from the n-dimensional image space to the k-dimensional latent space. After that the similarity between text and image can be measured in the k-dimensional latent space.

In Eq. (2) and (3), we employ different mapping matrixes U and V since the distributions of text and image are inherently different (i.e., heterogeneous) [23]. Latent space embedding attempts to capture the relevance across different modals thus the multi-modal data with similar semantics are close to each other [12].

Unlike [12], to achieve one-across-all retrieval, we use different mapping functions to map queries and documents to latent space for the documents across multi-modal media. If queries and documents have same modal, employing same mapping matrix in mapping functions; otherwise, using different ones.

Similar to text-query, the relevance mapping functions of image-query q_p are represented via:

$$f(q_p, d_p) = q_p^T (V^T V) d_p = (Vq_p)^T V d_p ,$$
(4)

$$f(q_p, d_t) = q_p^T (V^T U) d_t = (Vq_p)^T U d_t .$$
(5)

Obviously, for a given query q and the retrieval documents $\mathbf{d}$, we would like the more relevant documents are assigned higher scores (i.e., higher values of $f(q,d)$). The ranking prediction $\mathbf{y}$ is sorted in descending order of $f(q,d)$. Next, we will use structured SVM framework to address the relevance mapping functions which will be described in detail in the next section.

3.3 Learning Procedure

For any query q, RLMCR focus on the ranking y which determines the actual location of the target document. The RLMCR is a parameterized model of which the procedure of determining parameter is a learning procedure.

RLMCR is based on structured SVM framework [24], aims to learn a projection $f: \mathrm{X} \to \mathrm{Y}$ from input X (a query q and the retrieval documents $\mathbf{d}$) to output Y (rankings over the $\mathbf{d}$). We would like to predict a best output Y from the given input X. Therefore, we seek for a discriminant function $F: \mathrm{X} \times \mathrm{Y} \to \mathbb{R}$ to derive a prediction by finding the output Y that maximizes F. F is considered as a function to quantify the penalty or compatibility of input and output. The projection f is given by:

$$f(q, \mathbf{d}; W) = \arg\max_{\mathbf{y} \in Y} F(q, \mathbf{d}, \mathbf{y}; W) .$$ (6)

We choose F via $F(q, \mathbf{d}, \mathbf{y}) = \langle W, \Psi(q, \mathbf{d}, \mathbf{y}) \rangle$, where $\Psi(q, \mathbf{d}, \mathbf{y})$ is a combined feature function.

In order to quantify accurately, a non-negative loss function $\Delta: \mathrm{Y} \times \mathrm{Y} \to \mathbb{R}$ $(0 \le \Delta \le 1)$ should be defined for quantifying the penalty of making prediction $\mathbf{y}$ when the correct output is $\mathbf{y}^*$. We would like to obtain good performance of $f(q, \mathbf{d}; W)$ by minimizing the loss Δ. Δ satisfies $\forall \mathbf{y} \in \mathrm{Y}$, if $\mathbf{y}^* = \mathbf{y}$, $\Delta(\mathbf{y}^*, \mathbf{y}) = 0$; and if $\mathbf{y}^* \neq \mathbf{y}$, $\Delta(\mathbf{y}^*, \mathbf{y}) > 0$. For a training set S, the performance of $f(q, \mathbf{d}; W)$ is measured by the following empirical ranking risk:

$$R_S^{\Delta}(f) = \frac{1}{M} \sum_{i=1}^{M} \Delta(\mathbf{y}_i^*, \mathbf{y}_i) + \frac{1}{M} \sum_{j=M+1}^{M+N} \Delta(\mathbf{y}_j^*, \mathbf{y}_j) .$$
(7)

Unlike other loss function such as AUC loss [25] and MAP loss [11] which are evaluation measures of information retrieval, RLMCR adopts NDCG (Normalized Discounted Cumulative Gain) [26] loss to define the loss function. Since in real information retrieval, users pay more attention to the front of ranking results than the rear of them. The front ones should be assigned higher weights and this is the condition the evaluation measure NDCG takes into account. Loss function Δ is defined as: $\Delta_{NDCG}(\mathbf{y}^*, \mathbf{y}) = 1 - NDCG(rank(y^*), rank(y))$.

Ideally, we would like the empirical loss is zero which means the discriminant function is completely fitting training examples, i.e. $\forall (q_i, \mathbf{d}_i, \mathbf{y}_i)$, $f(q_i, \mathbf{d}_i) = \mathbf{y}_i$. However, in most cases we cannot achieve all feasible solutions to satisfy all constraints. By using slack variable ξ, the model does not have to completely fitting the examples of training set. Considering penalty of using slack variable and hinge loss slack of loss function Δ, as a tradeoff between them, the objective function minimizes the optimization problem as follows:

$$\min_{W, \xi} \frac{\lambda}{2} \|W\|_2^2 + \frac{1}{M+N} \sum_{i=1}^{M+N} \xi_i$$
$$s.t. \ \forall i \in \{1, \ldots, N+M\},$$
$$\forall \mathbf{y} \in \mathrm{Y} : \delta F(q_i, \mathbf{d}_i, \mathbf{y}) \ge \Delta(\mathbf{y}_i^*, \mathbf{y}) - \xi_i .$$
(8)

ISSN 2348 – 7968

Eq. (8) is the general presentation of structured SVM. A preselected parameter λ controls the tradeoff which is determined by the validation procedure. In Eq. (8), $\delta F(q_i,\mathbf{d}_i,\mathbf{y}) = F(q_i,\mathbf{d}_i,\mathbf{y}_i^*) - F(q_i,\mathbf{d}_i,\mathbf{y})$.

RLMCR adapts structured SVM to learn optimal U^* and V^* by 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 $\|.\|_F$ denotes the Frobenius inner product norm [12]. With text and images in training set, the optimization problem is defined as:

$$\min_{U,V,\xi_1,\xi_2} \frac{\lambda}{2}\|U\|_F^2 + \frac{\lambda}{2}\|V\|_F^2 + \frac{1}{M}\sum_{i=1}^{M}\xi_{1,i} + \frac{1}{M}\sum_{j=M+1}^{M+N}\xi_{2,j} \quad (9)$$

$$s.t.\ \forall i \in \{1,...,M\}, \forall \mathbf{y} \in Y:$$

$$\delta F(t_i,\mathbf{d}_i,\mathbf{y}) \geq \Delta(\mathbf{y}_i^*,\mathbf{y}) - \xi_{1,i} \quad (10)$$

$$\forall j \in \{M+1,...,M+N\}, \forall \mathbf{y} \in Y:$$

$$\delta F(p_j,\mathbf{d}_j,\mathbf{y}) \geq \Delta(\mathbf{y}_j^*,\mathbf{y}) - \xi_{2,j}. \quad (11)$$

By adapting the feature representation combined with partial order to cross-media ranking, we define F as:

$$F(q,\mathbf{d},\mathbf{y}) = \frac{1}{|\mathbf{d}^+|\cdot|\mathbf{d}^-|}\sum_{i:d_i\in\mathbf{d}^+}\sum_{j:d_j\in\mathbf{d}^-} y_{ij} q^T W(d_i - d_j). \quad (12)$$

As described in section 3.1, for $\forall \mathbf{y} \in Y$, if $d_i \succ d_j$, y_{ij}=+1, and if $d_j \succ d_i$, y_{ij}=-1. Assume that the real ranking is weak ranking [11] which means only consider the relative ranking position of $\mathbf{d}^+$ and $\mathbf{d}^-$. The combined feature function $\Psi(q,\mathbf{d},\mathbf{y})$ is defined as:

$$\Psi(q,\mathbf{d},\mathbf{y}) = \frac{q}{|\mathbf{d}^+|\cdot|\mathbf{d}^-|}\sum_{i:d_i\in\mathbf{d}^+}\sum_{j:d_j\in\mathbf{d}^-} y_{ij}(d_i^T - d_j^T). \quad (13)$$

For each query example $s_i = (q_i,\mathbf{d}_i,\mathbf{y}_i)$, model choose ranking $\bar{\mathbf{y}}$ to maximize $F(q,\mathbf{d},\mathbf{y})$ during prediction procedure. If $\bar{\mathbf{y}}$ is an incorrect ranking, i.e., $F(q,\mathbf{d},\bar{\mathbf{y}}) > F(q,\mathbf{d},\mathbf{y}^*)$, the corresponding slack variable ξ must be at least $\Delta(\mathbf{y}^*,\bar{\mathbf{y}})$ to satisfy the constraints. Taking all training examples $S = \{s_i\}_{i=1}^{M+N}$ into account, 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 upper-bound of empirical risk $R_S^{\Delta}(f)$ which is defined in Eq. (7).

3.4 Algorithm Description

To solve optimization problem of Eq. (9) in section 3.3, we uses 1-slack SVM to learn a "ranking structure". [27] has proved that 1-slack SVM is exactly equivalent to n-slack SVM. The algorithm complexity of n-slack algorithm is $O(1/\varepsilon^2)$ and the 1-slack SVM is $O(1/\varepsilon)$. The tolerance ε, tradeoff parameter λ and row number k of U and V are all preselected, which are determined by the validation procedure. Since optimization problem has complicated constraints, RLMCR uses cutting plane algorithm [27] to approximately solve this problem within the tolerance ε.

Algorithm1 RLMCR Learning Algorithm

Input: training examples $\{(t_i,\mathbf{d}_i,\mathbf{y}_i)\}_{i=1}^{M}$, $\{(p_i,\mathbf{d}_i,\mathbf{y}_i)\}_{i=M+1}^{M+N}$, control parameter $\lambda > 0$, accuracy tolerance threshold $\varepsilon > 0$

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

1: $W_1 \leftarrow \varnothing, W_2 \leftarrow \varnothing$

2: **repeat**

3: $(U,V,\xi_1,\xi_2) \leftarrow \min_{U,V,\xi_1,\xi_2} \frac{\lambda}{2}\|U\|_F^2 + \frac{\lambda}{2}\|V\|_F^2 + \xi_1 + \xi_2$

$s.t.\ \forall(\mathbf{y}_1,...,\mathbf{y}_M) \in W_1:$

$\frac{1}{M}\sum_{i=1}^{M}\delta F(t_i,\mathbf{d}_i,\mathbf{y}_i) \geq \frac{1}{M}\sum_{i=1}^{M}\Delta(\mathbf{y}_i^*,\mathbf{y}_i) - \xi_1$

$\forall(\mathbf{y}_{M+1},...,\mathbf{y}_{M+N}) \in W_2:$

$\frac{1}{N}\sum_{j=M+1}^{M+N}\delta F(p_j,\mathbf{d}_j,\mathbf{y}_j) \geq \frac{1}{N}\sum_{j=M+1}^{M+N}\Delta(\mathbf{y}_j^*,\mathbf{y}_j) - \xi_2$

4: *for* $i = 1,...,M$ *do*

5: $\hat{y}_i \leftarrow \arg\max_{y\in Y}\Delta(\mathbf{y}_i^*,\mathbf{y}) + F(t_i,\mathbf{d}_i,\mathbf{y}_i)$

6: *end for*

7: $W_1 \leftarrow W_1 \cup \{(\hat{\mathbf{y}}_1,...,\hat{\mathbf{y}}_M)\}$

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

9: $\hat{y}_j \leftarrow \arg\max_{y\in Y}\Delta(\mathbf{y}_j^*,\mathbf{y}) + F(p_j,\mathbf{d}_j,\mathbf{y}_j)$

10: *end for*

11: $W_2 \leftarrow W_2 \cup \{(\hat{\mathbf{y}}_{M+1},...,\hat{\mathbf{y}}_{M+N})\}$

12: **until**

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

and

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

13 : return U,V,ξ_1,ξ_2

The algorithm1 iteratively constructs two working set W_1 and W_2, which are composed of all possible "most violated" constraints $\{(\hat{\mathbf{y}}_1,...,\hat{\mathbf{y}}_{M+N})\}$, starting with two

ISSN 2348 – 7968

empty working sets (Line 1) and solving the parameters of current working sets (Line 3). For each text example $(t_i, \mathbf{d}_i, \mathbf{y}_i)$, find the most violated constraints $\hat{y}_i$ by the current weights (Line 4-6) and add them to the working set W_1 (Line 7). Similarly, for each image example $(p_i, \mathbf{d}_i, \mathbf{y}_i)$, find the most violated constraints $\hat{y}_j$ (Line 8-10) and add them into W_2 (Line 11). The algorithm 1 terminate until all constraints are fulfilled up to a threshold of ε (Line 12) then return the solutions (Line 13).

4. Cross-media Retrieval System Design and Implementation

This paper presents a structured SVM based rank learning model for cross-media retrieval, aims to realize one-across-all retrieval. We design and implement a prototype of cross-media retrieval system which uses text and image as training examples to verify its effectiveness.

4.1 System Design

The architecture of cross-media retrieval system is shown in Fig.3.

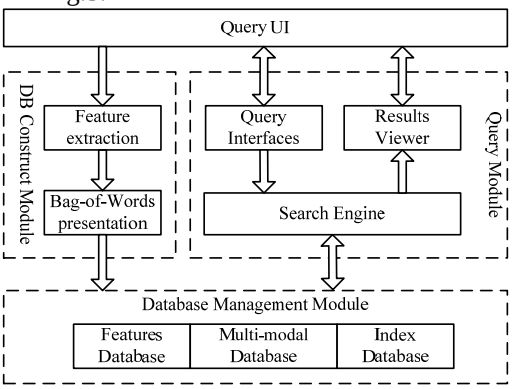

Fig.3 Architecture of cross-media retrieval system

4.1.1 Data Processing and Management

Retrieval system uses the bag-of-words with the TF-IDF weighting scheme for text feature representation and extracts visual feature to construct visual words, using bag-of-visual-words for image feature representation. The text and image feature vectors are stored in the database, constructing a feature database and generating an index database. The file directory of text and images are stored in multi-modal database. Storage path and file name of each file are both stored in feature database and index database.

4.1.2 Query Module

As the core of query module, retrieval engine employs RLMCR and the parameter learning procedure is shown in Fig.4. Users submit retrieval request in query UI, query module discriminates the type of query request, using different mapping functions to compute the correlation of multi-modal data in database and sort the results in descending order. System retrieval procedure is shown in Fig.5.

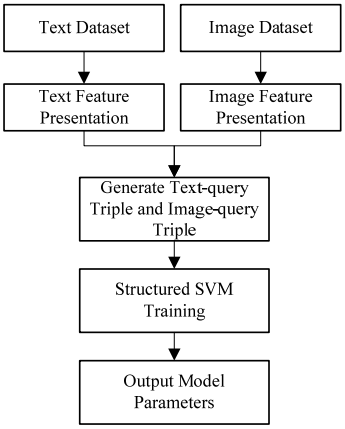

Fig.4 Parameters training procedure

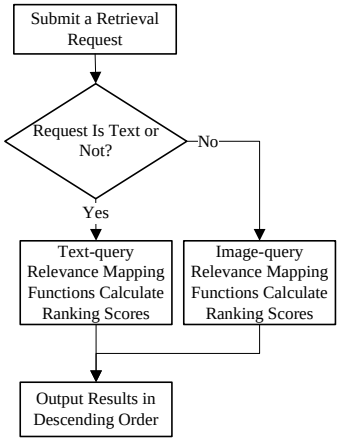

Fig.5 System retrieval procedure

4.2 System Implementation

The experiment is performed in Matlab R2013a development platform with its GUI interface, combined with represented RLMCR algorithm to realize a cross-media retrieval system. The experiment data comes from a representative public database with multi-modal: NUS-WIDE dataset [28]. For NUS-WIDE dataset, we extract 1000-dimensional text feature vectors and 500-

dimensional image feature vectors. The text and image feature vectors are formatted as Matlab MAT and are stored in features database. Users use select button of query UI to submit local text or image as retrieval request, and then retrieval engine calculate the ranking scores of data in multi-modal database. The results are ranked in descending order of scores and are shown in Retrieval Results viewer of query UI. Query UI is shown in Fig.6.

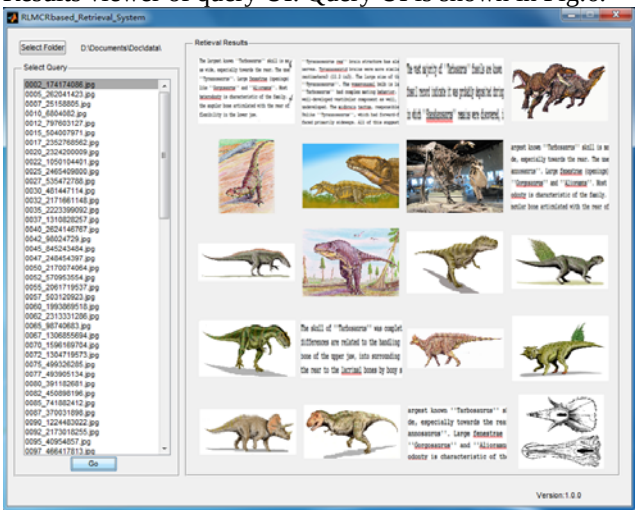

Fig.6 Query UI

As is shown in Fig.6, the query example is a local image and the returned results are semantic relevant text and images. The system realizes cross-media retrieval and the ranking results are reasonable in real situation.

5. Conclusions

This paper proposes a structured SVM based rank learning model for cross-media retrieval (RLMCR), which realizes one-across-all retrieval from the perspective of rank learning. By using the ranking list of query input in different modals as training data, this model utilizes the structured SVM to learn the relevance mapping functions to measure relevance of heterogeneous cross-media data. Based on the represented RLMCR, a cross-media retrieval prototype is designed and implemented to verify the validity of the method. The next work is to extend the model to apply to multimedia data in more modals.

Acknowledgments

This work is supported by National Natural Science Foundation of China (41201378), Natural Science Foundation Project of Chongqing CSTC (cstc2012jjA40014), and Doctoral Startup Foundation of Chongqing University of Posts and Telecommunications (A2012-34).

References

[1] Zhou Y, Hu J, Du J. Cross-media topic analysis and information retrieval[C]. //Fuzzy Systems and Knowledge Discovery (FSKD), 2012 9th International Conference on. IEEE, 2012:1211 - 1215.
[2] Wang W, Ooi B C, Yang X, et al. Effective Multi-Modal Retrieval based on Stacked Auto-Encoders[J]. Proceedings of the VLDB Endowment, 2014, 7(8).
[3] Zhang H, Wu F, Zhuang Y T. Cross-Media Retrieval Method Based On Feature Subspace Learning [J]. Pattern Recognition and Artificial Intelligence, 2009, 21(6): 739-745.
[4] Zhang H, Wu F, Zhuang Y T, et al. Cross-Media Retrieval Method Based on Content Relevances [J]. Chinese Journal of Computers, 2008, (5):820-826.
[5] Hardoon D, Szedmak S, Shawe-Taylor J. Canonical Relevance Analysis: An Overview with Application to Learning Methods[J]. Nral omaon, 2004, 16:2639 - 2664.
[6] Rasiwasia N, Costa Pereira J, Coviello E, et al. A new approach to cross-modal multimedia retrieval[C]//Proceedings of the international conference on Multimedia. ACM, 2010: 251-260.
[7] Grangier D, Bengio S. A Discriminative Kernel-Based Approach to Rank Images from Text Queries[J]. arn Analy and Mahn nllgn ranaon on, 2008, 30(8):1371 - 1384.
[8] Joachims T. Optimizing search engines using clickthrough data[C]//Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2002: 133-142.
[9] Lu X, Wu F, Tang S, et al. A low rank structural large margin method for cross-modal ranking[C]//Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval. ACM, 2013: 433-442.
[10] Baeza-Yates R, Ribeiro-Neto B. Modern information retrieval[M]. New York: ACM press, 1999.
[11] Yue Y, Finley T, Radlinski F, et al. A support vector method for optimizing average precision[C]//Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2007: 271-278.
[12] Wu F, Lu X, Zhang Z, et al. Cross-media semantic representation via Bi-directional learning to rank[C]//Proceedings of the 21st ACM international conference on Multimedia. ACM, 2013: 877-886.
[13] Liu T Y. Learning to rank for information retrieval[J]. Foundations and Trends in Information Retrieval, 2009, 3(3): 225-331.
[14] Lin Y, Lin H F. Research on Listwise Approaches to Learning to Rank Based on Neural Network [J]. Journal of the china society for scientific and technical information, 2012, 31(1): 47-59.
[15] Cheng F, Wang X. A new ranking model constructing algorithm based on direct optimization of NDCG[J]. Journal of University of Science and Technology of China, 2013, 43(1): 65-72.

ISSN 2348 – 7968

[16] Caruana R, Baluja S, Mitchell T. Using the future to "sort out" the present: Rankprop and multitask learning for medical risk evaluation[J]. Advances in neural information processing systems, 1996: 959-965.
[17] Crammer K, Singer Y. Pranking with ranking[C]//NIPS. 2001, 14: 641-647.
[18] Burges C, Shaked T, Renshaw E, et al. Learning to rank using gradient descent[C] //Proceedings of the 22nd international conference on Machine learning. ACM, 2005: 89-96.
[19] Freund Y, Iyer R, Schapire R E, et al. An efficient boosting algorithm for combining preferences[J]. The Journal of machine learning research, 2003, 4: 933-969.
[20] Taylor M, Guiver J, Robertson S, et al. Softrank: optimizing non-smooth rank metrics[C]//Proceedings of the 2008 International Conference on Web Search and Data Mining. ACM, 2008: 77-86.
[21] Hu Y, Li M, Yu N. Multiple-instance ranking: Learning to rank images for image retrieval[C]//Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE, 2008: 1-8.
[22] Yan R, Hauptmann A G. Efficient margin-based rank learning algorithms for information retrieval[M]//Image and Video Retrieval. Springer Berlin Heidelberg, 2006: 113-122.
[23] Bai B, Weston J, Grangier D, et al. Learning to rank with (a lot of) word features[J]. Information retrieval, 2010, 13(3): 291-314.
[24] Tsochantaridis I, Joachims T, Hofmann T, et al. Large margin methods for structured and interdependent output variables[C]//Journal of Machine Learning Research. 2005: 1453-1484.
[25] Joachims T. A support vector method for multivariate performance measures[C] //Proceedings of the 22nd international conference on Machine learning. ACM, 2005: 377-384.
[26] Järvelin K, Kekäläinen J. IR evaluation methods for retrieving highly relevant documents[C]//Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2000: 41-48.
[27] Joachims T, Finley T, Yu C N J. Cutting-plane training of structural SVMs[J]. Machine Learning, 2009, 77(1): 27-59.
[28] Chua T S, Tang J, Hong R, et al. NUS-WIDE: a real-world web image database from National University of Singapore[C]//Proceedings of the ACM international conference on image and video retrieval. ACM, 2009: 48.