

Bayesian Video Search Reranking

Xinmei Tian *
Univ. of Sci. & Tech. of China
Hefei, Anhui, 230027 China
xinmei@mail.ustc.edu.cn

Linjun Yang
Microsoft Research Asia
Beijing, 100190 China
linjuny@microsoft.com

Jingdong Wang
Microsoft Research Asia
Beijing, 100190 China
i-jingdw@microsoft.com

Yichen Yang *
Zhejiang University
Hangzhou, 310027 China
starswing1987@hotmail.com

Xiuqing Wu
Univ. of Sci. & Tech. of China
Hefei, Anhui, 230027 China
wuxq@ustc.edu.cn

Xian-Sheng Hua
Microsoft Research Asia
Beijing, 100190 China
xshua@microsoft.com

ABSTRACT

Content-based video search reranking can be regarded as a process that uses visual content to recover the “true” ranking list from the noisy one generated based on textual information. This paper explicitly formulates this problem in the Bayesian framework, i.e., maximizing the ranking score consistency among visually similar video shots while minimizing the ranking distance, which represents the disagreement between the objective ranking list and the initial text-based. Different from existing point-wise ranking distance measures, which compute the distance in terms of the individual scores, two new methods are proposed in this paper to measure the ranking distance based on the disagreement in terms of pair-wise orders. Specifically, *hinge distance* penalizes the pairs with reversed order according to the degree of the reverse, while *preference strength distance* further considers the preference degree. By incorporating the proposed distances into the optimization objective, two reranking methods are developed which are solved using quadratic programming and matrix computation respectively. Evaluation on TRECVID video search benchmark shows that the performance improvement up to 21% on TRECVID 2006 and 61.11% on TRECVID 2007 are achieved relative to text search baseline.

Categories and Subject Descriptors

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

General Terms

Algorithms, Experimentation, Performance

*This work was performed when Xinmei Tian and Yichen Yang were visiting Microsoft Research Asia as research interns.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MM'08, October 26–31, 2008, Vancouver, British Columbia, Canada.
Copyright 2008 ACM 978-1-60558-303-7/08/10 ...\$5.00.

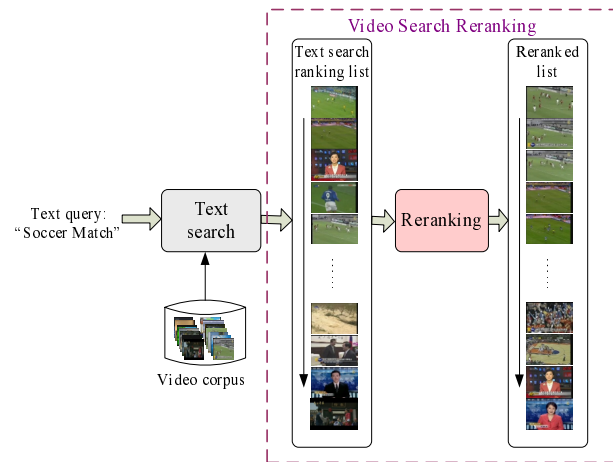


Figure 1: Example of video search reranking. Firstly the text search engine returns the video shots relevant to the query “Soccer Match” and then the reranking process is applied to reorder the text search result by mining visual information.

Keywords

Video Search Reranking, Bayesian Reranking, Pair-wise

1. INTRODUCTION

Most of the currently available video search engines are based on “query by keyword” scenario, which are built on text search engines mainly using the associated textual information such as surrounding text from the web page, speech transcript, closed caption, and so on. However, the performance of text-based video search is yet unsatisfying, due to the mismatch between surrounding text and the associated video, as well as the low performance of automatic speech recognition (ASR), video text recognition and machine translation (MT) techniques.

Figure 1 shows a typical process of video search reranking, in which a list of baseline search results is returned through textual information only and visual information is applied to reorder the initial results, so as to refine the text based search results. As illustrated in Figure 1, after a query, e.g., “Soccer Match”, is submitted, an initial ranking list of video segments (i.e., shots in general) is obtained by text search engine based on the relevance between the associated textual

information and the query keywords. It is observed that text-based search often returns “inconsistent” results, which means some visually similar ones (and semantically close to each other at the same time in most cases) are scattered in the ranking list, and frequently some irrelevant results are filled between them. For instance, as shown in Figure 1, four of the top five results of the query “Soccer Match” are the relevant samples and visually similar while the other, the anchor person, is not similar. It is reasonably assumed that the visually similar samples should be ranked together. Such a visual consistency pattern within the relevant samples can be utilized to reorder the initial ranking list, e.g., to assign the anchor person a lower ranking score. Such a process, which reorders the initial ranking list based on some visual pattern, is called content-based video search reranking, or video search reranking in brief.

Video search reranking can be regarded as recovering the “true” ranking list from the initial noisy one by using visual information, i.e., to refine the initial ranking list by incorporating the text cue and visual cue. As for text cue, we mean that the initial text-based search result provides a baseline for the “true” ranking list. Though noisy, it still reflects partial facts of the “true” list thus needs to be preserved to some extent, i.e., to keep the correct information in the initial list. The visual cue is introduced by taking visual consistency as a constraint, e.g., visually similar video shots should have close ranking scores and vice versa. Reranking is actually a trade-off between the two cues. It is worthy emphasizing that this is actually the basic underlying assumption of most of the existing video search reranking approaches, though it may not be clearly presented [7, 8, 10].

In this paper, we model the two cues from the probabilistic perspective within a Bayesian framework. The text cue is modeled as a *likelihood* which reflects the disagreement between the reranked list and the initial text-based one; and the visual cue is modeled as a *conditional prior* which indicates the ranking score consistency between visually similar samples. In the Bayesian framework, reranking is formulated as maximizing the product of the conditional prior and the likelihood. That is the reason that we call the proposed reranking approach Bayesian Reranking. Existing random walk based methods [8, 10] can be unified into such a framework.

In this paper, we mainly focus on the likelihood term while the conditional prior can be modeled by visual consistency directly. The likelihood is estimated by the ranking distance, i.e., the disagreement between the reranked list and the initial text-based one. Ranking distance is a crucial factor in video search reranking, which significantly affects the overall reranking performance but has not been well studied before. The point-wise ranking distance, which sums the individual score difference for each sample in the two ranking score lists, is used in existing video search reranking methods [8, 10]. However, such point-wise approach fails to capture the disagreement between two lists in terms of *ranking* accurately. To tackle this problem, two novel ranking distances are proposed based on the pair-wise order disagreement. Specifically, *hinge distance* penalizes the pairs with reversed order according to the degree to which they are reversed, while *preference strength distance* further considers the preference degree over pairs. By incorporating the distances into the optimization objective, *hinge reranking* and *preference strength reranking* are developed, which are

solved by Quadratic Programming (QP) and matrix computation, respectively.

The main contributions of this paper are summarized as follows:

- We explicitly formulate video search reranking as a global optimization problem within a Bayesian framework. Many effective reranking methods can be developed under this framework for different applications.
- By investigating the effects of ranking distances in video search reranking, two more reasonable ranking distance measures, *hinge distance* and *preference strength distance* are proposed.
- Two reranking methods, *hinge reranking* and *preference strength reranking*, are developed. With derivation, *hinge reranking* can be solved using quadratic programming, while *preference strength reranking* is solved through matrix computation efficiently.

The rest of this paper is organized as follows. Firstly, we briefly review the existing video search reranking methods in Section 2. In Section 3 reranking is formulated in a Bayesian framework and the general reranking model is derived. Two pair-wise ranking distances are developed and the corresponding reranking methods are presented in Section 4. Implementation details for video search reranking are considered in Section 5. The connections between our proposed methods and “learning to rank” as well as random walk reranking are presented in Section 6. Experimental results and analysis are given in Section 7, followed by the conclusion and future work in Section 8.

2. RELATED WORK

Recently many methods are proposed for video search reranking [7, 8, 10, 11, 18, 19], which can be divided into three categories: PRF (Pseudo-Relevance Feedback) based, clustering based and random walk based.

The first category is PRF based [11, 18, 19]. PRF is a concept introduced from text retrieval, which assumes that a fraction of the top-ranked documents in the initial search results are pseudo-positive [3]. In PRF based video search reranking there are normally three steps: (1) select the pseudo-positive and pseudo-negative samples from the initial text-based search results; (2) train a classifier using the selected samples; (3) rerank the video shots with the relevance scores predicted by the trained classifier. Due to the low performance of text-based video search, the top ranked video shots cannot be used as pseudo positives directly. Alternatively, [19] uses the query images or example video clips as the pseudo-positive samples. The pseudo-negative samples are selected from either the least relevant samples in the initial ranking list or the database with the assumption that few samples in the database are relevant to the query [11, 19]. In step (2), different classifiers, such as SVM [19], Boosting [18], and Ranking SVM [11], can be adopted. Although the above classifiers are effective, sufficient training data are demanded to achieve a satisfactory performance since a lot of parameters need to be estimated.

The second category is clustering based. In [7], each video shot is given a soft pseudo label according to the initial text-based ranking score, and then the Information Bottleneck principle is adopted to find optimal clustering which

maximizes the mutual information between the clusters and the labels. Reranked list is achieved by ordering the clusters according to the cluster conditional probability firstly and then ordering the samples within a cluster based on their local feature density estimated via kernel density estimation. This method achieves good performance on the named-person queries as shown in [7] while it is limited to those queries which have significant duplicate characteristic.

In the third category, random walk based methods [8, 10], a graph is constructed with the samples (video shots) as the nodes and the edges between them being weighted by multi-modal similarity. Then, reranking is formulated as random walk over the graph and the ranking scores are propagated through the edges. To leverage the text-based search result, a “dongle” node is attached to each sample with the value fixed to be the initial text-based ranking score. The stationary probability of the random walk process is used as the reranked score directly. In Section 6.2 we will show that random walk reranking can be unified into the proposed Bayesian reranking framework, while the adopted ranking distance is actually point-wise, which can not capture the “true” difference between the reranked list and the initial text-based one precisely.

There are also methods which incorporate auxiliary knowledge, including face detection [12], query example [19], and concept detection [9, 10, 13], into video search reranking. Though the incorporation of auxiliary knowledge leads to the performance improvement it is not a general treatment. They suffer from either the limited applicability to the specific queries (face detection), the desire of the specific user interfaces (query example) or the limited detection performance and small vocabulary size (concept detection). In this paper, we only consider the general reranking problem which doesn’t assume any auxiliary knowledge besides the visual information of samples and thus the reranking methods proposed in this paper can be applied to many tasks directly.

3. BAYESIAN RERANKING

3.1 Reranking

Before formulating reranking, a few terms are defined below.

DEFINITION 1. A ranking score list (score list in brief), $\mathbf{r} = [r_1, r_2, \dots, r_N]^T$, is a vector of the ranking scores, which corresponds to a sample set $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$.

DEFINITION 2. A ranking list \mathbf{l} is a permutation of \mathcal{X} sorted by the ranking scores with descending order.

Generally, reranking can be regarded as a mapping from the initial ranking list to the objective ranking list. However, the ranking scores are also useful in most situations. For this reason, we define reranking on the score list instead of the ranking list.

DEFINITION 3. A reranking function is defined as

$$\mathbf{r} = f(\mathcal{X}, \bar{\mathbf{r}}), \quad (1)$$

where $\bar{\mathbf{r}} = [\bar{r}_1, \bar{r}_2, \dots, \bar{r}_N]^T$ is the initial ranking score list.

Permutating the samples according to this reranking function is called reranking.

By defining reranking on the score list instead of the ranking list more flexibility will be achieved [7, 8]. For the application scenarios where the initial ranking scores are unavailable, such as Google image search reranking [5], the initial score list $\bar{\mathbf{r}}$ can be set according to the initial rank of samples, as detailed in 5.2.

3.2 Bayesian Reranking

The crucial problem in reranking is how to derive the optimal reranking function (1). In this paper, we will investigate the reranking problem from the probabilistic perspective and derive an optimal reranking function based on Bayesian analysis.

Supposing the ranking score list is a random variable, reranking can be regarded as a process to derive the most probable score list given the initial one as well as the visual content of samples. From the probabilistic perspective reranking is to derive the optimum \mathbf{r}^* with the maximum a posterior probability given the samples \mathcal{X} and the initial score list $\bar{\mathbf{r}}$,

$$\mathbf{r}^* = \arg \max_{\mathbf{r}} p(\mathbf{r}|\mathcal{X}, \bar{\mathbf{r}}). \quad (2)$$

According to Bayes’ formula, the posterior is proportional to the product of the conditional prior probability and the likelihood,

$$p(\mathbf{r}|\mathcal{X}, \bar{\mathbf{r}}) \propto p(\mathbf{r}|\mathcal{X}) \times p(\bar{\mathbf{r}}|\mathcal{X}, \mathbf{r}), \quad (3)$$

where $p(\mathbf{r}|\mathcal{X})$ is the conditional prior of the score list given the visual content of samples. For instance, the ranking score list with dissimilar scores for visually similar video shots may be assigned a small probability.

$p(\bar{\mathbf{r}}|\mathcal{X}, \mathbf{r})$ is the likelihood, which expresses how probable the initial score list $\bar{\mathbf{r}}$ is given the “true” ranking score list \mathbf{r} . It can be estimated based on the ranking distance which represents the disagreement between the reranked score list and the initial one, as discussed later.

In most of the video search systems, the initial ranking score list is obtained by using the textual information regardless of the visual content, therefore the conditional independency assumption of the visual information \mathcal{X} and the initial score list $\bar{\mathbf{r}}$ given the objective score list \mathbf{r} can be made, i.e.,

$$p(\bar{\mathbf{r}}, \mathcal{X}|\mathbf{r}) = p(\bar{\mathbf{r}}|\mathbf{r}) \times p(\mathcal{X}|\mathbf{r}),$$

hence,

$$p(\bar{\mathbf{r}}|\mathcal{X}, \mathbf{r}) = p(\bar{\mathbf{r}}|\mathbf{r}). \quad (4)$$

Substituting (4) into (3) we obtain

$$p(\mathbf{r}|\mathcal{X}, \bar{\mathbf{r}}) \propto p(\mathbf{r}|\mathcal{X}) \times p(\bar{\mathbf{r}}|\mathbf{r}). \quad (5)$$

Replacing the posterior in (2) with (5), we formulate reranking as maximizing the product of a conditional prior and a likelihood, which is defined as *Bayesian Reranking*.

DEFINITION 4. Bayesian Reranking is reranking using the function

$$f(\mathcal{X}, \bar{\mathbf{r}}) = \arg \max_{\mathbf{r}} p(\mathbf{r}|\mathcal{X}) \times p(\bar{\mathbf{r}}|\mathbf{r}), \quad (6)$$

where $\bar{\mathbf{r}}$ is the initial ranking score list, and \mathcal{X} is the corresponding samples.

In *Bayesian Reranking*, the likelihood and the conditional prior need to be estimated to complete the reranking function. In the following sections, we will show how to model the prior and likelihood using energy functions.

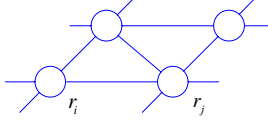


Figure 2: The graphical model representation for the conditional prior of the ranking list. A graph \mathcal{G} is built with the ranking scores as the nodes while the edges being weighted by the visual similarities.

3.3 The Conditional Prior

In video search reranking, it is expected that visually similar video shots should have close ranking scores. This can be modeled in the conditional prior in the Bayesian Reranking formulation. Specifically, we formulate the conditional prior as a pair-wise Markov network,

$$p(\mathbf{r}|\mathcal{X}) = \frac{1}{Z} \exp(-\sum_{i,j} \psi_{ij}(\mathbf{r}, \mathcal{X})),$$

where $\psi_{ij}(\mathbf{r}, \mathcal{X})$ is the energy function defined on a pair of samples $\{\mathbf{r}_i, \mathbf{x}_i, \mathbf{r}_j, \mathbf{x}_j\}$, Z is a normalizing constant with $Z = \sum_{\mathbf{r}} \exp(-\sum_{i,j} \psi_{ij}(\mathbf{r}, \mathcal{X}))$.

A graph \mathcal{G} , as illustrated in Figure 2, is constructed with nodes being the scores in \mathbf{r} while weights on edges being the similarities between the corresponding samples. Specifically, the weight w_{ij} on the edge between nodes r_i and r_j can be computed using *Gaussian* kernel, $w_{ij} = \exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2})$, where σ is the scaling parameter.

Various methods can be used to derive the energy function $\psi_{ij}(\mathbf{r}|\mathcal{X})$. Based on the assumption that if the samples \mathbf{x}_i and \mathbf{x}_j are visually similar then the corresponding scores \mathbf{r}_i and \mathbf{r}_j should be close as well and vice versa, so-called *visual consistency* assumption, the energy function is defined as

$$\psi_{ij}(\mathbf{r}, \mathcal{X}) = \frac{1}{2} w_{ij} (r_i - r_j)^2. \quad (7)$$

Hence the conditional prior is

$$p(\mathbf{r}|\mathcal{X}) = \frac{1}{Z} \exp(-\frac{1}{2} \sum_{i,j} w_{ij} (r_i - r_j)^2), \quad (8)$$

which is widely used in semi-supervised learning, and the exponent is named as Laplacian Regularization [22]. An alternative method, Normalized Laplacian Regularization [21], can also be used to derive the prior,

$$p(\mathbf{r}|\mathcal{X}) = \frac{1}{Z} \exp(-\frac{1}{2} \sum_{i,j} w_{ij} (\frac{r_i}{\sqrt{d_i}} - \frac{r_j}{\sqrt{d_j}})^2), \quad (9)$$

where $d_i = \sum_j w_{ij}$.

From our experimental analysis, Laplacian Regularization performs better than Normalized Laplacian Regularization, therefore we adopt the former in our methods.

3.4 The Likelihood

The likelihood is modeled as

$$p(\bar{\mathbf{r}}|\mathbf{r}) = \frac{1}{Z} \exp(-c \times \text{Dist}(\mathbf{r}, \bar{\mathbf{r}})), \quad (10)$$

where Z is the normalizing constant, c is a scaling parameter, and $\text{Dist}(\mathbf{r}, \bar{\mathbf{r}})$ is the ranking distance representing the disagreement between the two ranking score list, which will be discussed in detail in Section 4. The graphical model representation is illustrated in Figure 3.

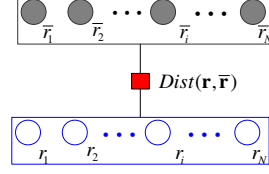


Figure 3: The factor graph representation of the ranking distance between \mathbf{r} and $\bar{\mathbf{r}}$.

Table 1: A toy example for ranking distance

Samples	\mathbf{x}_1	\mathbf{x}_2	\mathbf{x}_3	\mathbf{x}_4	\mathbf{x}_5
\mathbf{r}^0	1.0	0.9	0.8	0.7	0.6
\mathbf{r}^1	0.6	0.7	0.8	0.9	1.0
\mathbf{r}^2	1.5	0.7	0.8	0.9	1.0
\mathbf{r}^3	0.5	0.4	0.3	0.2	0.1

4. OUR APPROACHES

The Bayesian Reranking formulation in Eq.(6) is equivalent to minimizing the following energy function,

$$E(\mathbf{r}) = \frac{1}{2} \sum_{i,j} w_{ij} (r_i - r_j)^2 + c \times \text{Dist}(\mathbf{r}, \bar{\mathbf{r}}), \quad (11)$$

where the first and second terms correspond to the conditional prior in Eq.(8) and the likelihood in Eq.(10) respectively, and c can be viewed as a trade-off parameter to the two terms. The main work of this paper focuses on the second term, i.e., the evaluation of ranking distance.

4.1 Ranking Distance

In the below we will analyze the issues in existing ranking distances and propose to measure the ranking distance from the pair-wise perspective. A toy example is given for illustration, which comprises five samples $\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5\}$ and four ranking score lists $\{\mathbf{r}^0, \mathbf{r}^1, \mathbf{r}^2, \mathbf{r}^3\}$, as shown in Table 1.

Sorting the samples by their scores, the corresponding ranking lists are derived from \mathbf{r}^0 , \mathbf{r}^1 , \mathbf{r}^2 and \mathbf{r}^3 as

$$\begin{aligned} \mathbf{l}^0 &= \langle \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5 \rangle, \\ \mathbf{l}^1 &= \langle \mathbf{x}_5, \mathbf{x}_4, \mathbf{x}_3, \mathbf{x}_2, \mathbf{x}_1 \rangle, \\ \mathbf{l}^2 &= \langle \mathbf{x}_1, \mathbf{x}_5, \mathbf{x}_4, \mathbf{x}_3, \mathbf{x}_2 \rangle, \\ \mathbf{l}^3 &= \langle \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5 \rangle. \end{aligned}$$

To measure the ranking distance between the score lists, one intuitive idea is to take each score list as an “instance” and then use the list-wise approach, which has been exploited in “learning to rank”. However, as shown in [2], which defines the distance of two score lists as the cross entropy between the two distributions of permutations conditioned respectively on the each of the score lists, the list-wise approach is computationally intractable since the number of permutations is $O(N!)$ where N is the number of samples.

Alternatively, the most direct and simple method to measure the ranking distance between two score lists is to compute the individual score difference for each sample respectively and then sum them, so-called point-wise approach, as shown below,

$$\text{Dist}(\mathbf{r}, \bar{\mathbf{r}}) = \sum_i d(r_i, \bar{r}_i) = \sum_i (r_i - \bar{r}_i)^2. \quad (12)$$

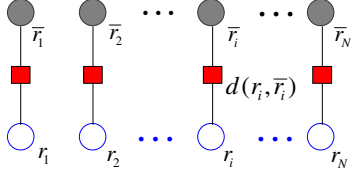


Figure 4: The factor graph representation of point-wise ranking distance in which the ranking distance is computed by summing each sample's distance.

The corresponding graphical model representation is illustrated in Figure 4. Such a point-wise approach has been applied in random walk reranking with a slightly different form, as detailed in Section 6.2.

Point-wise ranking distance, however, fails to capture the disagreement between the score lists in terms of *ranking* in some situations. Take the toy example in Table 1 for illustration. The distances between \mathbf{r}^0 and \mathbf{r}^1 , \mathbf{r}^2 , \mathbf{r}^3 computed using Eq.(12) are: $Dist(\mathbf{r}^1, \mathbf{r}^0)=0.63$, $Dist(\mathbf{r}^2, \mathbf{r}^0)=0.70$, and $Dist(\mathbf{r}^3, \mathbf{r}^0)=1.12$. $Dist(\mathbf{r}^3, \mathbf{r}^0)$ is the largest, however, in terms of *ranking*, the distance between \mathbf{r}^3 and \mathbf{r}^0 should be the smallest since \mathbf{l}^3 is identical with \mathbf{l}^0 while different from \mathbf{l}^1 and \mathbf{l}^2 .

As the ranking information can be represented entirely by the pair-wise ordinal relations, the ranking distance between two score lists can be computed from the pairs, so-called pair-wise approach. The graphical model representation of pair-wise distance is illustrated in Figure 5.

Before further discussing pair-wise approach, we firstly define the notation $\succ_{\mathbf{r}}$.

DEFINITION 5. $\mathbf{x}_i \succ_{\mathbf{r}} \mathbf{x}_j$ is a relation on a pair $(\mathbf{x}_i, \mathbf{x}_j)$ if $r_i > r_j$, i.e., \mathbf{x}_i is ranked before \mathbf{x}_j in the ranking list \mathbf{l} derived from \mathbf{r} .

All the pairs with $(\mathbf{x}_i, \mathbf{x}_j)$ satisfying $\mathbf{x}_i \succ_{\mathbf{r}} \mathbf{x}_j$ compose a set $\mathcal{S}_{\mathbf{r}} = \{(i, j) : \mathbf{x}_i \succ_{\mathbf{r}} \mathbf{x}_j\}$. For any two samples \mathbf{x}_i and \mathbf{x}_j either (i, j) or (j, i) belongs to $\mathcal{S}_{\mathbf{r}}$. Therefore, all the pair-wise ordinal relations are reflected in $\mathcal{S}_{\mathbf{r}}$.

The simplest pair-wise ranking distance could be defined as below,

$$Dist(\mathbf{r}, \bar{\mathbf{r}}) = \sum_{(i,j) \in \mathcal{S}_{\bar{\mathbf{r}}}} \delta(\mathbf{x}_j \succ_{\mathbf{r}} \mathbf{x}_i), \quad (13)$$

where

$$\delta(t) = \begin{cases} 1, & t = \text{ture} \\ 0, & t = \text{false} \end{cases}.$$

The basic idea of (13) is to count the number of pairs which disagree on the order relations in the two lists. Using (13), $Dist(\mathbf{r}^1, \mathbf{r}^0)=10$, $Dist(\mathbf{r}^2, \mathbf{r}^0)=6$, and $Dist(\mathbf{r}^3, \mathbf{r}^0)=0$. It really captures the differences of the ranking lists.

However, the optimization problem of (11) with the ranking distance (13) is computationally intractable. Below we will define two pair-wise ranking distances with which the optimization problem of (11) can be solvable.

4.2 Hinge Reranking

Intuitively, if a pair's order relation keeps the same before and after reranking, the distance of this pair will be zero, just as shown in (13). However, if a pair's order is reversed after reranking, instead of giving equal penalization (1 in (13))

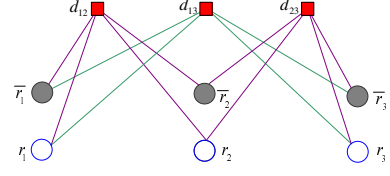


Figure 5: The factor graph representation of pair-wise ranking distance in which the ranking distance is computed by summing each pair's distance $d_{ij} = d((r_i, r_j), (\bar{r}_i, \bar{r}_j))$.

for each pair, the penalization should be given according to the degree to which the pair's order is reversed. Hence, we define *hinge distance* as

$$\begin{aligned} Dist(\mathbf{r}, \bar{\mathbf{r}}) &= \sum_{(i,j) \in \mathcal{S}_{\bar{\mathbf{r}}}} d((r_i, r_j), (\bar{r}_i, \bar{r}_j)) \\ &= \sum_{(i,j) \in \mathcal{S}_{\bar{\mathbf{r}}}} [(r_j - r_i)_+]^2, \end{aligned} \quad (14)$$

where

$$(x)_+ = \begin{cases} 0, & x \leq 0 \\ x, & x > 0 \end{cases}$$

is the hinge function.

Substitute hinge distance (14) into (11) and the following optimization problem is derived

$$\min_{\mathbf{r}} \frac{1}{2} \sum_{i,j} w_{ij} (r_i - r_j)^2 + c \sum_{(i,j) \in \mathcal{S}_{\bar{\mathbf{r}}}} [(r_j - r_i)_+]^2,$$

which is equivalent to

$$\begin{aligned} \min_{\mathbf{r}} \quad & \frac{1}{2} \sum_{i,j} w_{ij} (r_i - r_j)^2 + c \sum_{(i,j) \in \mathcal{S}_{\bar{\mathbf{r}}}} \xi_{ij}^2 \\ \text{s.t.} \quad & r_i - r_j > -\xi_{ij}, \text{ when } \mathbf{x}_i \succ_{\bar{\mathbf{r}}} \mathbf{x}_j, \end{aligned} \quad (15)$$

where ξ_{ij} is a slack variable.

By introducing a factor a which is a small positive constant the following quadratic optimization problem is achieved,

$$\begin{aligned} \min_{\mathbf{r}} \quad & \frac{1}{2} \sum_{i,j} w_{ij} (r_i - r_j)^2 + c \sum_{(i,j) \in \mathcal{S}_{\bar{\mathbf{r}}}} \xi_{ij}^2 \\ \text{s.t.} \quad & r_i - r_j \geq a - \xi_{ij}, \text{ when } \mathbf{x}_i \succ_{\bar{\mathbf{r}}} \mathbf{x}_j. \end{aligned} \quad (16)$$

Reranking with the above optimization problem is called *hinge reranking* since hinge distance is adopted. The optimization problem (16) can be solved using Interior-Point method [16]. In some situations the computational cost is high especially when the constraints are excessive. For instance, if there are 1000 samples in the initial ranking list, there will be about one million constraints. Below, we will develop a more efficient method using a different ranking distance, which can be solved analytically by matrix computation.

4.3 Preference Strength Reranking

In reranking, not only the order relation but also the preference strength, which means the score difference of the samples in a pair, i.e., $r_i - r_j$ for the pair $(\mathbf{x}_i, \mathbf{x}_j)$, is indicative. For example, given two pairs, one comprising two tigers with different typicality, and the other comprising a tiger and a stone, obviously the preference strength is different for these two pairs. Such information can be utilized in video search

reranking and then we define an alternative ranking distance, called *preference strength distance*, as follows

$$\begin{aligned} \text{Dist}(\mathbf{r}, \bar{\mathbf{r}}) &= \sum_{(i,j) \in \mathcal{S}_{\bar{\mathbf{r}}}} d((r_i, r_j), (\bar{r}_i, \bar{r}_j)) \\ &= \sum_{(i,j) \in \mathcal{S}_{\bar{\mathbf{r}}}} \left(1 - \frac{r_i - r_j}{\bar{r}_i - \bar{r}_j}\right)^2. \end{aligned} \quad (17)$$

From Eq.(17) we can see that with preference strength the order relations on pairs are also reflected in preference strength ranking distance.

Replacing the distance function in (11) with the preference strength distance (17), the optimization problem of *preference strength reranking* is

$$\min_{\mathbf{r}} \frac{1}{2} \sum_{i,j} w_{ij} (r_i - r_j)^2 + c \sum_{(i,j) \in \mathcal{S}_{\bar{\mathbf{r}}}} \left(1 - \frac{r_i - r_j}{\bar{r}_i - \bar{r}_j}\right)^2. \quad (18)$$

Supposing one solution of (18) is \mathbf{r}^* , it is apparent that $\mathbf{r}' = \mathbf{r}^* + \mu \mathbf{e}$ is also the solution of (18), where \mathbf{e} is a vector with all elements equal 1 and μ is an arbitrary constant. Obviously, all solutions give the same ranking list. Here, we simply add a constraint $r_N=0$ to (18) where N is the length of \mathbf{r} and then the unique solution can be derived, as given in the following proposition.

PROPOSITION 1. *The solution of (18) with a constraint $r_N=0$ is*

$$\mathbf{r} = \frac{1}{2} \tilde{\mathbf{L}}^{-1} \tilde{\mathbf{c}},$$

where $\tilde{\mathbf{L}}$ and $\tilde{\mathbf{c}}$ are obtained by replacing the last row of $\tilde{\mathbf{L}}$ with $[0, 0, \dots, 0, 1]_{1 \times N}$ and last element of $\tilde{\mathbf{c}}$ with zero respectively. $\tilde{\mathbf{L}} = \tilde{\mathbf{D}} - \tilde{\mathbf{W}}$ and $\tilde{\mathbf{c}} = 2c(\mathbf{A}\mathbf{e})^T$. $\tilde{\mathbf{W}} = [\tilde{w}_{ij}]_{N \times N}$ with $\tilde{w}_{ij} = w_{ij} + c\alpha_{ij}^2$. $\tilde{\mathbf{D}} = \text{Diag}(\tilde{\mathbf{d}})$ is a degree matrix with $\tilde{\mathbf{d}} = [\tilde{d}_1, \dots, \tilde{d}_N]^T$ and $\tilde{d}_i = \sum_j \tilde{w}_{ij}$. $\mathbf{A} = [\alpha_{ij}]_{N \times N}$ is a anti-symmetric matrix with $\alpha_{ij} = 1/(\bar{r}_i - \bar{r}_j)$.

PROOF.

$$\begin{aligned} &\min_{\mathbf{r}} \frac{1}{2} \sum_{i,j} w_{ij} (r_i - r_j)^2 + c \sum_{(i,j) \in \mathcal{S}_{\bar{\mathbf{r}}}} (1 - \alpha_{ij}(r_i - r_j))^2 \\ &= \min_{\mathbf{r}} \frac{1}{2} \sum_{i,j} w_{ij} (r_i - r_j)^2 + c \sum_{(i,j) \in \mathcal{S}_{\bar{\mathbf{r}}}} \alpha_{ij}^2 (r_i - r_j)^2 \\ &\quad - 2c \sum_{(i,j) \in \mathcal{S}_{\bar{\mathbf{r}}}} \alpha_{ij} (r_i - r_j) + \text{const} \\ &= \min_{\mathbf{r}} \frac{1}{2} \sum_{i,j} (w_{ij} + c\alpha_{ij}^2) (r_i - r_j)^2 \\ &\quad - 2c \sum_{(i,j) \in \mathcal{S}_{\bar{\mathbf{r}}}} \alpha_{ij} (r_i - r_j) \\ &= \min_{\mathbf{r}} \frac{1}{2} \sum_{i,j} \tilde{w}_{ij} (r_i - r_j)^2 - 2c \sum_{(i,j) \in \mathcal{S}_{\bar{\mathbf{r}}}} \alpha_{ij} (r_i - r_j) \\ &= \min_{\mathbf{r}} \mathbf{r}^T \tilde{\mathbf{L}} \mathbf{r} - \tilde{\mathbf{c}}^T \mathbf{r}. \end{aligned}$$

Take derivatives and equate it to zero gives:

$$2\tilde{\mathbf{L}}\mathbf{r} = \tilde{\mathbf{c}} \quad (19)$$

The solution of (19) is non-unique since the Laplacian matrix $\tilde{\mathbf{L}}$ is singular [4]. With the constraint $r_N=0$, we can replace the last row of $\tilde{\mathbf{L}}$ with $[0, 0, \dots, 0, 1]_{1 \times N}$ to obtain $\check{\mathbf{L}}$ and the last element of $\tilde{\mathbf{c}}$ with zero to obtain $\check{\mathbf{c}}$ respectively. Then, the solution is

$$\mathbf{r} = \frac{1}{2} \check{\mathbf{L}}^{-1} \check{\mathbf{c}}$$

□

5. IMPLEMENTATION

As aforementioned, we have developed two methods for reranking. When applied into video search reranking some implementation details should be considered.

5.1 Pair Selection

As can be observed in Eq.(11), ranking distance is actually employed to preserve the information of initial score list to some extent. Currently, all the pairs in $\mathcal{S}_{\bar{\mathbf{r}}}$ are involved; however, it will be better to preserve a portion of the pairs, i.e., a subset of $\mathcal{S}_{\bar{\mathbf{r}}}$, in some situations where the initial score list is very noisy. Below, we will introduce some methods to select the appropriate pairs.

• Clustering(C)

We directly use the pair selection method proposed in [11]. In this method, the pseudo-positive samples are selected by clustering and then the bottom-ranked samples in the initial ranking list are selected as pseudo-negative. Each one of the pseudo-positive samples and one of the pseudo-negative samples are selected to form a pseudo-positive pair.

• Top and Bottom(TB)

Usually, the samples ranked at the top of the initial ranking list are more likely to be positive than those which are ranked at the bottom. Based on this assumption, the top-ranked samples in the initial ranking list are selected as pseudo-positive and the bottom-ranked samples as pseudo-negative and then formed into pseudo-positive pairs.

• ρ -adjacent

The ρ -adjacent pair is defined as below.

DEFINITION 6. *A pair $(\mathbf{x}_i, \mathbf{x}_j)$ which satisfies $0 < l_{\mathbf{x}_j} - l_{\mathbf{x}_i} \leq \rho$, where $l_{\mathbf{x}_i}$ and $l_{\mathbf{x}_j}$ are the ranks of \mathbf{x}_i and \mathbf{x}_j in ranking list \mathbf{l} derived from score list \mathbf{r} , is ρ -adjacent pair in \mathbf{l} (or ρ -adjacent pair in \mathbf{r}).*

For a ranking list $\mathbf{l} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$, the 1-adjacent pairs are $\{(\mathbf{x}_1, \mathbf{x}_2), (\mathbf{x}_2, \mathbf{x}_3), \dots, (\mathbf{x}_{N-1}, \mathbf{x}_N)\}$. It is apparent that all other pairs can be deduced from the 1-adjacent pairs based on the transition property of the pair relation $\succ_{\mathbf{r}}$. From this perspective 1-adjacent pairs actually preserve all the necessary information of the ranking list. However, as can be seen from the optimization objectives that the preservation of pair relations is not hard due to the trade-off parameter c , other pairs than the 1-adjacent ones could also be useful. To be noted is that all the pairs in $\mathcal{S}_{\bar{\mathbf{r}}}$ are used when $\rho = N - 1$ where N is the sample count in the ranking list.

5.2 Initial Score

In video search, the performance of text baseline is often poor and the text scores are mostly unreliable because of the inaccuracy and mismatch of ASR and MT from the video. Besides, in some situations the text search scores are unavailable for reranking, e.g., in web image search. Below we propose three strategies to assign the initial scores.

- Normalized Text Score (NTS)

The initial scores $\bar{\mathbf{r}}$ can be assigned by normalizing the text scores \mathbf{r}^{text} into $[0, 1]$ as follows

$$\bar{r}_i = \frac{r_i^{text} - r_{min}^{text}}{r_{max}^{text} - r_{min}^{text}}.$$

- Normalized Rank (NR)

The normalized rank is widely used to estimate the sample’s relevance probability [7, 11, 19], which will be employed to assign the initial scores as

$$\bar{r}_i = 1 - i/N, \quad i = 1, \dots, N.$$

- Rank (R)

Different from NR, the rank can be used directly as the initial score without normalization

$$\bar{r}_i = N - i, \quad i = 1, \dots, N.$$

6. DISCUSSION

6.1 Connection to “Learning to Rank”

Firstly we define the ranking function analogical to reranking function.

DEFINITION 7. A ranking function is defined as

$$\mathbf{r} = f(\mathcal{K}),$$

where $\mathcal{K} = \{\mathbf{k}_j\}$ is a set of features with \mathbf{k}_j being extracted from the pair comprising the query q and the sample \mathbf{x}_j , and \mathbf{r} is the objective ranking score list.

The goal of most “learning to rank” methods [2, 6] is to learn a ranking function automatically from the training data,

$$f^* = \arg \max_f p(f|\{\mathcal{K}^i, \mathbf{r}^i\}_{i=1}^m), \quad (20)$$

and then predict the ranking score list of the samples under a test query q_t using the learned ranking function,

$$\mathbf{r}^t = f^*(\mathcal{K}^t),$$

where \mathcal{K}^t is the test feature set extracted from pairs of the test query q_t and samples, $\{\mathcal{K}^i, \mathbf{r}^i\}_{i=1}^m$ is the training data comprising m pre-labeled ranking lists for m queries $\{q_i\}$.

Reranking can be formulated as a learning to rank problem. Firstly a fraction of the initial ranking score list is selected based on some strategy as shown in Section 5.1; then the selected fractions of the initial ranking list are used to learn an optimal ranking function; finally the reranked list can be achieved using the learned ranking function. This is actually the method used in [11], which adopts Ranking SVM to learn a pair-wise ranking function.

The problem (20) can be regarded as inductive learning to rank, which learns an explicit ranking function without utilizing the unlabeled data. In reranking, however, an explicit ranking function is not necessarily needed and what we desire is just the reranked score list. A more effective way should be to deduce the optimal ranking list from the training data directly without explicitly learning a ranking function as

$$\mathbf{r}^t = \arg \max_{\mathbf{r}} p(\mathbf{r}|\mathcal{K}^t, \{\mathcal{K}^i, \mathbf{r}^i\}_{i=1}^m), \quad (21)$$

which corresponds to the transduction paradigm in machine learning.

Rewriting the reranking objective (2) as

$$\mathbf{r}^* = \arg \max_{\mathbf{r}} p(\mathbf{r}|\mathcal{X}, \{\mathcal{X}, \bar{\mathbf{r}}\}). \quad (22)$$

Since in reranking only one query is involved the features are extracted from the samples regardless of the query. Except this the objective (21) and (22) have the same form. We can see that reranking is actually transductive learning to rank with only one training sample, i.e., the initial ranking score list. From this perspective, the proposed hinge reranking and preference strength reranking can be applied as transductive learning to rank method as well. Meanwhile, any transductive learning to rank methods which will be developed in the future can be used for reranking seamlessly.

6.2 Connection to Random Walk Reranking

The objective function of random walk reranking [8] is derived as

$$\frac{1}{2} \sum_{i,j} w_{ij} \left(\frac{r_i}{d_i} - \frac{r_j}{d_j} \right)^2 + \sum_i \frac{1}{d_i} (r_i - \bar{r}_i)^2,$$

from which we can see that random walk reranking actually have the similar objective as Bayesian Reranking (11). The two terms in the objective function correspond to the visual consistency regularization and the normalized point-wise ranking distance respectively.

7. EXPERIMENTS

In this section, we evaluate our reranking methods on a widely used video search benchmark and compare them with several existing approaches. Besides, we also discuss the influence of different implementation strategies and parameters in our methods.

7.1 Experimental Setting

We conduct the experiments on automatic search task over the TRECVID 2006 and 2007 video search benchmark [15]. TRECVID 2006 dataset consists of 259 videos with 79,484 shots while TRECVID 2007 dataset consists of 109 videos with 18,142 shots. The data are collected from English, Chinese, and Arabic news programs, accompanied with automatic speech recognition (ASR) and machine translation (MT) transcripts in English provided by NIST [17].

The text search baseline we used in this paper is based on the Okapi BM-25 formula [14] using ASR/MT transcripts at shot level. For each of the 48 queries, 24 for TRECVID 2006 and TRECVID 2007 respectively, at most top 1400 relevant shots are returned as initial text search result.

The low-level feature we used in reranking is 225-Dimensional block-wise color moments extracted over 5x5 fixed grid partitions, each block of which is described by a 9-Dimensional feature. When constructing the graph \mathcal{G} , each sample is connected with its K-nearest neighbors.

The performance is measured by the widely used non-interpolated Average Precision (AP)[1]. We average the APs over all the 24 queries in each year to get the Mean Average Precision (MAP) to measure the overall performance.

7.2 Performance Comparison

We compare our two methods: hinge reranking and Preference Strength (PS) reranking with random walk [8] and

Table 2: MAP comparison between different reranking methods on TRECVID 2006 and 2007.

Method	TRECVID 2006		TRECVID 2007	
	MAP	Gain	MAP	Gain
Text Baseline	0.0381	-	0.0306	-
Random Walk	0.0398	4.46%	0.0318	3.90%
Ranking SVM PRF	0.0421	10.50%	0.0315	2.94%
GRF	0.0430	12.86%	0.0321	4.90%
LGC	0.0424	11.29%	0.0351	14.71%
Hinge Reranking	0.0399	4.72%	0.0493	61.11%
PS Reranking	0.0461	21.00%	0.0445	45.42%

Ranking SVM PRF [11]. In addition, we also compare them with two graph based transductive learning methods: GRF (Gaussian Random Filed) [22] and LGC (Local and Global Consistency) [21], which have the same form of objective functions as (11) so can be adopted into reranking directly. For GRF the point-wise ranking distance (12) and Laplacian regularization (8) are used while for LGC the point-wise ranking distance (12) and the Normalized Laplacian regularization (9) are adopted.

The ρ -adjacent strategy for pair selection and R strategy for initial score are adopted in our methods. The parameters are selected in global optimum for all the methods. However, in hinge reranking, K is fixed to 30 when constructing the graph \mathcal{G} and only the top 500 samples in the text search result are involved considering for the efficiency in practical application.

The experimental results are summarized in Table 2. PS reranking achieves consistent and significant performance improvements on both years, 21% on TRECVID 2006 and 45.42% on TRECVID 2007. Hinge reranking performs the best and obtains 61.11% improvement on TRECVID 2007 while little improvement is achieved on TRECVID 2006. As aforementioned, in hinge reranking, we only rerank the top 500 samples. However, the percentage of the positive samples in the top 500 among all the positive samples in the text based search result is 65.9% on TRECVID 2006. The remaining 34.1% positive samples beyond the top 500 will keep untouched in reranking.

The performances of the proposed methods on each query in TRECVID 2006 and 2007 are illustrated in Figure 6 and Figure 7. As can be seen, the proposed hinge reranking and PS reranking methods achieve performance improvements on most of the queries relative to the text search baseline. Significant improvements are obtained on some queries, such as query188 (“burning with flames visible”), query195 (“soccer goalposts”), query196 (“scenes with snow”), and query219 (“the Cook character in the Klokhuis series”). For these queries, the relevant samples share high visual similarity, which is coherent with the visual consistency assumption. Remarkable improvements on these queries demonstrate also the effectiveness of the proposed consistency regularization. On the other hand, for these queries, the text search baselines are better than the others so that the minimization of the ranking distance, i.e., the second term in the objective function (11), captures more useful information.

We can also see that the performances on some queries degrade after reranking, such as query182 (“soldiers or police with one or more weapons and military vehicles”) and

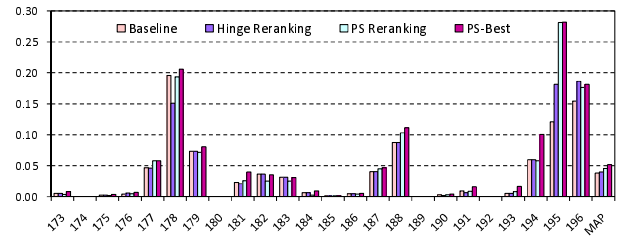


Figure 6: Performance of hinge reranking and PS reranking on TRECVID 2006. In PS-Best, the parameters are selected for each query respectively for PS Reranking.

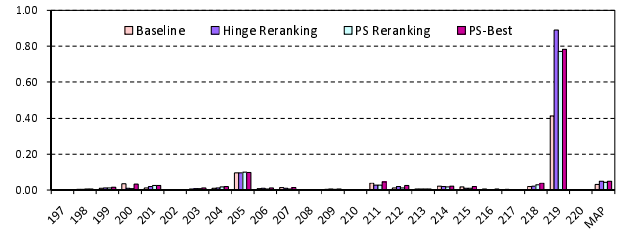


Figure 7: Performance of hinge reranking and PS reranking on TRECVID 2007. In PS-Best, the parameters are selected for each query respectively for PS Reranking.

query200 (“hands at a keyboard typing or using a mouse”). There could be two reasons. One is that the low-level feature is insufficient to represent the visual content with large variations. Hence, in the future semantic similarity will be incorporated into reranking methods. The other reason is that the parameters are set to be the same for each query, which is obviously not optimal. The performances of queries with optimal parameters for each in PS reranking (PS-Best) are shown in the last column of Figure 6 and Figure 7 from which we can see that almost all queries have significant improvements while no queries suffer obvious loss. In the future, we will develop some methods to select optimal parameters for each query or query class automatically.

7.3 Experimental Analysis

Below, we will analyze the performance of the proposed methods and strategies. If not explicitly stated, the experiments and analysis are conducted on PS reranking since the QP solving in hinge reranking is extremely slow when the constraints are excessive. The ρ -adjacent and R are set as default strategies for pair selection and initial score in the following experiments.

7.3.1 Pair Selection

Pair selection is a useful pre-processing step for video search reranking especially when the initial ranking score list is very noisy, as detailed in Section 5.1.

Firstly, we conduct the experiments using ρ -adjacent pairs with different ρ . As shown in Figure 8 1-adjacent pairs achieve significant performance improvements relative to the baseline on both TRECVID 2006 and 2007. The reason may be that 1-adjacent pairs have actually preserved all the necessary information in terms of ranking.

As illustrated in Figure 8, the performance increases with ρ growing on TRECVID 2006. On TRECVID 2007 the per-

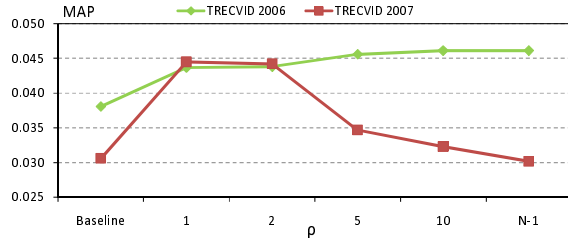


Figure 8: The performances of ρ -adjacent pairs.

Table 3: MAP comparison of pair selection strategies.

Method	TRECVID 2006		TRECVID 2007	
	MAP	Gain	MAP	Gain
Text Baseline	0.0381	-	0.0306	-
C	0.0408	7.09%	0.0227	-25.82%
TB	0.0418	9.71%	0.0120	-60.78%
ρ -adjacent	0.0461	21.00%	0.0445	45.42%

formance degrades when more pairs are added. By observing the data, we argue the reasons are: (1) the text search baseline on TRECVID 2007 is noisier so that better preservation of the initial result is not beneficial; (2) For query219 (“the Cook character in the Klokhuis series”), which basically dominates the performance of TRECVID 2007 as illustrated in Figure 7, there are only six relevant ones among the 575 samples in the text search baseline, so that preserving the order of the pairs far from each other in the initial ranking list brings no useful information but noises. In the following experiments, we will set $\rho=1$ for TRECVID 2006 and $\rho = N - 1$ for TRECVID 2007.

The performance of ρ -adjacent pair selection is compared with the other strategies in Table 3. As shown, ρ -adjacent outperforms the others on both TRECVID 2006 and 2007 data set though the other two methods generate more “correct” pairs. The reason is that ρ -adjacent pairs preserve all the necessary order information.

7.3.2 Initial Score

Different strategies for initial scores as presented in Section 5.2 should have different effects for PS reranking. We will show some observations on it. As shown in Table 4, R and NR, which only use the rank instead of text scores, outperform NTS on both TRECVID 2006 and TRECVID 2007. We argue the reason could be that the text search scores are not so reliable.

In addition, R performs better than NR especially on TRECVID 2006. The reason could be as follows. In R, the preference strength for 1-adjacent pairs equals 1 for each query. In NR, however, the preference strength for 1-adjacent pairs is $1/N$, which is different among queries. Based on the statistics, N varies from 52 to 1400 in TRECVID 2006 and from 28 to 1400 in TRECVID 2007. The optimal parameters, such as the tradeoff parameter c , should be different according to the preference strength, as can be observed in the optimization objective. Since in our experiment the parameters are globally selected, it is more appropriate to assign each query with equal preference strength for 1-adjacent pairs, i.e., R is much better in this situation.

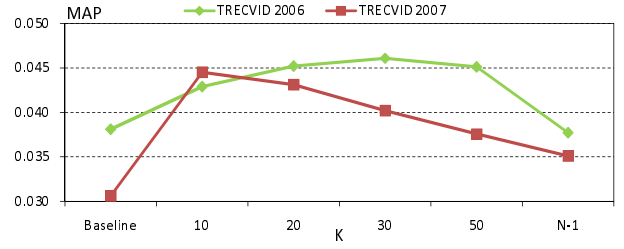


Figure 9: The performance with different K .

Table 4: MAP comparison of different \bar{r} strategies.

Method	TRECVID 2006		TRECVID 2007	
	MAP	Gain	MAP	Gain
Text Baseline	0.0381	-	0.0306	-
NTS	0.0382	0.26%	0.0310	1.31%
NR	0.0387	1.57%	0.0402	31.37%
R	0.0461	21.00%	0.0445	45.42%

7.3.3 The Number of Nearest-neighbors

K is an important parameter when constructing the graph \mathcal{G} . A larger K can ensure more relevant samples being connected to each other, however, the edges between relevant and irrelevant samples will be added, too, which could degrade the performance because the score consistency between relevant and irrelevant samples is not necessary. With a smaller K , the “incorrect” edges will be eliminated while some of the “correct” edges between relevant samples are also missed, which will weaken the necessary consistency.

The MAP- K curve is presented in Figure 9. $K=N-1$ means the graph is fully connected. As illustrated, on both TRECVID 2006 and 2007, the MAP increases with K growing and arrives at a peak at a certain K . However, the MAP decreases gradually when K larger than the peak point.

Specifically, on TRECVID 2006 the maximum MAP (0.0461) is obtained at $K=30$ while on TRECVID 2007 the maximum MAP (0.0445) is achieved at $K=10$. As analyzed from the data, the average numbers of relevant samples across queries are 55 in TRECVID 2006 and 24 in TRECVID 2007. We can observe that the optimal K are about half the average number of relevant samples. This can provide a rough guideline in setting K in the future practical application.

7.3.4 The Trade-off Parameter

The trade-off parameter c is used to balance the effects of the two terms: consistency regularization and ranking distance. A larger c indicates that more information is preserved from the text search baseline into the reranked list. When $c = \infty$, the reranked list will be the same as the initial one if all the pairs are used. A small c means that the visual consistency term plays a major role in reranking. When $c = 0$, the result would be totally dominated by the visual consistency regardless of the initial ranking score list at all.

As illustrated in Figure 10, the performance varies according to different c . The reranking method used is PS Reranking and only 1-adjacent pairs are used. We can see that the MAP increases with c growing and arrives at the peak at $c=100$ on TRECVID 2006 while on TRECVID 2007 the MAP gets the peak at $c=0.01$.

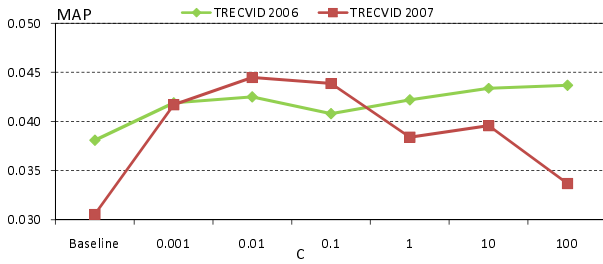


Figure 10: The performance with variant c .

As shown before, the MAP of the text search baseline of TRECVID 2006 and 2007 are 0.0381 and 0.0306 respectively. However, the performance of TRECVID 2007 text search baseline is dominated by the query219 (“the Cook character in the Klokhuis series”) while it would reduce to 0.0139 if such a query is removed. Basically, a larger c (100) is appropriate for a better text search baseline as in TRECVID 2006, while a smaller c (0.01) for worse text search baseline as in TRECVID 2007. It can be concluded that the trade-off parameter c can be set according to the performance of text search baseline.

c is also related to the number of pairs used. Specifically, on TRECVID 2006, the optimal c is 100, 10 and 0.01 respectively for 1-adjacent, 10-adjacent and $(N-1)$ -adjacent pairs.

8. CONCLUSION AND FUTURE WORK

In this paper, a general framework for video search reranking is proposed which explicitly formulate reranking into a global optimization problem from the Bayesian perspective. Under this framework, with two novel pair-wise ranking distances, two effective video search reranking methods, hinge reranking and preference strength reranking, are proposed. The experiments conducted on the TRECVID dataset have demonstrated that our methods outperform several existing reranking approaches.

We will conduct our future work in three directions. The first one is to speed-up the computation of hinge reranking via working set selection approach. To better represent the visual consistency, the semantic similarity which can be learned using distance metric learning [20] will be incorporated into the reranking objective function. Besides, we will also develop some model selection methods to automatically select optimal parameters on query class level.

9. REFERENCES

- [1] Trec-10 proceedings appendix on common evaluation measures. <http://trec.nist.gov/pubs/trec10/appendices/measures.pdf>.
- [2] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li. Learning to rank: from pairwise approach to listwise approach. In *ICML*, pages 129–136, 2007.
- [3] J. G. Carbonell, Y. Yang, R. E. Frederking, R. D. Brown, Y. Geng, and D. Lee. Translingual information retrieval: A comparative evaluation. In *IJCAI (1)*, pages 708–715, 1997.
- [4] L. B. Cremeant and R. M. Murra. Stability analysis of interconnected nonlinear systems under matrix feedback. In *Proceedings of the 42nd IEEE Conference on Decision and Control, Maui, Hawaii USA*, 2003.
- [5] R. Fergus, P. Perona, and A. Zisserman. A visual category filter for google images. In *ECCV (1)*, pages 242–256, 2004.
- [6] R. Herbrich, T. Graepel, and K. Obermayer. Large margin rank boundaries for ordinal regression. In *Advances in Large Margin Classifiers*, pages 115–132. MIT Press, Cambridge, MA, 2000.
- [7] W. H. Hsu, L. S. Kennedy, and S.-F. Chang. Video search reranking via information bottleneck principle. In *ACM Multimedia*, pages 35–44, 2006.
- [8] W. H. Hsu, L. S. Kennedy, and S.-F. Chang. Video search reranking through random walk over document-level context graph. In *ACM Multimedia*, pages 971–980, 2007.
- [9] L. S. Kennedy and S.-F. Chang. A reranking approach for context-based concept fusion in video indexing and retrieval. In *CIVR*, pages 333–340, 2007.
- [10] J. Liu, W. Lai, X.-S. Hua, Y. Huang, and S. Li. Video search re-ranking via multi-graph propagation. In *ACM Multimedia*, pages 208–217, 2007.
- [11] Y. Liu, T. Mei, J. Tang, X. Wu, and X.-S. Hua. Learning to video search rerank via pseudo preference feedback. In *ICME*, 2008.
- [12] T. Mei, X.-S. Hua, W. Lai, L. Yang, Z.-J. Zha, Y. Liu, Z. Gu, G.-J. Qi, M. Wang, J. Tang, X. Yuan, Z. Lu, and J. Liu. Msra-ustc-sjtu at trecvid 2007: High-level feature extraction and search. In *TREC Video Retrieval Evaluation Online Proceedings*, 2007.
- [13] A. Natsev, A. Haubold, J. Tesic, L. Xie, and R. Yan. Semantic concept-based query expansion and re-ranking for multimedia retrieval. In *ACM Multimedia*, pages 991–1000, 2007.
- [14] S. E. Robertson, S. Walker, M. Hancock-Beaulieu, M. Gatford, and A. Payne. Simple, proven approaches to text retrieval. In *Cambridge University Computer Laboratory Technical Report TR356*, 1997.
- [15] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and trecvid. In *MIR ’06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, pages 321–330, New York, NY, USA, 2006. ACM Press.
- [16] A. J. Smola. pr_LOQO, <http://www.kernel-machines.org/code/prloqo.tar.gz>.
- [17] TRECVID. TREC Video Retrieval Evaluation. <http://www-nlpir.nist.gov/projects/trecvid/>.
- [18] R. Yan and A. G. Hauptmann. Co-retrieval: A boosted reranking approach for video retrieval. In *CIVR*, pages 60–69, 2004.
- [19] R. Yan, A. G. Hauptmann, and R. Jin. Multimedia search with pseudo-relevance feedback. In *CIVR*, pages 238–247, 2003.
- [20] L. Yang and R. Jin. Distance metric learning: A comprehensive survey. http://www.cse.msu.edu/~yangliu1/frame_survey_v2.pdf. 2006.
- [21] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. In *NIPS*, 2003.
- [22] X. Zhu, Z. Ghahramani, and J. D. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *ICML*, pages 912–919, 2003.