The Combination Limit in Multimedia Retrieval

Rong Yan
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA, 15213
yanrong@cs.cmu.edu

Alexander G. Hauptmann
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA, 15213
alex@cs.cmu.edu

ABSTRACT

Combining search results from multimedia sources is crucial for dealing with heterogeneous multimedia data, particularly in multimedia retrieval where a final ranked list of items of interest is returned sorted by confidence or relevance. However, relatively little attention has been given to combination functions, especially their upper bound performance limits. This paper presents a theoretical framework for studying upper bounds for two types of combination functions. A general upper bound and two approximations are proposed for monotonic combination functions. We also studied the upper bounds for linear combination functions using a global optimization technique. Our experimental results show that the choice of combination functions has a considerable influence to retrieval performance.

1. INTRODUCTION

The growing amount of multimedia data in the form of video is driving the demand for content-based access to video information, known as content-based video retrieval or sometimes also referred to as content-based multimedia retrieval. Compared to image retrieval, multimedia retrieval presents new opportunities because more complete and diverse information can be extracted through facets of multimedia sources such as speech transcript text, audio, camera motion and visual features. Fusing these retrieval results into a final score is a crucial step in video retrieval[6, 8]. Even though the confidence for each single media component might be weak, their combination, however, makes it possible to utilize cross-modal relationships and thus boost the performance beyond any single component. The hope is that random errors in unrelated modalities will cancel out.

However, this also brings up new challenges because multimedia data are more heterogeneous than traditional data like text[2]. Due to this heterogeneity, a multimedia retrieval system typically first associates each video shot with a vector of individual retrieval scores from single media search module and generates multiple rank lists. For example, a shot might have scores (t_1, t_2) where t_1 is visual similarity and t_2 is speech transcript text similarity to a video query. Then the system fuses all these scores into a final ordered list. Different fusion strategies for multimodal information have been proposed, including linear combination[8], min/max aggregation[6] as well as machine learning methods such as bayesian networks[5]. However, several open questions remain: what are the limits of these combination methods if we already have scores for each different component? Is linear combination sufficient or are more complex functions necessary? How should scores be normalized when combining them? Should we simply assign equal weights for all media components in all queries? By viewing video retrieval as a "rank aggregation" problem in this paper, we are developing a theoretical framework for studying the performance limits over both monotonic and linear combination functions assuming the retrieval results from every media component are known. This analysis gives us some answers to the above questions and may help in boosting the dismal performance of all current multimedia retrieval systems.

2. BASIC DEFINITIONS

Given a video collection D, a *full ordered list* (or a list) τ of D is an ordering (ranking) of the full set D, i.e., $\tau = [x_1 \geq x_2 \geq \cdots \geq x_n]$, with each $x_i \in D$, and $t_\tau(x_i)$ is the corresponding retrieval score for each $x \in D$, which satisfies $t_\tau(x_1) \geq \cdots \geq t_\tau(x_n) \geq 0$. Also, let $\tau(i)$ be the shot at the position or rank i (lower number means highly ranked shots), $\tau^{-1}(x)$ be the rank of shot x. We define a useful operation called *projection*. Given a list τ and a subset T of the collection D, the projection of τ with respect to T (denoted $\tau_{|T}$) will be a new full list which contains only the shots from T. Also, we define another operation called *precedence* $\mu_\tau(x)$ as $\{y|t_\tau(y) \geq t_\tau(x)\}$, which is the set of video shots that have no higher rank than the shot x in the rank list τ. Similarly, the operation *multi-precedence* $\mu_{\tau_1,...\tau_k}(x)$ can be defined as $\{y|t_{\tau_1}(y) \geq t_{\tau_1}(x) \wedge ... \wedge t_{\tau_k}(y) \geq t_{\tau_k}(x)\}$ where $\wedge$ is conjunction. Moreover, $|\cdot|$ is the number of shots in the set, for instance $|D|$ is the number of shots in set D.

In this paper, our goal is to study the performance limits of rank combination strategies if the rank lists from multiple components τ_i . More specifically, our problem says: given the video collection D, the collection of relevant video shots D^+ and the ordered lists $\tau_1, ..., \tau_k$ from k different components of video, find the maximum retrieval performance for a final order list $\sigma = F(\tau_1, ..., \tau_k)$ with respect to the collection D, where F is the combination function. In our

MM'03, November 2–8, 2003, Berkeley, California, USA.

discussion, we assume that the combination function F is supposed to be monotonic, which means $t_\sigma(y) \geq t_\sigma(x)$ if $t_{\tau_i}(y) \geq t_{\tau_i}(x)$ for every i. It can be shown that monotonicity[2] is a reasonable property to the requirement of an combination function: if for every rank list, the score of a shot y has at least as high as that of shot x, then we can expect the overall score of y to be at least as high as that of x. The following property follows immediately from the definition,

PROPOSITION 1. *If the combination function F is monotonic, then $\mu_{\tau_1,...\tau_k}(x) \subset \mu_\sigma(x)$*

Among the different types of performance measures, average precision (AP) is adopted in this paper in accordance with the TREC'02 video track[7]. To compute average precision, the precision after every retrieved relevant shot is computed, and these precisions are averaged over the total number of retrieved relevant shots in the collection. Let $\sigma^+ = \sigma_{|D^+}$ be the projection of σ with respect to the set of relevant shots D^+. For any list σ with respect to D, the average precision $AP(\sigma)$ can be defined as,

$$AP(\sigma) = \sum_{x \in D^+} \frac{|\mu_{\sigma^+}(x)|}{|\mu_\sigma(x)|} = \sum_{i=1}^{|D^+|} \frac{i}{|\mu_\sigma(\sigma^+(i))|}. \quad (1)$$

Mean average precision(MAP) is the average of these average precision over all queries in the query set. Note that except when stated otherwise, we will discuss all these upper bounds on the basis of a query, and average them to be the upper bounds for MAP.

3. GENERAL BOUND

We now study the upper bound of average precision for any monotonic combination function F. Recalling the monotonicity assumption of combination function F, if the score of a irrelevant shot y is always as high as that of a relevant shot x for every τ_i, then it is impossible for y to have a lower combination score than x. This property allows us to derive a general upper bound as presented in the following theorem,

THEOREM 1. *Given the lists $\tau_1, ..., \tau_k$, if the combination function F is monotone, then $AP(\sigma)$ is no more than*

$$GB(\sigma^+) = \sum_{i=1}^{|D^+|} i \left| \bigcup_{j \leq i} \mu_{\tau_1,...,\tau_k}(\sigma^+(j)) \right|^{-1}.$$

Proof: For the proof of Theorem 1, let us consider a shot x which is in $\bigcup_{j \leq i} \mu_{\tau_1,...,\tau_k}(\sigma^+(j))$. Without loss of generality, we assume x is in $\mu_{\tau_1,...,\tau_k}(\sigma^+(l))$, where $l \leq i$. Then x is also in $\mu_\sigma(\sigma^+(l))$ from Proposition 1. Given $l \leq i$, $\mu_\sigma(\sigma^+(l))$ is a subset of $\mu_\sigma(\sigma^+(i))$ based on the fact that $\sigma^+(l)$ has a lower rank than $\sigma^+(i)$ with respect to list σ. Combining all these together, we have $x \in \mu_\sigma(\sigma^+(i))$. Therefore, we can show that $\bigcup_{j \leq i} \mu_{\tau_1,...,\tau_k}(\sigma^+(j))$ is a subset of $\mu_\sigma(\sigma^+(i))$. Thus, it yields

$$AP(\sigma) = \sum_{i=1}^{|D^+|} \frac{i}{|\mu_\sigma(\sigma^+(i))|} \leq \sum_{i=1}^{|D^+|} i \left| \bigcup_{j \leq i} \mu_{\tau_1,...,\tau_k}(\sigma^+(j)) \right|^{-1}.$$

Theorem 1 gives us a way to estimate the upper bound for monotonic combination functions. Since the general bound $GB(\sigma^+)$ is only related to the relevant shots' ordering σ^+ instead of the full ordering σ, a straightforward way to find the upper bound is to enumerate all the possible orderings for relevant shots and compute their maximal value $GB = \max_{\sigma^+} GB(\sigma^+)$.

Unfortunately, this cannot be done in reasonable time if the size of relevant shots is too large, since the time complexity grows exponentially to the size of relevant shots. Plugging in some heuristic search algorithms like branch-and-bound algorithm can mitigate the problem to some degree, but in the worst case it still requires exponential time to finish the process. To address this issue, we propose two other bounds which are more computationally efficient,

Loose General Bound For this bound, we simply modify the $GB(\sigma^+)$ to be

$$LGB(\sigma^+) = \sum_{i=1}^{|D^+|} \frac{i}{i - 1 + |\mu_{\tau_1,...,\tau_k}(\sigma^+(i))|}.$$

It can be shown that $LGB(\sigma^+) \geq GB(\sigma^+)$ because $\sigma^+(i) \neq \sigma^+(j)$ for every $j \leq i$ and we have $i - 1 + \left|\mu_{\tau_1,...,\tau_k}(\sigma^+(i))\right|$ is less than $\left|\cup_{j \leq i} \mu_{\tau_1,...,\tau_k}(\sigma^+(j))\right|$. So the loose general bound $LGB = \max_{\sigma^+} LGB(\sigma^+) \geq \max_{\sigma^+} GB(\sigma^+)$.

Greedy General Bound In this bound a greedy algorithm is used to find a suboptimal solution for GB, which is presented as follows,

1. Initialize $R = D^+$. Repeatedly, i from 1 to $|D^+|$
   (a) For every shot in R, find a shot x_i minimize $|\cup_{j \leq i} \mu_{\tau_1,...,\tau_k}(x_j)|$, where $x_j (j < i)$ is the shots chosen before
   (b) Delete x_i from R, $R = R \setminus \{x_i\}$
2. The final list $\sigma_g^+ = [x_1 \geq ... \geq x_{|D^+|}]$. The greedy general bound $GGB = GB(\sigma_g^+)$

This algorithm can provide a sub-optimal result to GB but its time complexity is only quadratic to $|D^+|$. Since σ_g^+ is a special case of σ_+, we can have $GGB \leq \max_{\sigma^+} GB(\sigma^+) = GB$

Putting these inequalities together, we can find that the general bound falls in the range between LGB and GGB, i.e. $LGB \geq GB \geq GGB$, however both LGB and GGB can be computed in polynomial time with respect to the size of relevant shots. Therefore, only these two bounds are computed in our experiments.

4. LINEAR BOUND

Thus far we provided a general upper retrieval bound for monotonic functions without assuming a specific function form. However, in practice more constraints might be taken into account for the combination functions due to various factors such as function continuity, implementation efficiency and user customizability. Let us consider a typical scenario that users want to assign different weights to the information from various components. Therefore, it is reasonable to associate some weighting factors $\theta_i (0 \leq \theta_i \leq 1)$ to each retrieval score t_{τ_i}, i.e. for any shot x, the retrieval score $t_\sigma(x) = F_{(\theta_1,..,\theta_k)}(t_{\tau_1}(x), .., t_{\tau_k}(x))$, or we simply write $t_\sigma = F_{(\theta_1,..,\theta_k)}(t_{\tau_1}, .., t_{\tau_k})$. The first desirable property says

intuitively that if a particular component has zero weight, then the component's score can be dropped without affecting the results, which we call compatibility

Compatibility $F_{(0,..,0,\theta_i,0,..,0)}(t_{\tau_1},..,t_{\tau_k}) = F_{(0,..,0,\theta_i,0,..,0)}(t_{\tau_i})$

For the sake of simplicity, let us denote $\Theta = (\theta_1,..,\theta_k)$, $T = (t_{\tau_1},..,t_{\tau_k})$. Considering the implementation efficiency, we propose that another natural desirable property for the combination function is linearity,

Linearity $F_{\alpha\cdot\Theta+\beta\cdot\Theta'}(T) = \alpha \cdot F_\Theta(T) + \beta \cdot F_{\Theta'}(T)$

Given these two natural properties, we can uniquely determine the form of the combination function using the following Theorem,

THEOREM 2. *The combination function has the form* $F_\Theta(T) = \sum_i \theta_i f_i(t_{\tau_i})$ *if and only if it satisfies both compatibility and linearity*

Proof Sketch: It is easy to show $\sum_i \theta_i f_i(t_{\tau_i})$ satisfies compatibility and linearity. Now we prove uniqueness. First, based on linearity, we can show by induction $F_{\sum_{i=1}^k \alpha_i\cdot\Theta_i}(T) = \sum_{i=1}^k \alpha_i \cdot F_{\Theta_i}(T)$. Given k linear independent points $E_1,...,E_k$ where E_i is row i of a $k\times k$ identity matrix, it can be verified that $\Theta = \sum_{i=1}^k \theta_i E_i$. Therefore, $F_\Theta(T) = \sum_{i=1}^k \theta_i F_{E_i}(T)$. Following the compatibility, we have $F_{E_i}(T) = F_{E_i}(t_{\tau_i})$. So this proves $F_\Theta(T) = \sum_i \theta_i f_i(t_{\tau_i})$ where $f_i(t_{\tau_i}) = F_{E_i}(t_{\tau_i})$.

4.1 General Linear Bound

In the following discussion, we investigate the performance limit for linear combination functions, i.e. the functions of the form $F_\Theta(T) = \sum_i \theta_i f_i(t_{\tau_i})$. We call f_i the transformation function and θ_i the weights over different components. Also, we assume monotonicity and non-negativity for function f_i and therefore F also satisfies these two properties. Let us denote $AP(F_\Theta(T))$ the average precision of order list σ where σ is determined by retrieval score $F_\Theta(T)$ with respect to D. Therefore our task can be rewritten as a bound constrained global optimization problem,

$$\max_{f_i,\theta_i} AP(\sum_i \theta_i f_i(t_{\tau_i})) \quad s.t. \quad \theta_i \in [0,1], i = 1..k. \qquad (2)$$

However, the major difficulty in optimizing the objective function in (2) is to optimize function f_i over the function space $\mathcal{R}^1 \to \mathcal{R}^1$, which is computationally intractable. To address this, we propose a set of well-established basis functions to approximate the function f_i and thus convert this problem into a parameter optimization problem. Hush et al.[3] suggest a linear combination of sigmoid functions, which form the basis set, is an efficient way to approximate the continuous functions. They have shown that under some very general conditions, the approximation error with a sigmoid basis set is bounded by O(1/n). The class of approximating functions considered here are one hidden layer neural network of the form,

$$f(x) = a_0 + \sum_{j=1}^M \frac{a_j}{1+e^{w_j x + b_j}}. \qquad (3)$$

In this context, the sigmoids are referred to as a "tunable" basis function[3] because they can be "tuned" to the data by varying w_i. Note that the constant term a_0 can be dropped here because it does not affect the ordering result. Substituting the reminder of (3) into the $F_\Theta(T)$, it yields

$$F_\Theta(T) = \sum_{i=1}^k \sum_{j=1}^M \frac{a_{ij}\theta_i}{1+e^{w_{ij}t_{\tau_i}+b_{ij}}} = \sum_{j=1}^M \sum_{i=1}^k \frac{\theta'_{ij}}{1+e^{w_{ij}t_{\tau_i}+b_{ij}}}, \quad (4)$$

where $\theta'_{ij} = a_{ij}\theta_i$. Following the monotonicity and nonnegativity of f_i, we can have $a_i \geq 0$, $w_i \leq 0$, and therefore $\theta'_{ij} \geq 0$. This way, the problem is transformed to a constrained global optimization problem over $3kM$ parameters. However, it is difficult to decide M without any prior knowledge of the data and limited computational resources do not allow setting M too high. Therefore, we developed a greedy optimization algorithm to address the problem,

1. Initialize $AP_0 = 0, m = 0$
2. Repeat until $AP_m = AP_{m-1}$
   (a) $m \leftarrow m + 1$
   (b) Compute the optimal value for $\theta'_{im}, w_{im}, b_{im}$ which is, for every $i = 1..k$,

$$arg \max_{\theta'_{im},w_{im},b_{im}} AP\left(\sum_{j=1}^m \sum_{i=1}^k \frac{\theta'_{ij}}{1+e^{w_{ij}t_{\tau_i}+b_{ij}}}\right)$$

   subject to $0 \leq \theta_{im} \leq 1$, $w_{im} < 0$. AP_m is set to the maximal average precision, which is at least as large as AP_{m-1}

Finally the linear bound LB is set to AP_m. In this algorithm, only $3k$ parameters are optimized for a single run, which reduces the number of parameters dramatically. To handle the bounded constraint optimization problem, we use a global optimization algorithm called MCS algorithm as proposed by Huyer et al [4]. Their method combines both global search and local search into one unified framework via multi-level coordinate search. It is guaranteed to converge if the function is continuous. Also in our implementation, multiple start points have been tried for each query to avoid the local minima problems.

4.2 Locally and Globally Fixed Linear Bound

In the previous section, a set of "tunable" sigmoid functions are added to approximate any continuous function. However, in the real case, the transformation functions f_i are always pre-defined to normalize the retrieval scores into a uniform space[6, 8]. This section studies the effect on retrieval performance if the transformation functions are already fixed. Various types of f_i can be chosen such as studentization and range normalization functions[6]. Among these choices, we use a well-known "positional" method called the Borda method[1], which only considers the position of every shot without any information about the score confidence. It assigns the shot's score to be the number of shots ranked below the shot. Formally, the score can be represented as, $f_i(t_{\tau_i}(x)) = (1-\tau_i^{-1}(x))/|D|$. Therefore, we want to optimize the average precision over the weights θ_i,

$$LLB = \max_{\theta_i} AP\left(\sum_{i=1}^k \theta_i f_i(t_{\tau_i})\right),$$

where we call LLB as locally fixed linear bound.

Note that all above discussions assume that users can assign unequal weights for different queries. If we add another

Combination	LGB	GGB	LB	LLB	GLB
T+C+M	0.201	0.192	0.184	0.182	0.157
T+P+M	0.220	0.205	0.194	0.189	0.165
T+P+C+M	0.277	0.254	0.222	0.213	0.165
T+C+X+M	0.310	0.279	0.236	0.224	0.170
T+P+F+C+M	0.306	0.280	0.236	0.225	0.170
T+P+C+X+M	0.359	0.324	0.252	0.240	0.178
T+P+F+C+X+M	0.386	0.348	0.259	0.250	0.179

Table 1: Various upper bound against different combinations of ranking lists for TREC02 video track search set, where T is Text, C is Color, X is Texture, P is PRF, F is Face and M is Movie

constraint, namely that the θ_i cannot be changed across queries, this might give a more tighter bound on the performance, which we call globally fixed linear bound(GLB). For both LLB and GLB, We also use the MCS optimization algorithm to solve it with multiple starting points. To summarize how these bounds are related to each other, we have the following equality, $LGB \geq GB \geq GGB, GB \geq LB \geq LLB \geq GLB$.

5. EXPERIMENTAL RESULT

The video data came from the video collection provided by the TREC Video Retrieval Track[7]. The 2002 Video Collection for the Video TREC retrieval task consisted of 40 hours of MPEG-1 video in the search collection. This corpus translated into 14,524 shots where the boundaries were provided as the common shot reference of the Video TREC evaluation effort. In the following experiment, only the top 100 shots are retrieved for each query. Results are averaged over 25 queries.

In our implementation, six types of retrieval scores are constructed from different video components. On the image processing side, both cumulative color histogram similarity in HSV color space (Color) and texture feature distances generated from various Gabor filters (Texture) were used in our system. A pseudo-relevance feedback (PRF) retrieval score was computed by automatically feeding back the ıleast similar video shots to be the negative examples of a SVM classifier. We used a face detector (Face) to detect the faces in each shot. We also have two types of text-based retrieval scores: one is from speech and Video OCR transcripts using the Okapi document retrieval formula (Text), the other score from externally provided video summaries (Movie). More details for these retrieval scores can be found in [8]. Each of these retrieval 'agents' forms their own ranking list, which will be the basis set τ_i of our combination function.

The upper bound of mean average precision are presented in Table 1. We studied several possible combinations of various retrieval scores. The general bound GB has not been computed, but it is guaranteed to be between LGB and GGB. As expected, all of the upper bounds become higher when more scores are included in the combination function. The linear bound has a performance close to the general bound when the number of basis rank lists k is small, however, we find that when k becomes higher there is a considerable gap between the general upper bound and the linear bound. The highest general bound using all possible rank lists can reach 0.348 - 0.386, however the highest linear bound is only 0.259, which means that a linear combination function might not be sufficient to produce the best performance from large number of components. It also suggests that non-linearity and cross-media relation should be introduced to reach the general bound. Also, the improvement comparing the LLB against the GLB indicates that if different weights θ_i can be assigned to various queries correctly, the retrieval performance will improve considerably. However, since the LLB is generally close to LB, this shows that converting the retrieval scores into positional scores might be a good strategy for linear type combination functions.

6. CONCLUSION

We have presented a theoretical framework for bounding the average precision of monotonic combination function and linear combination function in video retrieval. Our experimental results on the TREC02 video track data show that carefully chosen combination functions could considerably improve the retrieval performance with multimedia information. Linear combinations might be sufficient when fusing small number of components, but too restricted for more components. Future work will include automatically discovering the weights using prior knowledge for types of queries and score distribution information, exploring the relationship between multimedia sources and studying the effects of non-linear combination functions.

Acknowledgments

This research is partially supported by the advanced Research and Development Activity (ARDA) under contract number MDA908-00-C-0037 and MDA904-02-C-0451. We also thank Rong Jin and Yan Liu for stimulating discussion.

7. REFERENCES

[1] C. Dwork, S. R. Kumar, M. Naor, and D. Sivakumar. Rank aggregation methods for the web. In *World Wide Web*, pages 613–622, 2001.

[2] R. Fagin, A. Lotem, and M. Naor. Optimal aggregation algorithms for middleware. In *ACM Symposium on Principles of Database Systems*, 2001.

[3] D. Hush and B. Horne. Efficient algorithms for function approximation with piecewise linear sigmoidal networks. *IEEE Trans. Neural Networks*, 9(6), 1998.

[4] W. Huyer and A. Neumaier. Global optimization by multilevel coordinate search. *Journal Global Optimization*, 14, 1999.

[5] M. Naphade and et al. Probabilistic multimedia objects (multijects): A novel approach to video indexing and retrieval in multimedia systems. In *Proc. of ICIP*, 1998.

[6] J. R. Smith and et al. Interactive search fusion methods for video database retrieval. In *IEEE Intl. Conf. on Image Processing*, Barcelona, Spain, 2003.

[7] TREC2002. TREC2002 video track, http://www-nlpir.nist.gov/projects/t2002v/t2002v.html.

[8] R. Yan, A. Hauptmann, and R. Jin. Multimedia search with pseudo-relevance feedback. In *International Conference on Image and Video Retrieval*, Urbana, IL, USA, 2003.