Modeling video hyperlinks with hypergraph for web video reranking

Hung-Khoon TAN

Chong-wah NGO
Singapore Management University, cwngo@smu.edu.sg

Xiao WU

Modeling Video Hyperlinks with Hypergraph for Web Video Reranking

Hung-Khoon Tan, Chong-Wah Ngo and Xiao Wu
Department of Computer Science
City University Of Hong Kong
Kowloon, Hong Kong
{hktan,cwngo,wuxiao}@cs.cityu.edu.hk

ABSTRACT

In this paper, we investigate a novel approach of exploiting visual-duplicates for web video reranking using hypergraph. Current graph-based reranking approaches consider mainly the pair-wise linking of keyframes and ignore reliability issues that are inherent in such representation. We exploit higher order relation to overcome the issues of missing links in visual-duplicate keyframes and in addition identify the latent relationships among keyframes. Based on hypergraph, we consider two groups of video threads: visual near-duplicate threads and story threads, to hyperlink web videos and describe the higher order information existing in video content. To facilitate reranking using random walk algorithm, the hypergraph is converted to a star-like graph using star expansion algorithm. Experiments on a dataset of 12,790 web videos show that hypergraph reranking can improve web video retrieval up to 45% over the initial ranked result by the video sharing websites and 8.3% over the pair-wise based graph reranking in mean average precision (MAP).

Categories and Subject Descriptors

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

General Terms

Algorithms, Performance, Experimentation

1. INTRODUCTION

Web videos have recently become one of the fastest growing multimedia documents in the web. The growth is spurred by the popularity of Web 2.0 which provides a convenient platform for users to actively participate in the delivery of web contents. In April 2008, a YouTube search returns about 83.4 million videos, while in January 2008 alone, nearly 79 million users watched over 3 billion videos [10]. Therefore, it is critical to ensure that a web video search is convenient and effective where videos related to the query are always pushed to the top of the ranked list. In a typical scenario where users do not possess any sample videos or images to begin with, a text search is the only pragmatic manner to initiate a search. In a re-ranking paradigm, the search result returned by the search engine is further fortified by performing a reorganization of the top ranked web videos in an unsupervised fashion. Each video is associated with an initial relevance score and is described by some features extracted from the accompanying peripheral modalities. Re-ranking strives to find the posterior probability such that videos which are highly relevant to the query are given priority over lesser ones.

We can broadly categorize existing reranking methods into classification-based, information-theoretic and graph-based approaches. Classification-based approaches [3, 6, 12] basically utilize pseudo-relevance feedbacks (PRF) and proceed to formulate reranking as a classification problem. However, the selection of pseudo examples is difficult, especially when relevant videos are scattered randomly across the ranked list. In the information-theoretic approaches, mutual information (MI) has been employed to select the most informative subset of semantic concepts as the underlying features in [6] and to find the best cluster having the most relevant keyframes in [4]. Recently, ordinal reranking [13] which employs ranking functions such as ListNet and RankSVM to mine ordinal relations, has reported a superior performance compared to information-theoretic approaches.

Graph-based approaches build upon the success of PageRank [8] in web page ranking and adopt a similar random walk framework to rerank videos. In [5], each vertex of a graph represents a story which embodies a collection of keyframe each, and an undirected edge links two vertices if any two keyframes from the stories are found to be visually duplicate. In [7], separate graphs are constructed for different concepts, and each vertex represents a keyframe while directed edges are constructed from the initial ranking score. Current graph construction techniques have largely ignored noise issues, in particular when the visual-duplicate or concept detectors are not completely reliable. Such problem, as shown in this paper, could be alleviated by considering the building of hypergraph to model the higher order relationship among keyframes and stories when hyperlinking video content with visual near-duplicates.

1.1 Beyond Pair-wise Linking

Current graph-based approaches [5, 7] consider mainly the pair-wise linking of duplicate or similar keyframes. The ac-

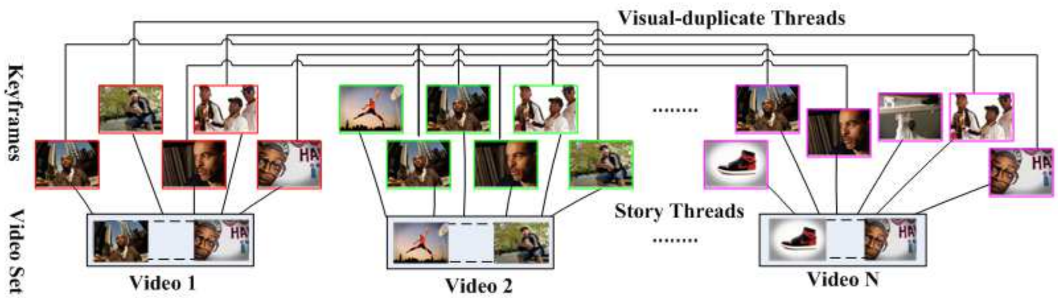

Figure 1: A hypergraph with visual-duplicate hyperedges connecting near-duplicate keyframes, and story hyperedges connecting the keyframes of the same video. Note that video sets are included in this figure for illustration purpose. They are not vertices in the hypergraph.

curacy of linking, which could affect the reranking performance, is dependent on the robustness of visual-duplicate detectors or thresholding of visual similarities. Problems such as missing or falsely detected links can propagate into the graph model and result in erroneous score distribution during random walk. A missing link under-distributes weights to a deserving vertex while an erroneous link over-distributes weights to unrelated vertex. The severity of the missing and false link is dependent on the level of abstraction of a graph model. At the story level abstraction, false detection is a severer problem because when one story is either a video or a group of semantically related shots, false links are propagated into all keyframes in the same story, while missing links are normally compensated by another visual-duplicate pair in the keyframe collection. The scenario is the inverse when keyframe level abstraction is employed.

In this paper, we propose a novel approach of exploiting visual duplicates for web video reranking using hypergraph [15, 2]. Our approach models the higher order relationship among keyframes using visual-duplicate threads which are groups of keyframes sharing near-duplicate content, resulting in a robust way of performing random walk when video content is hyperlinked. Figure 1 illustrates the proposed hypergraph for modeling the hyperlinks among keyframes. Two groups of threads are considered: visual-duplicate threads which group near-duplicate keyframes, and story threads which group keyframes of the same video.

We investigate two important issues for web video ranking. First, using hypergraph, we exploit higher order relations of keyframes to uncover the missing links and in the process mine the latent relation between unlinked keyframes. Second, to tolerate false links, hypergraph is constructed at the keyframe abstraction level and in addition, keyframes originated from the same video are treated as hyperedge to constrain the degree of randomness when traversing hyperlinks. Because the abstraction is at the level of keyframes, false links can be overcome by employing a proper fusion technique to combine the reranked scores of keyframes in a video and therefore is not as severe as missing links. In our approach, we adopt noisy-or model to fuse the scores of keyframes in a video.

2. VIDEO HYPERLINKING

Video hyperlinking can be achieved in different ways, for instance, using concept scores to link video keyframes which share similar semantic concepts [7]. Because our aim is not to demonstrate the effective way of hyperlinking using different video features, we exploit mainly the visual near-duplicates for inter-linking video content. More importantly, in web video search and management, visual duplicates have been repeatedly shown to be a useful resources to explore [11]. In our approach, basically each web video is partitioned into shots and each shot is represented by a keyframe. The detector in [14] is then employed to discover the pairs of keyframes which are visually near-duplicate. Eventually, videos are hyperlinked in a way that near-duplicate keyframes are inter-connected such that traversing of videos is achieved by surfing the links among keyframes. Given an initial rank list of web videos, such simple graph structure is already adequate for conducting PageRank which is shown to be particularly useful for reranking web pages.

While intuitive, video hyperlinking indeed faces the challenge of robustness issue, where the hyperlinks are not created by web users as in the case of web hyperlinks. As an example, the publicly available visual-duplicate detector [14] we use in this paper reported approximately 18% and 33% of missed and false detection respectively in the experiment. The noisy hyperlinks, in the form of missing and false links, could make the reranking performance unpredictable. To tackle this problem, we consider threads, instead of pair-wise links, to inter-connect keyframes.

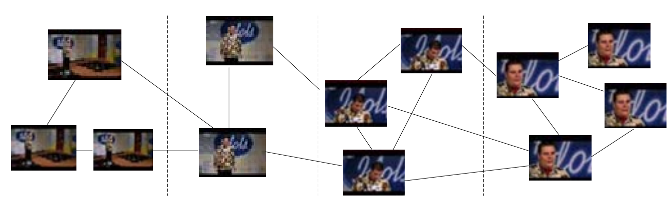

Figure 2: An example of visual-duplicate thread.

Thread is basically a group of keyframes sharing similar property and linked together as a whole. In practice, keyframes should be considered jointly because the cardinality of a thread better reflects the significance of the keyframes. Figure 2 shows an example of a visual-duplicate thread which is built from a connected series of pair-wise links. In this example, when viewed as traditional pair-wise relations and further strained by missing links, each visual-duplicate connects to at most four other vertices whereas in a thread, each keyframe is related to all other keyframes in the thread. Therefore, the pair-wise structure can hardly reflect the true relevance of the keyframes. In addition, higher order relation can span across multiple visual-duplicate series, enlightening the temporal connectivity between keyframes when traversing from one scene to another as shown in the example. In this respect, threading is able to link related keyframes which by definition does not fall into the visual-duplicate category.

Another family of thread is the story thread. Keyframes attributed to the same video exhibit high-level semantic relationship because they narrate the same story. Unlike visual-duplicate thread, the size of a story thread, which in general represents the time duration and content richness of the video clip, does not carry any additional information about the relevance of the video to the query. In this respect, a story thread plays a passive but important role of ensuring that the reranked scores of keyframes within the same story are consistent. In random walk, this is achieved by modeling a story thread as a weight source for all the keyframes connected to it.

3. HYPERGRAPH MODELING

Hypergraph is a generalization of graph to higher order edges. It has been proven in [1] that the higher order relation in hypergraph can be projected to a normal graph through hypergraph approximation. Two most common approximation algorithms are *clique expansion* and *star expansion* [15]. Clique expansion expands each hyperedge into a clique, while star expansion introduces a new vertex for every hyperedge and then connects all vertices in the hyperedge to it. We adopt star expansion because the higher order information of the hyperedges remains intact in the approximated graph where each of the new vertices effectively symbolizes a hyperedge.

Let $G(V, E)$ denote a hypergraph with the vertex set V and hyperedge set E. The edges are arbitrary subsets of V where each hyperedge $e \in E$ is associated with a weight $w(e)$. We set $w(e) = |e|$ where $|e|$ denotes the cardinality of e so that all edges in the approximated graph will eventually have a uniform weight of one. In the star expansion algorithm, a graph $G^*(V^*, E^*)$ is constructed from hypergraph $G(V, E)$ by introducing a new vertex for every hyperedge $e \in E$, i.e., $V^* = V \cup E$. $E^* = \{(v, e) : v \in e, e \in E\}$ is the set of all edges that connect the keyframe vertices to their corresponding hyperedge vertex. Star expansion then assigns the scaled weight to all the edges of a hyperedge where $w^*(v, e) = w(e)/|e|$.

For reranking, visual-duplicate threads enlighten the relevance of a particular keyframes based on the size of the threads while story threads regularizes the random process so that the scores for keyframes within a story thread are more consistent. To achieve this goal, undirected edges are used for visual-duplicate threads so that weight transfers happen in both direction while directed edges are used for story threads where a story hyperedge vertex acts as a weight source and transfer weights to its corresponding keyframe vertices but not vice versa.

3.1 Reranking via Random Walk

For reranking, random walk on a graph can be viewed as a weight distribution process where the initial ranking scores $\mathbf{r}$ of all vertices are distributed through the links. The initial scores for the keyframe vertices follow the text-based ranking scores returned from the websites if available or a monotonically decreasing function such as a sigmoid function. For hyperedge vertices, they are set uniformly to any arbitrary value so that all hyperedges start on an equal footing when random walk is initiated. The amount of weight to be distributed through the links during a random process is governed by the transition matrix $\mathbf{P}^{N \times N}$ where $N = |V^*|$ and each element of the matrix $p_{ij} = w^*(i|j)$. Denoting the weight accumulated at the vertices during iteration t as $\mathbf{y}^{(t)}$, the random walk process can be formulated as follows:

$$\mathbf{y}^{(t+1)} = \alpha \mathbf{P}^T \mathbf{y}^{(t)} + (1 - \alpha)\mathbf{r} \quad (1)$$

where the process is iterated until it converges to the stationary distribution $\mathbf{y}_\pi$. α is a weighting factor which regulates how far the transition process is allowed to deviate from the original ranking score. Similar to the PageRank algorithm, for dangling vertices, out-going links are connected to all other vertices. The reranked score $r(v)$ for a keyframe vertex v is assigned based on its corresponding visual vertex e^* where $r(v) = y_\pi(e^*)$.

The relationship between a keyframe vertex v and its corresponding (visual-duplicate) hyperedge vertex e^* in the approximated graph structure is one-sided. Since the keyframe vertex v has only one link to the hyperedge vertex e^*, if e^* is heavily connected, v distributes its weight unconditionally to e^* but receive little from it. Clearly, a larger visual thread will eventually collect more weight especially when the initial scores of its neighboring keyframe vertices are high. In this respect, star expansion preserves the hyperedge information which makes possible the distinction between the scores of keyframes and hyperedges while in the clique expansion, such information is not readily available.

Finally, the reranked score of a video D are decided collectively by combining individual scores of all the keyframes by means of a noisy-or model where $r(D) = 1 - \prod_{v \in T}(1 - r(v))$. The noisy-or model assumes that the probability of keyframes being irrelevant are independent of each other. In practice, this will translate into the case where the score $r(D)$ is enhanced as more keyframes with good reranked scores becomes available, while keyframes with low scores do not impose much penalty.

4. EXPERIMENTS

In this section, we demonstrate the effectiveness of the proposed hyperlink structure with comparisons with two pairwise based graph structures. Evaluation is performed on the web video dataset from [11]. The data set is a video collection of 24 queries of the top favorite videos from YouTube in November 2006. Each query text was issued to YouTube, Google Video and Yahoo! Video and a total of 12790 videos are collected. The ground truth of videos relevant to the query were manually labelled by two assessors. Readers are referred to [11] for more details. Reranking is performed on the top 100 videos returned by the video sharing websites. The reranking performance is measured with MAP (mean average precision), a popular evaluation measure used in video search [9].

We compare three types of graph models for video reranking: story-level graph, keyframe-level graph and the proposed hypergraph model. The story-level and keyframe-level graph are constructed in a pair-wise manner using the near-duplicate detector in [14]. These results are also compared against the baseline which is the performance based on the original ranked lists returned by the video sharing websites using text queries. In the story-level graph, two videos are linked by an edge if any near-duplicate keyframes are detected. In the keyframe-level graph, keyframes are simply linked by an edge if they are near-duplicate. In hypergraph, the number of visual-duplicate threads per query ranges from 5 to 670, with an average of 57 threads. The number of keyframes in each thread ranges from 2 to 9920, with an average of 53 threads per query. Random walk on hypergraph in average takes 50 iteration to converge for each query, which completes in less than one second. The parameter α in Equation 2 is set to 0.8. This value is also used when performing random walk on story and keyframe-level graphs.

Table 1 shows the experimental results. All three models provide a steady improvement (34% for story and keyframe-level graphs and 45% for hypergraph) over the baseline under the random walk framework. Despite running at a finer granularity, the keyframe-level graph model does not display a noticeable improvement over the story-level graph model. In contrast, the hypergraph model manages to deliver a steady increase in reranking scores for most of the queries. For the keyframe-level graph, the keyframes belonging to the same thread are observed to be fragmented where in most scenarios it has links only to a small subset of the keyframes which in our hypergraph model are categorized under the same thread. This limits its effectiveness during random walk especially when there exist competing groups of visual-duplicates from other irrelevant topics.

Table 1: Reranking performance for baseline (BL), story-level graph (SG), keyframe-level graph (KG) and Hypergraph (HG).

ID	Query	BL	SG	KG	HG
1	The lion sleeps tonight	0.61	**0.99**	0.58	0.95
2	Evolution of Dance	0.49	0.91	**1.00**	0.95
3	Fold shirt	0.93	**1.00**	**1.00**	**1.00**
4	Cat massage	0.63	**1.00**	**1.00**	**1.00**
5	OK go here it goes again	0.65	0.99	0.97	**0.99**
6	Urban Ninja	0.25	0.44	0.46	**0.45**
7	Real life Simpsoms	0.99	**0.99**	0.99	0.99
8	Free hugs	0.29	**0.47**	0.37	0.45
9	Where the hell is Mat	0.25	0.32	0.40	**0.44**
10	U2 and green day	0.61	0.78	0.85	**0.85**
11	Little superstar	0.67	0.74	**0.82**	0.74
12	Napolean dynamite dance	0.53	0.85	0.92	**0.99**
13	I will survive Jesus	0.99	0.99	0.99	**1.00**
14	Ronaldinho ping pong	0.82	0.88	**0.93**	0.93
15	White and Nerdy	0.79	0.92	0.92	**0.98**
16	Korean karaoke	0.17	0.17	0.17	**0.22**
17	Panic at the disco I write sins not tragedies	0.73	0.80	0.80	**0.96**
18	Bus uncle	0.44	0.85	**0.85**	0.82
19	Sony Bravia	0.56	0.67	0.68	**0.79**
20	Changes Tupac	0.76	0.76	0.79	**0.88**
21	Afternoon delight	0.25	0.32	0.32	**0.99**
22	Numa Gary	0.53	0.89	**0.90**	0.87
23	Shakira hips don't lie	0.53	0.94	0.98	**0.99**
24	India driving	0.43	0.56	0.55	**0.59**
MAP		0.57	0.76	0.76	**0.83**

5. CONCLUSIONS

We have presented the idea of using hypergraph to model the video hyperlinks for reranking based on random walk algorithm. Experimental results on web videos demonstrate that hypergraph, which is capable of modeling higher-order relation existent among video content, outperforms existing graph-based models built upon by pair-wise story or keyframe linking. The proposed work successfully address the problem of missing and false links which was not properly treated by other approaches. Currently, we only consider visual duplicates for video hyperlinking. An interesting direction to further explore is the use of other modalities such as semantic concepts to jointly model the higher-order relation among videos using hypergraph.

6. ACKNOWLEDGEMENTS

The work described in this paper was fully supported by a grant from City University of Hong Kong (Project No. 7002112) and a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (CityU 119508).

7. REFERENCES

[1] S. Agarwal, K. Branson, and S. Belongie. Higher order learning with graphs. In *ICML*, 2006.

[2] S. Agarwal, J. Lim, L. Zelnik-Manor, P. Perona, D. Kriegman, and S. Belongie. Beyond pairwise clustering. In *CVPR*, 2005.

[3] T.-S. Chua and et. al. Trecvid 2004 search and feature extraction task by NUS PRIS. In *TRECVID Workshop*, 2004.

[4] W. H. Hsu, L. S. Kennedy, and S.-F. Chang. Video search reranking via information bottleneck principle. In *ACM Multimedia*, 2006.

[5] W. H. Hsu, L. S. Kennedy, and S.-F. Chang. Video search reranking through random walk over document-level context graph. In *ACM Multimedia*, 2007.

[6] L. S. Kennedy and S. F. Chang. A reranking approach for context-based concept fusion in video indexing and retrieval. In *CIVR*, 2007.

[7] J. Liu, W. Lai, X. S. Hua, Y. Huang, and S. Li. Video search re-ranking via multi-graph propogation. In *ACM Multimedia*, 2007.

[8] L. Page, S. Brin, L. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[9] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and TRECVID. In *MIR*, 2006.

[10] Wikipedia. http://en.wikipedia.org/wiki/youtube.

[11] X. Wu, A. G. Hauptmann, and C.-W. Ngo. Practical elimination of near-duplicates from web video search. In *ACM Multimedia*, 2007.

[12] R. Yan and A. G. Hauptmann. Multimedia search with pseudo-relevance feedback. In *CIVR*, 2003.

[13] Y. H. Yang and W. H. Hsu. Video search reranking via online ordinal reranking. In *ICME*, 2008.

[14] W. L. Zhao, C.-W. Ngo, H.-K. Tan, and X. Wu. Near-duplicate keyframe identification with interest point matching and pattern learning. In *IEEE Trans. on Multimedia*, volume 9, pages 1037–1048, Aug 2006.

[15] J. Y. Zien, M. D. F. Schlag, and P. K. Chan. Multi-level spectral hypergraph partitioning with arbitrary vertex sizes. In *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, volume 18, pages 1389–1399, 1999.