Searching the Web Using Composed Pages

Ramakrishna Varadarajan Vagelis Hristidis Tao Li
Florida International University
{ramakrishna,vagelis,taoli}@cis.fiu.edu

Categories and Subject Descriptors:
H.3.3 [**Information Search and Retrieval**]

General Terms: Algorithms, Performance

Keywords: Composed pages

1. INTRODUCTION

Given a user keyword query, current Web search engines return a list of pages ranked by their "goodness" with respect to the query. However, this technique misses results whose contents are distributed across multiple physical pages and are connected via hyperlinks and frames [3]. That is, it is often the case that no single page contains all query keywords. Li et al. [3] make a first step towards this problem by returning a tree of hyperlinked pages that collectively contain all query keywords. The limitation of this approach is that it operates at the page-level granularity, which ignores the specific context where the keywords are found within the pages. More importantly, it is cumbersome for the user to locate the most desirable tree of pages due to the amount of data in each page tree and a large number of page trees.

We propose a technique called *composed pages* that given a keyword query, generates new pages containing all query keywords on-the-fly. We view a web page as a set of interconnected text fragments. The composed pages are generated by stitching together appropriate fragments from hyperlinked Web pages, and retain links to the original Web pages. To rank the composed pages we consider both the hyperlink structure of the original pages, as well as the associations between the fragments within each page. In addition, we propose heuristic algorithms to efficiently generate the top composed pages. Experiments are conducted to empirically evaluate the effectiveness of the proposed algorithms. In summary, our contributions are listed as follows: (i) we introduce composed pages to improve the quality of search; composed pages are designed in a way that they can be viewed as a regular page but also describe the structure of the original pages and have links back to them, (ii) we rank the composed pages based on both the hyperlink structure of the original pages, and the associations between the text fragments within each page, and (iii) we propose efficient heuristic algorithms to compute top composed pages using the uniformity factor. The effectiveness of these algorithms is shown and evaluated experimentally.

SIGIR'06, August 6–11, 2006, Seattle, Washington, USA.
ACM 1-59593-369-7/06/0008.

2. FRAMEWORK

Let $D=\{d_1,d_2,,\ldots,d_n\}$ be a set of web pages $d_1,d_2,,\ldots,d_n$. Also let $size(d_i)$ be the length of d_i in number of words. *Term frequency* $tf(d,w)$ of term (word) w in a web page d is the number of occurrences of w in d. *Inverse document frequency* $idf(w)$ is the inverse of the number of web pages containing term w in them. The *web graph* $G_W(V_W,E_W)$ of a set of web pages $d_1,d_2,,\ldots,d_n$ is defined as follows: *A* node $v_i \in V_W$, is created for each web page d_i in D. An edge $e(v_i,v_j) \in E_W$ is added between nodes $v_i,v_j \in V_W$ if there is a hyperlink between v_i and v_j. Figure 1 shows a web graph. The hyperlinks between pages are depicted in the web graph as edges. The nodes in the graph represent the web pages and inside those nodes, the text fragments, into which that web page has been split up using html tag parsing, are displayed (see [5]).

In contrast to previous works on web search [3,4], we go beyond the page granularity. To do so, we view each page as a set of *text fragments* connected through semantic associations. The *page graph* $G_D(V_D,E_D)$ of a web page d is defined as follows: (a) d is split to a set of non-overlapping text fragments $t(v)$, each corresponding to a node $v \in V_D$. (b) An edge $e(u,v) \in E_D$ is added between nodes $u,v \in V_D$ if there is an association between $t(u)$ and $t(v)$ in d. Figure 2 shows the page graph for Page 1 of Figure 1. As denoted in Figure 1, page 1 is split into 7 text fragments and each one is represented by a node. An edge between two nodes denotes semantic associations. Higher weights denote greater association. In this work nodes and edges of the page graph are assigned weights using both query-dependent and independent factors (see [5]). The semantic association between the nodes is used to compute the edge weights (query-independent) while the relevance of a node to the query is used to define the node weight (query-dependent).

A *keyword query* Q is a set of keywords $Q=\{w_1,\ldots,w_m\}$. A *search result* of a keyword query is a subtree of the web graph, consisting of pages $d_1,\ldots,d_l$, where a subtree s_i of the page graph G_{Di} of d_i is associated with each d_i. A result is total–all query keywords are contained in the text fragments–and minimal–by removing any text fragment a query keyword is missed. For example, Table 1 shows the Top-3 search results for the query "Graduate Research Scholarships" on the Web graph of Figure 1.

3. RANKING SEARCH RESULTS

Problem 1 (Find Top-k Search Results). *Given a web graph G_W, the page graphs G_D for all pages in G_W, and a keyword query Q, find the k search results R with maximum Score(R).*

The computation of *Score(R)* is based on the following principles. First, search results R involving fewer pages are ranked higher [3]. Second, the scores of the subtrees of

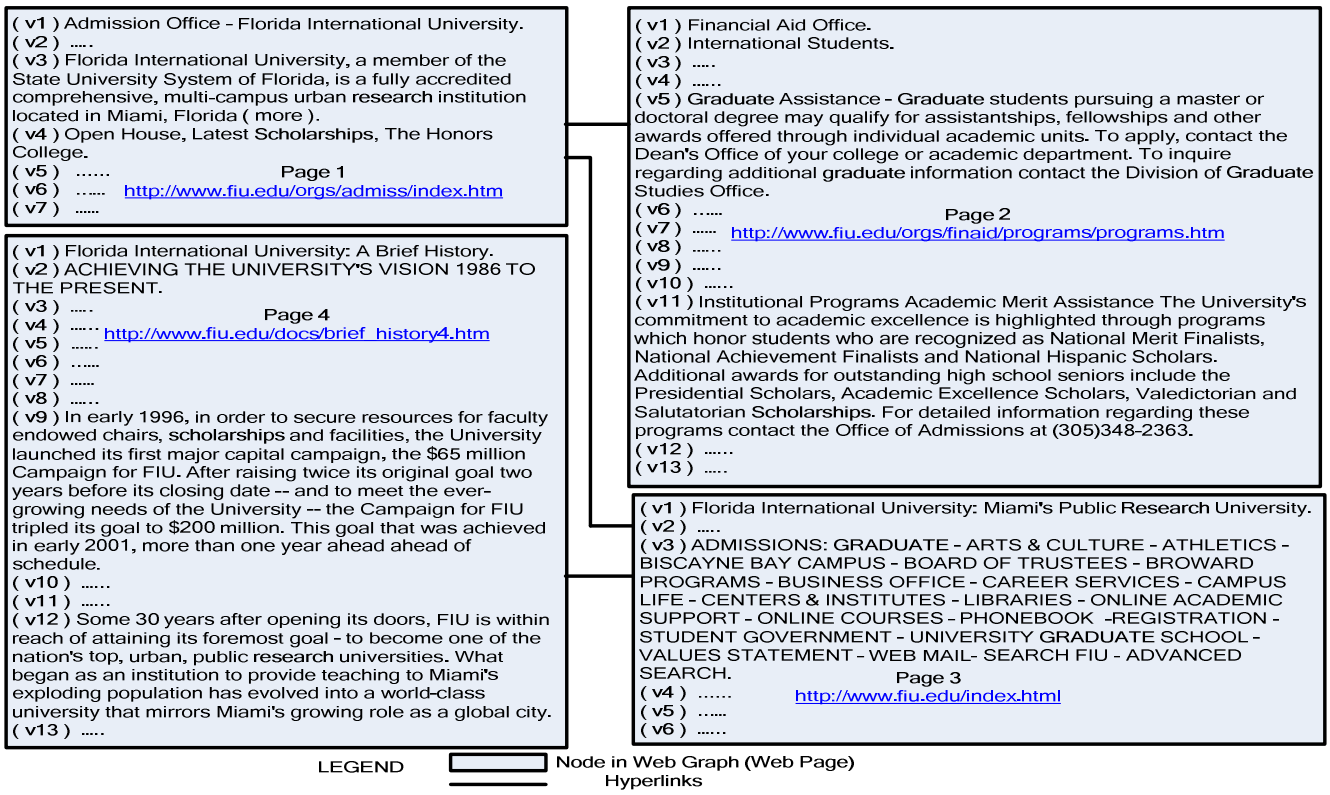


Figure 1: Sample web pages from www.fiu.edu.

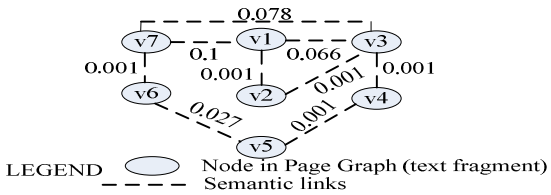
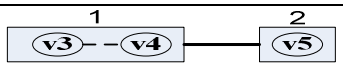
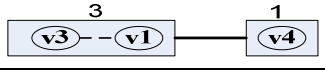


Figure 2: A page graph of Page 1 of Figure 1.

Table 1: Top-3 search results for query "Graduate Research Scholarships"

Rank	Score	Search Results
1	12.50	
2	33.58	
3	101.60	

the page graphs of the constituting pages of *R* are combined using a monotonic aggregate function to compute the score of the search result. A modification of the expanding search algorithm of [1] is used where a heuristic value combining the Information Retrieval (IR) score, the PageRank score [4], and the inverse of the *uniformity factor (uf)* of a page is used to determine the next expansion page. The *uf* is high for pages that focus on a single or few topics and low for pages with many topics. The *uf* is computed using the edge weights of the page graph of a page (high average edge weights imply high *uf*). The intuition behind expanding according to the inverse *uf* is that among pages with similar IR scores, pages with low *uf* are more likely to contain a short focused text fragment relevant to the query keywords. Figure 3 shows the quality of the results of our heuristic search vs. the quality of the results of the non-heuristic expanding search [1] (a random page is chosen for expansion since hyperlinks are un-weighted) compared to the optimal exhaustive search. The modified Spearman's rho metric [2] is used to compare two Top-*k* lists.

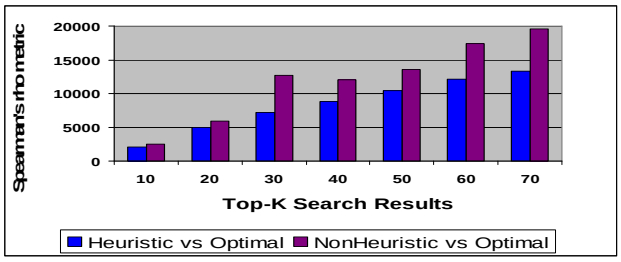


Figure 3: Quality Experiments using Spearman's rho.

4. REFERENCES

[1] G. Bhalotia, C. Nakhe, A. Hulgeri, S. Chakrabarti and S,Sudarshan: Keyword Searching and Browsing in Databases using BANKS. ICDE, 2002

[2] Ronald Fagin, Ravi Kumar, and D. Sivakumar: Comparing top-k lists. SODA, 2003

[3] W.S. Li, K. S. Candan, Q. Vu and D. Agrawal: Retrieving and Organizing Web Pages by "Information Unit". WWW, 2001

[4] L. Page, S. Brin, R. Motwani, and T. Winograd: The pagerank citation ranking: Bringing order to the web. Technical report, Stanford University, 1998

[5] R. Varadarajan, V Hristidis: Structure-Based Query-Specific Document Summarization. CIKM, 2005