

Continuous Top-k Monitoring on Document Streams

ABSTRACT

In information filtering the objective is to remove from an information stream those items that are of no interest to the end users. The actual filtering involves fixed thresholds rather than relative similarity and ranking.

EXISTING SYSTEM

In traditional text search, there are snapshot (i.e., one off) top-k queries over static document collections. The inverted file is the standard index to organize documents. It comprises a list for every term in the dictionary; the list for a term holds an entry for each document that contains the term. By sorting the lists in decreasing term frequency, and with appropriate use of thresholding, a snapshot query can be answered by processing only the top parts of the relevant lists. The task of the stream server is to update all query results as new documents arrive. Document arrivals are referred to as stream events. The primary performance metric in our work is the time required to refresh (update) all results in response to stream events.

DRAWBACKS

- Response time is more
- For updating it takes more time

PROPOSED SYSTEM

In this, user preferences are indicated by a set of keywords. A central server monitors the document stream and continuously reports to each user the top-k documents that are most relevant to her keywords. Our objective is to support large numbers of users and high stream rates, while refreshing the top-k results almost instantaneously. Our solution abandons the traditional frequency-ordered indexing approach. Instead, it follows an identifier-ordering paradigm that suits better the nature of the problem. When complemented with a novel, locally adaptive technique, our method offers (i) proven optimality w.r.t. the number of considered

queries per stream event, and (ii) an order of magnitude shorter response time (i.e., time to refresh the query results) than the current state-of-the-art.

ADVANTAGES

- Response time is less
- For updating it takes less time

MODULES

- Id-ordering techniques
- Minimal reverse id-ordering

SYSTEM REQUIREMENTS

H/W System Configuration:-

Processor	-	Pentium –III
RAM	-	256 MB (min)
Hard Disk	-	20 GB
Key Board	-	Standard Windows Keyboard
Mouse	-	Two or Three Button Mouse
Monitor	-	SVGA

S/W System Configuration:-

Operating System	:	Windows95/98/2000/XP
Application Server	:	Tomcat5.0/6.X
Front End	:	HTML, Jsp
Scripts	:	JavaScript.
Server side Script	:	Java Server Pages.

Database : MySQL 5.0

Database Connectivity : JDBC

www.takeoffprojects.com