ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 13, March 2017

# Structured Indexing Strategy for Real Microblog Search

Y.Roja Begam[1], A.Vegi Fernando[2]
ME Student[1], Assistant Professor[2],
SCAD College of Engineering and Technology,
Chenranmahadevi, Tirunelveli.

***Abstract:*** *Indexing micro-blogs for real-time search is difficult, as a result of new micro-blogs are created at tremendous speed, and user question requests keep perpetually dynamical. To ensure user acquire complete question results, micro-blogging web site maintains vast indices that ends up in index fragmentation or additional merging overhead throughout real time search. This paper proposes Structured Index Merging Strategy for real-time search on micro-blogs. This structure consists of associate degree inverted index buffer and a sequence of dynamically adjustable index packages with exponentially increasing sizes. These index packages manage their inverted indices discrimination reconciling merging strategy, which mightscale back the merging overhead to boost question performance and may alter the index structure supported environmental factors, like the arrival rate of question requests and new micro-blogs. Experimental results show that proposed structure will greatly improve question performance while not increasing the update value and improve the self-adaptability in dynamic atmosphere.*

***Keywords:*** Micro-blogs, Indexing, real-time search.

## I Introduction

Social media such as Twitter is an interesting source of information [1]. micro-blogs are typically short, but there is a large volume of them. This provides a challenge when a system is needed to obtain meaningful results from a set of micro-blogs in near real-time [2]. This paper compares three existing, open-source search engines that are capable of full-text keyword search and a distributed setup. Dealing with big data in computational social networks may require big machines with big storage.

Microblogging is a broadcast medium that exists in the form of blogging [3]. A microblog differs from a traditional blog in that its content is typically smaller in both actual and aggregated file size. Micro-blogs"allow users to exchange small elements of content such as short sentences, individual images, or video links", which may be the major reason for their popularity. These small messages are sometimes called microposts [4].

It propose a novel efficient index structure, for real-time search on micro-blogs. This structure consists of an inverted index buffer and a sequence of dynamically adjustable index packages with exponentially increasing sizes. When query resources are insufficient, the proposed structure merges the inverted indices into larger indices in batch to improve query efficiency. When query resources are abundant, The proposed structure divides the inverted indices into smaller ones to accelerate query progress.

The remainder of this paper is organized as follows: section II presents a related work for the proposed system, section III explains system analysis of proposed work and modules of the proposed system, section IV describes the simulation results and finally conclusion and future work.

## II Related Work

Apache Lucene [5] is a library for a high-performance, full-featured text search engine, which performs indexing and searching using small RAM, and generates an index file with reasonable size. It customized the Lucene core and built a data retrieval system that solved the problem of memory blowup. Our system consisted of two parts: (1) a real-time index server and (2) a search server. The real-time index server refreshes the database frequently and indexes dynamically the newly collected MBPs on the fly. The search server returns the MBPs that contain all the keywords entered by the user. These two parts may work simultaneously without conflicting each other, and so the search server can return real-time posts the system collects.

The nextword index consists of a vocabulary of distinct words and, for each word *w* in the vocabulary, a nextword list and the position list. The nextword list consists of each word *s* that succeeds *w* in an MBP in the database. For each triple of *(w, s,M)*, where *M* is an MBP that contains the phrase *ws*, the position list consists of the list of positions where the phrase *ws*appears in theMBP*M*and a pointer that links *(w, s)* to *M*. We also store in the position list the necessary information of each statistical feature. This structure will be used in the statistical analysis component of the system. [6] discussed about a system, In this proposal, a neural network approach is proposed for energy conservation routing in a wireless sensor network. Our designed neural network system has been successfully applied to our scheme of energy conservation.

Provenance discovery [7] is an important technique to derive the source and transformation from large amounts of data. Provenance information describes the origin and the development of data in their life cycles. It has been demonstrated useful in many domains, such as business workflow, scientific processing and database query analysis. For example, information about provenance can serve as the basis of data results correctness and in turn, determine the quality of the final outputs. Here we introduce a provenance model over microblog messages to capture the messages' development. The development can be captured by means of temporal grouping and alignment of related messages, i.e., a kind of connection discovery to explore the temporal context of messages. Here, related messages are extracted and organized in an ordered structure. The latter incoming messages are connected with previous related ones.

## III Proposed work

The mainstream real-time indexing system on micro-blogs basically includes two modules: Index module and Query module. Index module accepts new arrival micro-blogs and inserts them into the index structure. It also maintains a huge custom-designed index structure. Query module responds to user's query request and returns search results with a query thread pool. With these two modules, new micro-blogs are indexed into the index structure. When a user submits a query request, the query module retrieves the index structure, finds related micro-blogs, and returns ranked micro-blogs to user. The modules are

- Storage Index Structure
- Creation and Extension of Index
- Structure adjustment
- Merging strategy

### *Storage Index Structure*

Since index is frequently read and write, improve the efficiency of read and write operations can effectively improve the query performance. The read and write operations on index are closely related to the inverted lists, that how to store inverted lists is essential to the updating performance on query.

### *Creation and Extension of Index*

Whenever a new microblog m comes, it is appended to the posting lists in po. For instance, let (item1, item2, docid5) denotes the microblog doc been indexed, where item 1 and item2 denote the IDs of the terms in doc, and docID10 denotes the document id of doc. The proposed system starts anIndexWriter thread to find the posting lists of item 1and item 2 in p0, and insert docid5 to these two posting lists. Considering that the fresh micro-blogs need to be ranked early, docid5 is inserted into the front of the posting list, rather than in the back.

### *Structure Adjustment*

The proposed structure can adjust itself by changing the variable s, which controls the proportion of the index packages adopting different merging strategies. When s changes, the proposed structure starts the self-adjusting process by increasing or decreasing the variable s.

### *Merging Strategy*

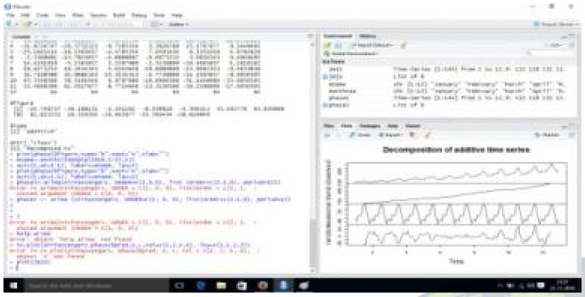
To achieve better query performance under different query loads, The proposed structure dynamically adjusts s with two goals:

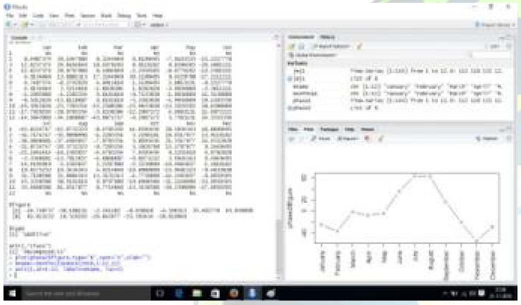
- Minimizing the query time
- Increase the query rate.

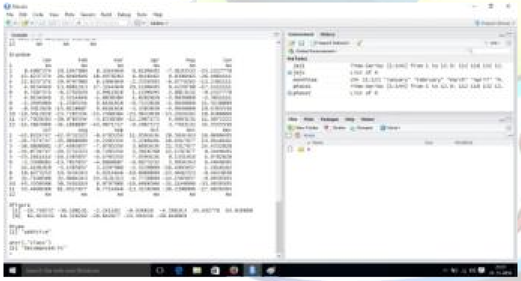
These two goals can conflict with each other, because allocating more resources to a query reduces search time, but may result in longer queuing delays and affect query throughput.

## V Simulation Results

The proposed algorithm is implemented using R-language. The requested queries are stored in the database and the queries are arranged based on the index of the items received from the users. The monthly reports are taken for the data are stored in the structured format in the database. The results are shown in the following figures.







Conclusion

In this paper, indexing structure for micro-blogsare proposed. In this algorithm, a new indexing and ranking scheme for supporting real-time search in microblogging systems is introduced. It adopts the adaptive indexing scheme to reduce the cost for updating query. It uses the batch indexing scheme for reducing the indexing latency.The results shows that the proposed algorithm efficiently index the items stored in the database.

Reference

[1] M. D. Choudhury, Y.-R. Lin, H. Sundaram, K. S. Candan, L. Xie, and A. Kelliher. How does the sampling strategy impact the discovery of information diffusion in social media? In *ICWSM*, 2010.

[2] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes twitter users: real-time event detection by social sensors. In *WWW*, pages 851–860, 2010.

[3] J. Weng, E.-P.Lim, J. Jiang, and Q. He.Twitterrank: finding topic-sensitive influential twitterers. In *WSDM*, pages 261–270, 2010.

[4] A. Sun, M. Hu, and E.-P. Lim. Searching blogs and news: a study on popular queries. In *SIGIR*, pages 729–730, 2008.

[5] The Apache Software Foundation: Apache Lucene (2/1/2015). https://lucene.apache.org/

[6] Christo Ananth, A.Nasrin Banu, M.Manju, S.Nilofer, S.Mageshwari, A.Peratchi Selvi, "Efficient Energy Management Routing in WSN", International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE), Volume 1, Issue 1, August 2015,pp:16-19

[7]Junjie Yao Bin Cui ZijunXueQingyun Liu, "Provenance-based Indexing Support in Micro-blog Platforms", Big Data Analytics, 2015.