International Journal of Technical Innovation in Modern Engineering & Science (IJTIMES)
Impact Factor: 5.22 (SJIF-2017), e-ISSN: 2455-2585
Volume 4, Issue 11, November-2018

KEEP TRACK OF YOUR TOP K ON THE STREAM

Gurram Suma[1] , K.Srinivas Babu[2]

[1]*Department of Computer Science and Engineering,Nalla Narsimha Reddy Group of Institutions,*
[2] *Department of Computer Science and Engineering, Nalla Narsimha Reddy Group of Institutions*

Abstract— Many data is an efficient action of the flow of report for solo structures. To update the news chapter and informal community ads, for example, the increased application will require users to have the most important aspect of their priorities. In this business, customer trends are shown by establishing phrases. Focus on server log flow and constantly report on the best collections related to each of your clients on the verification icon. Our goal is to help customers with large volumes and high flow rates with best results. Our response will respond to the traditional request request. Instead, the world perspective of the concept of this subject. At this time a novel, a versatile system locally, our modern technology (i) optimum w.r.t. When it's supplemented with. Each section study questions a volume, and (2) on a smaller basis (ie question results during the activation of the best time in the classroom).

Keywords— Continuous Query, top-k queries

I. INTRODUCTION

During larger data, more than the ability to detect and understand the size of information provided by users. For example, if a notification volume can be obtained, a user will be sent to this message on Twitter in short messages. In addition, information Filter dimensions and distributions are very important. For example, users get social and entertainment websites and articles (for example, reddit.com) for immediate updates. In this way, the hot key is important for monitoring effective applications and many applications. Looking at recent documents (CTQD) documents, it is related to the recent interest. In this case, the central server maintains the document flow and operates CTQs from different users. Each CTQD is defined by specific behaviour by the user or the Internet by specific behaviour. With continuous updates of the most relevant keywords above each training document, along with new documents and old discussions working on the server. Stock News Notifications are an application area for CTQD. Investment decisions are most sensitive to the shares in their portfolio. Make timely decisions to provide important news for immediate success and notification system. Such applications are available for monitoring live web content such as RSS / news feeds, blog entries, social media etc. Google.com/alerts and yahoo! Alerts are widely available notification systems (yahoo.com alerts), which have the importance of these applications. On the other hand, these systems provide updates from partial (for example, daily) or thick filtration (usually) themes rather than specific key sets). Another application for CTQD domain real-time search services. Currently, this service allows users to query the question (in a demand, one way) to meet a keyword. CTQD can extend The Service Function is constantly updated around new surveys that provide continuous monitoring / notifications. If you're looking for traditional lessons, or snapshots (like A), there are other questions about collecting hard documents. There is a standard index for maintaining bad file files. Every word related to this word has a list. Every word in the word related to this word has an entry. With the proper use of string shading, a snapshot question is answered, referring to the top sections in the relevant list.

II. RELATED WORK

Effective processing of document streams plays an important role in many filtration systems. Emerging apps such as news update filters and social network notifications demand the end-users with the most relevant content for their preferences. In this work, user preferences are represented by keywords. A central server oversees the transmission of the document and constantly reports the most relevant top-level documents for each user to its keywords. Our goal is to support a large number of users and high stream rates. If the top-k is readily refreshed. Our solution will terminate the traditional frequency-sorted indexing process. Instead, it follows an identifier-order paramecium that best suits the nature of the problem. When connected to a novel, a local adaptive technique, our method provides (i) proven optimality w.r.t. The number of observations pertaining to the current event, and (ii) the amount of response times less than the current state-of-art (ie, time to refine the results of the question). Due to the events stream, we need to investigate streaming papers, relatively stable questions, and against this index to remove the need for index management. Generally, the general idea of questioning the data on the streaming context and the subject is usually called question indexing and is used for a variety of persistent questions. However, the order of an order for an index is not common and as defined in Section 4.2, there is a need to carefully control the internal tasks. The RIO is faster than the current CTQD approach, but we have not stopped there. Thirdly, we will complete the RIO with a new, locally capable technique that produces a strong processing threshold. This technique is maximally CTQD method w.r.t. It gives us the number of questions per day, which will be counted as proof that this is a scheduling document. The minimum number of questions for any algorithms is guaranteed by ID-order modelling and authenticity. The resulting method is our maximum and more

advanced technology, at least known as MRIO. We demonstrate that by making extensive experimental diagnosis with real documents, MRIO intensifies the ARRD solution of the government. Furthermore, the "internal" comparison between MRIO and Rio shows that they are mainly on top sizes, improving the mass efficiency of third-generation compatible techniques above.

III. IMPLEMENTATION

A. Query Indexing Module:

In this module, the index can index the user's query to remove the question management query from the stream events.

B. ID-Ordering:

In this module, the documentation refers to the wrong file that contains every wrong word. This document contains an entry with document identification and weight reduction, which contains the word for each document. All documents are configured according to document identification.

C. Minimal Reverse ID-Ordering:

In this module we are reducing the number of iterations to retrieve relevant and top-k continuous documents to improve the performance of the proposed system.

Test Cases

TABLE I

Test Case Id	Test Case Name	Test Case Desc.	Test Steps			Test Case Status	Test Priority
			Step	Expected	Actual		
01	Upload document stream	Verify either document stream is uploaded or not	If document stream is not uploaded	Document is not streaming	Document is stream	High	High
02	Register	Verify either user is registered or not	If his not registered	He cannot authorizatio n person	Registration completed	High	High
03	login	Whether the user is login or not	If his not login	he cannot do the process	Login completed	High	High
04	Update preferences	Whether the keyword preferences is updated or not	If it's not updated	It cannot preferences the keyword	User preference details added	High	High
05	Download	Verify the file is down loaded or not	If it's not downloaded	user cannot download the file	Download completed	High	High
06	Iteration comparison graph	Verify the between RIO&MRIO iterations comparison or not	If it's not comparison	It cannot generate the graph	It can generate the graph	High	High
07	Query processing time graph	Whether the query processing time graph is generated or not	If it's not generate the chart	It cannot display the time chart	It can display the time chart	High	High

Analysis Graphs

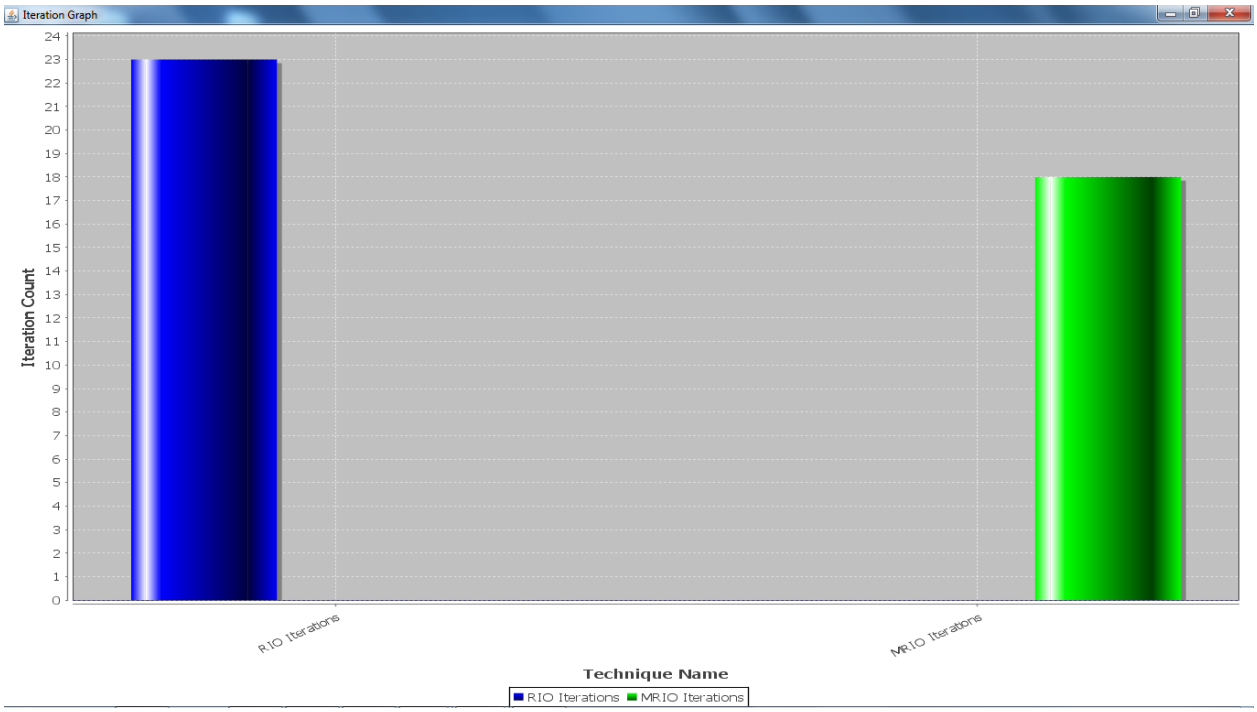

Fig. 2 A Sample Bar Chart for **RIO and MRIO iterations graph comparison**

A. This is a **RIO and MRIO iterations graph comparison which tell about the top k algorithm**

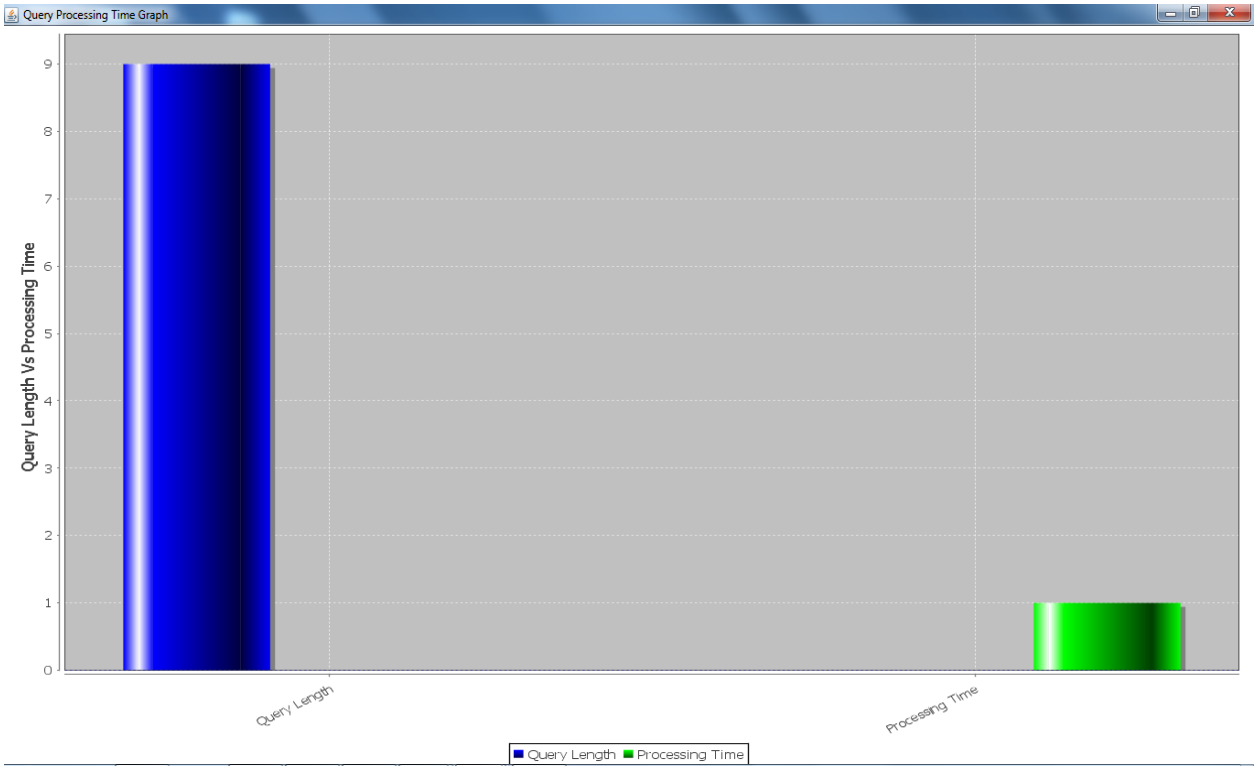

Fig. 2 A Sample Bar Chart for **Query processing time graph**

b. This is **Query processing time graph** of the sample data where it can be managed by the performance management

IV. CONCLUSIONS

CTQD refers to the most suitable documents for a key word. Our initial approach is applied to the previous order system identifier. Number of compilers by analysing their performance in determining Rio Factor This advanced approach encourages us to reduce the MRIO number, is only re-grouped, but shown to reduce. We will get this novel by displaying the boundaries of local communication. There is clear guidance for expanding our methods to assess important questions for future work

REFERENCES

[1] P. Haghani, S. Michel, and K. Aberer, "The gist of everything new: personalized top-k processing over web 2.0 streams." in CIKM, 2010, pp. 489–498.

[2] K. Mouratidis and H. Pang, "Efficient evaluation of continuous text search queries," IEEE Trans. Knowl. Data Eng., vol. 23, no. 10, pp. 1469–1482, 2011.

[3] N. Vouzoukidou, B. Amann, and V. Christophides, "Processing continuous text queries featuring non-homogeneous scoring functions." in CIKM, 2012, pp. 1065–1074.

[4] A. Hoppe, "Automatic ontology-based user profile learning from heterogeneous web resources in a big data context." PVLDB, pp. 1428–1433, 2013.

[5] A. Lacerda and N. Ziviani, "Building user profiles to improve user experience in recommender systems," in WSDM, 2013, pp. 759–764.

[6] M. Busch, K. Gade, B. Larson, P. Lok, S. Luckenbill, and J. J. Lin, "Earlybird: Real-time search at twitter," in ICDE, 2012, pp. 1360– 1369.

[7] L. Wu, W. Lin, X. Xiao, and Y. Xu, "LSII: an indexing structure for exact real-time search on microblogs," in ICDE, 2013, pp. 482–493.

[8] J. Zobel and A. Moffat, "Inverted files for text search engines," ACM Comput. Surv., vol. 38, no. 2, 2006.

[9] R. Fagin, A. Lotem, and M. Naor, "Optimal aggregation algorithms for middleware," J. Comput. Syst. Sci., vol. 66, no. 4, pp. 614–656, 2003.

[10] A. Z. Broder, D. Carmel, M. Herscovici, A. Soffer, and J. Y. Zien, "Efficient query evaluation using a two-level retrieval process." in CIKM, 2003, pp. 426–434.