International Journal of Engineering & Technology, 7 (2.21) (2018) 423-426

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Web image re-ranking using query specific in cloud computing

U.V. Anbazhagu[1*], R. Balakrishna[2], A. Sajeev Ram[3], M. Latha[4]

[1]*Department of Computer Science & Engineering, Vels Institute of Science, Technology and Advanced Studies (VISTAS), Chennai, India.*
[2]*Department of Computer Science & Engineering, Vels Institute of Science, Technology and Advanced Studies (VISTAS), Chennai, India.*
[3]*Department of Computer Science & Engineering, Vels Institute of Science, Technology and Advanced Studies (VISTAS), Chennai, India.*
[4]*Department of Computer Science & Engineering, Vels Institute of Science, Technology and Advanced Studies (VISTAS), Chennai, India.*
**Corresponding author E-mail: anbuveera@gmail.com*

Abstract

Question answering (QA) allows all users to get information in enhanced technique. In this project we suggest a system for inspiring textual answer with appropriate media data. Our system consists of three components Interpretation median picking, Inquiry propagation, Data pick and Launching. Interpretation median picking is used to select various types of answers. Inquiry propagation is used for extracting the root words from the given query. Data pick and Launching is used for selecting the appropriate answer and producing the result. We use Stemming algorithm, Naïve Bayes classifier algorithm and page ranking algorithms. Stemming algorithm is used to extract the root word from the given searched query. Naïve Bayes classifier algorithm is used for selecting the type of medium. By using the page ranking algorithm the optimal solution is got. Our approach automatically determines which media will be a best solution for the given query. It automatically harvests the data from website for getting the answer. Our approach can enable a novel multimedia question answering (MMQA) approach as users can find multimedia answers by matching their questions with those in the pool. We are enhancing community contributed answers. Any user who is unaware of data can get the information promptly. Our approach is to deal with the complex questions in an effective way. Based on the generated queries, we vertically collect image and video data with multimedia search engines.

Keywords: *Interpretation, ranking, queries, stemming, map reduce, range-aggregate.*

1. Introduction

Big data is defined as the breaking of huge quantity of the data into minor parts for enhanced understanding. Big data is produced since every individual is using the commercial, internet and social access, Business sites and educational sites for accepting. Every individual likes to connect to his/her friends, colleagues, family by the means of internet. Now a day's knowledge of technology and new updates are done with the help of internet that is creating life more attractive and simple. Big data is developed in all the things and here the logic is performed that another way to earn money in business.

Current work is implemented to detect the Black money using Hadoop technology in the big data environment. In this project the user details from various banks will be collected and it will be stored in the database. Whenever the users do any transaction, the details will be automatically saved. User details from various banks will be gathered and the user who used the same ID proof can be easily identified so that the Income tax of each user can be tracked easily. The goal of our work focuses on to reduce the black money by tracking the user details from various bank databases. Whenever the users do any transaction, the details will be automatically saved. User details from various banks will be gathered and the user who used the same ID proof can be easily identified so that the Income tax of each user can be tracked easily. Thus the information can be given to the Income tax department.

In the existing work, for the given query range an Range-aggregate queries applied with an aggregate function on all tuples. Existing approaches to range-aggregate queries are insufficient to quickly provide accurate results in big data environments.

The following are the demerits of the existing work,

- Congestion occurring.
- Less accuracy.
- Low data transmission rate.
- Replicate request.
- Unreliable.
- Waiting time is increased.

In the current system, Fast RAQ first divides big data into independent partitions with a balanced partitioning algorithm, and then generates a local estimation sketch for each partition. When a range-aggregate query request arrives, Fast RAQ obtains the result directly by summarizing local estimates from all partitions & collective results are provided

The following are the merits of the current work,

- Accuracy is improved.
- Avoid Congestion.
- Avoid replicate request.
- Data transmission rate is increased.
- Less time consumption and Reliable.

Fig. 1 illustrates the system architecture of the current work.

In the system architecture diagram, the overall process is explained.

Three different bank details are collected and those details are further splitted by using balance partitioning algorithm. Then by using the fast range aggregate queries the condition will be checked (any user> 50,000 Income more than 3 banks). If the

condition is satisfied, automatically the user details can be intimated to the Income tax department.

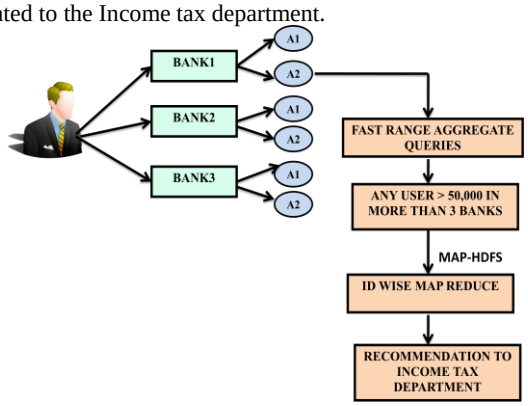

Fig. 1: Architecture diagram

2. Materials and methods

2.1. User profile and account registration

User account creation is the first step of the current work where the user will create an account for accessing the network and also the services provided by the service provider. The entire request given by the user are processed by the provider and given suitable response to them and everything stored in an unified database. In the current work, an interface as been developed for requesting services and accessing the network.

2.2. Bank service provider

User's information is maintained by bank service providers in their respective data repositories. That information is verified during the user's authentication. User interface frame as been developed to create a connection between user and other modules of the company server for communication .

2.3. Cloud setup

A large data repository has been maintained by cloud service provider for storing data along with that they maintain user information for authentication. The cloud service provider will transfer the request to the appropriate request handling module for processing and giving response to the user. Interface has been developed to establish connection between modules and the request are processed based upon FIFO (First In First Out) manner.

2.4. Big data setup

The term big data refers to collection of large volume of data which are complex to proceed using conventional information manipulating applications. The maintenance of large volume of data like analysis, duration, capture, storage, visualization, etc. To solve this issue big data has been introduced because a set of instructed information has been given to each and every employ and we need to analysis on those data's.

2.5. Black money detection using map reduce

In the current module, we need to give input for the Map Reduce technique and the output of mapping part will be given as the input for Map Reduce. The input of this module is the user's having more than three bank accounts and this module will track the users who are all depositing more than fifty thousands in all the three bank accounts annually. The output of this module are the above users and they will be monitored by the income tax department.

2.6 Account transaction review

In this module we get the information about the users who gave more than three accounts in the bank and we also filter the transaction done by the user and we review the information transacted by the user to their user through their online or manual Two techniques are used. They are,

- Partition Algorithm
- Fast Range Aggregate Queries

Partition algorithm

The concept of partitioning is to allocate every record of a big table to a small table based upon the points of an specific field in a record. Partitioning algorithm are mainly utilized in data center area to increase the efficiency of big-data. The performance of query processing is purely based upon the efficiency of partitioning algorithm.

Input: (S, PR); S: input record; PR: the partition vector set.

Output: QID; QID: a partition identifier for partition q.

1: The input record S is parsed into multiple columns-families by the defined schema;

2: Calculate the BID with its value from aggregation;

3: Get the partition vector P_{Ri} from PR with the BID, and let P_{Ri}=<BID, M_r>;

4: Set target partition identifier,

QID <- < BID, random [1, P_{Ri}.Mr]>;

5: A sample will be build in the partition BID, such as:

TrackQID<- TrackQID + 1

Sum_{PID} <- sumPID + N;

$Sample_{PID}$ <- $sum_{k,l,m,r}$ / TrackQID);

6: ZID <- Hash (PID, TrackQID);

7: Send S to partition QID;

8: Return QID.

Fast Range aggregate queries

A new estimated reasonable approach that produces accurate estimations rapidly for range-aggregate query in big data environments is known to be the Fast RAQ which is our proposed approach. For Fast RAQ the ad-hoc range aggregate queries have the O(N/P*B) time complication and O(1) time difficulty for data updates. FastRAQ will have O (1) time complication for range collective queries when proportion of edge-bucket cardinality (h0) is little. It is believed that the Fast RAQ process is providing a better initial point for mounting real-time answering system for big data analysis method.

Fast RA Quering (Q)

Input: M

Output: R

1: Transfer the request M to all partitions;

2: For each partition (k) in partitions do

3: Calculate the cardinality estimator of range Zk1< abl name K< Zk2 from the local histogram, and let Dik be the estimator of the kth dimensions;

4: Calculate the cardinality estimator of range Zn1<abl name n<Zn2 from the local Histogram, and let Den be the estimator of the nth dimensions;

5: Combine Dik and Den using opr = Di Merged;

6: Count (i) <- h (Ce Merged); // h is a function of cardinality estimation.

7: Calculate the sample for AggColumn, and let Sample f be the sample;

8: TOTAL (F) <- Count (f) * Sample (f); // TOTAL(f) is a local RAQ output;

9: R <- TOTAL (f), 1<f<Z // Z is the amount of partitions;

11. Return R;

3. Experimental setup

The performance of our proposed approach is identified by an experimental result which is conducted with the following requirements. CPU G2020, Windows 7, processor speed of 2.90 GHz and Intel Pentium are the subsequent configurations are used to execute our projected methods.
The below Fig. 2 is presenting the accuracy of the proposed technique FRAQ which is better than other existing technique. The proposed technique is providing more accuracy over the existing issues.

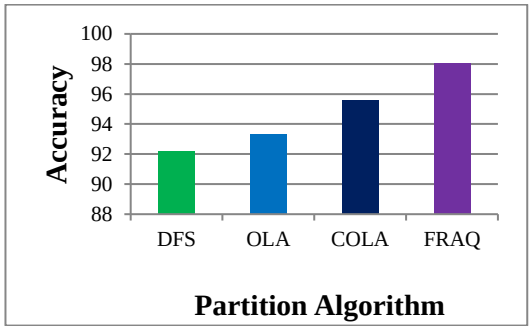

Fig. 2: Accuracy measurement

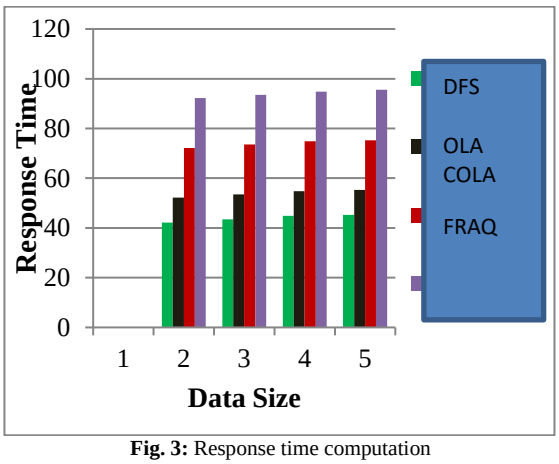

Fig. 3: Response time computation

The above fig. 3 is presenting the Response Time of the proposed Technique Fast Range aggregate queries which is better than other existing techniques. The proposed method is providing more response time over the existing issues. Hence as per the DFS(Distributed file system) has the least response time of 40 whereas OLA(online aggregation) has response time of 50 whereas COLA(Cloud online aggregation) has the response time of 70 and finally FRAQ (Fast range aggregate queries) has the response time of 90 which is the highest of other techniques.

4. Results and discussion

Fig. 4,5,6,7 illustrate the creation of bank database, creation of bank folder in Google drive, File Processing and Final outcome.

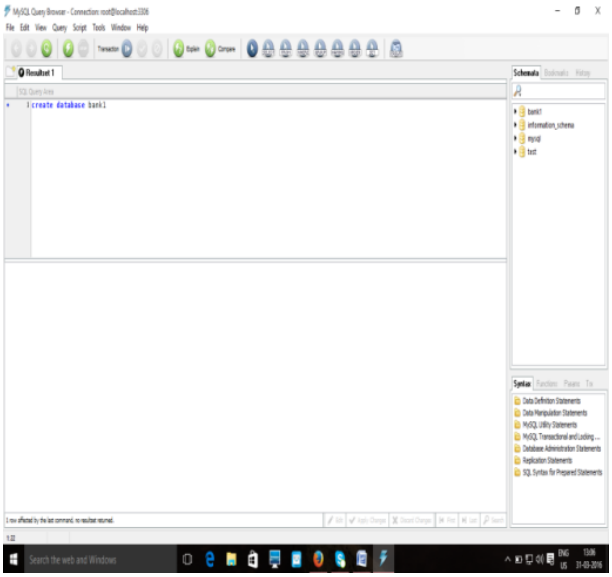

Fig. 4: Create bank database

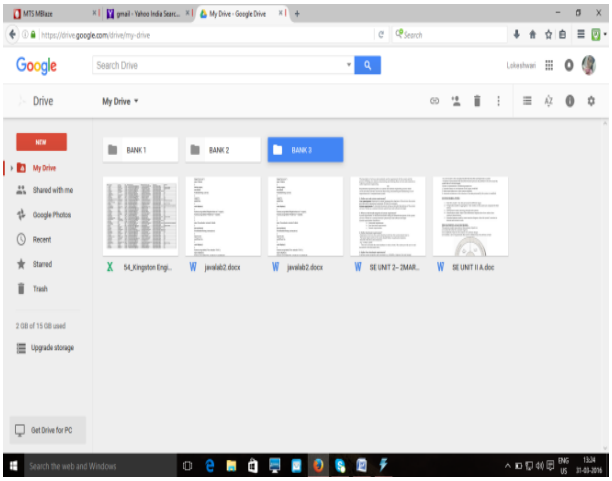

Fig. 5: Create bank folder in Google drive

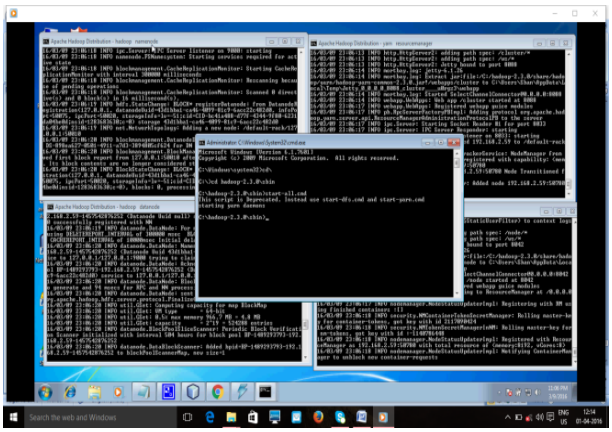

Fig.6: File Processing

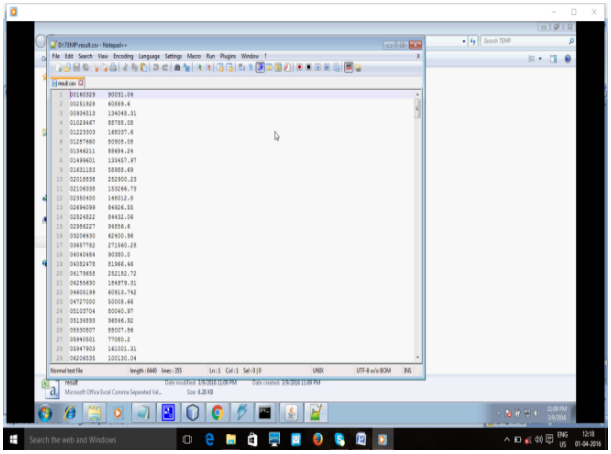

Fig. 7: Final Outcome

5. Conclusion

Big data is known to be the uncertain, real time and unstructured data that are present in an enormous amount. Even there are different technologies existing in today's world querying on such data is a quiet challenging task. The exact pattern matching method and Balance partition method, proposed in this paper are useful for managing theses queries. The balance partition technique is used for dividing the big data into division at first and then it stores in particular partition. The indexing is provided in the partitions which are used through an accurate pattern matching method for successful managing of queries. Also the paper is implemented on the crown of Hadoop technologies which sustain the java language.

References

[1] Adamic LA, Zhang J, Bakshy E & Ackerman MS, "Knowledge sharing and yahoo answers: everyone knows something", Proceedings of the 17th international conference on World Wide Web, (2008), pp.665-674.

[2] Agichtein E, Castillo C, Donato D, Gionis A & Mishne G, "Finding high-quality content in social media", *Proceedings of the international conference on web search and data mining* (2008), pp.183-194.

[3] Akihiro Tamura F, Hiroya T & Manabu O, "classification of multiple sentences", *Int. Joint Conf. Natural Language Processing*, (2007).

[4] Chua TS, Hong R, Li G & Tang J, "From Text question-answering to multimedia QA on web-scale media resources," *ACM Workshop Large-scale Multimedia Retrieval and Mining*, (2009).

[5] Tamura A, Takamura H & Okumura M, "Classification of multiple-sentence questions", *International Conference on Natural Language Processing*, (2005), pp.426-437.

[6] Cui H, Kan MY & Chua TS, "Soft pattern matching models for definitional question answering", *ACM Transactions on Information Systems (TOIS)*, Vol.25, No.2, (2007), pp.1-8.

[7] Hsu WH, Kennedy LS & Chang SF, "Video search reranking through random walk over document-level context graph", *Proceedings of the 15th ACM international conference on Multimedia*, (2007), pp.971-980.

[8] Li G, Li H, Ming Z, Hong R, Tang S & Chua TS, "Question answering over community-contributed web videos", *IEEE Multi Media*, Vol.17, No.4, (2010), pp.46-57.

[9] Nie L, Yan S, Wang M, Hong R & Chua TS, "Harvesting visual concepts for image search with complex queries", *Proceedings of the 20th ACM international conference on Multimedia*, (2012), pp.59-68.

[10] Nie L, Wang M, Gao, Y, Zha, ZJ & Chua TS, "Beyond text QA: multimedia answer generation by harvesting web information", *IEEE Transactions on Multimedia*, Vol.15, No.2, (2013), pp.426-441.