Optimization of Distributed Sensor Network Data Processing using Overlapped Area Data Removal Technique

Seungwoo Jeon[1], Bonghee Hong[1], Joonho Kwon[2], Yoon-sik Kwak[3], Seok-il Song[3]

1 Dept. of Computer Engineering, Pusan National University
Jangjeon-dong Geumjeong-gu, Busan 609-735, Korea
{i2825t, bhhong}@pusan.ac.kr
2 Institute of Logistics Information Technology, Pusan National University
Jangjeon-dong Geumjeong-gu, Busan 609-735, Korea
jhkwon@pusan.ac.kr
3 Dept. of Computer Engineering, Korea National University of Transportation
50 Daehak-ro, Chungju-si, Chungbuk, 380-702, Korea
{yskwak, sisong}@ut.ac.kr

Abstract. Sensor network is needed for managing huge ranch with a very large number of commodities. In the ranch, there will be deployed a certain number of sensor hub for retrieving the sensor data sent by the sensor. The ranch will be divided into several patterned regions so that the sensor hub can cover the whole ranch area. It is unavoidable that there will be some overlapped area between sensor hubs, since 100% coverage is a must. Thus, when a sensor moves inside this area, there will be redundant sensing data received by hubs covering this overlapped area. MapReduce and Hadoop big data processing framework will be used to process the data efficiently, and to optimize the processing time, we propose data removal technique to eliminate the redundant data. The data will be compared using a certain parameter and the redundant data will be eliminated according to predefined rules.

Keywords: Hadoop, MapReduce, Redundant data, Removal technique

1 Introduction

The biggest ranch in the world, Anna Creek, covers 6 million acres of land in Australia. It can hold more than 1 million cows, lambs and muttons. An effective monitoring system must be provided to help farmers do their job in producing a high quality livestock product. The most typical materials used for building the systems are: RFID that is attached to animal's body, Hub as sensor reader that will send the collected sensing data to the server, and an end user application that will provide some useful information about the ranch in real time.

Since the hub reading range is spherical, maintaining a hundred percent coverage of the ranch area requires the hubs to be deployed at a certain position that will make the overlapped area between hubs become unavoidable. If the sensor tag sends its data from the overlapped area, obviously it will be read by more than one hub covering that area. This will cause the redundant data sent to the server.

The remaining of this paper is organized as follows. We discuss previous works on the previous duplicate stream data removal and MapReduce framework in Section 2. We define a problem in

the target environment in Section 3. In Sec. 4 and 5, we propose the technique to solve the problem and measure performance. Finally in Sec. 6 summary of this paper is presented.

2 Related Work

Redundancy in stream data may occur due to an existence of more than one reader covering a specific region. Previous work on [1] tried to cope with this problem by proposing three algorithms: Intersection, Relative Complement, and Randomization Algorithm. Intersection algorithm compares sensing data between two readers, so that if there is a redundant data only one of it is kept. Relative Complement ignores all of the redundancy by keeping only the non-redundant data. Randomization Algorithm uses "0" and "1" values to mark the first reader so that only the first reader is kept. All of these three algorithms only work for two readers overlapping; they will not fit into the problem where there exist more than two readers covering the overlapped area.

3 Target Environment and Problem Definition

3.1 Target Environment

Most of the existing system in [2][3][4] attached sensor tag to non-moving objects. Therefore, we need a different approach to create a system with sensor tag attached to moving objects. At the beginning, ranch is divided into regions where each hub will cover each of it so that 100% coverage can be achieved. The sensor tag will be attached to the object so that it can transmit its condition periodically to the hub, then sever will collect all of the sensing data sent by the hub.

3.2 Problem Definition

The hub sensing range is spherical; in order to achieve 100% coverage area, overlap between hubs is unavoidable. When the sensor tag sends its data from this overlapped area, the sensing data will be read by multiple hubs covering this area. Obviously, this will cause redundancy on the server side. Moreover, these redundant data will raise the size of the input data for the map phase and also increase the intermediate data, resulting to longer data processing time.

4 Solution

We try to eliminate the redundancy by proposing a redundant data removal technique. The hub is assigned with a certain index depending on the region it covers. Using our removal technique, the sensing data will be compared and the redundancy will be eliminated from the server storage.

4.1 Hub Index Number

The ranch is divided into several regions where each hub will cover a certain region. The index number of the left-side region hub is smaller than the right-side, and the index number of upper-side region hub is also smaller than the lower-side hub.

4.2 Redundant Data Removal Technique

The sensing data received by server consists of several parameters. Some of them can be used to distinguish whether it is redundant or not. In this technique, we use Hub ID, Sensor ID and Timestamp to identify which sensing data should the server keep. As we explained before, each hub will have an index number assigned to it depending on the region that it covers. Whenever there are several sensing data which have the same Sensor ID within less than 3 minutes time difference, we will only keep the sensing data from the hub with the smallest index number. Figure 1 shows the illustration about the redundant data removal.

TIME	HUBID	SENSID	HUMID
5:10	HUB01	SENS_1	15%
5:10	HUB01	SENS_2	16%
5:11	HUB02	SENS_3	16%
5:11	**HUB02**	**SENS_2**	**16%**
5:11	HUB02	SENS_4	16%

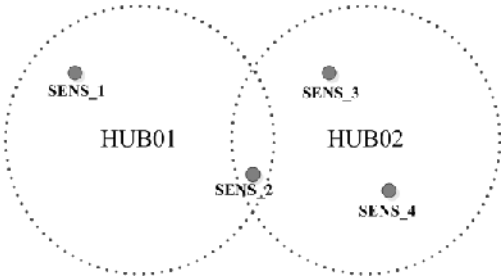

Fig. 1. Detected redundant data will be removed.

5 Test

Performance tests are made on Linux PC, with an Intel Core 2 Quad Q6600 (4 CPUs) 2.4 GHz processor, and 4 GB of main memory. These tests evaluation is conducted using synthetic dataset since there is no dataset for experimental purpose is published. The dataset consists of Hub ID, Sensor ID, timestamp, location, temperature and humidity, with the size of 1 Gigabytes. By using our redundant data removal technique, the system managed to eliminate the redundant sensing data from the original 1 GB to around 330 MB data. After that, we continue our test by doing the MapReduce process on a single node Hadoop Cluster (Ubuntu 12.04, 4 CPU @2.4 GHz, and 4 Gigabytes memory). The next tests are done to prove that file size can affect the processing time significantly. The process flow is defined as follow, mapper will read every record from the sensing dataset, and then the reducer will select a specific sensor ID which already specified by user, to be monitored. Result shows that the original dataset sized 1 GB takes 16 seconds to finish, while the non-redundant dataset only takes 10 second (6 seconds faster). By looking at this performance test, it is proven that optimizing the dataset by eliminating the redundant data can improve the processing time significantly.

6 Conclusion

Eliminating the redundancy inside the sensing dataset not only save the server storage spaces but also improves the future processing time. This paper proposes a new technique to remove

the redundant data that may occur between two or more sensor hub and proves that after the dataset is optimized, the MapReduce job can be done faster.

Acknowledgments

This research was supported by Bio-industry Technology Development Program, Ministry for Food, Agriculture, Forestry and Fisheries, Republic of Korea.

References

1. P. Pupunwiwat, B. Stantic, “Location Filtering and Duplication Elimination for RFID Data Streams”, International Journal of Principles and Applications of Information Science and Technology, vol.1, no.1. pp 29-43, 2007
2. W. M. Choi, J. S. Jeong, B. J. Kim, D. G. Kim, “Garlic cold storage”, Korea, 1020110012155, 2011
3. J. S. Seo, M. S. Kang, Y. G. Kim, C. B. Sim, S. C. Joo, C. S. Shin, “Implementation of Ubiquitous Greenhouse Management System Using Sensor Network”, Journal of Korean Society for Internet Information, vol.9, no. 3, pp. 129-139, 2008
4. K. O. Kim, K. W. Park, J. C. Kim, M. S. Chang, E. K. Kim, “Establishment of Web-based Remote Monitoring System for Greenhouse Environment”, Journal of The Korea Institute of Electronic Communication Sciences, vol. 6, no. 1, pp. 77-83, 2011