Aggregate Model: An Approach to Improve Query Response Time for Bulky Data

Aniqa Nawaz Bhutto Suhni Abbasi Sanober Behrani

***Abstract*—Sensor network is collection of huge number of sensor nodes to capture and monitor the physical sensing capabilities such as temperature, sound, light, humidity, pressure, etc. At the Present sensor networks supposed to be active and pre-programmed devices that can continuously forward data to the central storage location for analysis and to performing offline queries. For such tremendously growing data from sensors, prevailing tools and methods are required to process data efficiently and accurately instantly; otherwise the response time of data processing will be too much annoying. To overcome this challenge, an aggregation model was designed to improve query response time to overall improve the quality of the sensor network data. The developed model consists four sub modules i.e. source, time, query, and data module. Source module comprises different aggregate functions. Time module is about to apply aggregation at three different time intervals. Query Module had defined that one finest aggregation level is not sufficient to answer all queries because each query varies in their nature. Data module was performed to improve the accuracy of the data. The result of the study revealed that applying the aggregation at different time intervals improved the query response time and accuracy measurement has the significant impact on the quality of the data.**

***Index Terms*—Sensor Networks, Big Data, Aggregation, Query Response Time, Data Quality, Accuracy.**

I. Introduction

Sensor network is a combination of transducers with communication infrastructure that are planned to monitor and record conditions at varied locations. Mostly these networks are used to monitor parameters like temperature, wind direction, humidity, speed chemical concentration, power line voltage and vital body functions. It was very challenging for WSNs(Wireless Sensor Networks)-based application to make real time decisions due to the computing devices, communicating capacities, and gigantic and rapidly changed data generated by WSNs [2]. Day by day evolution in communication technology foresees to deploy sensor nodes in any area of everyday life. These types of networks generate a large amount of data. E.g. single temperature sensing device measures 262,800 data points in every two minutes per year. up to the year 2016 it would generate 1 million data points [16].

The bulky data of sensor networks or flood of digital data can be described by three characteristics, referred as three Vs; volume, variety and velocity. Volume means huge datasets. Variety means heterogeneous, complex, and variable. Velocity means data generated in the constant stream.. There are the number of challenges related to bulky data. Due to its characteristics some of the inherited challenges in bulky data are capture, storage, search, analysis, and virtualization. Some of the typical areas that produce bulky data are meteorology, genomics, physics, simulations, biology, and environmental science [4]. To improve the performance and storage utilization of bulky data of sensor networks there is the need to reduce the volume of data being stored and develop new techniques[21].

Data aggregation is a process in which raw data is gathered and expressed in a summary form, for the purpose of statistical analysis. Usually, data aggregation works on bulky data or data marts that do not provide much information value as a whole. Important applications of data aggregation are the gathering, utilization and presentation of data that is presented on Internet of Things (IoT). One of the main challenges of data aggregation is to determining the best aggregation level at which the data set can be aggregated. The decision of aggregation level will be determined by many factors like users query behavior, the nature of the data, the type of the application running over the data, and many more. Query processing is one of the important issues in data management to achieve accurate and fast query response. The query should be done in a way that no any important part of data left out in the result, and the query is done in minimal time. Due to the infinite nature of sensor data query processing is a crucial challenge in WSNs [22]. One way to improve query response time by using high processing power hardware, but due to rapid advancement in technology and information collection rate both are incomparable. Another way is to use powerful tools and methods.

A case study of airport weather systems was used for analysis. Weather information could be used for safety hazards for aircrafts to operate it in all types of weather. If the pilots quickly get updates and recognize fact and its relative location to their aircraft, they can try to divert around hazards like thunderstorms, air turbulence, and wind shear zones. Immediate and accurate query processing for data was a challenge. To overcome that challenge this study was designed.

In this research we had studied and analyzed various techniques to improve query response time needed for bulky data and then designed and implemented the data aggregation model following with the comparison with and without using the aggregate model was conducted.

II. RELATED WORK

Clustering and data aggregation together improves the efficiency of resource utilization (energy and memory respectively) [14]. Network cube hierarchies were revealed to summarize sensor data and, to improve the efficiency of spatial aggregate queries. In this study, authors had presented PTIME algorithm to compute the query plans over cube and showed how they could contain multiple queries [3]. BlinkDB approximate query engine [19] had supported interactive ad hoc exploration queries over gigantic datasets. Authors had used column set based optimization framework to calculate stratified sampling. Error latency profiles (ELPs) were created to estimate error or response time of each query on each available sample. Bucket based parallel aggregation (BPA), build quad tree by dividing query region in to several cells with respect to distribution of sensor nodes [13]. For optimizing object-oriented query response time a model was designed which had analyzed that query response time was directly proportional to the line of code that were executed [23]. It was proposed that semantic technologies and query optimization techniques to satisfy users to access the massive data. Authors had designed the no. of algorithms and efficient massive data query and optimization mechanism Seman Query. Their simulation results showed that Seman query can reduce the query cost and improve query efficiency on a large cluster [11]. For big data, different testing methods and, data quality issues for functional and nonfunctional requirements by using structural testing were analyzed. [15]. In-network data merge technique based on Exponential Weighted Moving Average (EWMA). Performance evaluation of EWMA showed that proposed data aggregation technique improved the network efficiency and latency as compared to delta method [5]. Influence of data transfer techniques on storage and requested small segment from chunks of data rather than a huge request was demonstrated. Several big data transfer techniques like, brute force parallel processing, history-based data allocation, equal page size request, varying page size request were examined. Evaluation results showed that in getting back bulky data sets users request had a great impact on storage performance. The size of independent storage request and extra time required for multiple requests for big data transfer approach based were inversely proportional. Larger requests had performed better than smaller requests [1]. Aggregate lineage technique to create summary of data, had effectively describe the performance of data in aggregate for SUM [10]. Proposed technique evaded full scan of the table for the query optimization and indexing of model-view of sensor cloud data. Suggested technique was consisted on KVI-index for modeled segments in key-value stores, which was composed of dual interval indices on the time and sensor value dimensions [22].

III. METHODOLOGY

Data aggregation is one of the effective strategies to deal with bulky data. Data aggregation simplifies large data set by summarizing groups of data based on some criteria to achieve the objectives of the study. The research study work aimed to design the data aggregation model to improve the query response time. As shown in Fig.1 we have proposed aggregation model for airport weather systems.

A. Data Collection

In this research study, secondary data of sensor networks of airport weather systems was used as sample data. It was downloaded in the comma delimited format from the Iowa environmental mesonet [12].

B. Sub-Modules of Aggregate Model

1) Source Module: The taken source data was a large data set in CSV (comma separated value) format, it contains approx. seven lac records. It was obvious that querying over this data, take quite longer execution time, whereas the real time system such as Airports Weather System always demands fast query response time. In source module, aggregation queries were applied on the main_weather table that contain the whole data set. Further the main_weather table was divided into small chunks, and queries were tested after joining these chunks. Following type of queries were used to perform basic aggregation functions like MAX , MIN, COUNT, SUM AVG for the temperature dew point, sea level pressure, humidity, and wind direction at each weather station from main_weather (source table) as well as on the chunks.

a) Relational Algebraic Query for Source Table: p (max_temperatue, max_dewpoint, max_Sea_Level_Pressure, max_Realtive_Humidity, max_Wind_Direction, (station, G MAX temf, MAX dewf, MAX vsby, MAX relh, MAX drct) (main_weather))

b) Relational Algebraic Query for Chunks with Joins: (w1, (station, G MAX temf, MAX dewf, MAX vsby, MAX relh, MAX drct)(weather1)) (w2,(station, G MAX temf, MAX dewf, MAX vsby, MAX relh, MAX drct)(weather2)) (w3, (station, G MAX temf, MAX dewf, MAX vsby, MAX relh, MAX drct)(weather3)) (w4, (station, G MAX temf, MAX dewf, MAX vsby, MAX relh, MAX drct)(weather4)) (max_temperatue,max_dewpoint, max_Sea_Level_Pressure, max_Realtive_Humidity, max_Wind_Direction, w1 U w2U w3 U w4).

2) Time Module: Three aggregation levels, namely 1,000 sec, 10,000 sec and 100,000 sec were taken as time module. For instance in the Aggregation Level-1 (1000 sec), we had considered all the data points arrive in 1000 second of interval. And this whole interval data was represented by the averaged value as single record in this module. Three such aggregation levels said Aggregation level-1, Aggregation Level-2 and Aggregation Level-3 were designed in this module for 1000, 10,000 and 1, 00,000 seconds respectively.

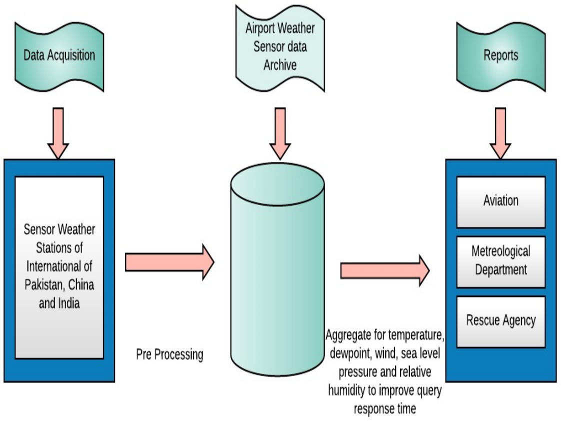

Data Acquisition
Airport Weather Sensor data Archive
Reports
Sensor Weather Stations of International of Pakistan, China and India
Pre Processing
Aggregate for temperature, dewpoint, wind, sea level pressure and relative humidity to improve query response time
Aviation
Metreological Department
Rescue Agency

Fig. 1: Aggregation Model for Airports Weather System

a) Aggregation Level-1 at 1000 Seconds: This level was used to create time interval of 1000 seconds in main_weather table. Following query was used to select the count, minimum temperature, maximum temperature, average dew point, average relative humidity, average wind direction, and starting time of interval from main_weatherrelation at the time interval of 1000 seconds.

b) Relational Algebraic Query: (MIN_Temperature, MAX_Temperature, AVG_Dewpoint, AVG_Realtive_Humidity, AVG_Wind_Direction, Time_to_Start, (valid_GMT_time DIV1640 G COUNT, MIN(temf), MAX(temf), AVG(dewf), AVG(relh), AVG(drct) valid_GMT_time¿= 20110101000000)(main_weather))

3) Query Module: The query module was design to analyse the query response time of different aggregation functions. Following question were supposed to be taken in analysis;

a) Why do we make the aggregation at different levels?: For the fast query executions, having a fine aggregation level works satisfactory. The fastness of query execution depends upon the no. of records in aggregation level. The less no. of rows have the fastest query execution time.

b) Why not we use only the finest aggregation levels?: If query was to be made for 2011-01-01, 19:00:20 to 2011-01-01 to 20:00:20. Then Aggr-100,000 would not help any more as it does not had an entry for a 3600 se-conds long event. This was the reason to use different aggregation level.

c) How is it possible to query over this aggregation levels?: The some queries would not answer Aggr-100,000, and even on Aggr-10,000, so there was necessary to built Aggre-1000 table, also in some case original data may be needed.

4) Data Module: Data quality refers to the degree of completeness, validity, consistency, timeliness, uniqueness, and accuracy of data. It can also be helpful for decision making for aggregation methods. Data quality depends upon the source of the data itself, the sensors reliability, measurement uncertainty and many more factors [28]. In this study, quality of data was measured on the basis of accuracy. To measure the accuracy of data statistical formula [24] was used:

Accuracy = No. of correct assessments/ No. of all assessments

IV. RESULTS

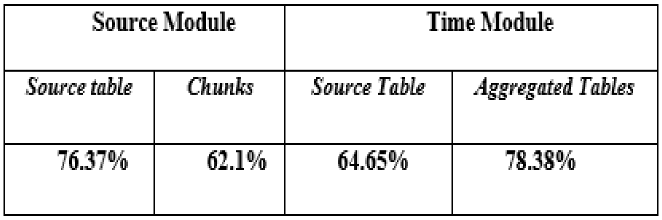
A. Source Module

When query was used to retrieve all the records from main_weather (source table) the query took 15.156 seconds, which was quite enough time for streaming real time applications, because in streaming applications the data is continuously receiving and in very huge quantity. As compare to this when standard aggregate functions were applied on source table QRT was 76.37% faster and on chunks it was 62.1 % faster.

B. Time Module

The whole data set was aggregated at three different time intervals. Namely Aggregation Level-1 at 1000 seconds, Aggregation Level-2 at 10000 seconds, and Aggregation Level-3 at 100000 seconds. And the same query was also applied on the source table with aggregation. Table.1 shows the results of Source and time module that how much it was fastest .

TABLE I. RESULTS OF SOURCE & TIME MODULE



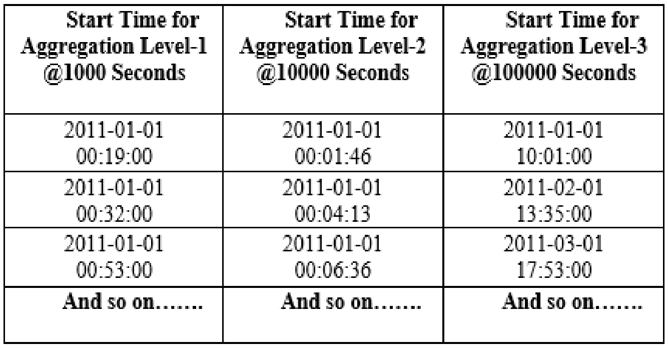
Source Module		Time Module	
Source table	*Chunks*	*Source Table*	*Aggregated Tables*
76.37%	62.1%	64.65%	78.38%

C. Query Module

To achieve the fastest query response by applying the aggregation, the Query module was design to answers these questions.

a) Why do we make the aggregation at different levels?: For the fast query executions, having a different aggregation level works fine. The fastness of query execution depends upon the number of records in aggregation level. The aggregation level which had less records was had the fastest query execution time. Table II. shows start time of each interval for aggregation levels. When query was executed on source table to aggregate it at 1000 seconds from 2011-01-01 00:00:00 for minimum temperature, maximum temperature, average dew point, relative humidity, and wind direction the QRT was recorded 73.4% faster than source query and 106547 rows were retrieved. When source table was aggregated at 10000 seconds from 2011-01-01 00:00:00 for minimum temperature, maximum temperature, average dew point, relative humidity, and wind direction the QRT was recorded 78.23%faster than source query and 11436 rows were retrieved. And when it was aggregated for 100000 seconds from 2011-01-01 00:00:00 for minimum temperature, maximum temperature, average dew point, relative humidity, and wind direction the QRT was recorded 83.5% faster than source query and 1068 rows were retrieved.

TABLE II. Starting Time of Interval for Aggregation Levels



Start Time for Aggregation Level-1 @1000 Seconds	Start Time for Aggregation Level-2 @10000 Seconds	Start Time for Aggregation Level-3 @100000 Seconds
2011-01-01 00:19:00	2011-01-01 00:01:46	2011-01-01 10:01:00
2011-01-01 00:32:00	2011-01-01 00:04:13	2011-02-01 13:35:00
2011-01-01 00:53:00	2011-01-01 00:06:36	2011-03-01 17:53:00
And so on.......	**And so on.......**	**And so on.......**

b) Why not we use only the finest aggregation levels?: When query was made for 2011-01-01, 19:00:20 to 2011-01-01 to 20:00:20. Then the table Aggr-100,000 had not helped any more as it did not had an entry for a 3600 seconds long event. This was the reason to use different aggregation level..

c) How is it possible to query over this aggregation levels?: The some queries would not answer Aggr-100,000, and even on Aggr-10,000, so there was necessary to built Aggre-1000 table, also in some case original data may be needed.

D. Data Module

At each aggregation level, there were the count of records encountered, and that count was also one field in the aggregated tables. As a consequence, measuring data quality is straight forward, as it can be simply fetched from the aggregated tables. Data quality was measured from aggregated table at 100000 seconds as it can be seen easily.

Fig. 2 shows that data quality from 10/01/2011 to 10/31/2011 It shows that from time interval 10-01-2011 to 10-31-2011 has the average of 1334 samples, whereas for the time interval 11-01-2011 to 11-30-2011 data quality gradually gets high and reaches its peak at average of 2015 samples. After that for the time interval 12-01-2011 to 12-31-2011 data quality decreased to the average of 921 samples. Based on above observations, it can be said that data quality for the time interval 11-01-2011 to 11-30-2011 was best. While data quality of time interval 10-01-2011 to 10-01-2011 was better than the time interval of 12-01-2011 to 12-31-2011.

V. Discussion

It was analyzed from literature that to improve the query response time, the new techniques were developed and even more are demanding today. One of the proposed technique was Lossy compression techniques, like wavelet and regression but it was also analyzed that this techniques degrades the accuracy of sensor data. Many research studies also focused on accuracy and storage utilization of bulky data generated from sensor networks.

Sensor networks are deployed in every field and data grows from these networks at the incredible rate. Therefore, this research focused on designing an aggregation model for ever and over growing data of sensor networks to improve overall query response time.

The Model was based on four sub modules namely the source, time, query and the data module. Maximum, Minimum, Average, Sum and count aggregate function were applied to analyze the query response time with and without using aggregation model.

The results for the Source module showed that when aggregate functions were applied on whole source table, the query response time was less as compared to applying aggregate functions after joining the chunks of table. The overall results showed that aggregate functions have great impact on the query response time.

The Time module results showed that query response time for aggregating the source table for count, maximum, minimum, average and sum functions at specific time interval depends upon the number of rows retrieved. Query response time of aggregated tables was less than source table. Query module had defined that one cannot use only the finest aggregation level, because all the queries could not get answer only from the finest level. It depends upon the nature of the query as well.

Finally the Data module was used to measure the quality of the data on the basis of accuracy. Statistical Formula of accuracy was applied to measure the accuracy. The results showed the significant effect on data quality by applying this module.

The results were well supported by [23] in which it was reported that query response time was directly proportional to the line of code. Results had also shown similarity with [25]in which it was analyzed, with FastRAQ, that by enlarging dataset size performance of range aggregate queries was improved. Results were also supported [22]in which it was suggested that KVI (key value Index) base on time interval and sensor value dimensions to improve the performance query response time.

VI. Conclusion

By applying the aggregate functions on bulky data of sensor networks query response time was improved. The Query response time could further decrease by aggregating the whole data set with time intervals. Moreover, for bulky data of sensor networks, quality of data was measured based on accuracy.

VII. Limitation and Future Work

This work is limited to the offline data of sensor networks, in future real time data of sensor networks could be analyzed. Data quality depends upon the source of the data itself, the sensors reliability, measurement uncertainty and many more factors, here we had only measured data quality based on accuracy it is possible data quality may vary in real time due to other factors.

References

[1] A. H. Villa. 2013. Storage Matters: Evaluating the Impact of Big Data Transfer Techniques on Storage Performance. Int'l Conf. Internet Computing and Big Data. pp .182-188.

[2] A. Mahmood, K. Shi, S. Khatoon and M. Xiao. 2013. Data Mining Techniques for Wireless Sensor Networks: A Survey. IJSDN.

[3] A. Meliou, C. Guestrin and J. M. Hellerstein. 2010. Multiresolution Cube Estimators for Sensor Network Aggregate Queries. J. CoRR.

[4] A. Zaslavsky, , C. Perera and D. Geogakopoulos. 2013. Sensing as Service and Big Data. J. CoRR.

[5] B. G. Kim, K. W. Kim, J. W. Choi and Y. K. Kim. 2013. A Delta Based Data Aggregation Scheme using EWMA in Wireless Sensor Network.J. ASTL, 23: 186-171.

[6] B. He, and Y. Li. 2014. Big Data Reduction and Optimization in Sensor Monitoring Network. Journal of Applied Mathematics.

[7] D. O. Kim, L. Lei, I. S. Shin, J. J. Kim and K. J. Han. 2013. Spatial TinyDB: A Spatial Sensor Database System for the USN Environment. IJDSN.

[8] D. Takaishi, H. Nishiyama, N. Kato and R. Miura. 2014. Towards Energy Efficient Big Data Gathering in Densely Distributed Sensor Networks. J. IEEE Transactions on Emerging Topics in Computing, 2(3): pp.388-397.

[9] D.V. Dimitrov. 2016. Medical Internet of Things and Big Data in Healthcare. Healthcare Informatics Research, 22(3), 156163.

[10] F.N. Afrati, D. Fotakis and A.Vasilakopoulos.2013. Ecient Lineage for SUM Aggregate Queries. J.ArXiv e-prints.

[11] G. Zhang, C. Li, Y. Zhang and C. Xing. 2012. Massive Data Query Optimization on Large Clusters: JCIS, 8(8): 3191-3198.

[12] IEM:: Download ASOS/AWOS /METAR Data. http://mesonet.agron.iastate.edu/request/download.phtml? network=MN_ASOS

[13] J. -J. Kim, S. I. In-Su, Y. -S., Zhang, D., -O. Kim and K. -J., Han. 2012. Aggregate Queries in Wireless Sensor Networks. IJDSN.

[14] L. Butyan, and T. Holczer. 2010. Perfectly Anonymous Data Aggregation in Wireless Sensor Networks. J.MASS, 513-518.

[15] M. Gudipati, S. Rao, N.D. Mohan and N.K. Gajja. 2013. Big Data: Testing Approach to Overcome Quality Challenges. Infosys Labs Briefings, 11(1):65-72.

[16] M. Keller, J. Beutel, O.Saukh, andL. Thiele. 2012. Visualizing Large Sensor Network Data Sets in Space and Time with Vizzly. In Proc. LCN Workshop, pp.925-933.

[17] M. Quwaider, and Y. Jararweh. 2014. An Efficient Big Data Collection in Body Area Networks. 5th Intl Conference on Information and Communication Systems.

[18] Mysql. n.d. About MySQL. 2014. Retrieved From. http://www.mysql.com/about/

[19] S. Agarwal,, B. Mozafari, A. Panda, H. Milner, S. Madden and I. Stoica. 2013 Blink. DB: Queries with Bounded Error and Bounded Response Time on Very Large Data. In Proc. of the 8th ACM European Conf. Computer System,pp. 29-42.

[20] S. Rekha, M. Nambiar. 2016. Predicting SQL Query Execution Time for Large Data Volume, Proceedings of the 20th International Database Engineering & Applications Symposium.

[21] S. Syal, I. Singla. 2013. Big Data Analysis and Storage Optimization: Intl J. Technological research, 1(4):213-216.

[22] T. Guo, T. G. Papaioannou, and K. Aberer. 2014. Efficient Indexing and Query Processing of Model-View Sensor Data in the Cloud .J. Big Data Research, 1: 52-65.

[23] V. Saxena, and S. Kumar. 2012. Object Oriented Query Response Time for UML Models: J. Software Engineering and Applications, 5(7): 508-512.

[24] W. Zhu, N. Zeng and N. Weng. 2010. Sensitivity, Specificity, Accuracy, Associated ConfidenceInterval and ROCAnalysis with Practical SASImplementations.NESUG proceedings: Health Care and Life Sciences.

[25] X. Yun, G. Wu, G. Zhang, K. Li and S. Wang. 2014. FastRAQ: A Fast Approach to Range-Aggregate Queries in Big Data Environments. IEEE Transactions on Cloud Computing, 6(1).

[26] Y. Chaowei , Q. Huang, Z. Li, K. Liu & F. Hu 2017 Big Data and cloud computing: innovation opportunities and challenges, International Journal of Digital Earth, 10:1, 13-53

[27] Z. Liu, B. Jiang and J. Heer. 2013. imMens: Real-time Visual Querying of Big Data. Computer Graphics Forum (Proc. EuroVis), 32(3).

[28] Z. Qin, Q. Han, S. Mehrotra and N.Venkatasubramanian. 2014. Quality-Aware Sensor Data Management: The of Wireless Sensor Networks. Pp.429-468.

AUTHORS

Aniqa Nawaz Bhutto, aniqabhutto1@gmail.com
Information Technology Centre
Sindh Agriculture University
Tandojam, Sindh , Pakistan

Suhni Abbasi, suhni.abbasi@sau.edu.pk
Information Technology Centre
Sindh Agriculture University
Tandojam, Sindh , Pakistan

Sanober Behrani, sanoberbehrani@gmail.com
Information Technology Centre
Sindh Agriculture University
Tandojam, Sindh , Pakistan