A Time Efficient Approach for Error Location Detection and Correction in Big Data Cloud Storage

Kamal Sharma¹, Karambir Bidhan²

¹University Institute of Engineering and Technology, Kurukshetra University, Kurukshetra

²Assistant Professor, University Institute of Engineering and Technology, Kurukshetra University Kurukshetra

Abstract: The approach, the burst error acknowledgment relies upon the without scale framework topology and a substantial part of disclosure operations driven in obliged transient or spatial datadivides as opposed to a whole big data set. From this time forward the recognizable proof and range method can be essentially stimulated. In addition, the acknowledgment and territory endeavors can be appropriated to cloud stage to totally exploit the algorithm power and big storage. Through the investigation on our cloud computing stage of Cloud data storage, it is shown that our proposed methodology can fundamentally diminish the ideal opportunity for error recognition and correctionlocation in big data sets created by extensive scale sensor system frameworks with adequate error distinguishing precision.

Keywords: burst error, big data, cloud data storage, error recognition and correction, spatial data

1. Introduction

An accumulation of data sets so enormous and composite that it gets the chance to be difficult to handle with standard data get ready and organization applications is called big data. Big data speaks to the headway of the human insightful capacity, technique to get, direct, and handle the data inside a snuck past time. Processing principles on big data at present differ at the essential level of reflection on whether the get ready will be done in group mode, or progressively/ close constant on gushing (data that is persistently coming in and ought to be taken care of quickly). Around there, we include two specific establishments: Hadoop for error taking care of and Ubuntu for steady get ready. MapReduce is a programming model and a related execution for taking care of and making unlimited datasets [1].

Big data contrasts from ordinary data in various estimations: (i) Quantity of data sources (ii) Heterogeneous nature of data sources (iii) Dynamic nature of data sources that is updating rapidly (iv) qualities of data sources moves in various points. Distributed computing gives a best stage to planning data which is complex. Temporary use and breaking point on intrigue are basic properties of cloud which makes it compelling for planning big data. For get ready big data applications, security is basic which is given using cloud [2].

Big dataconstructs and strategizeto keepadvancing at a fast pace, yet the major advancements they rely upon have, overall, been envisioned various years earlier. The remarkably extended digitization of human development and machine-to-machine exchanges, joined with generous scale temperate hardware, is making practical various effectively academic musings of parallel and passed on preparing, close by new changes imperative to make them fundamentally more accommodating in genuine applications [3]. The precision of a classifier on a given test set is the rate of tuples that are requested precisely (these test set tuples with names should not to have been used to set up the classifier). So additionally, the precision of a pointer implies how well a given marker can figure the estimation of the foreseen trademark for new or heretofore unnoticeable data. The error rate or misclassification rate of a classifier is fundamentally the rest of the rate of tuples that were not clustered precisely [4].

2. Related Work

For Analyzing shows of models from different edges, related composition for identification of error, big data planning on cloud, for complex framework structures will be investigated and pondered.

a) Big data processing

A process of maintaining how you look at any issue in these works is their adaptability to a considerable measure of data. Calculations have extended their diverse quality to beat more mind boggling procedures. This makes degree of calculations obliged to log off revelation. The Big Data essentials are found not simply in mammoth organizations, for instance, Amazon or Google, yet in various minimal business endeavors that require addressing, stockpiling and recuperation over considerable scale structures. It is at present essential to see the calculation in parallel; using thoughts, for instance, MapReduce [5] for better change.

Distributed computing gives a flawless stage to inciting of big data, stockpiling and unraveling with its huge calculation control [6]. It is unavoidable to encounter the issue of overseeing big data in various veritable applications. Nowadays unique kind of work has been proficient for planning big data with cloud. An average cloud based appropriated structure for big data taking care of is Amazon EC2 base as an organization. A scattered stockpiling is supported by Amazon S3. MapReduce [7] is held onto as a programming model for big data taking care of over distributed computing. The issue of taking care of incremental big data is investigated at various concentrations from various perspectives.

Volume 5 Issue 11, November 2016 <u>www.ijsr.net</u> Licensed Under Creative Commons Attribution CC BY

b) WSN processing in relation with cloud

At the point when data from generous sensor frameworks is ought to have been accumulated and watched remotely sensor-Cloud is profitable for a few applications. For environmental checking, social protection, business trades, transportation, WSN engages imaginative courses of action. Remote sensor framework structures have composed diverse game plans in different fields, for instance, calamity watching, disaster warming, environmental surveying, and business change technique and data gathering. Sensor cloud arranges has been delivered to set up the remote sensor data accumulated by WSN. Plan of sensor cloud is useful in various applications generally when the data is discovered remotely. Big data is difficult to prepare using near to database organization gadgets since volume of big data is extending rapidly with collection in data sets [8].

c) Error detection in networks

Dataerror is unavoidable in various certifiable complex framework structures. To find and discover errors in big data sets ends up being amazingly trying undertaking with commonplace computational powers of standard structures as there is passionate development of big data delivered from complex framework structures, for instance, interpersonal associations and enormous scale sensor frameworks. Wang et al. give a basic grouping to errors on interpersonal associations in perspective of error circumstances examination which outlines the lead of error circumstances. This clustering consolidates 6 sorts of customary errors with missing data or errordata. Quality of four center level framework measures is taken a gander at using this clustering structure [9].

Mukhopadhyay [10] proposed a model based error change procedure for Wireless sensor framework. Savvy sensor frameworks are used as a piece of this correction methodology. This framework relies on upon the change with data design estimate. To find the basic driver of errors is as basic as recognizing and curing error. To break down hidden driver of error, an instrument a sensor framework examining is used. Regardless, the things which ought to be improved are customer interface, flexibility and time execution.

3. Proposed Methodology

We intend to develop a novel error acknowledgment approach by abusing the enormous stockpiling, versatility and calculation constrain of cloud to recognize errors in big data sets from sensor frameworks. Snappy acknowledgment of dataerrors in big data with cloud stays testing particularly, how to use the calculation compel of cloud to quickly find and discover errors of centers in WSN ought to be researched. Proposed execution prepared in taking after modules:

a) Cloud Computing

Distributed computing base is getting the opportunity to be common in light of the fact that it gives an open, versatile, flexible and reconfigurable stage. The proposed error recognizable proof approach in this paper will be established on the request of error sorts. Specifically, nine sorts of numerical data varieties from the standard/errors are recorded and displayed in our cloud error revelation approach. The described error model will trigger the error recognizable proof process. Appeared differently in relation to past error area of sensor framework structures, our strategy on cloud will be made and made by utilizing the colossal data taking care of limit of cloud to enhance error acknowledgment speed and continuous reaction [9]. Our proposed error revelation approach on cloud is especially trimmed for finding errors in big data sets of sensor frameworks. The essential responsibility of our proposed acknowledgment is to fulfill big time execution change in error area without exchanging off error disclosure precision [10].

b) Big Data Processing On Cloud

Big data has transformed into a primary and essential test for forefront society. Distributed computing gives an impeccable stage to big data stockpiling, dissipating and translating with its immense calculation control. MapReduce has been extensively revised from a bundle taking care of structure into a more incremental one to inspect massive volume of incremental data on cloud. It is a structure for get ready parallelizable issues across over big data sets using a broad number of PCs (center points), aggregately implied as a group in which all PCs (center points) are on a similar neighborhood framework and utilize relative gear; or a system in which the center points are shared transversely over geographically and definitively scattered structures. It can sort a petabyte of data in only two or three hours. The parallelism also gives some likelihood of recovering from midway error of servers or limit in the midst of the operation [11].

c) Error Detection and Localization

We propose a two-arrange approach to manage conduct the calculation required in the whole strategy of error revelation and confinement. At the time of error acknowledgment, there are three contributions for the error ID calculation. The first is the diagram of framework. The second is the total assembled data set D and the third is the described error outlines p. The yield of the error revelation calculation is the error set D'.

After the error design organizing and error recognizable proof, it is basic to discover the position and wellspring of the recognized error in the principal WSN graph G(V, E). The commitment of the Algorithm 2 is the primary graph of a without scale framework G(V, E), and aerrordata D from Algorithm. The yield of the calculation 2 is G'(V', E') which is the subset of the G to demonstrate the error region and source [11]

4. Results

Below fig.1 shows the Home page of Friend Book application. Run the Ubuntu platform and initialize the Hadoop in respective directing using below command as shown in fig.1.

Volume 5 Issue 11, November 2016 <u>www.ijsr.net</u> Licensed Under Creative Commons Attribution CC BY

\$ hadoop –jar\ErrorAnalysis.jar

Contraction of the second second	Contraction of the owner own	contraction of the party of				and a second s
	e -/Error					
File Edit View Search	h Terminal Help					
codinginput correcti codingostpot corrects	input docleamup.sh		Perfgl.tst results.txt	Sheudreph.jer		
catinginput corrects cotingoutput corrects Linuxhburtu:-/Errors	input decleanup.sh wypet increading jar cheed 755 decleanup.s	, arrearract, jar- tik h	Perfgl.txt results.txt			
collogiopet correct collogorpet correct LinuxMobistu:-/Errors LinuxMobistu:-/Errors	input docleanup.sh wrtput wrrtpding.dar ./docleanup.sh Is		Perfgl.txt results.txt			
codingiaput carrecti linuagubantu -/frrors linuagubantu -/frrors hame/linua/frrors /hame/linua/frror linuagubantu -/frrors linuagubantu -/frrors linuagubantu -/frrors linuagubantu -/frrors	<pre>use doctmanup.sh an vs codingingui/ vs codingingui/ pud is function chaod 755 doctmanup.sh vdcctmanup.sh suits.txt'i he such fi is</pre>	reading for arr reading for arr h le or directory		Lib Perfgl.txt Lib Perfgl.txt		
codingingut correction LinuxQubuntur-/Errors Warning: SHADOOP HOME	hadoop -jar erroding is deprecated.	rending dire intro Gar		lih Perfgl.tet		
About to start app 14/18/22 65:38:15 INFO 16/10/22 65:38:15 WARS 1 for the same.	0 util MativeCodeLoade M mapred.JobClient: Us	r: Loaded the nat e demericOptional	tive-hadiop) Marser for pi	tbrary rsing the argum	mts. Applications	should implement 1
16/10/22 05:30:16 WAR 16/10/22 05:30:16 INF 16/10/22 05:30:16 INF	N snappy.LoadSnappy: 5 D mapred.FileInputForm D mapred.JobClient: Mu	nappy native lib at: Total input p mning jab: jab li	saths to proc	ed mes : 1		
16/18/22 05:38:10 INFO 16/10/22 05:38:10 INFO 504	D util.ProcessTree: se D mapred.Task: Using	tsid exited with ResourceCalculate	exit code # orPlogin : or	g.speche.hadoop.	util.LinuxResourc	eCalculatorFlugingS
16/18/22 05:38:16 1970 16/18/22 05:38:16 1970	0 mapred.MapTask: num 0 mapred.MapTask: Lo.s	educeTesks: 1 ort.mb = 100				
🐨 🖿 dist	· Errorcorrection	📮 codingingut		nonBubantur-J	E Impot (-/error)	

Figure 1: Hadoop initialization on Ubuntu

📀 Applications Places System 🖉 🍕 Waasta - VMware Player / Inc + Vitue Machine + Hulp +	0 x	60	S-38 AM	Q linux	¢
😟 🗇 🗇 linux@ubentu: -/Error					
File Edit View Search Terminal Help					
16/10/22 05:38:18 INFO mapred.Herger: Herging 1 sorted segments					
16/18/22 65:38:16 INFO mapred.Merger: Down to the last merge-pass, with 1 segments left of	total st		262 bytes		
16/10/22 05:38:16 INFO mapred.LocalJobRunner:					
16/18/22 65:38:16 INFO mapred Task: Task:attempt local 0001 r 000000 0 is done. And is in t	he proce	ss af	committen		
16/18/22 05:38:18 INTO mapred Local JobHunner:					
16/18/22 85:38:18 INFO mapred Task: Task attempt local 0001 r 000000 0 is allowed to commit	FIDH .				1000
Tev (2007) immunitient			to ifferty	HORE/ LLIN	ac/ar
14/14/22 45:38:10 THPD maured Local JobKunner; refuce a refuce					
15/18/22 65:38:16 DFO mapred Task: Task 'attempt local pool r 000000 8' done.					
16/10/22 05:38:17 DVG mapred.JubClient: map 100% reduce 100%					
18/18/22 05:38:17 INFO mapred.30bClient: 30b complete: job local 0001					
16/18/22 05:38:17 INFO mapred.JobClient: Counters: 21					
16/16/22 05:38:17 INFO mapred.JubClient: File Input Format Counters					
16/18/22 05:38:17 DWFO mapred.3obClient: Bytes Read-906					
10/10/22 05:38:17 DNPO mapred.SubClient: File Cutput Format Counters					
16/10/22 05/30127 INTO mapres.305(110); UPUS WITCOM22					
IN JULY 22 USING THE UNIT AND AND AND A SAME LANCE. THE PROPERTY MANAGER					
16/18/22 65:38:17 INC mapred abclient: FILE BYTES WRITTIN-SASE1708					
16/18/22 05:38:17 INFO mapred 30Client: Map-Reduce Frammork					
16/18/22 85:38:17 DFG mapred. JobClient: Pap output materialized bytes=1206					
16/19/22 05:38:17 INFO mapred.JobClient: Map input records=100					
16/18/22 45:38:17 INFO mapred.JobClient: Reduce shuffle bytes=0					
16/10/22 05:38:17 INFO mapred JobClient: Spilled Records=200					
16/18/22 05:38:17 INFO mapred_JobClient: Map output bytes=1000					
16/18/22 65:18:17 DWT0 mapred 305Client: Total committed Reap usage (bytes)=321527000					
16/16/22 05:18117 INFO Mapree Job Clent On Line spect 18514					
ANY					
16/18/22 65:38:17 DED mapred 3obClient: Combine input recordsal					
16/10/22 05:38:17 INTO mapred JobClient: Reduce input records=100					
16/10/22 05:38:17 INFO mapred JobClient: Reduce input groups=1					
16/18/22 05:38:17 INFO mapred.3obClient: Combine output records=0					
16/10/22 05:38:17 INFO mapred.JobClient: Physical memory (bytes) snapshot=0					
16/18/22 05:38:17 INFO mapred.3obClient: Reduce output records=1					
16/18/22 05:38:17 DWG mapred.305Client: Virtual memory (bytes) snapshot=0					
16/18/22 05:38:17 INFO mapred.JobClient: Map output records=100					
the second state of a second state of the seco	COLUMN TWO IS NOT				

Figure 2: Map size calculation algorithm

Result of Map size algorithm using general formulas as shown in fig.2

🗿 Applications Places System 🛷 🎂 Meeta - Wee	are Mayor file + Votual Machine + Holp +	- X - 139AM	O. Brux - C
	4		
the activ triane measure tracks recomments made			
The Loss Them and the house bottometric resp			
🕹 🔐 Open + 🎬 Save 📓 🦣 Unda 👘	A 1 1 4 4		
Lines 10 Pondenting but 16	212 1 M		
1 10050506 111605005180	COLUMN TO A		
2 01200110.010012000120			
3 13313310, 111111111110			
4 30202010.111103001010			
5 01010101,000110100101			
6 11012011,111110111011			
7 10110110, 111001100110			
8 01100110,010011000110			
10 10101010. 111101001010			
11 01010101 000110100101			
12 11011011.11110111011			
13 50110110,111001100110			
14 91109119,010911000110			
15 11111110, 11111111110			
26 10101010, 111101001010			
17 01010101,000110100201			
10 10110110 111001100110			
20 01100110 010011000110			
21 31313110, 111131111110			
22 10101010, 111101001010			
23 03030301,000110300301			
24 31013011,111120111011			
25 10110110,111001100120			
25 01100110,010011000110			
28 10101010 111101001010			
25 01010101.000110100101			
39 11013911.11110111011			
31 10110110, 111001100110			
32 01380110,010011000110			
33 33333330,1133333311			
54 10381010.111103903010	Main Text -	Tabletistic Res. Last Callet	-
	Pain Ibit •	INTERVIENCE - LN1, COL2D	Ins

Figure 3: Datainput file for MapReduction

Fig.3 shows the data input file.Fig.4 presents MapReduction algorithm and evaluation time taken to complete MapReduction process.

😋 Applications Places System 😂 🔮 Upanta Vieware Player Tix • Vetur Michile • Help • 🖉 🗙 🔛 5.33 AM	G linux O
900 linux@ubuntv: -/Error	
File Edit View Search Terminal India	
There is no error in the received data.	
There is no error in the received data.	
There is no error in the received data.	
There is no error in the received data.	
16/10/22 05:39:47 INFO mapred MapTask: Starting flush of map output	
16/10/22 05:59:47 IMFO mapred Task: Task attempt local 0001 m 000000 0 1s done. And is in the process of commiting	
10/10/22 05.10/17 TM/D more discussion of the television of the state	
10/10/2 05 15 47 IMFO mapred Task: Union Response alculator Flourin ; oro anothe hadnon util LinuxResponse alculator	torFlustents1
033	
16/10/22 05:39:47 IMFO mapred.Local3obRunner:	
16/10/22 05:39:47 IMFO magined.Herging 1 sorted segments	
10/10/22 05:39:48 IAFO mapred.Merger: Down to the last merge-pass. with 0 segments left of total size: 0 bytes	
10/10/22 05/39/48 INPO magred_LOCAL360Muther;	
10/10/22 05102/06 1800 Magreel Laski laskisteept tocal 0001 r 000000 0 15 00me. And 15 in the process of committe	8
IN 10/22 05:35:46 INFO BARRENT ALL ALL ALL ALL ALL ALL ALL ALL ALL AL	
16/10/22 05:39:48 INFO mapred FileOutputCommitter: Saved output of task 'attempt local 6001 r 000000 0' to file;/	home/linux/Er
ror/correctoutput	
16/18/22 05:39:48 IM/D mapred.Local3obRunner: reduce > reduce	
10/10/22 05:39:48 IMFO mapred.Task: Task 'attempt local 0001 r 000000 0' done.	
16/18/22 05:35:48 INFO mapred_boclient: map 180% reduce 180%	
10/10/22 05/19:40 INFO Bapres 200611m1: 300 COMplete: 300 Cocal 0001 16/10/22 05 30:40 HTMC Burned Solitient: Stumbers 31	
EVENUE RELEASED AND MADE AND ADDITION FOR THE ADDITION OF THE	
16/10/22 05:39:48 INFO maured JubClimit: Bytes Read-2200	
16/10/22 05:39:48 INFO mapred.3obClient; File Output Format Counters	
16/10/22 05:39:48 IMFO mapred.JobClient: Bytes Written=0	
16/10/22 05:39:48 IMFO mapred.200Client: FileSystemCounters	
16/18/72 05:19:48 18/0 800red_30bClient: FILE BYTES MRAD-5505918	
10/10/22 05/39/48 1990 magned_cool(ient: File Bries Wallien-Sebel/44	
10/10/22 (0)122:00 Jan Weight CO., 2004 Listis: Telepinetout or namework to 16/10/22 (0) 2014 2014 Table many and 16/07 (16/07) Mine outport material (16/07) And Automatical (16/07) Aut	
10/10/22 05:35:48 TAFO macred JobClient: Map input records=100	
16/10/22 05:39:48 INFO mapred.3obClient: Reduce shuffle bytes=8	
16/10/22 05:39:48 IMFO mapred.JobClient: Spilled Records=0	
10/10/22 05:35:48 INFO mapred.JobClient: Map output bytes=0	
16/18/22 05:39:48 INFO mapred_JobClient: Total committed heap usage (bytes)=321527808	
10/10/22 05:39:48 IMPO mapred.JobClient: CPU Line spent (ms)+0	
🔤 📄 [dist] 👘 [Errorcorrection 🚘 correctinput. 🥅 linux@ubuntu/ 🔮 codeostput.txt (

Figure 4: MapReduce evaluation

Applications Places 5	iystem 🛛 😂	Hountu -	Viterare Player	Tin w Ve	tool Machine + H	10 M I	O X		5:39 AM	Q. linux	Ċ
000 linux@ubuntur-	-/Error										
File Edit View Search	Terminal H	Help:									
About to start app											
16/10/22 05:39:47 INFO 1 16/10/22 05:39:47 MARN 6	apred.JobC	lient: Use	Genericoptio	nsParser	for parsing	the argumen	ts. Appli	ations	should	implement	Tas
for the same.											
16/10/22 05:19:47 WARN	inappy Load	ISnappy: Sna	ppy native L	ibrary ex	ot loaded						
16/10/22 65 19:47 INFO	sapred File	Limpute Bune	ing intri int	I paths	нартосевя т. Ма						
16/16/22 65:39:47 INFO	til. Proces	STree: sets	id exited wi	th exit	ode 8						
15/18/22 65:39:47 INFO 1	saured.Task	ti Using Re	sourceCalcul	atorPlugi	in i orgiapac	he.hadoop.u	til.Linud	tesourc	ecalcula	torPlugin	02a1
16/10/22 05:39:47 INFO #	sapred.Map1	lask: numled	uceTasks: 1								
16/10/22 05:39:47 INFO #	sapred.Map1	lask: 10.sor	timb = 100								
16/18/22 85:39:47 INFO (sapred.MapT	rask: data b	uffer = 7969	1776/9961	4720						
10/10/22 05:19147 1NF0 1	tapred, Mapi	ask: record	Butter = 26	2144/32/1	189						
Corrected role is:											_
111000000000											- 1
5 U.U.U.											
there is no error in the	received	deta,									
there is no error in the	received	data.									
there is no error in the	received	data.									
There is an error in the	received	data.									
there is no error in the	received	data									
There is no error in the	received	data.									
There is no error in the	received	data_									
There is no error in the	e received	data.									
There is no error in the	received	data.									
there is no error in the	receives	data.									
There is no error in the	received	date.									
There is no error in the	received	data									
There is no error in the	received	date.									
Error 1s at location 8.											
Corrected code is:											
000110100101											
there is no second to the	Sec. 1	dial and									
There is no error in the	received	date.									
(dist)	2 Itreat	correction	a corrector	out.	ET linus But	burtu: -/	a codeou	DUE DAT	-		
		-			1						_

Figure 5: Error location detection and correction

Fig.5 presents location of error detected using error detection algorithm and corrected code of detected error. Table I presents the results for time taken by proposed work to do process of error detection and correction.

Table 1: Time taken by proposed work to do p	process	of
error detection and correction		

citor detection a	
Size of data	Time taken
50	1407
60	1410
70	1420
80	1425
90	1432
100	1440

Fig.6 presents comparison results between existing and proposed approaches for time taken to calculate error detection and correction w.r.t. different input file size. Blue color line is for proposed work having less time and more reliable while red line is for existing work having much more time than proposed work.

		_				
		Tin	ne tal	cen		
000	-					•
1,500	•			- 6		-
1,000 -						1
500						
0		10				
	50	60	⁷⁰ Size (80 of data	90	100
		N	=10 N	=20		

Figure 6: Comparison results between two approaches for time taken to calculate error detection and correction w.r.t. different input file size

5. Conclusion

We propose a reply in perspective of Spatial and temporal relationship to deal with the issues in the present game plan. In perspective of transient relationship, we perceive if the sensor is assault and qualities are sent. We use spatial relationship, in light of the data of association of sensor we interface those sensor values and right the errors in perceived events. By executing both these philosophies, course of action gets the chance to be extreme against errors and with error amendment limit, the requirement for retransmission and overhead in view of retransmission is maintained a strategic distance from. The execution graph is shown using the parameters Time taken as a piece of error ID and change; and size of data. The outcomes demonstrates that time adopted by our proposed strategy as MapReduce execution utilizing time taken is half not exactly the current one as without MapReduce.

References

- A. Sheth, C. Hartung, and Richard Han, "A Decentralized Fault Diagnosis System for Wireless Sensor Netowrks," Proc. IEEE Second Conf. Mobile Ad-hoc and Sensor Systems (MASS '05), Nov. 2005.
- [2] S. Mukhopadhyay, D. Panigrahi, and S. Dey, "Model Based Error Correction for Wireless Sensor Networks," IEEE Trans. Mobile Computing, vol. 8, no. 4, pp. 528-543, Sept. 2008.
- [3] K. Ni, N. Ramanathan, M.N.H. Chehade, L. Balzano, S. Nair, S. Zahedi, G. Pottie, M. Hansen, M. Srivastava, and E. Kohler, "Sensor Network Data Fault Types," ACM Trans. Sensor Networks, vol. 5, no. 3, article 25, May 2009.
- [4] G. Ateniese, R.D. Pietro, L.V. Mancini, and G. Tsudik, "Scalable and Efficient Provable Data Possession," in Proc. 4th Int'l Conf. Security and Privacy in Commun. Netw. (SecureComm), 2008, pp. 1-10.
- [5] Chi Yang, Chang Liu, Xuyun Zhang, Surya Nepal, and Jinjun Chen, "A Time Efficient Approach for Detecting Errors in Big Sensor Data on Cloud," IEEE Trans. Parallel and Cloud Systems, vol. 26, no. 2, February 2015.

- [6] J. Han, M. Kamber, and J. Pei, Data mining: concepts and techniques. Burlington, MA: Elsevier, 2012.
- [7] R. Buyya, C.S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud Computing and Emerging IT Platforms: Vision, Hype, Reality for Delivering Computing as the 5th Utility," Future Gen. Comput. Syst., vol. 25, no. 6, June 2009.
- [8] M.Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.Katz, A.Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, andM. Zaharia, "A View of Cloud Computing," Commun. ACM, vol. 53, no. 4, pp. 50-58, Apr. 2010.
- [9] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing," in Proc. 30st IEEE Conf. on Comput. andCommun. (INFOCOM), 2010, pp. 1-9.
- [10] S. Mukhopadhyay, D. Panigrahi, and S. Dey, "Model Based Error Correction for Wireless Sensor Networks," IEEE Trans. Mobile Computing, vol. 8, no. 4, pp. 528-543, Sept. 2008.
- [11]C. Liu, J. Chen, T. Yang, X. Zhang, C. Yang, K. Kotagiri, and, R. Ranjan, "Authorized public auditing of dynamic big data storage on cloud with efficient verifiable fine-grained updates," IEEE Trans. Parallel and Cloud Systems, vol. 25, no. 9, pp. 2234–2244, Sept. 2014.

Volume 5 Issue 11, November 2016 <u>www.ijsr.net</u> Licensed Under Creative Commons Attribution CC BY