A Studyof Strategies For Identification of Straggler Node In Hadoopmapreduce Environment

Ajay Kumar Bansal, DrAshu Gupta, DrManmohan Sharma

Abstract : In today’s era, Big data processing capabilities is a challenging task and bigger than this is finding the node which is performing slow(est) in the distributed environment. This paper goes through the various technological as well algorithmically development to proactively identifying the straggler nodes in the HadoopMapReduce environment. Various algorithms have been reviewed on heterogeneous cluster and with varying loads at various nodes in the cluster.

Keywords : Straggler, MapReduce, Hadoop, speculation,Data center, partitioner

Introduction

In the recent years, information technology have evolved and penetrated in every aspect of the life. Digital devices have become a major source of capturing, storing and processing the data. [12] **Fig. 1.1** showed that companies, corporate, academia and even government have started using huge amount of data to gather intelligence and make informed social, business and strategic decisions. Almost half of the organizations out of 560 believe that Big Data will help them in optimizing or improving operational efficiency and many more aspects to increase profits. Thanks to technological advancements, there are vast amounts of data emanating from the various digital sources. These huge sets of structured, semi-structured or unstructured data are considered as big data, and it's taking the world by storm.

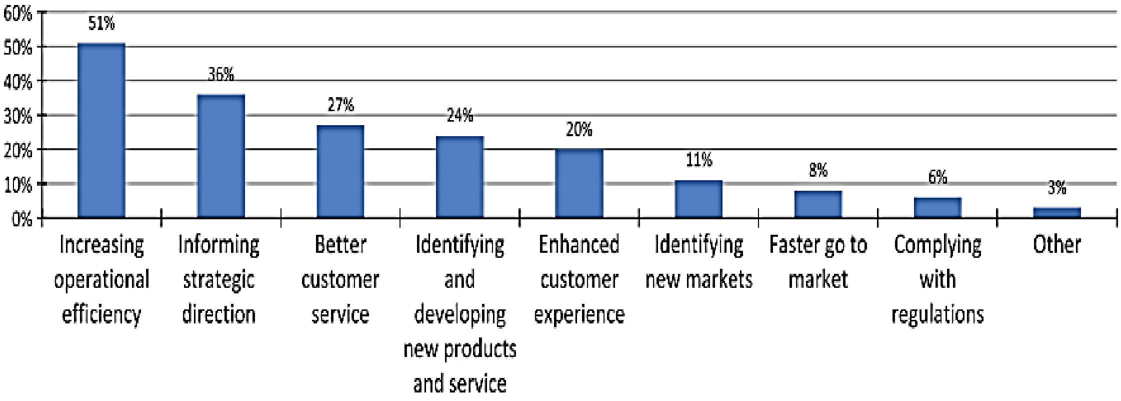

Fig. 1.1Big Data usage

The Big Data primarily is a collection of unstructured, heterogeneous,massive digital data that is difficult to handle and process by the most widely used structured data management tools and techniques like Oracle, SQL server etc.. A major challenge in front of researches and practitioners [3] Information and Communication Technology (ICT) tools is that their ability to handle such an exponentially growing data at fast pace is limited to

(1) Designing an effective system(s) to handle data.
(2) Analyzing and extracting meaningful information for decision making.

So, it becomes necessary to use more advanced or innovative tools and frameworks for effective management, organization and analysis of so much huge datasets.

There are many challenges in handling the Big data and related applications, difficulties lies in data capture, sharing, searching, analysis, storage and visualization. The real challenge lies in using the data and it is important that the data should be available on time which is accurate and complete in nature[7]. Various tools and techniques in terms of hardware and software are available right from capturing to visualization of data.
[1] **Table1**presentsthevarious frameworks, programminglanguagesand tools based on their BigDataapplications.

Table1 : Frameworks,programminglanguagesand tools forBigDataapplications

Name	**Advantages**	**Specifieduse**	**Programminglanguage**
		Batchanalysis	
GoogleMapReduce	Simple,scalable and faulttolerant	Dataprocessingonlargeclusters	Java
ApacheHadoop	Scalable,reliability,completeness, extensibility	Infrastructureandplatform	Java,Python,R,HiveQL,PigLatin
MicrosoftDryad	Ease in programmability and fault-tolerant	Infrastructureandplatform	DryadLINQ,SCOPE
ApacheMahout	Scalable,goodmaturity	Machinelearningalgorithms	Java
		Streamanalysis	
ApacheStorm	Simple,Fault-tolerant,efficient,scalable,easytouseandoperate	Real-timecomputationsystem	All
ApacheS4	Scalable, fault-tolerant,extensible and a proven model	Streamcomputingplatform	Java
ApacheSpark	Fast,general,easyto use	Engineforlarge-scaledataprocessing	Java,Scala,Python
MOA	Extensible,scalable	datastreammining framework	Java

Interactiveanalysis			
ApacheDrill	Agile,flexible,familiar	HadoopandNoSQL (SQLqueryengine)	SQL
SpagoBI	Agile,realtimeBusiness Intelligence onBigDatastreaming	BusinessIntelligence	Java
D3.js	Scalable	Interactive	JavaScript

MapReduce is a programming framework used for effectively addresses such wide variety of complex queries.MapReduce has been widely used and acclaimed by academia and industry because of its characteristics like Simplicity, capability, fault tolerance and scalability.[20]Fig. 2.1 and [18] fig. 2.2 represents the tremendous growth in the research work and scholarly articles on big data on year to year basis and by renowned publishers respectively.

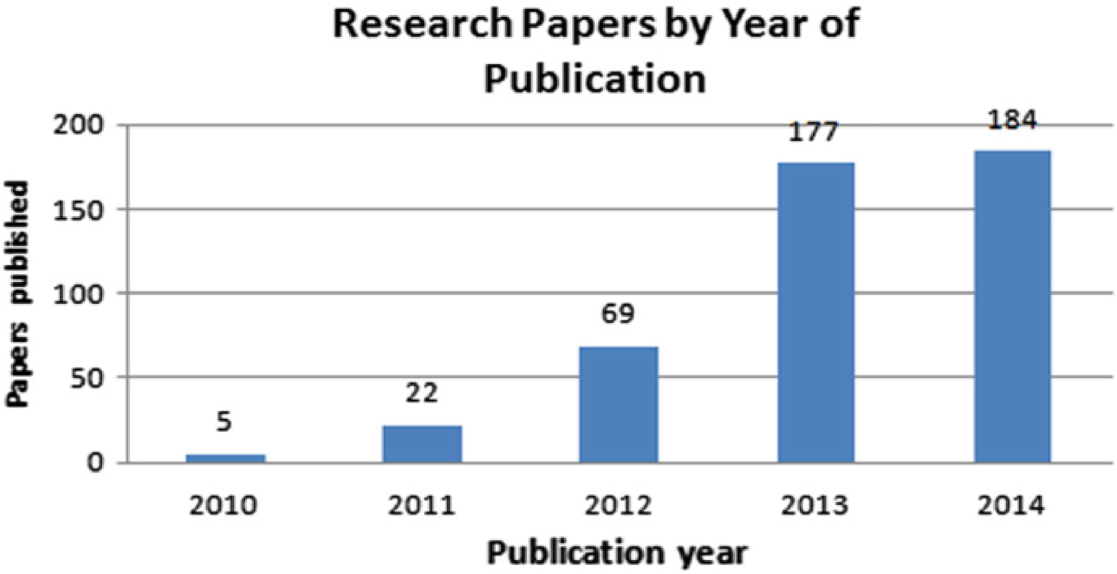

Fig. 1.2 Research papers published year wise on Bigdata

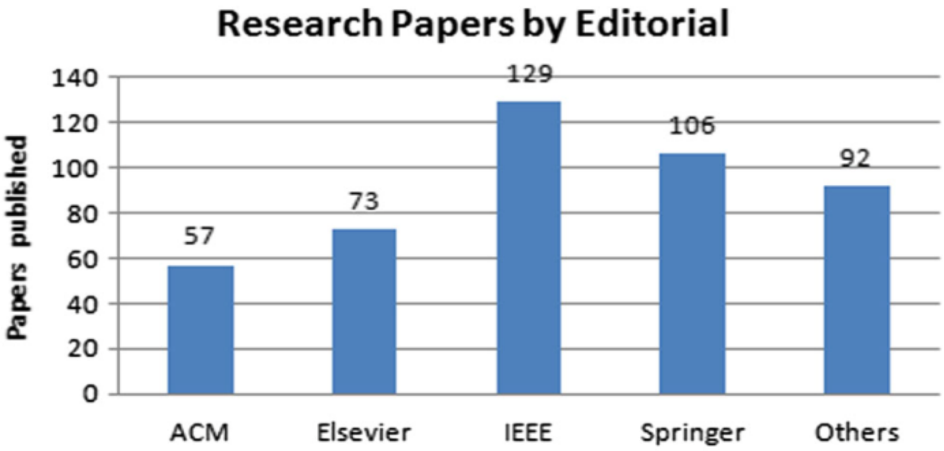

Fig. 1.3 Research papers published by different publishing house

MapReduce is a good programming model for Big Data processing and applications **(Fig. 1.4)**[22]. One node from the available node is selected as master node and responsible for scheduling the task and remaining nodes are treated as map or worker nodes. The input data are divided into chunks of almost equal size and the master node assigns chunks to Map worker nodes. Each worker node processes the assigned input chunk of data and produces key/value pairs and these key/value pairs are then stored in files for further processing. These files are called intermediate files. The master node inform the node which hold the Reduce functionality about the location of the intermediate files and the Reduce node read the data from the intermediate files and process it according to functions and the result produced stored to output files.

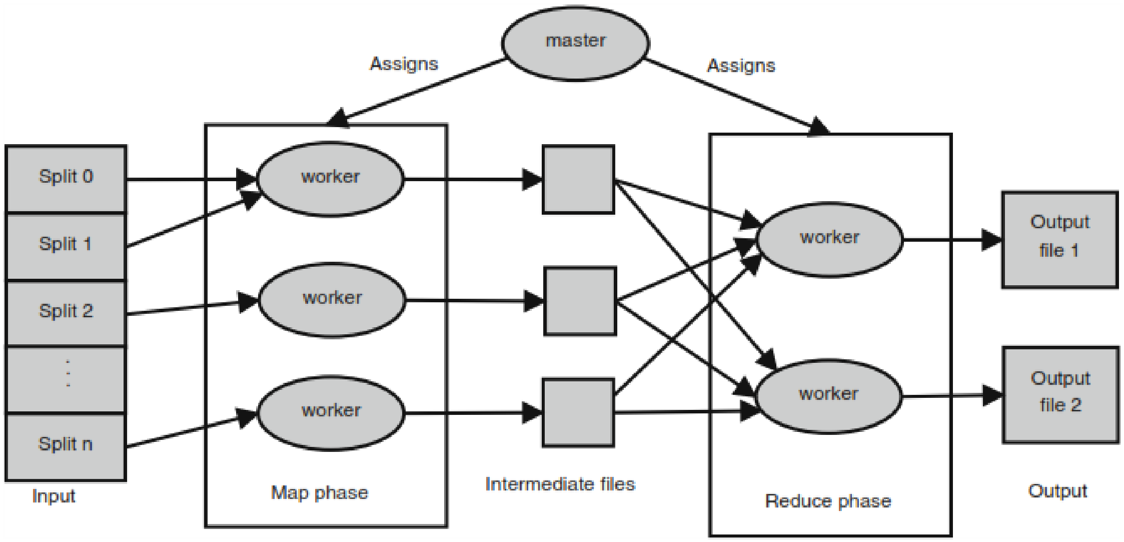

Figure 1.4MapReduce strategy

The problem lies in when one of the map workers fails or very slow in processing, the reduce function is either not able to produce the desired result in real time or fails to deliver the result. The map worker node which is very slow is called a straggler node.

Literature Analysis

GoogleMapReduce[2]isaJava-basedframeworkwhich is a batch-oriented parallelcomputingmodel distributed framework designedfor acluster ofmachines.Analytics withHadoop[5]involvesproviding files containing data to thedistributedfilesystem,andprocessing it by using parallelMapReducetool onthedata.MapReduceprocessing inHadoopcancoded using “general-purposeprogramming languages like Java or Python”, specific applications programming language R, or using Pig Latin or HiveQLto automatically generate declarative statements which looks like SQL-queries. The large collection of various popular tools or programming languages supported by Hadoop makes it more popular fordeepanalytics.

Also,extensibilityfeature of Hadoopmakes its popularity unstoppable. [8] Various core components of Hadoop like input/output dataformats, storage subsystem, scheduler, compression algorithms, data partitioner, monitoring and caching layer can be customized, reprogrammed and can be replaced very easily[19].

TheHadoop[10]systemhasvery rapidly becomea most popular tool inindustryasascalablewith high data-intensive*MapReduce*platform,andnow a days widelyusedfor many case specific applications like clickstream,Web indexing, log analysis, machine learning jobsand certain large volume informationextraction [4].*Hadoop*isusedbyvery renowned high tech companiessuchasFacebook,LinkedIn,AmazonandTwitter[9,13].

[6]The author discussed that most of the data resides on the map node itself while this data locality feature is not considered while creating a speculative tasks for the straggler node but the same is not applicable in visualized data centers. The author proposed a model by placing the data on various nodes to balance the load on various data processing nodes (map or reduce nodes). This model helps in achieving the improvised performance of data processing in a proactive manner and also helps in balancing the amount of data stored on various data nodes.

[11] The author discussed the reasons of poor performance of MapReduce implementations on heterogeneous clusters "(e.g., on a 90-node cluster that contains 10 Xeon-based servers and 80 Atom-based servers,Hadoop performs worse than on 10-node Xeon-only or 80node Atom-only homogeneous sub-clusters for many of our benchmarks)". The author proposed the *Tarazu*, a collection of tools for optimizing and improving the MapReduce performance on heterogeneous clusters. Tarazu consists of "*Communication-Aware Load Balancing of Map computation (CALB")* on map nodes which are involved in processing and "*Communication-Aware Scheduling of Map computation (CAS")* to avoidlarge volume data traffic on network and "*Predictive Load Balancing of Reduce computation (PLB)"*across the nodes.

[17] Author Discussed about the execution of MapReduce function on a cluster with homogeneous modes. The master node checked the failure node by pinging every map node periodically. If map node does not give response no response in a fixed interval of time, then the master node treat that node as a failure node and marked it as a failed node.The task performed by the map node is ignored and reset it to the initial state from it started working and the map node task is marked for rescheduling on other map nodes. The author also proposed to back up the task completed by nodes in case if the task is very close to complete its designated work.The master node backup the execution of the remaining task. This actually reduce the overall time of the task.

[14] In this paper, author focus on the tasks which are smaller in nature and interactive in nature. The impact of straggler nodes is calculated for these kinds of jobs which become a bottleneck for the analyzing the data available on data centers. The author talked about the Microsoft Bing and Facebook's production clusters. The author conclude that the impact of stragglers by using the existing latest straggler mitigation techniques, the overall task becomes on an average 8 times slower if compared with the median time to complete the task. The average job completion time is increased by 47% times due to stragglers. Author proposes to completely clone these small straggler jobs instead of waiting and speculating

after the identification of straggler with the existing algorithm. Author accepts the fact that these extra clones can create congestion for intermediate data.Author proposed a technique which cansefficiently avoids such congestion and it is called delay assignment. The evolution showed that by just using the 5% extra resources for cloning, the small jobs are increased by 36% to 46% after applying the latest state-of-the-art mitigation techniques.

[15] In this paper, the author discussed issue of statically assigning the task to workers in HadoopMapReduceenvironment and problems caused due to this. The author considered the resource utilization of every task tracker node and based on this utilization proposed an Automatic control technique to dynamically assign the task to various map workers. The results from experiment showed that this automatic assigning of tasks adjust the best optimal settings during execution.

[16]The author discussed that Speculating and creating clone for the stragglers are the existing techniques to help remove the problems of stragglers and reducing the performance in heterogeneous environment for big data processing. The author proposed to profile the results analysis and system parameters like processing power of nodes, current progress rate and utilization of various resourcesnamed as TPC-H. According to these parameters, the amount of task is adjusted dynamically. The experiment results are benchmarked with the existing HadoopMapReduce platform and identified that overall performance is improved by 20-30% and the stragglers are reduced by 35-88% as compared to existing techniques.

[23]Theexisting techniques fails to deliver in heterogeneous clusters and unbalanced task allocations to nodes with invariable computing powers. In this paper, the author proposes a technique named MrHeter in which MapReduce task is divided into two parts as mapshuffle and MapReduce stage. Author added that different optimization model are then used for these two different asks and tasks are allocated keeping in mind the processing capabilities of individual node. Experiment shows that MrMeter and D-MrMeter can reduce the overall completion time of the task from 30% to 70% heterogeneous environment as compared to existing Hadoop computing model. The author concluded that the proposed model works better especially in heavy workload and wide variations in computing powers of the nodes.

[24] The author has implemented an algorithm “maximum node hit ratepriority algorithm (MNHRPA)”. This is done by tracking node information in real timeby using exisiting monitoring module of HadoopMapReduce. [21] The algorithm suggests the dynamic adjustment of load to various nodes of different computing powers in run time keeping in mind the existing work load on the node. The experimental results show completion time of the task is reduced very drastically and helps in optimizing balance of work load of the cluster as compared to Hadoop's defaultalgorithm.

[25] In this paper, the author identified the ‘slow’ node in the Hadoop environment and speculates the task i.e. redundant task on other available node so that reliability is ensured. This mechanism accountably reduces the no. of tasks that are unnecessary speculated and uses the system resources redundantly. The proposed solution also able to pin point the node which can fail in a very early stage and avoid the dealy in job execution time.

[26] Author discussed the problems of increase in the job completion time and wastage of resource in handling stragglers. The reason is not proactively identifying stragglers rather focusing on reactive measures after a node is identified as stragglers. Author proposed a prognostic system based on predictive models. In first phase, the author uses the distributed machine learning model to predict the straggler nodes before allocating the task to a node and this way guiding the task scheduler to assign task. The proposed model is compared with Hadoop's native scheduler.

Conclusion

It has been observed after studying the various papers on stragglers node that there is no particular model which can effectively identify the straggler node in the early stage. Identifying the straggler node is a challenge in distributed environment and speculating the straggler on another node is another challenge. Different strategies are used for heterogeneous node vs homogeneous nodes, smaller jobs vs longer jobs or varying loads on worker nodes. No single strategy is effective in all environments. Further, the changes in hardware as well as software architecture in distributed or cloud environment also leads to innovate/modify the new/existing technique or algorithm.

6.1 References

[1] Dean J, Ghemawat S et al (2008) "MapReduce: Simplified Data Processing on Large Clusters", COMMUNICATIONS OF THE ACM 51(1) p. 107-113.

[2] DeanJ,Ghemawat S(2008) "MapReduce:SimplifiedDataProcessingonLargeClusters", ACM51(1) p. 107–113.

[3] Gordon Bell, Tony Hey, Alex Szalay (2009) "Beyond the data deluge", Science 323 (5919) p. 1297–1298.
[4] Tony Hey, Stewart Tansley, Kristin Tolle (2009) "The fourth paradigm: data-intensive scientific discovery", Microsoft Research.
[5] WhiteT(2009) "Hadoop :thedefiniteguide",1stedn.OReillyMediaInc,Sebastopol.

[6] Jiong X, Shu Y, Xiaojun R, Zhiyang D, Yun T, James M, Adam M, Xiao Q (2010) "Improving MapReduce Performance through Data Placement in Heterogeneous Hadoop Clusters", IEEE

[7] HerodotouH,LimH,LuoGetal(2011) "Starfish: aself-tuningsystemforbigdataanalytics", Proceedingofthe5thbiennialconferenceoninnovativedatasystemsresearch(CIDR 11),p.261–272.

[8] BorkarV,CareyMJ,LiC(2012) "Inside"BigDataManagement": ogres,onions,orparfaits?",ProceedingofEDBT/ICDTjointconference.ACM

[9] LinJ,RyaboyD(2012) "Scalingbigdatamininginfrastructure: thetwitterexperience", SIGKDD Explorations14(2) p. 6–19.

[10] BegoliE,HoreyJ(2012) "Designprinciplesforeffectiveknowledgediscoveryfrombigdata", ProceedingofthejointworkingIEEE/IFIPconferenceonsoftwarearchitecture(WICSA)andEurop ean conferenceonsoftwarearchitecture(ECSA),p.215–218.

[11] Ahmad F, Chakradhar S, Raghunathan A, Vijaykumar T N (2012) "Tarazu: Optimizing MapReduceOn Heterogeneous Clusters", ACM p. 61-74.

[12] James M, Michael C, Brad B, Jacques B, Richard D, Charles R, Angela HB (2012) "Big data: The Next Frontier forInnovation, Competition, and Productivity", McKinsey Global Institute.

[13] O'DriscollA,DaugelaiteJ,SleatorRD(2013) ""BigData",Hadoopandcloudcomputingingenomics", JBiomedInform46(6) p. 774–781.

[14] Ananthanarayanan G, Ghodsi A, Shenker S, Stoica I (2013) "Effective Straggler Mitigation: Attack of the Clones", 10th USENIX Symposium on Networked Systems Design and Implementation, p. 185-198.

[15] Weina W, Kai Z, Lei Y (2013) "A Throughput Optimal Algorithm for Map Task Scheduling in MapReduce with Data Locality", Performance Evaluation Review 40(4) p. 33-42.

[16] Xia Z, Kai K, YuZhong S, Yin S, Minhao X, Tao P (2013) "Insight and Reduction of Map Reduce Stragglers in Heterogeneous Environment", IEEE.

[17] DemchenkoY,GrossoP,deLaatC,MembreyP(2013) "Addressing BigDataIssuesinScientific DataInfrastructure", InternationalConferenceonCollaborationTechnologiesandSystems(CTS). IEEEComputerSociety.

[18] TaliaD (2013) "Cloudsforscalablebigdataanalytics",Computer46(5) p. 98–101

[19] Katal A, Wazid M, Goudar RH (2013) "Big data: issues, challenges, tools and good practices", Sixth international conference on contemporary computing (IC3), p. 404–409

[20] Zhang X, Liu C, Nepal S et al (2014) "A hybrid approach for scalable sub-tree anonymization over big data using MapReduce on cloud", Journal of Computer and System Sciences 80(5) p. 1008–1020

[21] ChenPCL,ZhangC-Y(2014) "Data-intensiveapplications,challenges,techniquesandtechnologies : asurveyonbigdata",InfSciElsevier275 p. 314–347.

[22] GrolingerK,HayesM,HigashinoWAetal(2014), "ChallengesforMapReduceinbigdata", Proceedingofthe2014IEEEworldcongressonservices(SERVICES) p.182–189.

[23] Zhang X, Wu Y, Zhao C (2016) "MrHeter: improving MapReduce performance in heterogeneous environments", Cluster Computing 19(4) p. 1691-1701.

[24] Wei Y, ChunLin L, ShuMeng D, XiJun M (2016) “An Optimization Algorithm for HeterogeneousHadoopClusters Basedon Dynamic Load Balancing”, 17th International Conference on Parallel and Distributed Computing, Applications and Technologies p.250-255

[25] J. Wang, T. Wang, Z. Yang, N. Mi and B. Sheng (2016) "eSplash: Efficient speculation in large scale heterogeneous computing systems," *2016 IEEE 35th International Performance Computing and Communications Conference (IPCCC)*, Las Vegas, NV p. 1-8

[26] Deshmukh S., Aghav J., Rao K.T., Rao B.T. (2017) “Avoiding Slow Running Nodes in Distributed Systems”, Computer Communication, Networking and Internet Security. Lecture Notes in Networks and Systems 5 p. 411-420