JOURNAL OF CRITICAL REVIEWS

ISSN- 2394-5125 VOL 7, ISSUE 03, 2020

A Scalable Methodology for Data Stream Analytics in Real-Time Internet of Things (IoT) Applications

B. Bhaskar Reddy[1], P. Imran Khan[2], Dr. B. Dhananjaya[3]
Associate Professor[1,3], Assistant Professor[2]
Department of Electronics and Communication Engineering
Bheema Institute of Technology and Sciences, Adoni-518301.[1,2,3]

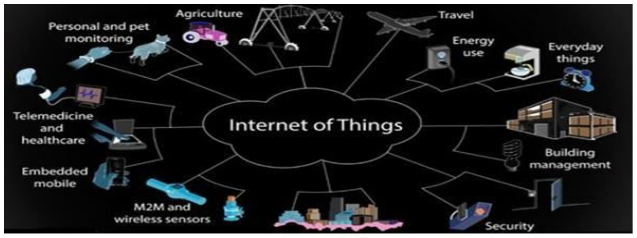
Abstract*: Analytical tools are being used in a variety of ways by the Internet of Things (IoT). Several applications strive to acquire data from various contexts, which may be homogeneous or heterogeneous, but the work of collecting, processing, storing, and analysing the data that is being collected from diverse environments is still difficult. Big data and untrusted networks make providing security for these things difficult. In the ever-expanding network, there may be various non-trivial problems with data gathering, data-efficient processing, analytics, and security. To achieve the aforementioned outcomes, large-scale sensor deployments are required in each of the examples provided. IoT devices may collect sensitive private information, which raises the problem of privacy exposure when sensors constantly transfer data to the cloud for real-time usage. A two-layer approach or paradigm for evaluating IoT data from numerous applications is proposed in this context. The initial layer serves mostly as a service-oriented interface for ingesting data from various settings. It is up to the second layer to ensure the safety of all the data it is in charge of. Open source components are used to implement the proposed solutions.

Keywords: **Data, Data Stream, Spark, Analytics, IoT.**

I. INTRODUCTION

Advances in communication technology have made it possible for everyday objects to be equipped with software and hardware components such as microcontrollers and transceivers to provide digital communication between nodes of various networks with the users. This makes the Internet Of Things a part of our daily lives. This is a fast-evolving technology in the context of current wireless telecommunications, with the primary goal of connecting everything on the planet [1]. The Internet of Things (IoT) concept increased the allure and universality of the internet even more. As a result of the development of smart apps, as well as the simplicity with which these applications may be accessed, IoT can be a powerful communication technology. To deliver services to individuals, businesses, government agencies, and other entities, the IoT applications created may make use of large amounts of data and produce a wide range of data streams.

domains. Indeed, the Internet of Things concept uncovers applications in several fields termed heterogeneous domains, such as industrial automation, health care automation, residential automation, and numerous others. [2]. Cloud computing is an advanced computing paradigm that allows users to use a shared resource pool of cloud resources, such as storage, access, processing, and applications, in an on-demand way. As an example, IoT Sensors first gather and transmit their information to gateways which then transmit it to the cloud for storage, processing, and analysis and it then transmits the data to the user on demand. This is an example of cloud computing's integration with IoT and its cloud computing capabilities, such as data retransmission. If data transmission fails at any point, it is retransmitted to the designated recipients until it is successfully delivered. In both academia and business,



cloud computing is attracting a lot of interest. Smart initiatives may be enabled by connecting with big data and analytics, which is why many IoT apps may not only focus on controlling various things but also on mining the data acquired from IoT devices. Sensor-equipped conventional protocols like MQTT-Message Queue Telemetry Transport Protocol, XMPP, and others are often used by IoT devices to gather data. Internet of Things research shows that by 2020, the number of things or devices interconnected to IoT is expected to reach 50 billion, as shown in Fig. 1. With IoT, many opportunities have been created that can help increase revenue, reduce costs, generate a large amount of data, and all the other things besides. [4]

Fig. 1. Services of IoT

To reap the advantages of the Internet of Things, businesses must develop a platform that can be utilised to gather, manage, and analyse large amounts of data. The sensors utilised in this platform create big data, which can be used efficiently and cost-effectively, and must be scalable to be useful.

ISSN- 2394-5125 VOL 7, ISSUE 03, 2020

Integration of heterogeneous datasets is critical in this context since big data may create enormous volumes of data.

The above-mentioned advantages of IoT may be achieved via the use of specialised data analytics tools established by various sectors, and this combination of disparate disciplines may open up a whole new field of study. In the ever-expanding network, there may be various non-trivial problems with data gathering, data-efficient processing, analytics, and security.

The following is a summary of the paper's contributions:

• Literature review on the Internet of Things.

• Providing two architectures for data collection and analysis.

• Big data management in IoT is made possible by the use of open source components.

• IoT and Big Data integration

IoT and Big Data Integration: Key Requirements

• Identify the unsolved problems in IoT research and the potential for big data analytics in the field.

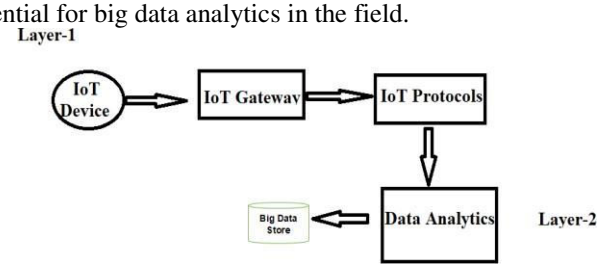


Fig. 2. Proposed Architecture

Figure 2 depicts a two-tiered design in this context. It is the job of the first layer to gather data from various sources and transfer it to the second layer of analytics, where it is processed to extract knowledge data necessary for real-time IoT applications.

Sections one through three of this paper are as follows: IoT and big data research is the emphasis of Section II Related Work. The initial layer of data collection is shown in detail in Section III. Our suggested design is discussed in Section IV, which focuses on analytics. Next, in Section V, we'll explain how we plan to execute our design. Sections VI and VII discuss the conclusion and future work. Heterogeneous data: data of many sorts and formats are some of the terminology used in this study. To handle massive amounts of data in a distributed manner, the MapReduce programming concept and implementation are used. Map and Reduce are the framework's two functions for filtering, sorting, and summarising.

When a customer's identification is verified, the process of authentication has taken place. A person's or object's uniqueness is verified via authentication, rather than through evidence of identification, which is the act of expressing that uniqueness.

Apache Hadoop For problems requiring vast volumes of data and computing, a network of numerous computers may be used in conjunction with a combination of ASCII text file software package tools, known as Hadoop. Distributed computing is supported by this software MapReduce is a programming technique for storing and processing large amounts of data.

Apache Spark Cluster-computing framework Spark is open source and distributed. Data parallelism and fault tolerance are built-in to the Spark programming interface.

II. RELATEDWORK

I. MapReduce: Google's MapReduce architecture was developed to analyse the company's massive amounts of data. MapReduce and its open-source components, such as Hadoop, it has been extensively utilised to enable enormous computations across vast information sets or data sets. Hadoop and other open-source data analysis frameworks like MapReduce are based on MapReduce. MapReduce is often used in social media and e-commerce to analyse huge datasets [5].

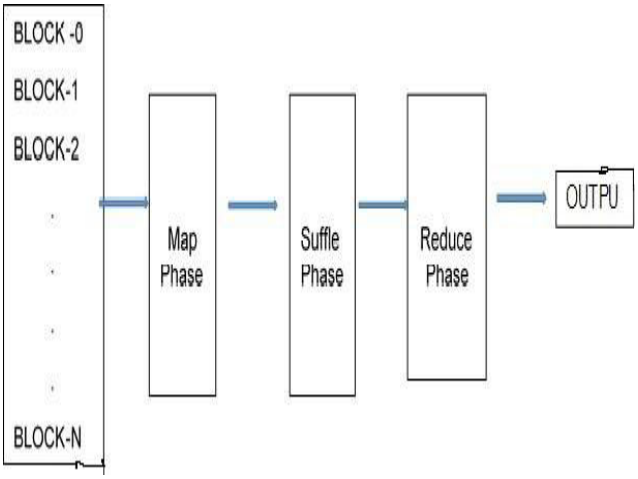


Fig. 3. MapReduce

Shuffle is a specific phase for sorting data, and MapReduce is a high-level data analytics methodology for grouping all of the data created by its stages. The popularity ofMapReduce was fueled by its simple programming interface and high performance. Because MapReduce is a distributed processing system in terms of its behaviour, Google employs the open-source Hadoop component to implement it [6]. There are three phases in the MapReduce implementation shown in Figure 3: the Map Phase accepts input from blocks for mapping and sends it to the second phase for Shuffle, and it may also send it to the third phase Reducer, which creates output. Figure 4 depicts an explanation and visual depiction of Hadoop.

ISSN- 2394-5125 VOL 7, ISSUE 03, 2020

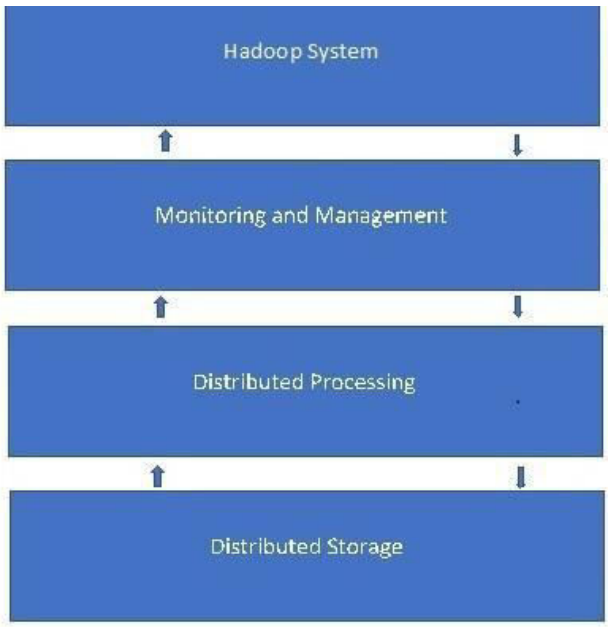


Fig. 4. Hadoop

II. MapReduce Authentication: Data in MapReduce is partitioned and stored on a network of dispersed and shared nodes, and once a job has been run, it may be executed several times, making authentication in MapReduce a big difficulty from the standpoint of data analytics. [7]

III. MAPREDUCE FRAMEWORK FORIOT:

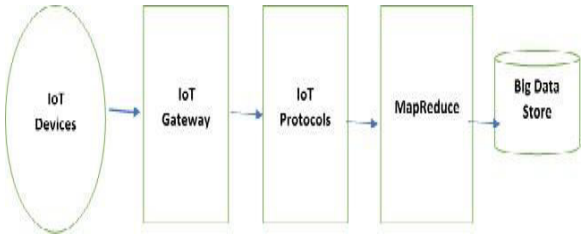


Fig. 5. Analytics Framework for IoT

Using MapReduce and Hadoop for data analytics has resulted in impressive results. Even while MapReduce is a wonderful tool for analysing massive amounts of data, it has a few drawbacks, such as the fact that every iteration requires the development of a new Mapper and Reducer and the repeated reading of the same disc. IoT data analytics fall into two major categories: batch processing and event processing. MapReduce is utilised for batch processing. Several applications may benefit from data analytics in the Internet of Things. These include public administrations, healthcare, individuals, businesses, and a variety of other sectors. In Hadoop, an Apache open-source system developed in Java, huge datasets may be processed over a large number of PCs using simple programming models that are distributed throughout the network. It is in this arena that the Hadoop system application operates, where computing power is scattered among a large number of computers. Since Hadoop is designed to grow, up as the number of machines grows, so does the amount of computation and capacity that can be performed on each one of them. [8]

IV. DATACOLLECTIONATTHEFIRSTLAYER

Data may be collected using a wide variety of sensors. When it comes to acquiring raw data, sensors are an absolute must. IoT's "structure squares" are vibrant articles like this one. More information is created and shared when sensors are mounted on a wide variety of objects. Sensors [9] helped to collect the ever-expanding stream of information. This data is used for both on- and offline information investigations, with the former being the primary focus. There is a subset of applications in which data is generated that has value even if it is not handled sequentially. An artificial intelligence (AI) computation was needed to connect data from many bright objects to prepare this naturally unrefined information. You may assume the same thing if you do a test computation that connects data from distant sensors to a cloud-based handling focus. Because it can handle such massive amounts of data, distributed computing is gaining popularity right now [10].

Sensor systems, interpersonal groups, and automobiles are only some of the sources of information that may be obtained in this manner. Concerns about the security of data sent to the cloud server farm from the aforementioned sources may still be addressed by further expansion. A standard architecture is required to aid in the collection of data from sensors and storage in the cloud. System norms are the bedrock of any correspondence framework that aims to provide a specific level of

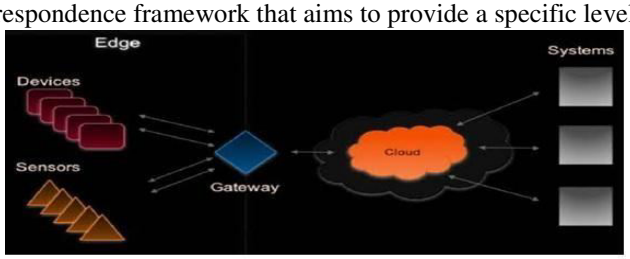


Fig.6. IoT Gateway

Quality of Service (QoS) for each communication request. Despite its topological structure and correspondence being progressively uncommon, the distant framework system is becoming dynamically fit as new development advances at a rapid rate of improvement. Furthermore, ZigBee mastermind has the related problem to other remote advancements. It is a remote development, which is a remote framework with little effort. Due to IEEE 802.15.4, the ZigBee Alliance developed a remote framework show. By looking at it from a Stack show viewpoint, ZigBee is an integral part of the framework, with its parts and varied levelled structure providing an exceptional base for the development of a powerful application framework.

ISSN- 2394-5125 VOL 7, ISSUE 03, 2020

Despite this, it is expected that an appropriate remote framework improvement system in light of embedded structure may shorten the improvement cycle, as well as reduce the improvement costs and increase structure quality. [11]

Figure 6 is an example of an IoT gateway used to gather data. It includes essential components such as sensors, devices, gateways, clouds, local databases, and systems. Each component's role is outlined below, with sensors providing raw data it appears as values by performing certain actions from the devices, therefore the gateway functions as a bridge between devices and the cloud. Here, the cloud plays a vital role, taking data from the gateway, processing and transforming it, and sending it to the local dB for local storage as requested by the systems.

Protocols for the Internet of Things: It is possible to develop internet of things applications using several protocols. Protocols are categorised depending on their application and use as follows:

MQTT-Message Que Telemetry Transport, CoAP-Constrained Application Protocol, AMQP-Advanced Message Que Protocol, HTTP-Hyper Text Transfer Protocol 6LOWPAN, ZigBee, Bluetooth LE, RFID, NFC, and SigFox are all examples of communication protocols.

Protocols such as IP Security, Wireless Hart, etc.

V. DATAANALYTICSATTHESECONDLAYER

When it comes to processing large amounts of data, Google MapReduce is an ideal solution. Figure 3 depicts a MapReduce project's high-level workflow. Parallelization is included in the MapReduce framework by default, thus application designers just have to provide a guide and a lower amount of labour. The simplicity of MapReduce's simple programming interface and its lightning-fast performance in a wide variety of applications have earned it widespread acclaim. While MapReduce is widely used for data analytics, there are several drawbacks to this approach, as we've discussed above. When it comes to large-scale computation, MapReduce provides an established approach for conducting large-scale disseminated calculations. However, the framework is constrained, as shown by the inefficiency of progressive preparation. It refers to the apps that are constantly updating their information and calculating the contribution to the request to provide yield over time. In this method, there are possible copy computations. The problem is that MapReducecannot differentiate such duplicate computations and speed up operations. As a result of this perspective, the usage of Hadoop resulted in a significant advantage. Distributed computing technologies may solve any distributed issue using a variety of algorithms; MapReduce is one of these methods [12]. To keep track of network health in real-time, we need a big data platform like Spark Streaming that can handle a large quantity of data efficiently while being resilient enough to survive a breakdown without halting the monitoring process altogether. In data warehouses, It is possible to efficiently handle vast amounts of data using frameworks like Hadoop and Spark. Hadoop and its open-source MapReduce paradigm [13], for example, are extensively used by the huge data analytics community because of their simplicity and easy programming [14]. Yet, since Hadoop's intermediate data is kept on disc (which often has poor I/O performance), algorithms needing many iterations would suffer greatly in terms of production. Hadoop's speed may be improved by using in-memory computing technologies like Apache Spark [15]. Since Spark's intermediate data is cached in memory in Resilient Distributed Datasets (RDD), it can process data significantly more quickly than Hadoop [16]. Big data platforms have been employed by several offline Internet monitoring systems to improve their processing speed. Online network monitoring, on the other hand, has received just a few investigations. Offline data analysis may benefit from the execution of Hadoop and Spark. execution is used to process huge datasets when several data items may be batched together for efficiency [17].

Tools and Technology Used

Sno	Tools and Technology Used	
1	Spark	Open-source distributed general purpose cluster framework
2	Cassandra	Database
3	Kafka	Used for real-time stream analysis
4	JDK	The Java Development Kit
5	Maven	Automation tool
6	Zookeeper	A service for DS
7	Spring Boot	Stand-alone App Creator

The input file must be available at all times throughout the computation so that all of the information may be processed concurrently. It's a stream analytics issue where the input is an infinite stream of knowledge. Online Internet traffic monitoring even thoughMapReduce does not allow stream processing, micro-batching may be used to partly process streams. In this case, the data in the stream is seen as a series of smaller batch sizes. [18] A piece of information is provided to the batch system at regular intervals to be processed [18]. The Spark Streaming library, for example, may be used to support this system. As a result of this, several systems are specifically intended to handle large data streams, such as Apache Storm and S4. One or more input streams may be processed by each node and a set of output streams can be generated. Arrived data will be handled immediately.

ISSN- 2394-5125 VOL 7, ISSUE 03, 2020

Fig.7.Traffic Scenario

Hours and Respective Value:

SNO	Hours	Value
1	0h	5.176531
2	1h	0.347362
3	2h	0.99331
4	3h	-0.31845
5	4h	-0.28967
6	5h	0.595623
7	6h	0.683265
8	7h	0.812636
9	8h	0.523157
10	9h	0.689745
11	10h	0.435689
12	11h	0.235479
13	12h	0.785217
14	13h	0.325416

VI. IMPLEMENTATION

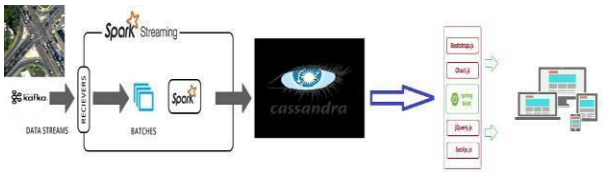


Fig.8.Spark Data Analytics

In this system, we've used a real-time traffic situation for data analytics. One of the major issues that many industrialised nations confront is the monitoring of traffic. As described in the preceding section, Apache Spark has been used to analyse traffic data in this layer.

Databaseforstoringdata,theoverallscenarioisrepresented in the belowfigure.

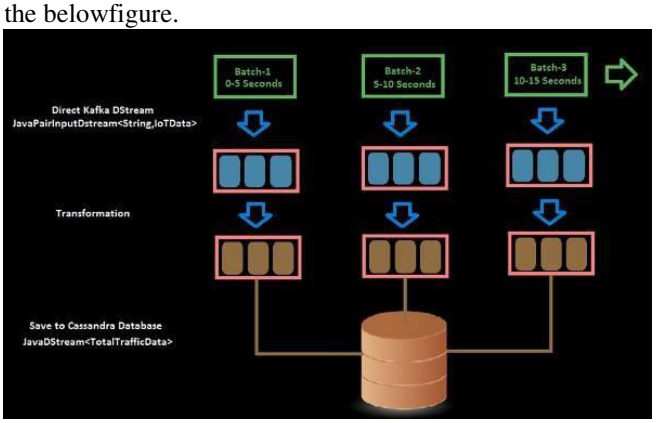
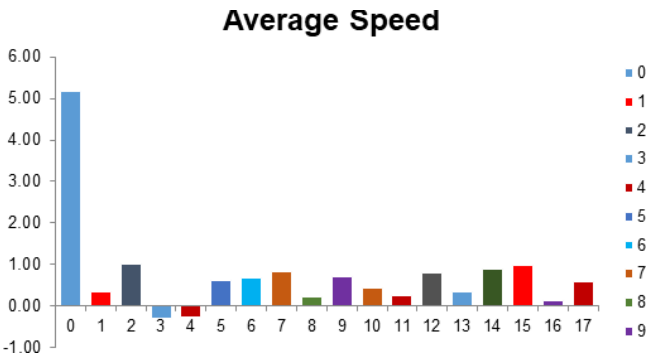


Fig.9. Data Processing and Storage

Results from spark are displayed in this table, which gives hours, the vehicle speed in hours, and a graph of average vehicle speed.

Results:



CONCLUSION

Real-time traffic management may be made more effective, scalable, and consistent by using the approach we provide in this article, which shows how to collect, analyse, and integrate heterogeneous data sources. A two-tiered approach to data collecting and analysis has been used here.

VII.

VIII. REFERENCES

1. L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," Comput. Netw., vol. 54, no. 15, pp. 2787–2805,2010.
2. Andrea Zanella ,NicolaBui,AngeloCastellani,LorenzoVangelista and Michele Zorzi " Internet of things for Smart Cities,"., vol. 1, no. 1, pp. 2327–4662,2014.
3. Halah Mohammedanl-kadhim and hamed s.al-raweshidy"EnergyEfficientandReliableTransportofData in Cloud-Based IoT,"., vol. 7, pp. 2169–3536,2019.

ISSN- 2394-5125 VOL 7, ISSUE 03, 2020

4. E. Ahmed et al., ''The role of big data analytics in Internet of Things,'' Comput. Netw., vol. 129,pp. 459–471, Dec. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128617302591
5. Nan Zhu, Xue Liu*, Jie Liu, and Yu Hua,"Towards a Cost-Efficient MapReduce: Mitigating Power Peaks for Hadoop Clusters,"., vol.19, no. 1, pp.24-32 1007–0214,2014.
6. YaxiongZhao' JieWu, and Cong Liu, "Dache: A DataAware Cachingfor Big-Data Applications Using the MapReduce Framework,"., vol. 19, no. 1, pp.39-50 1007–0214,2014.
7. Ibrahim lahmer and ningzhang, "Towards a virtual Domain Based Authentication on MapReduce," vol. 4, 2016,2558456.
8. Available Online:www.wikipedia.org/wiki/Apache_Hadoop.
9. "Fuzzy Assisted Event-Driven Data Collection from Sensor Nodes in Sensor-Cloud Infrastructure," Cluster, Cloud and Grid Computing (CCGrid), S. S. Bhunia, J. Pal and N. Mukherjee," 2014 14th IEEE/ACM International Symposium on, Chicago, IL, 2014, pp. 635-640
10. "A Dynamic Key-Length-Based Efficient Real-Time Security Verification Model for Big Data Stream." D. Puthal, S. Nepal, R. Ranjan, and J. Chen." DLSeF: ACM Transactions on Embedded Computing Systems (TECS), Vol. 16, no. 2, pp. 51,2016.
11. " Sensors Data Collection Architecture on the Internet of Mobile Things as a Service (IoMTaaS) Platform", PrasenjitMaiti, BibhudattaSahoo,Ashok Kumar, SuchismitaSatpathy, Conference Paper · February 2017 DOI:10.1109/I-SMAC.2017.8058245.
12. "Spark-Based Large-Scale Matrix Inversion for Big Data Processing: JUN LIU1, (Member, IEEE), YANG LIANG1, AND NIRWAN ANSARI2, (Fellow, IEEE)- Digital Object Identifier 10.1109/ACCESS.2016.2546544.
13. http://hadoop.apache.org/
14. "Trends in big data analytics"- vol. 74, no. 7, pp. 2561–2573, 2014. K. Kambatla, G. Kollias, V. Kumar, and A. Grama, J. Parallel Distrib. Comput.
15. Apache Spark,http://spark.apache.org/
16. " Cluster computing with working sets, in Proc.2nd USENIXConf. HotTopicsinCloudComputing, "Boston, MA, USA, 2010, M.Zaharia,M.Chowdhury,M.J.Franklin,S.Shenker,and I. Stoica,Spark.
17. "Data-intensiveapplications, challenges, techniques and technologies: Asurveyonbigdata"C.L.P.ChenandC.Y.Zhang,Inf.Sci.,vol.275,pp. 314–347,2014.
18. "Towardsreal-timeandstreamingbigdata,Computers,vol.3,no.4,pp. 117–129, 2014.S.Shahrivari,Beyondbatchprocessing