Scientific Journal Impact Factor (SJIF): 1.711

International Journal of Modern Trends in Engineering and Research

IJMTER

ISSN (ONLINE) : 2349 - 9745 ISSN (PRINT) : 2393 - 8161

BIG DATA PROCESSING WITH PRIVACY PRESERVING USING MAP REDUCE ON CLOUD

Kaushlendra Singh Parihar[1], Rakesh Pratap Singh [2], Uttam Kumar[3], Mysore Jayakrishna Yogesh[4]

[1,2,3,4] Computer Science & Engineering, The National Institute of Engineering

Abstract— A large number of cloud services require users to share private data like electronic health records for data analysis or mining, bringing privacy concerns. Anonymizing data sets via generalization to satisfy certain privacy requirements such as k-anonymity is a widely used category of privacy preserving techniques. At present, the scale of data in many cloud applications increases tremendously in accordance with the Big Data trend, thereby making it a challenge for commonly used software tools to capture, manage, and process such large-scale data within a tolerable elapsed time. As a result, it is a challenge for existing anonymization approaches to achieve privacy preservation on privacy-sensitive large-scale data sets due to their insufficiency of scalability. In this paper, we propose a scalable two-phase top-down specialization (TDS) approach to anonymize large-scale data sets using the Map-Reduce framework on cloud. In both phases of our approach, we deliberately design a group of innovative Map-Reduce jobs to concretely accomplish the specialization computation in a highly scalable way. Experimental evaluation results demonstrate that with our approach, the scalability and efficiency of TDS can be significantly improved over existing approaches.

Keywords— k-anonymity, anonymization, cloud applications, privacy preservation, two-phase top-down specialization, Map-Reduce, highly scalable,

I. INTRODUCTION

Cloud computing provides massive computation power and storage capacity via utilizing a large number of commodity computers together, enabling users to deploy applications cost-effectively with- out heavy infrastructure investment. Privacy is one of the most concerned issues in cloud computing, and the concern aggravates in the context of cloud computing although some privacy issues are not new Personal data like electronic health records and financial transaction. Data anonymization has been extensively studied and widely adopted for data privacy preservation in noninteractive data publishing and sharing scenarios. In our research, we leverage Map-Reduce, a widely adopted parallel data processing framework, to address the scalability problem of the top-down specialization (TDS) approach for large-scale data anonymization. The TDS approach, offering a good tradeoff between data utility and data consistency, is widely applied for data anonymization.

II. RELATED WORKS AND PROBLEM ANALYSIS

The recent emergence of cloud computing has drastically altered everyone’s perception of infrastructure architectures, software delivery and development models. Projecting as an evolutionary step, following the transition from mainframe computers to client/server deployment models, cloud computing encompasses elements from grid computing, utility computing and autonomic computing, into an innovative deployment architecture. This rapid transition towards the clouds, has fuelled concerns on a critical issue for the success of information systems, communication and information security. From a security perspective, a number of unchartered risks and challenges have been introduced from this relocation to the clouds, deteriorating much of the effectiveness of traditional protection mechanisms. As a result the aim of this paper is twofold; firstly to evaluate cloud security by identifying unique security requirements and secondly to attempt to present a viable solution that eliminates these potential threats. Anonymization algorithms typically aim to satisfy certain privacy definitions with minimal impact on the quality of the resulting data. While much of the previous literature has measured quality through simple one-size-fits-all measures and argue that quality is best judged with respect to the workload for which the data will ultimately be used. This article provides a suite of anonymization algorithms that incorporate a target class of workloads, consisting of one or more data mining tasks as well as selection predicates. y. The first extension is based on ideas from scalable decision trees, and the second is based on sampling. A thorough performance evaluation indicates that these techniques are viable in practice. Experiments on real-life data demonstrate that the anonymization algorithms can effectively retain the essential information in anonymous data for data analysis and is scalable for anonymizing large datasets. Handling of the large scale data sets are very difficult. Here it using the distribute anonymization and centralized anonymization to provides the privacy on cloud. Handling of the large scale data sets are very difficult.

III. PROPOSED SYSTEM

A. Privacy Preserving

The approach is centralized top-down approach. It’s does not have the ability for handle the large scale datasets in cloud**.**The scheduling mechanism called Optimized Balanced Scheduling(OBS) is used for anonymization. Here the OBS means individual dataset have its own separate sensitive field. It analyze each and every data set sensitive field and give priority for their sensitive field. Then apply anonymization on this sensitive field only depending upon the scheduling.

B. Two Phase TDS

Two Phase TDS approach is used to conduct the computation required in TDS in a highly scalable and efficient fashion. The two phases of the approach are based on the two levels of parallelization provisioned by MapReduce on cloud. Basically, MapReduce on cloud has two levels of parallelization, i.e., job level and task level. To achieve high scalability, parallelizing multiple jobs on data partitions in the first phase, but the resultant anonymization levels are not identical. To obtain finally consistent anonymous data sets, the second phase is necessary to integrate the intermediate results and further anonymize entire data sets. Details are formulated as follows. All intermediate anonymization levels are merged into one in the second phase.For the case of multiple anonymization levels, it can merge them in the same way iteratively.

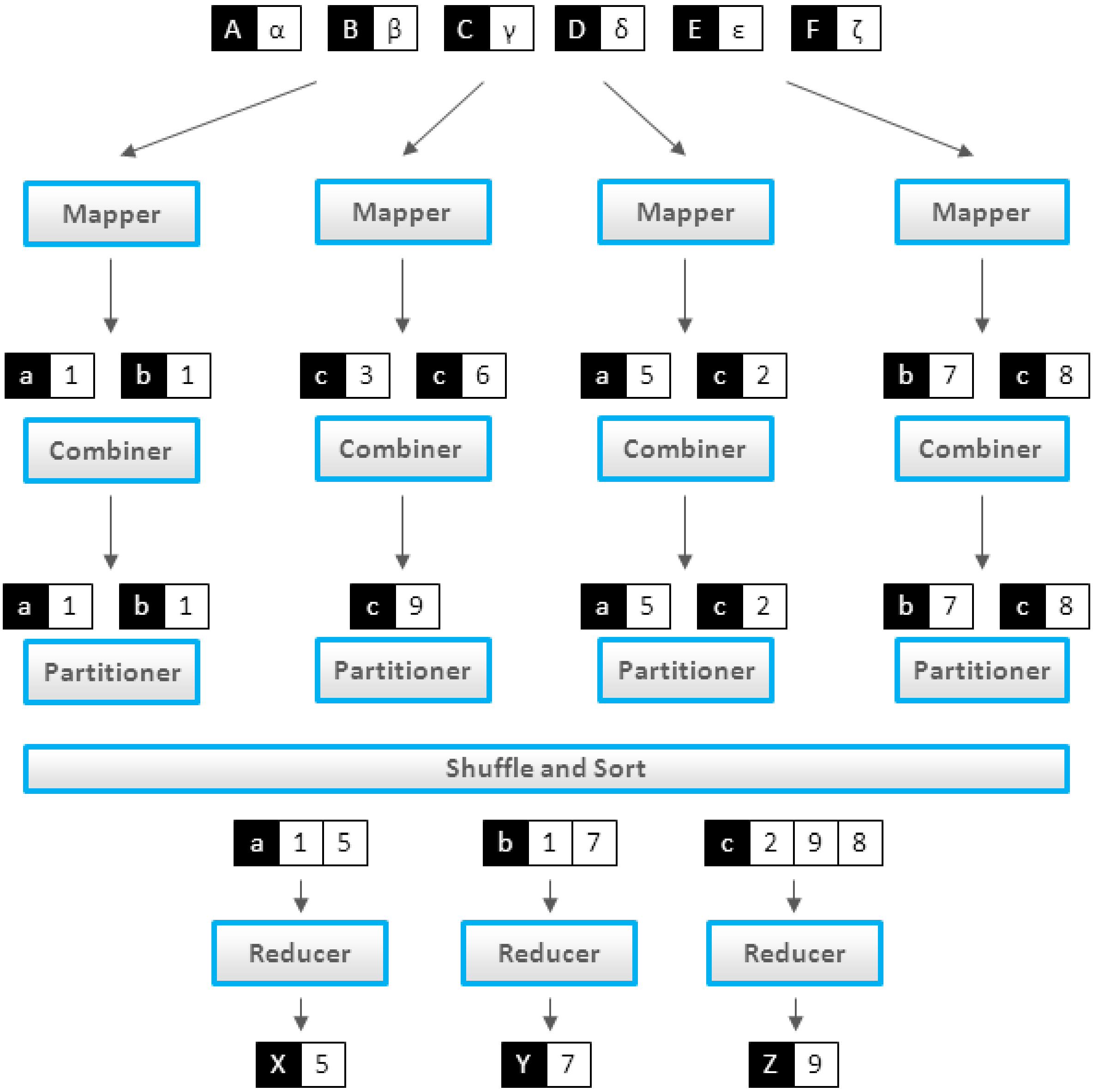
IV. MAP-REDUCE

MapReduce is a programming model for processing large data sets with a parallel, distributed algorithm on a cluster. A MapReduce program is composed of a Map() procedure that performs filtering and sorting (such as sorting students by first name into queues, one queue for each name) and a Reduce() procedure that performs a summary operation (such as counting the number of students in each queue, yielding name frequencies). The MapReduce System is orchestrates by marshalling the distributed servers, running the various tasks in parallel, managing all communications and data transfers between the various parts of the system, providing for redundancy and fault tolerance, and overall management of the whole process.

"Map" step: The master node takes the input, divides it into smaller sub-problems, and distributes them to worker nodes. A worker node may do this again in turn, leading to a multi-level tree structure. The worker node processes the smaller problem, and passes the answer back to its master node.

"Reduce" step: The master node then collects the answers to all the sub-problems and combines them in some way to form the output – the answer to the problem it was originally trying to solve.

MapReduce allows for distributed processing of the map and reduction operations.



V. ANONYMIZATION

Anonymization of data can mitigate privacy and security concerns and comply with legal requirements. Anonymization is not invulnerable countermeasures that compromise current anonymization techniques can expose protected information in released datasets. After gets the individual data sets it applies the anonymization. The anonymization means hide or remove the sensitive field in data sets. Then it gets the intermediate result for the small data sets. The intermediate results are used for the specialization process

Anonymization Algorithm :

```
DA(D,I,k,m)
1.    scan D and create count-tree
2.    initialize count
3.    for each node v in preorder count-tree traversal do
4.    if the item of v has been generalized in count then
```

```
backtrack
if v is a leaf node and v.count < k then
J:= itemset corresponding to v
find generalization of items in J that make J kanonymous
merge generalization rules with Cout
backtrack to longest prefix of path J,wherein no item has been generalized in Cout
Return Cout
for i :=1 to Cout do
initialize count=0
scan each transactions in Cout
Seperate each item in a transaction and store it in p
Increment count
for j:=1 to count do
for all g belongs Cout do
compare each item of p with that of Cout
if all items of i equal to cout
Increment the r
if ka equal to r then backtrack to i
else if r greater than ka then get the index position of the similar transactions
make them NULL until ka equal to r
else update the transactions in database
```

VI. OBS

The OBS called optimized balancing scheduling. Scheduling map tasks to improve data locality is crucial to the performance of MapReduce. It presents a new queuing architecture and proposes a map task scheduling algorithm constituted by the Join the Shortest Queue policy together with the MaxWeight policy. It identifies an outer bound on the capacity region, and then prove that the proposed algorithm stabilizes any arrival rate vector strictly within this outer bound. It shows that the algorithm is throughput optimal and the outer bound coincides with the actual capacity region.

It asymptotically minimizes the number of backlogged tasks as the arrival rate vector approaches the boundary of the capacity region. Therefore the proposed algorithm is also delay optimal in the heavy-traffic regime. Here it focus on the two kinds of the scheduling called time and size

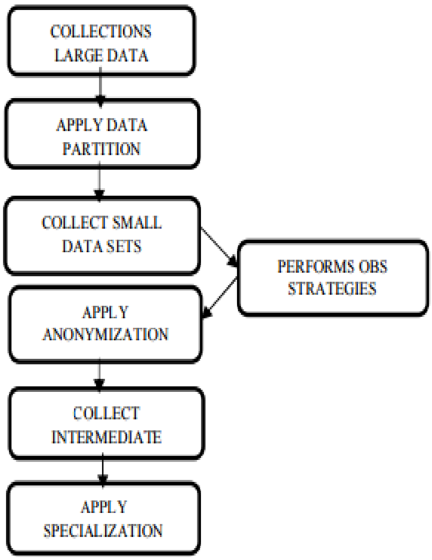


Fig3. Dataflow diagram

VII. CONCLUSION

The scalability problem of large-scale data anonymization by Top-Down Specialization and proposed a highly scalable two-phase TDS approach using MapReduce on cloud. Datasets are partitioned and anonymized in parallel in the first phase producing intermediate results. Then, the intermediate results are merged and further anonymized to produce consistent k-anonymous data sets in the second phase. It have creatively applied MapReduce on cloud to data anonymization and deliberately designed a group of innovative MapReduce jobs to concretely accomplish the specialization computation in a highly scalable way.

REFERENCES

[1]. S. Chaudhuri, “What Next?: A Half-Dozen Data Management Research Goals for Big Data and the Cloud,” in Proc. 31st Symp.Principles of Database Systems (PODS'12), pp. 1-4, 2012.

[2].NIST, "Standards for Security Categorization of Federal Information and Information Systems. FIPS-199", , Accessed Dec 2010.

[3]. W. Wang, K. Zhu, L. Ying, J. Tan, and L. Zhang, “Map task scheduling in mapreduce with data locality: Throughput and heavytraffic optimality,”Arizona State Univ., Tempe, AZ, Tech. Rep., Jul. 2012.

[4]. X. Zhang, Chang Liu, S. Nepal, S. Pandey and J. Chen, “A Privacy Leakage Upper-Bound Constraint Based Approach for CostEffective Privacy Preserving of Intermediate Datasets in Cloud,” IEEE Trans. Parallel Distrib. Syst., In Press, 2012.