International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2014): 5.611

A Scalable Approach for Scheduled Data Anonymization Using MapReduce on Cloud

Surumi K S[1], Joyal Ulahannan[2]

[1]Computer Science and Engineering, ICET, Muvattupuzha, India

[2]Assistant Professor, Information Technology, ICET, Muvattupuzha, India

Abstract: ***Cloud computing is a new development of grid, parallel, and distributed computing with visualization techniques. It is changing the IT industry in a prominent way. Cloud computing has grown due to its advantages like storage capacity, resources pooling and multi-tenancy. On the other hand, the cloud is an open environment and since all the services are offered over the Internet , there is a great deal of uncertainty about security and privacy at various levels. This paper aims to Anonymizing data sets via generalization to satisfy certain privacy requirements such as 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. we propose a scalable two-phase top-down specialization (TDS) approach to anonymize large-scale data sets using the MapReduce framework on cloud. Together with that we develop the system to deanonymize the same data within the specific scheduled time-to-live. Experimental evaluation results demonstrate that with our approach, the scalability and efficiency of TDS can be significantly improved over existing approaches.***

Keywords: Data anonymization, top-down specialization, MapReduce, cloud, privacy preservation

1. Introduction

CLOUD computing, a disruptive trend at present, poses a significant impact on current IT industry and research communities [1], [2], [3]. 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 without heavy infrastructure investment Cloud users can reduce huge upfront heavy investment of IT infrastructure, and concentrate on their own core business. However, numerous potential customers are still hesitant to take advantage of cloud due to privacy and security concerns .The research on cloud privacy and security has come to the picture.

Privacy is one of the most concerned issues in cloud computing, and the concern aggravates in the con text of cloud computing although some privacy issues are not new [1]. Person al data like electronic health records and financial transaction on records are usually deemed extremely sensitive although these data can offer significant human benefits if they are analyzed and mined by organizations such as disease research centers. For instance, Microsoft Health Vault, an online cloud health service, aggregates data from users and shares the data with research institutes. Data privacy can be divulged with les s effort by malicious cloud users or providers because of the failures of some traditional privacy protection measures on cloud . This can bring considerable economic loss or severe social reputation impairment to data owners. Hence, data priv acy issues need to be addressed urgently before data sets are analyzed or shared on cloud.

Data anonymization has been extensively studied and widely adopted for data privacy preservation in no interactive data publishing and sharing scenarios .Data anonymization refers to hiding identity and/or sensitive data for owners of data records. Then, the privacy of an individual can be effectively preserved while certain aggregate information is exposed to data users for diverse analysis and mining. A variety of anonymization algorithms with different anonymization operations have been proposed. However, the scale of data sets that need anonymizing in some cloud applications increases tremendously in accordance with the cloud computing and Big Data trends .Data sets have become so large that anonymizing such data sets is becoming a considerable challenge for traditional anonymization algorithms. The researchers have begun to investigate the scalability problem of large scale data anonymization.

In this paper, we propose a highly scalable two-phase TDS approach for data anonymization based on MapReduce on cloud. To make full use of the parallel capability of MapReduce on cloud, specializations required in an anonymization process are split into two phases. In the first one, original data sets are partitioned into a group of smaller datasets, and these data sets are anonymized in parallel, producing intermediate results. In the second one, the intermediate results are integrated into one, and further anonymized to achieve consistent k-anonymous datasets. We leverage MapReduce to accomplish the concrete computation in both phases. A group of MapReduce jobs is deliberately designed and coordinated to perform specializations on data sets collaboratively. We evaluate our approach by conducting experiments on real-world data sets. Experimental results demonstrate that with our approach, the scalability and efficiency of TDS can be improved significantly over existing approaches. Together with that we develop the system to deanonymize the same data within the specific scheduled time-to-live.

2. Related Work and Problem Analysis

2.1 Related Work

Recently, data privacy preservation has been extensively investigated. We briefly review related work below. LeFevre etal addressed the scalability problem of anonymization algorithm via introducing scalable decision trees and sampling techniques. Iwuchukwu and Naughton proposed an R-tree index-based approach by building a spatial index over data sets, achieving high efficiency.However,the above approaches aim at multidimensional generalization ,thereby failing to work in the TDS approach. Fung et al proposed the TDS approach that produces anonymous data sets without the data explaration problem. A data structure Taxonomy Indexed PartitionS (TIPS) is exploited to improve the efficiency of TDS. But the approach is centralized, leading to its inadequacy in handling large-scale data sets.

Several distributed algorithms are proposed to preserve privacy of multiple data sets retained by multipleparties.Jiang and Clifton and Mohammed et al. proposed distributed algorithms to anonymize vertically partitioned data from different data sources without disclosing privacy information from one party to another.Jurczyk and Xiong and Mohammed et al proposed ed distributed algorithms to anonymize horizontally partition- ed data sets retained by multiple holders. However, the above distributed algorithms mainly aim at securely integrating and anonymizing multiple data sources. Our research mainly focuses on the scalability issue of TDS anonymization, and is, therefore, orthogonal and complementary to them.

There is an assumption that all data processed should fit in memory for the centralized approaches. Unfortunately, this assumption often fails to hold in most data-intensive cloud applications now days. In cloud environments, computation is provisioned in the form of virtual machines (VMs). Usually, cloud compute services offer several flavors of VMs. As a result, the centralized approaches are difficult in handling large-scale data sets well on cloud using just one single VM even if the VM has the highest computation and storage capability. As to MapReduce- relevant privacy protection, Roy et al. Investigated the data privacy problem caused by MapReduce and presented a system named Airavat incorporating mandatory access control with differential privacy. Further, Zhang et al. leveraged MapReduce to automatically partition a compu ting job in terms of data security levels, protecting data privacy in hybrid cloud. Our research exploits MapRedue itself to anonymize large-scale data sets before data are further processed by other MapReduce jobs, arriving at privacy preservation.

2.2 Problem Analysis

We analyze the scalability problem of existing TDS approaches when handling large-scale data sets on cloud. The centralized TDS approaches in exploits the data structure TIPS to improve the scalability and efficiency by indexing anonymous data records and retaining statistical information in TIPS. The data structure speeds up the specialization process because indexing structure avoids frequently scanning entire data sets and storing statistical results circumvents recomputation overheads. On the other hand, the amount of metadata retained to maintain the statistical information and linkage information of record partitions is relatively large compared with datasets themselves, thereby consuming considerable memory. Moreover, the overheads incurred by maintaining the linkage structure and updating the statistic information will be huge when date sets become large. Hence, centralized approaches probably suffer from low efficiency and scalability when handling large-scale data sets.

There is an assumption that all data processed should fit in memory for the centralized approaches . Unfortunately, this assumption often fails to hold in most data-intensive cloud applications nowadays. In cloud environments, computation is provisioned in the form of virtual machines (VMs). Usually, cloud compute services offer several flavors of VMs. As a result, the centralized approaches are difficult in handling large-scale data sets well on cloud using just one single VM even if the VM has the highest computation and storage cap- ability.

A distributed TDS approach [20] is proposed to address the distributed anonymization problem which mainly concerns privacy protection against other parties, rather than scalability issues. Further, the approach only employs information gain, rather than its combination with privacy loss, as the search metric when determining the best specializations. As pointed out in, a TDS algorithm without considering privacy loss probably chooses a specialization that leads to a quick violation of anonymity requirements. Hence, the distributed algorithm fails to produce anonymous data sets exposing the same data utility as centralized ones. Besides, the issues like communication protocols and fault tolerance must be kept in mind when designing such distributed algorithms. As such, it is inappropriate to leverage existing distributed algorithms to solve the scalability problem of TDS.

3. Background Theory

3.1 Top-Down Specialization

Generally, TDS is an iterative process starting from the top-most domain values in the taxonomy trees of attributes. Each round of iteration consists of three main steps, namely, finding the best specialization, performing specialization and updating values of the search metric for the next round. Such a process is repeated until k-anonymity is violated, to expose the maximum data utility. The goodness of a specialization is measured by a search metric. We adopt the information gain per privacy loss (IGPL), a tradeoff metric that considers both the privacy and information requirements, as the search metric in our approach. A specialization with the highest IGPL value is regarded as the best one and selected in each round. We briefly describe how to calculate the value of IGPL subsequently to make readers understand our approach well. Interested readers can refer to for more details.

Given a specialization spec : p --->Child(p), the IGPL of the specialization is calculated by

IGPL(spec) = IG(spec) / (PL(spec) + 1).

The term IG(spec) is the information gain after performing spec, and PL(spec) is the privacy loss. IG(spec) and PL(spec) can be computed via statistical information derived from data sets. Let Rx denote the set of original records containing attribute values that can be generalized to x. |Rx| is the number of data records in Rx.

3.2 Two-Phase Top-Down Specialization (TPTDS)

The sketch of the TPTDS approach is elaborated in Section 3.2.1.Three components of the TPTDS approach, namely, data partition, anonymization level merging, and data specialization are detailed in other Sections .

3.2.1 Sketch of Two-Phase Top-Down Specialization

We propose a TPTDS approach to conduct the computation required in TDS in a scalable and efficient fashion. The two phases of our 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. Job level parallelization means that multple MapReduce jobs can be executed simultaneously to make full use of cloud infrastructure resources. Combined with cloud, MapReduce becomes more powerful and elastic as cloud can offer infrastructure resources on demand, for example, Amazon Elastic MapReduce service. Task level parallelization refers to that multiple mapper/reducer tasks in a MapReduce job are executed simultaneously over data splits. To achieve high scalability, we 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.

Then, we run a subroutine over each of the partitioned data sets in parallel to make full use of the job level parallelization of MapReduce. The subroutine is a MapReduce version of centralized TDS (MRTDS) which concretely conducts the computation required in TPTDS. MRTDS anonymizes data partitions to generate intermediate anonymization levels. An intermediate anonymization level means that further specialization can be performed without violating k-anonymity. MRTDS only leverages the task level parallelization of Map- Reduce. More details of MRTDS will be elaborated in Section 3.3.

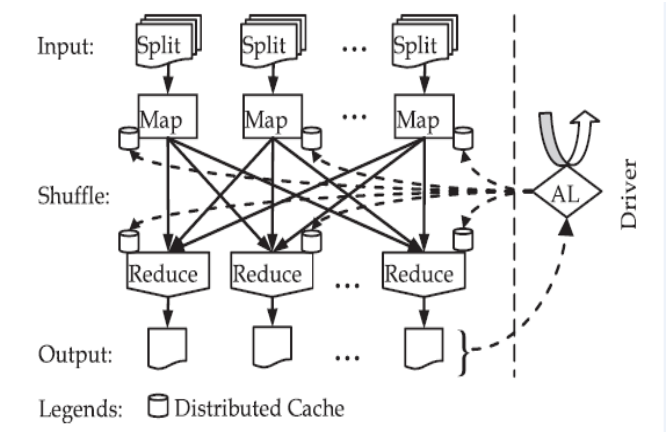

Fig. 1. Execution framework overview of MRTDS.

3.2.2 Data Partition

When Dis partitioned into Di, 1 <= i <=p, it is required that the distribution of data records in Di is similar to D. A data record here can be treated as a point in an m-dimension space, where m is the number of attributes. Thus, the intermediate anonymization levels derived from Di, 1<= i<=p, can be more similar so that we can get a better merged anonymization level. Random sampling technique is adopted to partition D, which can satisfy the above requirement. Specifically, a random number rand, 1 < = rand < = p, is generated for each data record. A record is assigned to the partition Drand.

3.2.3 Anonymization Level Merging

All intermediate anonymization levels are merged into one in the second phase. The merging of anonymization levels is completed by merging cuts.

3.2.4 Data Specialization

An original data set D is concretely specialized for anonymization in a one-pass MapReduce job. The Map function emits anonymous records and its count. The Reduce function simply aggregates these anonymous records and counts their number An anonymous record and its count represent a QI-group. The QI-groups constitute the final anonymous data sets.

3.3 MapReduce Version of Centralized TDS

We elaborate the MRTDS in this section. MRTDS plays a core role in the two-phase TDS approach, as it is invoked in both phases to concretely conduct computation. Basically, a practical MapReduce program consists of Map and Reduce functions, and a Driver that coordinates the macro execution of jobs.

3.3.1 MRTDS Driver

Usually, a single MapReduce job is inadequate to accomplish a complex task in many applications. Thus, a group of MapReduce jobs are orchestrated in a driver program to achieve such an objective. MRTDS consists of MRTDS Driver and two types of jobs, i.e., IGPL Initialization and IGPL Update. The driver arranges the execution of jobs.

3.3.2 IGPL Initialization Job

The main task of IGPL Initialization is to initialize information gain and privacy loss of all specializations in the initial anonymization level AL.

3.3.3 IGPL Update Job

The IGPL Update job is quite similar to IGPL Initialization, except that it requires less computation and consumes less network bandwidth. Thus, the former is more efficient than the latter.

3.4 Deanonymization Module

In this module we have to deanonymize the anonymized data by setting a time-to-live with anonymized data. When reaching the time-to-live the data will be deanonymized to outside.

4. Conclusion

In this paper, we have investigated the scalability problem of large-scale data anonymization by TDS, and proposed a high- ly scalable two-phase TDS approach using MapReduce on cloud. Data sets 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. We have creatively applied MapReduce on cloud to data anonymization and deliberately designed a group of innovative Map Reduce jobs to concretely accomplish the specialization computation in a highly scalable way. Experimental results on real-world data sets have demonstrated that with our approach, the scalability and efficiency of TDS are improved significantly over existing approaches. Here also we have to deanonymize the anonymized data by setting a time-to-live with anonymized data. When reaching the time-to-live the data will be deanonymized to outside.

5. Acknowledgment

The Author would like to thank Joyal Ulahannan Assistant Professor, Department of Information Technology, Ilahia College of Engineering and Technology, Muvattupuzha for his moral and technical support.

References

[1] S. Chaudhuri, “What Next?: A Half-Dozen Data Management Research Goals for Big Data and the Cloud,” Proc. 31st Symp.Principles of Database Systems(PO- ODS ’12), pp. 1-4, 2012.
[2] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R. Katz, A Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M.Zaharia, “A View of Cloud Computing,” Comm. ACM, vol. 53,no. 4, pp. 50-58, 2010.
[3] L. Wang, J. Zhan, W. Shi, and Y. Liang, “In Cloud, Can Scientific Communities Benefit from the Econom- ies of Scale?,” IEEE Trans. Parallel and Distributed Systems, vol. 23, no. 2, pp.296-303, Feb.2012.
[4] H. Takabi, J.B.D. Joshi, and G. Ahn, “Security and Privacy Challenges in Cloud Computing Environment-s,”IEEE Security and Privacy, vol. 8, no. 6, pp. 24-31, Nov. 2010.
[5] D. Zissis and D. Lekkas, “Addressing Cloud Computi-ng Security Issues ,” Future Generation Computer Systems, vol. 28, no. 3, pp. 583-592, 2011.

Author Profile

Surumi K S received the Bachelor of Technology degree in Information Technology from Mahatma Gandhi University, Kerala. She is currently doing Master of Technology degree in Computer Science and Engineering with Specialization in Information Systems from Mahatma Gandhi University, Kerala.

Joyal Ulahannan he is currently assistant professor at ICET, Muvattupuzha, Mahatma Gandhi University, Kerala.