Advances in Computer, Signals and Systems (2022)
Clausius Scientific Press, Canada

DOI: 10.23977/acss.2022.060707
ISSN 2371-8838 Vol. 6 Num. 7

# *Distributed Privacy-preserving Clustering Mining Algorithm for Heterogeneous Computing*

**Jing Qi***

*College of Intelligent Equipment, Shandong University of Science and Technology, Taian, Shandong, 271019, China*
*qijingkeda@163.com*
**Corresponding author*

***Keywords:*** Heterogeneous Computing, Distributed Architecture, Privacy Protection, Clustering Algorithm

***Abstract:*** As a typical unsupervised data mining method, cluster analysis can mine unknown knowledge and potential value from massive data. However, in the process of mining useful information, the personal privacy information in the data may be leaked. Therefore, privacy protection technology comes into being. This paper focuses on the distributed privacy preserving clustering mining algorithm for heterogeneous computing. This thesis first constructs a mathematical model based on heterogeneous Hadoop. In order to further improve the availability of the algorithm, an effective algorithm DPk means ev is studied and proposed. The algorithm improves the selection of the initial central point and avoids the blindness of the value setting and the sensitivity of the initial central point selection. The experimental results show that the algorithm effectively improves the efficiency and availability of clustering.

## 1. Introduction

Since the new century, the Internet industry has developed rapidly, and the convenience and rapidity of communication and data sharing have been greatly improved. However, the resulting risk of privacy leakage is also increasing. With the development of computer technology and the increasing richness of network attacks, protecting private data is no longer as simple as hiding sensitive attributes in the data. Personal privacy protection in the process of data release is a challenging problem. In the existing privacy protection models, k-anonymity and its extended model can anonymize records to a certain extent by hiding identification information, but this method needs to evaluate the background knowledge of the attacker and the corresponding attack model in advance [1]. There is also no quantitative analysis of the level of privacy protection. As a new privacy protection technology, differential privacy protection technology has a good effect in solving the above problems and can be applied to data mining algorithms to protect the original data and the privacy information in the process of data mining from being leaked [2]. Cluster analysis can be seen as an exploratory data analysis (EDA) process, which can reveal interesting unknown relationships in data and discover hidden patterns or structures of interest in data. Module analysis is widely used in many technical fields, including machine learning, model recognition, image

analysis, information retrieval, bioinformatics, etc. In addition, module analysis has been widely and effectively applied in commercial, biological, geological, and other industries [3]. At the same time, the disclosure of a large number of sensitive information in the process of clustering also brings large losses and adverse effects to users, and related reports also frequently enter the public view, thus causing unprecedented attention [4]. Therefore, it has become a hot issue in the field of data mining and data privacy protection to introduce privacy protection technology into cluster analysis.

In recent years, differential privacy technology has become a research point at home and abroad, but most of the existing differential privacy research focuses on the theoretical nature of the proposed user privacy model [5]. Privacy has become a key issue in data mining. Existing privacy technologies for grouping analysis include random disturbance, data rotation, data exchange, etc. Some scholars have implemented Differential Privacy k-means Algorithm (DPk-means) on SuLQ platform. However, it does not take into account the high sensitivity of the query function and does not give a detailed privacy budget allocation method, resulting in low clustering availability [6]. On the basis of Blum, some scholars analyzed in detail the sensitivity calculation methods of each query function in the differential privacy K-means algorithm and proposed two different privacy budget allocation methods [7]. In recent years, scholars at home and abroad have proposed a large number of clustering algorithms for differential privacy protection. However, they still have shortcomings in the following aspects: due to the differential privacy, the amount of noise added will inevitably affect the availability and accuracy of data; the scale of applicable samples is insufficient [8]. It is difficult to apply large-scale and high-dimensional sample sets; the types of applicable samples are insufficient, and some algorithms are difficult to apply to sample sets with different shapes or densities. The computation complexity is high, the computation amount is large, the time efficiency is low.

This paper studies the privacy protection problem in the framework of distributed computing, especially proposes a privacy protection method based on clustering for a specific distributed computing environment ecosystem, and improves the overall performance by improving the traditional clustering algorithm.

## 2. Heterogeneous Distributed Clustering Algorithm

### 2.1 Mathematical Modeling Based on Heterogeneous Hadoop

(1) Data Coflow: Coflow first appeared as a network subtraction of cluster applications, including not only MapReduce (Hadoop model) and data flow with cycles (Spark model), and many other communications in computing cluster applications.

Each Coflow can be expressed as c(S, D) = {f1, f2, f3,... , f | | c} says a bunch of machines (tribe) D () S and machine, the flow of data between | | c represents the clan (machine) S and clan (machine) D and all possible data flow between the number of combined. For example, in a Hadoop cluster containing only two physical machines, if the machine S needs to be processed with m, Map function we need to deal with r, D machine Reduce function, | | c = Mr.

The start time and end time of each data flow fi (fi ∈ c) are defined as start(fi) and end(fi), respectively. Then the start time and end time of data flow Coflow can be defined as:

$$start(c) = \min_{fi} start(fi) \tag{1}$$

$$end(c) = \min_{fi} end(fi) \tag{2}$$

The completion time of the data flow Coflow is:

$$\bar{c} = end(c) - start(c) \tag{3}$$

(2) In the process of processing a piece of data, there will be a lot of resources involved, such as the core CPU, memory, RAM and hard disk reading and writing, etc. In order to ignore these processing details and focus on the overall operation efficiency, define a Job completion Time (JPT) to represent the completion time of a job submitted by a specific user:

$$JPT = \max_{machine_j} T_e(job_{ij}) - \max_{machine_j} T_s(job_{ij}) \tag{4}$$

Where Ts(jobij) represents the time when machine ej receives the processing data from jobi, and Te(jobij) represents the end time when the local processing data is transferred to the next machine.

(3) The computation Task resource requirement matrix requires that all jobs are decomposed into specific tasks, and each task has the required amount of data, CPU resources required for computation, and memory resources required for computation. These attributes are native to each task. Therefore, the resource requirements of all tasks are integrated into a matrix. If there are N tasks to be processed in the cluster, the property of any ith task includes the CPU resource required for processing this task, and the memory capacity required for processing this task. That is, the amount of disk data required to process the task.

(4) After a distributed computing cluster is started, the cluster should be composed of M computing nodes, namely, physical machines. The resources required by each task are provided by physical nodes with available resources of corresponding dimensions, which provides convenience for the subsequent algorithm design.

As shown in Figure 1, the resource management module of the cluster collects historical job completion data from the worker node (as shown in Figure 2) through log files or current heartbeat information for clustering model training. The whole scheduling strategy is to schedule newly submitted jobs based on the clustering results. At the same time, as the processing of new tasks is completed, new log files will be generated and historical data will be processed, which will increase the integrity and accuracy of the clustering model, and finally form a cycle of data collection -- scheduling -- new data collection -- rescheduling.

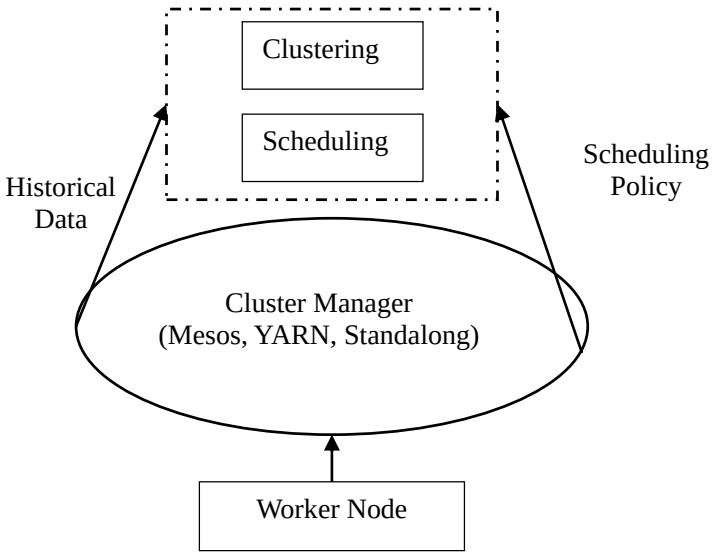


Figure 1: System diagram based on model scheduling strategy

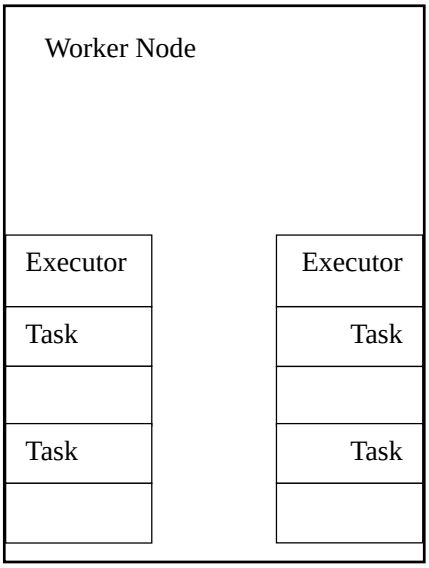


Figure 2: Schematic diagram of working node

## 2.2 Optimization of DPk-Means ++ Clustering Algorithm

k-means++ clustering algorithm is proposed to solve the problem that the accuracy of k-means clustering algorithm is significantly affected by the randomness of the selection of the initialization center. In the process of clustering, it is necessary to provide protection for relevant personal data. Based on the assumption of maximum background knowledge, the differential privacy model can define the attack model and quantitatively analyze the privacy protection strength [9-10].

In order to further improve the availability of DPk-means++ clustering results, an efficient DPK-means-IP clustering algorithm is studied and proposed. The algorithm of k - means++ clustering result for input, alternate with some specific mechanisms, finally run k - means algorithm, the selection of the process can improve the initial center, to avoid the blindness of setting values of k and choose to be the center initialization sensitivity, both to protect privacy, and can reduce the iteration times and improve the efficiency of clustering.

The specific design steps of DPk-means algorithm are as follows:

The error Ø(C,X) of the k-means++ algorithm running on the dataset is stored to Øbest while clustering the center point set C

It is stored in Cbest;

Initialize the centers μ and λ : move the most "useless" center μ in the cluster to the center with the largest error (the same applies to λ, with the opposite sign, with the small random number o);

Run k-means with C as the initialization center;

Determine the size of Ø(C,X) and Øbest. If Ø(C,X) is less than Øbest, then move to 1), otherwise proceed to the next step;

Repeat 2) -4) in the initial phase until the given number of retries is used up.

Go back to the optimal center point set Cbest.

The minimum distance between all data points and their center points is calculated, and then the data points are assigned to the corresponding center points to form k clusters.

The sum of data points in each cluster and the number of points in each cluster were calculated, and Lap(b), sum '=sum+Lap(b)ˆ and num' =num+Lap(b) were added to them respectively. Update

the cluster center to sum '/num.

The above two parts are iteratively executed until the sum of squared errors converges.

The experiment requires two parameters: the privacy budget ε and the number k of clusters to be clustered.

ε : A reasonable budget allocation strategy is required to make the ε budget sufficient. In general, the value of ε is set between the interval (0.01, 0.1), or in some cases ln2 or ln3. Therefore, in the experiment, ε is usually assigned by the linear distribution method to [0,1] interval.

Assuming that only one record is different between the two data sets, the process of calculating k center points is the same as the histogram query of dividing space [0,1]d, and the sensitivity of the denominator count is 1. Therefore, the sensitivity of the whole query sequence function is d+1.

In the clustering algorithm, the iteration times of different data sets are different, so the convergence conditions are also different

Let N represent the number of iterations. Then: If N is fixed, then each iteration consumes the privacy budget ε/ N. According to the differential privacy definition, to obtain ε-differential privacy protection, the noise added each time is Lap((d+1N)/ε). If N is unknown, the value of ε should be adjusted continuously during the iteration.

According to experience, the early iteration has a greater impact on the clustering results. Therefore, in the experiment, this paper will gradually increase the privacy budget for clustering, first allocation of privacy budget ε/ 2, noise size Lap((d+1N)/ε), and then each subsequent iteration

The budget consumption is always half of the previous one until the last iteration is completed.

k: This paper focuses on the differential privacy optimization application of K-means algorithm, especially the selection of the initial center point, rather than a simple clustering application. Therefore, based on the number of reference categories provided by the dataset, we set the size of k according to the recommended value.

## 3. Algorithm Simulation Experiment

In this section, the DPk-means-IP algorithm is tested on the dataset to verify its effectiveness. The programming language used is C++, the programming environment is Clion, the experimental environment is Intel(R) Core I5-10400, 16GB memory, Windows 10 operating system.

### 3.1 Experimental Data Set

Because the algorithm adopts the structure of a quartile tree, the algorithm is suitable for two-dimensional data. The specific data sets used are as shown in Table 1:

Table 1: Data set Information

| Name | Data set | Sample size | Cluster | Number of attributes | Attribute types |
|---|---|---|---|---|---|
| A | Pytest | 200000 | 20 | 2 | Real |
| B | Checkin | 1200000 | 100 | 2 | Real |
| C | Unbalance | 6000 | 10 | 2 | Real |
| D | Birch3 | 150000 | 100 | 2 | Real |

Pytest data set is a random data set generated by Python for preexperiment use, and the data type is numerical data. The Checkin dataset is the longitude, latitude, and location information of hotel customers checking in on the social networking site Gowalla. The data type is numerical data with sparse distribution. The Unbalance dataset is a numerical data set composed of 6000 vectors and 10 advanced clustering. birch3 data set is a two-dimensional data set composed of random locations and random size.

### 3.2 Experimental Setting

In this paper, the following algorithms are run on the four datasets listed: DPk-means algorithm, DPk-means++ algorithm, DPK-means-IP.

During the experiment, the setting of differential privacy budget is gradually raised from 0.01 to 1, which can better observe the performance effect of the algorithm.

In this paper, the Relative Clustering Performance (RCP) is used for the refinement judgment. The purpose of calculating RCP is to enlarge the relative relationship of NICV between algorithms. RCP is a percentage value, which is greater than 0, indicating that the proposed algorithm has more advantages than the traditional k-means algorithm, otherwise, it means that the traditional k-means algorithm has a better effect.

## 4. Analysis of Experimental Results

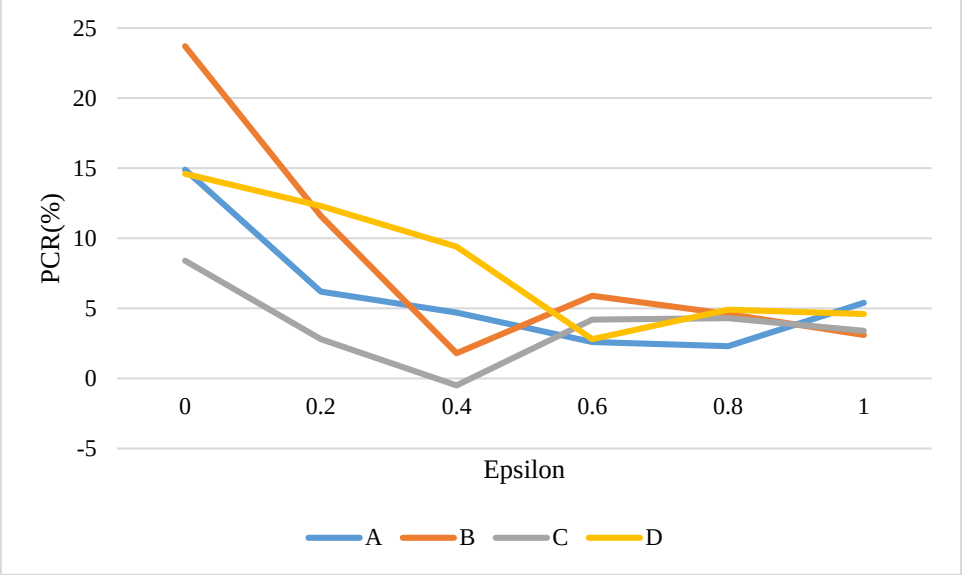


Figure 3: RCP indicator results

As shown in Figure 3, almost all the performance indexes of RCP are greater than 0, and when the differential privacy protection intensity is high (that is, when the differential privacy budget is low), the clustering effect of DPk-means-IP algorithm has advantages, and the average performance is increased by 10%-25%. This shows that the method of constructing differential privacy quadtrees to initialize the dataset is effective, and the method of dynamically partitioning the dataset can add more appropriate differential privacy noise than the method of equal partitioning, which reduces the error.

As shown in Figure 4, you can see that the computation time of the algorithm is proportional to the size of the dataset. In the case of the same data set, the computation time of DPk-means algorithm and DPK-means ++ algorithm is relatively long, because these two algorithms need to calculate all data repeatedly in every iteration, which leads to low efficiency.

The computation time of DPk-means-IP algorithm is much shorter than the other two, because the algorithm will initialize the data set, and then use the segmtioned block center to replace the data in the block, and the subsequent iteration will not go through all data one by one, which greatly improves the efficiency.

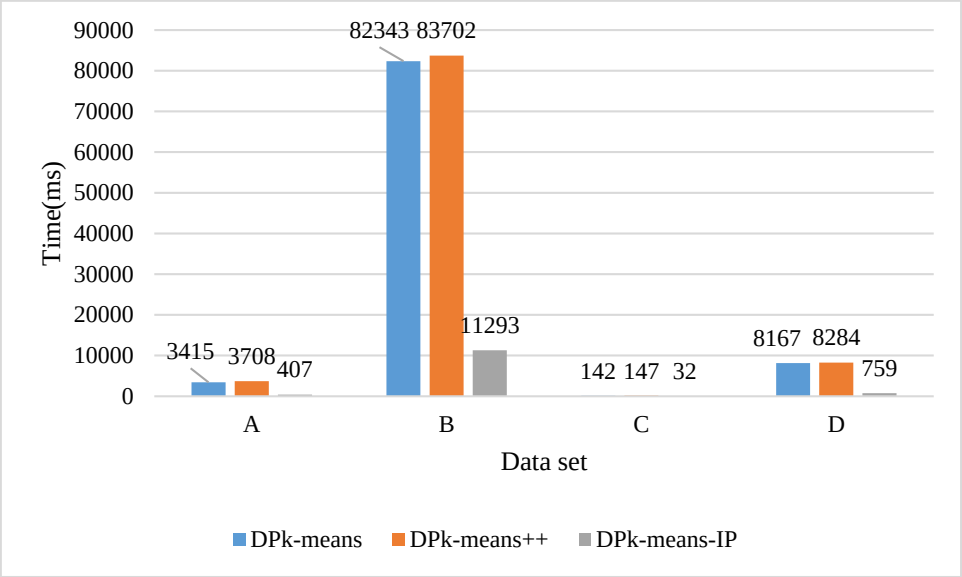


Figure 4: Computation time of each algorithm

## 5. Conclusions

Due to the increasingly serious problem of data privacy disclosure in today's society, good privacy protection has become an urgent need. Based on the theoretical research of differential privacy protection at home and abroad, the present study deeply studies the development of differential privacy protection in data mining and data publishing. Aiming at the problem of insufficient privacy of the k-mean++ algorithm throughout the process, in order to further improve the availability of the algorithm, this thesis studies and proposes an effective algorithm that can improve the selection of the initial central point. The experimental results show that the algorithm can provide different levels of data privacy protection within the range of privacy budget parameters. Compared with other differential data protection algorithms, the media algorithm can effectively improve the efficiency and availability of grouping under the same level of privacy protection.

## References

*[1] Manikandan V, Porkodi V, Mohammed A S, et al. Privacy Preserving Data Mining Using Threshold Based Fuzzy Cmeans Clustering[J]. ICTACT Journal on Soft Computing, 2018, 9(1).*

*[2] Ram Mohan Rao P, Murali Krishna S, Siva Kumar A P. Privacy preservation techniques in big data analytics: a survey [J]. Journal of Big Data, 2018, 5(1): 1-12.*

*[3] Shaham S, Ding M, Liu B, et al. Privacy preserving location data publishing: A machine learning approach[J]. IEEE Transactions on Knowledge and Data Engineering, 2020, 33(9): 3270-3283.*

*[4] Varma G, Chauhan R, Yafi E. ARTYCUL: A privacy-preserving ML-driven framework to determine the popularity of a cultural exhibit on display [J]. Sensors, 2021, 21(4): 1527.*

*[5] Garg S, Kaur K, Kaddoum G, et al. SDN-based secure and privacy-preserving scheme for vehicular networks: A 5G perspective[J]. IEEE Transactions on Vehicular Technology, 2019, 68(9): 8421-8434.*

*[6] Upadhyay S, Sharma C, Sharma P, et al. Privacy preserving data mining with 3-D rotation transformation[J]. Journal of King Saud University-Computer and Information Sciences, 2018, 30(4): 524-530.*

*[7] Le Nguyen B, Lydia E L, Elhoseny M, et al. Privacy preserving blockchain technique to achieve secure and reliable sharing of IoT data[J]. Computers, Materials & Continua, 2020, 65(1): 87-107.*

*[8] Menaga D, Saravanan S. GA-PPARM: constraint-based objective function and genetic algorithm for privacy preserved association rule mining [J]. Evolutionary Intelligence, 2022, 15(2): 1487-1498.*

*[9] Keshk M, Turnbull B, Moustafa N, et al. A privacy-preserving-framework-based blockchain and deep learning for protecting smart power networks [J]. IEEE Transactions on Industrial Informatics, 2019, 16(8): 5110-5118.*

*[10] Usman M, Jan M A, He X, et al. P2DCA: A privacy-preserving-based data collection and analysis framework for IoMT applications [J]. IEEE Journal on Selected Areas in Communications, 2019, 37(6): 1222-1230.*