International Core Journal of Engineering Volume 7 Issue 12, 2021
ISSN: 2414-1895 DOI: 10.6919/ICJE.202112_7(12).0063

# Incomplete Big Data Filling Algorithm based on Machine Learning

Liu Sun

Guangzhou Huali College, Guangdong, China

## Abstract

**For the application problems in big data environment, many traditional machine learning (ML)algorithms on small data are no longer applicable. Therefore, the study of ML algorithms in big data environment has become a topic of common concern in academia and industry. This paper uses ML algorithm to cluster incomplete data. The matrix multiplication idea is used to improve the process of solving the distance between data points in the subtractive clustering process, and the multilevel MapReduce parallelization is realized. The experimental results show that the proposed algorithm can cluster incomplete big data quickly and fill in missing data values effectively, which meets the requirements of big data processing and has certain theoretical research significance and practical application value.**

## Keywords

**Machine learning; Big data; Data population.**

## 1. Introduction

With the rise and development of the Internet of Things (IoT), social networks and e-commerce technologies, data is growing at an all-time speed. The era of research and application of big data has arrived [1]. In the IoT, a large number of sensor terminals work in a non-manual monitoring state, and these sensor terminals are prone to various faults, resulting in missing values in collected data, that is, incomplete data [2-3]. Incomplete data seriously affects the application of IoT. Therefore, analyzing and filling incomplete data is of great sense to the development of IoT and complex networks.

Machine learning (ML) algorithms have great practical value in academia and industry. Due to the large amount and complexity of big data, many traditional ML algorithms on small data are no longer applicable to the application of big data [4]. This is because there are abundant information dimensions in big data, and the conventional data filling algorithm cannot reflect the deep features of big data. To solve this problem, this paper proposes a ML algorithm for filling incomplete big data.

## 2. ML theory in big data environment

With the advent of the era of big data, big data has gradually become a hot spot in academia and industry, and has been widely used in many technologies and industries, from large-scale databases to business intelligence and data mining applications; Because big data is complex, high-dimensional, changeable and so on, how to mine the knowledge that people are interested in from real, messy, modeless and complex big data urgently needs the guidance of more profound ML theory.

The problems of traditional ML mainly include the following four aspects [5-6]:

(1)Understand and simulate human learning process;

(2) Research on natural language interface between computer system and human users;

(3) The ability of reasoning against incomplete information, that is, automatic planning;

(4)Construct programs that can discover new things.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202112_7(12).0063

A new challenge for traditional ML is how to deal with big data. At present, ML problems involving large-scale data are common. However, many existing ML algorithms are in view of memory, but big data cannot be loaded into computer memory, so many existing algorithms cannot deal with big data.

## 3. Overview of incomplete data filling algorithms

Data sets with missing data can not be processed in data mining, ML and some other information systems, so the missing data in the data set must be eliminated or estimated and filled in during the data preprocessing stage.

In incomplete data filling algorithms, we use many technical theories such as data averaging, probability statistics, stochastic algorithm, regression model, decision tree, expectation maximization, similarity measure, Bayesian network, support vector machine, fuzzy algorithm, neural network and genetic algorithm. According to the characteristics of different data sets, different technical theories are applied to achieve the effect of accurate data filling.

Not only to pursue the accuracy of the filling algorithm, but also to all round evaluate the efficiency of the algorithm in many aspects such as filling accuracy and processing speed, so it is particularly important to propose a data filling algorithm for specific applications. Especially with the arrival of the era of cloud computing and big data, many data filling algorithms for big data applications will be born. The following is a brief summary of some existing representative incomplete data filling algorithms.

## 4. Incomplete big data filling algorithm

### 4.1 Data preprocessing

There are some discrete attributes in the attribute set of data objects. In order to make discrete attributes directly participate in the calculation, this paper uses similarity distance to transform discrete attributes into numerical attributes, that is, $n$-dimensional data to represent a discrete attribute with $n$ different values. Through this transformation rule, it can be ensured that no information will be lost during the clustering process, that is, the mutual distance between different attributes is 1, and the distance between the same attributes is 0.

In order to eliminate the imbalance caused by different attribute value ranges, it is necessary to normalize all attribute values to a certain numerical range, so that different attributes have the same weight value. In this paper, standard deviation is used to normalize the continuous feature values in the attribute set.

### 4.2 Clustering algorithm for incomplete data.

The main steps of incomplete data clustering algorithm are as follows:

(1)Data set $O$ is divided into complete data set $C$ and incomplete data set $I$, i.e $O = C \cup I$.

(2)The discrete attribute values of data objects in the complete data set $C$ are converted into numerical values, and the attribute values of all data objects in $C$ are normalized.

(3)Calculate the similarity matrix $S$ of data objects in the data set $C$.

(4)Initialize the attraction matrix $R$ and the attribution matrix $A$, and update the attraction matrix $R$ and the attribution matrix $A$ according to formula (1) and formula (2).

The AP (Affinity Propagation) clustering algorithm updates the attraction matrix $R = [r(i,k)]$ and the attribution matrix $A = [a(i,k)]$ iteratively to determine the high-quality clustering center step by step, and updates the attraction matrix $R$ with the attribution matrix and the similarity matrix $S = [s(i,k)]$:

$$r(i,k) = s(i,k) - \max_{k' \neq k} \{a(i,k') + s(i,k')\} \quad (1)$$

ISSN: 2414-1895 DOI: 10.6919/ICJE.202112_7(12).0063

Update the attribution matrix $A$ with the attraction matrix $R$;

$$a(i,k)=\min\left\{0,r(k,k)+\sum_{i'\notin\{i,k\}}\max\{0,r(i',k)\}\right\}$$
$$a(k,k)=\sum_{i'\neq k}\max\{0,r(i',k)\} \quad (2)$$

In which $s(i,k)$ is the similarity between point $i$ and point $k$; $r(i,k)$ represents the attraction of point $k$ to point $i$; $a(i,k)$ indicates the degree of attribution of point $i$ to point $k$.

(5)Until the cluster center no longer changes, or after the specified number of iterations has been completed, stop the calculation, otherwise repeat step (4).

(6)The data point $k$ with diagonal $a(k,k)+r(k,k)>0$ is the cluster center found automatically, while the candidate cluster center with the largest $a(i,k)+r(i,k)$ for data point $i$ is the cluster center to which it really belongs.

(7)For the continuous numerical attribute in data set $C$, the similarity measure coefficient $\alpha,\beta$ in each cluster is calculated.

(8)Divide all data objects in incomplete data set $I$ into corresponding clusters, that is, calculate the similarity of data objects in $I$ to each cluster center according to formula (3), and select the cluster center with the greatest similarity as its cluster center.

$$Similarity(b,c_i)=\sum_{j=1}^{q}Pa_j(b,c_i)+\sum_{k=1}^{r}Pa_k(b,c_i) \quad (3)$$

Where $q$ represents the number of continuous numerical attributes; $k$ represents the number of discrete attributes.

### 4.3 Analysis of algorithm time complexity

Firstly, the algorithm uses improved subtractive clustering to cluster incomplete data sets directly. This section will mainly discuss the time complexity of the filling process, as follows:

The weighted distance between the data record with missing attribute value and other data records in its class is calculated, and the time complexity is $O(mn_k^{'},n_k)$, where $m$ is the number of data attributes, $n_k^{'}$ is the number of records with missing value in class $k$, $n_k$ is the total number of data records in class $k$, and $k$ represents a cluster of clusters.

Unit weighted distance, time complexity is $O(n_k^{'},n_k)$.

Fill in the missing attribute values, the time complexity is $O(n_k^{''},n_k)$, and $n_k^{''}$ is the number of missing attribute values.

Assuming that all clusters in the clustering result contain the same data records, the overall complexity of the algorithm is approximately equal to:

$$T=O(SC)+k(O(mn_k^{'},n_k)+O(n_k^{'},n_k)+O(n_k^{''},n_k)) \quad (4)$$

In which $O(SC)$ represents the time complexity of subtractive clustering and the number of clusters of clustering results.

From the time complexity of the filling algorithm, it can be seen that the main time of the filling process of the algorithm is spent on solving the weighted distance between the data record with missing attribute values and other data records in its class, and it can be parallelized under the model by using the parallelization idea of solving sample matrix.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202112_7(12).0063

## 5. Parallel design of algorithm

The main time of the algorithm is to divide the data set $S$ and cluster the data subset $C$ by using the subtractive clustering algorithm. In the process of clustering the data subset $C$ by using the subtractive clustering algorithm, the main operations are to calculate the distance between sample points and calculate and correct the density index of all sample points.

Therefore, this paper uses multi-level MapReduce to carry out distributed parallel computation on these steps. The working process of multi-level MapReduce is shown in Figure 1.

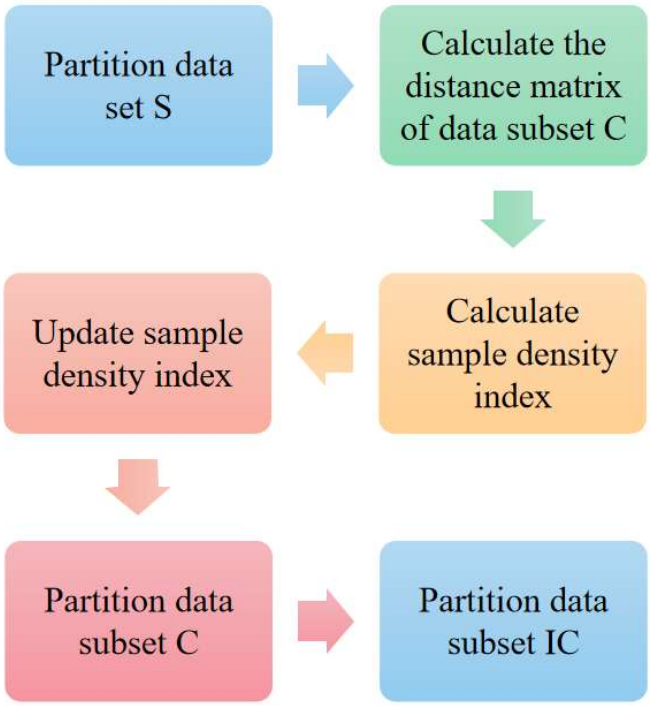


Figure 1 Multistage MapReduce process

The first MapReduce process divides data set $S$ into complete data subset $C$ and incomplete data subset $IC$.

The second MapReduce multiplies the two replica matrices of data subset $C$ to obtain the distance element between different trees.

Thirdly, the fourth MapReduce uses the distance element between samples to calculate and update the density index of each sample and determine the cluster center.

The fifth and sixth MapReduce divide all the data records in the data subsets $C$ and $IC$ into the representative clusters of the cluster center.

In order to make the data set $S$ suitable for MapReduce computing model and generate distance matrix, this paper uses matrix multiplication to realize MapReduce.

In the process of solving the distance matrix between data points, the global neighborhood radius variable is initialized by setting the MapReduce calculation model, and all sum results are accumulated in the Reduce function to obtain the calculation result of the neighborhood radius.

Under the serial condition, the time complexity of generating the distance matrix of the complete data set s is $O(mp^2)$, and $p$ is the number of number objects in the data subset $C$. For distributed computing, the time complexity of generating distance matrix is $O(2mp^2/t)+M(t)$, where $t$ represents the number of distributed nodes participating in computing.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202112_7(12).0063

# 6. Experimental results and analysis

## 6.1 Clustering performance analysis of incomplete data

Randomly choose 10% data objects from these 13 data sets, delete some attributes in these data objects, and simulate 8 incomplete data sets $O$. The clustering results are shown in Figure 2.

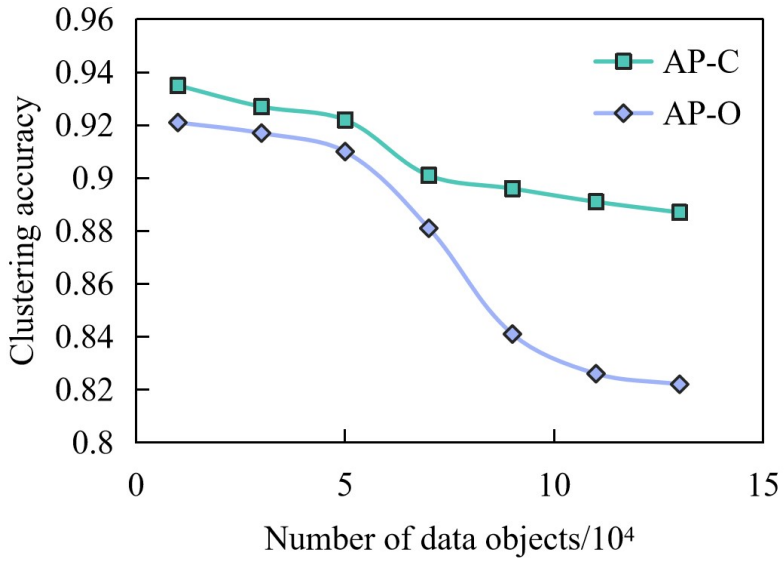


Figure 2 Comparison of clustering accuracy under different numbers of data objects

In Figure 2, the clustering accuracy curve of the complete data set $C$ by AP algorithm shows that with the raise of data volume, there are more and more noise data, and the clustering accuracy decreases, but the overall clustering accuracy remains above 80%, which fully verifies the effectiveness of AP algorithm.

However, due to the influence of incomplete data set $I$, the clustering accuracy of the algorithm for data set $O$ is lower than that of the AP algorithm for data set $C$. With the increase of data volume, the clustering accuracy of our algorithm gradually decreases, and when the data volume exceeds 30 GB, the clustering accuracy tends to be stable, which verifies the effectiveness of our algorithm for large-scale incomplete data clustering.

## 6.2 Comparison of filling accuracy

To test the performance of this algorithm, this algorithm is compared with EMI algorithm [7] and DMI algorithm [8], and the experimental results are shown in Figure 3.

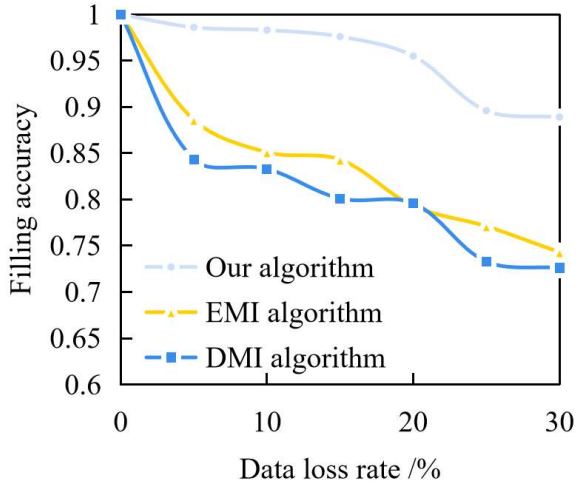


Figure 3 Comparison of filling accuracy of algorithms under different missing rates

ISSN: 2414-1895 DOI: 10.6919/ICJE.202112_7(12).0063

It can be seen from Figure 3 that with the increase of data missing rate, the filling accuracy of the three algorithms decreases. Among the three algorithms, this algorithm has the highest filling accuracy, because it effectively avoids the influence of different class objects on data filling.

In addition, when the data missing rate exceeds 15%, with the increase of missing data, the filling accuracy of EMI algorithm and DMI algorithm decreases sharply, while the algorithm in this paper is relatively stable, which shows that the algorithm in this paper can still maintain good results for data sets with high missing rate.

## 7. Summary

Big data has the characteristics of sparse attributes, super high dimension, high noise, data drift and complex relationships, which makes it difficult for traditional ML algorithms to effectively process and analyze. Aiming at the problem of filling incomplete big data, this paper uses incomplete information system theory to measure the similarity between incomplete data objects, and then uses AP algorithm to cluster incomplete data. The matrix multiplication idea is used to improve the process of solving the distance between data points in the subtractive clustering process, and the multilevel MapReduce parallelization is realized. Experimental results show that the proposed algorithm can cluster incomplete big data quickly and fill in missing data effectively, which meets the requirements of big data processing.

## References

[1] Zhang Hua. research on big data classification algorithm of optical fiber fault based on machine learning [J]. journal of Anyang institute of technology, 2019, v.18; No.102(06):57-60+112.

[2] Cai mingzhuang, kong xiangren. optimization of swarm intelligence big data based on machine learning and hashing algorithm-taking accurate vehicle detection as an example [J]. electronic world, 2020, no 584 (02): 73-74.

[3] Wan Xiaoyan. Big Data Mining Optimization Algorithm Based on Machine Learning Model [J]. Information and Computer, 2019, 031(021):68-69.

[4] Rodin. Big data analysis and prediction of college employment situation based on machine learning [J]. Wireless Internet Technology, 2020, v.17; No.174(02):120-121.

[5] Peng Chao, Hu Yongxiang, Chen Longfei, et al. Review of drug relocation algorithms based on machine learning and big data mining [J]. Pharmaceutical Progress, 2020, v.44(01):10-15.

[6] Cui Xiaoluo, Luan Xiaofei. Design of image restoration algorithm based on deep reinforcement learning [J]. Internet of Things Technology, 2019(6):58-60.

[7] Shark Wang. Discussion on machine learning algorithm based on big data analysis [J]. Information and Computer (Theoretical Edition), 2019, 422(04):64-65.

[8] Lin Qingxin. License plate recognition algorithm based on big data label and machine learning [J]. Journal of Jixi University, 2019, 019(009):49-54.