International Journal of Database Theory and Application
Vol.9, No.3 (2016), pp.25-36
http://dx.doi.org/10.14257/ijdta.2016.9.3.04

Representative Information Retrieval Algorithm Based on PageRank Algorithm and MapReduce Model

Ling Wei, Yang Li and Yongjiang Wei

School of Management, Harbin University of Science and Technology, Hei Longjiang Harbin150040, China
E-mail: weiling@hrbust.edu.cn

Abstract

Representative information retrieval is widely used in public opinion management, mobile commerce and knowledge management. In the era of big data, the value of representative information extraction is particularly prominent. In order to further explore the application value of representative information extraction, this paper proposes a representative information retrieval method based on PageRank algorithm and MapReduce model (PM-Rep), research on effect of the representative coefficient λ to the extraction scale and the coverage and redundancy of the results, comparative analysis of the advantages and disadvantages of the similar methods. In the parameter experiment, as the λ increases, the scale of the extraction results increases, the coverage degree and the redundancy degree also increases. In the effect experiment, PM-Rep's coverage is significantly better than Top-k, Heuristic and Random, and the redundancy of PM-Rep is the least. In the efficiency experiment, PM-Rep takes the least time in the four methods, embodies the advantages of PM-Rep method for massive data.

Keywords*: representative information, big data, PageRank algorithm, MapReduce model, coverage, redundancy*

1. Introduction

With the rapid development of social networking and mobile intelligent terminal, everyone is producers, disseminators and users of information. Since information acquisition costs decreased, which showing the explosive growth trend. According to a report by CISCO shows that in 2014 the global Internet has produced a new 4.1ZB data, the amount of data in 2020 will reach 40ZB. One form of abundance is accompanied by another form of scarcity. The massive data easily lead to information overload, and make the managers and decision makers of the organization drowning in the data sea. The organization's data mainly includes internal data and external data. Internal data such as database data, CRM data and ERP data, can help organizations plan their daily operations, optimize the organizational structure. External data such as social network data, online reviews and Web search data, can help organizations understand the external environment. Organization management and decision-making can not be separated from the data, but in the face of massive amounts of data it will still be helpless. Managers and decision makers can not view the data, but also want to try to grasp more information. Extracting the sub data sets from the original data set to follow a certain constraint is called information extraction, and users can read the results of the extraction to obtain the required information [1]. If the extracted sub data set can reflect most of the original data, and the redundancy is small enough, the process is called representative information extraction. The core of representative information extraction is to reflect the most content with the least effort, and can help managers and decision makers to get rid of the problem of massive data, master the most content of the original data.

ISSN: 2005-4270 IJDTA

There are a lot of methods to extract information from home and abroad, in which the Top-k method and the information clustering method are more classic. The Top-k method requires the user to set the weights of different attributes, and the system according to the weight returned to meet the requirements of the first K results. The result of Top-k extraction is different from the user's own preference and has a high degree of redundancy, which can't guarantee content coverage of the original information [2]. The information clustering method divides the original data into different categories according to the similarity degree, and requires that the similarity among the classes is as large as possible, and the difference between the classes is as large as possible. This method is mainly found potential similar patterns from a data set and focused on the organization of information, cannot effectively reflect the content of the original information [3]. Information extraction is widely used in the field of remote sensing information, text information and web information. There is little research on representative information extraction, and some scholars have used it to extract the product reviews. Lappas and Gunopulos extract the representative commodity review set from the original commodity review data, with the smallest comment set covering all the original comment set [4]. Tsaparas also considers that the extracted representative comments should contain all the positive and negative evaluation of the attributes [5].

The PM-Rep method combining the thought of PageRank algorithm and MapReduce model. PageRank algorithm in all kinds of algorithms of the application is more extensive, mainly used in the journal influence, Web information search and user recommendation [6-8], in this paper, PageRank is applied to the representation information extraction, which improves the coverage degree of representative information, and reduces the redundancy of representative information. The MapReduce model is designed with divide-conquer method, which is a simple but powerful parallel and distributed computing architecture [9], MapReduce reduces the time of representative information extraction, and improves the ability of the method to deal with large data.

2. Concept and Problem Definition

2.1 Representative Information

Representative information is a collection of small information which can reflect the vast majority of the original data set. Representative information not only requests to reflect the vast majority of the original information, but also the content of representative information collection itself as small as possible. The original data set D and representative data set R to meet such a relationship: R cover the content of D, and the similarity degree to R and D is greatest; the information redundancy of R is minimum, that is, the similarity between the information in R is small enough. Representative information extraction process can be described as follows:

$$Finding \quad R$$
$$s.t.\begin{cases} R \subseteq D \\ \max \ (data\_ \ \text{cov} \ erage(R,D)) \ // \ \text{Maximum coverage of D} \\ \min \ (data\_redu \ ndancy(R)) \ // \ \text{Minimum redundancy of R} \end{cases} \quad (1)$$

2.2 Representative Coefficient and Representative Set

Let D be the set of original data, $D = \{d_1, d_2, ..., d_n\}$, n is the number of data. Representative coefficient refers to the extent of the two data can be represented each other in D. Whether d_i represent d_j is determined by the size of the similarity between

them and the size of representative coefficient λ . The similarity between d_i and d_j record $sim\ (d_i, d_j)$. The data object d is mapped into a feature vector as follows:

$$V(d) = (feature_1, w_1; feature_2, w_2; ...; feature_n, w_n) \quad (2)$$

Calculate the similarity of d_i and d_j as follows:

$$sim\ (d_i, d_j) = \cos\theta = \frac{\sum_{k=1}^{n} w_k(d_i) \times w_k(d_j)}{\sqrt{(\sum_{k=1}^{n} w_k^2(d_i))(\sum_{k=1}^{n} w_k^2(d_j))}} \quad (3)$$

If $sim\ (d_i, d_j) \geq \lambda$, then d_i can represent d_j when representative coefficient is equal to

λ , record $rep_\lambda(d_i, d_j) = 1$; If $sim\ (d_i, d_j) < \lambda$, then d_i can not represent d_j when representative coefficient is equal to λ , record $rep_\lambda(d_i, d_j) = 0$. The representative set is a collection of data represented by the original data set in the case of a given representative coefficient. In the case of a given representative coefficient, all data sets that can be represented by a data d in original data set D are called representative set. given representative coefficient λ and data object d_i , representative set record $D_i^\lambda = \{d_k \mid sim\ (d_i, d_k) \geq \lambda, d_k \in D\}$.

2.3 Problem Definition

Assuming the original data set is D , $D = \{d_1, d_2, ..., d_n\}$,representative coefficient is λ , representative data sets is R , $R = \{d_{r1}, d_{r2}, \cdots, d_{rk}\}$ .Then searching for representative data sets R from the original data set D , R meet the following conditions:

$$\min\ |R|$$

$$s.t.\begin{cases} D_{r1}^\lambda \cup D_{r2}^\lambda \cup ... D_{rk}^\lambda = D \ // \ \mathbf{Coverageconstraint} \\ \sum_{i=1}^{k-1} (\sum_{j=i+1}^{k} sim\ (d_{ri}, d_{rj})) \\ \min(\ \dfrac{}{|R| \times (|R| - 1)}) \ // \ \mathbf{Redundanctonstrai} \end{cases} \quad (4)$$

3. Related Work

3.1 PageRank Algorithm

(1) The basic idea of PageRank algorithm

PageRank algorithm is the link analysis algorithm proposed by Google founder Larry Paige and Sergei Brin, and has gradually become a widely used computational model for the search engine and the academic community. PageRank is a method of Google to identify the importance of web pages, is also the only standard for Google to measure the quality of a web site. In the PageRank algorithm, the link to the page as a vote. A page's votes is determined by the right of other pages link to it, and a hyperlink is equivalent to a

vote.

(2) The steps of PageRank algorithm

PageRank algorithm take the Webpage links as a directed graph G, $G = (V, E)$, V represents a collection of nodes v_1, v_2, …, v_n in the graph. E represents a collection of directed edge. The specific steps of the algorithm are as follows:

1. Give an initial weight for each node, that is PageRank value, PageRank values of node v_i is represented as $P(i)$.

2. Assuming that node v_i to v_j has a directed edge (v_i, v_j), then the node v_i cast a vote to the node v_j.

3. The number of edges from a node v_i is represented by $C(i)$. The vote that node v_i cast to the node v_j is $P(i)/C(i)$ [10].

Assume that all nodes are pointing to v_1, then the node's PageRank is:

$$P(1) = \frac{P(2)}{C(2)} + \frac{P(3)}{C(3)} + \cdots + \frac{P(n)}{C(n)} \quad (5)$$

3.2. MapReduce Model

(1) The basic idea of MapReduce

MapReduce is a programming model proposed by Google , the main idea is "divide -conque", First part, after the whole [11]. The essence of MapReduce model can be summarized as a system to segment large data sets into small chunks. MapReduce model is simple, and many of the problems in reality can be transformed to MapReduce model for processing. Hadoop-MapReduce is a most widely-used model, which is a popular open source and runs on the Hadoop file system (HDFS)—a distributed storage system.

(2) The process flow of MapReduce

MapReduce is composed by function Map and function Reduce. The function Map parallel processing each bit of data sets, function Reduce collect the results. Function Map constructs the input data into a pair of key and value, <key，value>, and sort by key. The function Reduce merge the <key，value> with same key[12]. In HDFS, a large file is split into a number of large and small pieces of fixed size, and the distribution between the computer. MapReduce is a framework that can operate directly on these files, and its implementation process is shown in Figure 1.

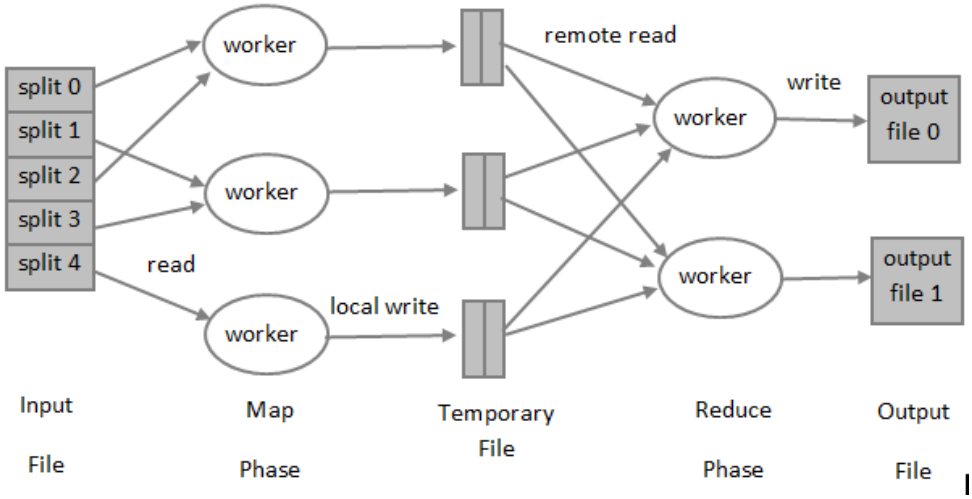

Figure 1. The Flow Chart of Mapreduce Programming Model

4. PM-Rep Algorithm

4.1 PM-Rep Algorithm Based on PageRank

(1) The Step of PM-Rep

Paper propose a heuristic method based on the PageRank algorithm to find the representative information set. With the basic idea of PageRank, the data d in D vote to the corresponding representative set in each round. Finally, the data's representative set has highest number of votes, then put the data into representative data set R. The steps of the representative information extraction based on PageRank are as follows:

Step1: Calculate the representative set. Calculate the similarity matrix M according to the original data set D, $D = \{d_1, d_2, \cdots d_n\}$. According to the representative coefficient λ get the representative matrix M_λ: if the similarity be equal or greater than λ in M, then the similarity value is obtained as an integer 1; if the similarity less than λ in M, then the similarity value is obtained as an integer 0. Finally, all the representative set D_i^λ are obtained according to the representative matrix M_λ.

Step2: Calculate the votes of representative set. The initial PageRank for d in D is 1, if d_j in D_i^λ, then $rep_\lambda(d_i, d_j) = 1$, d_j has the qualification to vote for D_i^λ. Calculate the number of d_j in the representative set, record n_j, then the number of votes cast by d_j for its representative set is $vote_j^i = \frac{1}{n_j}$. Finally, calculate the vote of representative set D_i^λ in each round, the formula is as follows:

$$vote_i = \sum_{d_j \in D} vote_j^i \times rep_\lambda(d_i, d_j) = \sum_{d_j \in D} rep_\lambda(d_i, d_j) / n_j \qquad (5)$$

Step3: Optimized redundancy. According to the votes of each representative set, to find the representative set who's votes in $[\alpha \cdot \max(vote_j^i), \max(vote_j^i)]$, if d_j meet the following conditions, put it into representative data set R.

$$\begin{cases} vote_j \in [\alpha \cdot \max(vote_j^i), \max(vote_j^i)] \\ \min\{\frac{1}{|R|} \times \sum_{d \in R} sim(d_j, d)\} \end{cases} \qquad (6)$$

Step4: Remove the d_j and the data d_j can represent from D, a new data set D and a representative set D_i^λ are obtained. Repeat 2, 3 steps until no more data is needed to join R.

(2)The example of PM-Rep

The similarity matrix M is obtained by similarity calculation form a raw data set , as follows:

$$\mathbf{M} = \begin{array}{c|cccccccc} & d_1 & d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 \\ d_1 & 1.00 & 0.91 & 0.22 & 0.88 & 0.31 & 0.34 & 0.89 & 0.26 \\ d_2 & 0.91 & 1.00 & 0.26 & 0.88 & 0.93 & 0.29 & 0.92 & 0.34 \\ d_3 & 0.22 & 0.26 & 1.00 & 0.31 & 0.89 & 0.45 & 0.33 & 0.47 \\ d_4 & 0.88 & 0.88 & 0.31 & 1.00 & 0.25 & 0.23 & 0.95 & 0.29 \\ d_5 & 0.31 & 0.93 & 0.89 & 0.25 & 1.00 & 0.90 & 0.19 & 0.94 \\ d_6 & 0.34 & 0.26 & 0.45 & 0.23 & 0.90 & 1.00 & 0.37 & 0.47 \\ d_7 & 0.89 & 0.92 & 0.33 & 0.95 & 0.19 & 0.37 & 1.00 & 0.35 \\ d_8 & 0.26 & 0.34 & 0.47 & 0.29 & 0.94 & 0.47 & 0.35 & 1.00 \end{array}$$

Let the representative coefficient $\lambda = 0.8$, After 0, 1 standardization, the representative matrix $\mathbf{M}_\lambda$ as follows:

$$\mathbf{M}_\lambda = \begin{array}{c|cccccccc} & d_1 & d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 \\ d_1 & 1.00 & 1.00 & 0.00 & 1.00 & 0.00 & 0.00 & 1.00 & 0.00 \\ d_2 & 1.00 & 1.00 & 0.00 & 1.00 & 1.00 & 0.00 & 1.00 & 0.00 \\ d_3 & 0.00 & 0.00 & 1.00 & 0.00 & 1.00 & 0.00 & 0.00 & 0.00 \\ d_4 & 1.00 & 1.00 & 0.00 & 1.00 & 0.00 & 0.00 & 1.00 & 0.00 \\ d_5 & 0.00 & 1.00 & 1.00 & 0.00 & 1.00 & 1.00 & 0.00 & 1.00 \\ d_6 & 0.00 & 0.00 & 0.00 & 0.00 & 1.00 & 1.00 & 0.00 & 0.00 \\ d_7 & 1.00 & 1.00 & 0.00 & 1.00 & 0.00 & 0.00 & 1.00 & 0.00 \\ d_8 & 0.00 & 0.00 & 0.00 & 0.00 & 1.00 & 0.00 & 0.00 & 1.00 \end{array}$$

According to the representative matrix $\mathbf{M}_\lambda$, get the representative set as shown in Table 1:

Table 1. Representative Set

Representative set(λ)	Data object
D_1^λ	{d_1, d_2, d_4, d_7}
D_2^λ	{d_1, d_2, d_4, d_5,d_7}
D_3^λ	{d_3, d_5}
D_4^λ	{d_1, d_2, d_4, d_7}
D_5^λ	{d_2, d_3, d_5, d_6, d_8}
D_6^λ	{d_5, d_6}
D_7^λ	{d_1, d_2, d_4, d_7}
D_8^λ	{d_5, d_8}

Firstly calculate the number of data in each representative set. For example, the number of d_1 in representative set is 4, record $n_1 = 4$, then every vote of d_1 is $\frac{1}{4}$. Secondly calculate the total vote of each representative set. For example, the vote of D_5^λ : $vote_5 = \frac{1}{n_2} + \frac{1}{n_3} + \frac{1}{n_5} + \frac{1}{n_6} + \frac{1}{n_8} = 1.9$. Calculate the number of votes from the D_1^λ to D_8^λ : 0.95, 1.15, 0.7, 0.95, 1.9, 0.7, 0.95 and 0.7. Because the votes of D_5^λ is the biggest, put d_5 into R . Remove d_2 , d_3 , d_5 , d_6 , d_8 from D , get new data set $D = \{d_1, d_4, d_7\}$, repeat the above steps until $D = \phi$. Finally, get the final representative data set $R = \{d_5, d_1\}$ or $\{d_5, d_4\}$ or $\{d_5, d_7\}$

4.2 The Pseudo Code of PM-Rep Algorithm

Input: Data Object Sets $D = \{d_1, d_2, \cdots d_n\}$, Specified Threshold $\lambda, \alpha \in [0,1]$

Output: Representative Data Sets R

Setup ()

$R = \phi$;//*Statement of representative data sets*

$M[\ ][\ ] = Compute\_Similarity(D)$;//*Calculate the similarity between data*

$M_\lambda[\ ][\ ] = Compute\_Represent(\mathbf{M})$;//*Calculate the representation matrix*

While $D \neq \phi$ do{

$avgsim = 1$;//*The initial redundancy is defined as 1*

$D^\lambda[\ ] = Compute\_Representative\_sets(\mathbf{M}_\lambda)$;// *Calculate representative set*

$Vote[\ ] = Compute\_vote\_value(\mathbf{M}_\lambda, D_\lambda)$;//*Calculate the votes*

$Max\_vote\_value = Find\_\max(vote)$;//*Find the maximum number of votes*

For d_i in D {

if($vote[i] \in [\alpha \times Max\_vote\_value, Max\_vote\_value]$){

if($Compute\_avg\_sim(d_i, R) < avgsim$){ //*Redundancy optimization*

$avgsim = Compute\_avg\_sim(d_i, R)$;

$d_r = d_i$;}}}

$R = R + d_r$;

$D = D - D^\lambda[d_r]$; }

$Output(R)$;

End;

4.3 Combine the MapReduce Model

In the second step of the PM-Rep algorithm paper introduce the MapReduce Model to greatly improve the ability and efficiency of dealing with massive data. Paper use the Table 1 to illustrate the process of calculating the representative information.

(1) Pre-processing phase

According to the characteristics of MapReduce, construct the pair of <key, value>. Let the representative data d_i for the Key , and the Value is made up of the data which is represent- -ed by d_i. For example, in representative set D_1^λ, the data which can be represented by d_1 is d_1, d_2, d_4 and d_7. So d_1 is key, value is (d_1, d_2, d_4, d_7). Table 1 to Table 2 after the pre-processing as follows.

Table 2. Output Format for Preprocessing Phase

Input	Key	Value
Map1	d_1	(d_1, d_2, d_4, d_7)
Map2	d_2	(d_1, d_2, d_4, d_5, d_7)
Map3	d_3	(d_3, d_5)
Map4	d_4	(d_1, d_2, d_4, d_7)
Map5	d_5	(d_2, d_3, d_5, d_6, d_8)
Map6	d_6	(d_5, d_6)
Map7	d_7	(d_1, d_2, d_4, d_7)
Map8	d_8	(d_5,d_8)

(2) Map phase

In Table 2, 8 Map is required to complete the following. The Map process needs to load the driver and initialize, leading to the need to consume a lot of resources, so reducing the number of calls can reduce system resource consumption. A large number of complex data calculation is on the Map, Reduce is only done a simple data statistics, which can improve the efficiency of the system. Each pair Key-Value is processed on the Map, calculate the number of each representative set, record n, then the PageRank value of the data node is $\frac{1}{n}$. For example$<d_1 \rightarrow d_1, d_2, d_4, d_7>$, after the Map process, several new Key-Value are obtained:$<d_1 \rightarrow 1/4>$, $<d_2 \rightarrow 1/4>$, $<d_4 \rightarrow 1/4>$and $<d_7 \rightarrow 1/4>$. All the processing results are shown in Table 3.

Table 3. Output Format for Map Phase

Input	Output
Map1	$d_1 \rightarrow 1/4$; $d_2 \rightarrow 1/4$; $d_4 \rightarrow 1/4$; $d_7 \rightarrow 1/4$
Map2	$d_1 \rightarrow 1/5$; $d_2 \rightarrow 1/5$; $d_4 \rightarrow 1/5$; $d_5 \rightarrow 1/5$; $d_7 \rightarrow 1/5$
Map3	$d_3 \rightarrow 1/2$; $d_5 \rightarrow 1/2$
Map4	$d_1 \rightarrow 1/4$; $d_2 \rightarrow 1/4$; $d_4 \rightarrow 1/4$; $d_7 \rightarrow 1/4$
Map5	$d_2 \rightarrow 1/5$; $d_3 \rightarrow 1/5$; $d_5 \rightarrow 1/5$; $d_6 \rightarrow 1/5$; $d_8 \rightarrow 1/5$
Map6	$d_5 \rightarrow 1/2$; $d_6 \rightarrow 1/2$
Map7	$d_1 \rightarrow 1/4$; $d_2 \rightarrow 1/4$; $d_4 \rightarrow 1/4$; $d_7 \rightarrow 1/4$
Map8	$d_5 \rightarrow 1/2$; $d_8 \rightarrow 1/2$

(3) Reduce phase

Reduce receives all Map output <key, value>, calculate the sum of Value with the same Key. For example, calculate the vote of the representative set D_5^{λ}, the vote as follows:

$$vote_5 = \frac{1}{5} + \frac{1}{2} + \frac{1}{5} + \frac{1}{2} + \frac{1}{2} = 1.9 \quad (7)$$

5. Experiment Results and Analysis

5.1 Experiment Environment

Build a Hadoop cluster with 15 servers in the Linux environment, the Hadoop platform version number is 2.2. A server as the master node and the remaining 14 servers as slave node, each node's processor use the Intel's core i5 with 4GB of memory. The operating system is Ubuntu12.04, the JDK version is JDK Sun 1.7. We use 30 UCI standard test data set, and mark U_1, U_2... $U_{30.}$ Then we have carried on the parameter experiment, the effect experiment and the efficiency experiment.

In order to facilitate the measurement of the degree of coverage and redundancy, $coverage = \frac{|R|}{|D|}$, $redundancy = \sum_{d \in R}(1 - 1/\sum_{d' \in R} sim(d, d'))/|R|$.

5.2 Experiment Results and Analysis

(1) Parameter experiment

In order to research the effect of the representative coefficient λ on the scale, the coverage, the redundancy and the approximate degree of the optimal solution, we have

carried out experiments on different values of λ, and the results are shown in Figure 1 and Figure 2. From the graph, we can see that when λ<0.5, it has no effect on the size of the result, the coverage and the redundancy. When λ >= 0.5, on the size of the result, the coverage and the redundancy increase with the increase of λ.

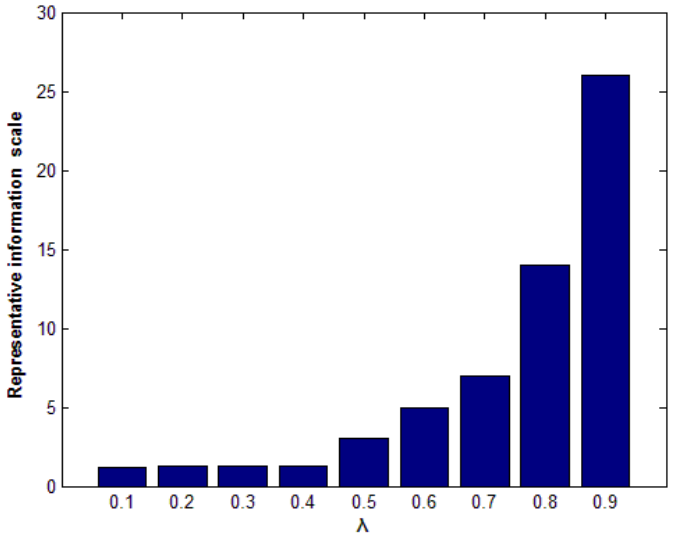

Figure 2. The Scale of the Extraction Results Under Different Λ

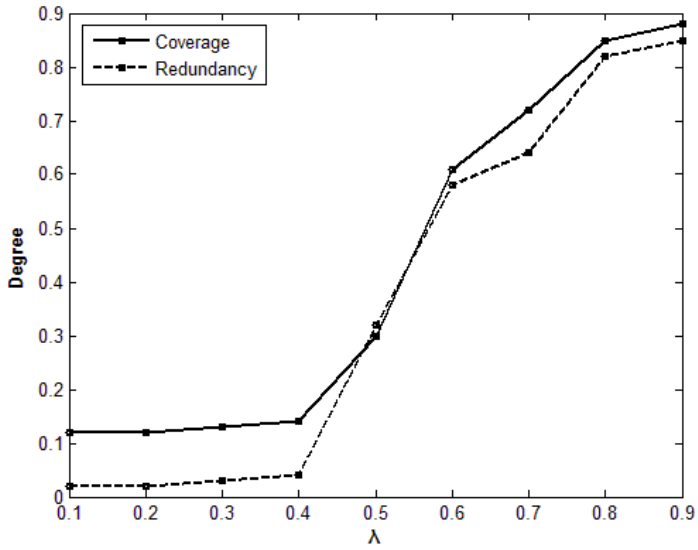

Figure 3. Coverage and Redundancy Under Different λ

(2) Effect experiment

In order to observe the effect of PM-Rep method in extracting representative information, this paper makes a contrast experiment. Under the same data set and parameter λ, the representative information is extracted by Top-k, Heuristic, Random and PM-Rep, and the results are as shown in Figure 4 to Figure 6. In the four method, the PM-Rep's average coverage is the largest, which means that PM-Rep can cover the original data well. In addition PM-Rep's average redundancy is the smallest and significantly better than the Top-k and random method, mean less than heuristic, which

means that PM-Rep in line with the intention of representative information extraction.

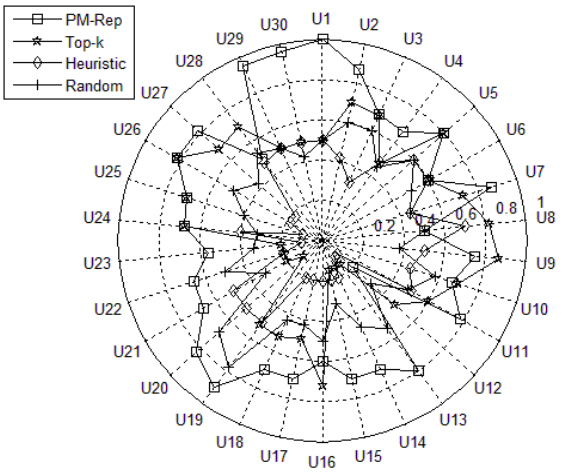

Figure 4. The Coverage of the Representative Data Set

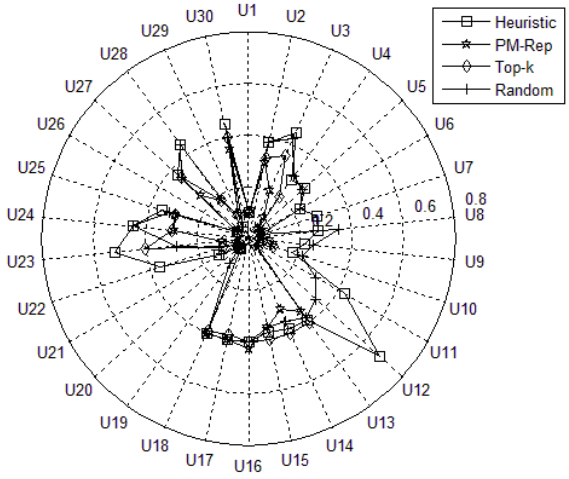

Figure 5. The Redundancy of the Representative Data Set

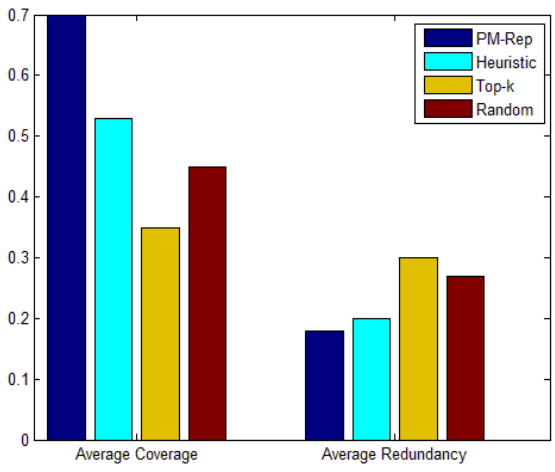

Figure 6. Average Coverage and Redundancy

(3) Efficiency experiment

In order to verify the ability of PM-Rep to handle large data, the execution time of four methods in different data sizes is recorded, as shown in Figure 7. PM-Rep algorithm has less running time than Top-k, Heuristic and Random, and it shows it's obvious advantages.

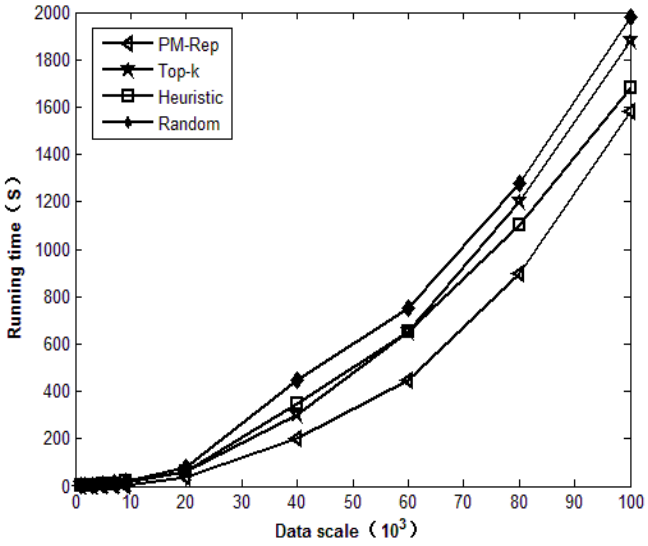

Figure 7. Running Time on Different Scale Data

6. Conclusion

This paper propose a new PM-Rep algorithm based on PageRank algorithm and MapReduce model for representative information retrieval, then experiment and test this algorithm in the Hadoop cluster environment. Experiments show that compared with the other three methods, the PM-Rep algorithm has the maximum coverage and minimum redundancy of the original data when dealing with massive data.In combination with the MapReduce model of distributed processing characteristics, PM-Rep have stronger data processing capacity than the Top-k, heuristic and random method. MapReduce model help PM-Rep reduce execution time and improve the efficiency. In the era of explosive growth of data, PM-Rep algorithm integrate distributed computing power, and improve the efficiency of information extraction in big data. PM-Rep provides a strong support for the further research on the efficient implementation of representative information extraction, and has good research value and application prospect.

Acknowledgment

The Research supported by the National Natural Science Foundation of China (No.71272191, No.71072085).

References

[1] H. Cunningham, "Information Extraction, Automatic", Encyclopedia of Language and Linguistics, 2nd Edition, no. 5, **(2006)**, pp. 665-667.
[2] R. Fagin, "Combining fuzzy information from multiple systems", Journal of Computer and System Sciences, vol. 58, no. 1, **(1999)**, pp. 83-99.
[3] R. Xu and D. Wunsch II, "Survey of clustering algorithms", IEEE Transactions on Neural Networks, no. 16, **(2005)**, pp. 645-678.

[4] T. Lappas and D. Gunopulos, “Efficient confident search in large review corpora”, In ECML/PKDD, no. 2, **(2010)**, pp. 195-210.
[5] P. Tsaparas, A. Ntoulas and E. Terzi, “Selecting a comprehensive set of reviews”, In KDD, **(2011)**.
[6] F. Ma, “Research on the influence of journals based on PageRank algorithm”, Journal of Information, vol. 33, no. 12, **(2014)**, pp. 104-106.
[7] Z. Qing, L. Zhang and N. Li, “Application of improved PageRank in Web information collection”, Journal of Computer Research and Development, vol. 43, no. 6, **(2006)**, pp. 1044-1049.
[8] L. Zhang and X. X. Yan, “A comparative analysis of the performance of the collaborative filtering algorithm based on user's recommendation”, Library and Information Service, vol. 58, no. 2, **(2014)**, pp. 215-216.
[9] J. Dai and Z. M. Ding, “MapReduce Based Fast kNN Join”, Chinese Journal of Computers, vol. 38, no. 1, **(2015)**, pp. 100-102.
[10] C. Chen, Y. W. Zhan and Y. Li, “PageRank parallel algorithm based on Web link classification”, Chinese Journal of Computer Application, vol. 35, no. 1, **(2015)**, pp. 48-52.
[11] J. Dean and S. Ghemawat, “MapReduce: Simplified data processing on large clusters”, Communications of the ACM, vol. 51, no. 1, **(2008)**, pp. 107-113.
[12] R. Karim, A. Hossain and M. Rashid, “A MapReduce Framework for Mining Maximal Contiguous Frequent Patterns in Large DNA Sequence Datasets”, IETE Technical Review, vol. 29, no. 2, **(2012)**, pp. 162-168.

