Advances in Computer Science Research, volume 63
International Conference on Artificial Intelligence and Engineering Applications (AIEA 2016)

Implementation of Parallel CASINO Algorithm Based on MapReduce

Li Zhang [a], Yijie Shi [b]

State key laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, 100876, China

[a]lizh9121@163.com, [b]yijieshi2000@bupt.edu.cn

Abstract. In recent years, with the rapid development of social network, deeply analyzing and mining the social network users is of great significance in the area of information propagation and advertising, and there have been increasing research efforts on it. In this paper, we analyze the CASINO algorithm deeply and propose a parallel CASINO algorithm based on MapReduce, and then test the performance of the algorithm under different conditions. The experimental result shows that compared to CASINO, the parallel CASINO algorithm has a better performance and a faster speed.

Keywords: CASINO algorithm; MapReduce; Parallel computation; Hadoop.

1. Introduction

The social network is a form of expression which indicates the relationship among people in the network [1]. And the influence of the social network users is based on this relationship, according to a certain rule that can make the abstractive influence digitized. So far the main methods which can calculate the influence of the social network users are as follows:(1)the PageRank[2-3] algorithm and the HITS [4] (Hypertext-Induced Topic Search) algorithm which are based on links in network. Many researchers used the thought of the PageRank algorithm when measuring the influence of the social network users while the thought of the HITS algorithm is not used so much. (2) Algorithms based on user behavior such as the number of microblogs, comments or likes, can be used as indexes to compute the influence. This is intuitive and has certain rationale, but the establishment of the relationship between users in the social network is somewhat accidental and the strength of the relationship between different users is also diverse. So sometimes these algorithms can't reflect the real influence of users.

CASINO algorithm [5] calculates the Influence and Conformity under different topics from the perspective of sentiment analysis, and it also firstly introduces the Conformity into measuring the influence of the social network users. The result fits the reality very well and the index of Influence and Conformity calculated by CASINO algorithm shows a high degree of accuracy. Applying this algorithm on the most popular social networks such as Weibo and Facebook, we can analyze the similarity and the difference of the users, and their characteristics in various social networks can also be found.

With the development of social networks, the user data of the whole network is getting larger and larger. If we apply the serial CASINO algorithm on only one computer, it would be inefficient and cost too many resources. In this paper, based on these problems we choose the MapReduce framework on the Hadoop platform to implement the distributed computing which makes CASINO algorithm parallel, and analyze the modified algorithm from aspects of performance and extendibility. We also compare the modified algorithm with the serial CASINO algorithm.

2. Related Work

2.1 CASINO Algorithm.

The CASINO algorithm can be divided into two parts: Data preprocessing and computation of influence and conformity. In data preprocessing stage, we first divide the social network into several subnetworks based on topic. For every subnetwork, each social network user is regarded as a node, and the relationship between users is converted to a directed edge. In this way, each subnetwork is transformed into a graph. Then, from the perspective of emotion analysis, the emotional differences between different users are analyzed. When the emotional difference between the two users is small, it is said that their opinions are similar. When when the emotional difference between the two users is great, it is said that their opinions are different. So we can judge whether a user trusts or distrusts another one. Based on this, we can label every directed edge as positive or negative one. In computation of influence and conformity stage, this is the real beginning of the algorithm. According to the above results, we initialize influence and conformity of all the users to 1, and then compute it based on a certain rule. In the computation process, influence index and conformity index affect each other. The result will not be output until it converges.

The basic formulas of the algorithm are as follows:

Influence Index:

$$\Phi(u) = \sum_{\overrightarrow{uv} \in E^{+}} \Omega(v) - \sum_{\overrightarrow{uv} \in E^{-}} \Omega(v) \tag{1}$$

Conformity Index:

$$\Omega(u) = \sum_{\overrightarrow{uv} \in E^{+}} \Phi(v) - \sum_{\overrightarrow{uv} \in E^{-}} \Phi(v) \tag{2}$$

U and v are nodes in the network, E^{+}presents the edge labeled as positive, and E^{-} presents the edge labeled as negative. From the basic formulas of the algorithm, we can see that the influence index of one user equals that the sum of influences of users who trust this user minus sum of influences of users who distrust this user .While the conformity index of one user equals that the sum of conformities of users who are trusted by this user minus sum of conformities of users who are distrusted by this user. After several rounds of iteration, the result will converge, so we can get the influence index and conformity index of every user.

The complete calculation procedure is as follows:

```
Input: G(V, E)= $G^{+}$(V, $E^{+}$)∪$G^{-}$(V, $E^{-}$)
Output: the influence index I= (Φ(d1), Φ(d2), …, Φ(dn)) and conformity index
C=(Ω(d1), Ω(d2),…, Ω(dn))for V={d1,d2,…,dn}
 begin
    m=1
    for each u<V do
      Φ(d)=Ω(d)=1
    while I or C not converged do
      for each u<V do
        $\Phi_{0}^{m+1}(u) = \sum_{\overrightarrow{uv} \in E^{+}} \Omega^{m}(v) - \sum_{\overrightarrow{uv} \in E^{-}} \Omega^{m}(v)$
        $\Omega_{0}^{m+1}(u) = \sum_{\overrightarrow{uv} \in E^{+}} \Phi^{m}(v) - \sum_{\overrightarrow{uv} \in E^{-}} \Phi^{m}(v)$
      for each u<V do
        $\Phi^{m+1}(u) = \frac{\Phi_{0}^{k+1}(u)}{\sqrt{\Sigma_{v \in v} \Phi_{0}^{m+1}(v)^{2}}}$
        $\Omega^{m+1}(u) = \frac{\Omega_{0}^{k+1}(u)}{\sqrt{\Sigma_{v \in v} \Omega_{0}^{m+1}(v)^{2}}}$
    $I^{k+1} = (\Phi^{m}(u_{1}), \Phi^{m}(u_{2}), \dots, \Phi^{m}(u_{n}))$
    $C^{k+1} = (\Omega^{m}(u_{1}), \Omega^{m}(u_{2}), \dots, \Omega^{m}(u_{n}))$
```

G^{+}(V, E^{+}) is the subnetwork which contains all the positive edges, and G^{-}(V, E^{-}) is subnetwork which contains all the negative edges. *I* represents a collection of the influence index of all users. *C* represents a collection of the conformity index of all users. From the basic formulas of the algorithm, we can see that the conformity index of one user depends on the conformities of users who are trusted or distrusted by this user. Similarly, the influence index of one user depends on the influences of users who trust or distrust this user.

2.2 Hadoop.

Hadoop is an open source project under the Apache foundation, which provides a convenient distributed computing framework for users. HDFS (distributed file system) and MapReduce engine are the most core part of Hadoop. The data is processed in a parallel way, so it has high efficiency; multiple copies of a file are copied and stored on a plurality of nodes, which can be recovered from other nodes when a node breaks down, so it is reliable. Hadoop is used to distribute data among the available computers in the cluster and to complete the calculation task, and the cluster can be easily extended to thousands of nodes and thus has high scalability. Because of these advantages, Hadoop is widely used by many companies, such as Amazon, Facebook and so on.

2.3 MapReduce.

MapReduce is a parallel software architecture first proposed by Google, and it can deal with the parallel computation with large-scale datasets [8, 9]. MapReduce includes two parts: Map and Reduce. Each task is also divided into two stages: map stage and reduce stage. In each stage, input and output are in the form of <key, value>. Map generates an intermediate result based on the input. The MapReduce framework will pass the same key value generated by Map-to-Reduce function. Reduce function can receive only one key and a set of value for processing at the same time, and thus produce the final result. MapReduce hides the implementation details of bottom layer, and offers programmers highly abstractive interface, which provides much convenience to the programmers. Hadoop-MapReduce is composed of Job Tracker (Master) and Task Trackers (Workers): Job Tracker is responsible for receiving the user's task, and divides the task into map tasks, then assigns it to Task Trackers, and reports to user when all tasks are completed. Each Task Tracker can handle map tasks as well as reduce tasks, and the scale can be set.

Hadoop-MapReduce execution process is as shown in Fig. 1:

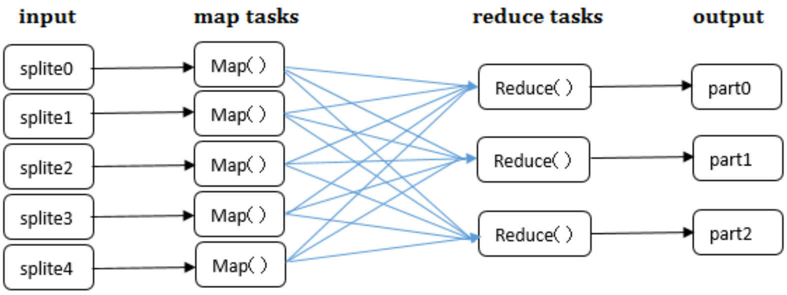

Fig. 1 Overview of MapReduce

3. CASINO Algorithm Based on MapReduce

3.1 Algorithm design.

In each round of iteration of the CASINO algorithm, the calculation of the conformity index and influence index of each node are not affected by each other, so it is suitable for Map-Reduce framework, and various nodes can perform Map tasks or Reduce tasks in the calculation process of the algorithm. Every iteration of parallel CASINO algorithm based on MapReduce is a MapReduce process, so designing Map and Reduce functions is the core of improved algorithm.

The format of data for the Map is (x, Ω(x) , Φ(x) , outbound links, inbound links) .x, Φ(x) and Ω(x) represent node x, its influence index and conformity index. Outbound links represent a set of directed edges emitted by the node x, inbound links represent a set of directed edges pointing to node x, and edges may be labeled as positive as well as negative. Map accepts input in this format, firstly outputs (x, links) as the input of Reduce, and then judges the relationship between current node and any node y in the outbound links is positive or negative. It will output (y, b, Φ(x)) if it is positive, otherwise output (y, b, -Φ(x)). That means donating conformity index of the current node to all the nodes in the outbound links as their influence index, and inbound links are similar. The second field of the output is the symbol of conformity index and influence index, where a presents influence index and b presents conformity index, and then entering the Reduce process, all the data line with the same key value will be received and processed by a Reduce process. In this Reduce process, we can obtain the influence index of a node and the sum of the conformity index.

Map function pseudo code is as follows:

```
Map(key, value){
    //key is node x in the network
            //value is($\Phi$(x), $\Omega$(x), outbound links, inbound links)
        Output (x,  outbound links, inbound links)
  for each y in outbound links{
     if y is positive
       output (y, b, $\Phi$(x))
     else
       output (y, b, -$\Phi$(x))
     }
        for each y in inbound links{
     if y is positive
       output (y, a, $\Omega$(x))
     else
       output (y, a, -$\Omega$(x))
              }
      }

Reduce(key, value){
     //key is node x in the network
     //value is list(a/b, $\Omega$(x)/$\Phi$(x))or links
   {
         $\Phi$(y)= The sum of  $\Omega$(x) whose second field is a in the list; //Symbol a presents influence index
         $\Omega$(y)= The sum of  $\Omega$(x) whose second field is b in the list; //Symbol b presents conformity index
     output(y,  $\Phi$(x), $\Omega$(x), outbound links, inbound links);
   }
```

After each round of iteration of MapReduce, the influence index and conformity index should be normalized. Then it will go to the next turn until influence index and conformity index of all nodes tend to be stable.

3.2 Algorithm analysis.

Map process of the parallel CASINO algorithm is to produce influence index that a certain node receives from others and the conformity index that other nodes give to a certain node. Reduce process is to accumulate all of the influence index and conformity index of a node to generate the final results. The time complexity of the single-machine serial CASINO algorithm is related to the number of nodes and the number of edges because the algorithm needs to calculate the influence index and conformity index of each node, and it also needs to find all edges that point to the node. While parallel CASINO algorithm can deal with several nodes at the same time, which depends on the number of map/reduce tasks that the cluster can be process simultaneously, and when the amount of data is huge, the algorithm can greatly reduce the time and improve the efficiency.

4. Experimental Results and Analysis

4.1 Experimental environment.

The operating environment of single-machine is shown in Table 1. The CASINO algorithm is implemented by using Java.

Table 1 The operating environment of single-machine

cpu	Intel Core i5-3470 3.2GHz
ram	4g
hard disk	1TB
operating system	win7 ultimate 64-bit

In the LAN environment, we use 4 sets of PC machines to build a multi-node cluster test environment, and choose one of them as the master node and the other three as slave nodes. The operating environment of every node is as follows:

Table 2 The operating environment of the machine in the cluster

cpu	Intel Core i5-3470 3.2GHz
ram	4g
hard disk	1TB
operating system	Cent OS 6.5
Linux kernel	2.6.32-431.el6.i686 GNOME 2.28.2
version of Hadoop	2.4.1
version of JDK	jdk1.7.0_45

4.2 Experimental data.

We use the dataset of hot topic of Micro-blog, choose the most discussed topic #Running Man# on Weibo and obtain the Micro-blogs and their comments. We collect a total of 316212 nodes: the relationship among them is "@" or "forward", which means if user A and user B both comment on a topic ,and A can @ B, and then we analyze their emotions; if A’s emotion is close to B, the edge of A to B is positive, otherwise negative. In this way, we can set up a labeled social network and transform the dataset into data with 316212 rows.

4.3 Experimental results and analysis.

4.3.1 Running time of the parallel CASINO algorithm for different number of slave nodes

We run parallel CASINO algorithm based on #1, #2, and #3 slave nodes respectively, and measure the running time of an iteration of MapReduce using the micro-blog dataset. The results are as follows:

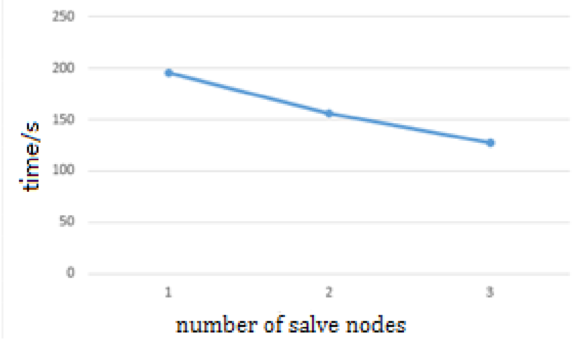

Fig. 2 Running parallel CASINO algorithm based on slave nodes

As shown in Fig. 2, the more the slave nodes are, the less time will be spent. Because we use more slave nodes, more map task and reduce task can be processed at the same time. The overall CPU and storage and other resources are fully utilized, so better results can be obtained, which shows that the algorithm has good scalability and extensibility.

4.3.2 Contrast experiments between serial CASINO algorithm and parallel CASINO algorithm

We divide the dataset collected from micro-blog into the 100k-row dataset and 300k-row dataset. For micro-blog data with different sizes, we measure the running time of a round of iteration of serial algorithm to calculate influence index and conformity index and time of a Map-Reduce of the parallel algorithm respectively. The result is shown in Fig. 3:

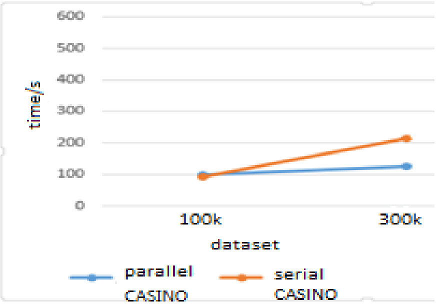

Fig. 3 Contrast experiments

When the scale of the data is small, for some reason like interaction between cluster nodes, the parallel algorithm takes more time than the serial algorithm. When the scale is large, the parallel algorithm has more advantages in handling large amounts of data, namely when the scale of data reaches to 300,000 rows, the efficiency of parallel CASINO algorithm is higher than serial CASINO algorithm. It can be easily predicted that the larger the scale of the data is and the more obvious advantages of parallel algorithms will have.

5. Conclusion

In the field of social networks, the CASINO algorithm is representative based on users' emotion among various algorithms that measure the influence of the social network users. In this paper, we proposed a parallel CASINO algorithm based on MapReduce and analyzed its performance under different scales of cluster, which proved that the algorithm has good expansibility. We also compared the parallel CASINO algorithm with the serial CASINO algorithm and proved that the parallel algorithm has an obvious advantage in dealing with large scale of data. So parallelizing the CASINO algorithm to adapt to the growing scale and increasing amount of computing of social network is of great significance.

References

[1] Li H, Cui J T, Ma J F. Social Influence Study in Online Networks: A Three-Level Review [J]. Journal of Computer Science & Technology, 2015, 30(1):184-199.

[2] Kamvar S D, Haveliwala T H, Manning C D, et al. Extrapolation methods for accelerating PageRank computations [C]// International Conference on World Wide Web. ACM, 2003:261-270.

[3] Zhi-Ying L I. Research on PageRank Algorithm [J]. Computer Science, 2011.

[4] Huang Y M. Web Structure Mining and Analysis of HITS Algorithms [J]. Computer & Modernization, 2007.

[5] Li H, Bhowmick S S, Sun A. CASINO: Towards conformity-aware social influence analysis in online social networks [C]// ACM Conference on Information and Knowledge Management, CIKM 2011, Glasgow, United Kingdom, October. 2011:1007-1012.

[6] Singh N, Agrawal S. A review of research on MapReduce scheduling algorithms in Hadoop [C]// International Conference on Computing, Communication & Automation. IEEE, 2015

[7] Kala Karun A, Chitharanjan K. A review on hadoop — HDFS infrastructure extensions [C]// Information & Communication Technologies. 2013:132-137

[8] Jiang W X, Zhang J, Wang Z M. Study on Parallel Programming Framework Model Based on MapReduce [J]. Microelectronics & Computer, 2011, 28(6):168-167.

[9] Jian-Jiang L I, Jian C, Dan W, et al. Survey of MapReduce Parallel Programming Model [J]. Tien Tzu Hsueh Pao/acta Electronica Sinica, 2011, 39(11):2635-2642.