Applying Ant Colony Algorithm for Page Ranking Calculation

CHONAWAT SRISA-AN
Faculty of Information Technology
Rangsit University
52/347 Phaholyothin Rd, Lakhok Pathumthani, 12000
Thailand
chonawat@yahoo.com

Abstract: - Nowadays almost search engines use classical keyword – based method in order to query but almost search engines just match keyword within database and return the result. Many times search engines return result for thousands pages and may be few pages that meet the requirements. This old method makes poor relevance information. Generally page rank index method used for search engine output sorting developed at Stanford University, that seem high correlation between high PageRank index and general page importance judged by human users, but this method has two drawbacks. First is necessary to access entire web structure to perform proper computation and the second is takes time to calculate entire data. This paper proposes applying Ant Colony algorithm for page ranking called Ant Colony Ranking (ACR) algorithm to reduce drawback of the old one by use artificial ants to construct the path and use standard page ranking to calculate page rank of web structure. From experiment express that Ant Colony Ranking is faster than standard page rank method when number of page is increasing rapidly.

Key-Words: - Ant Colony algorithm, Page ranking, search engine, Internet, Artificial Ant, Web Structure, Ant Colony Ranking.

Introduction

Nowadays almost search engines use classical keyword – based method to query the question in search engine web site. Many situations of users that search engine retrieve thousands (or more) pages of output but unfortunately almost of them are junk because of irrelevance information that get from search engines. PageRank indexing successfully used for search engine output sorting. There seems to be quite high correlation between high PageRank index and general page importance judged by human users [1]. There are two drawbacks of page rank, first is necessity to have access to entire of web structure to perform proper computation and secondly is CPU speed consuming to calculate all data. If network is huge, researcher need high CPU speed to calculate whole data. Goal of this paper is use Ant Colony Ranking (ACR) algorithm to reduce two drawbacks as mention above by use ants to find ** ... ** tag of each page (or links of each pages) and go to that web site following information in ** ... ** tag and repeat previous step until meet condition in section 4, then ant store path information in database and construct path. Finally compute page rank by formula (1) and destroy ants. The experiment indicates that ACR algorithm will faster than standard page rank in long term.

1 Standard Page Rank Equation

PageRank is one of the methods that Google uses to determine a page's relevance or importance [2]. Standard page rank equation is developed at Stanford University [3] and has standard equation as shown in equation (1).

$$PR(A) = (1-d) + d\left(\frac{PR(T_1)}{C(T_1)} + \frac{PR(T_2)}{C(T_2)} + \ldots + \frac{PR(T_n)}{C(T_n)}\right) \quad (1)$$

Which:

PR (A) = Page Rank of web site A

d = Damping Factor which can set among 0 to 1 (generally set to 0.85[2]).

$T_1 - T_n$ = Page *1* to *n* that point to page A

(1-d) = Probability that the random surfer get bored[3].

PR (T_n) = Page rank of page that linking to page A.

C (T_n) = Number of outgoing for page T_n.

$\frac{PR(T_n)}{C(T_n)}$ = Back link from page T_n

Example:
Find page rank of page A, B, C, D [2]

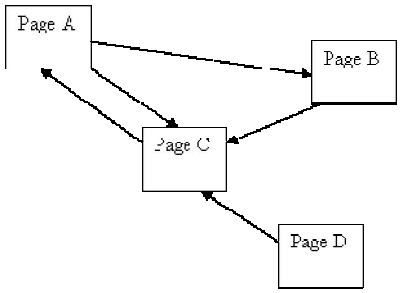

Fig 1 Example of page rank calculations

Back link from page C to page A or term $\left(\frac{PR(T_n)}{C(T_n)}\right)$ of page A = c then,

PR (A) = 1-d + d(c) (2)

Back link from page A to page B or term $\left(\frac{PR(T_n)}{C(T_n)}\right)$ of page B = a/2 then,

PR (B) = 1-d + d(a/2) (3)

Back link from page A, B and D to page C or term $\left(\frac{PR(T_n)}{C(T_n)}\right)$ of page A, B and D = a/2 + b + d then,

PR (C) = 1-d + d(a/2 + b + d) (4)

Because of no back link from any page to page D or term $\left(\frac{PR(T_n)}{C(T_n)}\right)$ of page D = 0 then,

PR (D) = 1-d + d(0) (5)

I use equation 2, 3, 4 and 5 to calculate page rank of this example by using Java code below.

```
class PageRank {
        public static void main(String[] args){
        double a = 0,b = 0,c = 0,d = 0,
        i = 90.0,
        damp = 0.85, k = 1;
        while (i >= 1){
        a = (1 - damp) + damp*c;
        b = (1 - damp) + damp*a/2;
        c = (1 - damp) + damp*(a/2 + b + d);
        d = (1 - damp);
System.out.println (" Round " + k + "[" + a  +
"]" + "["+b + "]" + "[" + b + "]" + "[" + d +
"]");
        i--; k++; } //end of While
System.out.println (" a = [" +  a  + "]");
System.out.println (" b = [" +  b  + "]");
System.out.println (" c = [" +  c  + "]");
System.out.println (" d = [" +  d  + "]");
}// end of Method Main
} // end of Class PageRank
```

```
Round 87.0[1.490107405313735][0.7832956472!
Round 88.0[1.490107405313736][0.7832956472!
Round 89.0[1.490107405313736][0.7832956472!
Round 90.0[1.490107405313736][0.7832956472!
a = [1.490107405313736]
b = [0.7832956472583378]
c = [1.5765969474279249]
d = [0.15000000000000002]
```

Fig 2 Some part of output of program

From Fig. 2 the value of page rank must converge to stable value (in this example round 87). For this example just only 4 pages, if calculate all web structure (around) 1.5 billons pages the researcher need CPU speed as much as possible to calculate all pages and it will occurs billions iteration to repeat calculations.

2 Ant Colony Ranking algorithm

Ant Colony Ranking (ACR) algorithm is a system based on agent who simulate the natural behavior and cooperative of ants. ACR based on following rules[1].

- Ants will find the first page to crawl and go to that page.
- Ants find <*a href* = "..."> ... </*a*> tag and record its data to ant's memory. After that ants will drop pheromone on <*a href* = "..."> ... </*a*> tag and created itself equal with number of <*a href* = "..."> ... </*a*> tag. Finally each ant crawl to *each* <*a href* = "..."> ... </*a*> tag on that page.
- After ants crawl on new page each and will find <*a href* = "..."> ... </*a*> tag and do previous step again until
  - Page that ants visit don't have any <*a href* = "..."> ... </*a*> tag.
  - Ant's memory full.
- After ants visit all web structure it will come back to home and store information of all ants in database and ants will be destroyed.

3 Ant Colony Ranking Software

In order to reduce two drawbacks of page rank, this research use Ant Colony Ranking instead of web crawler and adapt some ACR algorithm express in section 3 to fit with Ant Colony Ranking software. ACR software has following algorithm to work.

- ACR software read (or load) web page in main memory of computer (in this experiment use www.sanook.com for the first page). It will be fit with computer's memory.
- ACR software find and count number of <*a href* = "..."> ... </*a*> tag in that file to create session of ants. If loaded file has not any <*a href* = "..."> ... </*a*> tag ACR software will drop that page and load other page instead.
- Each ants go to web site follow links in <*a href* = "..."> ... </*a*> tag and download that page to memory.
- When ants visit to any link in <*a href* = "..."> ... </*a*> tag, ants will record in cache (or ants memory) and drop pheromone to tell another ants to neglect this path. The meaning of *"Drop pheromone"* in this situation ACR software insert "***" sign before and end of <*a href* = "..."> ... </*a*> tag. For example ***<*a href* = "..."> ... </*a*>***.
- Ant will do previous step again if it found "***" sign it will ignore that link and go to next link until
  - Page that ant visit don't have any <*a href* = "..."> ... </*a*> tag.
  - Completely number of loop defined in ACR software. ACR software determines loop of iteration because of Internet is very big and has many pages (around 1.5 billion pages) if don't determine loop researcher must waiting for long time until all ant tour all Internet (in ACR software set iteration to 60,000)
  - Ant's memory full (in ACR software set ant's memory to1 MB).
- After finished finding information in previous step, ants will return all information to ACR software and save to database.
- ACR software will collect information of all ants and construct path and destroy ants.
- ACR software will use equation (1) to find page rank and export output in coordinate of number of click (or link) and number of page in CSV (Comma Separate Value) format.
- Use spreadsheet program to plot graph as shown in Fig 3.

Note:

- Ant will load each page to computer, do "drop pheromone" and another action on computer not on that website.
- This figure below is average of 20 experiment output.

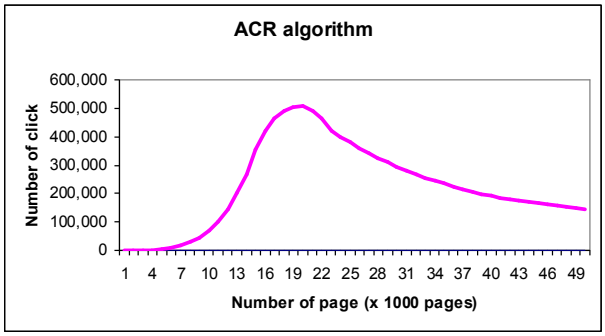

Fig 3 Average output of Ant Colony Ranking

From Fig 3, This graph plotted from coordinate x, y by x-axis is number of page that ants tour (in this case about 50,000 pages), y-axis is number of click (or number of ** ... ** tag) in each page of website that ants tours. The experiment shows that in the initial state number of ants are low when number of pages is increasing number of ants are high, ants makes number of click high because of in this stage probability that ants found pheromone is low. If number of page more than (about) 23,000 pages number of click is reduce because of probability of each ants found pheromone (or "***" sign) is high.

4 Conclusion

Ant Colony Ranking (ACR) algorithm built to reduce two drawbacks of standard ranking method, web structure and large iteration. Ants built whole web structure after ants come back to home (in real experiment, all action does on the computer not on website). Generally ants will less time spending than standard page rank because of number of ants will increase non linear when number of page increase linearly. That means ACR algorithm save the time to construct the web structure in long term.

Moreover ACR reduce large iteration by use ants to crawl to web structure and when number of page is increasing, the probability of ants that found "*Pheromone dropped*" is high, then probability that ants found path will decreasingly. It makes number of click (or ** ... ** tag) is slightly. That means ACR algorithm will reduce number of page that must calculate.

References:

[1] Chonawat Srisa-an, "Page ranking Calculation using Ant Colony algorithm", Rangsit University,2000

[2] Ian Rogers, " The Google Page rank Algorithm and How It Works", IPR Computing Ltd.,2000

[3] Wikipedia," Page rank", http://en.wikipedia.org/wiki/PageRank#PageRank_uses_links_as_.22votes.22

[4] T. Bray. "Measuring the Web". *The World Wide Web Journal*, 1996.

[5] S. Brin and L. Page. "The Anatomy of a Large-Scale Hypertextual Web Search Engine", Elsevier Science, April 1998.

[6] J. Cho, H. Garcia-Molina, and L. Page. Efficient Crawling Through URL Ordering, Elsevier Science, April 1998.

[7] Google, http://google.stanford.edu/,

[8] Cynthia Dwork, Ravi Kumar, Moni Naor, and D. Sivakumar. "Rank aggregation methods for the web", 2001.