ISSN: 2321-7782 (Online)
Volume 2, Issue 12, December 2014
International Journal of Advance Research in Computer Science and Management Studies
Research Article / Survey Paper / Case Study
Available online at: www.ijarcsms.com

Dynamic Assignment of Users to Distributed Servers in Social Networking Application

Aher Dipeeka Vijay[1]
Vishwabharti Academy's College of Engineering,
Ahmednagar, Maharashtra
Savitribai Phule Pune University
Ahmednagar, Maharashtra - India

Prabhudeva Irabashetti[2]
Assistant Professor
Vishwabharti Academy's College of Engineering,
Savitribai Phule Pune University
Ahmednagar, Maharashtra - India

Abstract: ***The issue of dynamic assignment of users to servers is studied widely but the proposed system substantiates the solution for this problem. The system of Dynamic Assignment of users to servers in social networking like application will resolve the extra load that the servers handling currently. The distributed algorithm which will be used in this system will also contain node failure policy and acknowledgement policy which will be very useful for optimal assignment of users. This will leads to the proposed system which can adopt the future dynamics. Along with previously mentioned features, we provide some major security features in terms of Authentication, which will consists Personal data feedback as 2nd level security. In this way the proposed system will bound to give the results which will handle the overload of different distributed servers and provide two level Authentication system which will give more convenience to the users as well as to the system.***

Keywords: Distributed Algorithms, Online Social Network, dynamic assignment of users, optimal assignment of users.

I. INTRODUCTION

Online Social Network (OSN) like Facebook, Twitter runs the largest social networking platform that serves hundreds of millions users at peak times using tens of thousands of servers located in many data centers around the world. There are strict operational requirements on Online Social Network (OSN) platform in terms of performance, reliability and efficiency, and to support continuous growth the platform needs to be highly scalable. For any Online Social Network (OSN) each user profile and its data are stored at its primary servers. the two designated servers Su and Sv communicate with each other on behalf of their users.

This indirect communication architecture enables rich application functionalities that is often difficult otherwise. This indirect communication architecture requires additional resources for handling the inter-communication among servers. In what follows, we will elaborate more on the user server assignment problem.

In many large scale systems, the problem of client-server assignment with a specific objective of reducing the total load while maintaining a good load balance among the servers is crucial. We assume the load at the servers is due mainly to handling messages/information passing among users.

We have some observations in Online Social Network (OSN).

1) The amount of load generated by messages exchanged between two users assigned to the same server will be less than that of when the users are assigned to different servers. This is because of local communication incurred less load than non-local communications.

2) One cannot assign all users to one server, when he/her do so this server might be overloaded, resulting in degraded performance. Also, from the robustness perspective, this assignment is prone to bottleneck failures.

3) It is beneficial to assign two users that exchange messages with each other often to the same server in order to minimize the overall communication load. On the other hand, because of the load balance consideration, two users that rarely exchange messages with each other should be assigned to two different servers.

One distributed algorithm is required to achieve optimal assignment of users to distributed servers in Online social networking (OSN) application, so that this algorithm can run on multiple servers at a time exploit local information at every server during their search for the optimal assignment[1][2]. The exploit of information basically required when the servers exchange and update information about their loads iteratively among themselves.

II. Related Work

The solution for dynamic assignment of users to distributed servers in Online Social networking applications is need to be tempered with some security features. Generally client server assignment problem is compare with k-way graph partitioning problem. In that G=(V,E) V is vertices and E is edges, whereas V nodes should be portioned with k subset such that weight of each server(node) must be balanced.

This problem is NP Hard and any existing way is not giving exact solution, so here we tried to focus on balancing the weights on different distributed servers with node failure policy. The graph node partitioning problem studied in many areas such as VLSI (very large scale integration) and many heuristic algorithms was developed like Kerninghan-Lin Algorithm[5]. There are many approaches used like simulated annealing[6][7][8], genetic algorithm[9], community structure based method[10]. These all methods is used to exploit structure specific to certain application. Many existing applications are also focusing on inter server message flow to reduce load on cluster. While considering quality of elements at each cluster some algorithms does not consider effect of edge cuts on weight of element(node). While referring different algorithmic solutions. They work on task scheduling and graph partitioning but focuses on size of each partition at the time of managing weights of edge cut.

Our algorithmic solution emphases through distributed algorithm execution for distribution of load at server which are affected by inter-server communication. There is no existing algorithm which balances load on distributed servers with minimum inter-server communication. This proposed algorithmic solution is substantiated with node failure and acknowledgement policy.

There are some existing online Social Network Database applications like Cassandra, but with all accumulated features our work mostly matches with Nishida & Nguyen in Some Context.

III. Working Scheme

The proposed algorithm is supposed to work in distributed environment so that servers can share information of allocation of users iteratively. There are two major approaches in optimal user assignment, first is iterative and another can be considered as concurrent. Iterative algorithmic approach can be more beneficial for sharing summarized data. Servers can also exchange their user At each iteration, based on the exchanged information. Pair of servers then decide to move their users in between appropriately. each server must update information about its assigned users and their immediate neighbors only.

Following factors are taken into consideration for distributed algorithm

1) What information is to be computed and kept at the servers?
2) What information is to be exchanged among the servers?
3) At every time step, how to decide which pairs of servers are involved in moving their users?
4) Within this chosen pair of servers, which users are to be moved?

The performance of Centralized Simulated Annealing (CSA) algorithm is used as a baseline for evaluating distributed algorithms. The CSA algorithm is based on the simulated annealing (SA) framework is highly effective in solving many large scale combinatorial optimization problems. The CSA algorithm is a type of stochastic greedy search in which the probability of searching in the next configuration is based on the objective value at the current and the next configuration.

IV. CONCLUSION

In this paper, we have proposed an iterative algorithm for optimal user assignment to distributed servers. The different interconnected servers shares summarized data for exchange of users. After exchange of users servers also exchange their adjacent nodes. Inter-server communication is basically targeted in proposed algorithm to balance the load of all servers. This algorithmic solution is substantiated with node failure and acknowledgement policy.

ACKNOWLEDGMENT

This paper work is completed successfully only because support from each and every one including teachers, colleague, parents and friends. Especially, I am very thankful to those who provide me guidance and make this work reachable. This paper work is supported by my senior, my teachers and some experienced personalities. My acknowledgment of gratitude towards the everyone who make this work reachable.

References

1. C.M.Fiduccia and R. Mattheyses, “A linear time heuristic for improving network partitions,” 19th Design Automation Conference, 1982.
2. J. Zien, M. D. F. Schlag, and P. Chan, “Multilevel spectral hypergraph partitioning with arbitrary vertex sizes,” Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on, vol. 18, no. 9, pp. 1389–1399, 1999.
3. P. Chan, M. D. F. Schlag, and J. Zien, “Spectral k-way ratio-cut partitioning and clustering,” Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on, vol. 13, no. 9, pp. 1088–1096, 1994.
4. J. Shi and J. Malik, “Normalized cuts and image segmentation,” IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 22, No. 8, pp. 888-905, 2000.
5. B. Kernighan and S. Lin, “An efficient heuristic procedure for partitioning graphs,” Bell Systems Technical Journal, 1970.
6. D. S. Johnson, C. Aragon, L. McGeoch, and C. Schevon, “Optimization by simulated annealing: An experimental evaluation,” Operations Research, 1989.
7. S. Kirkpatrick and M. P. Vecchi, “Optimization by simmulated annealing,” science, vol. 220, no. 4598, pp. 671–680, 1983.
8. C. A. D.S. Johnson and L. MacGeoch, “Optimization by simulated annealing: An experimental evaluation part i: Graph partitioning,” Operations research, vol. 37, no. 6, Nove. 1989.
9. T. N. Bui and B. R. Moon, “Genetic algorithm and graph partitioning.” IEEE Trans. on Computers, 1996.
10. M. Newman, “Modularity and community structure in networks.” In Proc Natl Acad Sci USA (2006).

AUTHOR(S) PROFILE

Aher Dipeeka Vijay, Received his B.E in Computer Engineering from Vishwabharti Academy’s College of Engineering, Ahmednagar of Savitribai Phule Pune University, Maharashtra (India). She is now Student of ME (Computer) in Vishwabharti Academy’s College of Engineering, Ahmednagar of Savitribai Phule Pune University, Maharashtra (India).

Prabhudeva Irabashetti, Received his B.E in Information Science and Engineering from BLDEA’S College of Engineering of Visvesvaraya Technological University, Karnataka(India). He has also received M.Tech in Computer Science and Engineering from BDT College of Engineering of VTU, Karnataka(India). He is having 6 years Teaching experience and currently working in Vishwabharti Academy’s College of Engineering, Ahmednagar, Maharashtra (India).

