ISSN:0975-9646

Designs of Parallel Glowworm Swarm Optimization Tool Using Map Reduce

Dr.R.N.Kulkarni
Dept of CSE
BITM, Bellary

Aishwarya H.M
Dept of CSE
BITM,Bellary

ArunKumar
Dept of CSE
BITM, Bellary

Deeksha Patil
Dept of CSE
BITM,Bellary

Diksha Jain .P
Dept of CSE
BITM,bellary

Abstract:In the last few decades managing large data has become challenging task because of the increasing volume and complexity of the data being created or collected. The problem here is how to effectively manage and analyze the data and resulting information. The solution requires a comprehensive approach that contains all the stages from the initial data collection to its final analysis. clustering is one of the solution which is the fundamental technique in data mining that is based on grouping a set of objects in such a way that objects in the same group called cluster are more similar to each other than to those in other clusters. Traditional clustering techniques do not address all the requirements adequately.
In this paper, we propose Map Reduce Clustering Glowworm Swarm Optimization (MRCGSO) that provides scalable environment that handles large data sets and achieves a good speedup and utilization when more computing nodes are used. MRCGSO uses the MapReduce methodology for the parallelization since it provides fault tolerance, load balancing and data locality.

Keywords: Data mining, Parallel processing, Apache hadoop.

INTRODUCTION

Clustering large data is gaining significant importance from past few years in several fields like science, business, Climate and so on. A clustering algorithm is said to be efficient if the similarity measure between objects in the same cluster are to be maximized and similarity measure between objects in different cluster are to minimized.

In recent years, some researchers discussed clustering based on the idea of swarm. Swarm intelligent algorithms have self organizing features developed to share information among swarm members to find the best solution. Swarm intelligence is one of the nature inspired algorithm based on the behavioral observation of swarm. There is no main member in the swarm. All members of swarm equally participate in achieving the task. There are many swarm intelligent algorithms and one among them is Glowworm swarm Optimization (GSO) for which we apply clustering and hence is known as CGSO.

GSO is inspired by behavior of insects called glowworms. These glowworms emit light (luciferin) which can be used for multiple purposes e.g., when glowworms go in search of food, a glowworm which finds the food emits more light indicating other worms that it has found the food .All other glowworms are move towards glowworm whose luciferin level is higher than its own within a local decision range. At the end, most glowworms is gathered at the many peak locations in the search space.

Thus GSO solves the optimization task in a way, wherein each glowworm searches for one of the cluster centroids, which is considered as a sub-solution. The combination of these sub-solutions forms the global solution of the clustering problem.[1].As volume of data is increasing, it is computationally expensive to handle it serially.

In this paper, we propose a scalable, parallel design of glowworm swarm optimization clustering [2] using the MapReduce methodology called MRCGSO. MRCGSO is different from CGSO since it implements the Map and Reduce functions in order to achieve the goal. MapReduce is a prominent parallel data processing framework, which has been gaining significant interest from both industry and academia. It is a new methodology proposed by Google in 2004, which is a programming model and an associated implementation for processing large datasets [3].

1. LITERATURE SURVEY

A parallel K-means algorithm clustering algorithm based on MapReduce was proposed in [4]. The algorithm finds the centroids by calculating the weighted average of each individual cluster points through the *Map* function; afterwards the *Reduce* function assigns a new centroid for each data point based on the distance calculations. Then, a MapReduce iterative refinement technique is applied to find the final Centroids.

In [5],a parallel genetic algorithm was proposed using the MPI library on a Beowulf Linux Cluster with the master slave paradigm.

In [6], an MPI based parallel particle Swarm optimization algorithm was introduced. However, MPI is not the best choice for parallelization because of the weakness of having to handle the failure of nodes.

In [7], MRPSO incorporated the MapReduce model to parallelize particle swarm optimization by applying it on computationally data intensive tasks. The authors presented a radial basis function as the benchmark for evaluating their MRPSO approach, and verified that MRPSO is a good approach for optimizing data-intensive functions.

In [8], the authors proposed a MapReduce based ant colony approach. They show how ant colony optimization can be modeled With the MapReduce framework. They designed and implemented their algorithm using Hadoop.

Comparing our proposed algorithm to the algorithms listed above, Map Reduce implementations were used to optimize single objective functions, whereas in our proposed algorithm, the algorithm searches for multiple maxima for multimodal functions. To the best of our knowledge, MR-CGSO is the first the parallelization of glowworm swarm optimization.

2. TERMINOLOGY

Data mining: Data mining is a process of exploring patterns in huge data sets by converting it into useful information. There are many data mining techniques such as association rule mining, data classification, and data clustering.

Parallel Processing: Parallel processing is a processing in which many calculations are carried out simultaneously [9], working on the principle that large problems can often be divided into smaller ones, which are then solved at the same time.

Apache Hadoop: Apache Hadoop [10] is an open source framework which is the commonly used for map reduces implementation and supports data-intensive applications (pet bytes of data) licensed under apache. It consists of two main components: Hadoop Distributed File System (HDFS), which is used for data storage, and Map Reduce, which is used for data processing.

3. PROPOSED METHODOLOGY

In our proposed methodology for implementation of MRCGSO, we have considered the input as student database which contains various details regarding examination marks scored by each students in each subject.

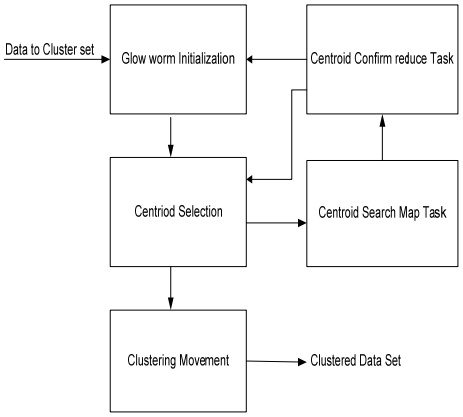

Fig: Block diagram

INPUT: Student dataset containing internal marks, external marks and total

OUTPUT: Clustered results based on marks

Notations:

In MRCGSO, each glowworm *gj* in the swarm has the following information:

- Luciferin level (*Lj)* carried by glowworm gj
- Fitness function value (*Fj)*
- M-dimensional position vector (*pj)*
- Coverage set size (*crj)*, which is the number of the data Instances that are covered by *gj*
- Intra-distance (*intraDj)*, which is the distance between the *crj* set members and the *gj* position
- Local decision range (*rd*), which is the range of the Glowworm to find the covered data records within this Range, and to find the glowworm's neighbors.

The steps are performed iteratively until optimized cluster is obtained

1 [Glowworm initialization]

Generate random position vector pj for each glowworm within search space between minimum and maximum

Fj=0

the luciferin level and local decision range are initialized.

Numbers of clusters and iterations to be performed are specified here

2 [Centroid selection]

In this step map and reduce functions are implemented. The map function takes the input dataset. Dataset is partitioned based on the number of mappers used. Each Mapper then calculates the sub-covered records (crj) and the sub-intra-distances (intraDj)

$$\text{IntraDj} = \sum_{i=1}^{|crj|} \text{distance}(crj, gj)$$

The reduce function takes intermediate output from map function. It calculates the sum of sub-coverage and sub-intradistances from all mappers to find the glowworm's coverage and intra-distance for the Whole data set. Where the coverage is the number of data instances Covered by the glowworm based on the whole data set, and the Intra-distance is the sum of distances between the glowworm and the covered data instances from the whole data set.

3 [Clustering movement]

This step reads the reducers Output and calculates the fitness of each glowworm, then continues the same CGSO process of updating the glowworm Position based on its glowworm neighbors.

$$SSE = \sum_{j=1}^{k} \sum_{i=1}^{|cj|} (\text{Distance}(xi, cj))^2$$

F (gj) =1/n |crj|/SSE*intra Dj/mix j(intra Dj)

The luciferin value is updated using the equation

$$Lj(t) = (1 - \rho)\, Lj(t-1) + \gamma\, F(pj(t)) \quad (1)$$

Where $(t-1)$ is the previous luciferin level for glow-wor*m j*;

P is the luciferin decay constant ($\rho \in (0, 1)$); γ is the luciferin Enhancement fraction and F (pj (t) represents the objective Function value for glow-wor*m j* at current glow-worm position (Pj); t is the current iteration.

Example

Input:

Sl. No	Internal marks	External marks	Total
1	16	75	91
2	15	44	59
3	15	37	52
4	15	21	36
5	22	73	95
6	17	40	57
7	24	80	104
8	23	65	88
9	21	64	85
…	…	…	…

Output:
Students who scored above 79 are in cluster1
Students who scored above 58 are in cluster2
Students who scored above 50 are in cluster3
Students who scored below 50 are in cluster4

Sl. No	Internal marks	External marks	Total	Student belonging to cluster
1	16	75	91	1
2	15	44	59	2
3	15	37	52	2
4	15	21	36	4
5	22	73	95	1
6	17	40	57	3
7	24	80	104	1
8	23	65	88	1
9	21	64	85	1
…	**…**	**…**	**…**	**…**

CONCLUSION:

In this paper, we have proposed a automated tool for clustering the data using MRCGSO method. The proposed tool is tested for its correctness and completeness using engineering college students data.

REFERENCES:

[1] Nailah Al-Madi, Ibrahim Aljarah and Simone A. Ludwig" parallel glowworm swarm optimization using map reduce"Dept of cse North Dakota State University Fargo, ND, USA

[2] I. Aljarah and S. A. Ludwig, "A new clustering approach based on glowworm swarm optimization." in IEEE Congress *on Evolutionary Computation*. Mexico, Cancun: IEEE, pp. 2642–2649, 2013.

[3] J. Dean and S. Ghemawat, "Map Reduce: simplified data processing on large clusters," in *Proceedings of the 6thconference on Symposium on Operating Systems Design &Implementation - Volume 6*, OSDI'04, pp. 10–10, 2004.

[4] Z. Weizhong, M. Huifang, and H. Qing, "Parallel k-means Clustering based on MapReduce," in *Proceedings oftheCloudCom'09*. Berlin, Heidelberg: Springer-Verlag, optimization algorithm accelerated by asynchronous evaluations. Journal of Aerospace Computing, Information, and Communication, 2005.

[5] M. Ismail. Parallel genetic algorithms (PGAs): master slave paradigm approach using MPI. In E-Tech 2004, pages 83 87, July 2004.

[6] G. Venter and J. Sobieszczanski-Sobieski. A parallel particle swarm Optimization algorithm accelerated by asynchronous evaluations. Journal of Aerospace Computing, Information, and Communication, 2005.

[7] A. McNabb, C. Monson, and K. Sappi. Parallel PSO using MapReduce. In IEEE Congress on Evolutionary Computation, pages 7{14, Sept. 2007.

[8] B. Wu, G. Wu, and M. Yang. A MapReduce based ant colony optimization approach to combinatorial optimization problems. In Natural Computation (ICNC), 2012 Eighth International Conference on, pages 728 {732, May 2012.

[9] Gottlieb, Allan; Almasi, George S. (1989). Highly Parallel Computing Redwood City, Calif.: Benjamin/Cummings. *ISBN 0-8053-0177-1*.

[10] (2012) Apache Software Foundation, Hadoop Map Reduce [Online]. Available:http://hadoop.apache.org/mapreduce.