Multi-Perspective Ant Colony Optimization for Mining and Understanding the Topology Oriented Big Data

Khalid. Hiba, Qamar. Usman, and Hameed. Mazhar

***Abstract*— Big data has the potential to transform how data can be used to manipulate, gather, collect, enforce and contribute towards data centric applications. Data can have meaning and big data has meaning hidden inside it which can be used to empower multiple applications. Big data has its challenges; this massive amount of data can be used to extract a few lines or a few million lines that actually serve the purpose and meaning. For such outputs the algorithmic techniques needs to be devised that can efficiently extract meaning from a topological shape based on the nature of application and the information that is being looked for. The search for hidden meaning in topology of data is a challenge. Many techniques are employed to discover the hidden meaning and patterns in the data. This paper presents an algorithm for mining the clustered topological data using ant colony optimization.**

***Index Terms*— Big Data, Topology, Clustering, Ant Colony, Optimization**

I. Introduction

DATA size and the influx rate of data has changed over a few years, data has taken a new shape and meaning depending upon associations, reference to shape objects, influx, out flux and many other factors. Big data does not only mean involvement of structured text or any single type of data. The data coming in could be structured, unstructured or semi-structured. Depending upon the type and structure of data the meaning and information extracted from it changes. Big data cannot be structured and resolved into database tables; the information is too much and cannot be organized in simple or dynamic tables. The three main concerns of big data involve the scale, motion and transform. Big data does not involve only one type or media of data, the data could be in any form or shape. It is even possible that one type of application can exhibit different shapes of data at different times. This shape is important in extracting the knowledge we are looking for in a data.

Data classification and clustering is very important for the purpose of extracting information from it. Traditional methods include k-means clustering, hierarchical clustering algorithms, self-organizing maps and many other efficient algorithms. However today's data cannot be clustered or classified using these algorithms, because the problems are no longer scalable thus the solutions need to be adjustable and comprehendible based on scalability and dynamicity of the applications and data influx. The implementation of big data structure requires some big data platform for its support such as Hadoop.

Ant colony optimization is one of the techniques that can be efficiently used to extract the information or more precisely mine the data from huge and massive data sets [1]. Ant colony optimization works on the principle of studying artificial systems that come or resemble the patterns of real life ant colonies. This behavior is then used to understand, analyze and solve discrete quantization problems. The big data applications have a lot of data which needs to be both gathered and scattered. This defines the shape our data would take up. Every time the data shape changes there is a probabilistic measure that examines how the information might have changed due to data shape shift.

For such purposes we need to define a customized algorithm for problems involving big data handling. Each topology in a single dimension can be mined for a known or unknown purpose using the ACO algorithms [2].

II. Problem Statement

Many applications today require for imparted and existent human intelligence without any human interaction. This need of artificially intelligent systems have led to discovery of many solutions that work towards developing artificially intelligent systems. These systems and technologies now face a shift from manageable instructions and data to unmanageable data and dynamic requests and instructions. These dynamic and data intensive applications involve many factors which need to be looked into before extracting information from these gathered collection sets. For the

Manuscript received Jan 02, 2017; revised Mar 28, 2017.

Miss Hiba Khalid did her MS from Department of Computer Engineering, College of Electrical & Mechanical Engineering, National University of Sciences and Technology (NUST), Pakistan. (e-mail: hb.khalid92@gmail.com)

Dr Usman Qamar, is the head of Knowledge and Data Engineering Research Centre (www.kdrc.live) at Department of Computer Engineering, College of Electrical and Mechanical Engineering, NUST, Pakistan. He has done his MS in Computer Systems from UMIST, UK whereas his MPhil, PhD and Post-Doc are from University of Manchester, UK in Data Engineering. His expertise are in Data and Text Mining, Expert Systems, Knowledge Discovery and Feature Selection. (e-mail: usmanq@ceme.nust.edu.pk).

Mr Mazhar Hameed did his MS from Department of Computer Engineering, College of Electrical & Mechanical Engineering, National University of Sciences and Technology (NUST), Pakistan. He is also a faculty member at HITEC University, Taxila, Pakistan. (e-mail: mazharhameed91@gmail.com)

purpose of using minimum memory, space and complexity and displaying useful information we need to use algorithms that can work in parallel with huge sets of data and still deliver the results in minimum time with most probable best achievable results. Thus using ACO and its similar transformations; The topology can be observed and manipulated to fit for purpose.

III. LITERATURE REVIEW

Big data storage and analytics not only have the architectural methodology but moreover it has great prominence in terms of analytics. Big data applications and their storage has always been a theme of interest. How so much data can be analyzed and stored without certain inferences. Flash Storage, which is highly accessible and supports multi-access for storage in big data analytics. Many graph analytics [5] have been suggested to accomplish standard level of big data storage. The major techniques for managing big data such as IBM's Hadoop [6] and Google's Map reduce [7] have been deployed and used to serve the purpose.

IV. METHODOLOGY

The Ant Colony Optimization is a metahueristic algorithm that comprises of many distributed computation concepts along with a positive feedback systems. It is the ACO's art for finding out the optimal solutions for combinatorial optimization problems. The algorithms motivation was based on real life ant behavior; the algorithm imparts that behavior and tries to find the most optimal solution as ants could find the shortest path among many routes for food gathering from their nest to food source. Similarly the algorithm can work for mining data from a topological data or for placing data into topology or for extracting data from a topology.

The figure 1 explains the behavior of ants and ACO algorithm double bridge experiment. In figure 1 part (a) the ants are exploring the double bridge in part (b) the ants have now realized for a path that is shortest based on locating the optimal solution between the food source and starting point or nest. Finding the shortest path is the property of ant colonies thus this trait can be used to both add and manipulate data in topologies for mining, retrieval or storage.

The stigmergic property of ant colony optimization is utilized in data shape shift. This property indicates that if one entity changes the environment then the other entity responds to the shift or change in environment at a later time. This property of ACO algorithm is used to study the data shape shifts and how information is manipulated extracted and understood.

The main purpose off analyzing the shape of data is to gather useful information from it. Based on this the ACO algorithm has its ants labeled as "agents". Each ant is an agent with these three characteristics

a. The ant which is an agent chooses a town or destination to visit based on a probabilistic function
P(F) = f(no of tracks, connecting distance)

b. No of revisits is, a list of nodes in case of topological data

c. On completion of a tour, leave a trail on the visited edge.

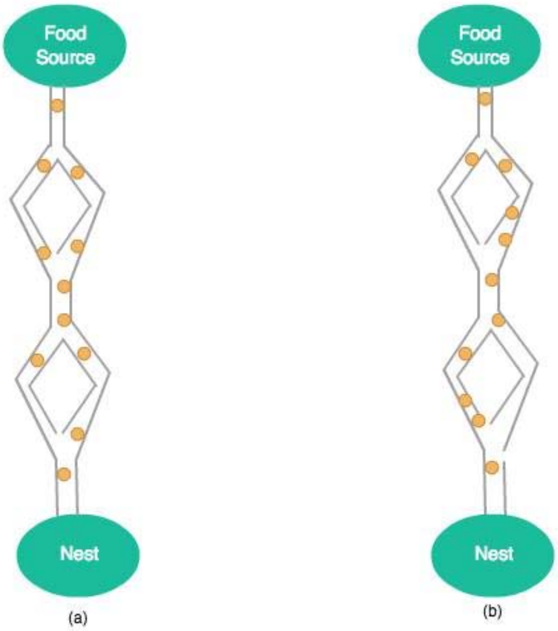

Fig1. Bridge Demonstration

These are the summarized steps of and colony algorithm

```
BEGIN
Initialize the pheromone trails and parameters;
Generate population of (m) to (m+l) solutions;
For each individual ant [k belongs to m]:
calculate (fitness (k));mn

Determine the best global ant;
Update pheromone trail;
Check if termination – TRUE
END
```

For topological data the algorithm would work in the same way but with different adjusted criteria's such as the collective perception of ants can be incorporated along with pheromone guidance to establish a faster mechanism of locating data, object or objects needed. The state transition rule is the stage where an ant decides to move to next stage or node, for data centric application these stages can be categorized and changed depending upon the situation. The two stages or exploitation and exploration are always incorporated. For topological mining the following alteration is suggested

```
BEGIN

Compute lower bound
Compute upper bound
Initialize with primal variable values For (k=l), m (m
=    no of ants) do; Repeat;
Compute data nodes in shape
Calculate probability
Choose next node of data
Append data to ant's TABU list
```

```
    Until ant has computed "K"
    Compute local minimum
  End for;
  For each ant move compute, waste, energy and lengths of runs;
  Update (trails);
  Validate
  END
```

Each time a shape territory is defined the edges and connection points can be rendered for further use. This point collection of more precisely node collection can be regarded as place for pheromone for ants. Every topologically independent graph can also have linearly or quadratic related function setts and similar shortest paths, but each graph may or may not be completely probability free. For each graph that can possibly have a same path might not return the same information. And for each subset the falls in the parent topology might have n-paths returning different results with same pheromone tracks and lengths.

The data nodes in a particular data shape can be expressed using graphs and paths between them. The solution can then be expressed as a feasible path on graph G as shown in figure 1 with respect to the given set of constraints. The population of ants solves the problem of topology under hand using graph representation and finding out the best optimal solution among the many. Pheromone details help build a memory for the current search process (K[i]) and for the next search processes (K [I+n]).

A. Mathematical Investigations

Under the current research only two possible ACO alterations have been observed. A multilayer logic of operation has been deployed for understanding the functionality and performance of algorithm on big data topological trees or structures. There are a total of two types of ant definitions that have been designed for this research. The first layer runs on the traditional ACO algorithm concepts. Each ant performs a number of visits and revisits. The second layer of ACO has been inspired from the field of multi agent collaborative systems. The characteristics, mathematical formulations and functionality of each ant group have been described below.

B. ACO Group one

This group is a simple illustration of ant colony optimization over topological big data structures. Thus behavior in case of topological data for ants is a type of data represented as goal to ants. Thus food for these computer ants is a specific range or type of data they are looking to mine or bring to surface. The phases have been described below

The first ant is designated with a search string suppose “look for image”. The keyword extraction for this ant is “Image” that is decoded as type to look for.

The first ant then randomly iterated through the layer of big data topology looking for the information regarded as “image”

The encounter of an image of any type suppose is then marked and a trail is left for others to follow up

The other ants then strengthen the path and markers while serving the search strings such as car image, cat image etc.

The very first advantage of using Ant colony optimization for big data topology is the changing nature of shape of topology and placement of data over a topology. The big data topologies are dynamic and change shape with every intake or influx of data. Thus the ants of the system serve the purpose of dynamic agents and adjust the tracks and paths for more optimized solution.

C. Controller Based ACO Agents

The second group of ants has been improvised for functionality. The second group has a mother or controller ant that initially runs over randomly with the knowledge of topological orientations and constructions of data. The knowledge is self-learning and adaptive in nature. The mother ant then focuses on deploying the nature of ants depending upon the task that has been encountered. For example the first scenario is the shape of data that represents a square. Each node has a distant memory collection. Every memory collection is connected to another node through a path or string of memories. The distance from one memory to another memory can be calculated in a topological shape. Thus the ants are controlled by the mother ant and are subjected to specific topologies. The child ants have a factor of independence which designates its ability to improvise the order and find a better food source than the already laid out one.

The very first scenario for second layer of ACO is to find the total number of people that bought City Honda in Asian countries. The very first information for mother agent is the topological region classification and extraction. The mother ant follows up the search string and dislocates or marks the area that needs to be penetrated for requested information.

The children are then entrusted with the task of following the boundary at any starting point “i” and calculating its runs over time and cost till a result is actually found.

The first challenge for these ant agents is that only a boundary has been marked by the mother controller thus no ant child has the markers to itself. The markers will be laid out by themselves using intra-communication. This intra-communication will enable each ant to start at a different point over the boundary and exchange information of lengths runs, executions and useful information over that path.

Each ant has a search space and its calculated cost of effectiveness. The multiple paths required to one single goal can thus be calculated over communication between two or more ants and thus markers can be decreased in potential or increased in potential overall.

The performance of each agent is given in Table 1.

V. CONCLUSION

In this paper we have analyzed the problem of extracting content information and generating classified information from data shapes. The ACO algorithm is studied and analyzed with different versions and incorporations to fully utilize and extract the meaning from a given data shape.

VI. FUTURE WORK

The research is focused for next phase as a base for a decision support system. The decision support system will actually enable the ants to be designated topological controllers. The data before storage or retrieval will be under the combination of swarm and deep learning layers. The technology will focus on delivering end to end decision results for storing data into free structured topologies.

REFERENCES

[1] Borrotti, M., Minervini, G., De Lucrezia, D. and Poli, I., 2016. Naïve Bayes ant colony optimization for designing high dimensional experiments. Applied Soft Computing, 49, pp.259-268.

[2] Asadi, S. and Shahrabi, J., 2016. ACORI: a novel ACO algorithm for rule induction. Knowledge-Based Systems, 97, pp.175-187.

[3] Krynicki, K., Houle, M.E. and Jaen, J., 2016. An efficient ant colony optimization strategy for the resolution of multi-class queries. Knowledge-Based Systems, 105, pp.96-106.

[4] Junior, L.S., Nedjah, N. and de Macedo Mourelle, L., 2013. Routing for applications in NoC using ACO-based algorithms. Applied Soft Computing, 13(5), pp.2224-2231.

[5] Fender, A., Emad, N., Eaton, J. and Petiton, S., 2016. Accelerated Hybrid Approach for Spectral Problems Arising in Graph Analytics. Procedia Computer Science, 80, pp.2338-2347.

[6] Bende, S. and Shedge, R., 2016. Dealing with Small Files Problem in Hadoop Distributed File System. Procedia Computer Science, 79, pp.1001-1012.

[7] Sheshikala, M., Rao, D.R. and Prakash, R.V., 2016. Parallel Approach for Finding Co-location Pattern–A Map Reduce Framework. Procedia Computer Science, 89, pp.341-348.

TABLE I. PERFORMANCE ANALYSIS OF AGENTS

Subject	**Efficiency** **(overall)**	**Time**	**E(Controlled)**
R1	90	Di + 4*P(R1)	94%
R2	80	Di + 4*P(R1)	84%
R3	56	Di + 4*P(R1)	44%
R4	50%	Mov(i,j) + Hit(tk)	57%
R5	56%	Di + 4*P(R1)	74%
R6	68%	Di + 4*P(R1)	64%
R7	49%	Di + 4*P(R1)	74%
R8	66%	Di + 4*P(R1)	84%
R9	30%	Mov(i,j) + Hit(tk)	57%
R10	26%	Di + 4*P(R1)	34%
R11	58%	Di + 4*P(R1)	77%
R12	78%	Di + 4*P(R1)	77%