ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016

SIGNIFICANT BIG DATA INTERPRETATION USING SAMR SCHEDULING ALGORITHM

M.SURIYA M.E
PG SCHOLAR
DEPARTMENT OF COMPUTER SCIENCE
AND ENGINEERING
KALASALINGAM INSTITUTE OF TECHNOLOGY
KRISHNAN KOIL,INDIA
suriyakhushi@gmail.com

S.JEEVITHA M.Tech
ASSISTANT PROFESSOR
DEPARTMENT OF COMPUTER SCIENCE
AND ENGINERRING
KALASALINGAM INSTITUTE OF TECHNOLOGY
KRISHNAN KOIL,INDIA
jeevitha.s@kit-edu.in

***Abstract*--Big data is a term for data sets that are so large or complex that traditional data processing applications are inadequate. Map Reduce concept is used with an incremental and distributed inference method for large-scale ontologies which realizes high-performance analysis and runtime searching, but it does not find out the slow node at execution time. This paper proposes to improve the efficiency of the map reduce scheduling algorithms by using SAMR scheduling technique which uses the factual information and finds the slow node and launches multi tasks. In addition, the usage timing of each user is calculated. Finally, implement and test the effectiveness of the proposed approach on the Hadoop framework. The purpose of this paper is to speed up the query to the user.**

Index terms- Big data, SAMR, Map Reduce, ontology reasoning.

1. INTRODUCTION

With a larger size of phonological Web [2] data and their fast gain, diverse applications have emerged in a plurality of domains such as healthcare and life sciences, business process management, expert systems, e-marketplace, Web service composition, and cloud system management. Resource description framework (RDF) [8] [11] is a general framework for reporting website metadata, or "information about the information" on the website. It produces interoperability between applications that transfer machine-understandable information on the Web. Traditional centralized logistics methods are not sufficient to process large ontologies. Distributed reasoning methods [10] are thus required to develop the scalability and performance of inference. By encouraging the inclusion of phonological content in web pages, the phonological Web [2] aims at converting the current web, dominated by unstructured and semi-structured script into a "web of data". The main scope of the phonological Web [2] is driving the evolution of the current Web by enabling users to find, share, and combine information more easily. Phonological web is based on RDF [8] [11], which integrates a variety of applications by using extensible markup language (XML) for pattern and universal resource identifier (URI) for naming. Resource description framework (RDF) is a basic representation of ontologies used to express the knowledge on the phonological web. Here, present two concepts, there are Transmission Inference Forest (TIF) and Effective Assertional Triples (EAT).

2. RELATED WORK :

Antoniou G and Bikakis A [2]

This presents an implemented defeasible reasoning system (DR-Prolog), which has been tested, evaluated and compared with existing similar implementations. Through the description of the system, process shown how user can combine the expressive power of a non-monotonic logic (defeasible logic) with the Semantic Web technologies (RDF(S), OWL, Rule-ML) to build applications for the logic and proof layers of the Semantic Web. Entirely describes reason for conflicts among rules arise naturally on the Semantic Web. To address this problem, we proposed to use defeasible reasoning that is known from the area of knowledge representation, and they had reported on the implementation of a system for defeasible reasoning on the Web. The proposed system is Prolog-based, supports Rule-ML syntax, and can reason with monotonic and non-monotonic rules, RDF facts and RDFS and OWL ontologies.

Grau B.C, Halaschek-Wiener C, Kazakov Y [5]

This paper proposed a technique for incremental ontology reasoning—that is, reasoning that reuses the

results obtained from previous computations. This is based on the notion of a module and can be applied to arbitrary queries against ontologies expressed in OWL. Here mainly focus on a particular kind of modules that exhibit a set of compelling properties and apply our method to incremental classification of OWL ontologies. It did not depends on a particular reasoned or reasoning method. Here applied to incremental classification of OWL ontologies. For ontology development, it is desirable to re-classify the ontology after a small number of changes. In this scenario, our results are very promising. Incremental classification using modules is nearly real-time for almost all ontologies and therefore the reasoned could be working transparently to the user in the background without slowing down the editing of the ontology.

Paulheim H and Bizer C [9]

An RDF knowledge base consists of an A-box, i.e., the definition of instances and the relations that hold between them, and a T-box, i.e., a schema or ontology. The SD-Type approach proposed in this paper exploits links between instances to infer their types using weighted voting. Assuming that certain relations occur only with particular types, they can heuristically assume that an instance should have certain types if it is connected to other instances through certain relations. For each property in a dataset, there is a characteristic distribution of types for both the subject and the object. They had discussed the SD-Type approach for heuristically completing types in large, cross-domain databases, based on statistical distributions. Unlike traditional reasoning, this approach was capable of dealing with noisy data as well as faulty schemas or unforeseen usage of schemas. This process can be applied to virtually any cross-domain dataset.

Lopez D, Sempere J.M, García P [7]

The inference of tree languages is related to the inference of context-free string languages using a structural sample, but the development of specific tree language learning algorithms should open new possibilities for the characterization of sub-classes of the context-free languages. The two classes of tree languages are characterized, some properties concerning these classes are proven, and they are also studied in relation to other well-known tree language classes. The first class of tree languages is obtained by extension of the notion of reversibility from string languages to tree languages. Here proved that this class contains several classes of tree languages and we propose an algorithm which learns the class in polynomial time complexity with respect to the size of the training sample set. The class of reversible tree languages could be seen as a function distinguishable language, and therefore it is possible to use the scheme of all.

3.THEORETICAL ANALYSIS

3.1 Project Scope

The scope of the project to find out the slow node at execution time. A SAMR scheduling algorithm, which improve the performance of resources through dynamic arrangement of resource allocation and reduce the usage of timing. More complicated queries can be decomposed into basic query type and through joining or merge the result the final query result.

3.2 Problem Statement

In the existing system, the Map Reduce[3] concept is used with an incremental and distributed inference method[1] for large-scale ontologies which realizes high-performance analysis and runtime searching, but it does not find out the slow node at run time. so, it takes more time to execute the process at a delay node occurs.

3.3 Proposed System

The goal of this project is to find out the delay node and calculate the usage of time. In this paper provide SAMR scheduling algorithm is used for incremental and distributed inference method for large-scale ontologies, which improve the performance of resources through dynamic arrangements of resource allocation.

3.3.1 Self Adaptive Map Reduces Algorithm

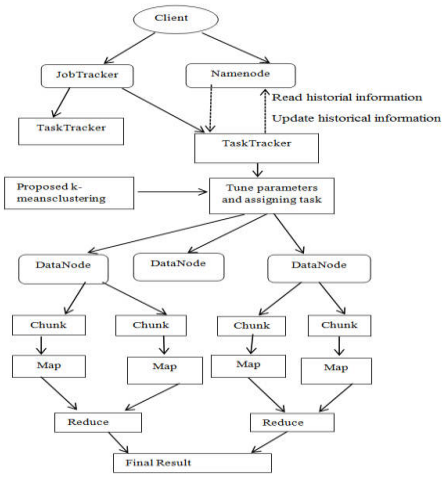
The SAMR technique uses the historical information that is being stored in each node and using that information it finds the real slow tasks. Then it maps the slow tasks and reduces the slow tasks.

Algorithm steps:

Step 1: input: Key/Value pairs
Step 2: output: Statistical results
Step 3: read historical information
Step 4: tune parameters using proposed k-means clustering
Step 5: Find slow tasks
step 6: Find slow tasktrackers
step 7: Launch back up tasks
step 8: Using the results update the historical information

3.3.1.1 mapreduce implementation

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016



3.3.2 K-Means Algorithm

In this paper, use the k-means clustering technique to tune the parameters in the historical information and finding the slow tasks very accurately. The proposed K-means algorithm can solve even the most difficult clustering issues. It requires the number of clusters that we are going to use in our process. The algorithm finds k centroids, one for each cluster. Depending on the location of the centroid the result will vary. During the map phase it finds the M1 temporary value and using this value it finds in the clusters which one is closest to the M1 value. Similarly inthe reduce phase it finds the R1 temporary value and using this value it finds in the clusters which one is closest to the R1 value. Based on the result the centroid location is changed and the values are recalculated again.

Algorithm steps:

Step 1: Input: D-set of n datanodes, n-number of datanodes, C-set of k centroids, k-number of clusters

Step 2: Output: A-set of k clusters

Step 3: Compute distance between each data nodes to all centroids

Step 4: For each Di find the closest Ci

Step 5: Add Di to A

Step 6: Remove Di from D

Step 7: Repeat for all Di……Dn andCi...Ck

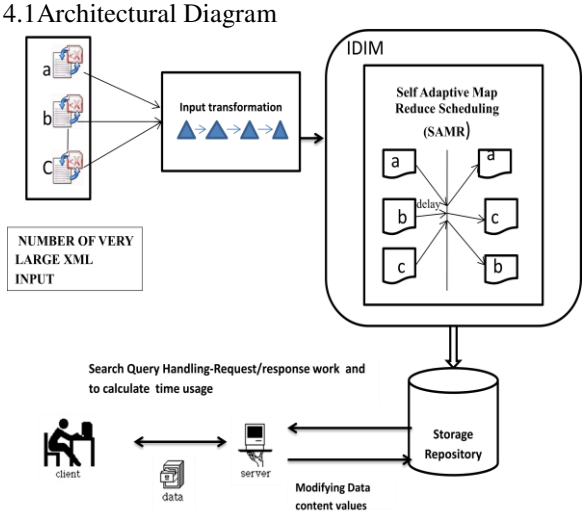
Algorithm Explanation

function Map **is**

input:integer K1 between 1 and 1100, representing a batch of 1 million social.person records
for eachsocial.person record in the K1 batch **do**
let Y be the person's age
let N be the number of contacts the person has
k-means clustering aims to partition *n* observations into *k* clusters in which each observation belongs to
produce one output record (Y,(N,1))
repeat
end function
function cache **is** searching and avoid the map phase
let K be the intermediate file having the search content
fetch(K)
output
else
map()
function Reduce **is**
input: age (in years) Y
for each input record (Y,(N,C)) **do**
Accumulate in S the sum of N*C
Accumulate in C_{new} the sum of C
repeat
let A be S/C_{new}
produce one output record (Y,(A,C_{new}))
end function

4. SIMULATION SYSTEM DESIGN

4.1Architectural Diagram



First to transfer the number of required file to IDIM. The SAMR mapreduce scheduling technique is being developed which uses the his factual information and find the slow nodes and launches backup tasks. The historical information is stored in each nodes in XML format. It adjusts time weight of each stage of map and reduce tasks according to the factual information respectively. It decreases the execution time of mapreduce job and improve the overall mapreduce performance in the heterogeneous environment.

5. Result

The proposed k-means clustering algorithm to improve the recital of the Self-adaptive MapReduce scheduling algorithm.The proposed k-means clustering algorithm find the closest distance between

each datanodes and each centroids . Using this result it update the historical information in the name node and find the accurate slow tasks , launch backup tasks and assign tasks to each task trackers. This proposed technique takes less amount of computation time.

6. CONCLUSION

In this paper, proposed a method to improve the capability of the map reduce scheduling algorithms. It works better than existing map reduce scheduling algorithms by taking less amount of computation and gives high precision. Use the proposed k-means clustering algorithm together with the Self-Adaptive MapReduce(SAMR) algorithm. However this technique works well it can assign only one task to each data node. In the future to improve its capability by allocating more number of tasks to the datanodes.

REFERENCES

[1] Keman Huang, Jianqiang Li, and MengChu Zhou, Jan- 2015," An Incremental and Distributed Inference Method for Large-Scale Ontologies Based on MapReduce Paradigm".

[2] Antoniou G and Bikakis A, 2007-"DR-Prolog: A system for defeasible reasoning with rules and ontologies on the Semantic Web," IEEE Trans. Knowl. Data Eng., vol. 19, no. 2, pp. 233–245.

[3] Billion Triples Challenge 2012 Dataset [Online]. Available: http://km.aifb.kit.edu/projects/btc-2012/

[4] Dean J and Ghemawat S, 2008-"MapReduce: Simplified data processing on large clusters," Commun. ACM, vol. 51, no. 1, pp. 107–113, 2008.

[5] Grau, B.C, Halaschek-Wiener C, and Kazakov Y, 2007- "History matters: Incremental ontology reasoning using modules," in Proc. ISWC/ASWC, Busan, Korea, pp. 183–196.

[6] Hadoop[Online]. Available: http://hadoop.apache.org/

[7] Lopez D , Sempere J.M, and García P, 2004- "Inference of reversible tree languages," IEE E Trans. Syst., Man, Cybern. B, Cybern.,vol. 34, no. 4, pp. 1658–1665.

[8] Milea V, Frasincar F, and Kaymak U, 2012 "tOWL: A temporal web ontol-ogylanguage,"IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 42, no. 1, pp. 268–281.

[9] Paulheim H and Bizer C,2013- "Type inference on noisy RDF data," in Proc. ISWC, Sydney, NSW, Australia, pp. 510–525.

[10] Schlicht A and Stuckenschmidt H, 2011- "MapResolve," in Proc. 5th Int. Conf. RR, Galway, Ireland, pp. 294–299.

[11] Urbani J, Kotoulas S, Oren E, and Harmelen F, 2009- "Scalable distributed reasoning using mapreduce," in Proc. 8th Int. Semantic Web Conf., Chantilly, VA, USA, Oct. 2009, pp. 634–649.

[12] Weaver J and Hendler J,2009 "Parallel materialization of the finite RDFS closure for hundreds of millions of triples," in Proc.

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 15, March 2016