International Journal of Computational Research and Development (IJCRD)
Impact Factor: 4.775, ISSN (Online): 2456 - 3137
(www.dvpublication.com) Volume 1, Issue 2, 2016

EVOLUTIONARY FEATURE SELECTION FOR BIG DATA PROCESSING USING MAPREDUCE AND APSO

D. Anusuya*, R. Senthilkumar & Dr. T. Senthil Prakash*****

* PG Scholar, Shree Venkateshwara Hi-Tech Engineering College, Gobi, Tamilnadu

** Associate Professor, Department of Computer Science and Engineering, Shree Venkateshwara Hi-Tech Engineering College, Gobi, Tamilnadu

*** Professor & Head, Department of Computer Science and Engineering, Shree Venkateshwara Hi-Tech Engineering College, Gobi, Tamilnadu

Cite This Article: D. Anusuya, R. Senthilkumar & Dr. T. Senthil Prakash, “Evolutionary Feature Selection for Big Data Processing Using Mapreduce and Apso”, International Journal of Computational Research and Development, Volume 1, Issue 2, Page Number 30-35, 2016.

Abstract:

Big Data -A, an acceleration framework that optimizes Big Data with plug-in components for fast data movement, overcoming the existing limitations. A novel network-levitated merge algorithm is introduced to merge data without repetition and disk access. In addition, a full pipeline is designed to overlap the shuffle, merge, and reduce phases. Our experimental results show that Big Data -A significantly speeds up data movement in Map Reduce and doubles the throughput of Big Data. In addition, Big Data -A significantly reduces disk accesses caused by intermediate data. In this paper, we propose, APSO, a distributed frequent sub graph mining method over Map Reduce. Given a graph database, and a minimum support threshold, APSO generates a complete set of frequent sub graphs. To overcome the dependency among the states of a mining process, APSO runs in an iterative fashion, where the output from the reducers of iteration i−1 is used as an input for the mappers in the iteration i. The mappers of iteration i generate candidate sub graphs of size i (number of edge), and also compute the local support of the candidate pattern. The reducers of iteration i then find the true frequent sub graphs (of size i) by aggregating their local supports. They also write the data in disk that are processed in subsequent iterations.

Index Terms: Feature Selection, Big Data, Classification & Particle Swarm Optimization.

1. Introduction:

The term "Big Data" has launched a veritable industry of processes, personnel and technology to support what appears to be an exploding new field. Giant companies like Amazon and Wal-Mart as well as bodies such as the U.S. government and NASA are using Big Data to meet their business and/or strategic objectives. Big Data can also play a role for small or medium-sized companies and organizations that recognize the possibilities (which can be incredibly diverse) to capitalize upon the gains. Interpretation of Big Data can bring about insights which might not be immediately visible or which would be impossible to find using traditional methods. This process focuses on finding hidden threads, trends, or patterns which may be invisible to the naked eye. Sounds easy, right? Well, it requires new technologies and skills to analyze the flow of material and draw conclusions. Apache Hadoop is one such technology, and it is generally the software most commonly associated with Big Data. Apache calls it "a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models." Just as Big Data can be both a noun and a verb, Hadoop involves something that is and something that does – specifically, data storage and data processing. Both of these occur in a distributed fashion to improve efficiency and results. A set of tasks known as MapReduce coordinates the processing of data in different segments of the cluster then breaks down the results to more manageable chunks which are summarized. A server cluster is a group of independent servers running Windows Server 2003, Enterprise Edition, or Windows Server 2003, Datacenter Edition, and working together as a single system to provide high availability of services for clients. When a failure occurs on one computer in a cluster, resources are redirected and the workload is redistributed to another computer in the cluster. You can use server clusters to ensure that users have constant access to important server-based resources. Server clusters are designed for applications that have long-running in-memory state or frequently updated data. Typical uses for server clusters include file servers, print servers, database servers, and messaging servers. A cluster consists of two or more computers working together to provide a higher level of availability, reliability, and scalability than can be obtained by using a single computer. Microsoft cluster technologies guard against three specific types of failure:

- ✓ **Application and Service Failures,** which affect application software and essential services.
- ✓ **System and Hardware Failures,** which affect hardware components such as CPUs, drives, memory, network adapters, and power supplies.
- ✓ **Site Failures in Multisite Organizations,** which can be caused by natural disasters, power outages, or connectivity outages.

The ability to handle failure allows server clusters to meet requirements for high availability, which is the ability to provide users with access to a service for a high percentage of time while reducing unscheduled outages. In a server cluster, each server owns and manages its local devices and has a copy of the operating system and the applications or services that the cluster is managing. Devices common to the cluster, such as disks in common disk arrays and the connection media for accessing those disks, are owned and managed by only one server at a time. For most server clusters, the application data is stored on disks in one of the common disk arrays, and this data is accessible only to the server that currently owns the corresponding application or service. Server clusters are designed so that the servers in the cluster work together to protect data, keep applications and services running after failure on one of the servers, and maintain consistency of the cluster configuration over time. Server clusters require network technologies that use IP-based protocols and depend on the following basic elements of network infrastructure:

- ✓ The Active Directory directory service (although server clusters can run on Windows NT, which does not use Active Directory).
- ✓ A name resolution service, that is, Windows Internet Name Service (WINS), the Domain Name System (DNS), or both. You can also use IP broadcast name resolution. However, because IP broadcast name resolution increases network traffic, and is ineffective in routed networks, within this Technical Reference we assume that you are using WINS or DNS.

2. Literature Review:

AIDE: An Active Learning-Based Approach for Interactive Data Exploration: Complex datasets found in many big data applications such as scientific and healthcare applications as well as for reducing the human effort of data exploration. Automatic Interactive Data Exploration framework that assists users in discovering new interesting data patterns and eliminate expensive ad-hoc exploratory queries. Data are being collected and stored at an unprecedented rate, so need to build more dynamic data-driven applications. AIDE operates on the unlabeled space of the whole data space that the user aims to explore. to achieve desirable interactive experience for the user and AIDE needs not only to provide accurate results, but also to minimize the number of samples presented to the user as well as to reduce the sampling and space exploration overhead. This approach leverages relevance feedback on database samples to model user interests and strategically collects more samples to refine the model while minimizing the user effort. It integrates machine learning and data management techniques to provide effective data exploration results as well as interactive performance.

Relevance Feature Discovery for Text Mining: It is a big challenge to guarantee the quality of discovered relevance features in text documents for describing user preferences because of large scale terms and data patterns. Most existing popular text mining and classification methods have adopted term-based approaches. An innovative model for relevance feature discovery. It discovers both positive and negative patterns in text documents as higher level features and deploys them over low-level features. Low support problem and misinterpretation problem are the two challenging issues in using pattern mining technique. So several techniques is being introduced to overcome with this issue. In this Feature Clustering is used to describe the process of feature clustering. Feature discovery and deploying, term classification, and term weighing are the three major steps. It also introduces a method to select irrelevant documents for weighting features. They continued to develop the RFD model and experimentally prove that the proposed specificity function is reasonable and the term classification can be effectively approximated by a feature clustering method. This model is best for comparing with term-based baseline model and pattern based baseline model.

Supervised, Unsupervised, and Semi Supervised Feature Selection: A Review on Gene Selection: The basic taxonomy of feature selection, and also reviews the state-of-the-art gene selection methods by grouping the literatures into three categories: supervised, unsupervised, and semi-supervised. This is so because feature selection in some places like gene expression micro array data contain hundreds of thousands of feature with small sample size. Feature selection is basically aims to select a subset of relevant features from the original set of features according to some criteria and generate the output accordingly. Supervised gene selection utilizes labelled data to select relevant features in gene expression data, Unsupervised gene selection approach selects the feature subset unassisted by labelled data and Semi-supervised gene selection approach utilizes both labelled and unlabeled data in the process of selecting a feature subset. But the usage of supervised gene selection is more acceptable by the researchers. Advancement of unsupervised and semi-supervised approaches can be considered as promising future directions in gene selection research.

Enumerating Subgraph Instances using Map-Reduce: The cited papers assume that the query (sample graph) specifies at least one node (individual) of the data graph. As a result, optimum algorithms for evaluation will surely start by searching from the fixed node or nodes. However, eventually, this search will lead to a neighbourhood that is sufficiently large that sequential search no longer makes sense. At that point, our methods can take over with what remains of the sample graph after removing nodes that have been explored on the data graph. We assume that edges are unlabeled. However, a graph with labelled edges can be represented by a collection of relations, one for each label. Search for instances of a sample graph can still be expressed as a conjunctive query and the same techniques applied.

Colorful Triangle Counting and a Mapreduce Implementation: In this note we introduced a new randomized algorithm for approximate triangle counting, which is implemented easily in parallel. We showed such an implementation in the popular Map Reduce programming framework. The key idea which improves the existing work is that by our new sampling method the degree of the multivariate polynomial expressing the number of triangles decreases by one, compared to previous work, We used the powerful result of Hajnal-Szemeredi Theorem to obtain a concentration result which is unlikely to be the best possible. We observe that our result extends any subset of triangles satisfying some predicate (e.g., containing a certain vertex), in the sense that counting such triangles in the sample leads to a concentrated estimate of the number in the original graph.

3. System Study:

Existing System: The size of the result-ant feature set is assumed fixed. Users are required to explicitly specify the maximum dimension for feature subset. Although the number of combination reduces. The major drawback is that users may not know in advance what would be the ideal size of *s*. The feature becomes minimal. By the principle of removing redundancy, the feature set may shrink to its most minimal size. The feature selection methods are custom designed for some particular classifier and optimizer. Two classical algorithms are Classification and Regression Tree algorithm (CART) for decision tree induction and Rough-set discrimination. Each time when fresh data arrive, which is typical in the data collection process that makes the big data inflate to bigger data, the traditional induction method needs to re-run and the model that was built needs to be built again with the inclusion of new data.

Drawbacks of Existing System:

- ✓ Big data mining algorithm will gradually change, and the computed results will become stale and obsolete over time.
- ✓ In many situations, it is not desirable to periodically refresh the mining computation in order to keep the mining results up-to-date.

Proposed System: In Big Data analytics, the high dimensionality and the streaming nature of incoming data aggravate great computational challenges in data mining. Big Data grows continually with fresh data are being generated at all times; hence it requires an incremental computation approach which is able to monitor large scale of data dynamically. Lightweight incremental algorithms should be considered that are capable of achieving robustness, high accuracy and minimum pre-processing latency. We investigated the possibility of using a group of incremental classification algorithm for classifying the collected data streams pertaining to Big Data. As a case study empirical data streams were represented by five datasets of different do-main that have very large amount of features, from UCI archive. A novel lightweight feature selection method by using Swarm Search and Accelerated PSO, which is supposed to be useful for data stream mining. A spectrum of experimental insights for anybody who wishes to design data stream mining applications for big data analytics using lightweight feature selection approach such as Swarm Search and APSO. APSO is designed to be used for data mining of data streams on the fly. The combinatorial explosion is addressed by used swarm search approach applied in incremental manner.

Advantages of Proposed System:

- ✓ It has develop into increasingly popular to mine such big data in order to gain insights to help business decisions or to provide better personalized, higher quality services.
- ✓ Performance improvements of i^2 MapReduce compared to both plain and iterative MapReduce performing re-computation.
- ✓ i^2 MapReduce improves the run time of re-computation on plain MapReduce by an eight fold speedup.

4. Software Design:

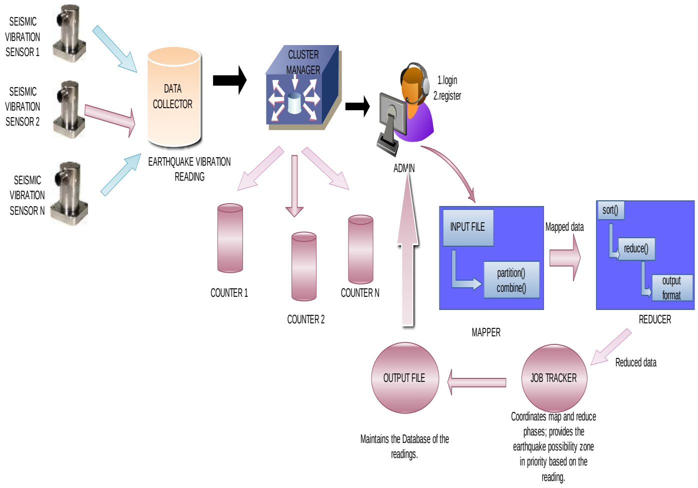
SEISMIC VIBRATION SENSOR 1
SEISMIC VIBRATION SENSOR 2
SEISMIC VIBRATION SENSOR N
DATA COLLECTOR
EARTHQUAKE VIBRATION READING
CLUSTER MANAGER
1.login
2.register
ADMIN
COUNTER 1
COUNTER 2
COUNTER N
INPUT FILE
partition()
combine()
MAPPER
Mapped data
sort()
reduce()
output format
REDUCER
Reduced data
JOB TRACKER
Coordinates map and reduce phases; provides the earthquake possibility zone in priority based on the reading.
OUTPUT FILE
Maintains the Database of the readings.

Figure 1: Architectural Design

Module Description:

User-Transparent Shuffle Service: The data-collector collects the Seismic Vibration Readings with some parameters from different Seismic Vibration Sensors. It collects parameters such as Src, Eqid, Version, Date, time, Lat, Lon, Magnitude, Depth, NST, Region. The seismic reading is then sent to the CLUSTER MANAGER. Every time when the seismic vibration sensor collects the data it is then sent to the cluster manager for further processing.

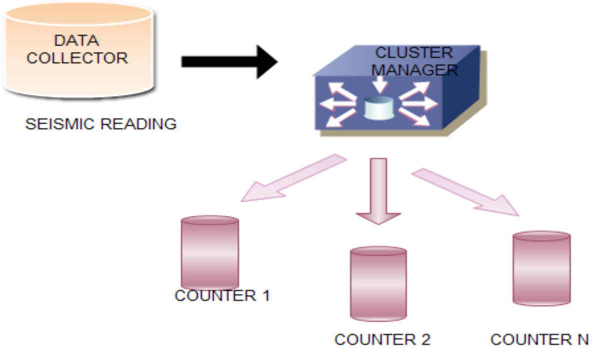

Figure 2: User Transparent Shuffle Service

Shuffle-On-Write: The shuffler implements a shuffle-on-write operation that proactively pushes different hadoop counters (nodes).This operation is performed every time when the readings is collected in the data collector. The shuffling is done based on the parameter Region. Then it is directed to the admin.

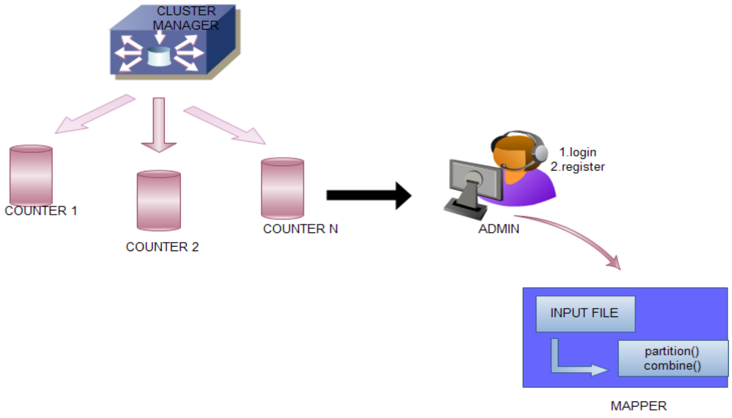

Figure 3: Shuffle-On-Write

Automated Map - Output Placement: The cluster manager distributes the data to different hadoop counters and then the admin signs in to proceed with the map-reduce process. The admin have to login once the registration is done with valid user name and password. The data can be processed only by the authenticated user. The mapping permission has to be given by the authenticated user .The mapping includes two processes: partition () and combine ().

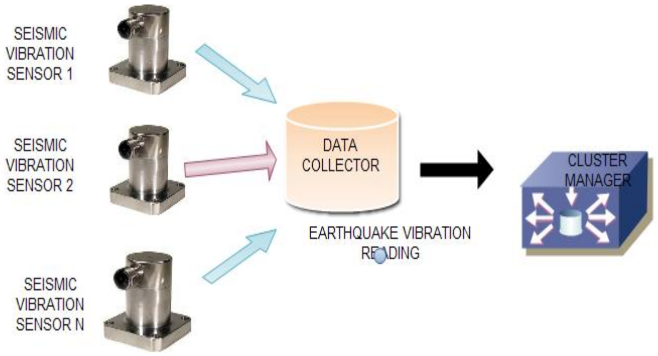

Figure 4: Automated Map-Output Placement

Flexible Scheduling of Apso Tasks:

The mapped data is then subjected to the reduce process. The reducer includes two functions: 1.sort () and 2.reduce().The Job-Tracker coordinates the map and reduce phases; provides the earthquake possibility zone in the priority based on the readings. The Output-File maintains the database to records the readings. The readings are transferred to the admin; the admin intimates about the earthquake zone in advance to the station.

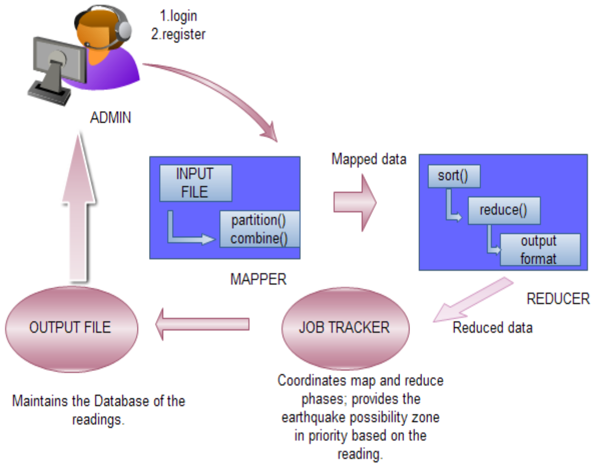
1.login
2.register
ADMIN
INPUT FILE
partition()
combine()
MAPPER
Mapped data
sort()
reduce()
output format
REDUCER
Reduced data
JOB TRACKER
OUTPUT FILE
Maintains the Database of the readings.
Coordinates map and reduce phases; provides the earthquake possibility zone in priority based on the reading.

Figure 5: Flexible scheduling of APSO tasks

5. Conclusion:

In Big Data analytics, the high dimensionality and the streaming nature of the incoming data aggravate great computational challenges in data mining. Big Data grows continually with fresh data are being generated at all times; hence it requires an incremental computation approach which is able to monitor large scale of data dynamically. Lightweight incremental algorithms should be considered that are capable of achieving robustness, high accuracy and minimum pre-processing latency. In this paper, we investigated the possibility of using a group of incremental classification algorithm for classifying the collected data streams pertaining to Big Data. As a case study empirical data streams were represented by five datasets of different do-main that have very large amount of features, from UCI archive. We compared the traditional classification model induction and their counter-part in incremental inductions. In particular we proposed a novel lightweight feature selection method by using Swarm Search and Accelerated PSO, which is supposed to be useful for data stream mining. The evaluation results showed that the incremental method obtained a higher gain in accuracy per second incurred in the pre-processing. The contribution of this paper is a spectrum of experimental insights for anybody who wishes to design data stream mining applications for big data analytics using lightweight feature selection approach such as Swarm Search and APSO. In particular, APSO is designed to be used for data mining of data streams on the fly. The combinatorial explosion is addressed by used swarm search approach applied in incremental manner. This approach also fits better with real-world applications where their data arrive in streams. In addition, an incremental data mining approach is likely to meet the demand of big data problem in service computing.

6. Future Enhancement:

In future decouples the problem of co placing related files from the applications that exploit this property. The extension requires minimal changes to HDFS. A new file property to identify related data files and modify the data placement policy of HDFS to co-place all copies of those related files is introduced. These changes retain the benefits of Hadoop, including load balancing and fault tolerance The decoupling also enables a wide variety of applications to exploit data co-placement by simply specifying related files. A flexible, dynamic and light weight approach to co-place related data files, which is implemented directly in HDFS is future. Identification of two use cases in log processing , i.e. Join and sessionization, where co- partitioning files and co-placing them speeds up the query processing significantly. Fault tolerance, data distribution and data loss properties of the future are studied using a stochastic model. The objective of HDFS default data placement policy is to achieve load balancing by distributing the data evenly across the data nodes, independently of the intended use of data. This simple data placement policy works well with most Hadoop applications that access just a single file, but applications that process data from different files can get a significant boost in performance with customized strategies.

7. References:

1. Anusuya.D, Senthil Kumar.R, Senthil Prakash.T, Manimozhi.N,(July - December 2016) ‘Novel Feature Selection for BigData Processing using MapReduce and APSO’ Volume 8, (P) 108-110.
2. Aggarwal, Charu C., (2007) ‘Data streams: models and algorithms’ Vol. 31.Springer.
3. Arinto Murdopo, (July 2013) ‘Distributed Decision Tree Learning for Mining Big Data Streams’, Master of Science Thesis, European Master in Distributed Computing.
4. S. Fong, X.S. Yang, S. Deb, (Dec. 2013) ‘Swarm Search for Feature Selection in Classi-fication’, The 2nd International Conference on Big Data Science and En-gineering (BDSE 2013), 2013, 3-5.
5. Mohamed Medhat Gaber, Arkady Zaslavsky, Shonali Krishnaswamy, (June 2005) ‘Mining data streams: a review’, ACM SIGMOD Record, Volume 34 Issue 2, pp.18-26.
6. Quinlan, J. R., (1993) ‘C4.5: Programs for Machine Learning’ Morgan Kauf-mann Publishers.

7. Rokach, Lior, and OdedMaimon, (2005) 'Top-down induction of decision trees classifiers-a survey Systems, Man, and Cybernetics', Part C: Ap-plications and Reviews, IEEE Transactions on 35, no. 4: 476-487.
8. Wei Fan, Albert Bifet, (April 2013) 'Mining Big Data: Current Status, and Forecast to the Future', SIGKDD Explorations, Volume 14, Issue 2, pp.1-5.