International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 03 Issue: 06 | June-2016 www.irjet.net p-ISSN: 2395-0072

Multi-Criteria Clustering For Big-Data using MR-Triage

[1]Vimal Gupta, Shweta Singh, Ranjan Kumar Sharma, [2]Aravinda C V

8th Sem, B.E. (ISE),SJBIT,VTU,BELGAUM, KARNATAKA

Professor, Dept. of I S E ,SJB Institute of Technology , Karnataka, India

---

Abstract - In present world that mining large amounts of security data can help generate actionable intelligence and make the understanding of Internet attacks better. To deal with internet attacks what we really need is cyber situational and attack attribution. Cyber situational awareness is attracting much attention. Practical clustering algorithms require multiple data scans so we need a scalable multi criteria clustering. Big Data is a new term which identify the datasets which cannot be managed due to their large size with typical software tools. Data mining is a process to find the data from the large databases for analysis . This security data mining process involves a considerable amount of features interacting in a non-obvious way, which makes it inherently complex. To deal with this challenge, we introduce MR-TRIAGE. The MR-TRIAGE workflow is made of a scalable data integration. The results will demonstrate that we can efficiently handle large datasets with our algorithm.

Key Words: Big Data, Data mining, MapReduce, Clustering.

1. INTRODUCTION

Big data is a term for data sets that are so large or complex that traditional data processing applications are inadequate. The term often refers simply to the use of predictive analytics or certain other advanced methods to extract value from data, and seldom to a particular size of data set. Big data can be described by the following characteristics– variety, velocity, variability, volume, veracity. Data mining is the computational process of discovering patterns in large data sets. We analyze a set of new MapReduce [1] based algorithms, MR-TRIAGE, that are designed to process massive datasets in Hadoop. Both algorithms and implementation details are outlined and discussed in this deliverable. Analyzing the data and finding out important part out of it is really difficult and is the most important need. Data mining can help us meet this need by providing goals to find out the important part.

There are three different ways in which the data set can be large: (1) there can be a large number of elements in the data set, (2) each element can have many features, and,(3) there can be many clusters to discover. Talking about clusters ,cluster is a group of similar things or people positioned or occurring closely together. clustering is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense or another) to each other than to those in other groups (clusters).The clustering technique which we are using is MR-TRIAGE. Map reduce is a programming model. Users specify the computation in terms of a map and a reduce function. The computation takes a set of input key/value pairs, and produces a set of output key/value pairs. This is a new technique for clustering these large, high dimensional datasets. The issues of how to parallelize the computation, distribute the data, and handle failures conspire to obscure the original simple computation with large amounts of complex code to deal with these issues. Clustering is the subject of active research in several fields such as statistics, pattern recognition, and machine learning. For improving the understanding of internet attack we need to deal with two critical aspects cyber situational and attack attribution.. Data modeling puts clustering in a historical perspective rooted in mathematics, statistics, and numerical analysis. Multi-criteria decision (MCDA) [2][3] is not scalable. In this paper, we propose a scalable algorithm which is based on map reduce map reduce algorithm consist of three phases: (1) prototype extraction (PE) ,(2) single feature clustering and (3) multi-criteria clustering. Our implementation focuses on retrieving points that are good representatives of a large number of points which are very similar (or very close) to the prototype points for each feature. In graph clustering stage, for each feature MR-TRIAGE builds the relationships among the prototypes using appropriate similarity metrics and turns the relationship into a graph by modeling prototypes as nodes and the similarity values as the edges with weight between nodes. Finally at the multi-criteria data clustering stage, MR-TRIAGE combines the prototypes and graph clusters found in previous steps using Map Reduce. The functional requirements of this paper is The application has to be able to segregate the big data into portions of data that would be clustered as data which has to be secured , non secured or low level secured. The multi-criteria algorithm has to be efficiently used in order to have a soft clustering applied on the dataset. The output of the clustering algorithm would provide the proportionality of how much the date must be secured. Talking about non-functional requirements which are flexible , robust, customized and has a low maintenance cost. The framework has already demonstrated its effectiveness in context of various security investigation. But existing system has some issues which were firstly MCDA

clustering is not scalable and needs to be transitioned to parallel processing to cope with the growing size of the data sets and secondly Map Reduce poses a whole new set of challenges and implementation choices [4].

1.1 Related Work

Considerable efforts have been devoted to applying data mining techniques to problems related to computer security[5].Many efforts was exclusively focused on the Improvement of intrusion detection system. Big Data is changing the traditional technology domains, introducing new security models and new security design approaches to address emerging security challenges. Clustering [6] is considered as one of the most effective approaches to explore large quantities of data generated by the security monitoring infrastructures. However, Clustering large data sets can be very time and memory consuming because most algorithms calculate the distance between each pair of points such as nearest neighbor Search [7],[8], data summarization [9],[10] incremental clustering [11], [12], density-based [13], [14] and hierarchical methods [15], [16], have been developed to efficiently handle large-size datasets.

Map-Reduce is a programming model and an associated implementation for processing and generating large datasets that is amenable to a broad variety of real-world tasks. It gradually gains attention in cyber security research. Users specify the computation in terms of a map and a reduce function, and the underlying runtime system automatically parallelizes the computation across large- scale clusters of machines, handles machine failures, and schedules inter-machine communication to make efficient use of the network and disks. Botnet detection, Zhao et al.[17] designed and implemented a MapReduce system called BotGraph to detect a new type of botnet spamming [18] attacks targeting major Web email providers.

Additionally, various real-world systems can be converted into complex networks, such as social network, internet, telecommunication networks, etc. It can be also interesting to use graph clustering methods to model certain security activities, e.g. spams campaign [19].Gibson et al. present an algorithm based on a recursive application of fingerprinting via shingles to identify nodes that share subsets of neighbor's. Rytsareva et al. [20] propose a MapReduce-based implementation of the algorithm. Bahami et al. [21] provide an approximated algorithm for finding densest sub-graph in logarithmic time using MapReduce. Lin et al. [22] present three design patterns that address these issues and can be used to accelerate a large class of graph algorithms.

1.2 System Architecture

Map-Reduce is a programming model and an associated implementation for processing and generating large datasets that is amenable to a broad variety of real-world tasks. It gradually gains attention in cyber security research. MCDM is concerned with structuring and solving decision and planning problems involving multiple criteria. The purpose is to support decision-makers facing such problems. The difficulty of the problem originates from the presence of more than one criterion. There is no longer a unique optimal solution to an MCDM problem that can be obtained without incorporating preference information. The concept of an is

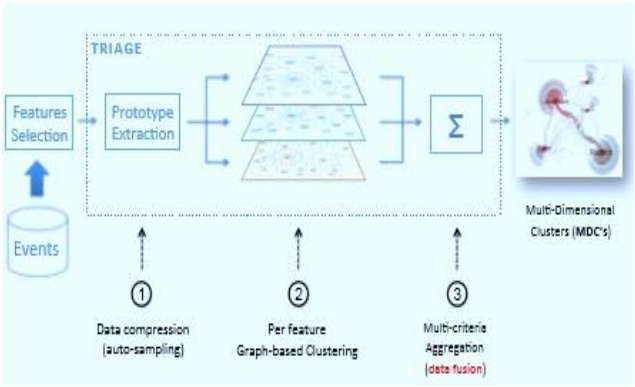

Fig 1: Overview of TRIAGE

often replaced by the set of non dominated solutions. A non dominated solution has the property that it is not possible to move away from it to any other solution without sacrificing in at least one criterion. Therefore, it makes sense for the decision-maker to choose a solution from the non dominated set.

MapReduce allows for distributed processing of the map and reduction operations. Provided that each mapping operation is independent of the others. MapReduce can be applied to significantly larger datasets than "commodity" servers can handle – a large server farm can use Map Reduce to sort a petabyte of data in only a few hours. The parallelism also offers some possibility of recovering from partial failure of servers or storage during the operation: if one mapper or reducer fails, the work can be rescheduled – assuming the input data is still available. Map reduce function works in phases: 1) mapper function 2)reducer function .

2. TRIAGE FRAMEWORK

In this section, we analyze MR-Triage framework that makes triage scalable using MapReduce. We first cover the technical challenge when implementing the framework and provide theoretical computational analysis of the MR-Triage algorithms. we describe the software modules

including description of the software components, the steps to run the software etc.

STEP 1: PROTOTYPE EXTRACTION: The goal of prototype extraction is to get rid of the combinatorial explosion in data clustering when the number of objects in the data set becomes too large and thus a pairwise approach is not possible any more, e.g., similarity matrix Ak(i; j) . It is ideal to identify prototypes within the data set, i.e., data points that are good representatives of a large number of points which are very similar (or very close) to the prototype point (acting as a sort of control).

1) we divide the data set into smaller blocks, each block is of a given window size k.
2) for each smaller k-block of data in a reducer, we extract prototypes by searching for very dense regions of points and we represent them by a single data point. We also label all the other points as "neighbor's".
3) we repeat the algorithm but now on the extracted prototypes. This has the effect to reduce the total number of prototypes by merging prototypes that are (almost) identical, but have been extracted in different data blocks.

STEP 2: GRAPH CLUSTERING: In this step, we first build the relationships among the prototypes using appropriate similarity metrics. This requires us to build a pairwise matrix to identify the relationship. Formally for a given feature Fk and Pk ={p1,p2,...,pm},we find Rf ={(pl, pq)|ωk(pl,pq) > k} where k is a predefined threshold and pl,pq ∈PFk. One of the most straightforward way to find clusters in a graph is identifying the connected components, which are sub graphs in which any two vertices are connected to each other by a path.Z. we begin the difficult work of defining what constitutes a cluster in a graph and what a clustering should be like; we also discuss some special classes of graphs. In some of the clustering literature, a cluster in a graph is called *community*. This uses a similarity metrics to build relationships . A metric that measures distance between strings . Useful for fuzzy string searching . Build a pairwise matrix to identify relationships ,Model prototypes as nodes and similarities as edges with weight.

STEP 3:MULTI-CRITERIA DATA CLUSTERING: By repeating Step 1 & 2 for different features, we obtain a set of clusters for every feature, which provide interesting viewpoints on the underlying phenomena that have generated the events in the first place.
The following metrics are used to analyze the theoretical costs of the proposed algorithms in the Map Reduce framework.

- Number of Tasks. This metric aims to determine how many tasks are required to accommodate the requirements of the algorithms.
- Communication Costs. This metric measures the potential network I/O costs of the algorithms.
- Computational Complexity. This metric measures the computational complexity of the algorithm at the reducer level.

3. PROPOSED SYSTEM

This paper main goal is to handle Scalability to large datasets (Big Data) ability to work with high dimensional data and ability to find clusters of irregular shape handling outliers. Cluster the data for internet attacks [23] , by means of multi-criteria evaluation process and Improving Interpretability (meaning) of results. We propose a set of distributed algorithm MR-TRIAGE built on Map-Reduce that can perform scalable multi-criteria data ,on very large security data sets.

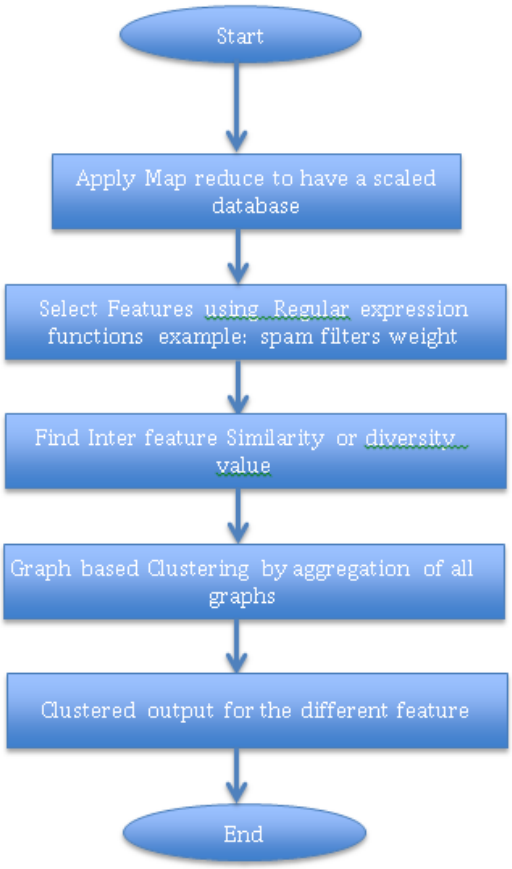

Fig 2 : Data Flow Diagram

3. CONCLUSIONS

In this paper we introduce a new framework called MR-TRIAGE leveraging multi-criteria data clustering (MCDC) to perform scalable data clustering on large security data sets and further implement a set of efficient algorithms in a 3-stage In this paper we introduce a new framework called MRTRIAGE leveraging multi-criteria data clustering (MCDC) to Map-Reduce paradigm. We optimize MR-TRIAGE performance by extracting a smaller set of representative prototypes. Map-Reduce programs have been implemented

internally at Google over the past four years, and an average of one hundred thousand Map-Reduce jobs are executed on Google's clusters every day, processing a total of more than twenty petabytes of data per day (Source). As future works, we would like to evaluate various statistical sampling methods to both efficiently and effectively identify prototypes and evaluate the effectiveness of different graph clustering algorithm.

REFERENCES

[1] S. Jajodia, P. Liu, V. Swarup, and C. Wang, Eds., Cyber Situational Awareness: Issues and Research, ser. Advances in Information Security. Springer, Nov 2009, vol. 46.

[2] U. Franke and J. Brynielsson, "Cyber situational awareness a systematic review of the literature," Computers & Security, 2014.

[3] G. Beliakov, A. Pradera, and T. Calvo, Aggregation Functions: A Guide for Practitioners, ser. Studies in Fuzziness and Soft Computing. Springer, 2007, vol. 221.

[4] V. Torra and Y. Narukawa, Modeling decisions - information fusion and aggregation operators. Springer, 2007.

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

[6] J. Lin, "Mapreduce is good enough? if all you have is a hammer, throw away everything that's not a nail!" CoRR, vol. abs/1209.2191, 2012.

[7] D. Barbara and S. J. (Eds), Eds., Applications of Data Mining in Computer Security, ser. Advances in Information Security. Springer, 2002, vol. 6.

[8] A. K. Jain, "Data clustering: 50 years beyond k-means," Pattern Recogn. Lett., vol. 31, no. 8, pp. 651–666, Jun. 2010.

[9] D. Pelleg and A. Moore, "Accelerating exact k-means algorithms with geometric reasoning," in Proceedings of ACM KDD, 1999, pp. 277–281.

[10] J. Buhler, "Efficient large-scale sequence comparison by locality sensitive hashing," Bioinformatics, vol. 17, no. 5, pp. 419–428, 2001

[11] McCallum, K. Nigam, and L. H. Ungar, "Efficient clustering of high-dimensional data sets with application to reference matching," in Proceedings of ACM KDD. ACM, 2000, pp. 169–178.

[12] R. Cilibrasi and P. M. B. Vitnyi, "Clustering by compression," IEEE Transactions on Information Theory, vol. 51, pp. 1523–1545, 2005.

[13] P. Bradley, U. Fayyad, and C. Reina, "Scaling clustering algorithms to large databases." AAAI Press, 1998, pp. 9–15.

[14] D. H. Fisher, "Knowledge acquisition via incremental conceptual clustering," Mach. Learn., vol. 2, no. 2, pp. 139–172, Sep. 1987.

[15] M. Ester, H.-P. Kriegel, J. S, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in Second International Conference on Knowledge Discovery and Data Mining. AAAI Press, 1996, pp. 226–231.

[16] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, "Optics: ordering points to identify the clustering structure," SIGMOD Rec., vol. 28, no. 2, pp. 49–60, Jun. 1999.

[17] J. C. Gower and G. J. S. Ross, "Minimum spanning trees and single linkage cluster analysis," Journal of the Royal Statistical Society. Series C (Applied Statistics), vol. 18, no. 1, 1969.

[18] F. Murtagh, "A survey of recent advances in hierarchical clustering algorithms," The Computer Journal, vol. 26, no. 4, pp. 354–359, Nov. 1983.

[19] Y. Zhao, Y. Xie, F. Yu, Q. Ke, Y. Yu, Y. Chen, and E. Gillum, "Botgraph: Large scale spamming botnet detection," in Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation. Berkeley, CA, USA: USENIX Association, 2009, pp. 321–334.

[20] O. Thonnard and M. Dacier, "A strategic analysis of spam botnets operations," in Proc. of ACM CEAS, 2011.

[21] I. Rytsareva and A. Kalyanaraman, "An efficient MapReduce algorighm for parallelizing large-scale graph clustering," 2012.

[22] B. Bahmani, R. Kumar, and S. Vassilvitskii, "Densest subgraph in streaming and mapreduce," Proc. VLDB Endow., vol. 5, no. 5, pp. 454– 465, Jan. 2012.. Lin and C. Dyer, Data-Intensive Text Processing with MapReduce, 2010.

[23] Symantec, "Internet Security Threat Report," http://www.symantec.com/business/threatreport/.