International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 7, July - 2014

A Review Paper on Parallel Implementation of Sentinel Mining Algorithm on GPU

N. M. Sonawane
Computer Engineering Department
Late G. N. Sapkal College of Engineering,
Anjneri, Nashik

Prof. B. R. Nandwalkar
Computer Engineering Department
Late G. N. Sapkal College of Engineering,
Anjneri, Nashik

Abstract— **This paper proposes parallel algorithms which are implemented using parallel programming language to exploit the power of today's advance hardware. The aim of proposed system is to improve the performance of existing system. It provide a scalable solution to sentinel mining approach using parallel programming. The proposed system is to provide a modular approach to separate data independent modules to execute them on parallel processors and store the result on shared memory. Modern GPU's highly parallel structure make them more effective than general-purpose CPUs for algorithms where processing of large blocks of data is done in parallel. GPU computing is the use of a GPU (graphics processing unit) together with a CPU to accelerate general-purpose scientific and engineering applications. To process data in parallel CPU contains few number of cores where as GPU contains 100s of cores. We can compare the speed up in the system by measuring time for execution on same data set for both sequential and parallel implementation. It would be interesting to develop a version of SentBit that is optimized to use a GPU processor.**

Keywords: **Pattern mining, Predictive mining, cube based data mining, sentinels.**

I. INTRODUCTION

To discover pattern in large data set is called as Data mining. The Data mining is useful where there are no predetermined notions about what will constitute an interesting outcome. Data Mining is the search for novel useful and nontrivial information in large amount of data. The goal of the data mining process is to extract information from a data set and transform it into an understandable structure for further use.This leads us to move Sentinel Mining approach. Sentinel Mining Approach can be used for discovering the relationship between measures of multidimensional data cube that is represented by Sentinels. As the name states, the sentinel mining to find out the warning period is the intent of itself. So with help of sentinel we can represent the relationships between source measure and target measure. When there are changes in one or many source measure then there are also changes in respective target measure. When sentinel are identified then it decreases the observation time. The indication stream is used to solve problem of sentinel mining techniques. With help of SentBit algorithm CPU execute instruction sequentially. Behind that there is use of parallel execution of instruction then we get less processing time of instruction. The Objective of this approach is to provide a generalized sentinel mining approach and to improve performance of processor by reducing processing time of processor, no matter whether the data is real data or synthetic data. In the proposed system we have to implement SentBit algorithm sequentially then this sequential algorithm should be converted into parallel SentBit algorithm. So our final output will show difference between time required by CPU and time required by GPU for this parallel SentBit algorithm. In proposed system there are total eight modules. First module shows reading of data from files and database. Then second and third shows encoding data into binary form and then create bitmap for each source measure. In fourth and fifth step it shows test sentinel which are created and calculate score for each sentinels. After that sort the sentinels if they are below threshold then discard it but if they are high then it can be used.Then lastly predicting warning period which will depend on output of sentinels. The rest of the paper is organized as follows. We present and discuss some related works in section II. Section III describes the System proposed in mentioned work. In section IV the algorithmic strategy is explained, and result analysis is given in section V. Finally, we conclude the paper in section VI.

II. RELATED WORK

Reference work contains the invention in modifying traditional data mining methods to fast growing data.,Various methods like pruning techniques, fast search techniques etc.

Morten Middelfart, Torben Bach Pedersen have proposed that there is use of sequential sentinel mining algorithm. But with help of Graphics processing unit this sentinel mining algorithm should be converted into parallel form. With help of this the processing time of processor should be reduced by using multicore architecture. So first of all find the sentinel and then find out warning period. After that finding processing time in central processing unit and graphics processing unit and then show the processing time difference between them.Using CPU, processing time required is more[1].

Middelfart and T.B. Pedersen have proposed that the sentinel concept, a scoring principle and its implementation in the

TARGIT BI Suite. This paper demonstrates so-called sentinels in the TARGIT BI Suite. Sentinels are a novel type of rules that can warn a user if one or more measure changes in a multi-dimensional data cube are expected to cause a change to another measure critical to the user. We present the concept of sentinels, and we explain how sentinels represent stronger and more specific rules than sequential patterns and correlation techniques. In addition, we present the algorithm, implementation, and data warehouse setup that are prerequisites for our demo. In the demo we present a dialogue where users, without any prior technical knowledge, are able to select a critical measure, a number of cubes, and a time dimension, and subsequently mine and schedule sentinels for early warnings[2].

J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M.C. Han Hsu have provides that we have developed a novel, scalable, and efficient sequential mining method, called PrefixSpan. Its general idea is to examine only the prefix subsequencesand project only their corresponding postfix subsequences into projected databases. Sequential pattern mining is an important data mining problem with broad applications. It is challenging since one may need to examine a combinatorially explosive number of possible subsequence patterns. Most of the previously developed sequential pattern mining methodsfollow the methodology of substantially reduce the number of combinations to be examined. However, still encounters problems when a sequence database is large and/or when sequential patterns to be mined are numerous and/or long. In this paper, we propose a novel sequential pattern mining method, called PrefixSpan (i.e., Prefix-projected Sequential pattern mining), which explores prefixprojection in sequential pattern mining. PrefixSpan mines the complete set of patterns but greatly reduces the efforts of candidate subsequence generation. Moreover, prefix-projection substantially reduces the size of projected databases and leads to efficient processing. Our performance study shows that PrefixSpan outperforms both the based GSP algorithm and another recently proposed method, FreeSpan, in mining large sequence[3].

T. Imielinski, L. 'Khachiyan, and A. Abdulghani have proposed cubegrades, showed how to generate them using efficient pruning techniques with the Grid Base Pruning GBP as the main foundation and finally defined a query language to query and retrieve (previously stored) cubegrades and cubes. Cubegrades are generalization of association rules which represent how a set of measures (aggregates) is affected by modifying a cube through specialization (rolldown) generalization (rollup) and mutation (which is a change in one of the cube's dimensions). Cubegrades are significantly more expressive than association rules in capturing trends and patterns in data because they use arbitrary aggregate measures, not just COUNT, as association rules do. Cubegrades are atoms which can support sophisticated "what if" analysis tasks dealing with behavior of arbitrary aggregates over different database segments. As such, cubegrades can be useful in marketing, sales analysis, and other typical data mining applications in business. We formally define cubegrades, show methods to generate them by using efficient pruning algorithms, and finally define two query languages to generate and retrieve sets of cubegrades which satisfy user defined conditions. We also demonstrate how to evaluate simple cubegrade queries and conclude with a number of open questions and possible extensions of the work[4].

P. Shenoy, J.R. Haritsa, S. Sudarshan, G. Bhalotia, M. Bawa, and D. Shenoy have proposed that we have addressed problem of designing a general purpose vertical mining algorithm whose applicability or efficiency must be increases. In a vertical representation of a market basket database each item is associated with column of values each representing transaction in which it is present. The association rule mining algorithm is proposed representation show performance improvement over their classical horizontal counter parts but either efficient for certain database sizes or assume particular characteristic or database contents which are applicable to particular specific kinds of database schemas. So it presents VIPER which is vertical mining algorithm. VIPER store data in compressed bit vector and integrates them[5].

J. Yang, W. Wang, P.S. Yu, and J. Han have proposed that to discover long sequential pattern in noisy environment. In this environment observed symbol in a sequence may differ from true value. The compatibility matrix is introduced to provide probabilistic connection from the observation underlying true value. In noisy envirment sequence may not actually reflect actual behaviour[6].

R. Agrawal, T. Imielinski, and A. Swami have proposed that we introduced the problem of mining association rules between sets of items in a large database of customer transactions. Each transaction consists of items purchased by a customer in a visit. We are given a large number of database of large transaction. Each transaction contain items purchased by customer by each customer visit. We generate a efficient algorithm that generate all significant association rule between items in the database. The algorithm show incorporate buffer management and buffer estimation and pruning techniques. We also represent result of algorithm to sale a data obtained from large retailing company which shows effectiveness of algorithm[7].

R. Srikant and R. Agrawal has proposed that with the increased dissemination of bar code scanning technologies it was possible to accumulate vast amounts of Market-basket dataset containing millions of transactions. Thus the need arose to study the patterns of consumer consumption that could help in improving the marketing infrastructure and related disciplines like targeted marketing[8].

III. IMPLEMENTATION DETAILS

A.System Architecture

Following diagram shows proposed system architecture.

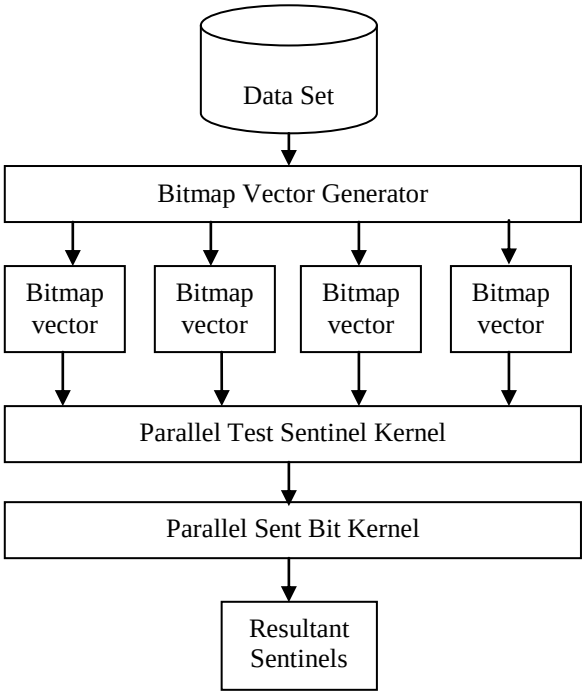

Fig.3.1 System architecture

In first step there is reading of data from disk. The data which is used by proposed system is store in database, so it must be received from data base. In the dataset consists a collection of data which is used for doing different operations on it. In second step there is generation of bitmap vector. So encode data in binary form. There is convert feature to one dimensional feature vector. This bitmap vector generator form number of bitmap vector. Then after that there is creation of bitmap vector for source measure. This bitmap vector are in 0 and 1 form. So with help of this there is creation of bitmap vector for each source measure. In fourth step test bitmap quality measure which is nothing but finding score of source measures. The testing should be done by parallel. Fifth step consists of calculation of score confidence, support, balance for bitmap. Then we have to find score of all sentinels which are source sentinel and target sentinels. There is finding score for each measure parallely. Then there is threshold value in which we have to compare score value with these threshold value. After that if score is greater than threshold then this value should be accepted otherwise it must be rejected. After comparing the value the next step is to add bitmap to sentinels. There is formation of bitmaps and add this to sentinels. The parallisation is done with help of sentbit parallel algorithm. Lastly get the resultant sentinels and to store the result in the CPU. After comparing score and threshold then there is addition of bitmap to the sentinels. So there is comparison of all process which take part in execution of process.

B.Algorithm

If the current database is too large then retrieving data from disk is too slow than retrieving data from RAM. For getting scalable data the efficient data mining techniques are applied.

The data mining process is to extract information from a data set and transform it into an understandable structure for further use. Successful data mining requires integrating several technologies.

So in existing system there are seven algorithm. So these algorithms are in sequential form. In first algorithm encode all measure into bitmap.

$$confidence = \frac{|A+B|}{\#changes\ to\ source}$$

$$balance = \frac{4 * |A||B|}{(|A|+|B|)2}$$

The intent of sentinel mining is to identify actionable causal relationships in data that can be used for warnings. Specifically, a sentinel is a causal relationship where changes in one or more source measures, are followed by changes to a target measure, within a given time period, referred to as the warning period. Once identified, a sentinel can be used to provide warnings of consequent threats when the premise is "triggered."

There are two basic mathematical formulae to find the trigger values from given data as above.

Let C be a multidimensional cube containing a set of facts,C = $\{(d1, d2, \ldots . dn, m1, m2, \ldots . mp)\}$ The dimension values,$d1, d2, \ldots . dn$ belong to the dimensions D1,D2,D3,…. Dn and we refer to the "dimension part" of a fact, $(d1, d2, \ldots . dn)$ as a cell. We say that a cell belongs to C, denoted by $(d1, d2, \ldots . dn) \in C$ exist. We say that a measure value, mi, is the result of a partial function, Mi : D1*D2*…*→R denoted by, Mi(d1,d2,….dn) = mi we define Ind as Ind(Mi,t,o,d2,d3,……,dn) We refer to ▲ as a positive indication and to ▼ as a negative indication. We define a wildcard, that can be either ▲ or▼.

So in this paper following algorithms are used :

1] Encode all measures into bitmaps :
EncodeData(a set of source measures, {S1, . . . , Sp} in C,
a target measure, Target ∈ C, an offset, o, and max.
warning period, Maxw)
1: for all S in {S1, . . . , Sp} do
2: CreateBitmap (S, o, 0)
3: for w = 1 to Maxw do
4: CreateBitmap (Targetw, o, w)

Above algorithm shows that encoding all measures into bitmaps. So there are number of sources like S1,S2,..Sp. Above algorithm shows parameter like offset, maximum warning period. Then create bitmap for all sources measures. If warning period is one to maximum then there is creation of bitmap. So for above algorithm the parallel sentinel mining algorithm is

Steps :

1. Generate bitmap for all source measure
2. Generate bitmap for target measure for all warning period

Input for algorithm first is database D(with source measure and target measure) offset,warning period.

Output for first algorithm is bitmap for all measure.

The CPU (central processing unit) perform following step for algorithm:

1. Load D from disk
2. Extract source measure and target measure via scanning D and store it on CPU memory.
3. Read offset (o) and warning period (w).
4. Transform 2D source measure into 1D vector
5. Transfer 1D vector to GPU.

The GPU (graghics processing unit) perform following step for algorithm:

1. Create bitmap for each source measure in parallel on GPU
a. On each work item(i)do
b. Generate ith bitmap on GPU
2. Transfer result to CPU

2] Generate bitmap for given source measure :

For this algorithm requires following :

Input : a source measure, offset,warning period

Output : bitmap for given measure.

So this algorithm will work on GPU as :

1.Create bitmap for given source measure in parallel on GPU
a.On each work item (i)do
b.Using indirect function for given dimension set a bit if Ind give positive result.
c.Reset a bit if Ind give negative result.
2.Transfer the result to CPU.

3] SentBit Algorithm :

SentBit : Sentinel mining using Bitmap

For this algorithm require max number of sentinels to be return,n,data set ,C,a set of source measure{S1….Sp} belongs to C, target measure Target belongs to C,offset o and threshold ,Maxw and MaxSource

Follwing step shows senbit algorithm :

1. Encode data
2. Allocate memory for SentList
3. Allocate memory for MaxElimSupp
4. For w=1 to Maxw do
5. For x=1 to p do
6. TestSentinel(x,Bitmap(w),Target,w,o)
7. Return the top n sentinel from SentList for value of w where sentinel with highest NewScore exists.

   So output score will be new score/ MaxElimSupp(w)

IV. CONCLUSION

The problem of sentinel mining algorithm is solved by using very efficient algorithm which is Sentbit algorithm. The Sentbit algorithm is faster as compared to state of art. But Sentbit algorithm cannot support the approximation techniques. So this algorithm gives the exact results. But Sentbit supports optimization techniques which use specific CPU instruction and multicore architecture. The modern processor provides specific CPU instruction and multicore architecture. If there is schema level relationships in data then only there is sentinels are found. So with help of sentinels we have to find out relationship that cannot found using sequential pattern mining. So by using Parallel Sentinel Mining we have to mine different target measure. Then to mine these different target measure we improve our performance. The multidimensional environment is become exploited with help of sentinel mining . Then we get aggregation level of dimension as well as select location and shape of data area. The fixed warning period and offset can be replaced by including intervals. So the GPU processor can use optimized Senbit algorithm. By using parallel Sentbit algorithm we have to reduce processing time of processor and with help of this we have to improve performance of processor.

It would be interesting to develop a version of SentBit that is optimized to use a GPU processor. So we perform optimization on SentBit algorithm with help of graphics processing unit then performance of processor will be increased.

.

REFERENCES

[1] Morten Middlefart,"*Efficient sentinel mining using bitmap on modern processor*,"IEEE transactions on knowledge and data engineering,vol 25,no 10,October 2013.
[2] M. Middelfart,"*Using Sentinel Technology in the TARGIT BI Suite,"* Proc. VLDB Endowment,vol. 3,no. 2, pp. 1629-1632, September 2010.
[3] J. Pei, "*PrefixSpan: Mining Sequential Patterns by Prefix-Projected Growth*," Proc. 17th Int'l Conf. Data Eng. (ICDE), pp. 215-224, 2001.
[4] T. Imielinski,"*Cubegrades: Association Rules,"*Data Mining Knowledge Discovery,vol.6,no.3,pp.219-257,September 2002.
[5] P. Shenoy, J.R. Haritsa, S. Sudarshan, G. Bhalotia, M. Bawa, and D Turbo-Charging ,"*Vertical Mining of Large Databases,"* Proc. ACM SIGMOD Int'l Conf. Management of Data, pp. 22-33, 2000.
[6] J. Yang,"*Mining Long Sequential Patterns in a Noisy Environment*,"Proc. ACM SIGMOD Int'l Conf. Management of Data, pp. 406-417, June 2002.
[7] R. Agrawal, T. Imielinski, and A. Swami, "Mining Association Rules between Sets of Items in Large Databases," Proc. ACM SIGMOD Int'l Conf. Management of Data, pp. 207-216, 1993.
[8] R. Srikant and R. Agrawal, "Mining Sequential Patterns: General-izations and Performance Improvements," Proc. Fifth Int'l Conf. Extending Database Technology: Advances in Database Technology (EDBT), pp. 3-17, 1996. R. Srikant and R. Agrawal, "Mining Sequential Patterns: General-izations and Performance Improvements," Proc. Fifth Int'l Conf. Extending Database Technology: Advances in Database Technology (EDBT), pp. 3-17, 1996.
[9] Y. Zhu and D. Shasha, "*StatStream: Statistical Monitoring of Thousands of Data Streams in Real Time*," Proc. 28th Int'l Conf. Very Large Data Bases (VLDB), pp. 358-369, June 2002.
[10] Intel, Intel SSE4 Programming Reference, July 2007.
[11] "Advanced Micro Devices," Software Optimization Guide for AMD Family 10h Processors, November 2008.
[12] J. Han and M. Kamber, "*Data Mining Concepts and Techniques,"* second ed. Morgan Kaufmann Publishers, 2006.

[13] J. Han, J. Pei, B. Mortazavi-Asl, Q. Chen, U. Dayal, and M. Hsu, "FreeSpan: Frequent Pattern-Projected Sequential Pattern Mining," Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), pp. 355-359, 2000.
[14] J. Pei, J. Han, B. Mortazavi-Asl, J. Wang, H. Pinto, Q. Chen, U. Dayal, and M. Hsu, *"Mining Sequential Patterns by Pattern-Growth: The PrefixSpan Approach,"* IEEE Trans. Knowledge Data Eng., vol. 16, no. 11, pp. 1424-1440, November. 2004.
[15] F. Nakagaito, T. Ozaki, and T. Ohkawa, *"Discovery of Quantita-tive Sequential Patterns from Event Sequences*," Proc. IEEE Int'l Conf. Data Mining Workshops (ICDM), pp. 31-36, 2009.
[16] J. Han and M. Kamber,"*Data Mining Concepts and Techniques*, "second ed. Morgan Kaufmann Publishers, 2006.
[17] R. Agrawal, K.I. Lin, H.S. Sawhney, and K. Shim, "Fast Similarity Search in the Presence of Noise, Scaling, and Translation in Time-Series Databases," Proc. 21st Int'l Conf. Very Large Databases (VLDB), pp. 490-501, 1995.
[18] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules in Large Databases," Proc. 20th Int'l Conf. Very Large Databases (VLDB), pp. 487-499, 1994.
[19] R. Agrawal and R. Srikant, "Mining Sequential Patterns," Proc. Int'l Conf. Data Eng. (ICDE), pp. 3-14, 1995.
[20] P. Bosc, O. Pivert, and L. Ughetto, "On Data Summaries Based on Gradual Rules," Proc. Sixth Int'l Conf. Computational Intelligence, Theory and Applications: Fuzzy Days, pp. 512-521, 1999.
[21] S. Brin, R. Motwani, J.D. Ullman, and S. Tsur, "Dynamic Itemset Counting and Implication Rules for Market Basket Data," Proc. ACM SIGMOD Int'l Conf. Management of Data, pp. 255-264, 1997.
[22] P. Shenoy, J.R. Haritsa, S. Sudarshan, G. Bhalotia, M. Bawa, and D. "Turbo-Charging Vertical Mining of Large Databases," Proc. ACM SIGMOD Int'l Conf. Management of Data, pp. 22-33, 2000