Abstract—Data mining used to find hidden knowledge from large amount of Databases. Periodic Pattern Mining is useful in Weather Forecasting, Fraud Detection and GIS Applications. In General, spatio-temporal pattern discovery process finds the partially ordered subsets of the event-types whose instances are located together and occur serially for a given collection of Boolean spatio-temporal event-types. Big Data concerns large-volume, complex, growing data sets with multiple, autonomous sources. With the fast development of networking, data storage, and the data collection capacity, Big Data is now rapidly expanding in all science and engineering domains, including physical, biological and bio-medical sciences. In this paper, a new framework is proposed to find spatiotemporal patterns from Big Data. Existing algorithms are well in computation of necessary patterns, but more problematic when they are applied to Big Data. Big Data is a new trend used to analyze the datasets that due to their large size and complexity, Developers cannot manage them with traditional current algorithms or data mining software tools. Big Data mining is the capability of extracting useful information from these large datasets or streams of data, that due to its volume, variety, and velocity, it was not possible before to do it. The Big Data challenge is becoming one of the most exciting opportunities for the next years. This Paper focuses on a broad overview of pattern mining algorithms and significance in Spatiotemporal Databases, its current status, trade-offs, and forecast to the big data pattern mining future.

Keywords — Periodicity Detection, Spatial Patterns, Big Data, Cascading Spatiotemporal Pattern Discovery, MapR

I. INTRODUCTION

Data mining is a useful tool for extracting nontrivial, implicit, previously unknown and potentially useful information or pattern from large databases [28]. It is the process of applying the methods like neural networks, clustering, genetic algorithms (1950s), decision trees (1960s) and support vector machines (1980s) to data with the intention of uncovering hidden patterns. A Time-Series Database is a collection of data values gathered generally at uniform interval of time to reflect certain behavior of an entity. In Real World, There are several examples of Time-Series such as Weather Conditions of a particular location, Spending Patterns, Stock Growth, and Transactions in a Supermarket, Network Delays, Power Consumption, Computer Network Fault Analysis and Security Breach detection, Earthquake Prediction.

The periodicity detection is a process of finding temporal regularities within the Time-Series and the goal of analyzing a Time-Series Database is to find how frequent a periodic pattern (full or partial) is repeated within time intervals. In general, there are three types of periodic patterns can be detected in a time series Database such as Symbol Periodicity, Sequence Periodicity or Partial Periodic Patterns and Segment or Full-Cycle Periodicity [22]. We consider a set of Boolean Spatio-Temporal (ST) event types such as crime types and their instances. The Boolean event types are important because primary concern is the occurrence or absence of an event type at a particular location and time. Given such a set, cascading spatio-temporal patterns are partially ordered subsets of event types whose instances are spatial neighbours and occur in a series of stages. It is an example of a Cascading Spatiotemporal Pattern Discovery CSTP [24]. The first event1 in the CSTP is the hurricane event. Subsequent events are represented by events such as heavy rainfall, localized flooding and wind damage creates a CSTP from an urban crime dataset. The first event here is bar-closings (represented by circles), and subsequent events are assaults (represented by triangles) and drunk driving (represented by squares). It shows individual instances of events of bar-closing, assault and drunk driving with their location and time. It shows the locations of all event instances. The problem is 80-90% of data produced today is unstructured and Gartner Predicts 800% data growth over next 5 years [26]. Big data is a term applied to data sets whose size is beyond the ability of commonly used software tools to capture, manage, and process the data within a tolerable elapsed time.

II. BIG DATA PATTERN MINING

This section gives motivation and directions for pattern mining towards Big Data research. The basic three parameters of Big Data are volume, variety, and velocity of information, drives unprecedented complexity and opportunity [25]. The main idea of this paper is to introduce periodic pattern mining in big data to be useful in rare event analysis converted as negative patterns including positive patterns [34]. Data and content increase over coming decade. Most of the data generated from web is unstructured. In 2025, 35 Zettabytes of data to be generated.
Today the storage levels of data in the forms of Gigabytes, Terabytes, Petabytes, Exabytes, Zettabytes, and Yottabytes only.

2.1 Pattern Mining Classification:

(a) Kinds of patterns and rules
   (i) Basic Patterns – Frequent patterns, Association rules and Closed/maximal Patterns etc.
   (ii) Multilevel and Multidimensional Patterns – Uniform and Item-set based High dimensional patterns etc.
   (iii) Extended Patterns – Approximate, Uncertain, Compressed, Rare/negative and Colossal Patterns etc.

(b) Mining Methods:
   (i) Basic Mining methods – Candidate Generation, Pattern Growth, Vertical format etc.
   (ii) Mining interesting patterns – Interestingness, Constraint-based, Correlation Rules and Exception Rules etc.
   (iii) Distributed, parallel & Incremental – Stream Patterns, Distributed/Parallel and Incremental Mining etc.

(c) Extensions and Applications:
   (i) Extended Data types – Sequential & Time-Series, Structural, Spatial, Temporal and Multimedia patterns.
   (ii) Applications – patterns based Classification & Clustering, Semantic Annotation and Privacy Preserving.

III. RELATED WORK

In 1994, Rakesh Agrawal and Ramakrishnan Srikant identified the problem of discovering association rules between items in a large database of sales transactions and proposed two new algorithms for solving this problem that are fundamentally different from the known algorithms such as AprioriHybrid [1]. AprioriHybrid is a combination of two algorithms such as Apriori and AprioriTid also has excellent scale-up properties with respect to the transaction size and the number of items in the database.

Again in 1995, same authors proposed two new algorithms such as AprioriSome and AprioriAll to mine sequential patterns from a large database of customer transactions, where each transaction consists of customer-id, transaction time, and the items bought in the transaction [2].

In 1998, Roberto J. Bayardo Jr. Proposed a new pattern mining algorithm called the Max-Miner, that scales roughly linearly in the number of maximal patterns embedded in a database irrespective of the length of the longest pattern [3]. Max-Miner is also easily made to incorporate additional constraints on the set of frequent itemsets identified. Later, Garofalakis et al. in 1999, proposed SPIRIT, a method of mining user-specified sequential patterns by using regular expression constraints [35]. In 2000, J. Han, J. Pei and R. Mao proposed an efficient CLOSET Algorithm, for mining closed itemsets with the help of three techniques such as FP-Tree structure, Single prefix path compression and partition based projection mechanism. It is efficient and scalable over large databases, and faster than the traditional algorithms [4].

In 2000, Yves Bastide, Rafik Taouil, Nicolas Pasquier, Gerd Stumme proposed the algorithm PASCAL which introduces a novel optimization of the well-known algorithm Apriori and it is based on pattern counting inference that relies on the concept of key patterns [5]. PASCAL is better than three algorithms such as Apriori, Close and Max-Miner and it is the most efficient algorithm for mining frequent patterns. Later, Jiawei Han, Jian Pei, and Yiwen Yin proposed a novel frequent pattern tree (FP-tree) structure, which is an extended prefix tree structure for storing compressed, crucial information about frequent patterns, and develop an efficient FP-tree based mining method, FP-growth, for mining the complete set of frequent patterns by pattern fragment growth [6].

Later in 2001, Doug Burdick, Manuel Calimlim, Johannes Gehrke proposed a new algorithm called MAFIA, for mining maximal frequent itemsets from a transactional database [7]. This algorithm outperforms previous work by a factor of three to five. Again in 2001, Ilias Tsoukatos and Dimitrios Gunopulos solved the problem of mining spatiotemporal patterns and finding sequences of events that occur frequently in spatiotemporal datasets with the help of DFS_MINE, a new algorithm for fast mining of frequent spatiotemporal patterns in environmental data, which has the advantage that the amount of space required is minimal [8].
In 2002, Mohammed J. Zaki and Ching-Jui Hsiao, proposed a new algorithm called CHARMM, an efficient algorithm for mining all frequent closed itemsets [9]. It also uses a technique called diffsets to reduce the memory footprint of intermediate computations and scalable in the number of transactions. In 2003, William Cheung and Osmar R. Zaïane, proposed a novel data structure called CATS Tree. CATS Tree extends the idea of FPtree to improve storage compression and allow frequent pattern mining without generation of candidate itemsets and allow mining with a single pass over the database as well as efficient insertion or deletion of transactions at any time. Later, Zaki in 2001, proposed SPADE, an algorithm proposed to find frequent sequences using efficient lattice search techniques and simple joins [30]. Another approach called MEMory Indexing for Sequential Pattern mining (MEMISP) is introduced by Lin and Lee in 2002 [31]. In 2003, Jianyong Wang, Jiawei Han and Jian Pei, proposed a winning algorithm CLOSET+. It integrates the advantages of the previously proposed effective strategies. It is better than existing mining algorithms including CLOSET, CHARMM and OP, in terms of runtime, memory usage and scalability [10]. Later Guimei Liu Hongjun Lu Wenwu Lou and Jeffrey Xu Yu, proposed a disk-based data structure, CFP-tree (Condensed Frequent Pattern Tree), for organizing frequent patterns discovered from transactional databases and also developed algorithms to efficiently support two important types of queries, namely queries with minimum support constraints and queries with item constraints, against the stored patterns, as these two types of queries are basic building blocks for complex frequent pattern related mining tasks[11].

In 2005, Jianyong Wang, Jiawei Han, Ying Lu, and Petre Tsvetkov, proposed a mining task: mining top-k frequent closed itemsets of length no less than min_l, where k is the desired number of frequent closed itemsets to be mined, and min_l is the minimal length of each itemset and developed an efficient algorithm, called TFP, is developed for mining such itemsets without generation of candidate itemsets and solved with a two stage algorithm, Periodica [12].

Later, Juyoung Kang and Hwan-Seung Yang, solved the problem of mining spatio-temporal patterns from trajectory data. This method first finds meaningful spatio-temporal regions and extracts frequent spatio-temporal patterns based on a prefix-projection approach from the sequences of these regions [15]. Later, Nizar R. Mabroukeh and C. I. Ezeife, proposed a taxonomy of Sequential Pattern Mining Algorithms, which is represented in Table1 [16]. Later, Zhenhui Li, Ming Ji, Jae-Gil Lee, Lu-An Tang, Yintao Yu, Jiawei Han and Roland Kays, proposed a system called MoveMine, is designed for sophisticated moving object data mining by integrating several attractive functions including moving object pattern mining and trajectory mining [18], [19]. In 2012, P.Mohan and S.Shekar proposed the cascading spatiotemporal pattern (CSTP) discovery process finds partially ordered subsets of these event-types whose instances are located together and occur serially for a given collection of Boolean spatiotemporal (ST) event-types [24]. Discovering CSTPs from ST data sets is important for application domains such as public safety (e.g., identifying crime attractors and generators) and natural disaster planning, (e.g., preparing for hurricanes). The proposed model is mentioned section 4.

<table>
<thead>
<tr>
<th>Table1</th>
<th>Tabular Taxonomy of Sequential-Pattern Mining Algorithms</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithm</td>
<td>Candidate-growth</td>
</tr>
<tr>
<td>CLOSET+</td>
<td>X</td>
</tr>
<tr>
<td>SPADE</td>
<td>X</td>
</tr>
<tr>
<td>CatTree</td>
<td>[</td>
</tr>
<tr>
<td>PetTree</td>
<td>X</td>
</tr>
<tr>
<td>Umlat</td>
<td>X</td>
</tr>
<tr>
<td>ODP</td>
<td>X</td>
</tr>
<tr>
<td>STG</td>
<td>X</td>
</tr>
<tr>
<td>AxialSTP</td>
<td>X</td>
</tr>
<tr>
<td>RAAP</td>
<td>X</td>
</tr>
<tr>
<td>FOR</td>
<td>X</td>
</tr>
<tr>
<td>LRFP</td>
<td>X</td>
</tr>
</tbody>
</table>

IV. PROPOSED MODEL

The following are the big data big challenges in the future era:

i) Data models for managing big data
ii) Big Data Testing – ex. generation of optimal test data
iii) Real-time streaming data analytics
iv) Scalable analytics on large data sets
v) Systems architecture for big data management
vi) Main memory data management techniques
Energy-efficient data processing  
Energy-aware Resource Management  
Benchmarking big data systems  
Security and Privacy of Big Data  
Failover and reliability for big data systems

Nowadays it is impossible for current databases, technologies and architectures to store and manage huge data. The above model proposes a solution by analyzing existing pattern mining algorithms which are mentioned in Table 1. In 2009, Ho Jin Woo and Won Suk Lee, proposed estMax algorithm to find closed or maximal frequent item sets over online transactional data streams is not easy due to the requirements of a data stream [20]. This method maintains the set of frequent item sets by a prefix tree and extracts all MFIs without any additional superset/subset checking mechanism. The Performance of the estMax method is comparatively analyzed by a series of experiments to identify its various characteristics. It has faster processing time and better performance than estDec algorithm. The main problem identified in stream data is the processing activity which is mentioned in the below figure 2.

V. EXPERIMENTAL RESULTS

We performed all experiments on a 2.2GHz Intel Core 2 Duo processor PC machine with 3GB main memory Windows XP OS and all programs are implemented in Java. The results from figure 3(a) to 3(f) show that user input screen, prefix tree generation, frequent item sets and generated CFIs & MFIs respectively.

Fig. 1. Architecture for Big Data Pattern Mining

Fig. 2. Data Stream Processing Engine

Fig. 3(a) Table with list of transactions

Fig. 3(b) List of Transactions in a table
The future work from this method is to improve the performance of the proposed method estMax, in terms of memory usage. To achieve this, a more compact frame of a prefix tree should be devised. In 2007, Huiping Cao, Nikos Mamoulis and David W. Cheung, solved the problem of mining periodic patterns in spatiotemporal data and proposed an effective and efficient algorithm called STPMine1, for retrieving maximal periodic patterns [32]. We implemented the same algorithm with few modifications in the parameter settings. The language used was Java and the experiments were performed on a Pentium IV 700MHz workstation with 2 GB of memory, running Windows XP Operating System. The future work from this algorithm include the automatic discovery of the period ‘T’ related to frequent periodic patterns and the discovery of patterns with distorted period lengths. The results represented by figure 4(a) to 4(f), show that Data preparation and generated periodic patterns from a large mobile object movement’s data.
In 2010, Jinlin Chen, proposed a novel data structure, UpDown Directed Acyclic Graph (UDDAG), is invented for efficient sequential pattern mining. UDDAG allows bidirectional pattern growth along both ends of detected patterns. It often outperforms PrefixSpan by almost one order of magnitude in scalability tests. UDDAG is also considerably faster than Spade and LapinSpam. We performed all the experiments on a Windows XP Operating System with 2.2 GHz Intel Core 2 Duo Processor PC Machine and 3 GB memory. We implemented in Java and compared with different algorithms such as PrefixSpan, Spade, and LapinSpam, which were all implemented in C++ by previous authors. The results represented by figure 5(a) to 5(c), show that generated sequential patterns by using proposed algorithm called UDDAG with few modifications.
One major feature of UDDAG is that it supports efficient pruning of invalid candidates. In the future, there is a requirement of storage of UDDAG in an efficient way. The future topics include mining with constraints, closed and maximal pattern mining, approximate pattern mining, and domain-specific pattern mining, etc. whenever large searching spaces are involved and pruning of searching spaces is necessary. In 2011, Faraz Rasheed, Mohammed Alshalalfa, and Reda Alhajj, proposed an algorithm called STNR, which can detect symbol, sequence (partial), and segment (full cycle) periodicity in time series [22]. The algorithm uses suffix tree as the underlying data structure. The algorithm is noise resilient; it has been successfully demonstrated to work with replacement, insertion, deletion, or a mixture of these types of noise.

We performed all the experiments on a Windows 7 Operating System with 2.3 GHz Intel Core 2 Duo Processor PC Machine and 3 GB memory. The results represented by figure 6 (a) to 6(d), show that finding the whole time series and in a subsection of it to effectively handle different types of noise. We also implemented pruning strategy to remove redundant patterns for a given time-series Database. The conducted study demonstrates the applicability and effectiveness of the proposed algorithm; it is generally more time-efficient and noise-resilient than existing algorithms. The future work includes online construction of suffix tree. This type of approach is used in sensor networks, ubiquitous computing, and DNA sequence mining.
In 2010, Osman Abul, Francesco Bonchi, and Fosca Giannotti, proposed the problem of hiding sequential patterns and show it’s NP-hardness [23].

We implemented sanitization algorithm by using java programming language on a Pentium 2.77GHz processor with 40 GB Hard disk, 3 GB RAM Windows Operating System with the back end, MySQL Database. This algorithm finds spatiotemporal trajectories which require hiding of trajectories that are unexploitable from available data. There are different ways to hide sensitive trajectory data considered as a future work from this approach.

Recently in 2012, Mr. Pradeep Mohan et al. proposed QSTP discovery process which finds the partially ordered subsets of the Boolean Spatio-Temporal event-types whose instances are located together & occur serially [24]. The discovered patterns are represented using Directed Acyclic Graph (DAG). Originally, this algorithm was implemented in MATLAB 7.9 and we implemented this approach with modifications by using PHP with the back end MySQL and performed experiments on windows Operating System with 3 GB Memory. The results from figure 7(a) to 7(c) show that how to get cascading spatiotemporal patterns called CSTPM. The future work includes how to perform expensive significance tests by randomly permuting ST instances and accounting for multiple testing, which is used in weather forecasting.

Fig. 6(b) periodicity within a subsection

Fig. 6(c) Pruning strategy on Full Series patterns

Fig. 6(d) Pruning strategy on subsection series

Fig. 7(a) QSTPs used in Crime Detection events

Fig. 7(b) Generated CSTPs
VI. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a new framework useful in Big Data pattern mining including the observations of many pattern mining algorithms with different scenarios. The future work contains how to mine frequent patterns and closed frequent patterns, sequential patterns, Periodic patterns with different parameters in Big Data field. The traditional algorithms and techniques may not be suitable for big data due to large, heterogeneous and stream data needs to be processed with in elapsed time [40]. Finally, there is a scope to develop new algorithms and methods works on MapReduce programming model to perform pattern mining, an active area of research in Big Data domain.

Acknowledgements

The authors appreciate the efforts of UG Students, Sai Vamsi.V, M.Sandeep Reddy, M.Girish Kumar Reddy, P.Keerthi and Sai Avinash and Research Scholar, K.Suresh including the comments from the Professor Dr.A.Rama Mohan Reddy, Department of CSE, S.V.University College of Engineering; their feedback and suggestions improved the quality of the research presented here. The views expressed here are those of the authors and the authors thank the handling associate editors and the anonymous reviewers for their constructive comments.

REFERENCES

[6] J. Han, J. Pei, Y. Yin and R. Mao, "Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach", Data Mining and Knowledge Discovery, an International Journal, Volume 8, Issue 1, pages 53-87, January 2004, Kluwer Academic Publishers.
[10] J. Wang, J. Han, and J. Pei, "CLOSET+: Searching for the Best Strategies for Mining Frequent Closed Itemsets", Proc. 2003 ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD’03), Washington, D.C., August, 2003.
[18] Zhenhui Li, Ming Ji, Yue Yu, Jiawei Han and Roland Kays, MoveMine: Mining Moving Object Databases. SIGMOD’10, June 6–11, 2010, Indianapolis, Indiana, USA. 2010 ACM 978-1-4503-0032-2/10/06 ...$10.00.
[20] Ho Jin Woo and Won Suk Lee. estMax: Tracing Maximal Frequent Item Sets Instantly over Online Transactional Data Streams. IEEE Transactions on Knowledge and Data Engineering, VOL. 21, NO. 10, OCTOBER 2009.
Osman Abul, Francesco Bonchi, and Fosca Giannotti, Hiding Sequential and Spatiotemporal Patterns. IEEE Transactions on Knowledge and Data Engineering, VOL. 22, NO. 12, DECEMBER 2010.


Xixian Han, Jianzhong Li, Donghua Yang, and Jinbao Wang. Efficient Skyline Computation on Big Data. IEEE Transactions on Knowledge and Data Engineering, VOL. 25, NO. 11, November 2013.

Manish Bhide and Krithi Ramamritham. Category-Based Infidelity Bounded Queries over Unstructured Data Streams. IEEE Transactions on Knowledge and Data Engineering, VOL. 25, NO. 11, November 2013.


Wei Fan, Albert Bifet. Mining Big Data: Current Status, and Forecast to the Future. SIGKDD Explorations Volume 14, Issue 2, Poo.s 1-5.


