ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

Max-Miner Algorithm Using Knowledge Discovery Process in Data Mining

[1]M.Rajalakshmi, [2]M.Sakthi

M.Phil Scholar, Dept. of Computer Science, NGM College, Pollchi, Tamilnadu, India[1]

Assistant Professor, Department of Computer Science, NGM College, Pollachi, Tamilnadu, India[2]

ABSTRACT: Discovering frequent item sets is an important key problem in data mining applications, such as the discovery of association rules, strong rules, episodes, and minimal keys. Typical algorithms for solving this problem operate in a bottom-up, breadth-first search direction. The computation starts from frequent itemsets (the minimum length frequent itemsets) and continues until all maximal (length) frequent itemsets were found. During the execution, every frequent item set is explicitly considered. A new algorithm is presented which combines both the bottom-up and the top-down searches. The primary search direction is still bottom-up, but a restricted search is also conducted in the top-down direction. This search is used only for maintaining and updating a new data structure, the maximum frequent candidate set. It is used to prune early candidates that would normally encountered in the bottom-up search. A very important characteristic of the algorithm is that it does not require explicit examination of every frequent item set. Therefore the algorithm performs well even when some maximal frequent item sets are long. As its output, the algorithm produces the maximum frequent set, i.e., the set containing all maximal frequent item sets, thus specifying immediately all frequent item sets. Pattern-mining algorithm (Max-Miner) presented scales roughly linearly in the number of maximal patterns embedded in a database irrespective of the length of the longest pattern. In comparison, previous algorithms based on Apriority scale exponentially with longest pattern length. Experiments on real data show that when the patterns are long, our algorithm is more efficient by an order of magnitude or more.

KEYWORDS: Max-Miner Algorithm, Data Cleaning, Data Access, System ideal state.

I INTRODUTION

A pharmaceutical company can analyze its recent sales force activity and their results to improve targeting of high-value physicians and determine which marketing activities will have the greatest impact in the next few months. The data needs to include competitor market activity as well as information about the local health care systems. The results can be distributed to the sales force via a wide-area network that enables the representatives to review the recommendations from the perspective of the key attributes in the decision process. The ongoing, dynamic analysis of the data warehouse allows best practices from throughout the organization to be applied in specific sales situations. A credit card company can leverage its vast warehouse of customer transaction data to identify customers most likely to be interested in a new credit product. Using a small test mailing, the attributes of customers with an affinity for the product can be identified. Recent projects have indicated more than a 20-fold decrease in costs for targeted mailing campaigns over conventional approaches. A diversified transportation company with a large direct sales force can apply data mining to identify the best prospects for its services. Using data mining to analyze its own customer experience, this company can build a unique segmentation identifying the attributes of high-value prospects. Applying this segmentation to a general business database such as those provided by Dun & Bradstreet can yield a prioritized list of prospects by region. A large consumer package goods company can apply data mining to improve its sales process to retailers. Data from consumer panels, shipments, and competitor activity can be applied to understand the reasons for brand and store switching. Through this analysis, the manufacturer can

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

select promotional strategies that best reach their target customer segments. Each of these examples has a clear common ground

II RELATED WORK

Martin Ester Xiang Zhang et al The emergence of automated high-throughput sequencing technologies has resulted in a huge increase of the amount of DNA and protein sequences available in public databases. KRALJ NOVAK, LAVRAC AND WEBB et al., This system gives a survey of contrast set mining (CSM), emerging pattern mining (EPM), and subgroup discovery (SD) in a unifying framework named supervised descriptive rule discovery. Geetha M R.J. DSouza et al, a new algorithm for mining frequent closed item sets from large volumes of data is implemented. A frequent item set is maximal if none of its proper supersets is frequent. Lot Lakhal, Gerd Stumme et al., Association rules are a popular knowledge discovery technique for warehouse basket analysis. They indicate which items of the warehouse are frequently bought together. Yu HIRATE, Eigo IWAHASHI, and Hayato YAMANAet al., Conventional frequent pattern mining algorithms require some user-specified minimum support, and then mine frequent patterns with support values that are higher than the minimum support.Marek Wojciechowski, Maciej Zakrzewicz et al.,Discovery of frequently occurring subsets of items, called item sets, is the core of many data mining methods. Most of the previous studies adopt Apriori-like algorithms, which iteratively generate candidate item sets and check their occurM. R. Aghaebrahimi, S. H. Zahiri, and M. Amiri et al., A data miner based on the learning automata is proposed and is called LA-miner. The LA-miner extracts classification rules from data sets automaticallyrence frequencies in the database. Assaf Schuster, Ran Wolff, and Dan Trock et al., They present a new distributed association rule mining (D-ARM) algorithm that demonstrates super linear speedup with the number of computing nodes.

III PROPOSED ALGORITHM

1.Knowledge Discovery in Data Mining:

Knowledge discovery in databases (KDD) has received increasing attention and has been recognized as a promising new field of database research. It is defined as "the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data". The key step in the knowledge discovery process is the data mining step, "consisting of applying data analysis and discovery algorithms that, under acceptable computational efficiency limitations, produce a particular enumeration of patterns over the data". Historically, the notion of finding useful patterns in data has received a variety of names, including data mining, knowledge extraction, information discovery, information gathering, data archeology, data processing and pattern. The rapid emergence of electronic data management methods has led to some recent times as the "Information Age call. "Powerful database for the collection and management systems in use in virtually all large and mid-range businesses – there is hardly a transaction that is not a computer record somewhere. Every year more automated transactions collect any information about the activities, activities and achievements. All these data have valuable information, eg, Trends and patterns, which can be used to improve business decisions and optimize success. However, today's databases contain so much data that it is almost impossible to analyze them manually valuable information for decision making. In many cases, hundreds of independent attributes should be considered simultaneously to accurately model system behavior. Data warehousing helps set the stage for KDD in two important ways:

(1) Data Cleaning

(2) Data Access.

Data cleaning As organizations are forced to consider a unified logical given the wide variety of data and databases that they own, they have to address the problem of mapping data to a single naming convention, uniformly and handling missing data, and handling noise and if possible errors. Uniform access to well-defined methods and data should be made for access to data and access paths to data that was previously difficult to achieve (eg stored offline). Once organizations and individuals have the problem of how to store and access to their data, the natural next step remaining is

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

the question, what else do we do with all information? This is where opportunities for KDD natural origin. A popular approach for the analysis of data warehouses called Online Analytical Processing (OLAP).

3.2 Techniques of KDD mining:

1. In the ***Selection***-step the significant data gets selected or created. Henceforward the KDD process is maintained on the gathered target data. Only relevant information is selected, and also meta data or data that represents background knowledge. Sometimes the combination of data from ubiquitous sources can be useful, but possible matters of compatibility have to observed.

2. In the ***Data Mining*** phase, the data mining task is approached. Fayyad gives a classified overview over existing data mining techniques. He makes suggestions, which technique may be used for which objectives, but most of the techniques are now improved. The output of this step is detected patterns. Data Mining will be focused on following articles.

3. The ***interpretation*** of the detected pattern reveals whether or not the pattern is interesting. That is, whether they contain knowledge at all. This is why this step is also called evaluation. The duty is to represent the result in an appropriate way so it can be examined thoroughly. If the located pattern is not interesting, the cause for it has to be found out. It will probably be necessary to fall back on a previous step for another attempt.

3.3 Problem Identification:

To address this problem, this paper proposes the Max-Miner algorithm for efficiently extracting only the maximal frequent itemsets, where an itemset is maximal frequent if it has no superset that is frequent. Because any frequent item set is a subset of a maximal frequent item set, Max-Miner's output implicitly and result in two or more orders of magnitude in performance improvements over Apriori on some data-sets. On other data-sets where the patterns are not so long, the gains are more modest. In practice, Max-Miner is demonstrated to run in time that is roughly linear in the number of maximal frequent item sets and the size of the database, irrespective of the size of the longest frequent item set. We present a Max-miner algorithm, which searches for the MFS from bothbottom-up and top-down directions. It performs well even when the maximal frequent item sets are long. The bottom-up search is similar to Apriori algorithms. However, the top-down search is novel. It is implemented efficiently by introducing an auxiliary data structure, the maximum frequent set(or MFS), as explained later. By incorporating the computation of the MFS in our algorithm, we are able to efficiently approach the MFS from both top-down and bottom-up directions. Unlike the bottom-up search that goes up one level in each pass, the MFS can help the computation "move down" many levels in the top-down direction in one pass.

3.3.1 The Apriori Algorithm

The Apriori algorithm is a typical bottom-up approach algorithm. We describe it in some detail, as we will find it helpful to rely on this in presenting our results. The Apriori algorithm repeatedly uses Apriori-gen algorithm to generate candidates and then count their supports by reading the entire database once. The techniques we introduce in this paper are flexible and can be extended in various ways and applied to other algorithms. To demonstrate this point, we optimize Apriori with the lower bounding technique mentioned above. While the fundamental limitations of Apriori with respect to pattern length remain, performance is improved by an order of magnitude on several datasets. We also show how Max-Miner can be extended to exploit additional pattern constraints during its search by creating a variant that identifies only the longest of the maximal frequent item sets in a data-set. This algorithm efficiently identifies all of the longest maximal frequent item sets even when the space of all maximal frequent item sets is itself intractably large.

3.3.2 The Max-Miner Algorithm

We begin with defining the necessary terminology for describing the Max-Miner algorithm. For simplicity of presentation, we will be dealing only with the problem of identifying frequent item sets. The application of our techniques to finding

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

other patterns (e.g. sequential patterns) is similar. The supportof an item set , denoted sup(I) , is the number of transactions that contain it. The min sup parameter will sometimes be specified as a percentage of the transactions in the data-set instead of as an absolute number of transactions. Max-Miner can be described using generic set enumeration tree search framework. The idea is to expand sets over an ordered and finite item domain The key to an efficient set-enumeration search is the pruning strategies that are applied to remove entire branches from consideration.. Max-Miner uses pruning based on subset infrequency, as does Apriori, but it also uses pruning based on superset frequency.

IV SIMULATION RESULT

1 Introduction to Max-Miner Algorithm

Max-Miner Max-Miner algorithm was recently proposed to discover the maximum frequent set. This algorithm partitions the candidate set into groups with the same prefix. Like Pincer-Search, it looks ahead at some long candidate item sets throughout the search. The main difference is the long candidate item sets that it examines. Max-Miner looks ahead at longest item sets that can be constructed from every group. A frequency heuristic is used to reorder the items such that the most frequent items appear in the most candidate groups. According to the experiments in the paper, this item-reordering heuristic improves the performance dramatically. So far, we only had the opportunity to perform preliminary comparison with the Max-Miner from the algorithmic point of view. We feel that Max-Miner and Pincer- Search could be complementary. One of the possibilities is to run Max-Miner in the first few passes and switch to Pincer-Search for the later passes.

4.1.1 Max-Miner at its top level: Max-Miner appears in accepts a data-set and the minimum support specified by the user. The while loop implements a breadth-first search of the set-enumeration tree that maintains every frequent item set encountered so long as it is potentially maximal**.**

4.1.2 Generating the initial candidate groups

The function Gen-Initial-Groups performs the initial scan over the data-set to identify the item domain and seed the search at the second level of the tree. Superset-frequency based pruning is performed by only expanding the sub-nodes of a candidate.

V.RESULTS AND DISCUSSION

Before the execution of Max-miner algorithm, the processor's state is as above,which explains the system performance on an idle time. This performance is calculated when no process runs and the system is said to be idle. The processor is said to be idle only when the performance is valid from 0% to 2%. This also gives us the memory details such as physical memory and kernel memory. Also number of running processes are also displayed in this. It contains two main things CPU usage and page file usage history.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

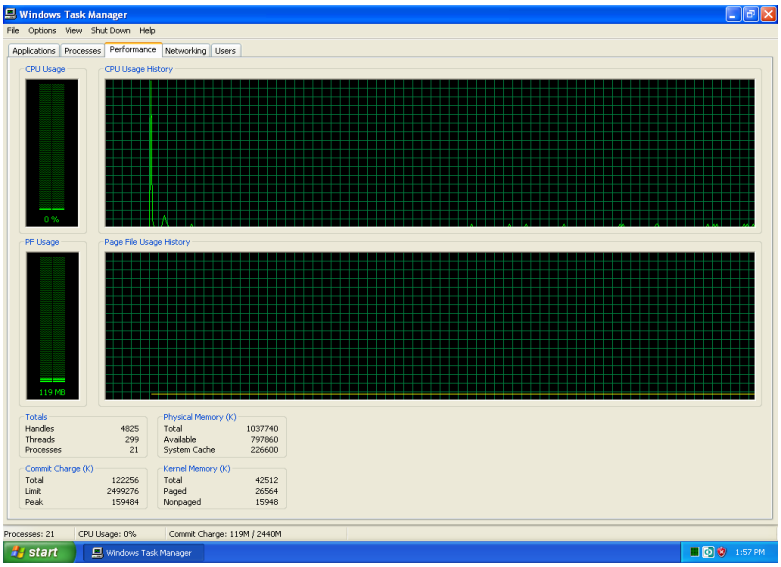

Figure 1 System ideal state

Before the execution of Max-miner algorithm, the processor's state is as above,which explains the system performance on an idle time. This performance is calculated when no process runs and the system is said to be idle. The processor is said to be idle only when the performance is valid from 0% to 2%. This also gives us the memory details such as physical memory and kernel memory. Also number of running processes are also displayed in this. It contains two main things CPU usage and page file usage history.

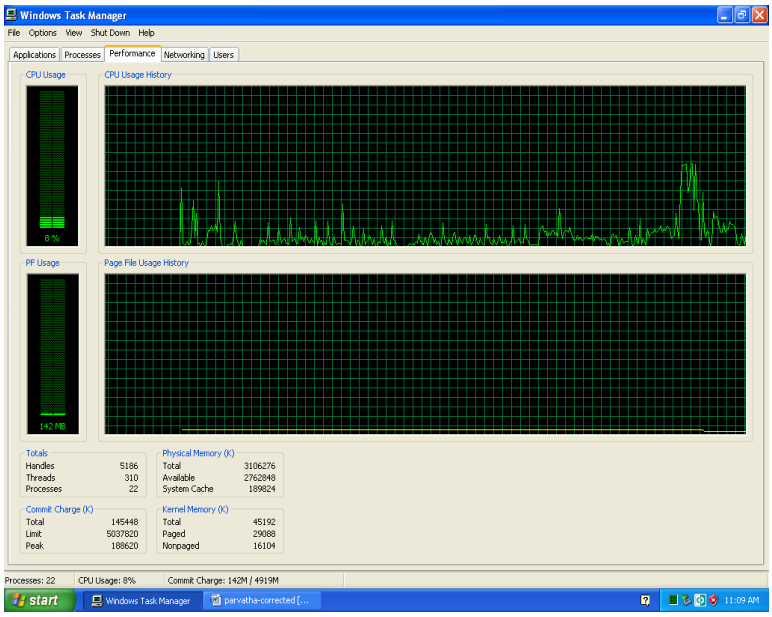

Figure 2 Process run state using Max-Miner algorithm

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

This figure explains us clearly how the CPU gets reduced when max-miner algorithm is implemented. This is a busy state, it is so called because the system executes something and therefore some process runs in this state. We can see it clearly from the above figure how the CPU usage history and also CPU usage varies from time to time (i.e.) for each process. Memory usages are also stated in this screen.

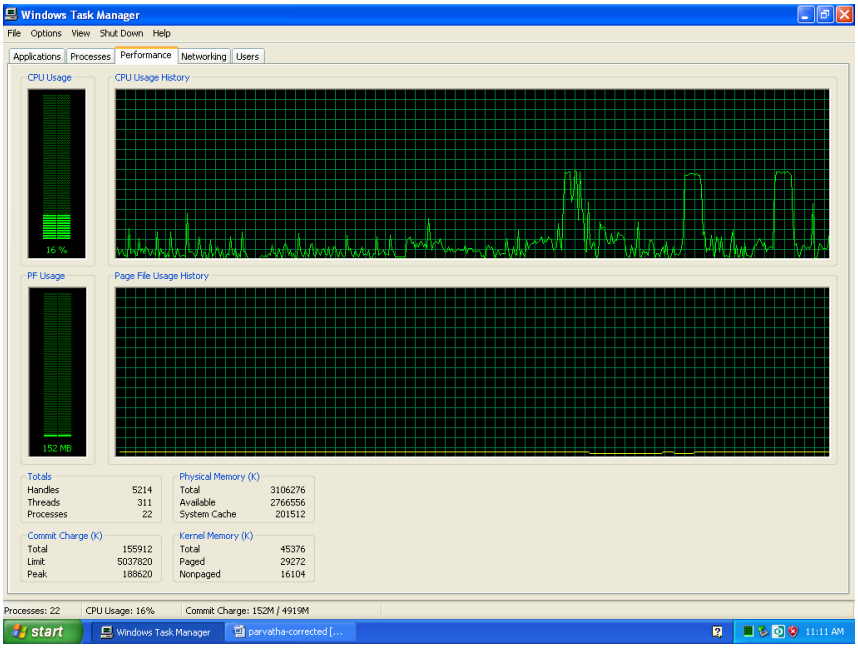

Figure3. Memory space using Max-Miner algorithm

This is similar to the third figure where the max miner algorithm is implemented here. It explains us clearly how memory is efficiently used when processes are running and also it shows the CPU usage history and page file usage history.

VI.CONCLUSION AND FUTUREWORK

We have presented and evaluated the Max-Miner algorithm for mining maximal frequent item sets from large databases. Max- Miner applies several new techniques for reducing the space of item sets considered through superset-frequency based pruning. The result is orders of magnitude in performance improvements over Apriori-like algorithms when frequent item sets are long, and more modest though still substantial improvements when frequent item sets are short. Max-Miner is also easily made to incorporate additional constraints on the set of frequent item sets identified. Incorporating these constraints into the search is the only way to achieve tractable completeness at low supports on complex datasets. we presented an algorithm that can efficiently discover the maximum frequent set. This algorithm could reduce both the number of CPU processing times and the number of search spaces. Experiments show that the improvement of using this approach can be very significant, especially when some maximal frequent item sets are long. A popular assumption is that the maximal frequent item sets are usually very short and therefore the computation of all frequent item sets is feasible. In this research work an approach to reduce the CPU process time is presented, Further work may be carried out to reduce the process time to near zero wait state, which is suggested as a future scope of research pertaining to this present work. For instance, if it takes 4% of process time for two processes, the work suggested is implemented, process time will reduce to less than 4% (i.e) reduce CPU time which makes a process to complete faster and also save memory and time.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

REFFERENCES

1) Srikant, R.; and Agrawal, R. 2011. Mining Association Rules with Item Constraints. In Proc. of theThird Int'l Conf. on Knowledge Discovery in Databases and Data Mining, 67-73.
2) Lin, D.-I and Kedem, Z. M. 2011. A New Algorithm for Discovering the Maximum Frequent Set. In Proc. of the Sixth European Conf. on Extending Database Technology, to appear.
3) R. Agrawal, A. Arning, T. Bollinger, M. Mehta, J. Shafer, R. Srikant. The Quest Data Mining System. In Proc. 2nd KDD, Aug. 2011.
4) R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In Proc. SIGMOD, May 2011.
5) R.Agrawal and J. Shafer. Parallel mining of association rules. IEEE Trans. on Knowledge and Data Engineering, Jan. 2011.
6) U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthrusamy (Eds.). Advances in Knowledge Discovery and Data Mining. AAAI Press, Menlo Park, CA, 2011
7) M. Houtsma and A. Swami. Set-oriented mining of association rules. Research Report RJ 9567, IBM Almaden Research Center, Oct. 2011.
8) H. Mannila and H. Toivonen. Levelwise search and borders of theories in knowledge discovery.Technical ReportTR C-1997-*8*, Dept. of Computer Science, U. of Helsinki, Jan. 2011.