MINISTERUL EDUCAŢIEI NAŢIONALE
ROMÂNIA

UNIVERSITATEA DE MEDICINĂ, FARMACIE, ŞTIINŢE ŞI TEHNOLOGIE „GEORGE EMIL PALADE" DIN TÂRGU MUREŞ

sciendo

Acta Marisiensis. SeriaTechnologica
Vol. 20 (XXXVII) no. 1, 2023
ISSN 2668-4217, ISSN-L 2668-4217

10.2478/amset-2023-0006

EFFICIENT ALGORITHMS FOR PATTERNS IDENTIFICATION IN MEDICAL DATA

Avram CALIN[1], Adrian GLIGOR[2], Victoria NYLAS[3], Roman DUMITRU[4]

[1,2,3]*George Emil Palade University of Medicine, Pharmacy, Science and Technology of Targu Mures*
Gh. Marinescu, 38, Târgu Mureş, Mureş, 540142, ROMANIA
[1]calin.avram@umfst.ro
[2]adrian.gligor@umfst.ro
[3]victoria.rus@umfst.ro

[4]*Sintef*
Forskningsveien 1, 0373 Oslo, Norway
[4]dumitru.roman@sintef.no

Abstract

Recently, medical databases have expanded rapidly, and the amount of information is huge. This abundance of data appears as a consequence of the new technologies that have been developed in the medical field and that allow easy data collection. The performance of the technique depends on the input data and available resources. Whereas, in Eclat the repeated scanning of the database is eliminated and consumes less time and we can conclude that Eclat is better than Apriori and Fpgrowth. If we refer to the execution time and memory usage, then the FP-Growth algorithm is more efficient than the Eclat algorithm or the Apriori algorithm. If we consider factor other than time, the result may vary from one factor to another.

Key words: big data, Eclat algorithm, Apriori algorithm, FP Growth Algorithm, Hopfield Algorithm

1. Introduction

Recently, medical databases have expanded rapidly, and the amount of information is huge. This abundance of data appears as a consequence of the new technologies that have been developed in the medical field and that allow easy data collection. Obtaining these large databases (big data) has opened up many research opportunities by analyzing the correlations in these large databases [1].

Big data represents a series of very large and complex data but which have a varied structure and confers a difficulty in handling these data and finding certain results [2,3].

Due to this large volume of data, which expands rapidly, information is continuously retrieved and there is a problem related to the storage of this data. The problem with these large data sets is finding useful information and identifying correlations between variables. Affiliation rules (mining of affiliation rules), which is an innovative method, are one of the most efficient ways to perform some processing and possibilities of identifying the functionality of the collected data [4].

The main problem in large medical data sets is primarily the identification of frequent measurements or those that provide information related to cross-presentations [4].

Finding data that have frequent sets of items from a set of transactions is a method found in many market studies that allows identifying the products that are most frequently bought [5]. This method of identifying the most frequent data sets can also be applied in medicine.

The most frequent algorithms used in such studies that also allow the identification of various related sets are Apriori, Fpgrowth and Eclat algorithms.

Data analysis is a very important element in scientific research because it allows the classification but also the analysis of data connections with the help of methods of grouping, arranging, waiting, learning, affiliation, etc. [6].

The medical data that we want to analyze with these algorithms allow the identification of an exact result for the purpose of research, but also a speed of execution as high as possible [7].

2. Methodology

We have proposed various techniques for the analysis of frequent data so that we have an efficiency of association rules. The data analyzed were medical data from a questionnaire applied to identify the health status of the patients. Data filled in the questionnaire are both quantitative and qualitative data. These data analyzes were carried out by extracting data with the help of horizontal layouts using the Apriori algorithm and vertical layout using the Eclat algorithm.

3. Results

The Eclat algorithm

Eclat is an acronym for equivalence class grouping and bottom-up network traversal. The algorithm is well known as the basic algorithm for Frequent Itemset Mining [8]. It was first proposed by Zaki et al in 1997 [9]. The algorithm checks the vertical database and the bottom-up technique to find the elements [10]. The algorithm uses the depthfirst search method with the advantage of the vertical layout for the database representation, where each item is denoted by a set of transaction IDs (called a tidset). The algorithm is implemented by intersecting itemset elements, and no counting support is needed, since the support of an itemset is annotated by the size of the itemset [1].

Considering the possibility of improving the Eclat algorithm, several variants were created. The traditional Eclat algorithm is known as tidset. The other 3 variants are as follows:

- Diffset: In addition to using tidsets, the algorithm uses the difference of tidsets (known as diffsets). The diffset approach will result in a lower cardinality of itemsets. Therefore, the speed of the intersection process can be improved with less memory consumption [11,12].
- Sortdiffset: Sortdiffset combines the tidset and diffset approach, then sorts the diffset in descending order. This approach is intentionally used to eliminate checking during the switch condition and changing the tidset to the diffset format. This approach is applicable to both sparse and dense databases [13].
- Postdiffset: The postdiffset approach starts with the tidsets process for the first level loop, then moves to the diffset approach for the second level forward [1].

An improvement to the Eclat algorithm has been proposed, which uses an incremental approach called Incremental-Eclat (i-Eclat). This improvement of the algorithm was applied to use a dynamic database in MySQL, incremental extraction is established for itemsets [11].

The benefit of these improvements to the incremental Eclat algorithm aims to generate a lower number of transactions during the Frequent Itemet Mining (FIM) process, as well as consume more time for execution. The algorithm is called Fast Incremental Eclat (Fi-Eclat). Through the incremental database concept, it has been improvised by adding more features in Fi-Eclat.

These new features include adding a filter to the null or empty value removed from the itemset, adding the minimum confidence threshold (MinConf) as cutoff criteria instead of using the minimum support threshold (MinSupp) as the main criteria for candidate itemset selection , using a text file to store transactions to reduce memory usage [1].

In checking the competitor's age, each k-item requester is created from two $k - 1 - itemsets$ and then the aid is calculated, assuming that its aid is less than the limit, it will be removed, otherwise it is set and continuous. used to produce $k + 1 - itemsets$. Because the ECLAT algorithm uses bottom-up design, checking the dataset is sometimes unpleasant. The applicant's age is definitely a hunt in the inquiry tree [4].

The Apriori algorithm

This is the most classical and important algorithm for extracting frequent sets of items from a database. Thus this algorithm is used to find all sets of frequent items in a given database. The key idea of the Apriori algorithm is to make multiple passes over the database. The Apriori algorithm is quite dependent on the Apriori property which states that "All non-empty itemsets of a frequent itemset must be frequent". He also described the anti-monotonic property which says that if the system cannot pass the minimum support test, all its supersets will fail the test. The Apriori algorithm follows the following phases [14]:

- Generation phase: In this phase, candidate itemset (k+1) is generated using k-itemset, this phase creates candidate itemset C_k.
- Pruning phase: In this phase, the candidate set is pruned to generate a large set of frequent items using “minimum support” as the parameter cut. This phase creates L_k set of large items.
- These disadvantages can be minimized by applying techniques for:
- Reduced transaction database scan passes
- Reducing the number of candidates
- Facilitates candidate count support

Apriori is a classic algorithm used for frequently mining datasets and learning association rules in transactional databases. The a priori algorithm is because the algorithm uses prior knowledge of frequent properties of itemsets [15]. This technique uses the property that any subset of a large itemset must be a large itemset. Apriori generates candidate itemsets by merging large itemsets from the previous pass and deleting those subsets that are small in the previous pass without considering database transactions.

An association rule is valid if its confidence and support are greater than or equal to the corresponding threshold values. Apriori uses an iterative approach known as level search, where k itemsets are used to explore (k+1) itemsets [4].

The FP Growth Algorithm

This is another important frequent pattern extraction method that generates a set of frequent items without candidate generation. It uses a tree-based structure. The problem of the Apriori algorithm was solved by introducing a new, compact data structure called a frequent pattern tree or FP tree, and then based on this structure, a FP tree-based pattern fragment growing method was developed.

Constructs the tree of frequent conditional patterns and the base of conditional patterns from the database that satisfy the minimum support. FP-growth tracks the set of competing elements [4].

The FP tree is built in two passes [4]:

Step 1: Scan the data and count support for each item

- Drop rare items
- Sort frequent items in descending order based on their support

Step 2: Read one transaction at a time and map it to the tree

- Fixed order is used so that the path can be shared
- Pointers are maintained between nodes that contain the same elements
- Frequent items are drawn from the list. He suffers from certain

Disadvantages:

- Fp tree may not fit in main memory
- Execution time is high due to complex compact data structure.

The Hopfield model

This algorithm is more complex than the previously presented ones.Hopfield neural networks are a particular kind of recurrent neural networks (RNN), for this reason, we may refer to them as Hopfield RNN. Hopfield RNN have several applications, [16].

Among others, these neural networks can be used as storage devices capable of storing patterns, and can model “associative memory”. The general concept in this applications is that the attractor states of these neural networks can be considered as stored patterns[17].

Extensive research has been devoted to enable the computing and functional applications of small-world neural network as a major candidate for functional approximation [18], early detection of diseases [19], associative memory [20], and so forth.

4. Discussion

Identifying datasets from large databases is a current research field, the frequent exploitation of item sets still faces a lot of challenges and problems that could lead to many issues that need to be discussed for further improvements. Since this research is based on only one type of data, an extension of future work could be performed on more types of data format, such as audio, video, and images. It can also be further studied on the database multiple number system.

Zaki (2000) [21] developed the Eclat algorithm which scans the dataset only once. Unlike the Apriori and FP-Growth algorithms, the Eclat algorithm uses a vertical dataset and a depth-first search approach. The only support value is calculated in the Eclat algorithm. After calculating the support value for all items, they are compared to the minimum support value. Items with a support value greater than or equal to the minimum support value are generated in frequent item sets. Zhang et al. (2021) [22] suggested that the Eclat algorithm takes less time and is therefore more efficient. However, since this algorithm has to save sets of items repeatedly, it needs more memory space.

The Apriori algorithm has a peculiarity for the way in which the elements are coded, thus having an impact on the execution time compared to the Eclat algorithm. The reason is that item encoding does not only affect the number and size of gaps in counter vectors for Apriori. Sorting articles usually leads to better structure. For sorting, there are practically the same options for Apriori and Eclat [5].

Eclat is a kind of frequency setting algorithm that uses vertical data pointing. Apriori must scan the database repeatedly. However, Eclat only needs to scan the database twice and thus to FP-Growth. But FP-Growth must recursively build a new FP tree and launch the original FP tree. In some cases, the FP-Growth algorithm has very poor efficiency. Eclat calculates the degree of support of the itemset by computing the intersection of Tidset. The intersection operation is the main operation of Eclat. So there is no need to build and scan the complex data structure repeatedly. This paper proposes a kind of improved Eclat algorithm, Eclat+. Compared to Eclat, before calculating the degree of support of the candidate sets, Eclat+ first detects the degree of support. When the candidate set is a potentially frequent set, Eclat+ performs the intersection operation. [23]

The Apriori algorithm runs slower than Eclat, resulting in more time needed to generate the association rules. Apriori's performance is slower but requires more computing power compared to Eclat's performance, especially when lowering the minimum support threshold to which the algorithms are applied [24].

For identifying various patterns in the dataset, we can opt to avoid using Apriori in favor of faster association rule extraction algorithms. A shorter time to apply the Eclat algorithm can be partly explained by the fact that it tends to generate fewer rules than the Apriori algorithm. Comparing the distribution of the number of rules generated by the two algorithms for the number of articles per association rule, it was possible to observe how Eclat tends to generate association rules with fewer articles compared to Apriori [25].

According to some authors [26] the Apriori algorithm is an iterative approach known as level search. It is subject to repeated data scans. The ECLAT algorithm supports the "Depth First Search" strategy and requires the generation of a set of candidate elements. It does not require scanning the database every time. The FP-growth algorithm adopts the "Divide and Conquer" method and does not require the generation of a set of candidate elements. It is not subject to repeated data scans.

Authors have performed tests to evaluate the execution time. According to them [26] based on the evaluation of the experimental results of the performance characteristics such as running time and memory usage, it is shown that the performance of the FP-Growth algorithm is better than the Apriori and ECLAT algorithms.

5. Conclusions

Frequent pattern mining is an important task in extracting association rules. It has been found useful in many applications such as market basket analysis, financial forecasting and medical data, etc. In the classic algorithms like Apriori and Fpgrowth we use horizontal data shapes and here it consumes more time because we have to scan the database multiple times.

The performance of the technique depends on the input data and available resources. Whereas, in Eclat the repeated scanning of the database is eliminated and consumes less time and we can conclude that Eclat is better than Apriori and Fpgrowth. Here we only consider time as a factor. If we consider factor other than time, the result may vary from one factor to another.

Acknowledgement

This article was elaborated under the EEA Financial Mechanism 2014–2021, project 21-COP-0038, implemented by the George Emil Palade University of Medicine, Pharmacy, Science and Technology of Targu Mures, Romania, in partneship with the Sintef AS, Norway.

Funding

This work was realized with the EEA Financial Mechanism 2014–2021's financial support. Its content (text, figures, tables) does not reflect the official opinion of the Program Operator, the National Contact Point, or the Financial Mechanism Office. The responsibility for the information and views expressed herein lies entirely with the authors.

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] Man, M., Jalil, M.A. and Bakar, W.A. (2023) "Fi-Eclat: An enhancement of Incremental Eclat algorithm," 1ST INTERNATIONAL POSTGRADUATE CONFERENCE ON OCEAN ENGINEERING TECHNOLOGY AND INFORMATICS 2021 (IPCOETI 2021) [Preprint]. Available at: https://doi.org/10.1063/5.0110230.

[2] Jain, P., Gyanchandani, M. and Khare, N. (2016) "Big Data Privacy: A Technological Perspective and Review," Journal of Big Data, 3(1). Available at: https://doi.org/10.1186/s40537-016-0059-y.

[3] Yusof, M.K. (2017) "Efficiency of JSON for data retrieval in Big Data," Indonesian Journal of Electrical Engineering and Computer Science, 7(1), p. 250. Available at: https://doi.org/10.11591/ijeecs.v7.i1.pp250-262.

[4] Srinadh, V. (2022) "Evaluation of Apriori, FP growth and Eclat Association rule mining algorithms," International journal of health sciences, pp. 7475–7485. Available at: https://doi.org/10.53730/ijhs.v6ns2.6729.

[5] Borgelt, C. (2003). Efficient Implementations of Apriori and Eclat. Proceedings of the IEEE ICDM Workshop on Frequent Itemset Mining Implementations; 19 November 2003.

[6] Kumbhare, T.A. and Chobe, S.V., (2014) An overview of association rule mining algorithms. International Journal of Computer Science and Information Technologies, 5(1), pp.927-930.x

[7] Chun-Sheng, Z. and Yan, L. (2014) "Extension of local association rules mining algorithm based on Apriori algorithm," 2014 IEEE 5th International Conference on Software Engineering and Service Science [Preprint]. Available at: https://doi.org/10.1109/icsess.2014.6933577.

[8] R. Ishita, and A. Rathod, International Journal of Computer Applications **143**, 33-37 (2016).

[9] M. J. Zaki, S. Parthasarathy, M. Ogihara, and W. Li, Data Mining and Knowledge Discovery 1, 343-373 (1997).

[10] K. Maniktala, J. Singh, and R. K. Gurm, International Journal of Technology and Computing 2, 547-548 (2016).

[11] W. A. W. A. Bakar, M. Man, and Z. Abdullah, Telkomnika **18**, 562-570 (2020).

[12] M. Benjamin, High-speed inserts with MySQL, Available: https://medium.com/@benmorel/high-speed-insertswith-mysql-9d3dcd76f723 (Accessed 17 Jan 2023).

[13] W. A. B. W. A. Bakar, Z. Abdullah, M. Y. B. M. Saman, M. Man, T. Herawan, and A. R. Hamdan, “Incremental-eclat model: an implementation via benchmark case study,” in Advances in Machine Learning and Signal Processing, edited by J. S. Ping, L. W. Wai, H. A. Sulaiman, M. A. Othman, and M. S. Saat (Springer International Publishing, Switzerland, 2016), pp. 35-46.

[14] Panjaitan, S., Sulindawaty, Amin, M., Lindawati, S., Watrianthos, R., Sihotang, H. T., &amp; Sinaga, B. (2019). Implementation of apriori algorithm for analysis of Consumer Purchase Patterns. Journal of Physics: Conference Series, 1255(1), 012057. https://doi.org/10.1088/1742-6596/1255/1/012057

[15] Wang, H.-B., &amp; Gao, Y.-J. (2021). Research on parallelization of Apriori algorithm in Association Rule Mining. Procedia Computer Science, 183, 641–647. https://doi.org/10.1016/j.procs.2021.02.109

[16] Haykin, S. Neural Networks: A Comprehensive Foundation, 2nd ed.; Prentice Hall PTR: Upper Saddle River, NJ,USA, 1999.

[17] Beyond the Maximum Storage Capacity Limit in Hopfield Recurrent Neural Networks

[18] Hu X, Feng G, Li H, Chen Y, Duan S (2014) An adjustable memristor model and its application in small-world neural networks. In: 2014 international joint conference on neural networks (IJCNN). Beijing, China

[19] Fekete T, Beacher FDCC, Cha J, Rubin D, Mujica-Parodi LR (2014) Small-world network properties inprefrontal cortex correlate with predictors of psychopathology risk in young children: a NIRS study. Neuroimage 85:345–353

[20] Taylor NR (2013) Small world network strategies for studying protein structures and binding. Comput Struct Biotechnol J5(6):1–7

[21] Zaki, M. J. (2000). Scalable algorithms for association mining. IEEE Transactions on Knowledge and Data Engineering, 12(3), 372–390. doi:10.1109/69.846291.

[22] Zhang, X., Tang, Y., Liu, Q., Liu, G., Ning, X., & Chen, J. (2021). A fault analysis method based on association rule mining for distribution terminal unit. Applied Sciences (Switzerland), 11(11), 5221. doi:10.3390/app11115221.

[23] Li, Z. F., Liu, X. F., &amp; Cao, X. (2011). A study on improved Eclat data mining algorithm. Advanced Materials Research, 328-330, 1896–1899. https://doi.org/10.4028/www.scientific.net/amr.328-330.1896

[24] Gayathri, G. (2017). Performance comparison of Apriori, Eclat and FPGrowth algorithm for association rules learning. International Journal of Computer Science and Mobile Computing, 81-89.

[25] Robu, V., dos Santos, V. D. (2019). Mining frequent patterns in data using apriori and Eclat: A comparison of the algorithm performance and Association Rule Generation. 2019 6th International Conference on Systems and Informatics (ICSAI). https://doi.org/10.1109/icsai48974.2019.9010367

[26] Sinthuja, M., Puviarasan, N., Aruna P. (2017). Evaluating the Performance of Association Rule Mining Algorithms. World Applied Sciences Journal 35 (1): 43-53. Doi: 10.5829/idosi.wasj.2017.43.53