Available online at: https://ijact.in

Date of Submission	19/10/2018
Date of Acceptance	18/11/2018
Date of Publication	30/11/2018
Page numbers	2867-2873 (7 Pages)

An International Journal of Advanced Computer Technology

ISSN:2320-0790

COMPARATIVE ANALYSIS OF THE DATA STRUCTURE FOR MINING ALL FREQUENT ITEMSETS

Thaweesak Khongtuk

Faculty of Business Administration and Information Technology,
Rajamangala University of Technology Suvarnabhumi, Suphanburi, Thailand.
dragonsuphan69@gmail.com

Abstract: Discovering All Frequent Items is one of the most important steps in the association rules mining process. Typically, the minimum support is used as a criterion for selecting an interesting itemset. There are many researchers who focus to improve the efficiency of the entire data set algorithm in various ways. For example, data reduction, structuring of new data, and search space reduction. This study analyzes the advantages and disadvantages of four-type data structure: Map Itemset - Horizontal Data, Map Itemset - Vertical Data, Map Different Set -Horizontal Data, and Map Different Set - Vertical data. The experiment was conducted with 6 datasets, which are dense and sparse datasets from the UCI standard datasets. The results show that Map Differential - Horizontal Data can reduce the size of datasets better than other techniques that use dense datasets. Map Itemset - Horizontal Data can reduce the size of datasets better than other techniques that use sparse datasets.

Keywords: data mining; apriori algorithm; itemset.

I. INTRODUCTION

Association rule mining consists of two main steps: 1) Finding All Frequent Itemset and 2) Generating Rules. In these two steps, the first step is the process where most research is focused on development to improve the performance. Since it is a time consuming and memory intensive process, this research focuses on the optimization algorithm for all Frequent Itemset mining algorithms.

In the All Frequent Itemset mining process, the characteristics of the data set that are being studied are important factors affecting the performance of the algorithm. There are many researches focuses to improved algorithms by design the data structure for All Frequent Itemset mining, to make the dataset smaller. This will reduce the workload of the algorithm to find all frequent patterns.

Apriori [20] algorithm is the first algorithm that is very popular. This algorithm uses the method of reading data from the source data to test the order of occurrence. Until the end of the data. This algorithm was easy to understand later on, there were several researches that adopted the Apriori algorithm to improve and use other structured methods such as the Eclat [17] algorithm. Vertical data is a way to read data from a source file to create a new structure, considering that any data set. What happens in the transaction list? This structure makes it easier to count support by the intersection method in mathematics, rather than repeating the transaction data in the Apriori algorithm. The dEclat [14] algorithm is an algorithm developed from the Eclat [17] algorithm by changing the data structure.

Instead of storing each item in the transaction list, I changed that item. Does not appear in any lists the research suggests that the algorithm makes the data smaller for dense data sets. This research will be conducted to analyze all four types of data structures, what kind of structure is suitable for a data characteristic? That is the best possible reduction, to apply to the All Frequent Itemset mining algorithm.

II. RELATED WORK

There are numerous examinations on calculation for discovering all continuous itemsets effectively. The majority of the already proposed calculations for mining continuous itemsets could be isolated into two classes: hopeful age and example development [4, 5, 6]. Calculations dependent on competitor age originally built hopeful itemsets and afterward distinguished incessant itemsets from applicant itemsets. These calculations utilized on an enemy of monotone property, specifically Apriori [20], to prune unpromising itemsets. The Apriori property expressed that if any k-itemset was not visit, its (k +1)- itemset additionally couldn't be visit as well. The general system of these calculations could be portrayed as pursues. They produced hopeful (k+1)- itemsets in the (k+1) that progression utilizing regular k-itemsets created in the past advance, and tallied the backings of these applicant itemsets in the database to finding incessant itemsets. A considerable measure of studies, for example, [20], had a place with the class of hopeful age. The applicant age technique accomplished great execution by decreasing the span of hopefuls. Notwithstanding, past investigations uncovered that it was very costly for applicant age technique to over and again filter the dataset and check a vast arrangement of competitors by itemset coordinating [20].
Visit itemsets mining technique without the applicant age, the example development strategy stayed away from the requirement for competitor age by building complex structures that contained adequate data about successive itemsets inside the dataset. The FP-development calculation, proposed by Han et al. [4], was the work of art and basic example development calculation. FP-development had been appeared to be exceptionally productive in the mining of thick dataset as the FP-tree structure embraced by FP-development succinctly typifies adequate itemsets data and no applicant itemsets were created. In addition, comparable structure named Diffsets had been proposed to mine continuous itemsets [14], another sort of mining undertaking that was unique in relation to visit itemsets, and the test results demonstrated that Diffsets was viable and the calculation dependent on it was exceptionally effective beated those already proposed calculations [14, 17]. A few looks into [23] received diffsets way to deal with mine regular itemsets, Map Different Set (MapDiff) is information structure to smaller the diffset. It very well may be decrease the datasets to smaller.

III. THE PROPOSED METHOD

This section will be conducted to analyze all four types of data structures & the type of structure suitable for a data characteristic. This is the best possible reduction, to apply to the All Frequent Itemset mining algorithm.
Table 1 shows sample data. This is used to describe how to reduce data in different ways.

TABLE I: SAMPLE DATASET

Tid	*Item*
1	A, B, C
2	B, C
3	C, D
4	B, C, D
5	A, C, D

The steps are as follows.
1. Reads all data sets and counts the frequency of items.

TABLE II: THE FREQUENCY OF ITEMS

Item	*Frequency*
A	2
B	3
C	5
D	3

Minimum support = 3 is 1-frequent: {B, C, D}, {A} is infrequent.2. Take 1-frequent consideration to reduce the size of the data. Then there are four ways to measure the performance of reduce the dataset.
Approach 2.1: Data reduction based on Map Itemset-Horizontal data concept.

TABLE III: MAP ITEMSET-HORIZONTAL DATA

Tid	*Items*	*Itemset-Horizontal data*	*Map Itemset - Horizontal data*
1	A, B, C	{B, C}	{B, C}:2
2	B, C	{B, C}	-
3	C, D	{C, D}	{C, D}:2
4	B, C, D	{B, C, D}	{B, C, D}:1
5	A, C, D	{C, D}	-

Map Itemset-Horizontal data size reduction uses -1frequent to find Itemset-Horizontal data from the below formulas. Itemset-Horizontal data=(1-frequent) $\cap$ Items(1)

Then combine Itemset-Horizontal data matched by the frequency to reduce the number of Itemset-Horizontal data, which results in the Map Itemset-Horizontal data.

Approach 2.2: Data reduction based on Map Itemset-Vertical data concept.

TABLE IV: MAP ITEMSET-VERTICAL DATA

Items	*TidLists*	*Itemset-Vertical data*	*Map Itemset-Vertical data*
A	1, 5	Infrequent	Infrequent
B	1, 2, 4	1, 2, 4	B{1, 2, 4}:1
C	1, 2, 3, 4, 5	1, 2, 3, 4, 5	C{1, 2, 3, 4, 5}:1
D	3, 4, 5	3, 4, 5	D{3, 4, 5}:1

Approach 2.3: Data reduction based on Map Different set - Horizontal data concept [3].

TABLE V: MAP DIFFERENT SET - HORIZONTAL DATA

Tid	*Items*	*Different set-Horizontal data*	*Map Itemset-Horizontal data*
1	A, B, C	{D}	{D}:2
2	B, C	{D}	-
3	C, D	{B}	{B}:2
4	B, C, D	{ }	-
5	A, C, D	{B}	-

Approach 2.4: Data reduction based on Map Different set-Vertical data concept.

TABLE VI: MAP DIFFERENT SET-VERTICAL DATA

Items	*TidLists*	*Different Set-Vertical data*	*Map Different Set-Vertical data*
A	1, 5	Infrequent	Infrequent
B	1, 2, 4	{3, 5}	{3, 5}:1
C	1, 2, 3, 4, 5	{ }	-
D	3, 4, 5	{1, 2}	{1, 2}:1

TABLE VII: COMPARE THE SIZE OF DATA IN THE FOUR DATA STRUCTURE

Map Itemset-Horizontal data	*Map Itemset-Vertical data*	*Map Different Set -Horizontal data*	*Map Different Set-Vertical data*
{B, C}:2	B{1,2, 4}:1	{D}:2	B{3, 5}:1
{C, D}:2	C{1, 2, 3, 4, 5}:1	{B}:2	D{1, 2}:1
{B, C, D}:1	D{3, 4, 5}:1	-	-

Table 7 shows the comparison study in four data structure for reduce the size of data. Map Different Set -Horizontal data outperformed those other techniques.

TABLE VIII: ANALYZE THE CHARACTERISTIC OF DATA IN THE FOUR DATA STRUCTURE

Reduction technique	**Number of record**	**Number of Items**	**Number of items in datasets**	**Average items per record**
Map Itemset-Horizontal data	3	3	7	7/3=2.3
Map Itemset-Vertical data	3	3	11	11/3=3.3
Map-Different set-Horizontal data	2	2	2	2/2=1
Map-Different set-Vertical data	2	2	4	4/2=2

Based on the data contained in the sample data set, the Map Itemset-Horizontal data size reduction is most effective because of the smallest data size consider from the average item per record.

Support Calculations Techniques

Let minimum support = 3, 1-frequent = {B: 3, C: 5, D: 3}, and so can be counting support as table below:-

TABLE IX: SHOWS HOW TO CALCULATE THE SUPPORT COUNTING ON MAP ITEMSET-HORIZONTAL DATA, MAP ITEMSET-VERTICAL DATA,MAP DIFFERENT SET -HORIZONTAL DATA AND MAP DIFFERENT SET-VERTICAL DATA

All Frequent Itemsets	***Map Itemset-Horizontal data***	***Map Itemset-Vertical data***	***Map Different set - Horizontal data***	***Map Different set-Vertical data***
{B, C}:3	{B, C} ∩ {B, C}:2 ∩{B, C, D}:1	B{1,2,4}:1 ∩ C{1,2,3,4,5}:1	DS:5 – {B}:2	DS:5 - B{3,5}:1
{B, D}:1	{B, D} ∩ {B, C, D}:1	B{1,2,4}:1 ∩ D{3,4,5}:1	DS:5 - {B}:2 U {D}:2	DS:5 - B{3,5}:1 U D{1,2}:1
{C, D}:3	{C, D} ∩ {C, D}:2 ∩ {B, C, D}:1	C{1,2,3,4,5}:1 ∩ D{3,4,5}:1	DS:5 – {D}:2	DS:5 - U D{1,2}:1
{B, C, D}:1	{B, C, D} ∩ {B, C, D}:1	B{1,2,4}:1 ∩ C{1,2,3,4,5}:1 ∩ D{3,4,5}:1	DS:5 - {B}:2 U {D}:2	DS:5 - B{3,5}:1 U D{1,2}:1

IV. EXPERIMENTAL EVALUATION

All four methods were tested with a set of 6 datasets from the UCI standard data sources and analyzed for use in the all frequent Itemset mining algorithm.

Table 1 shows the record, item, Item, average item per record, and minimum support. If the value is greater than or equal to 10, the data set should be very dense. All Frequent Itemset tests are set up with a minimum support of 50% - 90%. The average length of records is greater than or equal to 10 with four sets of data, including Mushrooms, Accidents, Record link, and Retails. It is very dense. But after the tests, Mushrooms, Accidents and Record link were very dense. Since the results are out of the minimum support range, the Retails data set does not come out. It is expected that the data set will be less dense. Randomly set Minimum support from 6 % - 2%Get the desired result. Therefore, it is concluded that Retails is a low density data set. Just like Online retail and Powerc, which requires minimum support of 6% - 2% to get All Frequent Itemset.

TABLE X: CHARACTERISTICS OF DATASETS

No.	***Datasets***	***Record***	***Item***	***Avg. item per record***	***Minimum support***
1	Mushrooms	8,416	119	23	50% - 90%
2	Accidents	62,331	360	34	50% - 90%
3	Record link	574,913	27	10	50% - 90%
4	Retails	88,162	16,470	10	2% - 6%
5	Online retail	540,455	2,603	4	2% - 6%
6	Powerc	1,040,000	125	7	2% - 6%

TABLE XI: ANALYZE THE APPROPRIATE ALGORITHM ON MUSHROOMS DATASET

Technique	***Minsup***	***Record***	***Items***	***Items on dataset***	***Average item per record***
Map Itemset-Horizontal data	50%	57	13	517	9.07
	60%	25	8	141	5.64
	70%	7	5	25	3.57
	80%	7	5	25	3.57
	90%	5	4	14	2.80
Map Itemset-Vertical data	50%	13	8,416	79,472	6,113.23
	60%	8	8,416	55,696	6,962.00
	70%	5	8,416	39,424	7,884.80
	80%	5	8,416	39,424	7,884.80
	90%	4	8,416	32,600	8,150.00
Map-Different set-Horizontal data	50%	57	12	224	3.93
	60%	24	7	59	2.46
	70%	6	4	10	1.67
	80%	6	4	10	1.67
	90%	4	3	6	1.50
Map-Different set-Vertical data	50%	12	8,416	29,936	2,494.67
	60%	7	6,104	11,632	1,661.71
	70%	4	2,144	2,656	664.00
	80%	4	2,144	2,656	664.00
	90%	3	848	1,064	354.67

TABLE XII: ANALYZE THE APPROPRIATE ALGORITHM ON ACCIDENTS DATASET

Technique	***Minsup***	***Record***	***Items***	***Items on dataset***	***Average item per record***

Map Itemset-Horizontal data	50%	29,667	25	530,353	17.88
	60%	11,346	21	165,529	14.59
	70%	1,483	15	15,177	10.23
	80%	102	9	617	6.05
	90%	25	6	94	3.76
Map Itemset-Vertical data	50%	25	62,331	1,190,762	47,630.48
	60%	21	62,331	1,055,349	50,254.71
	70%	15	62,331	811,044	54,069.60
	80%	9	62,331	523,868	58,207.56
	90%	6	62,331	362,197	60,366.17
Map-Different set-Horizontal data	50%	29,666	25	211,322	7.12
	60%	11,345	21	72,737	6.41
	70%	1,482	15	7,068	4.77
	80%	101	9	301	2.98
	90%	24	6	56	2.33

TABLE XII: (CONT'D) ANALYZE THE APPROPRIATE ALGORITHM ON ACCIDENTS DATASET

Technique	***Minsup***	***Record***	***Items***	***Items on dataset***	***Average item per record***
Map-Different set-Vertical data	50%	25	62,252	367,513	14,700.52
	60%	21	60,065	253,602	12,076.29
	70%	15	52,397	123,921	8,261.40
	80%	9	26,173	37,111	4,123.44
	90%	6	10,564	11,789	1,964.83

TABLE XIII: ANALYZE THE APPROPRIATE ALGORITHM ON RECORD LINK DATASET

Technique	***Minsup***	***Record***	***Items***	***Items on dataset***	***Average item per record***
Map Itemset-Horizontal data	50%	185	10	1,184	6.40
	60%	124	9	717	6.02
	70%	74	8	370	5.00
	80%	18	5	56	3.11
	90%	18	5	56	3.11
Map Itemset-Vertical data	50%	10	574,913	4,837,622	483,762.20
	60%	9	574,913	4,543,664	504,851.56
	70%	8	574,913	4,162,014	520,251.75
	80%	5	574,913	2,831,643	566,328.60
	90%	5	574,913	2,831,643	566,328.60
Map-Different set-Horizontal data	50%	184	10	666	3.62
	60%	123	9	399	3.24
	70%	73	8	222	3.04
	80%	17	5	34	2.00
	90%	17	5	34	2.00
Map-Different set-Vertical data	50%	10	574,417	574,417	57,441.70
	60%	9	384,210	630,553	70,061.44
	70%	8	356,238	437,290	54,661.25
	80%	5	39,890	42,922	8,584.40
	90%	5	39,890	42,922	8,584.40

TABLE XIV: ANALYZE THE APPROPRIATE ALGORITHM ON RETAIL DATASET

Technique	***Minsup***	***Record***	***Items***	***Items on dataset***	***Average item per record***
Map Itemset-Horizontal data	2%	3,108	20	14,820	4.77
	3%	766	12	3,470	4.53
	4%	120	7	412	3.43
	5%	63	6	192	3.04
	6%	31	5	80	2.58
Map Itemset-	2%	20	74,925	178,345	8,917.25

Vertical data	3%	12	73,396	161,945	13,495.41
	4%	7	72,636	146,827	20,975.29
	5%	6	72,372	142,990	23,831.67
	6%	5	71,661	138,518	2,703.60
Map-Different set-Horizontal data	2%	3,109	20	47,360	15.23
	3%	767	12	5,734	7.48
	4%	120	7	435	3.63
	5%	63	6	192	3.05
	6%	31	5	80	2.58
Map-Different set-Vertical data	2%	20	88,162	1,584,895	79,244.75
	3%	12	88,162	895,999	74,666.58
	4%	7	88,157	470,307	67,186.71
	5%	6	88,135	385,982	64,330.33
	6%	5	87,714	302,292	78,458.40

TABLE XV: ANALYZE THE APPROPRIATE ALGORITHM ON ONLINE RETAIL DATASET

Technique	***Minsup***	***Record***	***Items***	***Items on dataset***	***Average item per record***
Map Itemset-Horizontal data	2%	537	38	1,433	2.67
	3%	231	21	567	2.45
	4%	91	11	206	2.26
	5%	61	9	136	2.23
	6%	38	7	83	2.18
Map Itemset-Vertical data	2%	38	408,415	825,590	21,726.05
	3%	21	346,630	610,395	29,066.43
	4%	11	279,937	419,308	38,118.91
	5%	9	251,974	369,774	41,086.00
	6%	7	218,726	310,182	44,311.71
Map-Different set-Horizontal data	2%	538	38	19,011	35.34
	3%	232	21	4,305	18.56
	4%	92	11	806	8.76
	5%	62	9	422	6.81
	6%	39	7	190	4.87
Map-Different set-Vertical data	2%	38	540,455	19,711,700	518,728.95
	3%	21	540,455	10,739,160	511,388.67
	4%	11	540,455	5,525,697	502,336.09
	5%	9	540,455	4,494,321	499,369.00
	6%	7	540,455	3,473,003	496,143.29

TABLE XVI: ANALYZE THE APPROPRIATE ALGORITHM ON POWERC DATASET

Technique	***Minsup***	***Record***	***Items***	***Items on dataset***	***Average item per record***
Map Itemset-Horizontal data	2%	3,154	29	18,020	5.71
	3%	1,915	24	10,604	5.54
	4%	1,523	22	8,300	5.45
	5%	1,308	21	7,072	5.41
	6%	976	19	5,140	5.27
Map Itemset-Vertical data	2%	29	1,039,999	7,111,945	245,239.48
	3%	24	1,039,999	6,987,752	291,156.33
	4%	22	1,039,999	6,920,548	314,570.36
	5%	21	1,039,996	6,875,239	327,392.33
	6%	19	1,039,996	6,766,326	356,122.42
Map-Different set-Horizontal data	2%	3,155	29	73,475	23.29
	3%	1,916	24	35,380	18.47
	4%	1,524	22	25,228	16.55
	5%	1,309	21	20,417	15.60
	6%	7	3	12	1.71
Map-Different set-Vertical data	2%	29	1,040,000	23,048,055	794,760.52
	3%	24	1,040,000	17,972,248	748,843.67
	4%	22	1,040,000	15,959,452	725,429.64
	5%	21	1,040,000	14,964,761	712,607.67
	6%	21	1,040,000	14,964,761	712,607.67

V. Conclusion

The results from Table 11-16 show that if the average of items per record was compared to the original datasets, the concept of four approaches could be reduced the data. Compared with Zaki's research[13, 16], it is concluded that the data is very dense, Diffset and low density data are more suitable for Diffset structures. In terms of data density, the Diffset data of a horizontal structure is more diminished. When the Map is identical, the frequency is stored. In the sparse dataset, Zaki concludes that the Vertical Tidset method is the best way to reduce data. In this work, we found that the Map Itemset-Horizontal data method was more resilient compared with the Map Itemset-Vertical data method.

Acknowledgment

We would like to thank all academic staff in our university for supporting us in this research project.

VI. References

[1] B. Goethals. 2002, Efficient Frequent Pattern Mining. Phd Thesis in Transnationale University Limburg.

[2] C.C.Aggarwal and J. Han, Frequent Pattern Mining. Springer International Publishing Switzerland, 2014.

[3] F.Flouvat, F. De Marchi and J. Marc Petit. A new classification of Datasets for frequent itemsets. J IntellInfSyst (2010) 34:1–19.

[4] J.Han. Data Mining: Concepts and Techniques Second Edition. University of Illinois at Urbana- Champaign. Elsevier, 2006.

[5] J.Han, J. Pei, and Y. Yin, Mining Frequent Patterns Without Candidate Generation, Proc. ACM-SIGMODInt'l Conf. Management of Data, pp. 1-12, May 2000.

[6] J.Han, et al, Frequent pattern mining: current status and future Directions.

[7] J. Zhu. 2014, Efficiently Mining Frequent ItemsetsFrom Very Large Databases. Phd. Thesis. Concordia University, Canada.

[8] K. Garg and D. Kumar. Comparing the Performance of Frequent Pattern Mining Algorithms. InternationalJournal of Computer Applications (0975-8887). V. 69. No.25, 2013.

[9] K.Gouda and M.J.Zaki. Efficiently mining maximal frequent itemsets. In 1st IEEE International Conference onData Mining (ICDM), pages 163-170, San Jose, November 2001.

[10] L.Vu, and G.Alaghband. Mining Frequent Patterns Based on Data Characteristics. IEEE Department ofComputer Science and Engineering, University of Colorado Denver, Denver, CO, USA.

[11] L.XU. 2011, A Study of Frequent Pattern Mining in Transaction Datasets. Thesis of Doctor of Philosophy inComputer Science, Massey University, Palmerston North, New Zealand.

[12] M.J.Zaki and K.Gouda. Fast Vertical Mining Using Diffsets.Rensselaer Polytechnic Institute, Troy, NY,USA. Kyushu University, Fukuoka Japan.

[13] Santhosh, M., Kavitha, S., Keerthana, R., Suganya, L., & Krishnakumar, S. (2016). Electronic Voting Machine Using Internet. InternationalCommunication and computer Technologies.

[14] M.J. Zaki, et al. New algorithms for fast discovery of association rules. In 3rdIntl. Conf. on KnowledgeDiscovery and Data Mining. 1997, 283-296.

[15] M.J. Zaki and W.M. JR. Data Mining and Analysis Fundamental Concepts and Algorithms. CambridgeUniversity Press. New York, NY 10013-2473, USA. 2014.

[16] P. Sengar and M. Barot. Discovering Frequent Patterns with New Mining Procedure. IOSR Journal of Computer Engineering (IOSR-JCE) V. 10, Issue 5. 2013, 32-37.

[17] Q.Zou, et al. Mining Association Rule from Tabular Data Guided by Maximal Frequent Itemsets. This researchis supported by the NIH PPG Grant #4442511- 33780 and the NSF IIS Grant #97438.

[18] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In J.B. Bocca, Proceedings of the 20thInternational Confrerence on Very Large Data Bases (VLDB'94), Santiago dc Chile, Morgan Kaufmann,1994.

[19] R. Agarwal, T. Imielinski, and A. Swami. Mining association rules Between Sets of Items in Large Databases.Proc. ACM SIGMOD International Conference on Management of Data (SIGMOD'93), Washington, D.C., USA, 2007.

[20] R.C. Agarwal, C.C.Aggarwal, and V.V.V.Prasad. Depth first generation of long pattern, In Proceedings of thesixth ACM SIGKDD International conference on Knowledge discovery and data mining, Pages 108-118, 2000.

[21] R.J. Bayardo. Efficiently Mining Long Patterns from Databases. Proc. ACM-SIGMOD Int'l Conf.Management on Data, pp. 85-93, 1998.

[22] T. Hu, et al. Discovery of maximum length frequent itemsets, Journal of Information Sciences. 2007.

[23] T. Khongtuk, et.al. MapDiff-FI: Map Different Sets for Frequent Itemsets Mining, International Conference onApplied Science and Technology (ICAST. 2018), Penang, Malaysia, 10-12 April 2018.

[24] T.Lu, S.Tian, and S.Wang. Header Table Recursion Algorithm for Mining Frequent Patterns. Advances in information Sciences and Service Sciences (AISS) Volume 5, No 2, 2013.

[25] X. Zhu, H. Deng and Z. Chen. A Brief Review on Frequent Pattern Mining. Jianghan University, Wuhan, China. IEEE, 2011.