RESULTS AND DISCUSSION

4. RESULTS AND DISCUSSION

The results obtained for the research work entitled "A Study on the Applicability of Compact Transaction Database using CT-Apriori and Compressed Trees using CT-PRO for Pattern Mining" is discussed in this section. The experiments were conducted in two stages. The first stage analyzed the performance of the algorithms by using traditional market basket transaction database (synthetic data) and the second stage using web log files as transaction database. The results are also compared with the conventional Apriori and FP-Growth algorithms. This chapter presents the results obtained.

4.1. TEST DATASETS

The two models selected were tested with two types of datasets. One is the synthetic data which mimic the market basket database and other is the web data which belong to a web log databases. The synthetic data sets used in experiments were generated using the procedure described by Agarwal and Srikanth (1994). These transactions mimic the actual transactions in a retail environment. The transaction generator takes the parameters shown in Table 4.1.

TABLE 4.1

PARAMETERS USED IN THE SYNTHETIC DATA GENERATION PROGRAM

PARAMETERS	MEANING
|D|	Total number of transactions
|T|	Average size of transactions
|I|	Average size of maximal potentially frequent itemsets
|L|	Number of maximum potentially frequent itemsets
N	Total number of items

The synthetic datasets used are named after these parameters. For example, the data set T10.I5.D20K uses the parameters |T| = 10, |I| = 5, and |D| = 20000. For all the experiments, data sets were generated for setting N = 1000 and |L| = 2000 since these are the standard parameters used by Agarwal and Srikanth (1994). |T| value was varied with four values 5, 10, 15 and 20 and |I| values chosen were 4, 8, 10 and 12. The number of transactions |D| is set to 50,000, 100,000, 200,000 and 300,000. Table 4.2 summarizes the data set parameter settings used during experimentation.

TABLE 4.2

PARAMETER SETTINGS OF SYNTHETIC DATA SETS

TRANSACTION DATABASE	|T|	|I|	|D|
T5I4D50K	5	4	50k
T10I8D100K	10	8	100k
T15I10D100K	15	10	100k
T20I12D200K	20	12	200k
T20I12D300K	20	12	300K

The web log file used during experimentation was obtained from http://kdd.ics.uci.edu/databases/msweb/msweb.html. It was created by sampling and processing the web logs of Microsoft Tools like Weblog Expert and Weblog Explorer were used to download the log file. Welog Expert and Weblog Explorer are freeware software tools which can be used to download server log files of a website. The only constraint the web log server should be a linux server and their protection mechanism should allow third parties to access the log files. The server log file was extracted for 15 days, from 01 July 2010 until 15 July 2010. The data records contain transactions of users who use www.microsoft.com and contain 38000 anonymous, randomly-selected user transactions. For each user, the data lists all the areas of the web site that user

visited in a one week time frame. The data set contains 32711 instances (transactions) with 294 attributes (items); each attribute is an area of the www.microsoft.com web site. All the experiments were conducted on Pentium IV machine with 256MB RAM running in Windows environment.

4.2. PERFORMANCE METRICS

While evaluating the algorithms used, compression ratio was considered to be the most important performance metric. Compression ratio is defined as the ratio between the original transaction database size to the compact database size. The compression results with regard to number of association rules were also analyzed.

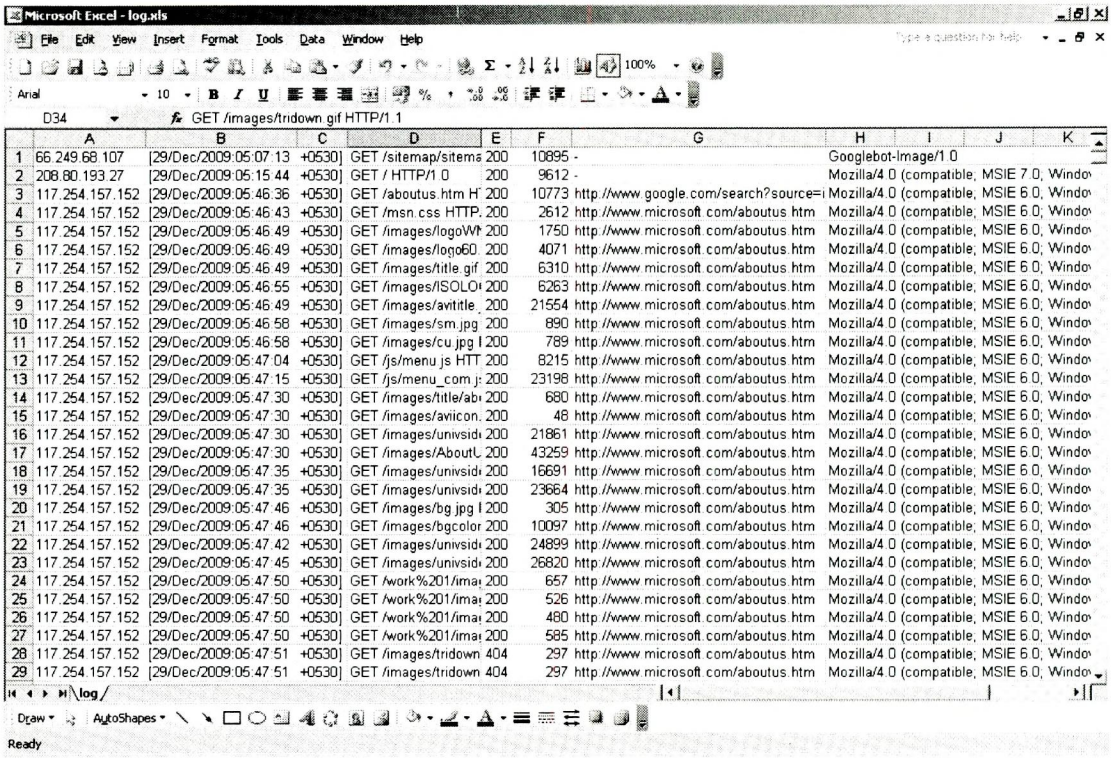
Apart from storage space required to store the resultant database, the amount of memory utilized during execution also plays a vital role during evaluation. The result of this metric can be used to evaluate the memory utilization complexity of the proposed algorithms.

Time taken to generate the association rules and mine frequent patterns was another parameter that was considered during evaluation. The algorithms were developed in JAVA with NetBeans 5.5 as front end. All the experiments were conducted on a Pentium IV machine with 512 MB RAM. The following section describes the results obtained.

4.3. PREPROCESSING RESULTS

Preprocessing is a step that was used only for converting a web log file into a format that is suitable for CT-Apriori and CT-PRO algorithms. The process is explained below. The part of the original log file is shown in Figure 4.1.

The first step of preprocessing removes all unsuccessful transactions (status code $\neq$ 200), image and video requests, transactions without IP addresses. The result of preprocessing is shown in Figure 4.2.



Microsoft Excel - log.xls

File Edit View Insert Format Tools Data Window Help

D34 GET /images/tridown.gif HTTP/1.1

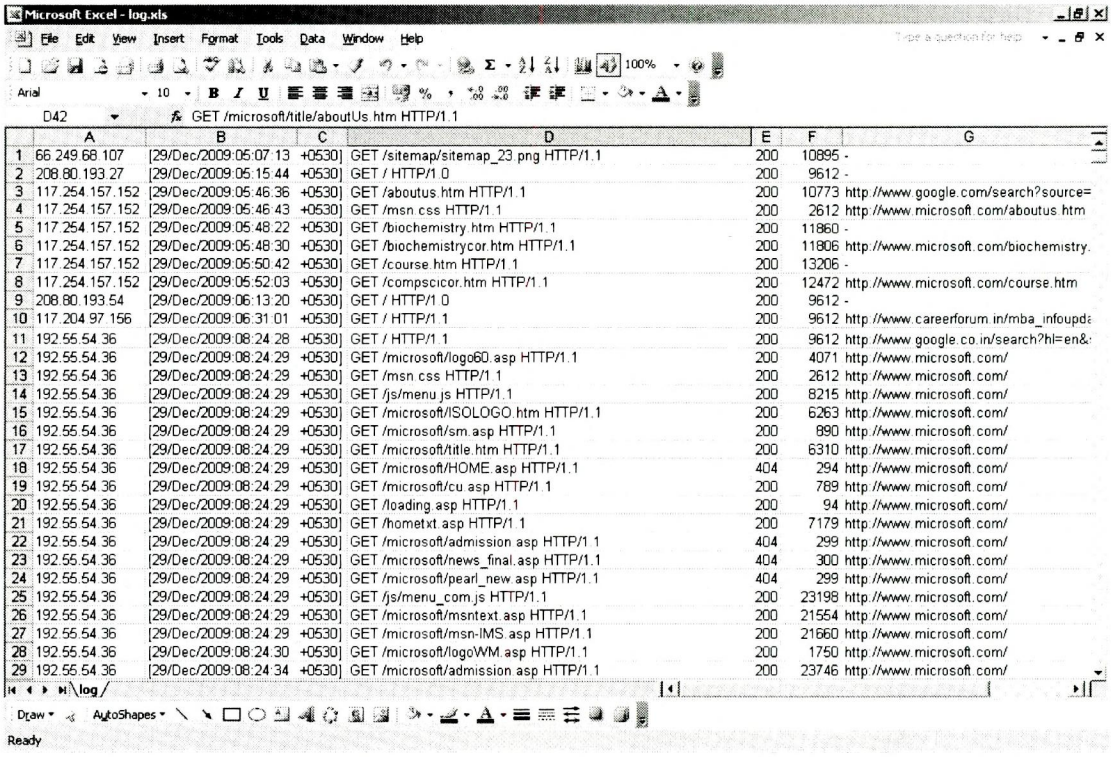
	A	B	C	D	E	F	G	H	I	J	K
1	66.249.68.107	[29/Dec/2009:05:07:13	+0530]	GET /sitemap/sitema	200	10895	-	Googlebot-Image/1.0			
2	208.80.193.27	[29/Dec/2009:05:15:44	+0530]	GET / HTTP/1.0	200	9612	-	Mozilla/4.0 (compatible; MSIE 7.0; Windo			
3	117.254.157.152	[29/Dec/2009:05:46:36	+0530]	GET /aboutus.htm H	200	10773	http://www.google.com/search?source=i	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
4	117.254.157.152	[29/Dec/2009:05:46:43	+0530]	GET /msn.css HTTP/	200	2612	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
5	117.254.157.152	[29/Dec/2009:05:46:49	+0530]	GET /images/logoWM	200	1750	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
6	117.254.157.152	[29/Dec/2009:05:46:49	+0530]	GET /images/logo60.	200	4071	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
7	117.254.157.152	[29/Dec/2009:05:46:49	+0530]	GET /images/title.gif	200	6310	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
8	117.254.157.152	[29/Dec/2009:05:46:55	+0530]	GET /images/ISOLO	200	6263	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
9	117.254.157.152	[29/Dec/2009:05:46:49	+0530]	GET /images/avititle.	200	21554	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
10	117.254.157.152	[29/Dec/2009:05:46:58	+0530]	GET /images/sm.jpg	200	890	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
11	117.254.157.152	[29/Dec/2009:05:46:58	+0530]	GET /images/cu.jpg I	200	789	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
12	117.254.157.152	[29/Dec/2009:05:47:04	+0530]	GET /js/menu.js HTT	200	8215	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
13	117.254.157.152	[29/Dec/2009:05:47:15	+0530]	GET /js/menu_com.j:	200	23198	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
14	117.254.157.152	[29/Dec/2009:05:47:30	+0530]	GET /images/title/ab	200	680	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
15	117.254.157.152	[29/Dec/2009:05:47:30	+0530]	GET /images/aviicon.	200	48	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
16	117.254.157.152	[29/Dec/2009:05:47:30	+0530]	GET /images/univsid	200	21861	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
17	117.254.157.152	[29/Dec/2009:05:47:30	+0530]	GET /images/AboutU	200	43259	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
18	117.254.157.152	[29/Dec/2009:05:47:35	+0530]	GET /images/univsid	200	16691	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
19	117.254.157.152	[29/Dec/2009:05:47:35	+0530]	GET /images/univsid	200	23664	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
20	117.254.157.152	[29/Dec/2009:05:47:46	+0530]	GET /images/bg.jpg I	200	305	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
21	117.254.157.152	[29/Dec/2009:05:47:46	+0530]	GET /images/bgcolor	200	10097	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
22	117.254.157.152	[29/Dec/2009:05:47:42	+0530]	GET /images/univsid	200	24899	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
23	117.254.157.152	[29/Dec/2009:05:47:45	+0530]	GET /images/univsid	200	26820	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
24	117.254.157.152	[29/Dec/2009:05:47:50	+0530]	GET /work%201/ima	200	657	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
25	117.254.157.152	[29/Dec/2009:05:47:50	+0530]	GET /work%201/ima	200	526	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
26	117.254.157.152	[29/Dec/2009:05:47:50	+0530]	GET /work%201/ima	200	480	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
27	117.254.157.152	[29/Dec/2009:05:47:50	+0530]	GET /work%201/ima	200	585	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
28	117.254.157.152	[29/Dec/2009:05:47:51	+0530]	GET /images/tridown	404	297	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			
29	117.254.157.152	[29/Dec/2009:05:47:51	+0530]	GET /images/tridown	404	297	http://www.microsoft.com/aboutus.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windo			

log

Draw AutoShapes

Ready

Figure 4.1 : Web log file



Microsoft Excel - log.xls

File Edit View Insert Format Tools Data Window Help

D42 GET /microsoft/title/aboutUs.htm HTTP/1.1

	A	B	C	D	E	F	G
1	66.249.68.107	[29/Dec/2009:05:07:13	+0530]	GET /sitemap/sitemap_23.png HTTP/1.1	200	10895	-
2	208.80.193.27	[29/Dec/2009:05:15:44	+0530]	GET / HTTP/1.0	200	9612	-
3	117.254.157.152	[29/Dec/2009:05:46:36	+0530]	GET /aboutus.htm HTTP/1.1	200	10773	http://www.google.com/search?source=
4	117.254.157.152	[29/Dec/2009:05:46:43	+0530]	GET /msn.css HTTP/1.1	200	2612	http://www.microsoft.com/aboutus.htm
5	117.254.157.152	[29/Dec/2009:05:48:22	+0530]	GET /biochemistry.htm HTTP/1.1	200	11860	-
6	117.254.157.152	[29/Dec/2009:05:48:30	+0530]	GET /biochemistrycor.htm HTTP/1.1	200	11806	http://www.microsoft.com/biochemistry.
7	117.254.157.152	[29/Dec/2009:05:50:42	+0530]	GET /course.htm HTTP/1.1	200	13206	-
8	117.254.157.152	[29/Dec/2009:05:52:03	+0530]	GET /compscicor.htm HTTP/1.1	200	12472	http://www.microsoft.com/course.htm
9	208.80.193.54	[29/Dec/2009:06:13:20	+0530]	GET / HTTP/1.0	200	9612	-
10	117.204.97.156	[29/Dec/2009:06:31:01	+0530]	GET / HTTP/1.1	200	9612	http://www.careerforum.in/mba_infoupd
11	192.55.54.36	[29/Dec/2009:08:24:28	+0530]	GET / HTTP/1.1	200	9612	http://www.google.co.in/search?hl=en&
12	192.55.54.36	[29/Dec/2009:08:24:29	+0530]	GET /microsoft/logo60.asp HTTP/1.1	200	4071	http://www.microsoft.com/
13	192.55.54.36	[29/Dec/2009:08:24:29	+0530]	GET /msn.css HTTP/1.1	200	2612	http://www.microsoft.com/
14	192.55.54.36	[29/Dec/2009:08:24:29	+0530]	GET /js/menu.js HTTP/1.1	200	8215	http://www.microsoft.com/
15	192.55.54.36	[29/Dec/2009:08:24:29	+0530]	GET /microsoft/ISOLOGO.htm HTTP/1.1	200	6263	http://www.microsoft.com/
16	192.55.54.36	[29/Dec/2009:08:24:29	+0530]	GET /microsoft/sm.asp HTTP/1.1	200	890	http://www.microsoft.com/
17	192.55.54.36	[29/Dec/2009:08:24:29	+0530]	GET /microsoft/title.htm HTTP/1.1	200	6310	http://www.microsoft.com/
18	192.55.54.36	[29/Dec/2009:08:24:29	+0530]	GET /microsoft/HOME.asp HTTP/1.1	404	294	http://www.microsoft.com/
19	192.55.54.36	[29/Dec/2009:08:24:29	+0530]	GET /microsoft/cu.asp HTTP/1.1	200	789	http://www.microsoft.com/
20	192.55.54.36	[29/Dec/2009:08:24:29	+0530]	GET /loading.asp HTTP/1.1	200	94	http://www.microsoft.com/
21	192.55.54.36	[29/Dec/2009:08:24:29	+0530]	GET /hometxt.asp HTTP/1.1	200	7179	http://www.microsoft.com/
22	192.55.54.36	[29/Dec/2009:08:24:29	+0530]	GET /microsoft/admission.asp HTTP/1.1	404	299	http://www.microsoft.com/
23	192.55.54.36	[29/Dec/2009:08:24:29	+0530]	GET /microsoft/news_final.asp HTTP/1.1	404	300	http://www.microsoft.com/
24	192.55.54.36	[29/Dec/2009:08:24:29	+0530]	GET /microsoft/pearl_new.asp HTTP/1.1	404	299	http://www.microsoft.com/
25	192.55.54.36	[29/Dec/2009:08:24:29	+0530]	GET /js/menu_com.js HTTP/1.1	200	23198	http://www.microsoft.com/
26	192.55.54.36	[29/Dec/2009:08:24:29	+0530]	GET /microsoft/msntext.asp HTTP/1.1	200	21554	http://www.microsoft.com/
27	192.55.54.36	[29/Dec/2009:08:24:29	+0530]	GET /microsoft/msn-IMS.asp HTTP/1.1	200	21660	http://www.microsoft.com/
28	192.55.54.36	[29/Dec/2009:08:24:30	+0530]	GET /microsoft/logoWM.asp HTTP/1.1	200	1750	http://www.microsoft.com/
29	192.55.54.36	[29/Dec/2009:08:24:34	+0530]	GET /microsoft/admission.asp HTTP/1.1	200	23746	http://www.microsoft.com/

log

Draw AutoShapes

Ready

Figure 4.2 : Removal of Irrelevant Transactions

The above file is stored in an ASCII file format and is fed as input to the compact transaction frequent pattern algorithms.

4.4. RESULTS

The results of the various experiments are presented and discussed in this section.

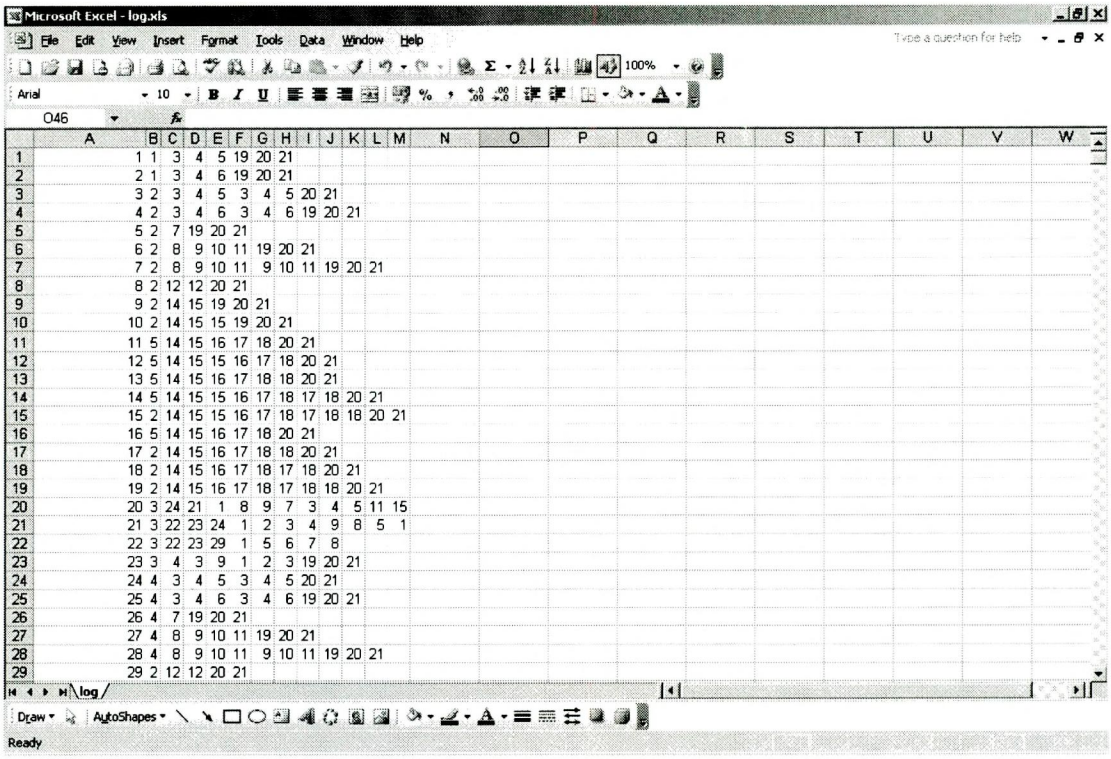
4.4.1. Compression Ratio

To evaluate the effectiveness of compact transaction databases, the compact transaction database was compared with the original database in terms of the size of the databases and the number of transactions in the databases. The compression result in terms of storage size is shown in Tables 4.3. Table 4.4 shows the compression performance in terms of number of association rules generated. .

TABLE 4.3

COMPRESSION RATIO IN TERMS OF DATABASE SIZE

Transaction Database	Original Size (KB)	CT-Apriori		CT-PRO	
		Compressed Size (KB)	Ratio (%)	Compressed Size (KB)	Ratio (%)
T5I4D50K	1,786	1,430	80.07	1,321	73.96
T10I8D100K	5,013	4,799	95.73	4,672	93.20
T15I10D100K	8,642	7,652	88.54	7,109	82.26
T20I12D200K	16,948	13,987	82.53	13,045	76.97
T20I12D300K	21,315	17,009	79.80	16,178	75.90
Web Data	545	344	63.12	287	52.66
Average Compression Ratio		**81.63**		**75.83**	



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1		1	1	3	4	5	19	20	21														
2		2	1	3	4	6	19	20	21														
3		3	2	3	4	5	3	4	5	20	21												
4		4	2	3	4	6	3	4	6	19	20	21											
5		5	2	7	19	20	21																
6		6	2	8	9	10	11	19	20	21													
7		7	2	8	9	10	11	9	10	11	19	20	21										
8		8	2	12	12	20	21																
9		9	2	14	15	19	20	21															
10		10	2	14	15	15	19	20	21														
11		11	5	14	15	16	17	18	20	21													
12		12	5	14	15	15	16	17	18	20	21												
13		13	5	14	15	16	17	18	18	20	21												
14		14	5	14	15	15	16	17	18	17	18	20	21										
15		15	2	14	15	15	16	17	18	17	18	18	20	21									
16		16	5	14	15	16	17	18	20	21													
17		17	2	14	15	16	17	18	18	20	21												
18		18	2	14	15	16	17	18	17	18	20	21											
19		19	2	14	15	16	17	18	17	18	18	20	21										
20		20	3	24	21	1	8	9	7	3	4	5	11	15									
21		21	3	22	23	24	1	2	3	4	9	8	5	1									
22		22	3	22	23	29	1	5	6	7	8												
23		23	3	4	3	9	1	2	3	19	20	21											
24		24	4	3	4	5	3	4	5	20	21												
25		25	4	3	4	6	3	4	6	19	20	21											
26		26	4	7	19	20	21																
27		27	4	8	9	10	11	19	20	21													
28		28	4	8	9	10	11	9	10	11	19	20	21										
29		29	2	12	12	20	21																

Figure 4.3 : Converted Web log file

TABLE 4.4

COMPRESSION RATIO IN TERMS OF NUMBER OF TRANSACTIONS

Transaction Database	Original Size	CT-Apriori		CT-PRO	
		Compressed Size	Ratio	Compressed Size	Ratio
T5I4D50K	50,000	37,878	75.76	36,077	72.15
T10I8D100K	1,00,000	86,928	86.93	85,765	85.77
T15I10D100K	1,00,000	89,347	89.35	87,621	87.62
T20I12D200K	2,00,000	1,62,421	81.21	1,46,411	73.21
T20I12D300K	3,00,000	2,41,931	80.64	2,11,113	70.37
Web Data	32,711	11,233	34.34	9,519	29.10
Average Compression Ratio		**74.70**		**69.70**	

From the results, it could be seen that both the algorithms are efficient in generating a compact version of the original database. While considering the storage size of the database, CT-Apriori on an average was able to compress the database by 81.63%, while it 75.83% by CT-PRO. The CT-Apriroi algorithm was able to achieve 74.70% compression ratio in terms of number of transactions while it was more (69.70%) in CT-PRO. In all the experiments conducted CT-PRO outperformed CT-Apriori algorithms.

While considering the web log data the algorithms were able to achieve more compression when compared to synthetic dataset. The algorithms achieved 63.12% and 52.66% compression ratio in terms of storage size required. The compactness achieved in terms of number of transactions was also high (34.34% and 29.10% for CT-Apriori and CT-PRO respectively).

The results show that the compact transaction databases provide effective data compression.

4.4.2. Execution Time

Both the selected algorithms were tested for the speed of execution with different support thresholds from 0.2 to 0.8% in steps of 0.2. Execution time while using synthetic database is shown in Figures 4.4a, b, c, d., e and Figure 4.5 shows the execution time for web log data.

The overall system performance is analyzed by comparing the average time taken by the selected algorithms. Table 4.5 presents the average time taken for synthetic datasets and web log data for various support thresholds.

TABLE 4.5

AVERAGE TIME TAKEN(Seconds)

Dataset	Apriori	CT-Apriori	FP-Growth	CT-PRO
T5I4D50K	214.75	180.75	186.75	159.00
T10I8D100K	302.25	265.50	272.5	237.75
T15I10D100K	343.00	315.75	297.5	268.50
T20I12D200K	396.75	362.50	350.5	325.50
T20I12D300K	439.75	409.75	394.50	351.75
Web Data	3.65	2.88	3.23	2.53

From the data projected in Table and Figures, showing the execution speed performance curves, it is evident that CT-PRO performs better than all the algorithms in all situations. Both CT-Apriori and CT-PRO outperforms their base algorithms Apriori and FP-Growth. Further, it can be seen that the execution speed is indirectly proportional to the support threshold, that is, when the threshold increases the algorithm becomes faster. Moreover, the smaller the threshold, the better the CT-Apriori and CT-PRO algorithm works. The performance gap between CT-Apriori and CT-PRO is more prominent at lower thresholds.

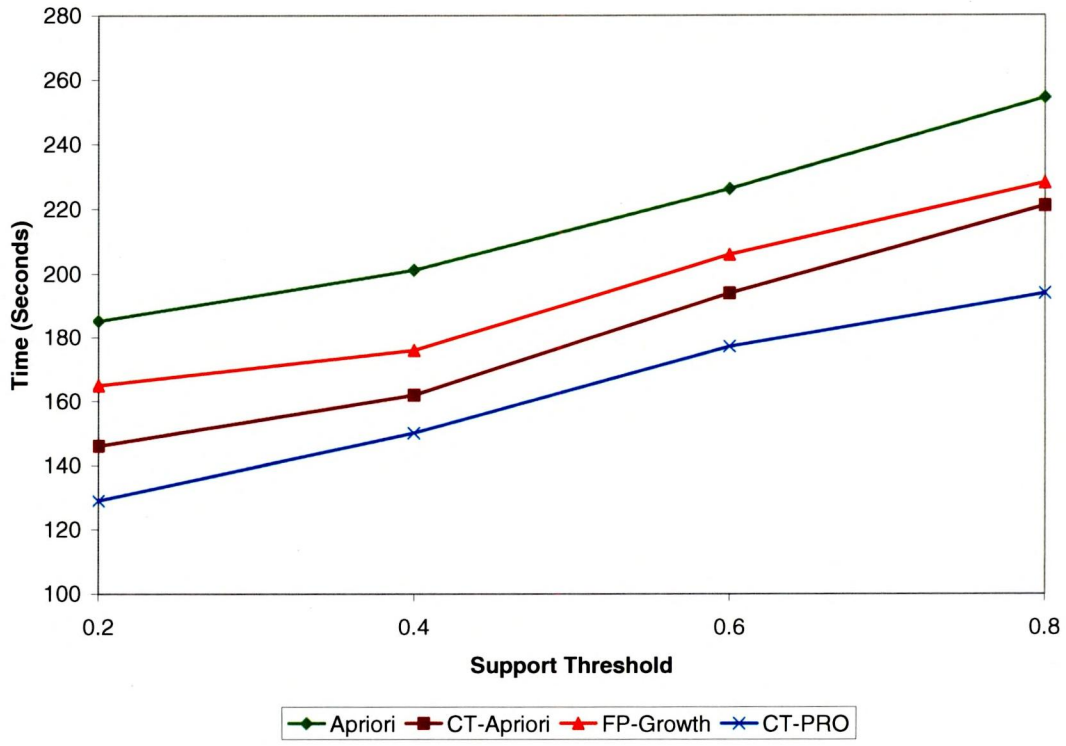


Figure 4.4a : T5I5D50K

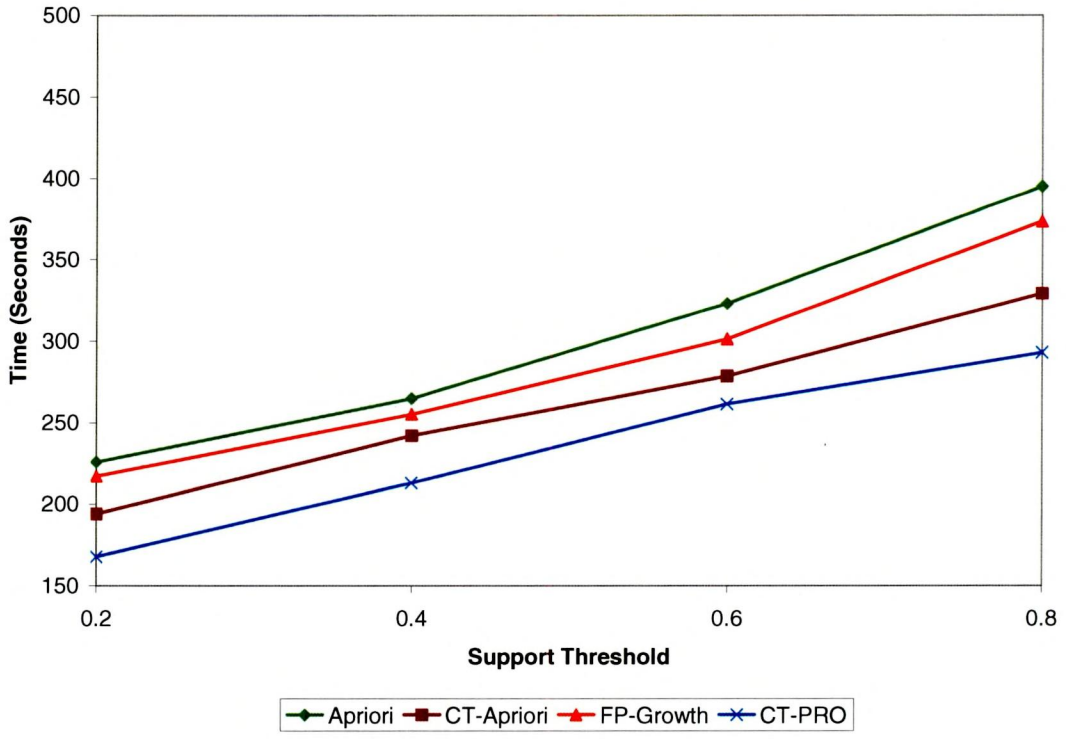


Figure 4.4b : T10I8D100K

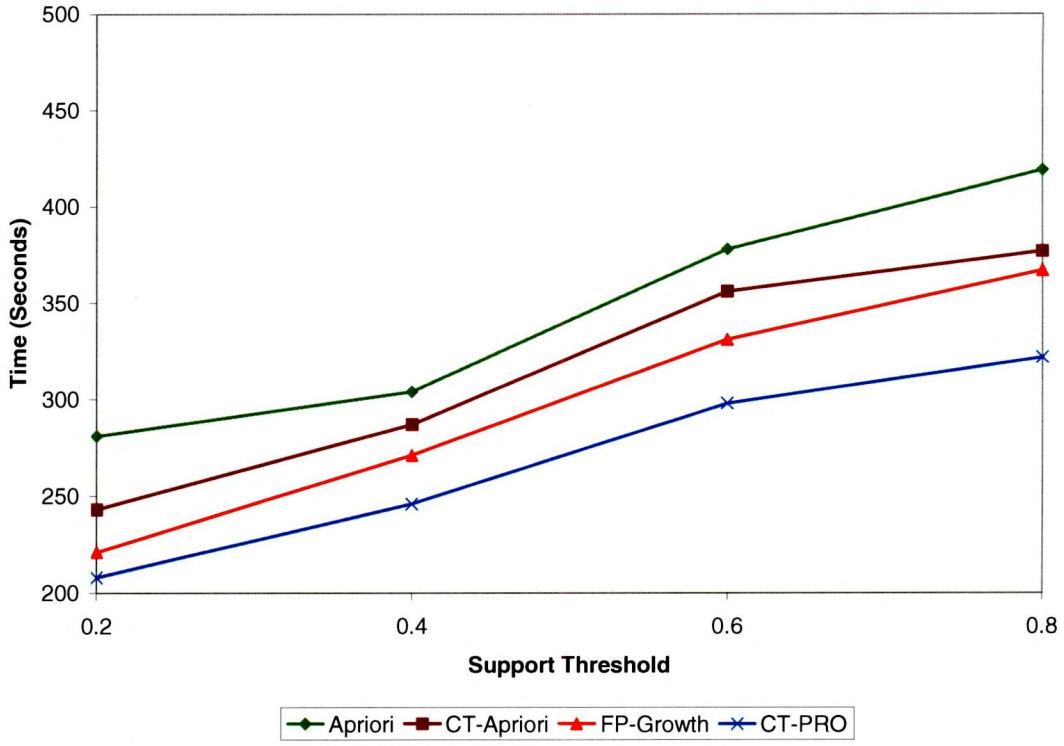


Figure 4.4c : T15I10D100K

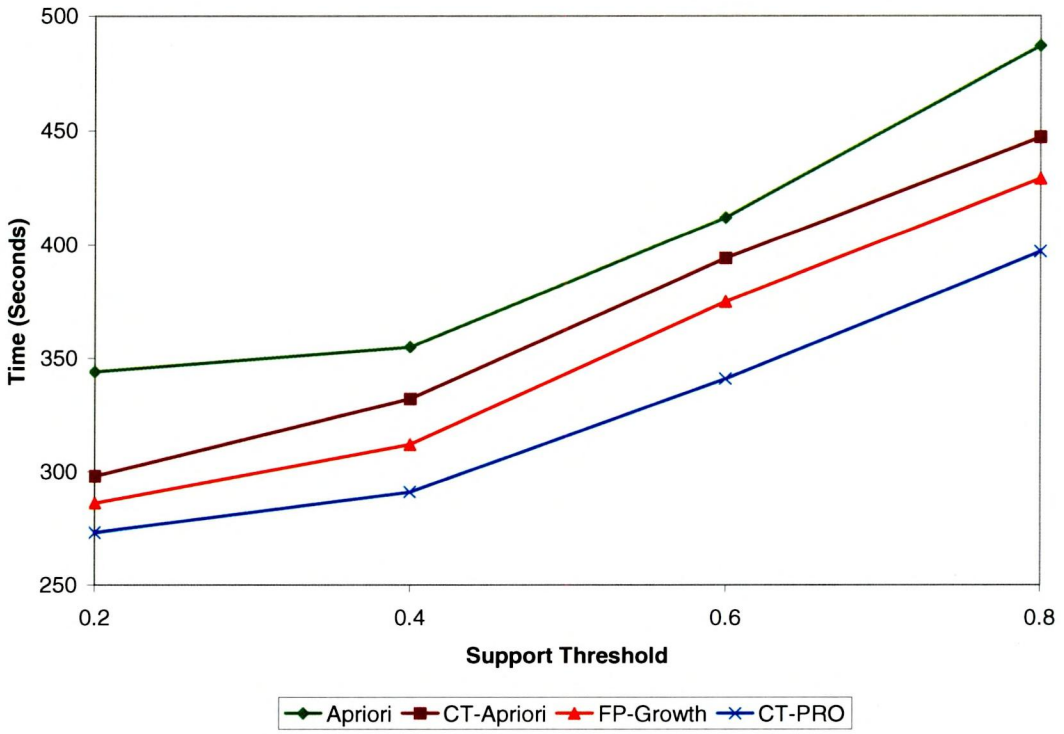


Figure 4.4d : T20I12D200K

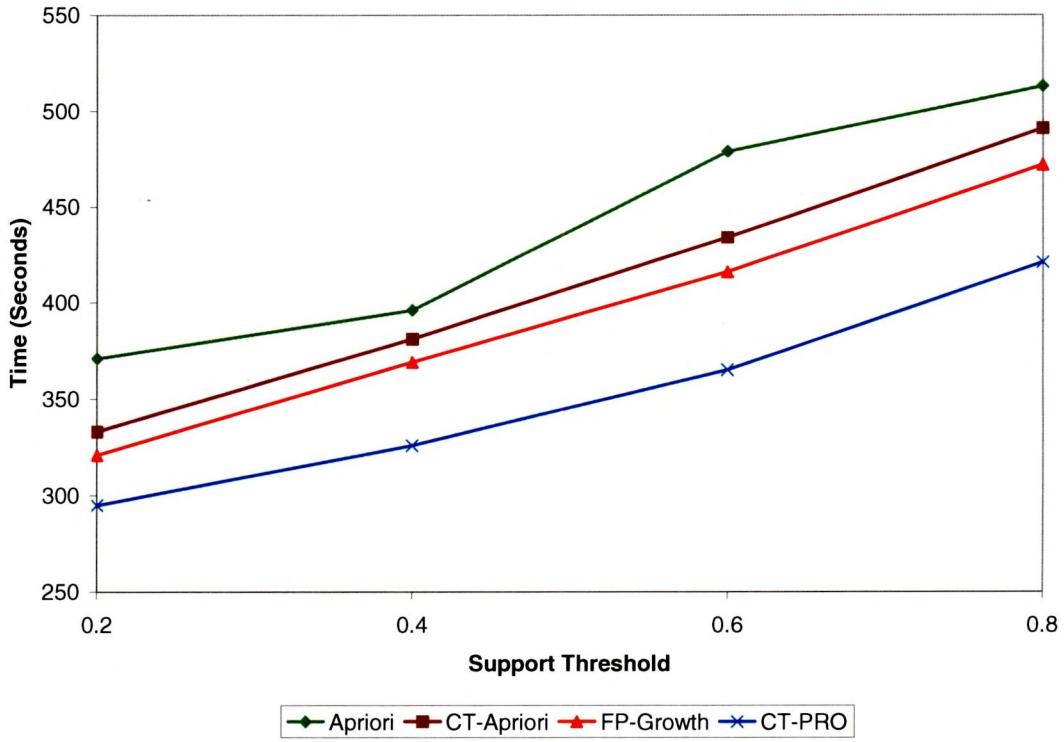


Figure 4.4e : T20I12D300K

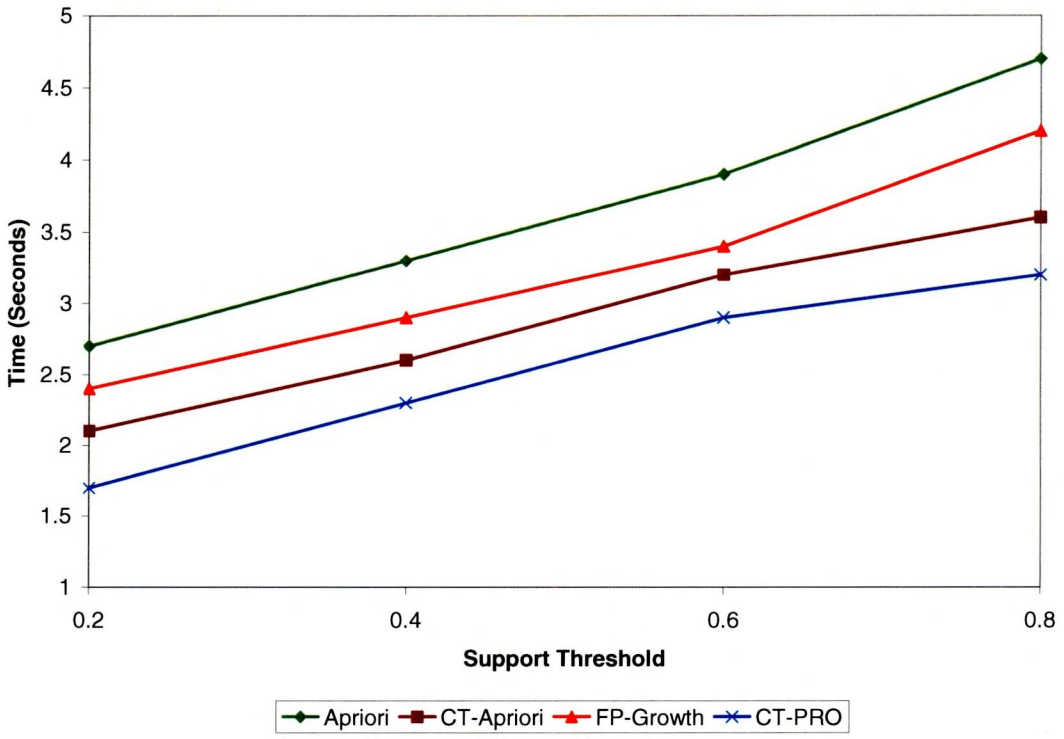


Figure 4.5 : Web log Data

The reason for this behaviour is because the number of scans performed becomes smaller in the selected algorithm. First, Apriori needs one complete database scan to find candidate 1-itemsets, while CT-Apriori can generate them from the head part of compact transaction database. Even though it takes time to construct a compact transaction database, the resultant compact transaction database can be used multiple times for mining patterns with different support thresholds. Second, when the support threshold gets lower, these two algorithms have to scan databases more times to discover the complete set of frequent patterns. For instance, the Apriori algorithm requires 18 passes over the database T15.I10.D200K when the support threshold is set to 0.2%. The CT-PRO algorithm on the other hand requires only two scans, which is done during the construction of the tree, after which the frequent mining process is performed only from the tree.

Efficiency and scalability have always been important concerns in the field of data mining. Data mining has two kinds of scalability issues

(i) Row (or database size) scalability and
(ii) Column (or dimension) scalability).

According to Hen and Kamber (2006), a data mining system is considered to be both row and column scalable, if, when the number of rows or columns (attributes or dimensions) is enlarged, the mining execution time increases linearly with the number of columns or rows. The same trend is observed by both algorithms (Figures 4.4 and 4.5) and hence can be concluded that both are efficient in terms of scalability also.

These results indicate that the performance of CT-PRO algorithm in terms of compactness achieved, in terms of storage size, number of transactions and execution speed with different datasets is efficient when compared with all the other algorithms.

4.5. CHAPTER SUMMARY

Two algorithms CT-Apriori and CT-PRO were selected in the present research work for producing compact transactional databases for frequent pattern mining. Experiments were conducted to analyze the performance of the algorithms. Both the algorithms can reliably be used in various applications where pattern mining is needed. Moreover, the speed of the algorithms further makes it suitable for online applications. The research work is summarized and concluded in the next chapter, Summary and Conclusion.