International Journal For Technological Research in Engineering
Volume 11 Issue 5 January-2024 ISSN (online) 2347-4718

ENHANCED DATA MINING TECHNIQUE FOR RAPID & EFFICIENT DECISION MAKING

Prashant Kumar* , Dr. Rakesh Kumar**
*Department of Computer Science & Engineering
Rabindranath Tagore University, India
Email- dceprashant0@gmail.com
**Department of Computer Science & Engineering
Rabindranath Tagore University, Raisen, Madhya Pradesh, India
Email- rakeshmittan@gmail.com

Abstract- Progress in digital and storage technology resulted in the growth of enormous databases. There are several real life domains, to name a few as supermarket transaction data, credit card data, mobile call details, bank transactions, government projects and medical records which generate bulky data every day. Digging out meaningful or useful information from bulky amount of data is a crucial and challenging task. Data mining normally deals with data which has already been collected from various resources for analysis purpose. It provides tools and techniques that are used to extract nontrivial significant information. It also assists in finding unidentified relationships and consolidate data in an intelligent way that is effortless to recognize and straightforward to exercise. The most important purpose of data mining is to determine hidden patterns, unexpected trends and relations in the data by using various techniques including statistical model, modern database technologies and machine learning tools to make businesses more proactive and knowledge based decisions

Keywords: Data mining, Rapid Mining etc.

1. INTRODUCTION

In the era of computers and smart phone the augmentation of data is incredibly high. The management of the voluminous data is a big challenge. In Dynamic data stream data are continuously added into dataset. Reducing size of data without loss of information is a complicated and challenging issue. There are several techniques have been developed to reduce space size of data.

Following two important issue which is always considers when the size of data to reduce.

1. Original database is transformed into a proper reduce format but whenever original data is required, it is needed to convert it again into original format, which is a reveres process.
2. Data to be reduced without any information loses.

Initially frequency based patterns are used to analyze shopping baskets of customers. The main objective is to discover regularities in customers shopping behavior. In general, it is a process to come across product purchase together frequently. Common applications are improving arrangement of products in shelf, Catalog's pages design, support in cross-selling fraud detection, technical dependence analysis etc. Let X is database of items and denoted as $X = \{x_1, x_2, \ldots, x_m\}$. Y is item set and $Y \subseteq X$ item set is set of products that are purchased by a customer. T is set of transactions denoted as $T = \{t_1, t_2, \ldots t_n\}$. Transaction over an item database X is a pair $t = (Tid, Y)$, where Tid is a unique transaction ID and $Y \subseteq X$ is an item set.

A transaction database contains transactions with the sets of items bought by the customers. Every transaction is collection of items, but some items could not appear in T. The support of item or item set is the number or fraction of transactions that contain it. Sometimes it is also called the frequency of item set with respect to T. The length of an item set is equal to the number of items presents in the data set; for example, if k is the length of an item set then k items were present in the transaction.

The fundamental approach is known as apriori approach which is basically a brute force approach that traverses all possible item sets, determines their support, and discards infrequent item sets is usually infeasible. In this approach the number of possible item sets grows up exponentially with the number of items. Fundamental approach is based on following important factors .

- Search procedure is a brute force (top down search from empty set/one- element sets to larger sets) approach that enumerates candidate item sets and checks their support.
- It improves search procedure by exploiting the apriori property and skip item sets that cannot be frequent.
- The search space is the partially ordered set (2^Y where $Y \subseteq X$). Partially ordered set helps to identify those item sets that can be skipped due to the apriori property.
- Since a partially ordered set can conveniently be depicted by a Hassel diagram, the search times are often acceptable only if the minimum support is not chosen very low.

the search procedure of traditional approach for five items. The search procedure is symbolized by Hasse diagram. Firstly, determine the support of the one element and discard the infrequent items, subsequent to frequent one item sets two item set must determine with support and discard the infrequent two item sets. Continuously, forming three, four and more frequent item sets until no candidate item set is frequent. The candidates which satisfy the give threshold value are used at higher level. Self joining is used to construct applicant at higher level. This procedureis repeated until no more candidates' generation is possible .

Problems with the fundamental approach were

3. Candidates generation is time consuming process and may carry out a lot ofredundant work
4. Repeatedly scanning of the database is to done for each time, need to checkthe candidates set to calculate the support value.

Due to these problems the performance of the fundamental approach decreases.

1. Hashing Technique

This technique is based on a hash function and a hash table. Consider a simple example with 9 transactions and five items.

Table 1 Simple truncation data set with 9 transactions

TID	Items
T1	x_1,x_2,x_5
T2	x_2,x_4
T3	x_2,x_3
T4	x_1,x_2,x_4
T5	x_1,x_3
T6	x_2,x_3
T7	x_1,x_3
T8	x_1,x_2,x_3,x_5
T9	x_1,x_2,x_3

Hash table structure contains bucket address, bucket count and bucket contents.

Table 2 Hash table for 2 items set for given 9 transactions

Bucket address	0	1	2	3	4	5	6
Bucket count	2	2	4	2	2	4	4
Bucket contents	x_1,x_4	x_1,x_5	x_2,x_3	x_2,x_4	x_2,x_5	x_1,x_2	x_1,x_3
	x_3,x_5	x_1,x_5	x_2,x_3	x_2,x_4	x_2,x_5	x_1,x_2	x_1,x_3
			x_2,x_3			x_1,x_2	x_1,x_3
			x_2,x_3			x_1,x_2	x_1,x_3

Bucket address is generated hash function given in equation number 1. For example item set (x1, x4) generate bucket number 0.Sequence of x_1 is 1 and sequence of x_4 is 4. Put these values in the equation number 1.

*h(x1, x4) =(1*10+4)mod7h(x1, x4)=0.*

Bucket count is the total number of item sets present in the bucket. For example in the bucket number 0 two item are present. Bucket contents are the item set in the bucket. (what is bucket address, bucket count and bucket contents) This technique reduces size of the candidate's item sets. After generating 1-itemsets, 2-itemsets for each transaction are directly hashed into the different buckets of a hash table. Bucket address is used to map 2 item set into appropriate bucket. Bucket count is increase according to the number of item sets present in the bucket.

Hash table for 2 items set by using hash function

$$h(x,y) = (sequnece\ of\ (x) \times 10 + (sequence\ of\ y))mod7 \quad 1.$$

The problem with this approach is creating hash function is a difficult task.

i. Partitioning of Dataset

This approach partitions the data set D into n number of parts. Partition is done in such a way that parts do not overlap each other.

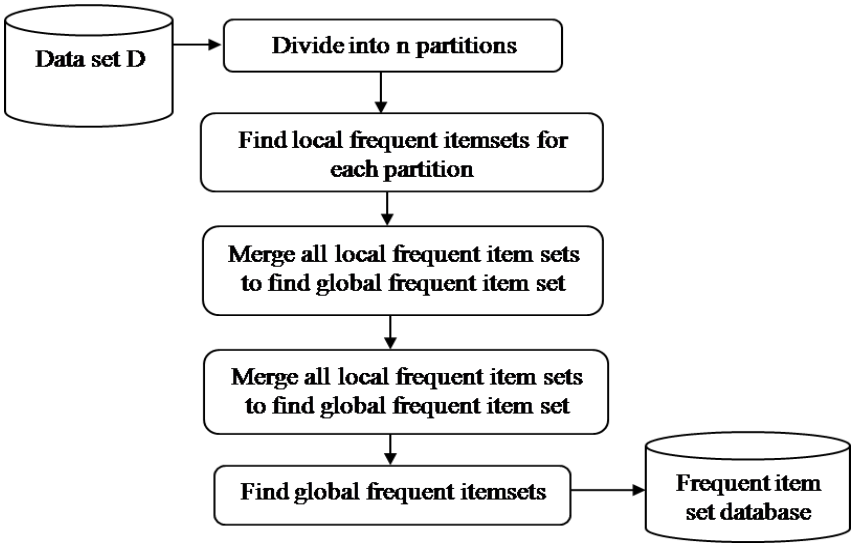


Figure 2 Partitioning approach.

For each partition, all frequent item sets within the partition are initiated. These are referred to as local frequent item sets. At this instant, merge all local frequent items and add respective support to discover global frequent item

set. Figure 2 exemplifies the working of partitioning approach. The efficiency depends on the size of partition and the number of partitions of the dataset, which is very complicated if not prepared appropriately.

ii. Maximal Frequent Patterns Based Approach

Maximal Frequent Pattern [1997] which was developed by Dao-I Lin & Zvi M. Kedem. This approach includes all maximal frequent sets. The maximal frequent pattern forms a border between frequent and infrequent pattern. Once the maximum frequent pattern is identified then all the frequent item sets can be computed easily by scanning database. This method is based on two way search from bottom-up and top- down search. Both searches apply at the time. This approach also used two important properties. First property is all maximal frequent patterns have subset and must also frequent. Seconds property all infrequent patterns have subset and they also infrequent.

iii. Transactions Condensing and Intersection Approach

This approach is proposed by Shui Wang et al in 2009 with the concept of maximal frequent pattern. The said approach used two steps to find frequent pattern:

(i) It scans the data base to come across one infrequent pattern and delete them. After that transactions are sorted in descending order as per the number of items in the transaction and calculate the support count of every transactions, subsequently merge the identical transactions with updating the count, this step is known as condensing operations and

(ii) Perform intersection operation to mine maximal frequent pattern. Useless subset is deleted by using intersections between transactions recursively.

2. PROPOSED APPROACH

The approach is based on two phases:

Phase I: - Create cluster of similar transaction based on the number of items and sequence of items present in the transaction.

Phase II: - Generate required pattern

Proposed approach is more accurate and efficient as compared to classical and transactions condensing approaches because it creates clusters of identical transactions. This approach arranges the transactions on the basis of the sequence of items, number of items and create cluster of similar transactions. In these clusters count of each item are maintain separately.

The proposed approach has following characteristics.

(1) Provides transactions variation and local control.

(2) Data compression is simple and easy.

(3) The size of the database is reduced form thousand to hundreds.

(4) Similar transactions are appeared in cluster conception

(5) Count of each items are separately manage while clustering.

(6) A separate database is managed for cluster database.

3. PROPOSED ALGORITHM

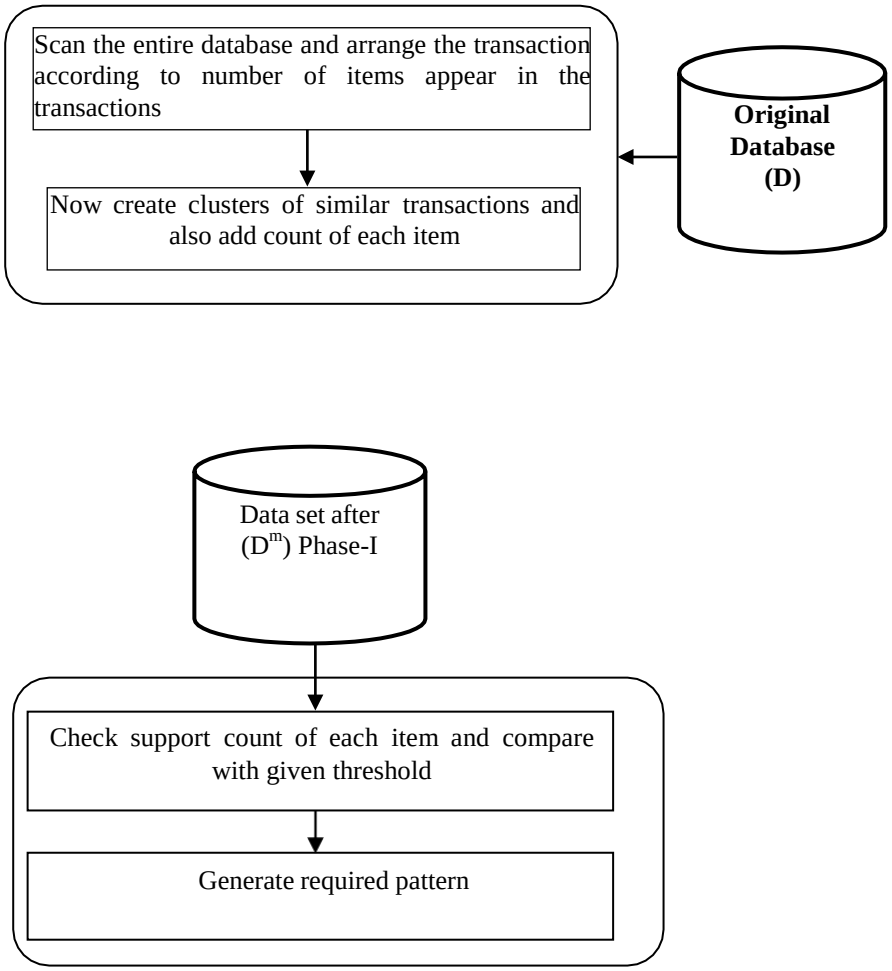


Figure 3 Architecture of proposed approach

Figure 3 explains the working of proposed approach. In the original data set transaction are arrange on the basis of number of items and sequence of the item present in the transitions to form clusters. Separate count of each item is maintained in every cluster. A separate data set is maintained for clusters transactions. Whenever new transactions are added to the database, new transactions are placed in most appropriate cluster and update the count value of corresponding item sets in the cluster. The item which has count less than the given support is not considering in the next level. The item which has support greater are equal to the given support are the required patterns.

Algorithm of Proposed Approach

a) Input:-Transactional data set D Output: - Required pattern Phase -I

Step 1: for all transactions $T_n \in D$

Scan transactions; arrange according to number of items and sequence of items.

Step2: for all transactions $T_n \in$ *Dif* $T_i \approx Ti+1$ *sup*(T_i) *+ sup*(T_i+1) *as* T_i *and delete* $Ti+1$ *cluster identical transactions*

Step3: repeat step 2 until no more group is possible

b) Phase II

Step4: *Let* D^m *clustered transactional data set* *for* $(k=1; L^m k+1 \neq \varphi; k++)$

for all transactions $T_m \in D^m$ *do begin*

$$c \in C_m^k \mid c.count > support$$

$$P = ULK$$

Consider a simple transactional data base with 15 transactions. Items are numbered x1, x2,…,x10.

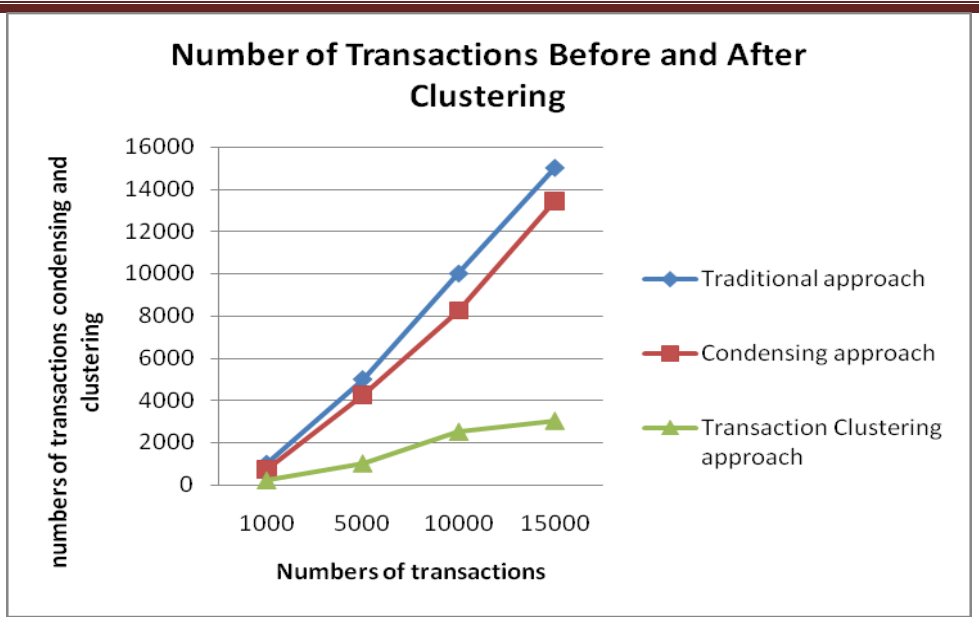
3.1. Comparisons Based on Number of Transactions Before and After Clustering

Table 1 Numbers of transactions before and after transactions clustering.

No of Transactions	Traditional Approach	Condensing approach	Transactions Clustering approach
1000	1000	725	204
5000	5000	4267	1006
10000	10000	8256	2508
15000	15000	13452	3025

Table 1 contains four columns the first column shows total number of transactions, second column shows traditional approach, third column shows transactions condensing approach and the last column shows proposed approach. For 1000 transactions, traditional approach used apriori based techniques to generate required pattern without any data compression. Transactions condensing approach compress identical transaction and treat them as single transaction. Transactions condensing approach compress 1000 transactions into 725 and reduce the size of the transactions. Proposed approach creates clusters of similar transactions reduce size of the dataset up to 204, which is very less in number as compared to transactions condensing approach. For 5000 transactions, condensing approach compress up to 4267 and proposed approach reduce size of transaction up to 1006. For 10000 transactions condensing approach compress the 10000 transactions into 8256 and proposed approach reduce up to 2508. For 15000 transactions condensing approach compress the 13452 and proposed approach reduce up to 2508.

Graph 4 shows the comparison between traditional, data condensing and proposed approach based on number of transactions. Blue line show the traditional approach, red line show condensing approach and line with green color represents the proposed approach. From the graph it is clear that data condensing approach reduce size of the data base up to 72.5% percentage, where proposed approach reduce size of the database 20.4 %. Proposed approach reduces size of original data set in meaningful amount using clustering. Proposed approach also reduces search space because most of the transaction is grouped on the basis of similarity.



Graph 4. Numbers of transactions before and after transactions clustering Consider item x1 and x2 both consider in the first group, they satisfy the support value so there is no need to search these item in any other cluster. Similarly x1 and x4present in the second cluster and both have count value greater the threshold value so they feasible and not need search in any other cluster even they present in the first cluster x1 with count 5 and x4 with count 1.Transaction condensing approach delete those item which has support value less than given threshold but in dynamic data set transaction are added regularly in the dataset ,it may be possible that after insertingsome new transaction the infrequent item become frequent, it is major drawback ofthis approach.

3.1 Comparisons Based on Number of Transactions and RequiredExecution Time

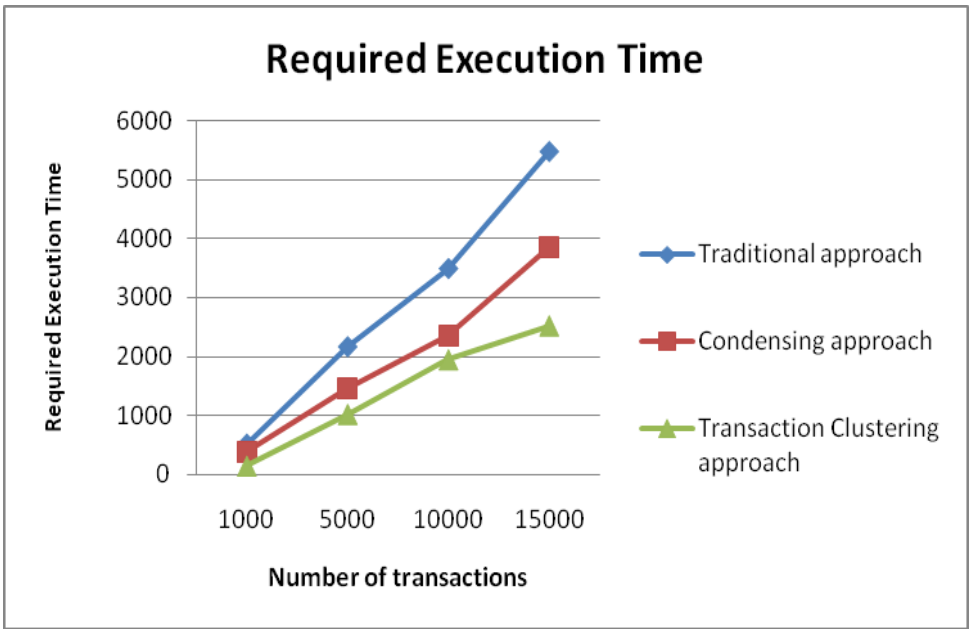
Table 4.2 Number of transactions and execution time

No of Transactions	Traditional Approach	Condensing Approach	Transactions Clustering approach
1000	504	370	146
5000	2156	1462	1022
10000	3482	2342	1948
15000	5468	3856	2522

Table 3.3 shows number of transactions and required execution time for all three approaches for threshold value 10 percentage. Traditional approach need 504 milliseconds, condensing approach need 370 milliseconds and proposed approach need only 146 milliseconds for 1000 transactions. Traditional approach need 2156 milliseconds, condensing approach need 1462 milliseconds and proposed approach need only 1022 milliseconds for 5000 transactions. Traditional approach need 3482 milliseconds, condensing approach need 2342 milliseconds and proposed approach need only 1948 milliseconds for 10000 transactions. Traditional approach need 5468 milliseconds, condensing approach need 3856 milliseconds and proposed approach need only 2522 milliseconds for 15000 transactions.

Graph 5. shows the comparison between traditional, data condensing and proposed approach based on execution time and number of transactions. For 1000 transactions traditional approach need 504 milliseconds and

data condensing approach reduce required time up to 370 milliseconds and proposed approach reduce time up 146 milliseconds. It is clear that proposed approach reduce execution time up to 28.96 percent. For 5000 transactions proposed approach reduce execution time up to 47.40 percentages as compared to traditional approach.



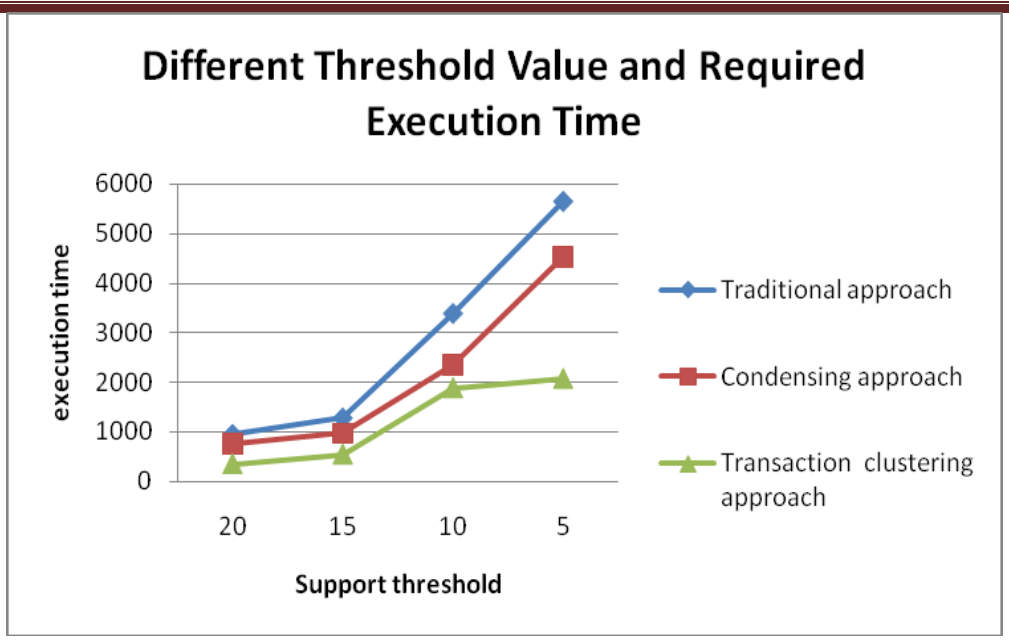
Graph 5 Numbers of transactions and required execution time

For 10000 transactions proposed approach reduce execution time up to 55.94 percentages as compared to the traditional approach. Similarly 15000 transactions proposed approach reduce execution time up to 46.12 percentages as compared to the traditional.

3.2 Comparisons Based on Different Threshold Value and RequiredExecution Time for 10000 Transactions.

Table 3.4 Support in percentage and execution time

Support threshold	Traditional approach	Condensing approach	Transactions clustering approach
20	943	762	353
15	1274	982	548
10	3384	2352	1896
5	5644	4541	2083



Graph 6 Comparison graph using different support and execution time

For the small support value number of pattern is are huge and need more execution time. When support value is high the number of pattern fewer and need less execution time. Table 4.3 shows different support value and required execution time. For 20 percentage of support value traditional approach need 943 milliseconds, data condensing approach need 762 milliseconds and proposed approach need only 353 milliseconds of time. For 15 percentage of support value traditional approach need 1274 milliseconds, data condensing approach need 982 milliseconds and proposed approach need only 548 milliseconds of time. For 10 percentage of support value traditional approach need 3384 milliseconds, data condensing approach need 2325 milliseconds and proposed approach need only 1896 milliseconds of time. For 5 percentage of support value traditional approach need 5644 milliseconds, data condensing approach need 4541 milliseconds and proposed approach need only 2883 milliseconds of time.

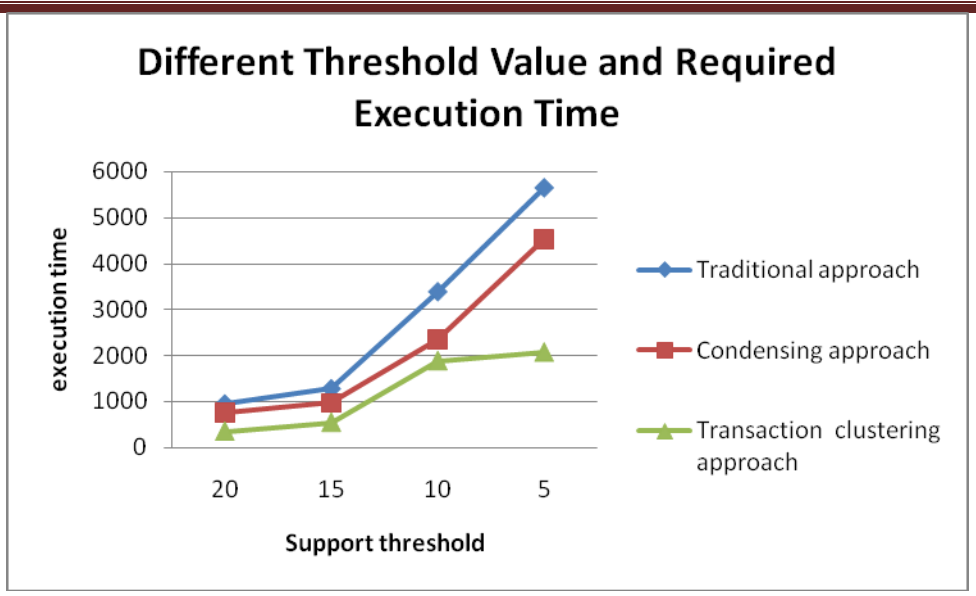
Graph 6 shows that as increasing in support threshold, decreasing in execution time. For support value 20 percent execution time is very less and for support value 5 percent execution time is very high.

3.3 Comparisons Based on Different Support and Required Memoryfor 10000 Transactions

Table 3.5 Support in percentage and required memory

Support threshold	Traditional Approach	Condensing approach	Transactions clustering approach
20	10546	8096	6422
15	16886	14264	10544
10	20834	17632	13682
5	24552	20872	15348

Table 4.4 shows different support value and required memory in KB for 10000 transactions. For 20 percent of support value traditional approach need 10546 KB memory, data condensing approach need 8096 KB memory and proposed approach need only 6422 KB memory. For 15 percentage of support value traditional approach need 16868 KB memory, data condensing approach need 14264 KB memory and proposed approach need only 10544 KB memory.



Graph 7 Comparison graph using different support and required memory time For 10 percentage of support value traditional approach need 20834 KB memory, data condensing approach need 17632 KB memory and proposed approach need only 13682 KB memory. For 5 percentage of support value traditional approach need

24652 KB memory, data condensing approach need 20872 KB memory s and proposed approach need only 15348 milliseconds of time. From the figure 7. it is clear that the memory requirement of the proposed approach is much smaller as compared to both traditional and condensing approach.

4. CONCLUSION

Traditional approach generates huge number of candidates and needs more database scan. The main drawback of this approach is joining operations which produce large number of applicants which is a time consuming process. Data condensing approach is based on merging of identical transactions and maximal set are used to generate required patterns. This approach is deleted all those one items which has support less than given threshold value, but it may be possible that after inserting some new transactions infrequent one item sets become frequent. Second problem of condensing approach is maintaining record of resulting intersections. Proposed approach not only reduces candidate's generations but also reduces time and memory to generate required patterns. Separate clustered dataset is maintained for patterns generation. The proposed approach handles both static and dynamic dataset.

REFERENCES

[1] Agirre-Basurko, E., Ibarra-Berastegi, G., and Madariaga, I. (2020), 'Regression and multilayer perceptron-based models to forecast hourly O3 and NO2 levels in the Bilbao area', Environmental Modelling and Software 21(4), 430-446.

[2] Aishwarya, D. C., and Babu, C. N. (2021), Prediction of Time Series Data Using GA-BPNN Based Hybrid ANN Model, 'IEEE 7th International Advance Computing Conference (IACC)' pp. 848-853.

[3] Al-Alawi, S. M., Abdul-Wahab, S. A., and Bakheit, C. S. (2021), 'Combining principal component regression and artificial neural networks for more accurate predictions of ground-level ozone', Environmental Modelling and Software 23(4), 396-403.

[4] Antanasijević, D. Z., Pocajt, V. V., Povrenović, D. S., Ristić, M. Đ., and Perić-Grujić, A. A., (2021), 'PM10 emission forecasting using artificial neural networks and genetic algorithm input variable optimization', Science of the Total Environment 443, 511-519.

[5] Arora, J. S., Snyman J.A. (2019), 'Practical Mathematical Optimization: An introduction to basic optimization theory and classical and new gradient-based algorithms', Structural and Multidisciplinary Optimization 31(3), 249-249.

[6] Ashish, M., and Rashmi, B. (2018), 'Prediction of daily air pollution using wavelet decomposition and adaptive-network-based fuzzy inference system', International Journal of Environmental Sciences 2(1), 185.

[7] Athanasiadis, I. N., Kaburlasos, V. G., Mitkas, P. A., and Petridis, V. (2023), Applying machine learning techniques on air quality data for real-time decision support, 'First international NAISO symposium on information technologies in environmental engineering (ITEE'2003)'.

[8] Bai, L., Wang, J., Ma, X., and Lu, H. (2018), 'Air pollution forecasts: An overview', International journal of environmental research and public health 15(4), 780.

[9] Balakrishnan, K., Dey, S., Gupta, T., Dhaliwal, R. S., Brauer, M., Cohen, A. J. and Sabde, Y. (2019), 'The impact of air pollution on deaths, disease burden, and life expectancy across the states of India: the Global Burden of Disease Study 2017'. The Lancet Planetary Health 3(1), e26-e39.

[10] Bellinger, C., Jabbar, M. S. M., Zaïane, O., & Osornio-Vargas, A. (2017). A systematic review of data mining and machine learning for air pollution epidemiology. BMC public health, 17(1), 907.

[11] Bělohlávek, R., Klir, G. J., Lewis, H. W., and Way, E. (2012), 'On the capability of fuzzy set theory to represent concepts', International Journal of General Systems 31(6), 569-585.

[12] Bottou, L. (1998), 'Online learning and stochastic approximations', On-line learning in neural networks 17(9), 142.

[13] Brunelli, U., Piazza, V., Pignato, L., Sorbello, F., and Vitabile, S. (2007), 'Two-day ahead prediction of daily maximum concentrations of SO2, O3, PM10, NO2, CO in the urban area of Palermo', Atmospheric Environment 41(14), 2967-2995.

[14] Burnett, R., Chen, H., Szyszkowicz, M., Fann, N., Hubbell, B., Pope, C. A., and Coggins, J. (2018), 'Global estimates of mortality associated with long-term exposure to outdoor fine particulate matter', Proceedings of the National Academy of Sciences 115(38), 9592-9597.

[15] Byrd, R. H., Chin, G. M., Nocedal, J., and Wu, Y. (2012), 'Sample size selection in optimization methods for machine learning', Mathematical programming 134(1), 127-155.

[16] Cai, S., Wang, Y., Zhao, B., Wang, S., Chang, X., and Hao, J. (2017), 'The impact of the "air pollution prevention and control action plan" on PM2. 5 concentrations in Jing-Jin-Ji region during 2012–2020', Science of the Total Environment 580, 197-209.

[17] Cascio, W. E., and Long, T. C. (2018), 'Ambient Air Quality and Cardiovascular Health Translation of Environmental Research for Public Health and Clinical Care', North Carolina medical journal 79(5), 306-312.

[18] Chaloulakou, A., Saisana, M., and Spyrellis, N. (2003), 'Comparative assessment of neural networks and regression models for forecasting summertime ozone in Athens', Science of the Total Environment 313(1-3), 1-13.

[19] Chang, K. W., and Roth, D. (2011), Selective block minimization for faster convergence of limited memory large-scale linear models, 'Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining', pp. 699-707.

[20] Chong, E. K., and Zak, S. H. (2013), 'An introduction to optimization', Vol. 76 John Wiley and Sons.

[21] Cobourn, W. G. (2010), 'An enhanced PM 2.5 air quality forecast model based on nonlinear regression and back-trajectory concentrations', Atmospheric Environment 44(25), 3015-3023.