JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 2898 - 2905
https://publishoa.com
ISSN: 1309-3452

An Innovative Approach for Association Rule Mining In Grocery Dataset Based On Non-Negative Matrix Factorization And Autoencoder

Dr. Harvendra Kumar Patel, Prof. (Dr.) K. P. Yadav

harvendra.patel82@gmail.com, drkpyadav732@gmail.com
Department of Computer Science and Engineering
Mats University, Chhattisgarh, India

ABSTRACT

Data mining with Deep Neural Networks becomes a very promising research area. In the course of our daily lives, enormous amounts of transactional data are produced. As a result, the transactional dataset is growing exponentially. The handling of data is a major issue. To handle transactional data, tools and strategies are therefore required. The tool that accelerates the decision-making process and enables unique knowledge management of the information found in transaction data is data mining. Many writers provided their methods to handle transactional data and discover new patterns. This study aims to provide a new algorithm for data mining that performs currently used algorithms in terms of speed and interest. The effectiveness of the DAENMF-ARM algorithm will be demonstrated by comparing it with the most popular and widely used data mining algorithms, namely Apriori ECLAT and FP-Growth, and establishing association rules. The organization of the paper is: the 1st portion is the introduction, the 2nd portion is the literature work, the 3rd portion is the methodology, 4th portion is the experimental setup, and last portion is conclusion of the paper.

Keywords: Data Mining, Association Rule Mining, Deep Neural Networks, Auto Encoder.

1. Introduction

Data mining (DM) is recognized for extracting information from huge amounts of data. DM can be defined as the process of extracting knowledge from huge database. Many tasks can be accomplished with the received data such as scientific exploration in the areas of market analysis, fraud detection, customer retention, and production control [1, 2]. The amount of data available today grows leaves and bounds. It has been discovered in various formats and has been compiled from numerous sources. Hence the processing of data is challenging and uncovering the hidden information is challenging. There are numerous methods and tools available to aid with data mining. It is one of the newest and fastest-growing machine learning areas and it has a lot of potential for analyzing crucial data in databases and data warehouses. Data mining, commonly referred to as KDD (Knowledge Discovery from Databases), is a technique used to identify patterns in massive or highly dimensional datasets using a variety of techniques. These techniques give us valuable knowledge or trends that we can apply to our business. Two steps make up the association mining algorithm, the first step is to find frequent itemset by using different algorithms and the second step is to generate rules. A Deep Neural Network i.e. autoencoder has been used to process the data and frequent items have been extracted with the help of non-negative matrix factorization.

An artificial neural network having several layers between the input and output levels is referred to as a deep neural network [3]. It has several layers that are concealed. Auto-Encoder (AE) is a type of deep neural network that is an unsupervised learning method that uses a back propagation algorithm. The AE is typically used to process a complex high-dimensional data. Using weights and mapping techniques, the input x is processed to produce the lower dimensional output Y. Then, using the weights and mapping procedures, we transform y into an output x' whose dimensions match those of the initial input x. The AE is divided into three sections: encoder, code, and decoder.
Encoder: - The encoder condenses the inputs into a small amount of space. It can be represented through the function (encoding function) $h = f(x)$.
Decoder: - The goal of the decoder is to reconstruct the input from low-dimensional space. This can be represented through a function (decoding function) $r = g(h)$.

2. RELATED WORK

Agarwal et al. originally presented association mining in 1993. Various strategies for association rule mining have been presented in recent years by scientists and academics. The AIS algorithm (Agrawal, Imielinski, and Swami) [1993] was the first algorithm for mining association rules. This algorithm's biggest flaw is that it requires too many operations over the entire database, which slows it down and requires more storage. Agrawal and Ramakrishan was proposed Apriori algorithm in 1994. Algorithm uses a “breadth-first search” strategy. Han et al., 2004) [4] was proposed FP Growth in 2004. The fundamental concept underlying FP Growth is to first use an FP-tree structure to compress the database, and then it uses the “divide-and-conquer” technique to decompress the database for rules mining.

Execution times and measures of interestingness for various fuzzy ARM algorithms were compared by Tasnia Rahman et al. in 2019 [5]. The fuzzy Apriori method, the fuzzy Apriori, and the genetic fuzzy Apriori DC algorithm were used in the major comparison between two separate transactional datasets. The most popular and widely utilized ARM algorithms, Apriori and Eclat, were compared by Vlad Robu and Vitor Duarte dos Santos [6] (2019). These algorithms are employed to extract frequent and interesting patterns. A SCADA frequent itemsets mining method was proposed by Liya Ma et al. in [7] (2020). The Apriori algorithm and the FP-Growth algorithm were compared in a study by Islamiyah et al., [8] (2020), to analyze sales data and identify patterns.
Non-negative Matrix Factorization was first presented by Paatero and Tapper in 1994, and after papers by Lee and Seung in 1999, it gained widespread attention [9]. The primary goal of NMF, according to Xiao et al. [10], is to split the data matrix into two low-ranked factor matrices with non-negativity restrictions. The NMF and LNMF (Linear Nonnegative Matrix Factorization) may be applied to the facial database to recognize the fundamental facial expressions of people, according to Buciu and Pitas [11].

3. METHODOLOGY

To find combinations of items more likely to appear together in the transactional dataset, ARM is frequently used for market basket analysis. Finding fascinating patterns, connections, frequent item groups, correlations, or even just haphazard data structures in datasets is helpful. A rule of association might be thought of as $X \rightarrow Y$, where $X \, and \, Y$ are disjoint itemsets (set of items). The interestingness of association rules is denoted by mainly its support, confidence, lift, and conviction. Where support of rule $X \rightarrow Y$ is the percentage of transactions that contain both $X \, and \, Y$, mathematically support $(X \rightarrow Y)$ is equal to $\sigma(X \rightarrow Y)$. Some of the well-known algorithms such as Apriori, FP growth, and our optimized algorithm (DAENMF-ARM (denoising autoencoder and non-negative matrix factorization based on association rule mining)) using DNN (Deep Neural Network) are considered for the experiments of analyzing the performance of mining. A deep neural network is a special type of artificial neural network which have a more hidden layer. The DAENMF-ARM model has two components. One is denoising autoencoder and the second is non-negative matrix factorization. A denoising autoencoder (DAE) is a feed-forward artificial neural network and it used an unsupervised learning algorithm. The deep network is broadly accepted in many fields like data mining, finance, medicine, engineering, etc. Mainly two measures (support and confidence) are needed for finding a strong relationship among the itemsets [12, 13, 14]. A supermarket transactional data containing ten transactions $\{T_1, T_2, T_3, \ldots\ldots, T_{10}\}$ and eight items $\{A_1, A_2, A_3, \ldots\ldots A_8\}$. These transactions are $\{\{T_1: A_1, A_2, A_3\}, \{T_2: A_1, A_2, A_4\}, \{T_3: A_1, A_3, A_5\}, \{T_4: A_2, A_4, A_6\}, \{T_5: A_3, A_6, A_7\}, \{T_6: A_1, A_2, A_3, A_4\}, \{T_7: A_2, A_3, A_4\}, \{T_8: A_3, A_7\}, \{T_9: A_2, A_3, A_4, A_8\}, \{T_{10}: A_6, A_7\}\}$.

$$supp(A_2,\, A_4 \longrightarrow A_3) = 3/10 = 0.3$$
$$supp(A_2,\, A_4) = 5/10 = 0.5$$
$$conf\ (A_2,\, A_4 \longrightarrow A_3) = 0.3/0.5 = 0.6$$

In this case, conf$(A2, A4 \longrightarrow A3)$ is 0.6, which is bigger than minconf, association rule XY may be justified as a strong rule. But we need to check it more thoroughly than only support and confidence can in order to find a robust rule. In the association rule $(A2, A4 \longrightarrow A3)$ above, the support of the consequent $(s(A3) = 7/10 = 0.7)$ is higher than the rules of confidence (0.6). This is impractical. Consequently, this rule might be misleading. Irrelevant datasets can produce misleading rules. Therefore, further measures are required to prevent deceptive rules. Therefore, lift and conviction are two more measures that can be utilised to address the issue of misleading rules.

3.1 Lift: - The Lift is an ARM metrics used to gauge how closely X and Y are reliant on one another. The formula lift(X⟶Y) = lift(Y⟶X) does not affect the measure. A lift might be defined as:

$$lift\ X \longrightarrow Y = \frac{conf(X \longrightarrow Y)}{supp(Y)} = \frac{supp(XUY)}{\mathrm{supp}(X) * supp(Y)}$$

So,

$$lift(A_2,\, A_4 \longrightarrow A_3) = \frac{0.3}{0.5 * 0.7} = 0.35 < 1$$

Lift is the ratio of observed support and projected support under the assumption that X and Y are free for each other. It can take one of three values:
Probability of occurrence and result are independent of one another if lift = 1. The itemsets are interdependent if lift >1. If lift < l, indicates that one item replaces another, then this indicates that one thing negatively affects another. As lift(X⟶Y) <1 exists in this situation, the rule "$(A_2, A_4 \longrightarrow A_3)$" is invalid and cannot be taken as a strong fact.

3.2 Conviction: - it is an ARM metrics that aims to assess the severity of a rule's execution through evaluation [15]. Contrary to Lift, Conviction is considerate of rule direction, as in $(conv\ (X \longrightarrow Y) \neq conv\ X \longrightarrow Y)$.

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 2898 - 2905
https://publishoa.com
ISSN: 1309-3452

A high conviction value indicates that the outcome is heavily dependent on the prior. The conviction may be expressed as follows:

$$conv(X \longrightarrow Y) = \frac{1-supp(Y)}{1-conf(X \longrightarrow Y)} = (1 - supp\ (Y))/(1 - conf(X \longrightarrow Y))$$

They are more likely to believe that X will happen without Y if they are depending on the genuine regularity of its existence. It functions similarly to lift in that circumstance.
So,

$$conv(A_2, A_4 \longrightarrow A_3) = \frac{1 - 0.7}{1 - 0.6} = 0.75$$

$$conv(A_3 \longrightarrow A_2, A_4) = \frac{1 - 0.5}{1 - 0.42} = 0.86$$

Here, the value of $conv(A_2,\ A_4 \longrightarrow A_3)$ is less than the value of $conv(A_3 \longrightarrow A_2,\ A_4)$, therefore the $conv(A_3 \longrightarrow A_2,\ A_4)$ can be recognized as a strong rule.

Table 1: Quality indicators with range

Name	Equation	Feasible values
Support	P_{MN}	[0,1]
Confidence	$\frac{P_{MN}}{P_M}$	[0,1]
Lift	$\frac{P_{MN}}{P_M * P_N}$	[0,1]
Conviction	$\frac{P_M * P_{\bar{N}}}{P_{M\bar{N}}}$	$\left[\frac{1}{n}, \frac{n}{4}\right]$

A. Proposed model

A deep neural network (DNN) is a model for processing information that was inspired by the biological nervous system. Its simulation of the brain mimics its most fundamental functions. Large numbers of interconnected nodes or neurons make up deep neural networks. The ability of DNNs to remember, learn, and generalize the training patterns to those of the human brain allows for the identification of their collective behavior. Back-propagation is a technique employed by the artificial neural network system (ANS) to handle text categorization issues. It makes sense to use it for text categorization. For issues that cannot be resolved sequentially or by sequential algorithms, ANS offers a superior solution. Image matching is one of the artificial neural networks' other uses [16]. They lead to improved performance. Among the other algorithms provided by ANS, the back-propagation algorithm is frequently employed.

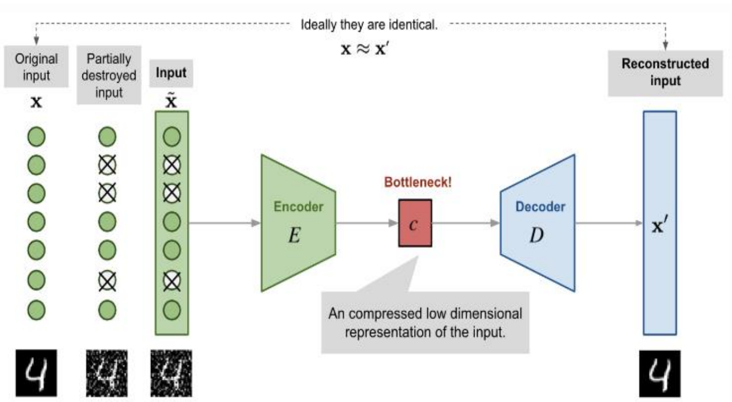

Fig 1: Architecture of Denoising Autoencoder

The denoising autoencoder model learns to approximately reconstruct its input. While doing so, it learns a salient representation of the data. Hence, it is also used for dimensionality reduction purposes. The model consists of two components, an encoder, and a decoder. The encoder maps the input $Rn\ --> \ Rd$ and the decoder reproduces the input from the reduced dimensions i.e. mapping $Rd\ --> \ Rn$. The noise in the input is usually introduced in the training phase (e.g. by means of Dropout) to avoid learning an identity function; hence get the name of denoising autoencoder. In our case, during model training, a noisy or corrupted basket with some items missing will be used as input. At test time, the one-hot encoded basket items will be fed into the model to get item predictions. Then from the output, we will have to

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 2898 - 2905
https://publishoa.com
ISSN: 1309-3452

select items with probabilities greater than some threshold values ($if\ p\ >=\ 0.1\ then\ 1\ else\ 0$), to convert back into a one-hot encoded vector. Finally, items with one in the binary vector can be recommended (frequent itemset) to the user.

Our denoising autoencoder model has 4 layers in the encoder and 4 layers in the decoder. The soft sign activation function has been used in the first three layers of the encoder and decoder, while the sigmoid activation function has been used in the fourth of the encoder and decoder. Binary cross entropy has been used as a loss function to minimize error using a variant of the stochastic gradient descent method, commonly known as Adam. The architecture is illustrated in figure 1, it can be seen that the input dimensionality is 169 i.e. number of unique items in the dataset. Likewise, the number of neurons in the first, second, third, and fourth layers of the encoder are 128, 64, 32 and, 16 respectively and the number of neurons in the first, second, third and fourth layers of decoder are 16, 32, 64 and 128 respectively. The dropout is introduced at the input layer with the probability of 0.6 i.e. dropping 40% of the inputs randomly. Moreover, the weights of the encoder are regularized using l2-regularization with a rate of 0.00001. Backpropagation is a powerful tool for identifying complex patterns and carrying out challenging mapping tasks. The neural network's processing components are neurons. The lines that connect the neurons are called weights.

4. EXPERIMENTAL SETUP

An Intel(R) Core(TM) i3-8130U CPU running at 2.20GHz and 4.00 GB of RAM or more have been used for this experiment. Microsoft Windows 10 Enterprise served as the operating system. No other application was run throughout the comparison of the three algorithms' performance (Apriori, FP Growth, and DAENMF-ARM). This allowed for a feasible examination of the run time and the interestingness. This experiment uses tensorflow version 1.13.1 and the Python IDE environment with Python version 3.6. Numerous interdisciplinary domains, including statistics, machine learning, information retrieval, etc., have contributed to the field of data mining. All libraries can be combined in Python with just a few lines of code.
Both the supermarket dataset and the grocery transactional dataset have been used in the experiment. The data are collected from https://www.kaggle.com/datasets and https://data.world/datasets/world. The following information describes the parameters of the chosen datasets.

TABLE 2: Unique items in the dataset

	Grocery Dataset_1	Grocery Dataset_2	Grocery Dataset_3
No. of unique items	120	169	271
No. of transaction	7501	9835	17336

A. EXPERIMENTAL RESULT

Run time analysis and interest analysis are the two parts of the experiments section aimed at examining the performance of different algorithms which are to be discussed. Lesser execution time means better performance of the algorithm.

B. Data Visualization: - Visualization of three datasets (Grocery dataset_1, Grocery dataset_2, Grocery dataset_3) in four rows. Dataset_1 has columns and data_2 and dataset_3 each have 32 columns.

Grocery dataset_1

```
             1          2           3  ...                18       19         20
0       shrimp    almonds     avocado  ...   frozen smoothie  spinach  olive oil
1      burgers  meatballs        eggs  ...              None     None       None
2      chutney       None        None  ...              None     None       None
3       turkey    avocado        None  ...              None     None       None
4  mineral water     milk  energy bar  ...              None     None       None

[5 rows x 20 columns]
```

Grocery dataset_2

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 2898 - 2905
https://publishoa.com
ISSN: 1309-3452

```
                  1                  2                3  ...    30    31    32
0       citrus fruit  semi-finished bread        margarine  ...  None  None  None
1     tropical fruit               yogurt           coffee  ...  None  None  None
2         whole milk                 None             None  ...  None  None  None
3          pip fruit               yogurt     cream cheese  ...  None  None  None
4   other vegetables           whole milk  condensed milk  ...  None  None  None

[5 rows x 32 columns]
```

Grocery dataset_3

```
                  1                  2                3  ...   30   31   32
0       citrus fruit  semi-finished bread        margarine  ...  NaN  NaN  NaN
1     tropical fruit               yogurt           coffee  ...  NaN  NaN  NaN
2         whole milk                  NaN              NaN  ...  NaN  NaN  NaN
3          pip fruit               yogurt     cream cheese  ...  NaN  NaN  NaN
4   other vegetables           whole milk  condensed milk  ...  NaN  NaN  NaN

[5 rows x 32 columns]
```

C. **Rules Generation: -** Rules for three datasets (Grocery dataset_1, Grocery dataset_2, Grocery dataset_3) in four rows. Dataset_1 has columns and data_2 and dataset_3 each have 32 columns.

Rules for grocery dataset_1

	item1	item2	Support	Confidence	Lift	Conviction
0	['chocolate', 'frozen vegetables']	['shrimp']	0.005333	0.232558	3.254512	1.296082
1	['ground beef', 'cooking oil']	['spaghetti']	0.004799	0.571429	3.281995	2.322135
2	['ground beef', 'eggs']	['herb & pepper']	0.004133	0.206667	4.178455	1.255295
3	['spaghetti', 'frozen vegetables']	['ground beef']	0.008666	0.311005	3.165328	1.438812
4	['milk', 'frozen vegetables']	['olive oil']	0.004799	0.203390	3.088314	1.249294
5	['tomatoes', 'milk']	['frozen vegetables']	0.004133	0.295238	3.097316	1.413055

Rules for grocery dataset_2

	item1	item2	Support	Confidence	Lift	Conviction
0	['other vegetables', 'beef']	['pork']	0.004169	0.211340	3.665839	1.262688
1	['other vegetables', 'beef']	['root vegetables']	0.007931	0.402062	3.688692	1.659150
2	['beef', 'tropical fruit']	['other vegetables']	0.004474	0.586667	3.031985	2.408531
3	['rolls/buns', 'beef']	['root vegetables']	0.004982	0.365672	3.354833	1.568616
4	['beef', 'whole milk']	['root vegetables']	0.008033	0.377990	3.467851	1.594778
5	['beef', 'yogurt']	['root vegetables']	0.004575	0.391304	3.589998	1.635340

Rules for grocery dataset_3

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 2898 - 2905
https://publishoa.com
ISSN: 1309-3452

	item1	item2	Support	Confidence	Lift	Conviction
0	['other vegetables', 'beef']	['pork']	0.004169	0.211340	3.665839	1.262688
1	['other vegetables', 'beef']	['root vegetables']	0.007931	0.402062	3.688692	1.659150
2	['beef', 'tropical fruit']	['other vegetables']	0.004474	0.586667	3.031985	2.408531
3	['rolls/buns', 'beef']	['root vegetables']	0.004982	0.365672	3.354833	1.568616
4	['beef', 'whole milk']	['root vegetables']	0.008033	0.377990	3.467851	1.594778
5	['beef', 'yogurt']	['root vegetables']	0.004575	0.391304	3.589998	1.635340

D. Runtime Analysis

This section compares the performance (run-time) of a given three algorithms using different support values. Shorter execution time means better performance of the algorithm. The following tables show the run-time estimation of the dataset.

TABLE 3: Run time (in seconds) analysis of the different algorithms.

Dataset	Min Support Value	Apriori Algorithm	FP Growth Algorithm	DAENMF-ARM
Grocery Dataset_1	0.002	90.03	57.85	43.6
	0.004	67.82	49.28	37.43
	0.006	60.79	45.41	35.3
	0.008	57.43	42.6	33.5
	0.01	54.52	43.86	31.15
Grocery Dataset_2	0.002	112.18	78.4	60
	0.004	90.98	70.63	54.39
	0.006	85.7	61.8	49.2
	0.008	76.24	55.73	42.7
	0.01	72.45	52.31	41.15
Grocery Dataset_3	0.002	130.06	89.42	72.04
	0.004	102.41	79.63	63.09
	0.006	91.92	70.08	56.24
	0.008	83.08	64.53	51.07
	0.01	76.87	58.96	49.18

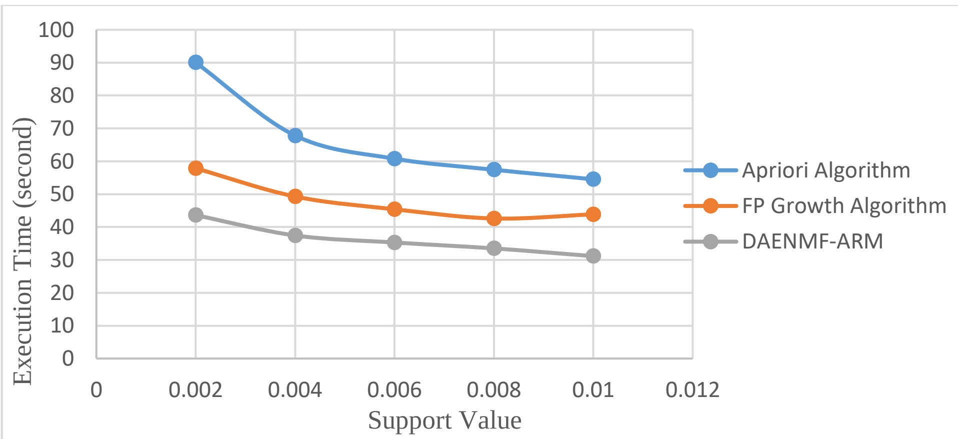

Fig. 2: Run-time evaluation for various methods on the grocery dataset_1

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 2898 - 2905
https://publishoa.com
ISSN: 1309-3452

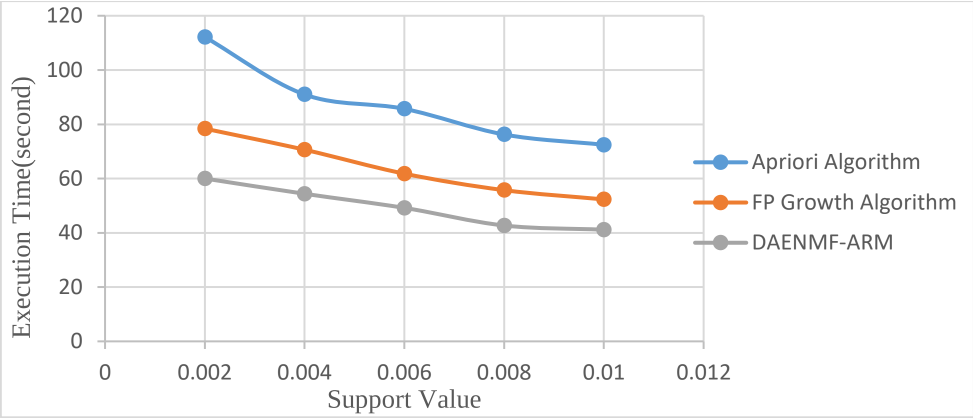

Fig. 3: Run-time evaluation for various methods on the grocery dataset_2

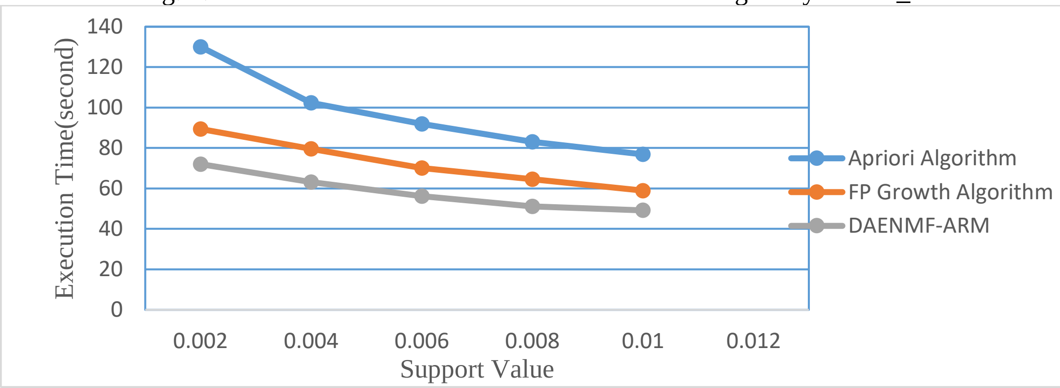

Fig. 4: Run-time evaluation for various methods on the grocery dataset_3

We observed from Fig-2, Fig-3 and Fig-4 that Apriori has higher runtime than the FP Growth algorithm while the FP Growth algorithm has higher runtime than the DAENMF-ARM algorithm. As the Apriori algorithm scans the whole dataset for each iteration while the FP Growth algorithm scan only twice and the DAENMF-ARM algorithm only once.

5. CONCLUSION

With the enormous amount of data stored in files, databases, and other repositories, it is extremely important to develop a powerful analytical system/tool. Maybe the analysis of such data to draw out relevant information that might be useful in making decisions. DM is the automated technique of extracting predicted, hidden information from huge databases. Extraction, transformation, and loading of transaction data onto the data warehouse system are all included in data mining. Although neural networks have been effectively employed in a variety of supervised and unsupervised learning applications. This approaches are not only used for data mining tasks due to their complex structure, lengthy training period, and challenging (tedious) representation of findings. Models created by neural networks frequently are not understandable. However, because of their great accuracy, neural networks are widely accepted for noisy data and are therefore chosen in data mining. The paper focuses on the data mining techniques, association rule mining, and the different key measures: like support, confidence, lift and conviction to find strong association rules in a large transactional dataset. To achieve the goal there is a complete comparison between the two most popular data mining algorithms that is Apriori and FP Growth with a new algorithm developed by us that uses the concept of ANNs. In the paper, there is related work, methodology, and results analysis. The observation is employed to investigate how artificial neural networks are used in data mining methods. Research is also done on the essential tools and techniques for data mining based on neural networks. Given the current state of the art, artificial neural networks should be included in data mining professionals' toolkits because they perform better than Apriori and FP-Growth, two well-known data mining algorithms, in terms of run-time, efficiency, and interestingness.

REFERENCES

1. Dr. Yongjian Fu "Data Mining: Tasks, Techniques, and Applications", department of computer science, University of Missouri, pp 10-11, Sep 2015.
2. Halil İbrahim Dede et al., "Comparison of Frequent Pattern Mining Algorithms in Internet of Things", 28th Signal Processing and Communications Applications Conference (SIU), pp.1057-1060, 2021.

3. Harvendra Kumar, Dr. Rakesh Kumar Yadav, “A Novel Approach for Association Mining Based on Matrix Factorization and Deep Neural Network”, Journal of Advanced Research in Dynamical & Control Systems, Vol. 11, 04-Special Issue, 25 March 2019.
4. Han, J., J. Pei, Y. Yin and R. Mao, “Mining frequent patterns without candidate generation: A frequent-pattern tree approach”, Data Mining Knowledge Discovery, 2004, 8: 53-87.
5. Tasnia Rahman et al., "Performance Evaluation of Fuzzy Association Rule Mining Algorithms", 4th International Conference on Electrical Information and Communication Technology (EICT), pp.272-275, 2019.
6. Vlad Robu and Vitor Duarte dos Santos, "Mining Frequent Patterns in Data Using Apriori and Eclat", 6th International Conference on Systems and Informatics (ICSAI), pp.1478-1481, 2019.
7. Liya Ma et al., "Frequent Itemsets Mining of SCADA Data based on FP-Growth Algorithm", 4th IEEE Conference on Energy Internet and Energy System Integration, pp.3057-3061,2020.
8. Islamiyah et al., "Comparison of Apriori and FP-Growth Algorithms in Determining Association Rules", International Conference on Electrical, Electronics and Information Engineering (ICEEIE), vol.4, pp.320-323, 2020.
9. Lee and Seung, “Algorithms for Non-negative Matrix Factorization”, 13th International conference on neural information processing systems, Pages 535–541, January 2016.
10. Yu-Xiong Wang and et al., “Non- negative Matrix Factorization: a Comprehensive review”, IEEE transactions on knowledge and data engineering, pp 1336-1353, June 2017.
11. Bucio and Pitas, “Application of non-negative and local non negative matrix factorization to facial expression recognition”, Proceedings of the 17th International Conference on Pattern Recognition, pp. 1-4, 2016.
12. P.Naresh and Dr.R.Suguna, "Association Rule Mining Algorithms on Large and Small Datasets: A Comparative Study", Proceedings of the International Conference on Intelligent Computing and Control Systems (ICICCS), pp. 587-592, 2019.
13. H. K. Soni et al., “Frequent Pattern Generation Algorithms for Association Rule Mining: Strength and Challenges”, IEEE International Conference on Electrical, Electronics and Optimization Techniques (ICEEOT), pp. 3744-3747, 2016.
14. Aditya Shukla, Dr. Harvendra Kumar, “A Comparative Study on Association Rule Mining and Its Preliminaries”, International Conference on Intelligent Systems (ICIS-2022), 11-12 March 2022.
15. Lee and Seung, “Algorithms for Non-negative Matrix Factorization”, 13th International conference on neural information processing systems, Pages 535–541, January 2020.
16. Hossein Riahi-Madvar et al., "Derivation of optimized equations for estimation of dispersion coefficient in natural streams using hybridized ANN with PSO and CSO algorithms", IEEE Access, vol.8, pp.156582-156599, 2020.