

HOSTED BY



ELSEVIER

Contents lists available at ScienceDirect

Journal of King Saud University –
Computer and Information Sciencesjournal homepage: www.sciencedirect.com

Multi-unit Discrete Hopfield Neural Network for higher order supervised learning through logic mining: Optimal performance design and attribute selection

Nur 'Afifah Rusdi^{a,b}, Mohd Shareduwan Mohd Kasihmuddin^{a,*}, Nurul Atiqah Romli^a, Gaethry Manoharam^a, Mohd. Asyraf Mansor^c^a School of Mathematical Sciences, Universiti Sains Malaysia, Penang 11800, Malaysia^b Institute of Engineering Mathematics, Universiti Malaysia Perlis, Kampus Pauh Puta, 02600 Arau, Perlis, Malaysia^c School of Distance Education, Universiti Sains Malaysia, Penang 11800, Malaysia

ARTICLE INFO

Article history:

Received 26 January 2023

Revised 3 April 2023

Accepted 5 April 2023

Available online 17 April 2023

Keywords:

Multi-unit Discrete Hopfield Neural Network

Higher order satisfiability

Logic mining

Supervised learning

Permutation

ABSTRACT

In the perspective of logic mining, the attribute selection, and the objective function of the best logic is the two main factors that identifies the effectiveness of our proposed logic mining model. The non-significant attributes selected will cause the Discrete Hopfield Neural Network to learned and obtain wrong synaptic weight. Thus, this will result to suboptimal solution. Although we might select the correct attributes, the conventional objective function of the best logic limits the search space to obtained more induced logic during the retrieval phase of Discrete Hopfield Neural Network. Therefore, this paper proposes a novel logic mining by integrating statistical analysis in the pre-processing phase to ensure that only optimal attributes will be selected. Supervised learning approach via correlation analysis is implemented for the purpose of attribute selection. Additionally, permutation operator serves to enhance the probability of the higher order satisfiability logical rule to be satisfied by having finite arrangement of attributes. During the learning phase, we proposed multi-unit Discrete Hopfield Neural Network to enhance the search space which leads to optimal solution. The efficiency of the proposed model is tested on 15 real-life datasets by comparing the performance of the model with existing works in logic mining using five performance metrics including accuracy, sensitivity, precision, Matthews Correlation Coefficient (MCC) and F1 Score. According to the results, the proposed model has its own strength by dominating most of the average rank of the performance metrics. This demonstrates that the proposed model can differentiate across all domains in the confusion matrix. Additionally, the p -value obtained based on the five-performance metrics indicate that there is a significantly difference between the proposed model and all existing works since the value obtained for accuracy (0.000), sensitivity (0.001), precision (0.000), F1 score (0.000) and MCC (0.000) are less than 0.05. This finding statistically prove that the proposed model is more effective compared with existing works in logic mining.

© 2023 The Author(s). Published by Elsevier B.V. on behalf of King Saud University. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The goal of data mining is to extract meaningful information from massive amounts of data and organise those findings in a

comprehensible structure (Yu, et al., 2022). Unfortunately, as the number of data increase, it requires involuntary way for these data to be extracted when necessary. As a result, numerous data mining techniques are discussed in the literature to address the aforementioned problem. However, according to Purusothaman and Krishnakumari (2015), classification grab the attention of most researchers since it is more beneficial data mining technique which organize data in a collection into specific groups or classes. Classification is utilised to extract models describing significant data classes in order for the proposed data mining model to be capable of offering potential solutions to the problems. As been mentioned by Mistry and Shah (2016) in their work, classification techniques are proposed based on difference cases which include Artificial

* Corresponding author.

E-mail address: shareduwan@usm.my (M.S.M. Kasihmuddin).

Peer review under responsibility of King Saud University.



Production and hosting by Elsevier

<https://doi.org/10.1016/j.jksuci.2023.101554>

1319-1578/© 2023 The Author(s). Published by Elsevier B.V. on behalf of King Saud University.

This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

Neural Network (ANN), k- Nearest Neighbor (KNN), Regression Tree (RT) and Support Vector Machine (SVM). However, even while most approaches have achieved great success in terms of accuracy and dependability, the classification decision for most of them is remain questionable, making it impossible for them to comprehend the reasoning behind the classification.

A variant of Artificial Neural Network (ANN), known as Discrete Hopfield Neural Network (DHNN) has been introduced to tackle this issue. Discrete Hopfield Neural Network was coined by Hopfield and Tank in 1985 in handling traveling salesman problem (TSP) and a linear programming (LP) problem (Wen, et al., 2009). The main highlight of DHNN is the present of content associative memory (CAM) whereby it is used to store the pattern of the problem and it consists of interconnected neurons with no hidden layer which act as the potential characteristics for solving optimization problems (Guo, et al., 2022). Due to this speciality, DHNN has widely been applied to solve various problems (Wen, et al., 2009), (Khoshahval and Fadaei, 2012), (Xu, et al., 2022), (Sani and Shermeh, 2022), and (Ji, et al., 2023). However, the DHNN model still been labelled as a “black box” model because of the limitation in interpreting the output of the network to the user. Therefore, incorporating the idea of satisfiability throughout the learning and retrieval phase of DHNN allows us to better understand its behaviour (Zamri, et al., 2020). This idea was proposed by Abdullah (1992) whereby satisfiability is embedded in DHNN so that the output of the network can be interpreted by the user. In layman’s terms, satisfiability representation (SAT) is a logical rule to represent the output of DHNN. According to Sidik et al. (2022), satisfiability can be represented as systematic and non-systematic logical rule whereby in systematic logical rule, the number of variables in each clause is restricted to only k variables but non-systematic logical rule considers non-restrictive number of variables in each clause.

Systematic logical rule in DHNN is among the pioneer works that capitalize satisfiability concept in DHNN. Kasihmuddin, et al., (2017) proposed 2 satisfiability logical rule (2SAT) whereby the number of variables in each clause is restricted to two. In this study, 2SAT is embedded in DHNN and effective learning method is implemented to minimize the cost function that is associated with the final minimum energy. As a result, the proposed model demonstrates the efficacy of 2SAT to represent the clause since the network can sustain more neurons and reach a global minima ratio above 90%. In other development, Mansor, et al., (2017) proposed another systematic logical rule which is 3SAT. This study focusses on maximizing the capabilities of 3SAT logic by incorporating with an artificial immune system algorithm in DHNN. The idea of implementing 3SAT instead of 2SAT is because of probabilities for the clause to be satisfied is higher if the number of variables per clause is greater. The finding of the study validates the ability of 3SAT logic in representing the clause when the proposed hybrid model can reduce the complexity of the network. The experimental results show that the proposed model able to achieve global minimum ratio which is close to 1 and reduce the computational time as well as compared to existing method. In other works, Kasihmuddin, et al., (2018) proposed Maximum k -Satisfiability (MAX- k SAT). The main goal of this study is to identify the optimal assignment that can fulfil the most clauses with $k = 2$ and $k = 3$. The performance of MAX-2SAT and MAX-3SAT with Hopfield Neural Network will be compare with Kernel Hopfield network, KHNN-MAX k SAT. The experimental results indicates that the proposed MAX k SAT can be regarded as robust since the global minima ratio obtained is $Z_m = 0.9720$ approaching 1 meaning that nearly all neurons attained the correct final state during the retrieval phase. In addition to that, the propose HNN-MAX k SAT can obtained more than 80% global minimum solution.

On the other side, the first study considering non-systematic logical rule was proposed by Sathasivam, et al., (2020). Random k Satisfiability (RANKSAT) with Hopfield Neural Network utilized the random number of variables per clause to obtain the optimal final state neurons in the retrieval phase. The logical structure of RAN2SAT proposed in this study is not restricted compared to 2SAT and 3SAT whereby it consists of first order and second order logical structure. This flexibility of the logical structure contributes to ensuring the logical rule is satisfied. The result of this study encourages Karim, et al., (2021) to investigate the performance of higher order random k satisfiability by considering $k = 3$. The finding of this study demonstrates that the proposed DHNN-RAN3SAT capable in obtaining more neuron variations throughout the entire solution space and the variety of logical combinations leads to new perspective on the behaviour of non-systematic logical rules in achieving the objective functions in each phase of DHNN. Even though many studies in the literatures report the effectiveness of satisfiability concept in DHNN, the proposed model can be considered as a good model if it can perform well in the perspective of logic mining. Hence, this inspired the researchers to perform logic mining in extracting the behaviour of the dataset by using the proposed logical structure.

Logic mining involves constructing a logical rule that can describe the association between the input features and the output variable, particularly in classification tasks. The feature set used in model building is the only information source for any learning algorithm, making it crucial to identify an optimal subset that is the representative of the original dataset (Mwadulo, 2016). Thus, selecting the appropriate input features can help improve data comprehension, reduce computational demands, overcome the curse of dimensionality, and ultimately enhance prediction accuracy. In general, feature selection methods can be classified into several types including filter methods, wrapper methods, embedded methods and hybrid methods. Various number of studies in the current literature have demonstrated the impact of feature selection methods on classification performance. For instance, Labani, et al., (2018) employed the filter method approach to tackle a text classification problem. Their research aimed to introduce a feature selection technique that not only eliminates irrelevant features but also handles redundant ones. Other than that, Gao, et al., (2021) proposed a feature selection algorithm for biomedical applications that integrates inter-feature correlations. Their approach improved upon the randomly restarted incremental feature selection (RIFS) algorithm by integrating two-dimensional information from the feature subset selected by RIFS, creating the RIFS2D algorithm. This approach was proposed based on the observations that a subset of low-ranked features could potentially achieve excellent prediction performance. The proposed feature selection method was evaluated using numbers of binary classification datasets with three feature selection approach including filter, wrapper and hybrid methods. The results demonstrate RIFS2D outperformed existing feature selection algorithm on most of the investigated datasets. Other than that, Mahendran and PM (2022) proposed an embedded classification approach in which they used four feature selection methods - Ada Boost, Random forest, LASSO, and SVM - and select the best method for their proposed classification model specifically in the detection of Alzheimer’s disease. This approach is an example of embedded feature selection, where the feature selection process is integrated into the model building process.

Other recent developments in feature selection methods include using evolutionary techniques such as Genetic Algorithm (GA), Ant Colony Optimization (ACO), Grey Wolf Optimization (GWO), and others. These techniques have garnered attention from researchers due to their ability to conduct a global search. For example, Piri and

Mohapatra (2021) examined feature selection method using a wrapper classifier to determine the most optimal feature subsets for medical datasets, specifically for medical records. Due to the broad search area and non-significant features, it is challenging to identify the critical factors resulting in lower classification performance. Thus, the study proposed Multi-Objective Quadratic Binary Harris Hawk Optimization (MOQBHHO) algorithm to categorize medical records based on their characteristics. In addition to the Harris Hawk Optimization method, there are other proposed feature selection algorithms in the literature. Wang, et al., (2022) proposed a Multi-Objective Genetic Algorithm for feature selection in pulmonary hypertension, while Kaur, et al., (2022) explored the use of the Quantum Whale Optimization algorithm to address the issue of depression detection. On another note, Zhao, et al., (2022) recently introduced an online feature selection method to tackle clinical treatment challenges encountered by cancer patients' post-operation. While their method demonstrated good performance, it does have certain limitations, such as its suitability for only small sample size data and the need for considerable computational time to identify important features. Additionally, the proposed method also requires a high level of human intervention in handling the proposed method.

In the context of logic mining, the feature selection method, also known as attribute selection method, can be classified into unsupervised and supervised learning approaches. This method improves the performance of the classification task by selecting only the optimal attributes, which leads to optimal induced logic. The unsupervised attribute selection method identifies the most relevant and informative attributes based on the intrinsic data structure. In other words, unsupervised learning does not require prior knowledge to select optimal attributes. Several studies in the literature have focused on the unsupervised learning approach for attribute selection, including the pioneering work inspired by the work of Sathasivam and Wan Abdullah, 2011, which specifically focuses on random attribute selection. They proposed a Reverse Analysis (RA) method, which is based on neural logic integration to extract logical rules that represent the performance of students in each subject and thus describing the data sets. However, not much effort has been done to find the optimal induced logic to represent the datasets. Then, Kho et al., 2020a proposed 2 Satisfiability Reverse Analysis method (2SATRA). This model incorporating the idea of Hopfield Neural Network, 2 Satisfiability and Reverse Analysis method. In this study, the best logical rule of how League of Legends (LoL) games has been conducted to classify either win or lose is obtained by using 2SATRA. This proposed model emphasizes on 2SAT logical rule to represent the clause and the attributes are selected randomly. Besides that, Alway, et al., (2020) also maximize the use of 2 Satisfiability Reverse Analysis method (2SATRA) to determine the price trend for palm oil in Malaysia. In other studies, conducted by Jamaludin et al., 2022b, Artificial Bee Colony algorithm has been integrated in 2SATRA to optimize the learning phase of Discrete Hopfield Neural Network in solving the defaulting issue in credit card. The induced logic obtained during the retrieval phase will classifies the default payment status and amount paid in the given month. However, limitations found in all these studies is the arrangement of the attributes on the structure of 2SAT logical rule. The standard arrangement of the attributes leads to low accuracy values since the connectivity among the attributes are limited and the randomized attribute selection will reduce the interpretability of the induced logic. This encouraged Jamaludin, et al., (2023) to propose logical permutation operator into 2SATRA namely, P2SATRA. The logical permutation operator implemented in 2SATRA allow finite arrangement of attributes which able to boost the probability of 2SATRA to increase the accuracy of the induced logic obtained in the retrieval phase of DHNN. Nevertheless, the effectiveness of all these unsu-

perervised learning methods is still subject to question, as the interpretability issue of the induced logic remains unresolved.

On the other sides, supervised learning approaches in the current literature concentrate on utilizing statistical analysis to select the optimal attributes. Kasihmuddin, et al., (2022) proposed supervised learning approach on 2SATRA, namely S2SATRA. The main goal of this study is to encounter the interpretability issues in data mining whereby the output is poorly understood. Therefore, instead of randomized attribute selection, correlation analysis has been conducted during the pre-processing phase as a benchmark in choosing which attributes will be selected to represent the 2SAT logical rule. The finding of the study shows that the classification performance by S2SATRA outperforms all existing works in logic mining. This supervised approach on attribute selection only allows the significant attribute to be selected. Thus, this reduces the task of permutation operator in obtaining the optimal induced logic. In other development, Jamaludin et al., 2022a proposed another statistical analysis which can be implemented to solve the issue of randomized attribute selection. In this study, log linear analysis has been conducted to determine which attributes will be selected to represent the 2SAT logic before it can be learned by the network. By using log linear analysis, only attributes with significance level less than α will be selected. This will ensure that only optimal attributes will be trained resulting in correct synaptic weight acquired during the learning phase.

Unfortunately, to our best knowledge, there is no attempt work has been proposed to discover the optimal choice of attributes to represent the higher order logical rule. The research conducted by Zamri, et al., (2020) only concentrates on the randomized attribute selection and standard attributes arrangement to represent the 3SAT logical rule in the clause which cannot represent the real-life data set especially when involving large number of attributes. The randomized attribute selection will have the tendency of not choosing the significant attributes which in turn prevent the final state neurons from obtaining the optimal induced logic in the retrieval phase of DHNN. This reveals the need of empowering the pre-processing phase before it can be learned by DHNN. Hence, the contributions of this paper are listed as follows:

- (a) To propose supervised learning for higher order logic during pre-processing phase of Discrete Hopfield Neural Network. The correlation analysis will be conducted during the pre-processing phase to filter the attributes. Thus, this guarantee that only optimal attributes will be selected to find the best logic structure.
- (b) To develop new objective function of the best logic. In this context, the true positive and true negative outcomes will be considered. The proposed objective function will be learned in Discrete Hopfield Neural Network to obtain the synaptic weight by comparing the cost function and energy function.
- (c) To propose multi-unit Discrete Hopfield Neural Network by introducing more than one best logic. The proposed model will utilize ten best logical structures so that more induced logic will be produced during the retrieval phase. Thus, this will increase the probability to retrieve optimal induced logic.
- (d) To implement logical permutation operator for higher order logic in Discrete Hopfield Neural Network. Considering permutation operator will enhance the search space in finding the optimal final neuron state. This logical permutation will provide finite arrangement of attributes which create flexibility to the induced logic obtained.
- (e) To propose new logic mining model namely Supervised Higher Order Reverse Analysis method in doing various real-life datasets. This logic mining highlights the implementation of supervised learning, permutation operator

and multi-unit Discrete Hopfield Neural Network to ensure the induced logic obtained is in optimal state. Therefore, the proposed logic mining could extract the information from the datasets and the induced logic will have the capabilities to classify.

The paper has the following structure. Section 2 explains the motivation for this study followed by describing satisfiability representation in Section 3. Satisfiability in Discrete Hopfield Neural Network will be detailed in section 4. In Section 5, we discuss on the proposed model and Section 6 focuses on the experimental set up for this study. Lastly, Section 7 is the illustrated results and the concluding remarks in Section 8 complete the paper.

2. Motivation

2.1. Optimal attribute selection

Pre-processing phase in logic mining is vital to ensure that Discrete Hopfield Neural Network (DHNN) can attained optimal solution while maintaining the unique characteristics of the data. One of the main issues is does the attributes are chosen appropriately and does the selected attributes are optimal? Note that having a correct and optimal attribute is important because it leads to optimal induced logic. Random attribute selection has widely been applied by the researchers to select the attributes that will represent the clause. For instance, the work of Kho, et al., (2020a, 2020b) and Jamaludin et al. (2020) capitalizes the random attribute selection to represent the clause of 2SAT logic. Even if the induced logic produced by the proposed 2SATRA and E2SATRA is thought to be ideal, there are still a greater probability that unimportant attributes are selected, and this random selection indirectly makes the induced logic uninterpretable. Not only that, having an optimal attribute selection is also crucial when dealing with large number of attributes in a dataset. Random selection method might discard the important attributes since there is no benchmark or guideline in choosing the attributes. The wrong attributes chosen will lead the DHNN to learn wrong synaptic weight and end up with suboptimal induced logic. To cater this issue, supervised learning is introduced by Kasihmuddin, et al., (2022). In this study, correlation test is applied in pre-processing phase in choosing the attributes. Results shows that S2SATRA outperform other existing works in logic mining. The proposed S2SATRA achieve accuracy more than 80% and there are few datasets with $Acc = 1$ which means the proposed model can predict true positive (TP) and true negative (TN) correctly. In addition, S2SATRA also has winning points for other performance metric such as sensitivity, precision, F1 score and MCC when compared with other existing logic mining models. However, up to our knowledge, 3SATRA only concentrates on the randomized attributes selection (Zamri, et al., 2020). This urge the study to proposed supervised attribute selection method to ensure that the attribute which will be selected is in optimal condition. By doing so we can also know the performance of higher order logic when using supervised learning approach as compared to second order logic structure. Hence, in this study, we incorporate the correlation analysis proposed by Kasihmuddin, et al., (2022) into our work which expected to generate the best induced logic that can be used for the purpose of classification.

2.2. Permutation of the logical structure

Despite of the success of 3SATRA in obtaining a good performance, one of the limitations addressed in this existing work is the limited connectivity of the attribute among the 3SAT clause

(Zamri, et al., 2020). The predetermined attribute in each clause will cause the minimal interaction among attributes in different clause. Thus, the induced logic obtained in the final state will lose their flexibility. In addition to that, the low connectivity will cause the 3SATRA to face with overfitted issue due to the other possible connection was disregarded. To address this issue, we capitalize the work of Jamaludin, et al., (2023) whereby the permutation operator will be implemented which would improve the accuracy of the induced logic. Permutation is defined as the rearrangement of the attributes to represent the clause for the proposed logic. By altering the logical formula's attribute and increasing the neuron permutation, we guarantee that the proposed model will achieve higher accuracy as compared to the conventional 3SATRA.

2.3. Lack of optimal objective function of the best logic, Q_{best}

During the learning phase of DHNN, the objective function, Q_{best} represent our proposed logical structure. Most of the study in literatures including Alway, et al., (2020), Kho et al., 2020a, 2020b and Alzaeemi and Sathasivam, (2020) focus on $Q_i^{learn} = 1$ in finding Q_{best} which only consider the true positive outcomes. However, real-world datasets often have imbalanced distributions, where one or more classes may be underrepresented or overrepresented, resulting in an uneven distribution of classes. This can lead to biased performance or an ineffective model in classification, as minority classes may be overlooked due to having less data to learn from. Patel et al. (2020) explain how imbalanced datasets can be challenging, and Ramyachitra & Manikandan, (2014) describe characteristics of imbalanced datasets, such as small disjuncts, noisy data, dataset shift, and lack of density, which can exacerbate these challenges and affect classification performance. These factors can make it difficult to train and evaluate machine learning models on imbalanced datasets, emphasizing the importance of considering them when working with such data. In our context of problems, we may have imbalanced data whereby the true negative outcome is dominant compared to true positive. In this case, finding Q_{best} by using $Q_i^{learn} = 1$ is not appropriate since it will lead to bias solutions. To solve this issue, this study proposed new Q_i^{learn} by considering both true positive and true negative outcomes such that $Q_i^{learn} = (1, -1)$. Other than that, it also been reported that the previous studies only consider single Q_{best} and this will just lead to limited induced logic obtained. This indirectly has constrained the ability of the proposed model to find the optimal induced logic. Based on this perspective, this study explores the effectiveness of using multiunit Discrete Hopfield Neural Network (Mu -DHNN) so that we can obtain more final neurons state which indirectly will produce more optimal induced logic.

3. Higher order satisfiability representation

Generally, satisfiability (SAT) refers to the task of identifying the truth assignments that makes the logical rule satisfiable. In this paper, higher order satisfiability (3SAT) will be used as a symbolic form of the problem. The generalized form of 3SAT logical rule can be explained as follows (Zamri, et al., 2020; Zhu, et al., 2022):

- A set of logical rule comprises of n attributes q_1, q_2, \dots, q_n and for each clause $n = 3$
- Represented in bipolar form $\{-1, 1\}$ that exemplify false and true respectively.
- A set of literals that can be either positive literal or negative literal such that $q_i \in \{q_i, \neg q_i\}$.
- A set of x definite clauses, M_1, M_2, \dots, M_x where all clauses are connected to logical AND (\wedge) and literals in each clauses are connected to logical OR (\vee).

By considering all the above properties, the generalized 3SAT can be represented as in Eq. (1)

$$Q_{3SAT} = \bigwedge_{i=1}^x M_i \quad (1)$$

where M_i represents the list of clauses with three attributes each

$$M_i = \bigvee_{i=1}^n (P_i, R_i, Z_i) \quad (2)$$

Based on Eqs. (1) and (2), a simple example of Q_{3SAT} can be written as

$$Q_{3SAT} = (A \vee B \vee \neg C) \wedge (\neg D \vee E \vee \neg F) \wedge (G \vee H \vee I) \quad (3)$$

such that $M_1 = (A \vee B \vee \neg C)$, $M_2 = (\neg D \vee E \vee \neg F)$ and $M_3 = (G \vee H \vee I)$. Note that, in this study we do not consider redundant attributes and M_1 , M_2 and M_3 must satisfied the interpretation so that it will be evaluated as true or else it will become false. For instance, based on Eq. (3), if $(A \vee B \vee \neg C)$ is read as $(-1, -1, -1)$, then Q_{3SAT} will evaluate as false.

4. Higher order satisfiability in Discrete Hopfield Neural Network

Discrete Hopfield Neural Network (DHNN) has a finite set of neurons S_i , $1 < i < N$ whereby each neuron is defined in bipolar state, $S \in \{1, -1\}$. General formulation of DHNN can be written as:

$$S_i = \begin{cases} 1, & \text{if } \sum_j^N W_{ij} S_j \geq \theta \\ -1, & \text{otherwise} \end{cases} \quad (4)$$

W_{ij} and θ in Eq. (4) represent the synaptic weight and predefined threshold value respectively. The predefined threshold value is set as a condition of DHNN to excite the neuron state to be either 1 or -1 . According to Sathasivam (2010), $\theta = 0$ is used to ensure the energy of DHNN decrease uniformly. W_{ij} can be represented in a matrix form. Since DHNN do not consider self-connection, the diagonal value of the synaptic weight will be zero, $W_{ii} = W_{jj} = 0$ and the connection is symmetrical such that $W_{ij} = W_{ji}$. In this paper, 3SAT logical rule, Q_{3SAT} will be embedded into DHNN as a symbolic rule that can regulate the outcomes of the network. During the learning phase of DHNN, the main agenda is to find the value of W_{ij} that will be stored in CAM. The cost function of Q_{3SAT} can be written as:

$$E_{Q_{3SAT}} = \sum_{i=1}^{NC} \prod_{j=1}^3 Z_{ij} \quad (5)$$

where NC is the number of clause in $E_{Q_{3SAT}}$ and Z_{ij} can be defined as in Eq. (6) where $\neg X$ is the negation of the literal in 3SAT logical rule.

$$Z_{ij} = \begin{cases} \frac{1}{2}(1 - S_X), & \text{if } \neg X \\ \frac{1}{2}(1 + S_X), & \text{otherwise} \end{cases} \quad (6)$$

The cost function, $E_{Q_{3SAT}}$ will be compared with Lyapunov energy function, $H_{Q_{3SAT}}$ as in Eq. (7) to find the value of synaptic weight.

$$H_{Q_{3SAT}} = -\frac{1}{3} \sum_i^N \sum_j^N \sum_k^N W_{ijk}^{(3)} S_i S_j S_k - \frac{1}{2} \sum_i^N \sum_j^N W_{ij}^{(2)} S_i S_j - \sum_i^N W_i S_i \quad (7)$$

During the retrieval phase, the focus is to find the final state neuron that will produce the optimal induced logic. By using the synaptic weight that is stored in CAM, the local field of Q_{3SAT} is updated as follows:

$$h_i = \sum_j^N W_{ijk}^{(3)} S_j S_k + \sum_j^N W_{ij}^{(2)} S_j + W_i \quad (8)$$

The local field in Eq. (8) will determine the final state neuron of DHNN. Then, Hyperbolic Tangent Activation Function (HTAF), $\tanh(h_i)$ as in Eq. (9) will be used to squeeze the final neuron state to be either 1 or -1 before we can obtain the induced logic. If $\tanh(h_i) \geq 0$ it will become 1 or otherwise it will be -1 .

$$\tanh(h_i) = \frac{e^{h_i} - e^{-h_i}}{e^{h_i} + e^{-h_i}} \quad (9)$$

Lastly, to ensure the DHNN-3SAT achieve global minimum energy, the following condition must be satisfied:

$$|H_{Q_{3SAT}} - H_{Q_{3SAT}}^{\min}| \leq Tol \quad (10)$$

Referring to Hopfield and Tank, (1985), the symmetric connection and the nonnegative diagonal elements of the synaptic weight of DHNN cause the network to always converge to a stable state such that the energy cannot be reduced any further. The global minimum energy produced will ensure that the network achieve optimal induced logic so that the prediction of the outcomes will be identical as Q_i^{test} meaning that the network able to obtain correct true positive and true negative outcomes. Based on the implementation of satisfiability concept in Discrete Hopfield Neural Network, it will act as an intelligent learning system for solving data mining problems specifically on classification problem through logic mining method.

5. Proposed supervised higher order Reverse analysis method

3-satisfiability reverse analysis (3SATRA) is a logic mining technique that utilize the higher order logical rule with DHNN to extract the information from the dataset. In addition, the value of true positive (TP), true negative (TN), false positive (FP) and false negative (FN) of the induced logic produced in the retrieval phase will determine the effectiveness of the proposed logic mining method. This section explained the proposed logic mining model namely Supervised Higher order Reverse Analysis Method (SHoRA). The implementation of the proposed logic mining model can be divided into 3 main phases including pre-processing phase, learning phase and retrieval phase. There are several works to be considered in the pre-processing phase. Firstly, let N be the number of neurons which represent the attributes for each data sets $(S_1, S_2, S_3, \dots, S_N)$. By considering Wan Abdullah method, the neurons will be represented in bipolar form such that $S_i \in \{-1, 1\}$ (Abdullah, 1992). Hence, we will transform all the dataset in bipolar form.

Next, instead of randomly choose the attributes, we will employ the statistical test namely correlation analysis test to select the attributes that will be used in the study. By taking into account the learning logic, Q_i^{learn} and neuron S_i and since this study emphasize on the 3SAT logical rule, the arrangement of each clause, M_i must consists of three attributes (S_i, S_j, S_k) where $i \neq j \neq k$. The neuron selection S_i , respected to each clause M_i , can be written as follows:

$$Q_{3SAT} = \bigwedge_{i=0, i \neq j \neq k}^{NC} \left(S_i^{\min|p_i|} \vee S_i^{\min|p_j|} \vee S_i^{\min|p_k|} \right), \quad i \neq j \neq k, \quad (11)$$

where $\min|p_i|$, $\min|p_j|$ and $\min|p_k|$ referred to the minimized value obtained from the correlation test when compared to pre-defined significant value, α . Therefore, the attribute will only be selected if it satisfies $0 \leq p_i \leq \alpha$, $0 \leq p_j \leq \alpha$, $0 \leq p_k \leq \alpha$. In this study, the significant value, α is set to be 0.05. As can be seen, we attempt to avoid utilising redundant attributes because doing so causes

the synaptic weight to be zero, which renders the classification model useless. We will select the 10 best attributes from the correlation test whereby 9 attributes will represent the input and the tenth attributes represent the outcome of the dataset.

After considering which attributes to be selected by using Eq. (11), Q_{3SAT}^{best} will be determined before it can be learned by DHNN to obtain the value of the synaptic weight. In the previous works in the literature (Zamri, et al., 2020), (Jamaludin et al., 2022a), (Kasihmuddin, et al., 2022) and (Jamaludin, et al., 2022b) segregate the collection of neurons per clause that leads to $Q_i^{learn} = 1$ is considered and the highest frequency will be considered as the best logic. However, this study proposes new method of finding Q_{3SAT}^{best} by considering the maximum summation of true positive and true negative which implies 1 and -1 respectively. In this context, we only consider the logical structure of Q_{3SAT} that leads to highest value of Q_{3SAT}^{learn} to be ours Q_{3SAT}^{best} . The formation of Q_{3SAT}^{best} can be obtained by using Eq. (12) as follows:

$$Q_{3SAT}^{best} = \max \left[\sum_{i=1}^n (\tau_i) \right], \quad Q_{3SAT}^{learn} = (1, -1) \quad (12)$$

Referring to Eq. (12), τ is the summation of true positive and true negative outcomes obtained from Q_i^{learn} . Let consider the following example to explain on how to obtain Q_{3SAT}^{best} . Let say we have two random logical structures obtained by using Eq. (11) as follows:

$$Q_{3SAT}^{learn} = (\neg A \vee B \vee \neg C) \wedge (D \vee \neg E \vee F) \wedge (\neg G \vee \neg H \vee I) \quad (13)$$

$$Q_{3SAT}^{learn} = (A \vee B \vee \neg C) \wedge (\neg D \vee \neg E \vee F) \wedge (G \vee \neg H \vee I) \quad (14)$$

If the sum of true positive and true negative outcomes for Eq. (13) is higher than Eq. (14), then Eq. (13) will be selected as Q_{best} . Next, we will define the inconsistency of Q_{3SAT}^{best} before it will be

learned to obtain the synaptic weight of DHNN. The synaptic weight can be found by comparing the cost function as in Eq. (5) with the energy function in Eq. (7) and the value will be store in CAM. During the retrieval phase, the final state neuron, S_i^A is obtained by using Eq. (15) as follows:

$$S_i^A = \begin{cases} 1, & \tanh(h_i) \geq 0 \\ -1_i, & \text{Otherwise} \end{cases}, \quad \tanh(h_i) = \frac{e^{h_i} - e^{-h_i}}{e^{h_i} + e^{-h_i}} \quad (15)$$

Finally, the final state neuron in Eq. (15) will be converted to induced logic, Q_i^A and will be compared with is Q_i^{test} such that $Q_i^A = Q_i^{test}$ in order to obtain the value of TP , TN , FP and FN . In addition, this paper proposed multi-unit Discrete Hopfield Neural Network (Mu -DHNN) so that we can enhance the search space in obtaining the optimal induced logic. Therefore, 10 Q_{best} will be formed in each neuron combination rather than single Q_{best} and the equation of Mu -DHNN for 10 Q_{best} is as in Eq. (16).

$$\mu Q_{3SAT}^{best} = \sum_{i=1}^{10} \max \left[\sum_{i=1}^n (\tau_i) \right], \quad Q_{3SAT}^{learn} = (1, -1) \quad (16)$$

Hence, each Q_{best} from $i = 1$ till $i = 10$ will repeat the same process in DHNN including obtaining the synaptic weight value, updating the local field and the final state neuron will be converted into induced logic. Fig. 1 demonstrates the schematic diagram on how Supervised Higher order Reverse Analysis method (SHoRA) is implemented in DHNN model. The left side of the diagram is the pre-processing phase whereby correlation analysis was done to select the attributes which will be trained by DHNN on the right side of the diagram. Meanwhile, the process in DHNN will be explained based on three phases including pre-processing, learning and retrieval phase. Fig. 2 illustrates the details of the flowchart for the proposed model.

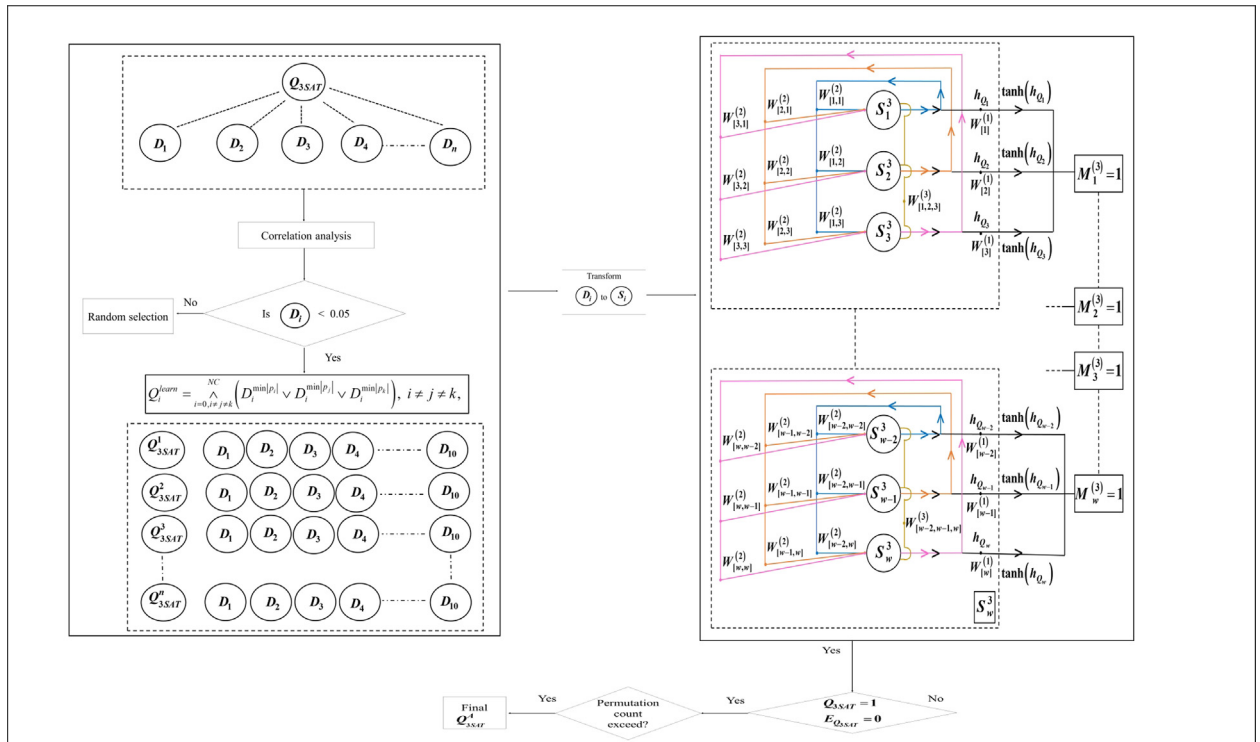


Fig. 1. Schematic diagram of SHoRA.

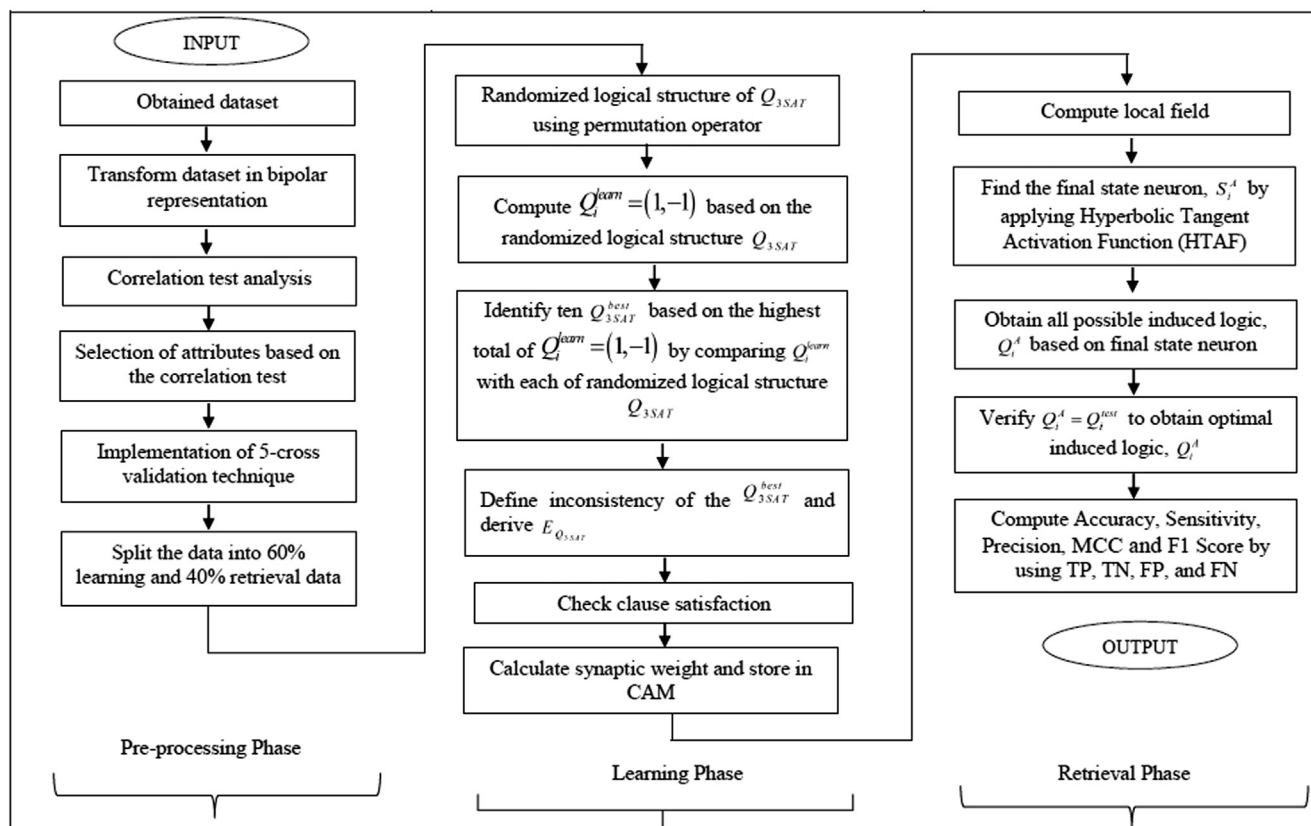


Fig. 2. The Implementation of Supervised Higher order Reverse Analysis method (SHoRA).

6. Experimental setup

We provide the following descriptions of the experiment's components and the set up that are used to assure the reproducibility of the experiment.

6.1. Benchmark Datasets

This 15 datasets from various field of study have been selected for this analysis which can be obtained from UCI repository <https://archive.ics.uci.edu/ml/index.php> and Kaggle machine learning repository <https://www.kaggle.com/datasets>. All these datasets have widely been used by the researchers to solve classification issue. For this experiment, two main criteria have been considered before the dataset can be chose. First, we only focus on dataset with more than 150 instances. According to [Ajiboye, et al., \(2015\)](#), the size of dataset can affect the accuracy and the generalization of the proposed model. The finding of the study shows that employing a substantial data set can improve the accuracy and generalizability of the predictive model since it exhibits greater accuracy as compared to small dataset. Other than that, we only prefer dataset greater than 150 instances to avoid overfitting issue whereby the model might perform well in training, but it has poor performance in testing data and to dodge with sampling bias concern. The second criterion that we have taken into consideration is the specification of the number of attributes to be used. This is important for various reasons. Firstly, including a large number of attributes can lead to overfitting, where the model becomes excessively complex and closely fits the training data. According to [Sugumaran and Ramachandran \(2011\)](#), selecting good attributes

becomes more challenging as the number of attributes increases. In other words, a high number of attributes can increase model complexity and make it difficult to extract meaningful patterns. Secondly, including too few attributes can lead to underfitting, where the model is not complex enough to capture the underlying relationships and patterns in the data, resulting in poor performance. Moreover, using too few attributes may not capture all relevant information in the data. However, there is no one optimal number of attributes that fits all problems. Therefore, this study specifically concentrates on number of attributes, $10 < \alpha < 40$. This is supported by previous studies by [Kasihmuddin, et al., \(2022\)](#) and [Jamaludin et al., 2022a](#) who also recommend the same range of attributes to be used in order to review the performance and the effectiveness of the proposed model in selecting the optimal attributes.

[Table 1](#) summarize the details of the datasets. All the selected datasets will be converted into bipolar representation, 1 and -1 by using Statistical Package for Social Sciences (SPSS) software. For quantitative data type, k-mean clustering will be utilized to convert the data into bipolar by finding the mean value for each attribute. Meanwhile, frequency distribution table is used as a benchmark to convert the qualitative data type into 1 and -1. Since majority of the selected datasets consist of missing value, frequency distribution table again is used to determine whether it will fall into 1 and -1. To ensure that it is comparable with all current logic mining methods, this study uses the train-split method, where 60% of the dataset is utilised as learning data and the remaining 40% is used as retrieval data. The use of k fold validation technique makes this experiment superior to all other logic mining models whereby five cross validation process is implemented in this experiment.

Table 1
List of Data Sets employed in the simulation.

| Code | Dataset | Attributes | Instances | Missing Value | Field | Outcome Q_{3SAT}^A |
|------|---|------------|-----------|---------------|------------|-----------------------------------|
| A1 | Horse Colic | 27 | 300 | Yes | Zoology | Treated with or without surgery |
| A2 | Credit Approval | 15 | 690 | Yes | Finance | Credit card approval |
| A3 | Absenteeism | 21 | 740 | No | Business | Absenteeism time in hour |
| A4 | Early-Stage Diabetes Risk Prediction | 17 | 520 | Yes | Medical | Early-Stage Diabetes |
| A5 | Chronic Kidney Disease | 24 | 400 | Yes | Medical | Chronic Kidney Disease |
| A6 | Single Proton Emission Computed Tomography (SPECT) images | 22 | 267 | No | Medical | Diagnosis |
| A7 | Congressional Voting Records | 16 | 435 | Yes | Social | Type of vote |
| A8 | Hepatitis | 19 | 155 | Yes | Medical | Class: Die or live |
| A9 | Autistic disorder for children | 21 | 292 | Yes | Medical | Autistic disorder |
| A10 | Automobile | 26 | 205 | Yes | Automotive | Price |
| A11 | Primary Tumour | 17 | 339 | Yes | Medical | Class: Cause of tumour |
| A12 | Facebook metrics | 19 | 500 | Yes | Social | Total interactions |
| A13 | Hungarian Chicken Pox | 20 | 521 | No | Medical | Weekly counts of chickenpox cases |
| A14 | Alcohol effect on math study | 33 | 395 | No | Life | Final grade |
| A15 | Soybean | 35 | 307 | Yes | Life | Class: Plant growth |

6.2. Performance metrics

According to Hossin and Sulaiman, (2015), measurement tool that is used to measure the performance of classifier is known as evaluation metrics. Numbers of evaluation metrics have been proposed in the literatures because of the ideology of one performance metric does not fix all problems. Therefore, in this experiment we will only focusses on one of the performance evaluation metrics known as confusion metric. Before diving into what the confusion metric is all about, it is important for us to understand the basic terms associated with confusion matrix.

Fig. 3 illustrates the confusion matrix that is used for classification. In general, there are only two classes to classify preferably a positive and negative class. True positive (TP) are the cases when the actual outcome is true and the predicted is also true meanwhile, true negative (TN) refer to the cases when the actual outcome is false and the predicted is also false. On the other side, false positive (FP) is defined when the actual outcome is false but the predicted is true. Lastly, false negative (FN) is referring to the case whereby the actual outcome is true, but the predicted outcome is false (Sokolova and Lapalme, 2009). For this experiment, the effectiveness and the efficiency of the proposed model can be evaluated by considering the most frequent used metrics which include Accuracy (Acc), Sensitivity (Se), Precision (Prec), Matthews Correlation Coefficient (MCC) and F1 Score (F₁).

As mentioned by Ayalew, et al., (2022), accuracy is defined as a proportion of true predictions made compared to the total number of predictions which in the range of [0, 100]. The accuracy values can be calculated by using Eq. (17) as follows:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{17}$$

While accuracy focus on true positive and true negative of the outcomes, sensitivity or known as recall in Eq. (18) used to measure how well a case correctly produce true positive outcomes which range from 0 to 1 (Valero-Carreras, et al., 2023).

$$Se = \frac{TP}{TP + FN} \tag{18}$$

Meanwhile, precision is the proportion of true positive outcomes to set of all actual positive values. The precision value is between 0 and 1 which can be measured as in Eq. (19) as follows (Sokolova and Lapalme, 2009):

$$Prec = \frac{TP}{TP + FP} \tag{19}$$

According to Sunil, et al., (2022), F1 score can be interpreted as a harmonic mean of the precision and sensitivity which is formulated as in Eq. (20). The value of F1 score is also between 0 and 1.

$$F1 = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} \tag{20}$$

Lastly, Matthews Correlation Coefficient (MCC) which originally developed by Matthews in 1975 has been utilized in this experiment that can be represented as in Eq. (21) (Chicco and Jurman, 2020). MCC is considered as the best metric among all the stated confusion matrix since it is based on the eight major derived ratio from the combination of all components in confusion matrix.

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{21}$$

6.3. Baseline methods

We restrict our comparison to only methods that produce induced logic since the goal of this experiment is to assess the performance of the induced logic obtained by Supervised Higher order Reverse Analysis method (SHoRA). The proposed SHoRA will be compared with all the existing logic mining models including RA (Sathasivam and Wan Abdullah, 2011), 2SATRA Kho et al., 2020a, P2SATRA (Jamaludin et al., 2023), E2SATRA (Jamaludin et al., 2020), L2SATRA (Jamaludin et al., 2021), A2SATRA (Jamaludin et al., 2022a), S2SATRA (Kasihmuddin, et al., 2022) and 3SATRA (Zamri, et al., 2020). To ensure the comparability of the results, some changes have been made. For RA, we assigned each neuron with attributes instead of assigning neuron for each instance as proposed by Sathasivam and Wan Abdullah, (2011). In addition,

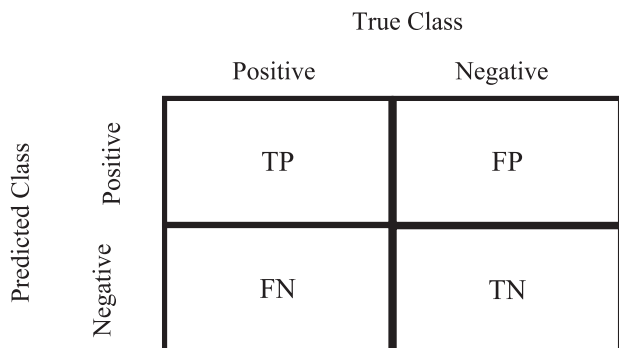


Fig. 3. Confusion Matrix for classification.

the RA will still maintain the two attributes per clause and the result will be recorded based on the induced logic that have the HornSAT property such that each clause consists of one positive and one negative literal.

As for 2SATRA, it follows the same as been proposed by Kho et al., 2020a whereby it utilize random attribute selection to find attributes to represent the 2SAT clause and utilize Wan Abdullah method in finding the synaptic weight during the learning phase of DHNN-2SAT. During the retrieval phase of DHNN, the network will retrieve all the final state neuron, S_i^A that leads to optimal induced logic by using Eqn 15. However, for 2SATRA method, there is no guarantee that S_i^A produced is global minimum or not since that has not been verified. The difference between the conventional 2SATRA and P2SATRA as proposed by Jamaludin, et al., (2023) is the implementation of the logical permutation operator to permutate the arrangement of the attributes in the 2SAT clause. This permutation operator will enhance the search space of finding the best combination of Q_{best} as it allows finite arrangement of attributes but at the same time, we avoid redundant permutation during the selection of attributes.

While P2SATRA highlight the permutation operator in the process of finding the optimal induced logic, E2SATRA on the other sides consider energy-based logic mining method whereby this method only considers the induced logic that produced global minimum energy. Instead of calculating the energy function as in Eq. (7), the global minimum energy for the induced logic can be verified by comparing the induced logic with Q_{best} . The induced logic is said to be in global state if the induced logic obtained in the retrieval phase is identical as Q_{best} or at least two clause is identical as in Q_{best} . Note that, P2SATRA and E2SATRA also utilize random attribute selection as in 2SATRA to select the attributes to represent the clause.

Jamaludin et al. (2021) and Jamaludin et al. (2022) use different approach on selecting the attributes such that log linear analysis has been conducted before deciding which attributes will be selected. By using log linear analysis, only significant attributes will be considered to represent the 2SAT clause. The rest of the process follows the conventional 2SATRA such that Q_{2SAT} integrated with Wan Abdullah method is employed in finding the synaptic weight and the retrieval property of L2SATRA and A2SATRA will have the property as 2SATRA. However, Jamaludin et al. (2022) also utilize the permutation operator as in P2SATRA to enhance the search space of finding the optimal induced logic. S2SATRA proposed by Kasihmuddin et al. (2022) introduced another attribute selection method such that correlation analysis has been conducted before the attributes are selected. In addition, S2SATRA employed the permutation operator as in P2SATRA and energy analysis as in E2SATRA. In this case, S2SATRA only consider induced logic that produce global minimum energy. Lastly, 3SATRA in this experiment will follows the conventional 3SATRA as been proposed by Zamri, et al., (2020). The parameter of the proposed

Table 2
List of parameters in SHoRA.

| Parameter | Parameter value |
|---------------------|---|
| No of attributes | 9 |
| No of clause | 3 |
| Qbest | $Q_i^{learn} = (1, -1)$ |
| Logical rule | Q_{3SAT} |
| Attribute selection | Correlation (Kasihmuddin, et al., 2022) |
| Tolerance value | 0.001 (Sathasivam, 2010) |
| No of trial | 10,000 |
| Neuron Combination | 100 |
| Maximum permutation | 100 |
| Activation function | Hyperbolic Tangent Activation Function (HTAF) |

Table 3
List of parameters in 2SATRA.

| Parameter | Parameter value |
|---------------------|---|
| No of attributes | 6 |
| No of clause | 3 |
| Qbest | $Q_i^{learn} = 1$ |
| Logical rule | Q_{2SAT} |
| Attribute selection | Random (Kho et al., 2020a) |
| Tolerance value | 0.001 (Sathasivam, 2010) |
| No of trial | 100 |
| Neuron Combination | 100 |
| Activation function | Hyperbolic Tangent Activation Function (HTAF) |

Table 4
List of parameters in P2SATRA.

| Parameter | Parameter value |
|---------------------|---|
| No of attributes | 6 |
| No of clause | 3 |
| Qbest | $Q_i^{learn} = 1$ |
| Logical rule | Q_{2SAT} |
| Attribute selection | Random (Kho et al., 2020a) |
| Tolerance value | 0.001 (Sathasivam, 2010) |
| No of trial | 100 |
| Neuron Combination | 100 |
| Maximum permutation | 100 |
| Activation function | Hyperbolic Tangent Activation Function (HTAF) |

Table 5
List of parameters in E2SATRA.

| Parameter | Parameter value |
|---------------------|---|
| No of attributes | 6 |
| No of clause | 3 |
| Qbest | $Q_i^{learn} = 1$ |
| Logical rule | Q_{2SAT} |
| Attribute selection | Random (Kho et al., 2020a) |
| Tolerance value | 0.001 (Sathasivam, 2010) |
| No of trial | 10,000 |
| Neuron Combination | 100 |
| Activation function | Hyperbolic Tangent Activation Function (HTAF) |

Table 6
List of parameters in S2SATRA.

| Parameter | Parameter value |
|---------------------|---|
| No of attributes | 6 |
| No of clause | 3 |
| Qbest | $Q_i^{learn} = 1$ |
| Logical rule | Q_{2SAT} |
| Attribute selection | Correlation (Kasihmuddin, et al., 2022) |
| Tolerance value | 0.001 (Sathasivam, 2010) |
| No of trial | 100 |
| Neuron Combination | 1000 |
| Maximum permutation | 100 |
| Activation function | Hyperbolic Tangent Activation Function (HTAF) |

Table 7
List of parameters in L2SATRA.

| Parameter | Parameter value |
|---------------------|---|
| No of attributes | 6 |
| No of clause | 3 |
| Qbest | $Q_i^{learn} = 1$ |
| Logical rule | Q_{2SAT} |
| Attribute selection | Log Linear (Jamaludin et al., 2021) |
| Tolerance value | 0.001 (Sathasivam, 2010) |
| No of trial | 100 |
| Neuron Combination | 100 |
| Activation function | Hyperbolic Tangent Activation Function (HTAF) |

Table 8
List of parameters in A2SATRA.

| Parameter | Parameter value |
|---------------------|---|
| No of attributes | 6 |
| No of clause | 3 |
| Qbest | $Q_i^{learn} = 1$ |
| Logical rule | Q_{2SAT} |
| Attribute selection | Log linear (Jamaludin et al., 2022a) |
| Tolerance value | 0.001 (Sathasivam, 2010) |
| No of trial | 100 |
| Neuron Combination | 1000 |
| Maximum permutation | 100 |
| Activation function | Hyperbolic Tangent Activation Function (HTAF) |

Table 9
List of parameters in RA.

| Parameter | Parameter value |
|---------------------|---|
| No of attributes | 6 |
| No of clause | 3 |
| Qbest | $Q_i^{learn} = 1$ |
| Logical rule | $Q_{Horn2SAT}$ |
| Attribute selection | Random (Kho et al., 2020a) |
| Tolerance value | 0.001 (Sathasivam, 2010) |
| No of trial | 10,000 |
| Neuron Combination | 100 |
| Activation function | Hyperbolic Tangent Activation Function (HTAF) |

Table 10
List of parameters in 3SATRA.

| Parameter | Parameter value |
|---------------------|---|
| No of attributes | 9 |
| No of clause | 3 |
| Qbest | $Q_i^{learn} = 1$ |
| Logical rule | Q_{3SAT} |
| Attribute selection | Random (Kho et al., 2020a) |
| Tolerance value | 0.001 (Sathasivam, 2010) |
| No of trial | 100 |
| Neuron Combination | 100 |
| Activation function | Hyperbolic Tangent Activation Function (HTAF) |

SHoRA model is shown in Table 2 and Table 3-10 illustrate the parameters for all baseline logic mining models.

7. Results and discussion

7.1. Correlation analysis for SHoRA

One of the major goals of this experiment is to evaluate the performance of higher order reverse analysis method (SHoRA) when pre-processing structure is applied during selection of attributes. In this experiment, the attributes are selected based on the supervised learning approach introduced by Kasihmuddin, et al., (2022) namely Pearson’s product moment coefficient of correlation. The correlation coefficient computed measure the relationship between the two attributes in each dataset. For clear illustration, the null hypothesis will assume as there is no significant correlation between the variable, $H_0 : \rho = 0$ and the alternative hypothesis will reflect as there is a significant correlation between the compared variables such that $H_1 : \rho \neq 0$. This analysis will verify which attributes, D_i will be selected to represent the clause, M_i for our proposed logic that leads to optimal attributes. Our assumption is to reject the null hypothesis, H_0 so we can conclude that two attributes are highly correlated. Therefore, the null hypothesis, H_0 will be rejected if the significance value of each attribute is less than $\alpha = 0.05$. In this phase, the non-correlated

attributes will be discarded before entering the learning phase of DHNN. The correlation analysis is necessary to dodge the SHoRA from choosing the wrong attributes. Table 11 and Table 12 demonstrate the first ten correlation values of the attribute D_i for A1 until A15 with respect to $Q_{3SAT}^{k_i}$.

By referring to all these tables, we can summarize that A6, A9 and A13 are considered as the best performing correlation datasets and achieve an optimal number of attributes since all the first ten attributes consists of significant value less than $\alpha = 0.05$. Therefore, we reject the null hypothesis, H_0 and we can conclude that the attributes are highly correlated. Note that, the lower the significance value, the higher the correlation value. The weak correlation can be identified if the significance value is greater than $\alpha = 0.05$. A11 is acknowledge as a weak dataset and the worst performing correlation value as compared to the rest of the dataset since only four out of ten attributes have the significant value which is less than 0.05. Worth mentioning that dataset A11 consists of 17 attributes. Thus, by using this correlation test, we can avoid from choosing the wrong attributes.

Despite the higher significance value, the proposed SHoRA is still able to avoid attributes with no correlation at all. The correlation test also works well for the rest of the datasets (A1-A5, A7, A8, A10, A12, A14 and A15) whereby more than half of the attributes able to obtain significant value less than 0.05. This shows that the correlation analysis able to distinguish between important and unimportant attributes meaning that we can know how well the attributes contributes to the output. Additionally, when dealing with numerous attributes, such as in the case of A14 and A15, which consist of 33 and 35 attributes respectively, supervised learning is essential to prevent the selection of insignificant attributes that would result in incorrect synaptic weight and ultimately result in suboptimal induced logic obtained in the retrieval phase of DHNN. Overall, based on the correlation analysis results, since all correlation values agree with the association threshold value, we can say that the proposed SHoRA does not require any random attribute selection.

7.2. Accuracy for all logic mining model

In the previous subsection, we already discuss on how the correlation test has been applied in this experiment. Therefore, only optimal attributes will be selected to represent the 2SAT and 3SAT clauses for this experiment. By considering the new way of finding Q_{best} and the implementation of five cross validation technique, the accuracy of the induced logic produced for the proposed SHoRA model will be compared with all the accuracy of the induced logic produced by the existing logic mining models. The accuracy of the SHoRA model was compared to all existing works in logic mining from Tables 13-20 whereby +/- referred to total number of wins, ties, and losses respectively as compared to SHoRA model. As shown in Table 13 till Table 20, the proposed model has several advantages including the proposed model surpasses all existing logic mining models in terms of accuracy when comparing with one-to-one existing logic mining models for all fifteen datasets. Worth mention that the proposed model outperforms two out of eight existing logic mining models which are 2SATRA and RA by having the highest accuracy for all fifteen datasets. It essential to know that, when compared with 3SATRA, L2SATRA and E2SATRA, the proposed model still achieve the greatest percentage of accuracy for 14 out of 15 datasets.

The S2SATRA model is the closest competitor to SHoRA and has achieved the highest accuracy in seven out of fifteen datasets. Both models utilize correlation analysis to select the most significant attributes to be learned by DHNN, indicating that correlation analysis can improve model accuracy.

Table 11
Correlation analysis for first ten attributes for A1-A9.

| | | D_1 | D_2 | D_3 | D_4 | D_5 | D_6 | D_7 | D_8 | D_9 | D_{10} |
|----|-------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|
| A1 | Significant value | 0.119 | 0.028 | 0.315 | 0.005 | 0.158 | 0.098 | 0.000 | 0.043 | 0.058 | 0.008 |
| | Decision H_0 | Fail to reject H_0 | Reject H_0 | Fail to reject H_0 | Reject H_0 | Fail to reject H_0 | Fail to reject H_0 | Reject H_0 | Reject H_0 | Fail to reject H_0 | Reject H_0 |
| A2 | Significant value | 0.716 | 0.004 | 0.000 | 0.000 | 0.000 | 0.042 | 0.160 | 0.000 | 0.000 | 0.000 |
| | Decision H_0 | Fail to reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Fail to reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 |
| A3 | Significant value | 0.000 | 0.000 | 0.565 | 0.005 | 0.754 | 0.000 | 0.782 | 0.039 | 0.002 | 0.008 |
| | Decision H_0 | Reject H_0 | Reject H_0 | Fail to reject H_0 | Reject H_0 | Fail to reject H_0 | Reject H_0 | Fail to reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 |
| A4 | Significant value | 0.064 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.012 | 0.000 | 0.761 |
| | Decision H_0 | Fail to reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Fail to reject H_0 |
| A5 | Significant value | 0.000 | 0.039 | 0.817 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.016 | 0.000 |
| | Decision H_0 | Reject H_0 | Reject H_0 | Fail to reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 |
| A6 | Significant value | 0.001 | 0.001 | 0.000 | 0.001 | 0.004 | 0.001 | 0.000 | 0.000 | 0.008 | 0.001 |
| | Decision H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 |
| A7 | Significant value | 0.000 | 0.951 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.091 |
| | Decision H_0 | Reject H_0 | Fail to reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Fail to reject H_0 |
| A8 | Significant value | 0.002 | 0.031 | 0.105 | 0.106 | 0.000 | 0.000 | 0.0124 | 0.715 | 0.185 | 0.006 |
| | Decision H_0 | Reject H_0 | Reject H_0 | Fail to reject H_0 | Fail to reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Fail to reject H_0 | Fail to reject H_0 | Reject H_0 |
| A9 | Significant value | 0.000 | 0.001 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |
| | Decision H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 |

Table 12
Correlation analysis for first ten attributes for A10-A15.

| | | D_1 | D_2 | D_3 | D_4 | D_5 | D_6 | D_7 | D_8 | D_9 | D_{10} |
|-----|-------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|
| A10 | Significant value | 0.605 | 0.275 | 0.734 | 0.101 | 0.001 | 0.296 | 0.033 | 0.000 | 0.026 | 0.000 |
| | Decision H_0 | Fail to reject H_0 | Fail to reject H_0 | Fail to reject H_0 | Fail to reject H_0 | Reject H_0 | Fail to reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 |
| A11 | Significant value | 0.109 | 0.115 | 0.005 | 0.000 | 0.973 | 0.256 | 0.012 | 0.095 | 0.043 | 0.280 |
| | Decision H_0 | Fail to reject H_0 | Fail to reject H_0 | Reject H_0 | Reject H_0 | Fail to reject H_0 | Fail to reject H_0 | Reject H_0 | Fail to reject H_0 | Reject H_0 | Fail to reject H_0 |
| A12 | Significant value | 0.352 | 0.359 | 0.002 | 0.023 | 0.811 | 0.427 | 0.260 | 0.000 | 0.000 | 0.000 |
| | Decision H_0 | Fail to reject H_0 | Fail to reject H_0 | Reject H_0 | Reject H_0 | Fail to reject H_0 | Fail to reject H_0 | Fail to reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 |
| A13 | Significant value | 0.011 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |
| | Decision H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 |
| A14 | Significant value | 0.006 | 0.021 | 0.038 | 0.011 | 0.338 | 0.000 | 0.298 | 0.676 | 0.386 | 0.016 |
| | Decision H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Fail to reject H_0 | Reject H_0 | Fail to reject H_0 | Fail to reject H_0 | Fail to reject H_0 | Reject H_0 |
| A15 | Significant value | 0.017 | 0.005 | 0.002 | 0.683 | 0.038 | 0.000 | 0.461 | 0.124 | 0.000 | 0.104 |
| | Decision H_0 | Reject H_0 | Reject H_0 | Reject H_0 | Fail to reject H_0 | Reject H_0 | Reject H_0 | Fail to reject H_0 | Fail to reject H_0 | Reject H_0 | Fail to reject H_0 |

By calculating the correlation coefficient, we can identify which attributes are strongly correlated with the output and, therefore, likely to be important. Attributes that have weak or no correlation with the output can be excluded, as they are unlikely to provide any useful information. Furthermore, correlation analysis can also identify any multicollinearity or redundancy among the attributes, which occurs when two or more attributes are highly correlated

with each other. In such cases, only one of the correlated attributes needs to be included in the model to avoid overfitting. Therefore, correlation analysis helps in selecting only the most significant attributes by providing a data-driven approach to identify which attributes are most informative and relevant for the learning process. This can improve the accuracy of the model and reduce the complexity of the model by focusing only on the relevant

Table 13
Accuracy SHoRA vs RA.

| Dataset | SHoRA | RA |
|---------|---------------|---------------|
| A1 | 65.83% | 47.50% |
| A2 | 84.28% | 44.20% |
| A3 | 66.96% | 48.45% |
| A4 | 85.38% | 61.92% |
| A5 | 92.63% | 50.88% |
| A6 | 75.51% | 70.28% |
| A7 | 87.59% | 42.07% |
| A8 | 86.45% | 53.87% |
| A9 | 78.97% | 48.55% |
| A10 | 89.27% | 40.00% |
| A11 | 66.76% | 61.91% |
| A12 | 89.00% | 46.80% |
| A13 | 77.22% | 37.61% |
| A14 | 71.14% | 54.43% |
| A15 | 57.07% | 53.33% |
| +/-/- | | 15/0/0 |

Table 16
Accuracy SHoRA vs E2SATRA.

| Dataset | SHoRA | E2SATRA |
|---------|---------------|---------------|
| A1 | 65.83% | 56.67% |
| A2 | 84.28% | 47.39% |
| A3 | 66.96% | 56.62% |
| A4 | 85.38% | 53.37% |
| A5 | 92.63% | 76.25% |
| A6 | 75.51% | 55.33% |
| A7 | 87.59% | 43.10% |
| A8 | 86.45% | 38.39% |
| A9 | 78.97% | 72.31% |
| A10 | 89.27% | 73.66% |
| A11 | 66.76% | 55.74% |
| A12 | 89.00% | 49.70% |
| A13 | 77.22% | 64.59% |
| A14 | 71.14% | 54.68% |
| A15 | 57.07% | 60.65% |
| +/-/- | | 14/0/1 |

Table 14
Accuracy SHoRA vs 2SATRA.

| Dataset | SHoRA | 2SATRA |
|---------|---------------|---------------|
| A1 | 65.83% | 45.33% |
| A2 | 84.28% | 51.88% |
| A3 | 66.96% | 50.20% |
| A4 | 85.38% | 57.40% |
| A5 | 92.63% | 45.38% |
| A6 | 75.51% | 61.87% |
| A7 | 87.59% | 44.02% |
| A8 | 86.45% | 38.71% |
| A9 | 78.97% | 74.70% |
| A10 | 89.27% | 77.56% |
| A11 | 66.76% | 61.91% |
| A12 | 89.00% | 46.7% |
| A13 | 77.22% | 65.17% |
| A14 | 71.14% | 54.68% |
| A15 | 57.07% | 62.28% |
| +/-/- | | 14/0/1 |

Table 17
Accuracy SHoRA vs L2SATRA.

| Dataset | SHoRA | L2SATRA |
|---------|---------------|---------------|
| A1 | 65.83% | 42.33% |
| A2 | 84.28% | 56.45% |
| A3 | 66.96% | 46.15% |
| A4 | 85.38% | 59.71% |
| A5 | 92.63% | 77.00% |
| A6 | 75.51% | 56.45% |
| A7 | 87.59% | 68.62% |
| A8 | 86.45% | 43.23% |
| A9 | 78.97% | 61.71% |
| A10 | 89.27% | 47.07% |
| A11 | 66.76% | 57.65% |
| A12 | 89.00% | 34.50% |
| A13 | 77.22% | 52.96% |
| A14 | 71.14% | 46.58% |
| A15 | 57.07% | 62.29% |
| +/-/- | | 14/0/1 |

Table 15
Accuracy SHoRA vs P2SATRA.

| Dataset | SHoRA | P2SATRA |
|---------|---------------|---------------|
| A1 | 65.83% | 60.00% |
| A2 | 84.28% | 84.49% |
| A3 | 66.96% | 57.09% |
| A4 | 85.38% | 77.79% |
| A5 | 92.63% | 98.00% |
| A6 | 75.51% | 75.89% |
| A7 | 87.59% | 77.82% |
| A8 | 86.45% | 82.90% |
| A9 | 78.97% | 75.38% |
| A10 | 89.27% | 87.32% |
| A11 | 66.76% | 67.65% |
| A12 | 89.00% | 97.00% |
| A13 | 77.22% | 75.12% |
| A14 | 71.14% | 58.86% |
| A15 | 57.07% | 70.73% |
| +/-/- | | 9/0/6 |

Table 18
Accuracy SHoRA vs A2SATRA.

| Dataset | SHoRA | A2SATRA |
|---------|---------------|---------------|
| A1 | 65.83% | 60.17% |
| A2 | 84.28% | 68.62% |
| A3 | 66.96% | 59.66% |
| A4 | 85.38% | 84.62% |
| A5 | 92.63% | 92.25% |
| A6 | 75.51% | 66.54% |
| A7 | 87.59% | 86.90% |
| A8 | 86.45% | 82.58% |
| A9 | 78.97% | 76.21% |
| A10 | 89.27% | 67.07% |
| A11 | 66.76% | 61.32% |
| A12 | 89.00% | 77.20% |
| A13 | 77.22% | 70.81% |
| A14 | 71.14% | 63.80% |
| A15 | 57.07% | 62.44% |
| +/-/- | | 14/0/1 |

attributes. The optimal attributes obtained from correlation analysis will ensure the correctness of the synaptic weight obtained during DHNN training and lead to optimal induced logic in the retrieval phase. However, SHoRA's structure, which focuses on higher order logical structure, leads to higher accuracy as it allows for more neuron interactions compare to S2SATRA and the higher logical structure has a higher possibility of satisfying the clause. The flowchart in Fig. 2 shows that ensuring the satisfaction of the clause is crucial before calculating the synaptic weight, and

having more possibilities of logic state ensures the satisfaction of the clause. On the other hand, although 3SATRA has a logical structure like SHoRA, SHoRA still outperforms it because 3SATRA only utilizes random attribute selection which may result in non-significant attributes being trained, leading to lower TP and TN values.

Of all the logic mining models mentioned, the RA model has the worst performance as eight out of fifteen datasets have an accuracy of less than 50%. This is followed by 2SATRA and L2SATRA, with

Table 19
Accuracy SHoRA vs S2SATRA.

| | SHoRA | S2SATRA |
|-------|---------------|---------------|
| A1 | 65.83% | 81.00% |
| A2 | 84.28% | 84.71% |
| A3 | 66.96% | 64.73% |
| A4 | 85.38% | 87.60% |
| A5 | 92.63% | 89.63% |
| A6 | 75.51% | 73.64% |
| A7 | 87.59% | 87.93% |
| A8 | 86.45% | 84.84% |
| A9 | 78.97% | 77.26% |
| A10 | 89.27% | 88.54% |
| A11 | 66.76% | 64.71% |
| A12 | 89.00% | 97.00% |
| A13 | 77.22% | 75.12% |
| A14 | 71.14% | 92.78% |
| A15 | 57.07% | 64.39% |
| + = - | | 8/0/7 |

Table 20
Accuracy SHoRA vs 3SATRA.

| | SHoRA | 3SATRA |
|-------|---------------|---------------|
| A1 | 65.83% | 60.17% |
| A2 | 84.28% | 66.96% |
| A3 | 66.96% | 56.55% |
| A4 | 85.38% | 67.69% |
| A5 | 92.63% | 57.63% |
| A6 | 75.51% | 68.41% |
| A7 | 87.59% | 78.55% |
| A8 | 86.45% | 67.10% |
| A9 | 78.97% | 65.13% |
| A10 | 89.27% | 73.90% |
| A11 | 66.76% | 57.21% |
| A12 | 89.00% | 56.70% |
| A13 | 77.22% | 64.11% |
| A14 | 71.14% | 66.84% |
| A15 | 57.07% | 62.60% |
| + = - | | 14/0/1 |

five and six out of fifteen datasets, respectively, scoring less than 50% accuracy. The low accuracy of these models is mainly due to their logical structure. RA logic mining model consists of Horn2sat logical structure (Sathasivam and Wan Abdullah, 2011) whereas 2SATRA and L2SATRA are based on 2SAT logical structure (Kho et al., 2020a, 2020b; Jamaludin et al., 2021). The Horn2sat logical structure of the RA model is rigid as it demands that each literal be both positive and negative, while the lower logical structure of both 2SATRA and L2SATRA limits the interaction between the neurons, leading to lower TP and TN. The Friedman test is utilized to determine the average rank and to ascertain whether there is a significant difference between the proposed SHoRA model and the existing models.

Referring to the Friedman test results in Table 24, it was found that S2SATRA achieved the highest average rank of 1.900 for accuracy, followed by SHoRA with an average rank of 2.200, indicating that these two logic mining models are the most effective in terms of accuracy on the datasets used in the study. These results shows that S2SATRA and SHoRA models can capture the underlying patterns and relationships in the datasets better than the other models. This could be attributed to the fact that both models utilize correlation analysis to select the most significant attributes for learning by DHNN, which could have led to the better performance in terms of accuracy. Based on Table 24, the p-values obtained from the Friedman test for accuracy values is 0.000. This indicated that there is a significant difference in the accuracy value between the proposed SHoRA model and existing works. The null hypothesis is therefore rejected, suggesting that the SHoRA model performs significantly better in terms of accuracy than the existing models.

7.3. Sensitivity for all logic mining model

Fig. 4 compares the sensitivity values obtained by the proposed ShoRA model with other existing logic mining models, meanwhile Table 21 summarized the result of sensitivity for all the logic mining models and the ratio of improvement between the proposed model and existing logic mining models. As been mentioned in the previous section, this performance metrics measure how well a model correctly classified true positive value. The positive ratio of improvement for SHoRA is higher than the negative ratio of improvement, indicating that ShoRA outperform majority of the existing works in terms of sensitivity. SHoRA wins six out of eight existing logic mining models in producing higher value of sensitivity for 15 datasets. Although it cannot compete with P2SATRA and S2SATRA in producing better sensitivity values, SHoRA model can produce sensitivity values that are comparable to those P2SATRA and S2SATRA for thirteen out of fifteen datasets. In addition, SHoRA model exhibits a remarkable consistency in producing sensitivity values more than 0.7 for 86.6% of the datasets. This proves that this proposed model is consistent in predicting the True Positive (TP) values well during the retrieval phase of DHNN. Not only that, the proposed SHoRA model also obtained the highest sensitivity value on dataset A7 and A9 which is **0.920** and **0.954** respectively compared to other existing works in logic mining. Based on these results, the implementation of multiunit Discrete Hopfield Neural Networks, *Mu*-DHNN enhance the performance of the proposed model specifically in producing high sensitivity values. Instead of having single Q_{best} , the proposed SHoRA model formed 10 Q_{best} by employing *Mu*-DHNN. Given that all current logic mining models concentrate solely on single Q_{best} , this implies that the number of induced logics discovered during the retrieval phase will be limited. However, the use of 10 Q_{best} in SHoRA model enhances the performance by allowing multiple patterns to be stored and retrieved simultaneously. Therefore, this allows the model to capture more complex relationship between attributes and indirectly enhanced the search space in obtaining optimal induced logic in which will leads to producing higher true positive values.

On top of employing *Mu*-DHNN, permutation operator that is implemented in this study also boost the performance of SHoRA model. Additionally, the permutation operators implemented in SHoRA model allow the model to explore a larger search space and find better combinations of logical rules that can correctly classify TP values. The permutation operator rearranges the order of the attributes, which can generate new combinations and potentially identify new patterns that were not previously discovered. The flexible connection among attributes indirectly enhances the search space of getting optimal induced logic with high TP. This in turn improve the model's ability to correctly classify TP and thus increase the sensitivity values. Therefore, by applying *Mu*-DHNN and permutation operator iteratively during the training process, the model can continue to explore the search space and refine its logical rules, leading to further improvements in sensitivity. The factors mentioned above highlight the capabilities of the proposed model to produce sensitivity values that are comparable to those P2SATRA and S2SATRA models.

Besides that, the result from the Friedman test of sensitivity in Table 24 shows that $p - value = 0.001$ which is less than 0.05 indicates that the null hypothesis can be rejected, and we can conclude that there is a significant difference between all those models in finding the sensitivity values.

7.4. Precision for all logic mining model

The sensitivity values on the previous sub section emphasize on true positive outcomes, whereas precision on the other hand

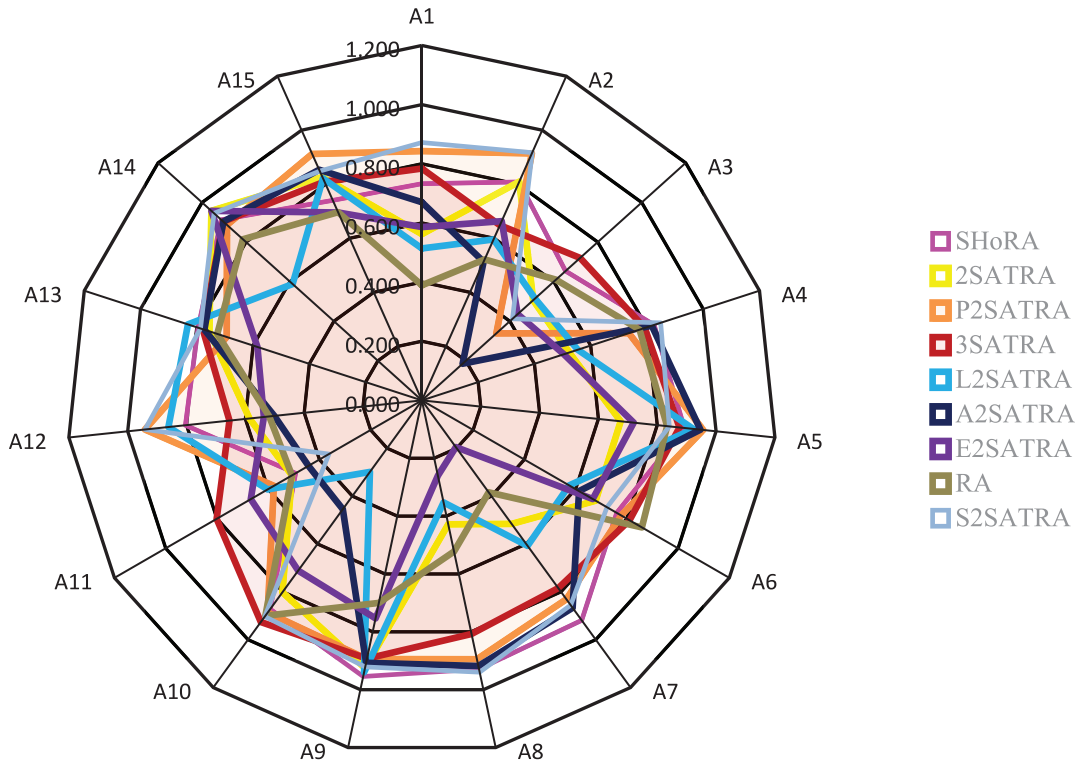


Fig. 4. Sensitivity for all logic mining models.

Table 21

Sensitivity for all logic mining model. The bracket indicates the ratio of improvement and the negative ratio implies the method outperformed the proposed SHoRA model.

| Dataset | SHoRA | 2SATRA | P2SATRA | 3SATRA | L2SATRA | A2SATRA | E2SATRA | RA | S2SATRA |
|---------|--------------|----------------|----------------|----------------|---------------|---------------|---------------|----------------|----------------|
| A1 | 0.733 | 0.562 (0.303) | 0.843(-0.131) | 0.784(-0.066) | 0.514(0.426) | 0.670(0.093) | 0.586(0.250) | 0.389 (0.883) | 0.872 (-0.16) |
| A2 | 0.808 | 0.809 (-0.001) | 0.914 (-0.116) | 0.648 (0.247) | 0.596(0.357) | 0.513(0.576) | 0.663 (0.219) | 0.522(0.550) | 0.917 (-0.118) |
| A3 | 0.654 | 0.506 (0.291) | 0.339(0.930) | 0.720(-0.092) | 0.515(0.270) | 0.185(2.526) | 0.436(0.499) | 0.610 (0.071) | 0.412 (0.586) |
| A4 | 0.826 | 0.524(0.576) | 0.736(0.122) | 0.792 (0.042) | 0.554 (0.490) | 0.817(0.010) | 0.505(0.636) | 0.774(0.067) | 0.849 (-0.027) |
| A5 | 0.892 | 0.676(0.320) | 0.956 (-0.067) | 0.876(0.019) | 0.923(-0.033) | 0.947(-0.058) | 0.720 (0.239) | 0.834(0.070) | 0.841 (0.061) |
| A6 | 0.760 | 0.678 (0.121) | 0.778(-0.023) | 0.811(-0.062) | 0.575(0.323) | 0.613(0.240) | 0.667(0.140) | 0.859(-0.115) | 0.723 (0.052) |
| A7 | 0.920 | 0.516 (0.781) | 0.831 (0.106) | 0.785(0.172) | 0.608(0.512) | 0.867(0.060) | 0.195 (3.708) | 0.384 (1.392) | 0.856 (0.074) |
| A8 | 0.929 | 0.428 (1.168) | 0.895(0.038) | 0.807 (0.151) | 0.353(1.634) | 0.918(0.012) | 0.260 (2.570) | 0.522 (0.778) | 0.940 (-0.011) |
| A9 | 0.954 | 0.926(0.031) | 0.894 (0.067) | 0.893(0.069) | 0.940 (0.015) | 0.907(0.052) | 0.754(0.2665) | 0.700 (0.363) | 0.921 (0.036) |
| A10 | 0.872 | 0.802 (0.088) | 0.887(-0.016) | 0.926 (-0.058) | 0.299(1.917) | 0.455(0.918) | 0.711 (0.226) | 0.900 (-0.031) | 0.903 (-0.034) |
| A11 | 0.493 | 0.507 (-0.028) | 0.578(-0.147) | 0.801 (-0.385) | 0.602(-0.182) | 0.443(0.113) | 0.668(-0.262) | 0.507 (-0.028) | 0.369 (0.358) |
| A12 | 0.804 | 0.586 (0.372) | 0.941 (-0.146) | 0.654 (0.229) | 0.862(-0.067) | 0.502(0.603) | 0.537 (0.497) | 0.519(0.549) | 0.941 (-0.146) |
| A13 | 0.795 | 0.755 (0.053) | 0.694(0.147) | 0.779 (0.021) | 0.832(-0.043) | 0.773 (0.029) | 0.585(0.361) | 0.727 (0.094) | 0.787 (0.011) |
| A14 | 0.913 | 0.958(-0.047) | 0.887(0.030) | 0.904(0.010) | 0.588(0.552) | 0.903 (0.010) | 0.958(-0.047) | 0.814 (0.121) | 0.946(-0.035) |
| A15 | 0.740 | 0.827(-0.105) | 0.912 (-0.188) | 0.810(-0.086) | 0.822(-0.099) | 0.854(-0.133) | 0.695 (0.065) | 0.697(0.063) | 0.849 (-0.128) |
| (+/-) | - | 11/0/4 | 7/0/8 | 9/0/6 | 10/0/5 | 13/0/2 | 13/0/2 | 12/0/3 | 7/0/8 |

concerned on how accurate the positive prediction made by the model. It is essential as well to examine the precision values obtained by all models in detail to evaluate the effectiveness of the logic mining models. The precision values and the ratio of improvement between SHoRA and other logic mining models were recorded and compared as in Table 22. Fig. 5 illustrates the trend of precision values obtained by the proposed SHoRA model and current logic mining models. In this study, the SHoRA model achieved an average precision value of 77%, and outperformed all existing logic mining models. by having the highest precision values when compared one to one existing works. Not only that, SHoRA also attained highest precision value for seven out of fifteen datasets when compared to all existing works as shown in Table 22. The reason for this superior performance is this study proposed new Q_i^{learn} that both true positive and true negative values such that $Q_i^{learn} = (1, -1)$. By considering both true positive and true negative

values, Q_{best} is formed based on the summation of true positive and true negative values and the maximum value of $TP + TN$ will be selected as Q_{best} . However, all current logic mining models focusing solely on true positive values in which the Q_{best} is obtained based on the frequency of true positive values. This becomes more worst as it may face with the issue of imbalanced datasets since we deal with real life datasets which will affect the performance of logic mining models by causing overfitting or underfitting. When most values in the learning dataset are 1 compared to -1, the model tends to overfit and become too complex. Conversely, when the majority of values in the learning dataset are -1 compared to 1, the model tends to underfit due to having less data to be learned. This will turn into bias and ineffective models as the DHNN is not able to produce correct synaptic weight resulting in retrieving sub-optimal final neuron states. Thus, this will lead to lower precision values as the models are not able to classify TP and TN values

Table 22

Precision for all logic mining models. The bracket indicates the ratio of improvement, and the negative ratio implies the method outperformed the proposed SHoRA model. The bracket (*) indicates that no ratio of improvement since no precision value of the respected dataset.

| Dataset | SHoRA | 2SATRA | P2SATRA | 3SATRA | L2SATRA | A2SATRA | E2SATRA | RA | S2SATRA |
|---------|--------------|---------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| A1 | 0.707 | 0.537 (0.316) | 0.628 (0.126) | 0.634 (0.115) | 0.518 (0.365) | 0.674(0.050) | 0.659 (0.073) | 0.610 (0.16) | 0.822(-0.139) |
| A2 | 0.788 | 0.461 (0.708) | 0.751 (0.048) | 0.604 (0.303) | 0.451(0.748) | 0.611 (0.288) | 0.440(0.791) | 0.444(0.775) | 0.750(0.051) |
| A3 | 0.535 | 0.371 (0.440) | 0.391 (0.367) | 0.443(0.208) | 0.355 (0.506) | 0.286 (0.867) | 0.410 (0.304) | 0.372(0.438) | 0.525 (0.018) |
| A4 | 0.896 | 0.648(0.384) | 0.823 (0.089) | 0.687 (0.304) | 0.751 (0.193) | 0.910 (-0.015) | 0.634(0.415) | 0.655(0.369) | 0.922(-0.028) |
| A5 | 0.972 | 0.568 (0.710) | 0.938(0.037) | 0.614(0.584) | 0.703 (0.383) | 0.864(0.125) | 0.901(0.080) | 0.606(0.604) | 0.941 (0.033) |
| A6 | 0.912 | 0.793(0.150) | 0.906 (0.006) | 0.788(0.157) | 0.850(0.073) | 0.937 (-0.027) | 0.745(0.224) | 0.787 (0.158) | 0.925(-0.014) |
| A7 | 0.884 | 0.552 (0.601) | 0.815(0.084) | 0.861 (0.026) | 0.838(0.055) | 0.915 (-0.034) | 0.442(1.000) | 0.590 (0.498) | 0.944(-0.064) |
| A8 | 0.899 | 0.718(0.252) | 0.880(0.022) | 0.800 (0.123) | 0.889(0.012) | 0.866 (0.038) | 0.867 (0.036) | 0.842 (0.067) | 0.871 (0.032) |
| A9 | 0.721 | 0.687 (0.049) | 0.700 (0.030) | 0.609(0.184) | 0.580 (0.243) | 0.710 (0.015) | 0.732 (-0.016) | 0.495 (0.456) | 0.713 (0.011) |
| A10 | 0.854 | 0.671(0.273) | 0.815 (0.048) | 0.603 (0.416) | Nan(*) | 0.599(0.426) | 0.606 (0.409) | 0.379(1.253) | 0.827(0.033) |
| A11 | 0.682 | 0.462(0.477) | 0.634(0.077) | 0.526(0.299) | 0.542(0.260) | 0.573(0.191) | 0.469(0.456) | 0.462(0.477) | 0.593 (0.150) |
| A12 | 0.809 | 0.287 (1.817) | 0.950 (-0.149) | 0.405(0.999) | 0.279(1.902) | 0.617(0.312) | Nan(*) | 0.264(2.068) | 0.950 (-0.149) |
| A13 | 0.659 | 0.554 (0.189) | 0.644 (0.023) | 0.541(0.219) | 0.448(0.472) | 0.582(0.133) | 0.521 (0.265) | 0.336 (0.961) | 0.621(0.062) |
| A14 | 0.665 | 0.543 (0.225) | 0.571(0.164) | 0.631 (0.053) | 0.422(0.576) | 0.605(0.099) | 0.543(0.225) | 0.545 (0.219) | 0.924(-0.281) |
| A15 | 0.615 | 0.650(-0.053) | 0.699(-0.120) | 0.654 (-0.060) | 0.648 (-0.051) | 0.649 (-0.053) | 0.636 (-0.034) | 0.615 (-0.001) | 0.663 (-0.073) |
| (+/-) | - | 14/0/1 | 13/0/2 | 14/0/1 | 14/0/1 | 11/0/4 | 13/0/2 | 14/0/1 | 8/0/7 |

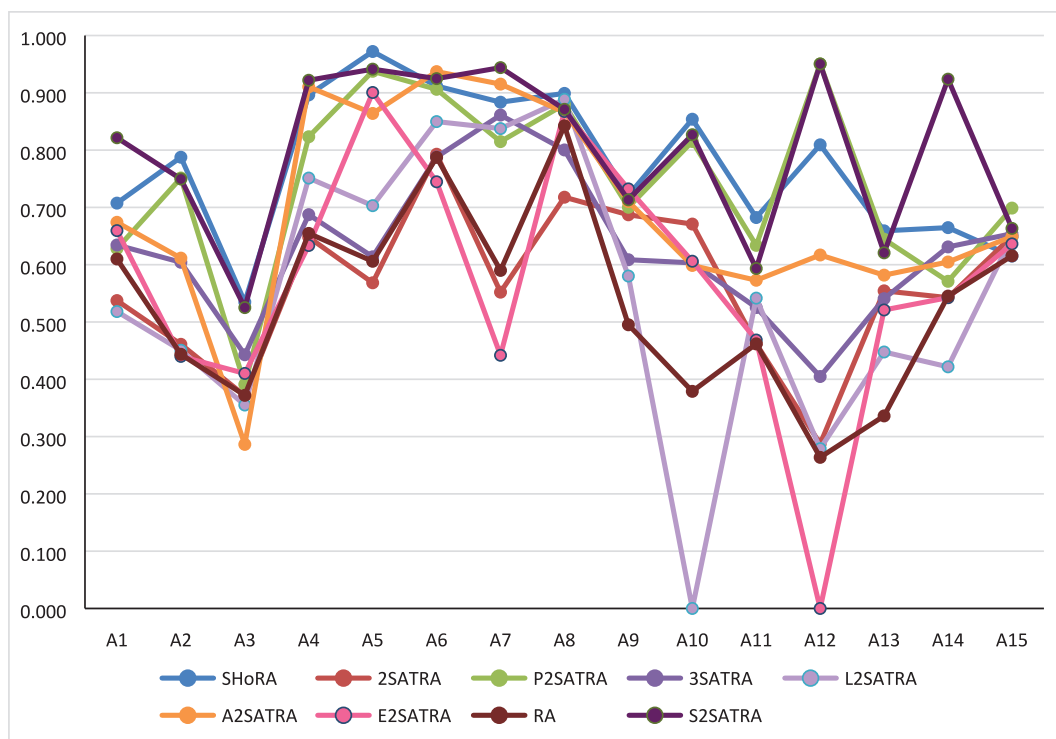


Fig. 5. Precision for all logic mining models.

correctly. Note that high values of TN are equal to low values of FP. This is also contributed to no precision value of L2SATRA and E2SATRA models for A10 and A12 respectively because $TP = FP = 0$. However, SHoRA model addresses this issue by treating true positive and true negative values fairly in forming Q_{best} . This can avoid the model to be overfitted and underfitted thus resulting induced logic to produce high TP and TN values and based on Eq. (18), high TP can elevate the precision values. Therefore, we can summarize that the SHoRA model can distinguish between TP and TN values very well.

In addition, the average rank of the Friedman test in Table 24 for the SHoRA model is 2.267 which is the closest after S2SATRA model that attained 2.100. However, p-value obtained is still significant since it is less than 0.05 which means that there is a significant difference between the SHoRA model and existing logic mining models in obtaining the sensitivity values.

7.5. Matthews correlation coefficients for all logic mining model

While others performance metric considering certain confusion matrix only, MCC on the other hand considers all four possible confusion matrix including TP, TN, FP and FN making it more reliable to measure the performance of all logic mining models. By considering all of these outcomes, MCC can provide a more accurate representation of how well a model is able to produce balanced TP, TN, FP, and FN, which is crucial for avoiding a model from becoming a random classifier. The MCC values which range from -1 to 1 indicates that -1 as the worst MCC value and 1 refers as the best MCC value (Chicco, et al., 2021). Table 23 presented the result of MCC for all the logic mining models. It has clearly been seen that the arrows pointing downward which are in red colour are the majority as compared to the arrows pointing upwards which is in green colour. In other

Table 23

MCC for all logic mining model. The arrows pointing downwards, ↓ indicates that the proposed SHoRA model outperform other logic mining model and the arrows pointing upwards, ↑ shows that the other logic mining model outperform proposed SHoRA model.

| Dataset | SHoRA | 2SATRA | P2SATRA | 3SATRA | L2SATRA | A2SATRA | E2SATRA | RA | S2SATRA |
|---------|--------------|----------|---------|---------|----------|----------|----------|----------|---------|
| A1 | 0.281 | ↓ -0.145 | ↓ 0.076 | ↓ 0.114 | ↓ -0.196 | ↓ 0.154 | ↓ 0.102 | ↓ -0.010 | ↑ 0.602 |
| A2 | 0.609 | ↓ 0.074 | ↑ 0.623 | ↓ 0.298 | ↓ 0.005 | ↓ 0.257 | ↓ -0.018 | ↓ -0.016 | ↑ 0.626 |
| A3 | 0.318 | ↓ 0.005 | ↓ 0.042 | ↓ 0.189 | ↓ -0.078 | ↓ -0.004 | ↓ 0.065 | ↓ 0.022 | ↓ 0.206 |
| A4 | 0.596 | ↓ 0.069 | ↓ 0.441 | ↓ 0.214 | ↓ 0.214 | ↓ 0.593 | ↓ 0.045 | ↓ 0.148 | ↑ 0.654 |
| A5 | Nan | Nan | Nan | Nan | Nan | Nan | Nan | Nan | Nan |
| A6 | Nan | Nan | Nan | Nan | Nan | Nan | Nan | Nan | Nan |
| A7 | 0.736 | ↓ -0.186 | ↓ 0.535 | ↓ 0.567 | ↓ 0.413 | ↓ 0.730 | ↓ -0.060 | ↓ -0.144 | ↑ 0.759 |
| A8 | 0.456 | ↓ -0.177 | ↓ 0.354 | ↓ 0.062 | ↓ 0.146 | ↓ 0.331 | ↓ 0.054 | ↓ 0.0710 | Nan |
| A9 | 0.608 | ↓ 0.523 | ↓ 0.523 | ↓ 0.327 | ↓ 0.269 | ↓ 0.541 | ↓ 0.461 | ↓ -0.051 | ↓ 0.567 |
| A10 | 0.778 | ↓ 0.547 | ↓ 0.750 | ↓ 0.546 | Nan | ↓ 0.269 | ↓ 0.454 | ↓ -0.002 | ↓ 0.773 |
| A11 | 0.207 | ↓ -0.074 | ↓ 0.078 | ↓ 0.090 | ↓ 0.057 | ↓ 0.060 | ↓ -0.049 | ↓ -0.074 | ↓ 0.102 |
| A12 | 0.730 | ↓ Nan | ↑ 0.925 | ↓ 0.186 | ↓ Nan | ↓ 0.405 | ↓ Nan | ↓ -0.054 | ↑ 0.925 |
| A13 | 0.541 | ↓ 0.360 | ↓ 0.467 | ↓ 0.361 | ↓ Nan | ↓ 0.434 | ↓ 0.290 | ↓ -0.116 | ↓ 0.496 |
| A14 | 0.446 | ↓ Nan | ↓ 0.180 | ↓ 0.372 | ↓ -0.107 | ↓ 0.299 | ↓ Nan | ↓ 0.073 | ↑ 0.857 |
| A15 | 0.010 | ↓ Nan | ↑ 0.361 | ↑ 0.141 | ↓ Nan | ↓ Nan | ↓ Nan | ↑ 0.018 | ↑ 0.107 |
| (+/-/-) | | 13/2/0 | 10/2/3 | 12/2/1 | 13/2/0 | 13/2/0 | 13/2/0 | 12/2/1 | 6/2/7 |

Table 24

Friedman test for all logic mining model. The P-value is obtained from Friedman test and ** indicates there is significant difference between the model and existing logic mining models.

| Average Rank | Accuracy | Sensitivity | Precision | MCC | F1 Score |
|--------------|--------------|--------------|--------------|--------------|--------------|
| SHoRA | 2.200 | 3.533 | 2.267 | 2.333 | 2.133 |
| 2SATRA | 6.667 | 5.600 | 6.800 | 6.833 | 6.200 |
| P2SATRA | 2.900 | 4.100 | 3.500 | 3.767 | 3.233 |
| 3SATRA | 5.267 | 4.000 | 5.467 | 4.600 | 4.533 |
| L2SATRA | 7.400 | 5.600 | 6.600 | 6.967 | 7.067 |
| A2SATRA | 4.067 | 5.600 | 4.200 | 4.567 | 4.800 |
| E2SATRA | 7.100 | 6.833 | 6.433 | 6.667 | 7.300 |
| RA | 7.500 | 6.367 | 7.633 | 6.700 | 7.233 |
| S2SATRA | 1.900 | 3.367 | 2.100 | 2.567 | 2.500 |
| P- VALUE | 0.000** | 0.001** | 0.000** | 0.000** | 0.000** |

words, SHoRA model achieved the most optimal MCC value for 6 out of 15 datasets. Not only that, the SHoRA model outperforms other existing logic mining models by having the highest MCC value for 7 out of 8 existing logic mining. Note that the only dataset that approaches zero MCC is A15. This demonstrates the SHoRA's strong ability to distinguish between all domains of the confusion matrix.

The main reason which contributes to the performance of MCC values is the higher order k satisfiability logic proposed by SHoRA model has a higher logical capacity to represent the dimensionality of the dataset compared to other existing works. As a result, SHoRA produces induced logic with high MCC values which prevents it from turning into random classifier. The logic mining models with lower order logic such as 2SATRA, L2SATRA, E2SATRA, and RA tend to have MCC values mostly less than 0, indicating that they have lower logical capacity compared to the SHoRA model. This lower capacity leads to the models failing to correctly predict TP and TN, which possibly turn the models into random classifier. In particular, L2SATRA is considered to have the worst performance among all logic mining models because it cannot provide MCC values for 6 out of 15 datasets.

Despite having a similar high logical capacity as 3SATRA, the SHoRA model outperforms 3SATRA in terms of MCC values due

to its additional strengths such as the permutation operator. In contrast, 3SATRA's rigid attribute positions limit its ability to generate optimal final neuron states, resulting in inflexible induced logic. Consequently, the 3SATRA model tends to produce unbalanced TP, TN, FP and FN values, leading to low MCC values due to incorrect predictions of TP and TN. Besides that, the S2SATRA model is the second highest performing model in terms of MCC values, as evident from Table 23, with most of the datasets having MCC values close to 1 and the highest A12 dataset with MCC value being 0.925. This highlights the effectiveness of the attribute selection method in improving the model's performance by selecting only the significant attributes and ensuring that the induced logic is optimal. This optimized induced logic results in better prediction of TP and TN values while minimizing the values of FP and FN, thereby balancing the values of TP, TN, FP and FN and ultimately leading to higher MCC values.

Worth mentioning that the average rank of SHoRA model for the performance of MCC as in Table 24 is the highest which is 2.333 compared to other existing logic mining models. Thus, we can conclude that this model is dominant, and this result statistically validates the superiority of SHoRA model compared to existing work. This also can be supported by the Friedman test rank which is conducted based on MCC values for all logic mining

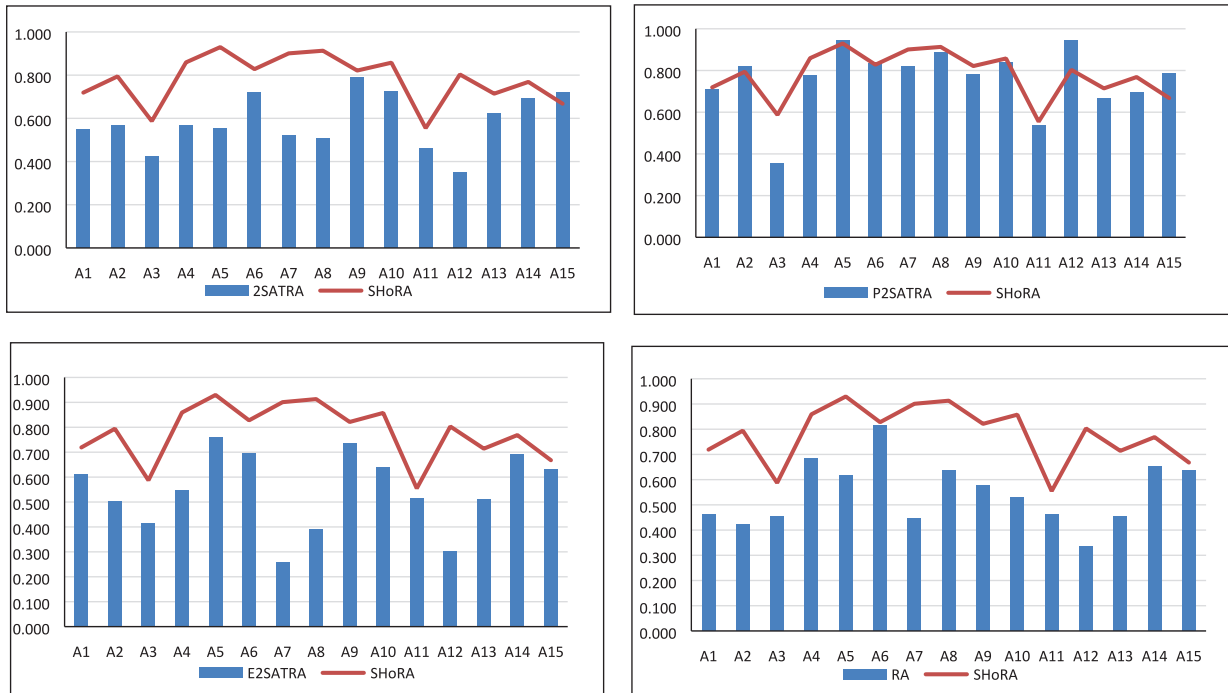


Fig. 6. Comparison of F1 score of SHoRA model (fluctuate line) and existing logic mining models (bar chart).

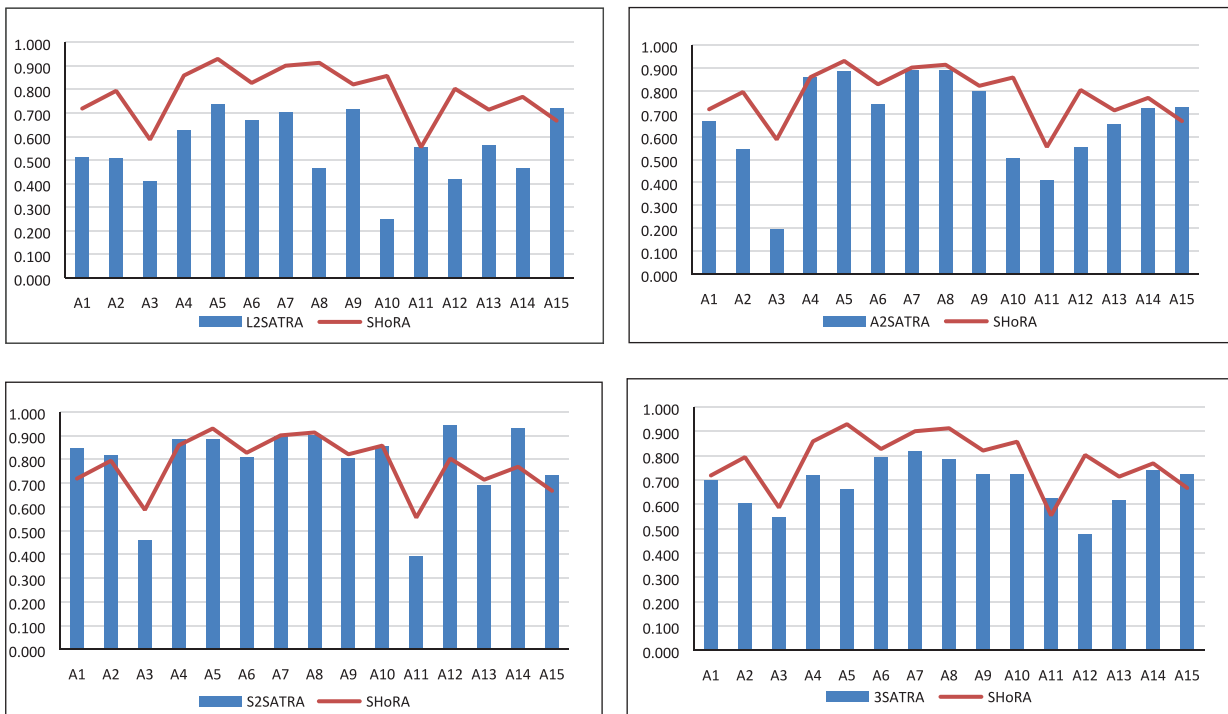


Fig. 7. Comparison of F1 score of SHoRA model (fluctuate line) and existing logic mining models (bar chart).

models with $\alpha = 0.05$. According to Table 24, the p – value of MCC is 0.000 which is less than $\alpha = 0.05$. Therefore, we can reject the null hypothesis and conclude that there is a significant difference between the proposed and existing logic mining in providing MCC value.

7.6. F1 score for all logic mining models

The last performance metric that is considered in evaluating the performance of SHoRA model is F1 score as this metric assess the performance of the classification models. It is the combination of

sensitivity and precision into a single metric and considering both FP and FN as well. Fig. 6 and Fig. 7 illustrated the difference between the F1 score of proposed SHoRA model with other existing logic mining models. The bar chart represents the F1 score for existing logic mining models and the fluctuate line represent the value of F1 score for proposed SHoRA model. In general, we can summarize that proposed model produce highest F1 score for all fifteen datasets when compare with other existing works. A high F1 score obtained by SHoRA model indicates that the model has both high sensitivity and high precision, which means the model is more effective in correctly identifying both positive and negative cases in the dataset. Thus, it can be considered as a better model for classification tasks compared to all existing logic mining models. The implementation of Mu-DHNN contributes to ensure high sensitivity and high precision values by allowing the networks to treasure more than one logical structure. This creates a larger search space, which guarantees that the synaptic weight obtained and stored in CAM during the learning phase of DHNN is optimal. As a result, we obtained a local field with the most optimal final neuron states, resulting in the induced logic correctly identify positive and negative cases in the dataset.

Referring to Table 24, the average rank attained by the SHoRA model is 2.133, which is the highest of all the logic mining models. This demonstrates that, out of all the logic mining models, the SHoRA model is the most effective at achieving the highest F1 score. In Addition, p -value of F1 score obtained from Friedman test is 0.000 which is less than $\alpha = 0.05$. Based on this result, the null hypothesis which assumed that all logic mining models would perform equally is rejected.

8. Conclusion

In this paper, an efficient and effective logic mining model is proposed in solving the classification of the real-life problems. As in standard 3SATRA logic mining model, attributes are randomly selected. Thus, there is a high chance that the non-significant attributes will be selected which results in low capability of the DHNN to train the selected attributes to obtain correct synaptic weight. Therefore, supervised learning by using correlation test has been adopted in this study to analyse all the attributes and select only the optimal attributes to represent the logical rule. Not only that, this study also explore the new approach of finding Q_{best} by considering true positive and true negative outcomes such that $Q_i^{learn} = (1, -1)$. The logical structure with the highest total of Q_i^{learn} will be choose as Q_{best} . In addition to that, multiunit Discrete Hopfield Neural Network (Mu-DHNN) is proposed in this study which will enhance the search space that leads to optimal induced logic produce in retrieval phase of DHNN. Lastly, permutation operator has been implemented to enhance the connectivity among attributes in the clause. The performance of SHoRA model will be compared with all existing works in logic mining based on five performance metric including accuracy, sensitivity, precision, MCC and F1 score. Based on the results, the proposed SHoRA able to outperforms seven out of eight existing logic mining models. This shows that the proposed model has a high capability in solving classification problems. Since this study focus on systematic logical rule, future exploration might be more exciting if we can compare the performance of systematic logical rule and non-systematic logical rule in the perspective of logic mining since non-systematic logical structure offer more flexibility among the attributes compared to systematic logical structure.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

All the authors gratefully acknowledged the financial support from the Universiti Sains Malaysia for Short Term Grant with Grant Number 304/PMATHS/6315655. We would like to thank all AIRDG members and those who gave generously of their time, ideas and hospitality in the preparation of this manuscript.

Funding information

This research was supported by Universiti Sains Malaysia for Short Term Grant with Grant Number 304/PMATHS/6315655.

References

- Abdullah, W.A.T.W., 1992. Logic programming on a neural network. *Int. J. Intell. Syst.* 7 (6), 513–519. <https://doi.org/10.1002/int.4550070604>.
- Ajiboye, A.R., Abdullah-Arshah, R., Qin, H., Isah-Kebbe, H., 2015. Evaluating the effect of dataset size on predictive model using supervised learning technique. *Int. J. Comput. Syst. Softw. Eng.* 1 (1), 75–84. <https://doi.org/10.15282/ijsecs.1.2015.6.0006>.
- Alway, A., Zamri, N.E., Mohd Kasihmuddin, M.S., Mansor, A., Sathasivam, S., 2020. Palm oil trend analysis via logic mining with discrete hopfield neural network. *Pertanika J. Sci. Technol.* 28 (3), 967–981.
- Alzaeemi, S.A., Sathasivam, S., 2020. Artificial immune system in doing 2-satisfiability based reverse analysis method via a radial basis function neural network 1–28 *Processes* 8 (10), 1295. <https://doi.org/10.3390/pr8101295>.
- Ayalew, A.M., Salau, A.O., Abeje, B.T., Enyew, B., 2022. Detection and classification of COVID-19 disease from X-ray images using convolutional neural networks and histogram of oriented gradients. *Biomed. Signal Process. Control* 74. <https://doi.org/10.1016/j.bspc.2022.103530> 103530.
- Chicco, D., Jurman, G., 2020. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics* 21 (1), 1–13. <https://doi.org/10.1186/s12864-019-6413-7>.
- Chicco, D., Tötsch, N., Jurman, G., 2021. The Matthews correlation coefficient (MCC) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation. *BioData mining* 14 (1), 1–22.
- Gao, S., Wang, P., Feng, Y., Xie, X., Duan, M., Fan, Y., Zhou, F., 2021. RIFS2D: A two-dimensional version of a randomly restarted incremental feature selection algorithm with an application for detecting low-ranked biomarkers. *Comput. Biol. Med.* 133. <https://doi.org/10.1016/j.combiomed.2021.104405> 104405.
- Guo, Y., Kasihmuddin, M.S.M., Gao, Y., Mansor, M.A., Wahab, H.A., Zamri, N.E., Chen, J., 2022. YRAN2SAT: A novel flexible random satisfiability logical rule in discrete hopfield neural network. *Adv. Eng. Softw.* 171. <https://doi.org/10.1016/j.advengsoft.2022.103169> 103169.
- Hossin, M., Sulaiman, M.N., 2015. A review on evaluation metrics for data classification evaluations. *Int. J. Data Mining Knowledge Manage. Process* 5 (2), 1. <https://doi.org/10.5121/ijdkp.2015.5201>.
- Jamaludin, S.Z.M., Romli, N.A., Kasihmuddin, M.S.M., Baharum, A., Mansor, M.A., Marsani, M.F., 2022a. Novel logic mining incorporating log linear approach. *J. King Saud Univ. -Computer Information Sci.* 34 (10), 9011–9027. <https://doi.org/10.1016/j.jksuci.2022.08.026>.
- Jamaludin, S.Z.M., Sa'ari, N.S., Kasihmuddin, M.S.M., Marsani, M.F., Zamri, N.E., Azhar, S.A., Mansor, M.A., 2022b. Artificial bee colony for logic mining in credit scoring. *Malaysian J. Fundamental Appl. Sci.* 18 (6), 654–673. <https://doi.org/10.11113/mjfas.v18n6.2661>.
- Ji, Y., Wang, L., Xie, D., 2023. Balance optimization method of energy shipping based on Hopfield neural network. *Alex. Eng. J.* 67, 171–181. <https://doi.org/10.1016/j.aej.2022.12.038>.
- Karim, S.A., Zamri, N.E., Alway, A., Kasihmuddin, M.S.M., Ismail, A.I.M., Mansor, M.A., Hassan, N.F.A., 2021. Random satisfiability: a higher-order logical approach in discrete Hopfield Neural Network. *IEEE Access* 9, 50831–50845. <https://doi.org/10.1109/ACCESS.2021.3068998>.
- Kasihmuddin, M.S.M., Mansor, M., Sathasivam, S., 2017. Robust artificial bee colony in the hopfield network for 2-satisfiability problem. *Pertanika J. Sci. Technol.* 25 (2), 453–468.

- Kasihmuddin, M.S.M., Mansor, M.A., Sathasivam, S., 2018. Discrete Hopfield neural network in restricted maximum k-satisfiability logic programming. *Sains Malaysiana* 47 (6), 1327. <https://doi.org/10.17576/jism-2018-4706-30>.
- Kasihmuddin, M.S.M., Jamaludin, S.Z.M., Mansor, M.A., Wahab, H.A., Ghadzi, S.M.S., 2022. Supervised learning perspective in logic mining. *Mathematics* 10 (6), 915. <https://doi.org/10.3390/math10060915>. 1-35.
- Kaur, B., Rathi, S., Agrawal, R.K., 2022. Enhanced depression detection from speech using Quantum Whale Optimization Algorithm for feature selection. *Comput. Biol. Med.* 150, <https://doi.org/10.1016/j.combiomed.2022.106122> 106122.
- Kho, L.C., Kasihmuddin, M.S.M., Mansor, M., Sathasivam, S., 2020a. Logic Mining in League of Legends. *Pertanika J. Sci. Technol.* 28 (1), 211–225.
- Kho, L.C., Kasihmuddin, M.S.M., Mansor, M.A., Sathasivam, S., 2020b. Logic mining in football matches. *Indones. J. Electr. Eng. Comput. Sci* 17, 1074–1083. <https://doi.org/10.11591/ijeecs.v17.i2>.
- Khosahval, F., Fadaei, A., 2012. Application of a hybrid method based on the combination of genetic algorithm and Hopfield neural network for burnable poison placement. *Ann. Nucl. Energy* 47, 62–68. <https://doi.org/10.1016/j.anucene.2012.04.020>.
- Labani, M., Moradi, P., Ahmadizar, F., Jalili, M., 2018. A novel multivariate filter method for feature selection in text classification problems. *Eng. Appl. Artif. Intel.* 70, 25–37.
- Mahendran, N., PM, D.R.V., 2022. A deep learning framework with an embedded-based feature selection approach for the early detection of the Alzheimer's disease. *Comput. Biol. Med.* 141, <https://doi.org/10.1016/j.combiomed.2021.105056> 105056.
- Mansor, M., Kasihmuddin, M., Sathasivam, S., 2017. Artificial immune system paradigm in the hopfield network for 3-Satisfiability problem. *Pertanika J. Sci. Technol.* 25 (4), 1173–1188.
- Mistry, A., Shah, V., 2016. Brief survey of data mining techniques applied to applications of agriculture. *Int. J. Adv. Res. Computer Commun. Eng.* 5 (2), 301–304. <https://doi.org/10.17148/IJARCC.2016.5263>.
- Mohd Jamaludin, S., Mansor, M., Baharum, A., Mohd Kasihmuddin, M., Wahab, A., Marsani, M., 2023. Modified 2 satisfiability reverse analysis method via logical permutation operator. *Computers Mater. Continua*, 2853–2870.
- Mohd Jamaludin, S.Z., Mohd Kasihmuddin, M.S., Md Ismail, A.I., Mansor, M.A., Md Basir, M.F., 2020. Energy based logic mining analysis with hopfield neural network for recruitment evaluation. *Entropy* 23 (1), 40. <https://doi.org/10.3390/e23010040>.
- Mohd Jamaludin, S.Z., Ismail, M.T., Mohd Kasihmuddin, M.S., Mansor, A., Farwina Mohamad Anwar Antony, S.N., Makhul, A.A., 2021. Modelling benign ovarian cyst risk factors and symptoms via log-linear model. *Pertanika J. Sci. Technol.* 29 (3), 2199–2216. <https://doi.org/10.47836/pjst.29.3.26>.
- Muhammad Sidik, S.S., Zamri, N.E., Mohd Kasihmuddin, M.S., Wahab, H.A., Guo, Y., Mansor, M.A., 2022. Non-Systematic weighted satisfiability in discrete hopfield neural network using binary artificial bee colony optimization. *Mathematics* 10 (7), 1129. <https://doi.org/10.3390/math10071129>.
- Mwadulo, M.W., 2016. A review on feature selection methods for classification tasks. *Int. J. Computer Appl. Technol. Res.* 5 (6), 395–402.
- Patel, H., Singh Rajput, D., Thippa Reddy, G., Iwendi, C., Kashif Bashir, A., Jo, O., 2020. A review on classification of imbalanced data for wireless sensor networks. *Int. J. Distrib. Sens. Netw.* 16 (4).
- Piri, J., Mohapatra, P., 2021. An analytical study of modified multi-objective Harris Hawk Optimizer towards medical data feature selection. *Comput. Biol. Med.* 135, <https://doi.org/10.1016/j.combiomed.2021.104558> 104558.
- Purusothaman, G., Krishnakumari, P., 2015. A survey of data mining techniques on risk prediction: Heart disease. *Indian J. Sci. Technol.* 8 (12), 1–5. <https://doi.org/10.17485/ijst/2015/v8i12/58385>.
- Ramyachitra, D., Manikandan, P., 2014. Imbalanced dataset classification and solutions: a review. *Int. J. Comput. Bus. Res. (IJCBR)* 5 (4), 1–29.
- Sani, S., Shermeh, H.E., 2022. A novel algorithm for detection of COVID-19 by analysis of chest CT images using Hopfield neural network. *Expert Syst. Appl.* 197, <https://doi.org/10.1016/j.eswa.2022.116740> 116740.
- Sathasivam, S., 2010. Upgrading logic programming in Hopfield network. *Sains Malaysiana* 39 (1), 115–118.
- Sathasivam, S., Mansor, M.A., Kasihmuddin, M.S.M., Abubakar, H., 2020. Election algorithm for random k satisfiability in the Hopfield neural network. *Processes* 8 (5), 568. <https://doi.org/10.3390/pr8050568>. 1-19.
- Sathasivam, S., Wan Abdullah, W.A.T., 2011. Logic mining in neural network: reverse analysis method. *Computing* 91 (2), 119–133. <https://doi.org/10.1007/s00607-010-0117-9>.
- Sokolova, M., Lapalme, G., 2009. A systematic analysis of performance measures for classification tasks. *Inf. Process. Manag.* 45 (4), 427–437. <https://doi.org/10.1016/j.ipm.2009.03.002>.
- Sugumaran, V., Ramachandran, K.I., 2011. Effect of number of features on classification of roller bearing faults using SVM and PSVM. *Expert Syst. Appl.* 38 (4), 4088–4096. <https://doi.org/10.1016/j.eswa.2010.09.072>.
- Sunil, G.C., Zhang, Y., Koparan, C., Ahmed, M.R., Howatt, K., Sun, X., 2022. Weed and crop species classification using computer vision and deep learning technologies in greenhouse conditions. *J. Agric. Food Res.* 9, <https://doi.org/10.1016/j.jafr.2022.100325> 100325.
- Valero-Carreras, D., Alcaraz, J., Landete, M., 2023. Comparing two SVM models through different metrics based on the confusion matrix. *Comput. Oper. Res.* 152, <https://doi.org/10.1016/j.cor.2022.106131> 106131.
- Wang, M., Li, X., Chen, L., 2022. An enhance multimodal multiobjective optimization genetic algorithm with special crowding distance for pulmonary hypertension feature selection. *Comput. Biol. Med.* 146, <https://doi.org/10.1016/j.combiomed.2022.105536> 105536.
- Wen, U.P., Lan, K.M., Shih, H.S., 2009. A review of Hopfield neural networks for solving mathematical programming problems. *Eur. J. Oper. Res.* 198 (3), 675–687. <https://doi.org/10.1016/j.ejor.2008.11.002>.
- Xu, S., Wang, X., Ye, X., 2022. A new fractional-order chaos system of Hopfield neural network and its application in image encryption. *Chaos Solitons Fractals* 157, <https://doi.org/10.1016/j.chaos.2022.111889> 111889.
- Yu, B., Mao, W., Lv, Y., Zhang, C., Xie, Y., 2022. A survey on federated learning in data mining. *Wiley Interdisc. Rev.: Data Mining Knowledge Discovery* 12 (1), e1443.
- Zamri, N., Mansor, M., Mohd Kasihmuddin, M., Alway, A., Mohd Jamaludin, S., Alzaeemi, S.A., 2020. Amazon employees resources access data extraction. *Entropy* 22 (596), 1–25. <https://doi.org/10.3390/e22060596>.
- Zhao, X., He, Y., Wu, Y., Liu, T., Wang, G., 2022. IOFS-SA: an interactive online feature selection tool for survival analysis. *Comput. Biol. Med.* 150, <https://doi.org/10.1016/j.combiomed.2022.106121> 106121.
- Zhu, J., Salhotra, A., Meinecke, C.R., Surendiran, P., Lyttleton, R., Reuter, D., Korten, T., 2022. Solving the 3-Satisfiability problem using network-based biocomputation. *Adv. Intell. Syst.* 4, 2200202. <https://doi.org/10.1002/aisy.202200202>.