AIMS Mathematics, 9(2): 3150–3173.
DOI: 10.3934/math.2024153
Received: 24 July 2023
Revised: 10 September 2023
Accepted: 24 September 2023
Published: 03 January 2024

http://www.aimspress.com/journal/Math

Research article

Logic mining with hybridized 3-satisfiability fuzzy logic and harmony search algorithm in Hopfield neural network for Covid-19 death cases

Farah Liyana Azizan[1,2], Saratha Sathasivam[1,*], Nurshazneem Roslan[1,3] and Ahmad Deedat Ibrahim[2]

[1] School of Mathematical Sciences, Universiti Sains Malaysia, 11800, USM, Penang, Malaysia
[2] Centre for Pre-University Studies, Universiti Malaysia Sarawak, 94300, Kota Samarahan, Sarawak, Malaysia
[3] Institute of Engineering Mathematics, Universiti Malaysia Perlis, 02600, Arau, Perlis, Malaysia

* **Correspondence:** Email: saratha@usm.my; Tel: +6046532428.

Abstract: Since the beginning of the Covid-19 infections in December 2019, the virus has emerged as the most lethally contagious in world history. In this study, the Hopfield neural network and logic mining technique merged to extract data from a model to provide insight into the link between factors influencing the Covid-19 datasets. The suggested technique uses a 3-satisfiability-based reverse analysis (3SATRA) and a hybridized Hopfield neural network to identify the relationships relating to the variables in a set of Covid-19 data. The list of data is to identify the relationships between the key characteristics that lead to a more prolonged time of death of the patients. The learning phase of the hybridized 3-satisfiability (3SAT) Hopfield neural network and the reverse analysis (RA) method has been optimized using a new method of fuzzy logic and two metaheuristic algorithms: Genetic and harmony search algorithms. The performance assessment metrics, such as energy analysis, error analysis, computational time, and accuracy, were computed at the end of the algorithms. The multiple performance metrics demonstrated that the 3SATRA with the fuzzy logic metaheuristic algorithm model outperforms other logic mining models. Furthermore, the experimental findings have demonstrated that the best-induced logic identifies important variables to detect critical patients that need more attention. In conclusion, the results validate the efficiency of the suggested approach, which occurs from the fact that the new version has a positive effect.

Keywords: 3SAT; Covid-19; fuzzy logic system; Hopfield neural network; logic mining; metaheuristic algorithms; reverse analysis

Mathematics Subject Classification: 03B52, 68T27, 68N17, 68W99

1. Introduction

All of today's technological development is driven by artificial intelligence (AI). As a result, it fosters the creation of advanced machine-learning methods to address those issues. Artificial neural networks (ANNs), a subfield of AI, are frequently employed to enhance decision-making across numerous fields [1–3]. AI technology has advanced significantly by creating an integrated structure for neural networks, logic programming models and satisfiability (SAT) forms [4]. The ANN is a robust analytical data processing type that has encountered massive analysis and implementation by experts and scientists due to its function to handle and exhibit complex tasks. A Hopfield neural network (HNN) is an example of the ANN type. Every neuron's output in the HNN is connected to every other neuron's response in a single-level recursive neural network (RNN) [5]. HNN employs a distinctive symbolic training model to efficiently coordinate the propagation of the input and output neurons when addressing issues. The HNN's ability to find the closest minimal solution impacts the dynamic behavior of the neuron state. Abdullah [6] suggests the process for performing logic programming on the HNN. The network accomplished a logical inconsistency reduction in programming once distinguishing the connection strengths, or what is commonly referred to as the synaptic weight, with logic programming; that is, by contrasting cost and energy functions. Abdullah [7] introduces the HNN's training phase straightforwardly. The Abdullah logic paradigm has gained prominence and is currently used often [8–10]. A mathematical framework can explain various everyday scientific and technical problems. However, to do so, one must first develop strategies for handling some mathematical issues.

The application of intelligent models like fuzzy logic techniques that incorporate symbolic logic and neural computing can make it possible to simulate everyday human actions in a computer. Speech recognition [11], regression concerns [12], pattern or image classification [13] and many more issues can be solved using models like ANN. Due to its intricacy, it is challenging to convey information from this intelligent model to those not involved in the AI industry [14]. To enable intelligent systems to deliver outcomes more closely matched with conventional expectations, fuzzy system concepts have arisen. They allow us to make the problem's representation more comprehensible [15]. Combining fuzzy logic with neural network theory might increase the ability of AI technologies to learn from experience and adapt to changes in an environment that contains qualitative, imprecise, unclear or incomplete information. These approaches have been debated in the literature since the 1960s and have successfully addressed many societal problems [16]. The reasons for their growth include the simplicity with which these networks can be analyzed and the simplicity with which net topology knowledge can be derived. These networks offer a range of training capacities and are produced via fuzzy set theory and neural networks. They provide models that combine the capacity for training neural networks with the handling of ambiguous information by fuzzy systems [17]. Realistic solutions can be found for critical issues, such as classifying or locating an ordered list. Using a computer will make it possible to solve these problems. One of the most well-known problems is the SAT or satisfiability. It is defined as a strategy for completing the optimal work while using Boolean values to confirm that the 3SAT expression is met. Earlier research used a single data mining method incorporating the HNN model with 3SAT logic programming to characterize the innovation.

The evaluation of the model's effectiveness has been accomplished and can offer prospective solutions using datasets from real-life problems. Several real-world datasets relevant to various fields, such as the commodity sector, are evaluated using the data mining approach [18]. The procedure is anticipated to produce a common relationship rule that can categorize and forecast the characteristics of anonymous information by obtaining the relationship and information from the data. The use of data mining has been increasingly important in various academic domains over the past 10 years [19]. Although it has been claimed that current data mining techniques produce good accuracy, the emphasis on the black box concept makes it difficult to comprehend the results. In other words, the AI understands the output, but the human as the user does not. Logic mining is a comparatively latest approach to obtaining the characteristics of the dataset utilizing logical rules. Logic mining is one of HNN's most beneficial uses of logic programming. This approach was invented by Sathasivam and Abdullah [20]. When they originally invented the logic mining method, they used HNN to execute the reverse analysis (RA) approach in Horn logic. They successfully exceeded 80% for both support and confidence measures. The RA method extracted each logical rule describing the students' performance, and the critical drawback is the absence of crucial components like generalization and classification abilities. The amount of induced Horn logic emitted by HNN provides the basis for the logical rule retrieved from the datasets. Therefore, finding the greatest induced logical rule corresponding to the datasets is not thoroughly pursued. Data mining is designed to draw valuable information from a set of data using particular techniques. Most techniques' classification decisions are kept from the user, which impedes their grasp of the classification's justification. As a result, data mining is becoming less relevant to actual practitioners, despite several kinds of literature reporting various sets of data but lacking clear user interpretation. The symbolic rule requirement can be implemented into the traditional data mining approach to assure the interpretability of the data gathered. Some studies contain unique SAT logical principles to be inserted into HNN to supplement the prior RA's limitations. The development of the systematic logic mining methodology and the satisfiability-based RA (SATRA) method are widely emerging.

Cases of pneumonia due to a virus later known to be severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) were first reported in Wuhan, Hubei, China. The associated disease, Covid-19, was then confirmed as an outbreak by the World Health Organization (WHO) on 12th March 2020. SARS-CoV-2 was highly infectious with efficacy, mainly recognized to be transmitted through droplets and aerosols. As of May 2020, other suggested transmission routes included sewage, saliva, urine and inanimate surfaces [21]. The total number of reported deaths due to Covid-19 was 193,710 as of 26th April 2020 [22], and 6,935,958 as of 14th May 2023 [23]. SARS-CoV-2, being a virus, mutates over time and forms distinct viral lineages. Some mutations may result in phenotypical changes that carry unusual viral behaviors. To classify notable strains, WHO classifies variants of interest (VOIs) and variants of concerns (VOCs) based on five identified main domains of emerging variants: Increased transmissibility, atypical clinical course, diagnostic failure, declined efficiency of natural and vaccine-derived immunity and decreased susceptibility to therapeutics. Numbers of overall cases, cases among health and care workers, case fatality or test positivity rate departing from the trend or age-disaggregated cases or deaths, hospitalizations rate or bed occupancy rate increasing in specific age groups are trigger thresholds for heightened epidemiological surveillance for possible VOIs and VOCs [24]. On 20th January 2022, the transmission of the Omicron variant was reported to have outpaced the Delta variant alongside the reported increase in the number of new deaths, possibly due to the Omicron variant's immune evasion among others; the Omicron variant was then marked as a VOC [25] Omicron infections in China spiked after the Chinese government's 'Zero Covid' strategies were ended in December 2022. The lethality of Omicron infections was unclear [26].

The work is to build a new hybrid model of HNN for 3SAT and RA methods to extract data from a real-life problem such as Covid-19 cases. The newly developed learning phase of the network is optimized using a fuzzy logic method and two metaheuristic algorithms called genetic and harmony search algorithms. In this case, the experiment outcomes can indicate the relationships between significant characteristics that lead to a more prolonged time of death for Covid-19 patients. The best-induced logic can help to find essential variables to identify crucial patients who need extra attention. The 3SATRA integrated with the fuzzy logic and metaheuristics will improve the ability of the network to select the most significant amount of optimal induced logic. The suggested 3SATRA method will determine the best logical rule from the Covid-19 datasets. The actual synaptic weight throughout the training stage will verify the capacity of the logic mining model. It also will determine the induced logic's accuracy level in the testing phase. This work examined the performance of the hybridized method strategy in the HNN on the data mining and extraction performance of 3SATRA. The article's contribution consists of the following:

(a) To formulate a newly hybridized HNN for 3SAT by integrating the fuzzy logic with the genetic algorithm (GA) and harmony search algorithm (HSA).
(b) To evaluate the hybridized network's capability that can improve the training phase in the HNN by minimizing the cost function during the learning phase, which results in an optimal final neuron state.
(c) To incorporate the suggested hybridized networks into logic mining called 3SATRA to obtain more diverse induced logic.
(d) To induce Covid-19 datasets into a 3SAT logical representation to represent the relationship of best datasets.
(e) To compare the effectiveness of the suggested logic mining with the fundamental logic mining method based on Covid-19 datasets.

The organization of this article is described below. The basic theory of 3SAT in HNN is discussed in section two, followed by the 3SATRA method with fuzzy logic and metaheuristic algorithms in section three. Section four discusses the theory of fuzzy logic, GA and HSA. A description of the experimental setup is discussed in the following section. In section six, we display the results and discussion. Finally, the study's conclusion comes in the last section.

2. 3SAT in HNN

The SAT structure investigates whether a specific SAT formula is satisfiable once it evaluates the formula to be true. Equation (1) shows the principal formula for 3SAT:

$$\theta_{3SAT} = \wedge_{i=1}^{l} \beta_i^{(3)} \ \wedge_{i=1}^{m} \beta_i^{(2)} \wedge_{i=1}^{n} \beta_i^{(1)} \tag{1}$$

where l, m and n imply the number of clauses in third-order, second-order and first-order, respectively. Meanwhile, β_i indicates a set of a clause and the definition of β_i is as Eq (2).

$$\beta_i = \vee_{j=1}^{n} \left(x_{ij}, y_{ij}, z_{ij}\right) \tag{2}$$

Note that Eqs (3) and (4) below are the complete clauses (nc) in θ_{3SAT} and the total literal value (τ) included in θ_{3SAT} for all order clauses, respectively.

$$nc = l + m + n \tag{3}$$

$$\tau = 3 \times nc \tag{4}$$

Equation (5) indicates an example of the logical rule with three literals (ω_i) per clause.

$$\theta_{3SAT} = (\omega_1 \vee \neg\omega_2 \vee \neg\omega_3) \wedge (\omega_4 \vee \omega_5 \vee \neg\omega_6) \wedge (\omega_7 \vee \omega_8 \vee \omega_9) \tag{5}$$

If the logical rule provides true value, then $\theta_{3SAT} = 1$ is satisfied. However, a false value is identified and is not satisfied if the $\theta_{3SAT} = -1$. The expression for the neuron's activation is given in Eq (6).

$$S_i = \begin{cases} 1, & \sum_j W_{ij} S_j > \xi_i \\ -1, & \text{otherwise} \end{cases} \tag{6}$$

where ξ_i denotes the threshold value and W_{ij} signifies the weightiness of elements j to i. The HNN's synaptic weight connection is symmetrical with $W_{ij} = W_{ji}$ and there is no self-connection of $W_{ii} = W_{jj} = 0$ between any of the neurons. Each variable with neurons will be allocated to the indicated cost function $E_{\theta_{3SAT}}$ to insert into the θ_{3SAT}. The formulation is in Eq (7).

$$E_{\theta_{3SAT}} = \frac{1}{2^3} \sum_{i=1}^{nc} \left(\prod_{j=1}^{3} V_{ij} \right) \tag{7}$$

where nc is the number of clauses. Consider that the quantity of $E_{\theta_{3SAT}}$ increases in a precise ratio [8] to the total of inconsistencies in the clauses. During the retrieval phase, the HNN will update neurons iteratively using the local field from Eq (8) below:

$$h_i(t) = \sum_j W_{ij} S_j + W_i \tag{8}$$

Using Wan Abdullah's (WA) method, it is possible to determine the synaptic weight's parameters by linking the cost function and Lyapunov energy function of HNN [6]. Once the number of unfulfilled clauses rises up, $E_{\theta_{3SAT}}$ will ascent as well. V_{ij} in Eq (9) can be transferred equivalence to the related condition of the literal using the equivalent expression below:

$$V_{ij} = \begin{cases} (1 - S_X) & \text{if } \neg X_1 \\ (1 + S_X) & \text{if } \quad X_1 \end{cases} \tag{9}$$

The minor numbers of the energy function represent the neuron's stable condition [27]. According to Eq (10), the Lyapunov final energy function varies depending on the HNN-3SAT variation.

$$E = -\frac{1}{3} \sum_i \sum_j \sum_k W_{ijk} S_i S_j S_k - \frac{1}{2} \sum_i \sum_j W_{ij} S_i S_j - \sum_i W_i \tag{10}$$

3. 3SAT hybrid models

3.1. Fuzzy logic

The essential elements of a fuzzy logic system (FLS) are shown in Figure 1, including the fuzzifier stage, the rules-based inference engine and the defuzzifier stage.

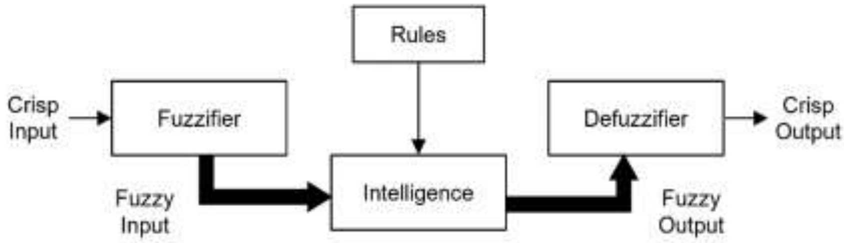

Figure 1. The key elements of a fuzzy logic system (FLS).

The suggested procedure is divided into two phases: 1) Fuzzy logic system's phase 2) Metaheuristic algorithm's phase. In the first phase, each ω_i will randomly be assigned with $\omega_i(x): X \rightarrow [0,1]$. Fuzzy logic will then be used to determine the amount of the first output $\beta_i(x): X \rightarrow [0,1]$. Fuzzification, the initial stage in a fuzzy logic system, assigns each literal to its $\tau_i = \{(\omega_i, \mu_\tau(\omega_i)) | \omega_i \in X\}$ membership function in which $0 \leq \mu_\tau(\omega_i) \leq 1$. Figure 2 depicts the design for the input and output of the fuzzy control.

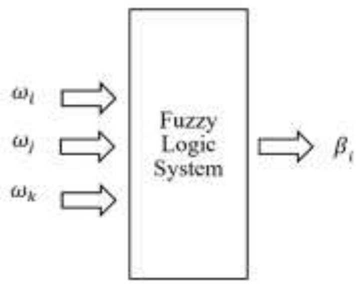

Figure 2. Fuzzy control design.

Afterward, the fuzzy rules will operate using clause block β_i, eg. $\beta_1 = \omega_1 \vee \neg\omega_2 \vee \neg\omega_3$. By enabling $\mu_\tau(\omega_i)$, we insert the Zadeh's operators to allow whichever amount in [0,1]. The logical operator AND ($\wedge$) holds the operation of $max\{\mu_\tau(\omega_i), \mu_\tau(\omega_j)\}$ and logical operator OR ($\vee$) signifies $min\{\mu_\tau(\omega_i), \mu_\tau(\omega_j)\}$. Due to this conversion, the fuzzy and binary functions (the boolean functions) are logically compatible.

In fuzzy logic, numerous membership functions can model an element's degree of membership in a fuzzy set. Membership function varies depending on application and data type. Some standard membership functions that can be used are triangular, trapezoidal, Gaussian and many more. The membership function chosen affects how a fuzzy logic system models and analyzes uncertainty and imprecision. In many practical applications, the most straightforward membership functions are most effective [28]. Previous research suggests that triangular and trapezoidal membership functions perform better in specific applications [29].

In this work, we choose triangular and trapezoidal membership functions, which are often suitable for data with clear central values and symmetric spread. The most distinct motivation for choosing triangular and trapezoidal membership functions is their simplicity and ease of interpretation [30,31].

These functions are computationally efficient because they use minimal mathematical procedures. This efficiency benefits real-time applications and structures that demand rapid decisions.

The structure of the membership function for each fuzzy logic area constructs membership functions by dividing them into right and left sides. The membership functions are outlined by four conditions: a, b, c and d. The degree of membership rises between a and b, flattens between b and c and then declines between c and d. Meanwhile, triangular functions are trapezoidal functions with b = c.

In the last part, defuzzification stages transform fuzzy values back to their crips output using the alpha-cut method [32]. The alpha-cut method will amend neuron clauses during defuzzification until the right neuron state is reached. The alpha-cut defuzzification is in Eq (11):

$$\text{alpha-cut} = \frac{\sum_i \alpha_i \overline{[\mu_{\omega_\iota}]}}{\sum_i \alpha_i} \qquad i = 1, \dots L \tag{11}$$

where L is the number of discretization stages along the vertical axis. This representing phase transforms a fuzzy value into a clear output. The estimation uses the following alpha-cut defuzzification procedure in Eqs (12) and (13).

$$\text{if } \mu_{\omega_i} \geq \alpha, \text{ then } \delta_{\omega_{i\alpha}} = 1 \tag{12}$$

$$\text{if } \mu_{\omega_i} < \alpha, \text{ then } \delta_{\omega_{i\alpha}} = 0 \tag{13}$$

3.2. GA

The first suggested method for applying the metaheuristic algorithm in the second stage is the GA. GA will be employed in the second phase after the fuzzy logic system in the learning section stage. Previous work has shown that the HNN integrated with GA in the training phase can establish optimal synaptic weight management assignments according to the mentioned logic [9,33]. The GA searches a vast variety of solutions quickly, reducing central processing unit (CPU) time during training. GA operators like crossover and mutation simplify and enhance the process, leading to better solutions. Additionally, GA's robustness in determining satisfied clauses, particularly during crossover stages, lead to accurate interpretations during training and correct states during testing.

The flow of GA involved: (1) Initialization (2) fitness evaluation (3) selection (4) crossover (5) mutation. First, the initialization of the parameters and chromosomes, Cr_i. The fitness of every chromosome Cr_i is computed by employing a fitness function in Eq (14):

$$\Phi_{Cr_i} = \sum_{i=1}^{l} \beta_i^{(3)} + \sum_{i=1}^{m} \beta_i^{(2)} + \sum_{i=1}^{n} \beta_i^{(1)} \tag{14}$$

where the situation satisfies the properties in Eq (15).

$$\beta_i^{(k)} = \begin{cases} 1, & \text{fullfilled} \\ 0, & \text{otherwise} \end{cases} \tag{15}$$

The total number of clauses in θ_{3SAT} involves in the optimum fitness Φ_{Cr_i} of each Cr_i. Eq (16) represents the objective function of GA.

$$\max\left[\Phi_{Cr_i}\right] \tag{16}$$

First selection process is done by the selection operator. Then, the genes information (β_i) of two parents (chromosome, Cr_i) is switched via crossover operators to initiate the offspring (new chromosome, Cr_i). Next, mutation is a mechanism that maintains genetic variation from one group to the next. One gene is flipped from another during the mutation process. A gene (β_i) of a specified chromosome (Cr_i) is distributed within itself by the mutation operator. With the aim of ensuring that both the outcome and a random shift mutation are applicable, the position is randomly chosen from the specified gene (β_i) for displacement. Note that the GA procedure will be recurring until the $max[\Phi_{Cr_i}]$ is accomplished. Finally, the final solution in the continuous space needs to be translated (decoded) back into the bipolar space. The flowchart for the basic GA method is indicated in Figure 3.

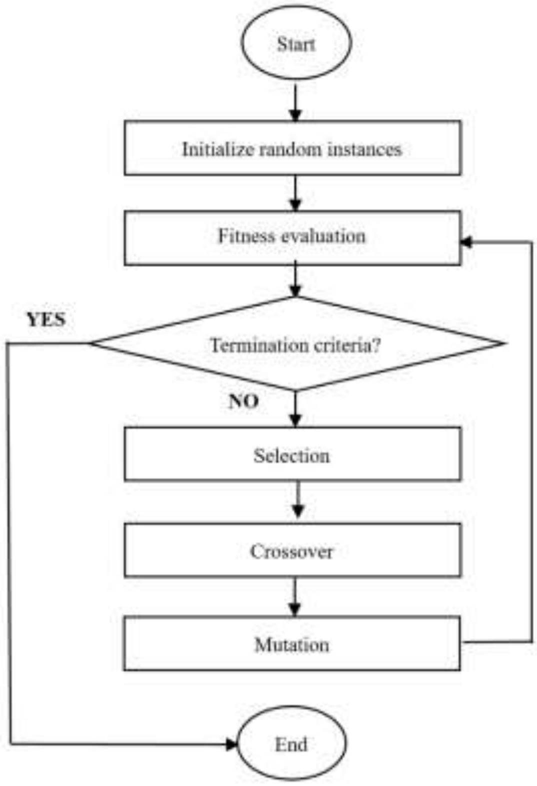

Figure 3. Flowchart of main stages in the primary GA method.

3.3. HSA

In the second suggested method, HSA is employed for the solutions to converge optimally. The flow of HSA involved: (1) Initialization of parameters (2) initialization of harmony memory (3) improvising by establishing a new harmony (4) updating harmony memory (5) termination criterion.

The first steps of HSA are initializing the algorithm variables, such as the harmony memory size (HMS), the harmony memory considering rate (HMCR) and the termination criterion $(imax)$. Then the initialisation of the harmony memory, Hm_i is set. The fitness of each harmony memory Hm_i is calculated using a fitness function in Eq (17) and the required conditions in Eq (18):

$$\Phi_{Hm_i} = \sum_{i=1}^{l} \beta_i^{(3)} + \sum_{i=1}^{m} \beta_i^{(2)} + \sum_{i=1}^{n} \beta_i^{(1)} \tag{17}$$

$$\beta_i^{(k)} = \begin{cases} 1, & \text{fullfilled} \\ 0, & \text{otherwise} \end{cases} \tag{18}$$

Equation (19) signifies the objective function of HSA where the number of clauses in θ_{3SAT} involves the optimum fitness Φ_{Hm_i} of Hm_i individually [4].

$$\max\left[\Phi_{Hm_i}\right] \tag{19}$$

Note that Eq (20) below implies the minimum fitness Φ_{Cr_i} which represents the worst harmony (wh) vector:

$$\min\left[\Phi_{Hm_i}\right] \tag{20}$$

In step three, a new harmony is created with $\eta\beta_i^k = (\beta_1', \beta_2', \ldots, \beta_i')$. Initially, it is constructed on three control parameters, namely, (1) memory consideration, (2) pitch adjusting and (3) random selection [34]. In the memory consideration stage, the new harmony vector $\eta\beta_i^k$ is constructed with the HMCR (Γ_{HMCR}). The Γ_{HMCR} is described as the possibility of choosing an element from the initialized HM members, and $1 - \Gamma_{HMCR}$ is the possibility of the random selection as in Eq (21). The operation for memory consideration is followed by the operation for pitch adjustment. Only the values selected from the HM are used for the pitch adjusting procedure. However, according to Geem [35] works, the pitch adjusting operation is not carried out since the outcomes for every choice variable are in binary space $\{0,1\}$. Hence, in this study, the pitch adjustment operator is abolished because of the usage of bipolar value $\{-1,1\}$. Furthermore, the operator is eliminated since it is the same as the improvization process [35].

$$\eta\beta_i^k = \begin{cases} \beta_i' \in \left\{hm_1^1, hm_2^2, \ldots, hm_i^{HMS}\right\}, & \text{probability } \Gamma_{HMCR} \\ \beta_i' \in \{0,1\} & \text{, probability } (1 - \Gamma_{HMCR}) \end{cases} \tag{21}$$

The fitness value of the most unsatisfied harmony vector $\min\left[\Phi_{Hm_i}\right]$ is compared to the fitness value of the newly created harmony vector $\eta\beta_i^k$. The worst harmony (wh) is switched with the recently created harmony vector if the new one has a higher objective function. If not, the newly created harmony vector is not considered. The HSA process will be recurring until the execution criterion $(imax)$ is reached. Lastly, the final solution in the continuous space needs to be translated (decoded) back into the bipolar space. The flowchart for the fundamental HSA technique is displayed in Figure 4.

Metaheuristic algorithms are categorized as non-gradient-based optimization methods. Compared to a gradient-based method, metaheuristic algorithms are considered black-box optimizers that do not require continuous or semi-continuous objective functions and gradient values to establish step size and search direction at the initial setting [36]. Metaheuristic algorithms are designed to balance exploration and exploitation since they can identify better options and help discover better solutions to escape local minima [37]. The metaheuristic algorithms are appropriate for complex and expensive computations. GA maintains a population of candidate solutions and uses genetic operators to investigate various solutions. HSA balances the exploration and exploitation to seek and develop candidate solutions in its population.

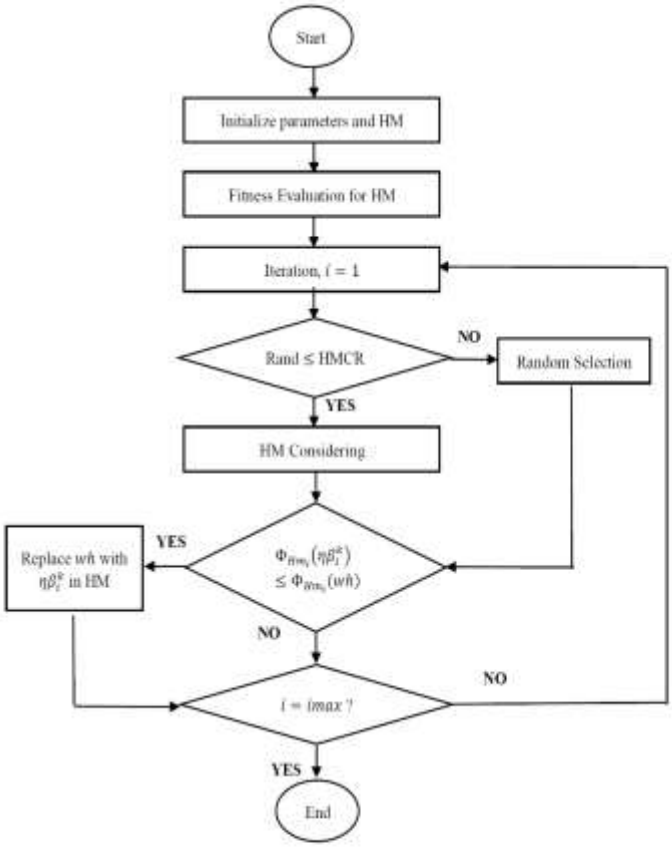

Figure 4. Flowchart of fundamental stages in the HSA method.

The following Figure 5 is the algorithm of the hybridized model of 3SAT using fuzzy logic and metaheuristic algorithms in HNN.

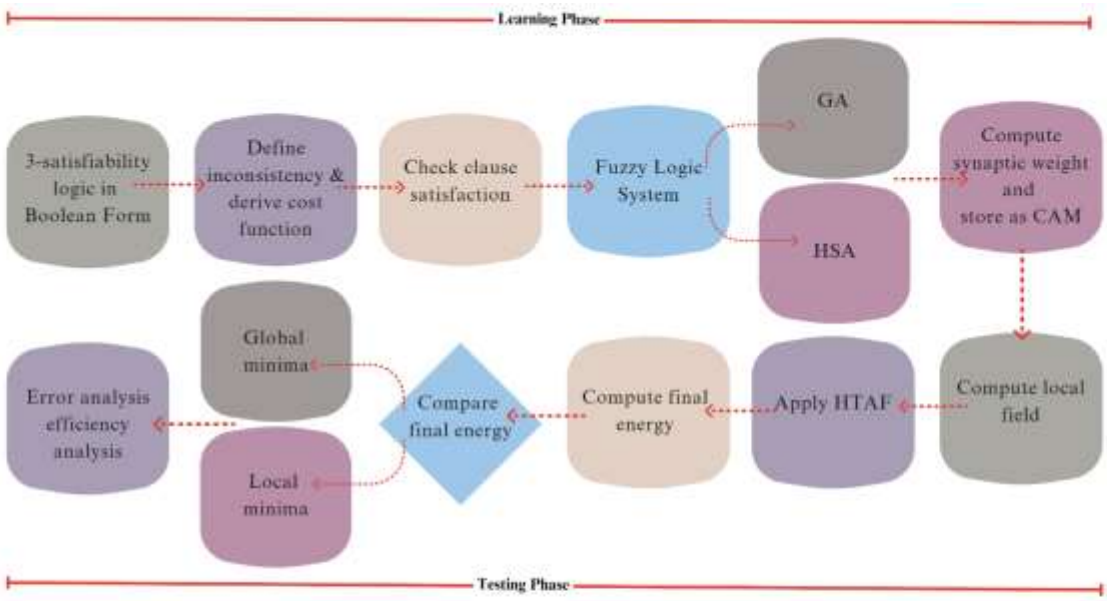

Figure 5. The algorithm of the 3SAT hybridized model in HNN.

4. 3SATRA Method

The 3SATRA method aims to obtain the finest optimal logical rule of Equation (1). Combining logic mining in 3SAT aims to extract the logical representation of the dataset's behavior. To successfully separate the logical rules that influence a dataset's outcome, a logic mining approach using 3SATRA has been presented. The efficiency of logic mining is supported by the success of the HNN-3SAT incorporating fuzzy logic, metaheuristics and 3SATRA in managing the dataset and transforming it to induced logic.

Given $S_i = (S_1, S_2, S_3, \ldots, S_m)$ with $m = 9$ where the attributes can be expressed as a 3SAT clause with a bipolar representation of $S_i = \{-1,1\}$. The WA technique in 3SATRA will disclose the degree of relationships between two elements in the dataset by calculating the linking synaptic weight. Furthermore, 3SATRA will use the WA technique throughout the learning phase to determine the possible synaptic weight between two qualities in each clause. The following steps are the implementation of 3SATRA.

Step 1:
Given a dataset for learning and testing with $m = 9$ attributes $S_i = (S_1, S_2, S_3, \ldots, S_m)$ and transform the entire dataset to a bipolar representation using the k-mean clustering strategy. The current state of S_i is described using neural network principles, where one represents TRUE and -1 represents FALSE. Note that β_i in Eq (2) is a clause for θ_{3SAT}.

Step 2:
A set of β_i that results in a positive result of the learning data $\left(\theta_i^{learn} = 1\right)$ will be separated.

Step 3:
Compute the frequency of $\theta_i^{learn} = 1$ of the set of β_i from the learning datasets. The optimum logic θ_{Best} of the datasets is obtained using Eq (22) below:

$$\theta_{Best} = max\left(\theta_i^{learn} = 1\right) \tag{22}$$

Step 4:
Obtain the cost function by using Eq (7) from section two.

Step 5:
The state of S_i that relates to $E_{\theta_{\text{Best}}} = 0$ is computed; thus, the synaptic weight is achieved by equating $E_{\theta_{\text{Best}}}$ with the final energy $H_{\theta_{\text{Best}}}$.

Step 6:
In the testing phase, the induced logic of θ_i^P is formulated using all the possible 3SAT logic based on Eq (5) and the $S_i^{induced}$ for each β_i that are constructed using the local field from Eq (8).

Step 7:
Utilizing the test data from the datasets provided, examine $\theta_i^P = \theta_i^{test}$ to choose the final induce logic θ_{best}^P.

5. Experimental setup

5.1. Datasets

This analysis uses the strategy, with 60% of the dataset as training data and the remaining 40% as testing data, based on the analysis from multiple prior works [18,38]. The data will be transformed using k-mean clustering into a bipolar representation (-1 and 1). Both the retrieval phase and the training phase will use the conversion method. In this analysis, we use the Covid-19 datasets acquired from the Ministry of Health (MoH) Malaysia from an open website [39]. We utilized line list data on Covid-19 fatalities which the Malaysian Ministry of Health made available to the general public. The datasets were retrieved on 21st September 2022. 35,914 Covid-19 fatalities were reported between 17th March 2020 and 21st July 2022. Most fatalities (8308/35,914, 23.1%) happened in people aged 60 to 69. The ratio of men to females was 1.35:1. Malaysians (88.9%) and people with comorbid conditions (78.6%) had a higher death rate. Individuals who received two or more doses for all vaccination types were considered complete and fully immunized. People who only received the first dosage or did not receive any dose were regarded as incomplete or non-immunized. Table 1 shows the elements of the Covid-19 datasets.

Table 1. Attributes for Covid-19 Dataset.

Attributes	**Name**	**Description**	**Output** $\boldsymbol{E_{\theta_{3SAT}}}$
G	Gender	Female, Male	To classify the timeframe of death for Covid-19 patients
B	BID cases	No, Yes	
M	Malaysian citizen	No, Yes	
A	Age	Age years (between 0 – 130 years old)	
S	State	State density population (between 0 – 6.6 million)	
V	Vaccinated	Not vaccinated, vaccinated	
T	Type of vaccine	Non or Singular-type, Mix-type vaccine	
N	Total vaccine	Not complete, complete	
C	Comorbidity	No, Yes	

Different characteristics are established in 3SATRA in this work. Extraction of the logical rule that describes the characteristic of the patient's death is the goal of the Covid-19 data. Data from Covid-19 will be separated into learning and testing data. In the learning dataset, {longer, shorter} will be transformed into the corresponding bipolar representations, $\{1, -1\}$. In 3SATRA, each patient's attribute will be expressed as a neuron. As a result, each clause in this dataset will be considered by a total of three neurons. To achieve the outcome, the 3SATRA comprises a group of neurons that points to θ_i^{learn} = longer time death $\left(\theta_i^{learn} = 1\right)$ or θ_i^{learn} = shorter time death $\left(\theta_i^{learn} = -1\right)$. For example, in case 1, one of the patient θ_i^{learn} is a female, is not a BID (brought in dead) case, a Malaysian citizen, aged less than 62 years old, living in a state with a population density of more than 2 million, has been vaccinated, has a single-type vaccine, has completed the vaccine doses and does not has a comorbidity. The $\theta_1^{learn} = 1$ has the following neuron interpretation as in Eq (23) below.

$$
\begin{gathered}
\text{Gender (G)} = -1 \\
\text{BID case (B)} = -1 \\
\text{Malaysian (M)} = 1 \\
\text{Age (A)} = -1 \\
\text{State } (S) = 1
\end{gathered}
\tag{23}
$$

$$\text{Vaccinated (V)} = 1$$

$$\text{Type of vaccinated (T)} = -1$$

$$\text{Total vaccine (N)} = 1$$

$$\text{Comorbidity (C)} = -1$$

The above attributes will be assigned in the 3SAT formula with another example from a different case (Case 2), as shown in Table 2. Case 1 is satisfiable since all the clause offers true value. Meanwhile, Case 2 is not satisfiable.

Table 2. Example of cases for Covid-19 datasets in 3SAT representations.

Case	Attributes	Outcome
1	$(\neg \mathrm{G} \vee \neg B \vee M) \wedge (\neg A \vee S \vee V) \wedge (\neg T \vee N \vee \neg C)$	$\theta_1^{learn} = 1$
2	$(\neg \mathrm{G} \vee \neg B \vee M) \wedge (A \vee S \vee V) \wedge (\neg T \vee \neg N \vee \neg C)$	$\theta_1^{learn} = -1$

5.2. Experimental design

The 3SATRA is trained using fuzzy logic and metaheuristic techniques using Matlab 2020b software in this exploratory modeling. Using Covid-19 datasets, the HNN algorithm used in this work generated 3SAT clauses of various levels of complexity. The final energy execution parameter has been established at 0.001 because it significantly eliminated statistical mistakes [27]. The effectiveness of this study will be assessed by contrasting all model's outcomes: 3SAT, 3SATFuzzy and 3SATFuzzyGA and 3SATFuzzyHSA, for accuracy and effectiveness. The comprehensive investigations were conducted with a similar device to eliminate the possibility of an inaccurate memory sector during simulations.

5.3. Method

We restrict the simulation analysis to the standard strategy that can derive induced logic from actual datasets. The leading objective of this proposed work is to assess the level of accuracy of the created logic developed by 3SATRA using the integration of fuzzy logic and metaheuristics. The critical parameters used by logic mining algorithms are displayed in Tables 3–6. We also select the Hyperbolic tangent activation function (HTAF) since it has the most stable activation method. The HTAF suppresses the neurons' ultimate state because of its capability and behavior, including continuous, smooth and nonlinear. HTAF is used as a post-optimization technique to accelerate the proposed algorithm. It is the most robust in Boolean satisfiability (SAT) structure, and it will minimize the logical inconsistency of logic programming and HNN [40]. Other than that, HTAF is chosen because of its capability to squash the local field to obtain the final state of the neuron. The neuron initialization is set to be random throughout the retrieval phase of the logic mining method to reduce the network's potential bias. Tables 3–6 present a list of factors included in all the methods.

Table 3. Listing of related parameters utilized in 3SAT.

Parameter	Parameter Value
Number of learning	100
Tolerance measurement	0.001
Activation function	HTAF
Synaptic weight method	Wan Abdullah (WA) method
Final neuron states, S_i	$\{-1,1\}$

Table 4. Listing of related parameters utilized in 3SATFuzzy.

Parameter	Parameter Value
Number of learning	100
Tolerance measurement	0.001
Activation function	HTAF
Fuzzy membership neuron, μ_τ	[0,1]
Input parameter, ω_i	[0,1]
Output parameter, β_i	[0,1]

Table 5. Listing of related parameters utilized in 3SATFuzzyGA.

Parameter	Parameter Value
Number of learning	100
Tolerance measurement	0.001
Activation function	HTAF
Fuzzy membership neuron, μ_τ	[0,1]
Input parameter, ω_i	[0,1]
Output parameter, β_i	[0,1]
Populations, pop	100
Selection rate, Γ_S	0.1
Crossover rate, Γ_C	0.9
Mutation rate, Γ_M	0.01

Table 6. Listing of related parameters utilized in 3SATFuzzyHSA.

Parameter	Parameter Value
Number of learning	100
Tolerance measurement	0.001
Activation function	HTAF
Fuzzy membership neuron, μ_τ	[0,1]
Input parameter, ω_i	[0,1]
Output parameter, β_i	[0,1]
Harmony memory size, HMS	100
Harmony memory considering rate, Γ_{HMCR}	0.9
Number of improvisations, $(imax)$	100

5.4. Performance metric evaluation

Several performance measures, including accuracy (δ), root mean square error (RMSE), mean absolute error (MAE) and sum squared error (SSE), will be used to estimate the proposed model's execution of logic mining.

Accuracy δ is the standard metric used to evaluate the effectiveness of classification processes. It is used to establish the correctness of the network's solutions constructed on the accomplishment of the results and the model's data-training capacity. This study's accuracy δ will represent the effective solutions generated by the induced logic for the dataset, as shown in Eq (24).

$$\delta = \frac{\text{Correct } P_{induced}}{\text{Total } P_{test}} \times 100\% \tag{24}$$

RMSE gives details on the difference of the estimated quantities and the calculated value [41]. The formula of RMSE is given in Eq (25):

$$RMSE = \sum_{i=1}^{n} \sqrt{\frac{1}{n}\left(I_{highest} - I_x\right)^2} \tag{25}$$

MAE is a helpful measurement for estimating the model by calculating the difference of the average involving the calculated and expected values [38]. The formula of MAE is provided in Eq (26):

$$MAE = \sum_{i=1}^{n} \frac{1}{n}\left|I_{highest} - I_x\right| \tag{26}$$

A statistical method for estimating the extent to which the information deviates from predicted values is called SSE as shown in Eq (27) below [42]:

$$SSE = \sum_{i=1}^{n}\left(I_{highest} - I_x\right)^2 \tag{27}$$

Processing time (τ) indicates the total interval of time to finish the process. The processing time is used to determine robustness and stability. This investigation will employ the second international system of units (SI) for processing time. When the model's CPU period is reduced, the simulation's productivity is believed to be improved. The calculation is provided in Eq (28):

$$\tau = \text{Training phase time} + \text{Testing phase time} \tag{28}$$

In prior work, global minima (zM) are used to thoroughly explore energy analysis [43]. The formula is given in Eq (29):

$$zM = \frac{1}{NT.COMBMAX} \sum_{i=1}^{n} N \tag{29}$$

6. Results and discussion

6.1. Simulated datasets

Before investigating the Covid-19 datasets, the experiment's initial section used simulated datasets to test the methods. Simulated datasets is data that was randomly generated by a computer.

This section compares the effectiveness of HNN-3SAT, HNN-3SATFuzzy, HNN-3SATFuzzyGA and HNN-3SATFuzzyHSA as the learning rule in the HNN model. The following presents the simulated dataset results for all models.

Figures 6 and 7 show the global minima ratio and computational time obtained by four methods: HNN-3SAT, HNN-3SATFuzzy, HNN-3SATFuzzyGA and HNN-3SATFuzzyHSA, respectively. Previous work by [27] learned that there is a correlation between the global minima value and the input of energy achieved in the network's final stage. Based on Figure 6, the results in the presented methods achieve higher global minima energy, except for HNN-3SAT. The new integrated network of HNN-3SATFuzzy, HNN-3SATFuzzyGA and HNN-3SATFuzzyHSA have reached the global minima ratio with the value of one. The suggested method can tolerate a wider range of neuron values since fuzzy logic reduces the computing burden by expanding the neuron search space to determine the correct states. The fuzzy logic technique involves two central parts: Fuzzification and defuzzification. The alpha-cut technique will also enhance the unfulfilled neuron clauses in the defuzzification stage until the correct neuron state is discovered. This hybrid property allows the algorithms to attain effective global minima. However, because the HNN-3SAT network lacks an accelerator technique to expand the search space; it gets trapped in a suboptimal state as the number of neurons increases. Thus, the network potential decelerates as the number of neurons rises and is caught in local minima. In conclusion, the property of applying fuzzy logic and/or metaheuristic algorithms yields its potential to converge effectively to global minima compared to HNN-3SAT.

A crucial parameter or indicator for assessing the efficacy of our proposed method is the processing time. From the efficiency of the overall calculation process, one may roughly deduce the durability of our techniques. The processing time (τ), is the overall time for the network to finish the computation process. The computation technique comprises the main phase of the network, training and testing phases. Figure 7 indicates the processing time (τ) for the HNN-3SAT, HNN-3SATFuzzy, HNN-3SATFuzzyGA and HNN-3SATFuzzyHSA. The more neurons there were, the more likely it was that the same neuron was implicated in additional sentences [44]. As shown in Figure 7, the value of τ increases with the number of neurons in all hybrid networks. However, the network for HNN-3SAT was more prone to get caught in local minima and take up additional time as it got trapped and became more complicated. Now that the logical inconsistencies have been resolved, a thorough search in the HNN will determine the best selection. A method that accelerates the training process is therefore necessary. Figure 7 demonstrates how superior HNN-3SATFuzzyHSA is as the value of τ remains low compared to other methods. Although there were more clauses per neuron, the HNN-3SATFuzzyHSA model managed to progress faster than the other network. Furthermore, the time spent by all networks at a lower number of neurons is not significantly different. However, the difference can be seen when the complexity increases. The results show that the processing time was shorter when the hybrid method of applying fuzzy logic and metaheuristic algorithms was employed. Thus, it proved that the hybridized system works efficiently as the complexity increases.

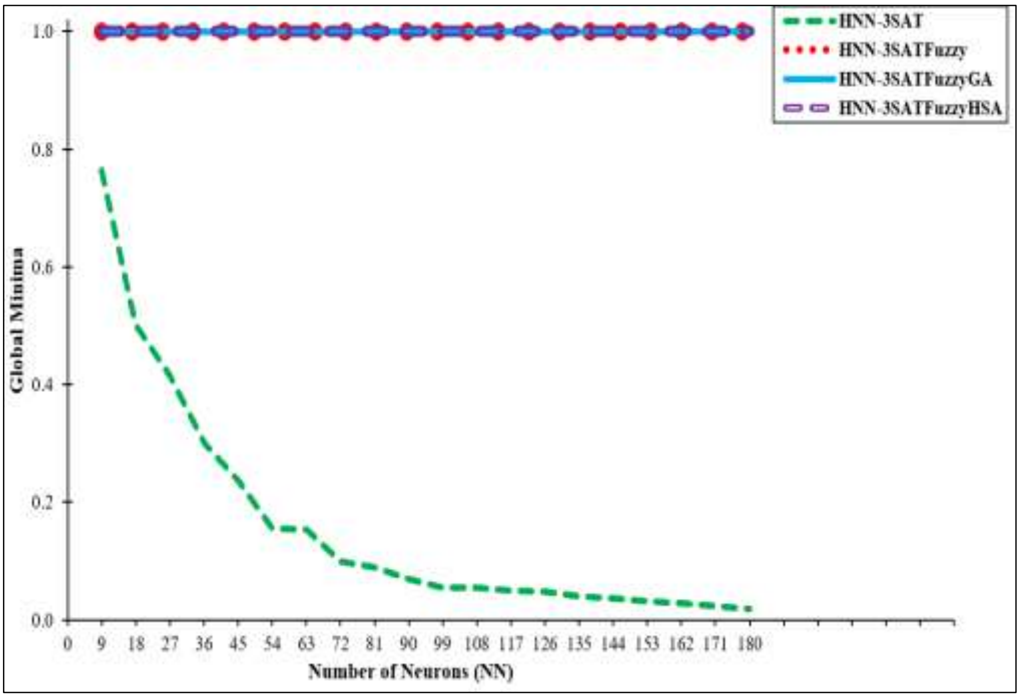

Figure 6. Global minima ratio for all models.

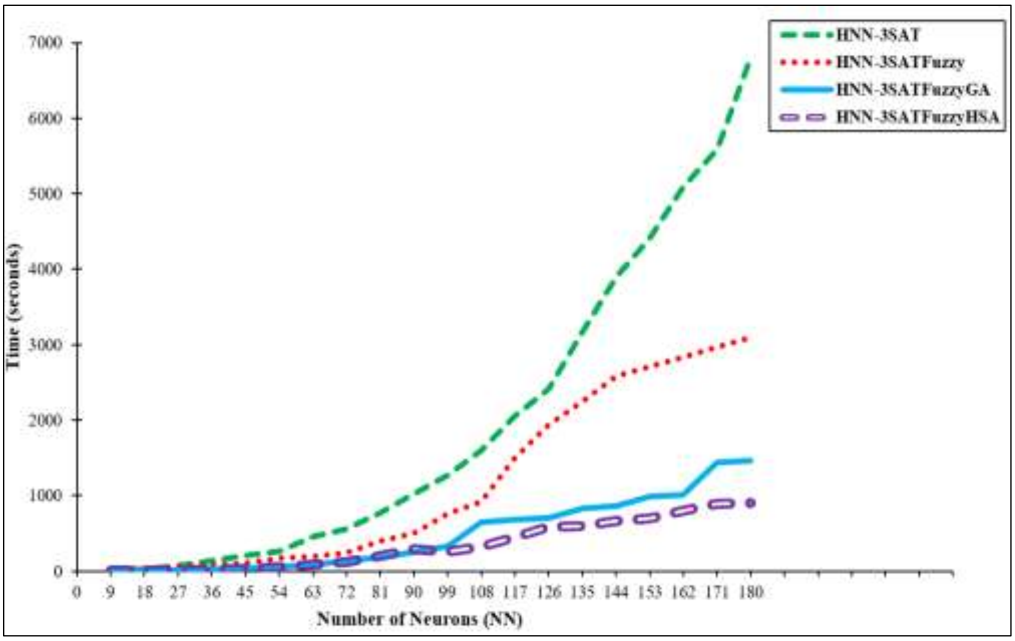

Figure 7. Processing time (τ) for all models.

Figures 8–10 display the RMSE, MAE and SSE results during the training stage for all the approaches using simulated datasets. According to all the metrics, HNN-3SATFuzzyHSA outperformed all the methods. The results show that the RMSE, MAE and SSE scores for the HNN-3SATFuzzyHSA network are superior to other networks considering a rise in the number of neurons (NN). The HNN-3SATFuzzyHSA solutions diverged less from the possible solutions. Based on RMSE, MAE and SSE results, the recommended strategy, HNN-3SATFuzzyHSA, achieves $E_{\theta_{3SAT}} = 0$ at lower results compared to the rest. The primary reason is that 3SAT's fuzzy logic technique expands the search space better and HSA provides better optimization, permitting the success of $E_{\theta_{3SAT}} = 0$ in smaller series. While training for precise interpretations, the fuzzy logic system process is prone to expand the search space. Similarly, the HNN-3SATFuzzyHSA employed a methodical approach throughout the searching neuron step, employing the fuzzification and defuzzification procedures. The integration process with metaheuristic algorithms also helps the network to optimize more efficiently. HNN-3SATFuzzyHSA could effectively check the correct interpretation and manage further constraints than the other networks.

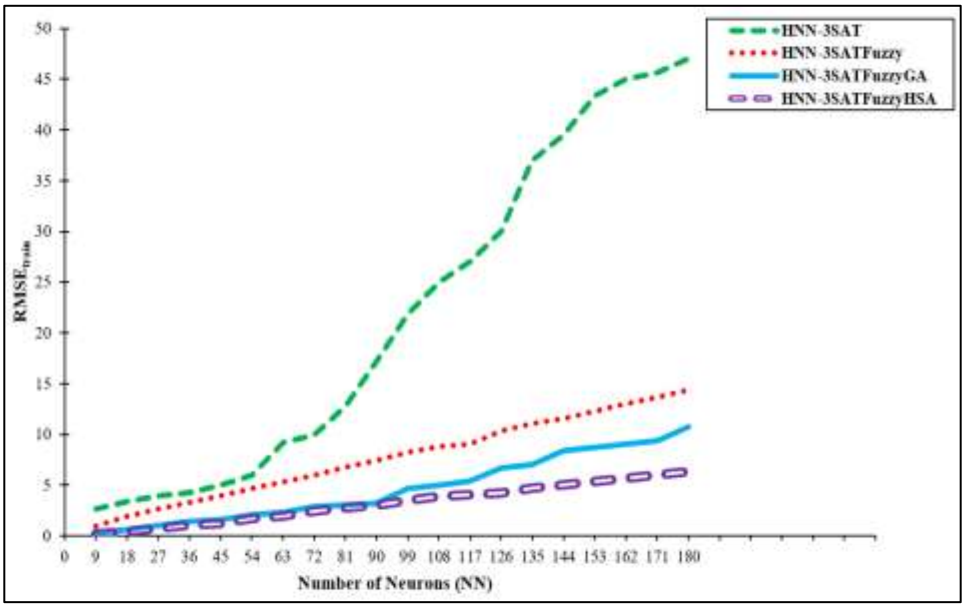

Figure 8. Values of RMSE for all models in simulated datasets.

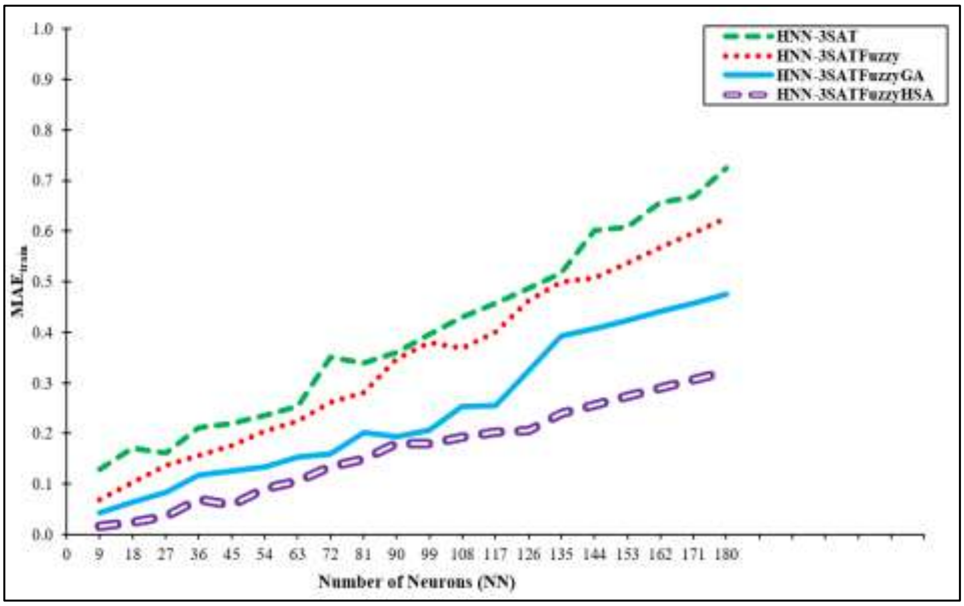

Figure 9. Values of MAE for all models in simulated datasets.

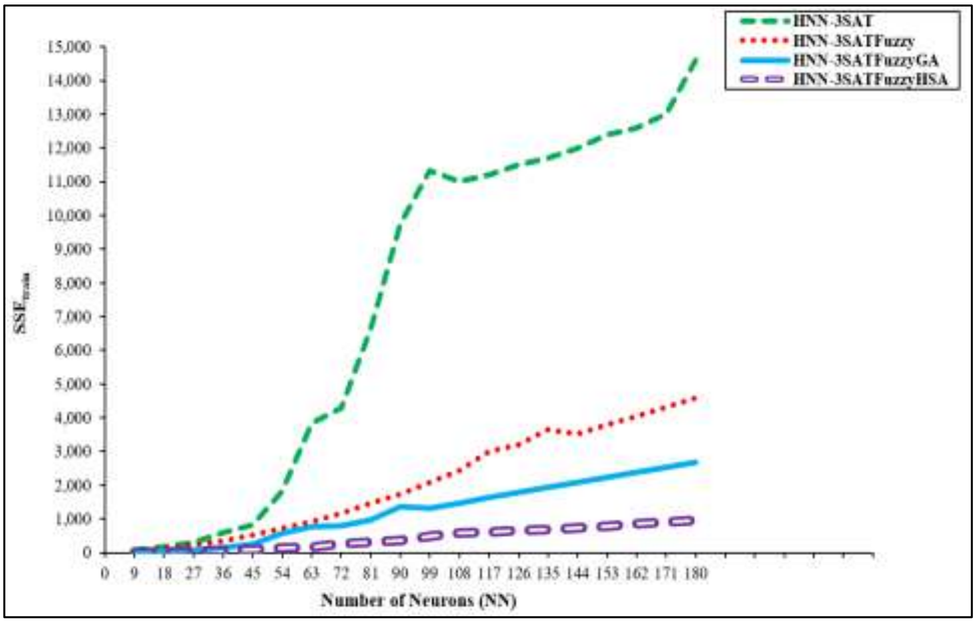

Figure 10. Values of SSE for all models in simulated datasets.

6.2. Covid-19 datasets

In this part, we assess the induced logic θ_i^P performance, and the instances of the datasets will be split into θ_i^{learn} (60%) and θ_i^{test} (40%) groups. It is significant to consider that this approach strictly implies three literals per clause. The data has effectively developed an adaptable 3SAT logic and has been successfully included in the network. Five performance evaluations were examined to assess the effectiveness and accuracy of HNN-3SAT incorporating fuzzy logic and the metaheuristic algorithm in executing 3SATRA. The metrics used are accuracy, CPU time, RMSE, MAE and SSE. Table 7 shows the accuracy δ and CPU time τ for all the models assessed in the Covid-19 datasets. Table 8 shows all the error analyses of RMSE, MAE and SSE findings. The highest value of δ is used to determine which model is the best based on the accuracy; meanwhile, the lowest τ regulates the robustness of the network. The lowest error analysis indicates that the method improves the best quality of the solutions.

We illustrate the accuracy and robustness of all models' retrieval properties in Table 7. According to the value of δ of each model in Table 7, the implementation of 3SATRA in the proposed model of HNN-3SATFuzzyHSA achieved a 98.34% positive outcome, 97.79% for HNN-3SATFuzzyGA and 78.24% for HNN-3SATFuzzy. The outcomes surpass the conventional HNN-3SAT model with only 48.53% accuracy. The value achieved by the most outstanding accuracy from HNN-3SATFuzzyHSA is relatively excellent by at most 49.8% compared to the lowest accuracy from HNN-3SAT. The existing HNN-3SAT logic mining work cannot produce at least 50% accuracy for Covid-19 datasets because it has no hybridized method or optimizer function that can widen the search space and improve the solution. In conclusion, HNN-3SATFuzzyHSA combined with 3SATRA is the most effective model for training and evaluating HNN since it has the maximum logical rule accuracy. The value of τ reported for the processing time for all models for Covid-19 datasets. From Table 7, HNN-3SAT required higher consumption time since the basic mechanism has no hybridized method that has the effect of causing the logical rules to fail when any of the clauses is not met. Therefore, the process causes the HNN-3SAT method to utilize more time and additional iterations to find a reliable solution. Contrarily, HNN-3SATFuzzyHSA permits the network to finish the process with the fewest iterations possible in the least time.

Table 7. Accuracy and CPU time for all models in Covid-19 dataset.

Models	δ	τ
3SAT	48.53	69.1599
3SATFuzzy	78.24	54.9863
3SATFuzzyGA	97.79	14.9878
3SATFuzzyHSA	98.34	12.3535

The error analysis for Covid-19 datasets for the four different approaches—HNN-3SAT, HNN-3SATFuzzy, HNN-3SATFuzzyGA and HNN-3SATFuzzyHSA are listed in Table 8 for RMSE, MAE and SSE, respectively. Based on Table 8, the integration of fuzzy logic and HSA mechanism enhances the quality of solutions for HNN-3SATFuzzyHSA, resulting in the least errors and enabling $E_{\theta_{3SAT}} = 0$. Meanwhile, the HNN-3SAT creates a higher value of errors with the highest error values. Table 8 shows that all error analyses for HNN-3SATFuzzyHSA performed better than those of other networks. The findings show that the HNN-3SATFuzzyHSA network has the lowest RMSE, MAE and SSE values compared to other methods. It proves the hybridized network between fuzzy logic and HSA performs positively. Less variation existed between the HNN-3SATFuzzyHSA solutions and the possible solutions. RMSE, MAE and SSE values revealed that the suggested technique, HNN-3SATFuzzyHSA, managed to achieve $E_{\theta_{3SAT}}$ with the lowest results. The main reason for this is that HSA and fuzzy logic find a better solution during the training phase, allowing $E_{\theta_{3SAT}} = 0$ to be reached in a smaller amount of series. When training, the fuzzy logic system has the tendency to widen its search when seeking precise interpretations. The HNN-3SATFuzzyHSA then systematically used the defuzzification process to gain satisfying solutions. Utilizing HSA additionally improves solution optimization. Furthermore, HNN-3SATFuzzyHSA could handle more limitations than the other networks and efficiently verify the correct interpretation.

$$\theta_{best}^{P} = (G \vee \neg B \vee M) \wedge (A \vee S \vee V) \wedge (\neg T \vee N \vee C) \tag{30}$$

Table 8. Values of error analysis for all models in Covid-19 datasets.

Models	RMSE	MAE	SSE
3SAT	26.9124	112.4850	703.1300
3SATFuzzy	21.0364	14.8750	31.0100
3SATFuzzyGA	1.5569	0.3930	16.4000
3SATFuzzyHSA	0.5796	0.0587	9.7800

Equation (30) displays the achieved θ_{best}^{P} by HNN-3SATFuzzyHSA. The generated θ_{best}^{P} will help the responsible sectors recognize a vital cause to identify the critical patients needing more attention. This work focused on HNN-3SATFuzzyHSA on transforming Covid-19 into 3SAT logic representation with 3SATRA to produce optimum θ_{best}^{P} to extract valuable information. Applying HNN-3SATFuzzyHSA in 3SATRA will offer valuable understanding to any industry requiring substantial components that may lead to better services. The new developments in improving hybrid models between fuzzy logic and metaheuristic algorithms are constantly required to supervise optimization tasks.

7. Conclusions

In conclusion, we anticipate that the study's findings will widen the scope of the fundamental optimization method in the HNN. This study demonstrates that 3SATRA is an effective information extraction method and behavior for estimating the outcomes of Covid-19 datasets. Besides its vast entries of datasets, the hybridized 3SATRA created in this study, manages to extract valuable information from the datasets. Therefore, 3SATRA is essential to produce induced logic that reveals negligible elements that contribute to the issues faced by the world today. Additionally, it was discovered that developing our modified HNN-3SAT model combined with fuzzy logic and two metaheuristic algorithms helped enhance the conventional learning phase of HNN. We tested our hybrid HNN model of HNN-3SATFuzzyHSA using Covid-19 datasets in contrast to other approaches. Various performance evaluation indicators were used to carry out the comparative analysis. The results showed that HNN-3SATFuzzyHSA performed better than other approaches. The experimental model to perform the fuzzy logic and metaheuristic strategies for training the 3SATRA can be used in other medical fields' datasets, such as diabetes, hepatitis and heart failure, to extract and reveal the behavior of the datasets.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgements

This research was supported by the Ministry of Higher Education Malaysia (MOHE) through the Fundamental Research Grant Scheme (FRGS), FRGS/1/2022/STG06/USM/02/11 and Universiti Sains Malaysia.

Conflict of interest

All authors declare no conflicts of interest in this paper. All authors approved the version of the manuscript to be published.

References

1. Z. Mustaffa, M. H. Sulaiman, Stock price predictive analysis: An application of hybrid Barnacles Mating Optimizer with Artificial Neural Network, *Int. J. Cognitive Comput. Eng.*, **4** (2023), 109–117. https://doi.org/10.1016/j.ijcce.2023.03.003
2. K. Linka, E. Kuhl, A new family of Constitutive Artificial Neural Networks towards automated model discovery, *Comput. Method. Appl. M.*, **403** (2023), 115731. https://doi.org/10.1016/j.cma.2022.115731

3. Y. Li, Application analysis of artificial intelligent neural network based on intelligent diagnosis, *Procedia Comput. Sci.*, **208** (2022), 31–35. https://doi.org/10.1016/j.procs.2022.10.006
4. S. Sathasivam, M. A. Mansor, M. S. M. Kasihmuddin, H. Abubakar, Election algorithm for random k satisfiability in the hopfield neural network, *Processes*, **8** (2020), https://doi.org/10.3390/PR8050568
5. J. J. Hopfield, D. W. Tank, "Neural" computation of decisions in optimization problems, *Biol. Cybern.*, **52** (1985), 141–152. https://doi.org/10.1007/BF00339943
6. W. A. T. W. Abdullah, Logic programming on a neural network, *Int. J. Intell. Syst.*, **7** (1992), 513–519. https://doi.org/10.1002/int.4550070604
7. W. A. T. W. Abdullah, The logic of neural networks, *Phys. Lett. A*, **176** (1993), 202–206. https://doi.org/10.1016/0375-9601(93)91035-4
8. S. Sathasivam, M. A. Mansor, A. I. M. D. Ismail, S. Z. M. Jamaludin, M. S. M. Kasihmuddin, M. Mamat, Novel random k Satisfiability for k ≤ 2 in Hopfield Neural Network, *Sains Malays.*, **49** (2020), 2847–2857. https://doi.org/10.17576/jsm-2020-4911-23
9. F. L. Azizan, S. Sathasivam, M. K. M. Ali, N. Roslan, C. Feng, Hybridised network of fuzzylogic and a genetic algorithm in solving 3-satisfiability hopfield neural networks, *Axioms*, **12** (2023), 250. https://doi.org/10.3390/axioms12030250
10. J. Sun, S. Sathasivam, M. K. M. Ali, Analysis and optimization of network properties for bionic topology hopfield neural network using Gaussian-Distributed Small-World rewiring method, *IEEE Access*, **10** (2022), 95369–95389. https://doi.org/10.1109/ACCESS.2022.3204821
11. F. Adolfi, J. S. Bowers, D. Poeppel, Successes and critical failures of neural networks in capturing human-like speech recognition, *Neural Networks*, **162** (2023), 199–211. https://doi.org/10.1016/j.neunet.2023.02.032
12. F. Martínez, F. Charte, M. P. Frías, A. M. Martínez-Rodríguez, Strategies for time series forecasting with generalized regression neural networks, *Neurocomputing*, **491** (2022), 509–521. https://doi.org/10.1016/j.neucom.2021.12.028
13. M. Mirbod, M. Shoar, Intelligent concrete surface cracks detection using computer vision, pattern recognition, and artificial neural networks, *Procedia Comput. Sci.*, **217** (2023), 52–61. https://doi.org/10.1016/j.procs.2022.12.201
14. C. H. Hoffmann, Is AI intelligent? An assessment of artificial intelligence, 70 years after Turing, *Technol. Soc.*, **68** (2022), 101893. https://doi.org/10.1016/j.techsoc.2022.101893
15. L. A. Zadeh, A fuzzy-algorithmic approach to the definition of complex or imprecise concepts, *Int. J. Man-Machine Studies*, **8** (1976), 249–291.
16. J. J. Buckley, Y. Hayashi, Fuzzy neural networks: A survey, *Fuzzy Set. Syst.*, **66** (1994), 13.
17. S. C. Lee, E. T. Lee, Fuzzy sets and neural networks, *J. Cyb.*, **4** (1974), 83–103. https://doi.org/10.1080/01969727408546068
18. A. Alway, N. E. Zamri, M. S. M. Kasihmuddin, M. A. Mansor, S. Sathasivam, Palm oil trend analysis via logic mining with discrete hopfield neural network, *Pertanika J. Sci. Technol.*, **28** (2020), 967–981.
19. S. Sathasivam, M. Mamat, M. S. M. Kasihmuddin, M. A. Mansor, Metaheuristics approach for maximum k satisfiability in restricted neural symbolic integration, *Pertanika J. Sci. Technol.*, **28** (2020), 545–564.
20. S. Sathasivam, W. A. T. W. Abdullah, Logic mining in neural network: reverse analysis method, *Computing*, **91** (2011), 119–133. https://doi.org/10.1007/s00607-010-0117-9

21. M. Ciotti, M. Ciccozzi, A. Terrinoni, W. C. Jiang, C. B. Wang, S. Bernardini, The COVID-19 pandemic, *Cri. Rev. Cl. Lab. Sci.*, **57** (2020), 365–388. https://doi.org/10.1080/10408363.2020.1783198
22. *World Health Organization*, Coronavirus disease 2019 (COVID-19) Situation Report – 97, 2020. Available from: https://apps.who.int/iris/handle/10665/331907
23. *World Health Organization,* COVID-19 Weekly Epidemiological Update, 2023. Available from: https://www.who.int/docs/default-source/coronaviruse/situation-reports/20230518_weekly_epi_update_143.pdf
24. *World Health Organization*, Guidance for surveillance of SARS-CoV-2 variants Interim guidance, 2021. Available from: https://apps.who.int/iris/rest/bitstreams/1361901/retrieve
25. *World Health Organization*, Enhancing response to Omicron SARS-CoV-2 variant: Technical brief and priority actions for Member States, 2022. Available from: https://www.who.int/docs/default-source/coronaviruse/2022-01-21-global-technical-brief-and-priority-action-on-omicron-sars-cov-2-variant.pdf
26. J. P. A. Ioannidis, F. Zonta, M. Levitt,. Estimates of COVID-19 deaths in Mainland China after abandoning zero COVID policy, *Eur. J. Clin. Invest.*, **53** (2023). https://doi.org/10.1111/eci.13956
27. S. Sathasivam, Upgrading Logic Programming in Hopfield Network, *Sains Malaysiana*, **39** (2010), 115–118.
28. O. Kosheleva, V. Kreinovich, Why Triangular Membership Functions are Often Efficient in F-transform Applications: Relation to Probabilistic and Interval Uncertainty and to Haar Wavelets, *Communications in Computer and Information Science*, Springer Verlag, (2018), 127–138. https://doi.org. 10.1007/978-3-319-91476-3_11.
29. S. H. Khairuddin, M. H. Hasan, M. A. Hashmani, M. H. Azam, Generating Clustering-Based Interval Fuzzy Type-2 Triangular and Trapezoidal Membership Functions: A Structured Literature Review, *Symmetry*, **13**, (2021), https://doi.org.10.3390/sym13020239
30. J. Dombi, P. R. Rigó, The construction of multidimensional membership functions and its application to feasibility problems, *Fuzzy Set. Syst*, **469**, https://doi.org.10.1016/j.fss.2023.108634
31. V. Kreinovich, O. Kosheleva, S. N. Shahbazova, Why Triangular and Trapezoid Membership Functions: A Simple Explanation, (2020), 25–31. https://doi.org.10.1007/978-3-030-38893-5_2
32. A. Pourabdollah, J. M. Mendel, R. I. John, Alpha-cut representation used for defuzzification in rule-based systems, *Fuzzy Set. Syst.*, **399** (2020), 110–132. https://doi.org/10.1016/j.fss.2020.05.008
33. N. E. Zamri, S. A. Azhar, S. S. M. Sidik, M. A. Mansor, M. S. M. Kasihmuddin, S. P. A. Pakruddin, et al., Multi-discrete genetic algorithm in hopfield neural network with weighted random k satisfiability, *Neural Comput. Appl.*, **34**, 19283–19311, https://doi.org/10.1007/s00521-022-07541-6
34. Z. W. Geem, J. H. Kim, G. V. Loganathan, A New Heuristic Optimization Algorithm: Harmony Search, *Simulation*, **76** (2001), 60–68. https://doi.org/10.1177/003754970107600201
35. Z. W. Geem, Harmony Search in Water Pump Switching Problem, In *Lecture Notes in Computer Science*, Springer, (2005), 751–760. https://doi.org/10.1007/11539902_92
36. A. Mortazavi, Bayesian Interactive Search Algorithm: A New Probabilistic Swarm Intelligence Tested on Mathematical and Structural Optimization Problems, *Adv. Eng. Softw.*, **155**, (2021), https://doi.org/10.1016/j.advengsoft.2021.102994
37. N. A. Zainal, S. Azad, K. Z. Zamli, An adaptive fuzzy symbiotic organisms search algorithm and its applications, *IEEE Access*, **8**, 225384–225406, (2020), https://doi.org/10.1109/ACCESS.2020.3042196

38. S. A. S. Alzaeemi, S. Sathasivam, Examining the forecasting movement of palm oil price using RBFNN-2SATRA metaheuristic algorithms for logic mining, *IEEE Access*, **9** (2021), 22542–22557. https://doi.org/10.1109/ACCESS.2021.3054816
39. *Ministry of Health Malaysia,* Open data on COVID-19 in Malaysia, 2022. Available from: https://github.com/MoH-Malaysia/covid19-public
40. M. A. Mansor, S. Sathasivam, Accelerating Activation Function for 3-satisfiability Logic Programming, *Int. J. Intell. Syst. Appl.*, **8** (2016), 44–50, https://doi.org/10.5815/ijisa.2016.10.05
41. C. J. Willmott, K. Matsuura, Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance, *Clim. Res.*, **30** (2005), 79–82. https://doi.org/10.3354/cr030079
42. M. Bilal, S. Masud, S. Athar, FPGA design for statistics-inspired approximate sum-of-squared-error computation in multimedia applications, *IEEE T. Circuits-II*, **59** (2012), 506–510. https://doi.org/10.1109/TCSII.2012.2204841
43. S. A. S. Alzaeemi, S. Sathasivam, M. Velavan, Agent-based modeling in doing logic programming in fuzzy hopfield neural network, *Int. J. Modern Edu. Comput. Sci.*, **13** (2021), 23–32. https://doi.org/10.5815/IJMECS.2021.02.03
44. S. Sathasivam, W. A. T. W. Abdullah, Logic Learning in Hopfield Networks, *Modern Appl. Sci.*, **2** (2008), 57–63. https://doi.org/10.5539/mas.v2n3p57