S-Type Random $k$ Satisfiability Logic in Discrete Hopfield Neural Network Using Probability Distribution: Performance Optimization and Analysis

Suad Abdeen $^{1,2}$, Mohd Shareduwan Mohd Kasihmuddin $^{1,*}$, Nur Ezlin Zamri $^3$, Gaeithry Manoharam $^1$, Mohd. Asyraf Mansor $^3$ and Nada Alshehri $^2$

1. Introduction

A Discrete Hopfield Neural Network (DHNN) is a significant type of Artificial Neural Network (ANN) that employs a learning model based on association features formulated by Hopfield and Tank [1]. ANNs have long been used as a mathematical method with which to solve a range of issues [2–8]. DHNN is a recurrent ANN with feedforward connections that comprise interconnected neurons in which every neuron output is fed back into every neuron input. Neurons are stored in either a binary or bipolar form in the input and output neurons of the DHNN structure [9]. Further, to approximate optimization solutions for problems, the structures of DHNN have been extensively modified. This network has many interesting behaviors. Fault tolerance is also a feature of the Content Addressable Memory (CAM) technique, which has an infinite capacity for pattern storage and is useful for its
converging iterative process [10]. Numerous applications have made use of DHNNs, including optimization problems [1], clinical diagnosis [11–13], the electric power sector [14], the investment sector [15], location detectors [16], and others. Despite the importance of using the intelligent decision systems of the DHNN to solve optimization problems, it is necessary to implement the symbolic rule to guarantee that the DHNN always converges to the ideal solution, because recent studies failed to conduct a thorough analysis of a DHNN based on neural connections. This issue was solved by Wan Abdullah [17], who suggested a logical rule for ANNs by associating each neuron’s connection with a true or plausible interpretation.

The Wan Abdullah approach is a novel approach, and it is interesting to note that the synaptic weight is determined by matching the logic cost function and the Lyapunov energy function. This approach led to better performance than traditional teaching techniques such as Hebbian learning with respect to obtaining the synaptic weight during the training phase. A more specific logical rule has been developed since the logical rule was first introduced in the original DHNN. Sathasivam [18] decided to expand the work of Wan Abdullah and proposed Horn Satisfiability (HORNSAT) as a new Satisfiability (SAT) concept. This study introduced the Sathasivam method of relaxation to improve the finalized state of neurons. This proposal demonstrates the strong capabilities of the HORNSAT in terms of reaching the absolute minimum amount of energy. The outcome demonstrates that logical rules can be included in DHNNs. Nevertheless, because DHNNs relax too quickly and offer fewer possibilities for neurons to interchange information, more local minimum solutions result, which makes it difficult to understand how different logical rules affect DHNNs. This motivated the emergence of a new era of research with different perspectives, beginning with Kasihmuddin et al. [9], who introduced systematic $k$ Satisfiability ($k$SAT) for $k = 2$, 2 Satisfiability (2SAT). With each clause containing two literals and all clauses joined by a disjunction, the implementation of 2SAT in a DHNN was reported to achieve a high global minima ratio while keeping computational time to a minimum. Subsequently, Mansor et al. [19] continued the research by proposing a high degree of order of $k$SAT for $k = 3$, namely, 3 Satisfiability (3SAT), in a DHNN. With each clause containing three literals and all clauses joined by a disjunction, the proposed 3SAT in a DHNN increases the storage capacity of a network because each neuron’s number of local minimum solutions tends to be low. Despite the success of the implementation of systematic logic in DHNNs, this approach lacks control with respect to distributing the number of negative literals as well as regarding a variety of clauses. Furthermore, as the number of such neurons increases, the efficiency of the training phase in the DHNN decreases. During the testing phase of DHNNs, there is less neuronal variation. Sathasivam et al. [20] clarified that the rigidity of the logical structure contributes to overfitting solutions in DHNNs. When the number of neurons is large, the restricted number of literals per clause results in suboptimal synaptic weight values, thereby decreasing the likelihood of locating diverse global minima solutions. The necessity of variance in the recovered solutions ensures that the search space is well-explored. Further stated by [21], DHNNs are still vulnerable to various challenges, including a lack of generality as a result of non-flexible logical rules and a strict logic structure, despite the fact that the accuracy of research acquired from the real-world dataset has been satisfactory.

Due to the need for a different logical clause set that contributes to the degree of connection between the logical formulae, Sathasivam et al. [20] proposed a non-systematic SAT called Random $k$ Satisfiability (RAN$k$SAT) by using first-order and second-order logic 2SAT in conjunction, where $k = 1, 2$; Random 2 Satisfiability (RAN2SAT); and all clauses are connected by disjunction. RAN2SAT introduces a flexible logic structure that contributes to the generation of more logical inconsistency, which expands the diversity of synaptic weights. The proposed RAN2SAT in a DHNN achieved about 90% of the global minima ratio with fewer neurons. Due to the necessity of increasing the storage capacity of RAN2SAT and dealing with the absence of interpretation in a typical systematic satisfiability logic and limited $k \leq 2$, Karim et al. [22] were inspired to resolve this problem
and thus proposed a flexible logic structure that increases storage capacity by incorporating third-order clauses into the formulation. Random 3 Satisfiability (RAN3SAT) suggests three logical \((k = 1, 3; k = 2, 3; \text{ and } k = 1, 2, 3)\) literal structures per clause, and for all clauses to be joined by a disjunction. This increases the capacity of the DHNN to recover neuronal states based on different logical orders, which can lead to a variety of convergent interpretations of global minimum solutions. Both RANKSAT types experience difficulty regarding the selection system in terms of the composition represented by the first, second, and third logical formulations, which is still poorly defined. Thus, the combination of correct interpretations is restricted to the number of \(k\)-order clauses with a predefined term assigned in the logical formula.

Another fascinating study on non-systematic logic with a different perspective was introduced by Alway et al. [23]; this solution increases the representation of 2SAT compared to 3SAT clauses in non-systematic SAT logic through an assigned 2SAT ratio \((r^*)\) in DHNN in order to decrease the duplication of final neuron state patterns. The proposed Major 2 Satisfiability (MAJ2SAT) in the DHNN successfully provides more neuronal variation. Zamri et al. [24] introduced Weighted Random \(k\) Satisfiability \((r^kSAT)\) as a non-systematic method with a proposed logical structure that ideally produces the proper \(r^kSAT\) logical structure using a Genetic Algorithm (GA) by taking into account the desired proportion of negative literals \((r)\). Another method introduced by Sidik et al. [25] consisted of altering the \(r^kSAT\) logic phase by adding a binary Artificial Bee Colony algorithm to guarantee that negative literals are distributed properly. The proposed \(r^kSAT\) in a DHNN with a weighted ratio of negative literals leads to a significant global minima ratio. Nonetheless, despite this significant advancement in controlling the logical structure of selecting clauses and using a metaheuristic approach to distribute the number of negative literals, these techniques fail to account for the representation of the probability distribution of the dataset in the selection system.

Unique, flexible logical systems were formed by combining systematic and non-systematic approaches with a unique perspective. This approach leads to a great potential for solution diversity as it randomly generates a number of clauses. Guo et al. [26] proposed Y-Type Random 2 Satisfiability (YRAN2SAT), in which a number is randomly assigned to the first-order and second-order clauses, while further final states can be retrieved by YRAN2SAT in a DHNN with the minimum global energy. With high order logic, Gao et al. [27] proposed a G-Type Random \(k\) Satisfiability (GRAN3SAT) system, in which a set of clauses of first, second, and third orders is randomly generated. In a DHNN, GRAN3SAT can exhibit a larger storage capacity and is capable of investigating complex dimensional issues. Despite this success, its system of selection still has a flaw: there is no clear system with which to control a distribution over the desired number of negative literals based on the probability distribution of a dataset.

The Probabilistic Satisfiability problem (PSAT) involves assigning probabilities to a set of propositional formulations and deciding whether this assignment is consistent. The pioneering work was introduced by George Boole [28] as another perspective. He proposed the PSAT to determine if he could discover a probability measure for truth assignments that satisfy all assessments. The PSAT framework was developed to demonstrate these details as logical sentences with linked probabilities to infer the likelihood of a query sentence. The PSAT was initially suggested by George Boole and, subsequently, was refined by Nilsson [29]. This intelligent perspective was followed by different studies [30–33], which all aimed to integrate the probability tools into satisfiability without considering their implementation in a DHNN. The present study addresses this gap by introducing a probability distribution to the prevailing attribute in the data set, which is represented in a DHNN through desire logic.

There are no studies in this area regarding the way in which the probability distribution for literals with SAT may be represented in a DHNN. Thus, the findings addressing this issue can be used to guarantee the most effective search for satisfying interpretations. Therefore, this study introduces S-type Random \(k\) Satisfiability \((\delta^kSAT)\), where \(k = 1, 2\)
(δ2SAT) and with the probability distribution of the prevailing attribute in the simulation dataset. It aims to address the problem regarding RANSAT, where $k$ randomizes structural issues by utilizing two statistical features, the probability distribution and the sample size formula, to obtain an estimator for the binomial distribution dataset. In addition to helping to assign the negative literal that was mapped to the prevailing attribute in a dataset with a non-systematic logical RANSAT, the main feature of RANSAT is its structural flexibility, which takes advantage of another logical rule, 2SAT, whereas the non-systematic logical rule provides a more diversified solution [34,35]. Furthermore, the probability distribution is used to control the composition’s probability of appearing in first- and second-order logic to avoid a poorly explained or lack of interpretation in non-systematic SAT by providing suitable logical combinations depending on the dataset’s distribution. Moreover, the logic system uses the binomial distribution’s sample size to determine the appropriate number of negative literals based on the predetermined proportion appearing in the dataset. Then, the clauses are distributed in each order depending on the probability distribution governing appearance. This approach will help us determine the appropriate weight of a negative literal number in logic systems based on the distributed clauses in order to create suitable solutions [24]. Notably, researchers tend to neglect negative literals because they are indirectly mapped errors in a logical structure [36]; however, in this study, negative literals represent the prevailing attribute in a binomial distribution that has only two characteristics.

Our proposed logical rule will provide flexibility with respect to controlling the overall structure of δ2SAT in terms of the dataset’s characteristics by combining both the effects of statistical parameters and non-systematic features to identify suitable neuronal variation and diversity in the proposed logic. The main aims of this study are as follows:

(a) To formulate a novel logical rule called S-Type Random $k$ Satisfiability, where $k = 1, 2$ and statistical tools are integrated to structure first- and second-order logic in order to select the most suitable number of negative literals.

(b) To propose a probability logic phase to determine the probability of the appearance of the number of the first- and second-order literals and the distribution of the desired number of negative literals on every clause by considering the selected dataset.

(c) To implement the proposed S-Type Random 2 Satisfiability as a symbolic structure in the Discrete Hopfield Neural Network by reducing the logical inconsistency of the corresponding zero-cost function’s logical rule, as well as determine the synaptic weight of the DHNN that achieves the cost function equivalent to the satisfied δ2SAT.

(d) To compare the effectiveness of δ2SAT with respect to producing the appropriate logical structure during the probability logic phase before training in the Discrete Hopfield Neural Network by using three proposal metrics in accordance with the existing benchmark works.

(e) To examine the capability of the proposed δ2SAT under the current logical rules with respect to the training and testing phase, demonstrate synaptic weight management, and ascertain the quality of neuronal states’ efficiency in the DHNN via well-known performance metrics.

(f) To investigate the proposed δ2SAT system’s structural behavior during the training phase and thereby demonstrate the flexibility of this logical structure by using a novel form of analysis—synaptic weight analysis—via the mean of the synaptic weights.

The framework of this paper is as follows: The motivation for this study is described in detail in Section 2. An overview of δ2SAT’s structure is given in Section 3. The integration of δ2SAT into a DHNN is described in Section 4. Section 5 explains the experimental setup and performance assessment metrics incorporated into the simulation. In Section 6, the effectiveness of the proposal logic in a DHNN is discussed and analyzed, with comparisons made to several existing logical structures with regard to various parameters and phases. The conclusions and future work are presented in Section 7 at the end of the article.
2. Motivation

2.1. Issue with the Identified Probability Distribution

With reference to the structural issue regarding existing systematic and non-systematic satisfiability, that is, the systematic logic \( k \text{SAT} \) [19,37], the relevant approaches in this respect implement random selection for the literal states from within clauses, where the clauses are selected uniformly, without regard to the individual probability or chance of appearing in the required population dataset. Whereas the non-systematic logic \( \text{RAN}k\text{SAT} [20,22] \) structure is defined randomly, wherein the clauses are selected uniformly. Moreover, the chance of obtaining both negative and positive literals is uniformly distributed [38], with both outcome having an equally likely chance of appearing. This implies that the population follows a uniform distribution and is thus considered a limited option. In this study, we address this research gap by giving the clauses and negative literals inside clauses the priority of a population dataset’s probability distribution, and when the dataset has two characteristics, i.e., negative and positive literals, we assign the negative literal for the prevailing attribute that is withdrawn from a binomial distribution.

2.2. Initialization for the Number of Clauses and Number of Neuron

The investigation into controlling the general structure of SAT is still ongoing. Cai and Lei’s [39] work proposed a Partial Maximum Satisfiability (PMAXSAT) clausal weighting mechanism, with a positive integer as its weight. This method demonstrated the power of weight in terms of controlling the distribution of a logical structure based on the desired result. Conversely, Always et al. [23] suggested a non-systematic logical rule, MAJ2SAT, which seeks to create bias in the selection of 2SAT over 3SAT via the \( r^* \) ratio. The MAJ2SAT system successfully provides more neuronal variations that increase the composition of the 2SAT with the same number of neurons. Despite the benefit of extracting information from real datasets that exhibit the behaviors of 2SAT and 3SAT, the persistent issue is the system of selection, which limits the value of \( r \) in the set of limited pre-defined intervals and is chosen randomly without considering a dataset’s probability distribution. Therefore, we propose the non-systematic logical rule \( \delta2\text{SAT} \), which incorporates a probability logic phase to calculate the probability of first- and second-order clauses appearing from the dataset by determining the required number of literal and clauses.

2.3. Initialization for the Number of Negative Literals

The structure of SAT should be subjected to a systematic analysis to avoid the poor description of a dataset. Dubois and Prade [40] examined the role of logic in dealing with uncertainty in an ANN. The work concluded that it was crucial to use the generalization method to determine how many negative literals should be distributed for technical convenience. Zamri et al. [24] introduced \( r\text{SAT} \) with the (logic phase) as a new phase to produce a non-systematic logical structure based on the ratio of negative literals. The ratio is generated in the logic phase by employing GA to increase the logic phase’s effectiveness. Nevertheless, the findings showed that the proposed model performed well, indicating that having a dynamic distribution of negative literals will benefit the generation of global minimum solutions with different states of the final neurons. One of the limitations of the weighting scheme is the method of choosing the number of negative literals, where the value of \( r \) is in the set of limited pre-defined intervals and is subject to the issue of random system selection without considering the probability distribution of literals.

Alway and Zamri’s studies motivated the current study, in which we propose the non-systematic logical rule \( \delta2\text{SAT} \), which incorporates a probability logic phase to calculate the appearance-related probability distribution in the first-order and second-order clauses from the real dataset by predetermining the required number of neurons or number of clauses via harnessing the behavior of 2SAT so as to explore a wider solution space and extract information from datasets, as well as assign the number of negative literals required for logic by using the sample size formula with a predefined, prevailing attribute proportion from the dataset that will be exposed in the logic.
2.4. Synaptic Weight Performance Using Statistical Analysis

The research on satisfiability in DHNNs suffers from a lack of statistical analysis, especially in terms of synaptic weight, which is considered the backbone for the global minimum solution achieved during testing phases. We determine synaptic weight by contrasting the cost function with Lyapunov energy. The previous studies on systematic and non-systematic approaches were limited in terms of assessing the performance accuracy of the logic in different phases, as mentioned in [9,21,22]. The synaptic weight was analyzed at several points in this study since they were not completely comprehensible in [20,26], wherein the authors describe the dimensions of the synaptic weight values. In addition, [27] measured the accuracy of the error in the synaptic weight by evaluating the differences between the synaptic weight obtained by Wan’s method and the synaptic weight achieved in the training phase. The gap was addressed in this study by using new statistical tests to capture the impact of changing the synaptic weight during training phases due to the absence of statistical tools in the synaptic weight analysis.

3. S-Type Random 2 Satisfiability Logic

S-Type Random 2 Satisfiability ($\delta 2SAT$) is a new category of non-systematic-clause SAT in which the probability distribution is used to assign prevailing attributes in the dataset via two methods: First, depending on the dataset requirements, we assigned the probability of the appearance of first- and second-order logic. Second, we used the sample size from a binomial population [41] to ascertain the appropriate number of negation literals inside each clause based on its assigned probability since the probability of a negative literal appearing follows a binomial distribution. The novelty of the mentioned methods is that they determine the suitable weight of negative literal numbers ($\xi$) in logic depending on the probability clauses distributed, which will lead to greater structural diversity. In addition, the negative literal number is not fixed, and by increasing or decreasing the probability of obtaining a literal number in the logic system, there is greater flexibility in the dataset.

Our approach can be introduced as a form of non-systematic logic comprising $n$ literals per $T$ clauses. It is a general form of RANKSAT logic, where $k = 1,2$ is expressed in the $k$ Conjunctive Normal Form ($kCNF$). The components of the S-Type Random 2 Satisfiability Logic problem are as follows:

(a) A set of $h$ variables, $\tau_1, \tau_2, \tau_3, \ldots \tau_h$, where $\tau_i \in \{-1,1\}$ for all items in our logic system;
(b) A set of $h$ non-redundant literals $r_j$, where $r_i$ is the positive ($r_i$) or a negative ($\neg r_j$) nature of a literal;
(c) A set of $\lambda$ distinguishable clauses, $T_1, T_2, T_3, \ldots, T_{\lambda}$, where every clause is composed of $h$ literals joined by $\land$ logical (AND) Booleans, which is distributed as follows:

i. A set of $x$ first-order clauses: $T_1^{(1)}, T_2^{(1)}, T_3^{(1)}, \ldots, T_x^{(1)}$, $x \in \mathbb{N}$.
ii. A set of $y$ second-order clauses: $T_1^{(2)}, T_2^{(2)}, T_3^{(2)}, \ldots, T_y^{(2)}$, where $T_y^{(2)} = (r_i \lor r_j)$, $y \in \mathbb{N}$.

The general formulation of S-Type Random 2 Satisfiability is given as follows:

\[
\Theta_{\delta 2SAT} = \frac{\chi}{i} T_i^{(1)} \land \frac{\gamma}{j} T_j^{(2)} \quad \text{for } k = 1,2
\]  

\[
T_i^k = \begin{cases} 
(r_i), & k = 1 \\
(r_i \lor r_j), & k = 2
\end{cases}
\]  

where $\Theta_{\delta 2SAT}$ in Equation (1) is $\delta 2SAT$ for $k = 1, 2$. The difference between $\delta 2SAT$ and RAN2SAT lies in the selection system for the number of clauses and the number of negative literals in $\delta 2SAT$. This system is established under the condition that the number of clauses corresponds to:

\[
\begin{align*}
x_m &= p(x) \cdot \lambda_m \\
y_m &= p(y) \cdot \lambda_m
\end{align*}
\]
where \( \lambda_m \) denotes the total number of literals \( \lambda_1 \) or total number of clauses \( \lambda_2 \); \( y_m \) and \( x_m \) denote the number of literals in the first- and second-order clauses or the number of clauses when \( m = 1, 2 \), respectively; \( y_m, x_m \geq 0 \) represent clauses \( T_k^1 \) for different values of \( k \); and \( p(x_m) \) and \( p(y_m) \) denote the probability of first- and second-order logic appearing, which is calculated by the Laplace formula [42] to find the probability \( A_{y_m} \) from population \( \Omega \) expressed as follows:

\[
p(y_m) = \frac{|A_{y_m}|}{|\Omega|}
\]

\( |A_{y_m}| \) represents a number of elements that contain a prevailing attribute from the total number of a dataset \( |\Omega| \) in this study. We will denote the probability of second-order \( p(y_m) \) by \( Y \), which is considered as the first parameter in \( \delta^{2SAT} \).

The number of negated literals that exist in each \( T_k^1 \) will be determined by \( \xi \), where \( \xi \in \mathbb{N} \) is the negative literal number used to obtain \( \rho \) in the dataset [41] and is calculated as follows:

\[
\xi = \frac{\lambda_m \rho_0 (1 - \rho_0)}{(\lambda_m - 1)(d^2/\xi^2) + \rho_0 (1 - \rho_0)}
\]

where:
- \( \rho \): The pre-defined negative literal proportion required in the logic system (Second parameter in Logic).
- \( \rho_0 \): the negative literal proportion in the population (which is available before the survey; if no estimate of \( \rho_0 \) is available prior to the survey, a worst-case value of \( \rho_0 = 0.5 \) can be used to determine the sample size).
- \( d \): the margin of error (or the maximum error) of the negative literal proportion, which is calculated as follows:

\[
d = Z_{\alpha/2} \sqrt{\frac{p(1-p)}{\lambda}}
\]

\( Z \): the upper \( \alpha / 2 \) point of the normal distribution when \( \alpha = 0.01 \), where Significance Level = \( P \) (type I error) = \( \alpha \).

The distribution of the number of negated literals in each order logic clause \( T_k^1 \) is dependent on the value \( \beta_k \), where:

\[
\begin{align*}
\beta_1 &= (\xi \times p(x)) \\
\beta_2 &= (\xi \times p(y)) 
\end{align*}
\]

In (7), \( \beta_1 \) and \( \beta_2 \) denote first- and second-order logic, respectively, and \( \sum \beta_k \) is the total number of negated literals existing in \( \delta^{2SAT} \) logic, where:

\[
\sum \beta_i - \xi = 0
\]

The structure of \( \Theta_{\delta^{2SAT}} \) is believed to provide more variations and greater diversity of the final neuron states and to be able to find more global solutions in other solution spaces via two effective parameters: \( Y \) and \( \rho \). The implementation of S-type Random \( k \) Satisfiability logic in this study is outlined in Figure 1.

**Probability Logic Phase in \( \delta^{2SAT} \)**

The probability logic phase was developed to assess the features of a prevailing attribute in the dataset via probability distribution, which are then reflected in the logic system by the two parameters \( Y \) and \( \rho \); this differs from the logic phase in \( r^{SAT} \) [24], where the phase is established to allocate the correct ratio of the negative literals and the position in the \( r^{SAT} \) logic via metaheuristics. The main purpose for the probability logic phase is to extract the required information from the dataset, and then generate the correct structure of RAN2SAT logic depending on the dataset features assigned by the two probability Equations (3) and (5). Subsequently, once the desired logic has been attained, the probability logic phase is complete. This section will introduce some logic generated
from the dataset using the two parameters \( Y \) and \( \rho \); the restriction in the probability logic phase is as follows:

\[
p(y_m) + p(x_m) = 1, \quad p(y_m) > p(x_m), \quad p(x_m) \neq 0
\]

whose probability function can be defined as follows (Nilsson 1986) [29]:

\[
\begin{align*}
p(\lambda) &= 1 \\
r_i \land r_j \equiv 0, \text{ its mutually exclusives, then} \\
p(r_i \land r_j) &= p(r_i) + p(r_j)
\end{align*}
\]

Figure 1. Block diagram of the proposed S-type Random 2 Satisfiability logic \( \Theta_{\delta 2 SAT} \).

According to the applied method for the determination of probability, there are two types of \( \delta 2 SAT \): First, there is the type of probability logic phase that determines the probability of the appearance of the number of first-order logic and second-order logic literals \( \lambda_1 \) and the distribution of the desired number of negative literals in each clause depending on the selected dataset. Second, there is the type of probability logic phase that determines the probability of the appearance of the number of first-order logic and second-order logic clauses \( \lambda_2 \) and the distribution of the desired number of negative literals in each clause depending on the selected dataset. Table 1 introduces some possible examples of two cases of the logic of \( \delta 2 SAT \) that can be used to generate the dataset using Equations (4), (5) and (7) when \( \rho = 0.7 \).

Table 1. Possible structures of \( \delta 2 SAT \) when \( \rho = 0.7 \).

<table>
<thead>
<tr>
<th>( \lambda_m )</th>
<th>( \xi )</th>
<th>( Y )</th>
<th>Possible ( \delta 2 SAT )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \lambda_1 = 10 )</td>
<td>6</td>
<td>0.6</td>
<td>Case 1: ( \neg r_1 \land r_2 \land \neg r_3 \land r_4 \land (r_5 \lor \neg r_6) \land (r_7 \lor \neg r_8) \land (\neg r_9 \lor \neg r_{10}) )</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>0.8</td>
<td>Case 2: ( \neg r_1 \land \neg r_2 \land (r_3 \lor \neg r_4) \land (r_5 \lor r_6) \land (r_7 \lor \neg r_8) \land (r_9 \lor \neg r_{10}) )</td>
</tr>
<tr>
<td>( \lambda_2 = 5 )</td>
<td>4</td>
<td>0.6</td>
<td>Case 3: ( \neg r_1 \land \neg r_2 \land (r_3 \lor \neg r_4) \land (r_5 \lor \neg r_6) \land (r_7 \lor \neg r_8) \land (r_9 \lor \neg r_{10}) )</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>0.8</td>
<td>Case 4: ( \neg r_1 \land (r_2 \lor \neg r_3) \land (r_4 \lor \neg r_5) \land (r_6 \lor r_7) \land (r_8 \lor \neg r_9) \land (r_9 \lor \neg r_{10}) )</td>
</tr>
</tbody>
</table>

We observe that applying the same probability to more clauses \( \lambda_2 \) results in a reduced number of first-order logic items than applying it to a greater number of neurons \( \lambda_1 \); notably, the number of unique logic combinations that a probability logic phase can create by using a specific value of the two parameters \( Y \) and \( \rho \) is \( \binom{x}{1} \times \binom{y}{1} \). Algorithm 1
presents the pseudocode for the steps taken to generate the $\Theta_{2SAT}$, which starts with the determination of the value of the two parameters $Y$ and $\rho$; then, by applying the constraint of the logic in Equation (9), the probability logic phases operate under the following conditions: (a) $\rho \neq 0.5$, because we need to expose the prevailing attribute. (b) The $z$ is a random number generated to ensure the negative values will be distributed in the logic phase randomly. (c) The loop will run $w$ times to ensure that the logic system will be correctly generated. (d) The probability logic phase ends when Equation (8) is satisfied, at which point the DHNN training phase begins.

The limitation that we observed in $\delta_{2SAT}$’s logic structure is the position of negative literals; these are selected randomly depending on $z$ random numbers, and this randomization clearly effects results in an inconsistent interpretation. In addition, there are no redundant literals. Also, due to the high probability of $2SAT$, the Exhaustive Search (ES) algorithm is unable to find the best number of instances of first-order logic for a small number of clauses that satisfies Equation (9). The utilization of $\Theta_{2SAT}$ in a DHNN is presented as $DHNN - \delta_{2SAT}$. In the next section, we clarify how $\Theta_{2SAT}$ functions as a representational command to control the neurons of the DHNN mappings.

Algorithm 1: Pseudocode for generating the probability of logic phases

| Input: $\lambda_m, \rho, p(y_m), \text{Set of } r_i$ |
| Output: The best $\Theta_{2SAT}$ |

Begin

Generate $\Theta_{2SAT}$
- Initialized $\lambda_m$;
- Initialized Proportion $\rho$;
- Initialized Second-order clauses $p(y_m)$;

Calculate The number of first- and second-order clauses
While $b_1 \leq y \& b_2 \leq x \& (b_1 \leq y \& \beta_1 \neq 0.5 \& \beta_2)$
Do
- Calculate $y_m, x_m$ by Equation (3);
- Calculate $x$ by Equation (5);
- Calculate $\beta_1$ & $\beta_2$ by Equation (7);

End while

Distributed negative literal in logic
While $(w \leq 1000)$
While $(b = \beta_1 \& \rho \neq 0.5 \& \beta_2)$
Do
- Generate random number $z$;
- Generate proportion to be initial negative literal $\rho^*;$
- IF ($\rho^* \geq z$) THEN $\neg r$;
- ELSE $r$;
- End IF
- End for
- Generate random number $z$;
- Generate proportion to be initial negative literal $\rho^*;$
- IF ($\rho^* \geq z$) THEN $\neg b$;
- ELSE $b$;
- End IF
- End while

End While

End

Note: $b, b^*$ is a counter.
4. $\Theta_{\delta_{2}SAT}$ in Discrete Hopfield Neural Network

A DHNN is a type of free, self-feedback information comprising $N$ interconnected neurons with no hidden layers. The neurons are updated one at a time; Ref. [23] asserts that the possibility of neuronal oscillations is eliminated by asynchronous updating. This network has parallel computing, quick convergence, and is also effective in terms of its CAM capacity, which has encouraged researchers to use DHNNs as mediums for solving challenging optimization problems. A general description of the state of activated neurons in a DHNN is provided below:

$$S_i = \begin{cases} 
1, & \sum_j W_{ij} S_j \geq \varepsilon \\
-1, & \text{otherwise}
\end{cases}$$

where the synaptic weight from unit $i$ to unit $j$ is $W_{ij}$. The synaptic weight of a DHNN is always symmetrical, whereby $W_{ij} = W_{ji}$, and has no self-looping, $W_{ii} = W_{jj} = 0$. $S_i$ represents the state of neuron $j$; $\varepsilon$ is a predetermined threshold value, and in this study, $\varepsilon = 0$ to guarantee a uniform decrease in DHNN energy [18]; and $h$ is the number of logic variables. The $\delta_{2}SAT$ is implemented in a DHNN according to the following equation \((DHNN - \delta_{2}SAT)\), due to the requirement for a symbolic rule that can control the network’s output and decrease logical inconsistency by minimizing the network’s cost function. To derive the cost function $E_{\Theta_{\delta_{2}SAT}}$ of $\Theta_{\delta_{2}SAT}$, the following formula can be used:

$$E_{\Theta_{\delta_{2}SAT}} = \sum_i \left( \frac{1}{2} \prod_{j=1}^{y_2} \Psi_{ij} \right) + \sum_i \left( \frac{2}{2} \prod_{j=1}^{x_2} \Psi_{ij} \right)$$

where $x_2$ and $y_2$ are the number of clauses. The inconsistency of $\Theta_{\delta_{2}SAT}$, denoted as $\Psi_{ij}$, is specified in Equation (13), as literals are possible in $\Theta_{\delta_{2}SAT}$:

$$\Psi_{ij} = \begin{cases} 
\frac{(1 - S_i)}{2}, & \text{if } \neg r \\
\frac{(1 + S_i)}{2}, & \text{if } r
\end{cases}$$

where $r$ denotes the random literals assigned in $\Theta_{\delta_{2}SAT}$. If $\frac{(1 + S_i)}{2} = 0$, which leads to $E_{\Theta_{\delta_{2}SAT}} = 0$; this indicates that all clauses in $\Theta_{\delta_{2}SAT}$ are satisfied with the value of the mean task for the logic program during the training phase (i.e., a consistent interpretation is found). A consistent interpretation will help the logic program to derive the correct synaptic weight of $\Theta_{\delta_{2}SAT}$ clauses, and the Wan Abdullah (WA) method [17] can be used to directly compare the cost function and Lyapunov energy function of the DHNN to determine the values of $W_{ij}$. However, it is noted that the DHNN’s synaptic weight can be effectively trained using a traditional approach such as Hebbian learning [1]; nevertheless, Ref. [43] demonstrated that the (WA) method, when compared to Hebbian learning, can achieve the optimal synaptic weight with minimal neuron oscillation. Synaptic weight is a building block (matrix) of CAM. Therefore, a specific output-squashing mechanism will be applied to every neuron in DHNN $- \delta_{2}SAT$ via the Hyperbolic Tangent Activation Function (HTAF) to retrieve the correct logic pattern of the CAM; according to Karim et al. [22], the equation is expressed as follows:

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

A DHNN’s testing phase allows for the asynchronous updating of the neuronal state based on the following equation:

$$h_i = \sum_{j=1}^{N} W_{ij}^{(2)} S_j + W_{ij}^{(1)}$$
$h_i$ represents the network’s local field, where $W_{ij}^{(2)}$ is the second-order synaptic weight and $W_{ij}^{(1)}$ is the first-order synaptic weight. By applying the HTAF to the $h_i$ values, the final state of the neurons is retrieved, and the neuron states $S_i(t)$ are updated by:

$$S_i(t) = \begin{cases} 1, & \text{if } \tanh(h_i) \geq 0 \\ -1, & \text{otherwise} \end{cases} \quad (16)$$

The information that results in $E_{\Theta_{\delta 2SAT}} = 0$ must be present in the neuron’s final state [44], which corresponds to $H_{\Theta_{\delta 2SAT}}$, the Lyapunov energy function [18]:

$$H_{\Theta_{\delta 2SAT}} = -\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1, j \neq i}^{n} W_{ij}^{(2)} S_i S_j - \sum_{i=1}^{n} W_{i}^{(1)} S_i \quad (17)$$

The convergence of the energy will indicate when the degree of convergence has reached a stable state according to [22]. This is supported by Sathasivam [18], who states that if a DHNN is stable and oscillation-free, the Lyapunov energy will reach its lowest value (the equilibrium state). Hence, [45] a DHNN will always converge to the global minimum energy. One can see the convergence of the final neuron state based on the following Equation:

$$\left| H_{\Theta_{\delta 2SAT}} - H_{\Theta_{\delta 2SAT}}^{\min} \right| \leq Tol \quad (18)$$

where $H_{\Theta_{\delta 2SAT}}^{\min}$, the final neuron state, produces the anticipated global minimum energy and is calculated as follows:

$$H_{\Theta_{\delta 2SAT}}^{\min} = -\left(\frac{x_2}{2} + \frac{y_2}{4}\right) \quad (19)$$

where $x_2$ and $y_2$ denote the number of first- and second-order clauses, respectively. Algorithm 2 is an example of the $DHNN - \delta 2SAT$ given in pseudocode, which explains the processes of the training phase and testing phase of $DHNN - \delta 2SAT$. Conventionally, the logic program employs a $2^n$ search space to find consistent interpretations by ES in the training phase.

Figure 2 illustrates the schematic diagram of $DHNN - \delta 2SAT$. Different orders of $k = 1, 2$ are shown in two different blocks. In the orange block, there are two inputs and an output (I/O) line, which are green and yellow, representing the two types of logic distributed by clauses and neuron, respectively. Inside the orange box, the second-order clauses are depicted, and every line represents the connection of the neuron state via weights. On the right side, the dashed blue line denotes the first-order clause that is present in this phase as well, with two (I/O) lines: green and yellow. On the inside, the line represents the connection of the neuron state via weights. The satisfied clauses from the two boxes will result in $E_{\Theta_{\delta 2SAT}} = 0$; the figure only represents the satisfied clauses of $\Theta_{\delta 2SAT}$.

**Algorithm 2: Pseudocode of $DHNN - \delta 2SAT$**

**Begin**

Probability logic phase

- Initialized $\Theta_{\delta 2SAT}$

Training phase

- According to Equation (12), minimize cost function;
- Use WA method to calculate Synaptic weight and store it in CAM;
- According to Equation (19), calculate global minimum energy $H_{\Theta_{\delta 2SAT}}^{\min}$

End

Testing phase

- Initialize Random neuron state;
- According to Equation (14), calculate the HTAF;
- According to Equation (15), calculate the local field;
- According to Equation (16), update neuron state;

End

According to Equation (17), calculate the final neuron energy;

- By using Equation (18), confirm global or local minimum energy;
- Recognize global or local minimum solutions;
- Global minima solutions

Else

- Local minima solutions

End
Figure 2. Schematic diagram of DHNN - δ2SAT for both types of logic; the total of literal is n for first-second-order logic.

5. Experimental Procedure for Testing DHNN - δ2SAT

In this section, we explain the proposed logic output and evaluate it using several evaluation metrics at all phases to guarantee the effectiveness of adding statistical parameters in RAN2SAT, which aimed to produce Θδ2SAT logic. Furthermore, the simulation platform, the assignment of parameters, and the metrics for performance are all explained. All models were used with the ES algorithm, where the algorithm utilizes trial and error to achieve a cost function that is minimized (EΘδ2SAT = 0) [23].

5.1. Simulation Platform

All simulations were carried out using an open-source software, visual basic C++ (Version 2022), and a 64-bit Windows 10 operating system. To avoid biases in the interpretation of the results, the simulations were run on a single personal computer equipped with an Intel Core i5 processor. The open-source software R studio was used to perform the statistical analysis. Eight different simulations—depending on the statistical parameters (probability and proportion)—were conducted, including those involving different numbers of clauses and neurons. In addition, different numbers of logic combinations (η) were tested in this study.

Each simulation’s specifics are as follows:

(a) Various range of parameter Y. This section assesses and examines the effects of the various probabilities that can be obtained from the dataset applied to δ2SAT. The performance metrics at each phase and the effect of parameter alterations on Θδ2SAT were determined.

(b) Various proportions of negative literals, ρ. In this section, we evaluate the impact of different proportions of negative literals on Θδ2SAT, evaluating the performance metrics at each phase and determining the effects of parameter alterations on the proposed logic.

(c) A variety of logic structure analyses. In this section, we compare Θδ2SAT with a number of well-known logical rules in terms of the diversity-satisfying clauses of the logical rule.

(d) Synaptic weight mean analysis for Θδ2SAT models’ simulation includes boxplot and whiskers and a probability function curve.
5.2. The Parameter Setting in Probability Logic Phase

The proposed model incorporates a probability logic phase. As we previously mentioned, there are two types of $\Theta_{2SAT}$ depending on the probability that is applied to the number of neurons or the number of clauses. Numerous types of simulations are conducted to examine the impacts of different probabilities and several types of expected negative literal proportions on the dataset, in which the probability logic phase is dependent upon the dataset. The different probability logic phase will be denoted as $\delta_{2SAT}$, where $\gamma = 1, 2$ (1 refers to the probability with respect to the number of neurons and 2 refers to the probability of the number of clauses), and $\rho$ refers to the negative literal proportion; the overall model can be denoted as $\delta_{12SAT_{0.9}}$. Another type of logic is possible if the range of the probability parameter $Y$ with respect to the number of neurons or clauses stated in the simulation step generates only one type of neuron or clause state, and this will yield a systematic $2SAT$ during initialization, which is not covered in this study; alternatively, the first-order logic clauses will correspond to more than second-order logic. When this occurs, the proposed system’s structural benefit cannot be seen because only one specific type of solution can be found in the final neuron state. In order to prevent these two types of logic, it is proposed that $Y > 0.5$, wherein more features of second-order as opposed to first-order logic are implemented in the DHNN. In parallel, to determine the range proportion, we proposed $\rho > 0.5$ to determine the correct number of negative literals that represent the prevailing attribute in the dataset, and we also considered $\rho_0 = 0.5$ since there is no available information prior to the survey; the symbols of the stages are presented in Table 2.

Table 2. Parameter list for probability logic phases.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Parameter Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predefined proportion range ($\rho$)</td>
<td>[0.6, 0.9]</td>
</tr>
<tr>
<td>Negative literal proportion ($\rho_0$)</td>
<td>0.5</td>
</tr>
<tr>
<td>Probability second-order logic range ($Y$)</td>
<td>[0.6, 0.9]</td>
</tr>
<tr>
<td>The upper $\alpha/2$ point of the normal distribution ($Z$)</td>
<td>2.576</td>
</tr>
<tr>
<td>Significance Level ($\alpha$)</td>
<td>0.01</td>
</tr>
<tr>
<td>Number of learning stages in probability logic phases ($\omega$)</td>
<td>1000</td>
</tr>
</tbody>
</table>

5.3. Parameter Setup of DHNN – $\delta_{2SAT}$

All simulations were run with 100 logical combinations ($\eta = 100$). This method aids the DHNN model’s analysis and the approximate evaluation of the efficacy of the proposed logic in a DHNN with various distributions of the two parameters $Y$ and $\rho$. The number of total literals in the logic system is represented by the number of neurons ($\lambda_1$) in the DHNN. We chose a specific number of neurons: $5 < \lambda_1 < 50$. For the DHNN, we apply a relaxation procedure in accordance with [18]. We select $R = 3$ in this context because a further reduction in the potential neuron oscillation has been observed, and a value of $R$ greater than 4 will yield the same outcome as [27]. Table 3 summarizes the establishment of all the parameters necessary for $DHNN – \delta_{2SAT}$p. In addition, it is notable that each $\delta_{2SAT}$ has a neuron combination that is equivalent to the other DHNN logic systems, which eliminates the issue of a small sample size.

Table 3. List of parameters for $DHNN – \delta 2SAT$.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Parameter Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Learning stages ($\upsilon$)</td>
<td>100 [9]</td>
</tr>
<tr>
<td>Number of Combinations ($\eta$)</td>
<td>100 [9]</td>
</tr>
<tr>
<td>Number of Trials ($\phi$)</td>
<td>100 [44]</td>
</tr>
<tr>
<td>Number of Neurons ($\lambda_1$)</td>
<td>$5 &lt; \lambda_1 &lt; 50$</td>
</tr>
<tr>
<td>Tolerance value ($Tol$)</td>
<td>0.001 [18]</td>
</tr>
<tr>
<td>Method of determining synaptic weight</td>
<td>Wan Abdullah (WA) [17]</td>
</tr>
<tr>
<td>Rate of relaxation ($R$)</td>
<td>3 [18]</td>
</tr>
<tr>
<td>Time of threshold CPU</td>
<td>24 h [20]</td>
</tr>
<tr>
<td>Learning iteration ($\phi$)</td>
<td>$\phi \leq \upsilon$ [26]</td>
</tr>
<tr>
<td>Initialization of neuron states</td>
<td>Random [27]</td>
</tr>
<tr>
<td>Training Algorithm</td>
<td>Exhaustive Search</td>
</tr>
<tr>
<td>Threshold constraint of DHNN ($\theta$)</td>
<td>1.0 [9]</td>
</tr>
<tr>
<td>Activation function</td>
<td>HTAF [22]</td>
</tr>
<tr>
<td>Order of clauses</td>
<td>First- and second-order logic</td>
</tr>
</tbody>
</table>
5.4. Performance Metrics

The objective of each phase includes the evaluation of the performance of the proposed model. Therefore, this study will utilize several performance metrics to assess the efficacy of each simulation in the different phases with respect to the DHNN – δ₂SATρ model to verify the effectiveness of the proposed logic system in terms of the probability logic, learning, and testing analysis phases.

5.4.1. Assessment Logic Structure

The probability logic phase is the phase in which the correct logic sequence is generated and that controls the number of clauses and negative literals by solving Equations (3), (5) and (7). We attempt to evaluate the features of the output logic by comparing it with other models to guarantee well-produced logic in terms of clauses and negative numbers, which will be the acquisition of the minimum cost function given in Equation (12). To determine the appropriate synaptic weight based on the main objective of this phase, we express three features: (a) the number of negative literals affected by parameter ρ, (b) the weights of the second-order logic clauses affected by parameter Y, and (c) the full-negativity second-order logic clauses affected by the two parameters Y and ρ. The goal is to compare these features to determine whether the probability logic phase will be successful in achieving the desired logic system by changing this parameter and demonstrating its excellence with respect to expressing the logic features. The parameter ρ controls the proportion of negative literals; hence, in this section, we test the effectiveness of this parameter based on several different aspects, which are provided below.

The proportion of negativity: in the probability logic phase, the optimal value of negative literals in the logic system will be assigned ξ, which is a constant ratio that is dependent on λ₂, and the probability of negative literals in the logic system will be computed using the following equation:

Probability Of total Negativity (PON):

\[
PON = \frac{1}{\eta} \sum_{i=1}^{\eta} \frac{\lambda_1 - \xi}{\lambda_1}
\]  

Equation (20) is derived from a Laplace formula [42]; we need to test whether the change in ρ will affect the probability of a negative literal structure occurring in the two types of logic compared to other forms of logic that introduce random proportions of negative literals in the logic structure. When compared to other types of logic, this matrix’s scale, if corresponding to the necessary proportion, gives us the correct negative literal probability in the logic structure. While analyzing the deviation of the negative literal in terms of the whole logic system, we introduce a second measure to determine the state of the negative literals in the whole logic system, as shown below:

Negativity Absolute Error (NAE):

\[
NAE = \frac{1}{\eta} \sum_{i=1}^{\eta} \left| \frac{\lambda_1 - \xi}{\xi} \right|
\]  

(21)

The proposed NAE scale measures the amount of error that is not negative if it fits the desired proportion in Equation (5). The optimal NAE is zero, which is equivalent to the required number of negative literals.

The probability of the full negativity of second-order logic: Full negativity second-order \((-r_i \lor \neg r_j)\) logic helps us to represent a greater number of the attributes in the final solution. The main objective of the δ₂SAT is to control the number of negative literals and second-order logical items in the logic structure. We need to expose the features of second-order logic as mentioned previously to fully enjoy the benefits of 2SAT in terms of our proposed logic system. Therefore, the next measure is presented as follows:

Full-Negativity Absolute Error second clauses (FNAE):

\[
FNAE = \frac{1}{\eta} \sum_{i=1}^{\eta} \left| \frac{\xi - \lambda_{2SAT}}{\lambda_{2SAT}} \right|
\]  

(22)

where \(\xi_{2SAT}\) is the number of full negativity second-order clauses and \(\lambda_{2SAT}\) is the number of second-order clauses in a specific string of logic. The accuracy of the logic will be measured by the FNAE scale in terms of generating the full-negative second-order clauses, which are expressed as \((-r_i \lor \neg r_j)\), from the rest of the second-order clauses, that is, \((-r_i \lor r_j), (r_i \lor \neg r_j), \) and \((r_i \lor r_j)\). Similarly, using this scale, we will address the effectiveness degree of the two factor parameters Y and ρ with respect to their significance in terms of altering the second-order clauses. We can determine if the required
logic can represent the prevailing attributes by the properties of this measure. The optimal best of FNAE scale is zero, which is equivalent to the required number fully negative second-order clauses.

To address the effect of a parameter $Y$ in the second-order weight, we propose the weighted error measure, which gives the accuracy of the changing of the effect of $Y$ in both proposed logic types when compared to other logic systems, as follows:

Weight Full-Negativity Absolute Error (WFNAE):

$$WFNAE = \frac{1}{\eta} \sum_{i=1}^{\eta} \left( |\xi_{2SAT} - \lambda_{2SAT}| \right) \times w(y_m)$$

(23)

where $\lambda_{2SAT}$ is the mean number of second-order clauses, and $w(y_m)$ is the weight of second-order clauses, which equals $Y$ because the Laplace formula determines an equally likely probability for all the elements. Using this measure, we can determine the effect of $Y$ on the amount of deviation of the full negative clauses from the mean. We can calculate the real weight for this deviation by multiplying it with the weighted $w(y_m)$. A large scale signifies a high degree of representation in terms of the weight of the negative strings, which greatly improves our understanding of the weight of dominating attribute in logic. By comparing the scale to the other reasoning and assigning weight to that prioritized (completely negative sentences), the deviation is biased towards. Table 4 lists the symbols that we require during this phase.

Table 4. List of parameters used in $DHNN - \delta 2SAT$ experimental setup.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Parameter Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F_{desired}$</td>
<td>Maximum fitness</td>
</tr>
<tr>
<td>$F_i$</td>
<td>Current fitness</td>
</tr>
<tr>
<td>$W_E$</td>
<td>Expected Synaptic weight obtained by the Wan method.</td>
</tr>
<tr>
<td>$W_A$</td>
<td>Actual synaptic weight</td>
</tr>
<tr>
<td>$\lambda_1$</td>
<td>Total number of neurons</td>
</tr>
<tr>
<td>$\xi$</td>
<td>The Total of negative literals in logic system</td>
</tr>
<tr>
<td>$p(y_m)$</td>
<td>The probability of obtaining second-order clauses</td>
</tr>
<tr>
<td>$\xi_{2SAT}$</td>
<td>The number full-negativity second-order clauses</td>
</tr>
<tr>
<td>$\lambda_{2SAT}$</td>
<td>The number of second-order clauses</td>
</tr>
<tr>
<td>$\lambda_{2SAT}$</td>
<td>The mean number of second-order clauses</td>
</tr>
<tr>
<td>$H_{\Theta_{2SAT}}$</td>
<td>Minimum energy value</td>
</tr>
<tr>
<td>$H_{min}$</td>
<td>Final energy</td>
</tr>
<tr>
<td>$R_G$</td>
<td>Ratio of global minimum solutions</td>
</tr>
<tr>
<td>$G_{\Theta_{2SAT}}$</td>
<td>Number of global minimum solutions</td>
</tr>
<tr>
<td>$S_i$</td>
<td>Neuron state</td>
</tr>
<tr>
<td>$S_{max}$</td>
<td>Benchmark neuron state</td>
</tr>
<tr>
<td>Sokal</td>
<td>Sokal and Michener Index</td>
</tr>
<tr>
<td>$R_{tv}$</td>
<td>The Ratio of cumulative neuronal variation</td>
</tr>
</tbody>
</table>

5.4.2. Assessment during the Training Phase

In the training phase, we achieved satisfying assignments of the clauses, which generated the optimal synaptic weights in terms of $\Theta_{\delta 2SAT}$ by minimizing Equation (12). The Root-Mean-Square Error (RMSE) has been used as a basic statistical metric for measuring the quality of a model’s prediction in many fields [24], and it is utilized to identify the quality of the training phase, wherein the value of RMSE training (RMSEtrain) signifies the root square of the error between the neurons’ desired fitness value $F_{desired}$ generated and their current fitness $F_i$ [22]. The RMSEtrain formula is:

$$RMSE_{\text{train}} = \sqrt{\frac{1}{\eta} \sum_{i=1}^{\eta} (F_i - F_{desired})^2}$$

(24)

The optimal value of the RMSE in the DHNN model is achieved when it is zero, which means the WA method derived the correct synaptic weight. Furthermore, a good model is achieved when the measure is between 0–60. Whereas the Root-Mean-Square Error in synaptic weight (RMSEweight) used will be assessed based on the following formula

$$RMSE_{\text{weight}} = \sqrt{\frac{1}{\eta \times \eta} \sum_{i=1}^{\eta \times \eta} (W_E - W_A)^2}$$

(25)
where $W_E$ denotes the Expected synaptic weight obtained by the WA method, and $W_A$ is the actual synaptic weight obtained in the testing phases; this measure gives us a complete understanding of the error produced by the WA method, wherein the best result is 0, which corresponds to Equation (12).

### 5.4.3. Assessment for Testing Phase

In the event that the suggested network satisfies the requirement in Equation (18), the proposed $DHNN - \delta 2SAT$ will act in conformance with the embedded logical rule during the testing phase. The final neuron state will enter a state of minimum energy, which corresponds to the cost function of the proposed $DHNN - \delta 2SAT$ logical rule. Therefore, based on the synaptic weight generated in the training phase, we evaluate the quality of the retrieved final neuron states (global), namely, the minima solutions. Thus, we apply the next measure as follows: Global minima ratio ($R_G$)—the goal of the global minima ratio is to assess the retrieval efficiency of the $DHNN - \delta 2SAT$. The formula of the $R_G$ is:

$$R_G = \frac{1}{\eta \times \varphi} \sum_{i=1}^{A} G_{\Theta_{2SAT}}$$ (26)

where $G_{\Theta_{2SAT}}$ is the number of global minimum solutions that satisfy condition (18) after being distributed in Equation (19), $\varphi$ is the number of trials in the training phase, and $\eta$ is the logical combination for each run. This metric was frequently used in articles such as [21,38] to assess the proposed $DHNN - \delta 2SAT$’s convergence property.

The second measure in the testing phase is the Root-Mean-Square Error energy (RMSEenergy) [22], which is used to evaluate the minimization of energy achieved by $DHNN - \delta 2SAT$. The energy profile can be determined using RMSEenergy:

$$RMSE_{energy} = \sqrt{\frac{1}{\eta \times \varphi} \sum_{i=1}^{\varphi} (H_{\Theta_{2SAT}} - H_{min_{\Theta_{2SAT}}})^2}$$ (27)

We use RMSEenergy to analyze the converge of $\delta 2SAT$ to determine the actual energy difference between the absolute minimum energy $H_{min_{\Theta_{2SAT}}}$ and the final minimum energy $H_{\Theta_{2SAT}}$.

### 5.4.4. Similarity Index

The similarity index [38] and cumulative neuronal variation [24] can be used to evaluate SAT performance using a DHNN. The similarity index values will be compared with benchmark neuron states $S_i^{max}$ to determine the quality of each optimal final neuron state that achieved global lowest energy, as indicated in the following formula:

$$S_i^{max} = \begin{cases} 1, & r_i \\ -1, & \neg r_i \end{cases}$$ (28)

where 1 denotes a positive literal of $r_i$, and $-1$ denotes a negative literal of $\neg r_i$ in each clause. It should be noted that the benchmark neuron states are the DHNN model’s ideal neuron states that satisfy the conditions in Equation (18). The retrieved final neuron states are compared to the benchmark neuron states indicated in Table 5 to provide a comprehensive comparison of the benchmark neuron states and final neuron states.

### Table 5. Variables’ similarity index specifications.

<table>
<thead>
<tr>
<th>Variable</th>
<th>$S_i^{max}$</th>
<th>$S_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>e</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>f</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>g</td>
<td>-1</td>
<td>1</td>
</tr>
<tr>
<td>h</td>
<td>1</td>
<td>-1</td>
</tr>
</tbody>
</table>

The overall comparison of the benchmark and final neuron states is conducted as follows [9]:

$$C_{S_i, S_i^{max}} = \{(S_i, S_i^{max})| i = 1, 2, \ldots, n\}$$ (29)

According to Case 1 in $\Theta_{2SAT}$ given in the examples in Table 1, the final neuron states are generalizable, as follows: $S_i^{max} = (-1, 1, -1, 1, 1, -1, 1, -1, -1, -1)$.
In this study, we selected a well-known measure with which to determine the similarity index for diverse perspectives, namely, that developed by Sokal and Michener (Sokal) [46], which will be employed to evaluate the viability of the recovered final neuron states. It should be noted that Sokal measures the similarity of negative cases of \( S_i \) with \( S^{\text{max}}_i \) over a range of \((0, 1)\). The formulation is as follows:

\[
\text{Sokal}(S_i, S^{\text{max}}_i) = \frac{f + e + h + g}{f + e + h + g + f + e + h + g} \quad (30)
\]

The Ratio of Cumulative Neuronal variation (\( R_{tv} \)) is used because the testing phase uses the DHNN’s ability to directly memorize the final neuron states ratio without the need to create a new state. This is expressed as follows:

\[
\begin{align*}
R_{tv} &= \frac{1}{\phi \times \eta \times \upsilon} \sum_{i=1}^{\phi} \sum_{j=1}^{\eta \upsilon} E_{ij} \\
E_{ij} &= \begin{cases} 
1, & S_i \neq S^{\text{max}}_i \\
0, & S_i = S^{\text{max}}_i
\end{cases}
\end{align*}
\quad (31)
\]

where \( E_{ij} \) denotes the points scores used to assess the difference between newly recovered final neuron states and the benchmark neuron states. The symbol that we require for this Testing and Training phase is shown in Table 4.

5.5. Comparison of Method and Baseline Models

Since this study focuses on investigating \( \delta, \gamma, \rho, 2\text{SAT} \) performance with respect to its logical behavior, we need to investigate the \( \delta, \gamma, 2\text{SAT} \rho \)’s performance in terms of \( Y \) and \( \rho \) with regard to constructing a good logical structure in the probability logic phase. Therefore, we compare \( \delta, \gamma, 2\text{SAT} \rho \) with the existing logic systems in DHNNs based on the logic structures, testing phases, and the quality of the solution to examine two behaviors relating to logic:

(a) The effects of controlling a number of clauses on the second-order weight and non-systematic logic structure.
(b) The capability of \( \delta 2\text{SAT} \) to control the negative literals and accurately reflect the behavior of the dataset.

In order to examine the logic in a DHNN after its implementation, we also compare its final neuron state’s quality to that of RAN2SAT. We also evaluate the variation introduced by the testing phase, global minima solutions, and variation of neurons. The most recent logic systems with a 2SAT structure were selected for this reason, and one of their features was the decision to compare the logic systems’ structures. Each clause contains two literals and all clauses are joined by a disjunction.

(a) 2SAT [37]: This is a systematic logical rule that was implemented into a DHNN, with each clause containing two literals. It is a special type of logic of general Boolean satisfiability. Each phrase in the 2SAT model can withstand no more than one suboptimal neuron update, leaving it more akin to a two-dimensional decision-making system. When included into logic mining, this logic system has demonstrated good applicability in task classification. Neuron counts varied from 5 < \( \lambda_1 \) < 50.
(b) MAJ2SAT [23]: The initial focus of the effort was on developing the current non-systematic SAT logic structure. MAJ2SAT suggests structural modifications when considering unbalanced clauses. The unbalanced feature result from different compositions of 2SAT and 3SAT. As a result, MAJ2SAT prefers a greater number of 2SAT clauses. Moreover, to avoid any bases, we limited the number of neurons ranging from 5 < \( \lambda_1 \) < 50.
(c) RAN2SAT [20]: This system is a second-order and first-order clause logical rule that was implemented in a DHNN as an initial form of non-systematic logic. The \( \delta, \gamma, 2\text{SAT} \rho \) has no structural differences compared to RAN2SAT but consists of a logic probability phase. Due to the connection of the first-order clause, RAN2SAT is reported to provide a greater variety in terms of synaptic weight. Although each literal state was chosen at random, the number of clauses in each order can be determined in advance. Specifically, the number of neurons ranged from 3 < \( \lambda_1 \) < 50.
(d) RAN3SAT [22]: This work expanded on the previous work by [20], incorporating higher-order logic of 3SAT clauses in a non-systematic SAT structure, which improved the lack of interpretability of the current non-systematic SAT by storing more neurons per clause. Although the number of clauses for each sequence was selected at random, each literal state was defined. In this case, again, we restricted the number of neurons; the range was 6 < \( \lambda_1 \) < 50.
(e) YRAN2SAT [26]: This system is known as the Y-Type Random 2-Satisfiability logical rule. YRAN2SAT’s novelty is introduced by randomly generating first- and second-order clauses. It is a combination of systematic and non-systematic logic. YRAN2SAT can explore the search area with a high potential for solution diversity by adding the features of both clauses. YRAN2SAT introduces remarkable logical flexibility, while the number of all clauses is predefined by the user and the literal states are defined at random. The range of the number of neurons is $1 < \lambda_1 < 50$.

(f) rSAT [24]: This is a new, non-systematic satisfiability logic class, known as Weighted Random k Satisfiability for $k = 1, 2$, which includes a weighted ratio of negative literals and adds a new logic phase to produce a non-systematic logical structure based on the number of negative literals specified. More diverse final neuron states were obtained by integrating rSAT into a DHNN. The proposed model showed outstanding promise as an advanced logic-mining model that can be used further in the forecasting and prediction of real-world problems. In this study, we select ($r = 0.5$) because it has been discovered that it performs well in the logic phase of the rSAT [24]. The range of the number of neurons was $5 < \lambda_1 < 50$.

5.6. Benchmark Dataset

In this study, the proposed model generated bipolar interpretations randomly from a simulated dataset. More specifically, the logical illustration that was used in the simulations will serve as the foundation for the structure of the simulated data. The simulated dataset is commonly used in the modeling and evaluation of the efficacy of SAT logic programming, as demonstrated in the work of [18,22,27].

5.7. Statistical Test

This section provides a brief definition of the statistical measures that will be used in this study for two purposes (description and testing):

(a) The measure of central tendency is defined as “the statistical measure that designates a single value as being indicative of a whole distribution” [47]. Therefore, we selected two measures: (a) The average, which is known as the arithmetic mean (or, simply, “Mean”). It is calculated by adding all of the values in the dataset and dividing them by the number of observations. One of the most significant measures is the central tendency measure. The mean has the disadvantage of being sensitive to extreme values/outliers, especially when the sample size is small. As a result, it is ineffective as a measure of central tendency for a skewed distribution [48]. Its formula is expressed as follows:

$$\bar{X} = \frac{\sum_{i=1}^{n} x_i}{n^*} \quad \text{(32)}$$

where $\bar{X}$ denotes the mean, $x_i$ represents the set of data, and $n^*$ denotes the sample size of the data.

(b) The median is the value that, when all observations are arranged in ascending/descending order, occupies the central position. It divides the frequency distribution into two halves, is not biased by outliers, and is determined by the following formula [49]:

$$\tilde{X} = \begin{cases} \frac{x_{\frac{n^*}{2}} + x_{\frac{n^*}{2} + 1}}{2}, & \text{if } n^* \text{ even} \\ x_{\frac{n^*}{2}}, & \text{if } n^* \text{ odd} \end{cases} \quad \text{(33)}$$

where $\tilde{X}$ denotes the median, and $n^*$ denotes the sample size of the data.

(b) The measure of dispersion: Variability measures inform us about the distribution of the data and allow us to compare the dispersion of two or more sets of data. We can determine whether the data are stretched or compressed using dispersion metrics, namely, the Standard Deviation (SD), which evaluates variability by considering the distance between each score and the distribution’s mean as a reference point. It is a variance square root and gives an indication of the standard deviation or average separation from the mean. It is presented as follows:

$$\sigma x \ (SD) = \frac{\sum_{i=1}^{n} (X_i - \bar{X})^2}{n^* - 1} \quad \text{(34)}$$
(c) The boxplot and whiskers (measure of position): The boxplot [Tukey, 1977] is a well-known tool for displaying significant distributional features of a dataset. The classical box-plot displays the quartiles $Q_1, Q_2, Q_3$, and whiskers, where the median is equal to $Q_2$, which is used to estimate the 25th ($Q_1$) and 75th ($Q_3$) quantiles, thus providing an estimate of the interquartile range $IQR = Q_3 - Q_1$. The range of the majority of the data (the whisker’s length) ends at those values just inside the whisker’s “limits” (referred to as “fences” and defined by $LF = Q_1 - 1.5 \times (IQR)$ and $UF = Q_3 - 1.5 \times (IQR)$, lower (LF) and upper (UF) respectively. Observations outside the whiskers (the outliers), observations beyond the fences [51], plotted individually, are defined as the data points outside the boundaries. When comparing different datasets, the boxplot is particularly helpful. Instead of using a Table of Values, we can quickly compare all reported statistics across numerous datasets. The simple, effective design of the boxplot aids the comparison of summary statistics (location, spread, and range of the data in the sample or batch).

(d) The Laplace Principle of Probability states that in a space of elementary events $\Omega$, where each element has the same chance of appearing, the probability of a compound event, $A$, is equal to the ratio of outcomes that are favorable to the occurrence of all other outcomes. This is demonstrated by the formula in Equation (4):

(e) The probability density function curve is a schematic illustration of the probability of random variable density function that is given by:

$$f(x) = \int_{-\infty}^{\infty} f_x(x) \, dx$$  \hspace{1cm} (35)

where $f(x)$ denotes the probability density function for random variables; the shape provides a visualization of the distribution of continuous random variables and provides the probability that a continuous random variable’s value will fall within a specific interval.

(f) The Wilcoxon signed-rank test: The Wilcoxon signed-rank test was introduced for the first time by Frank Wilcoxon in 1945 [52]. It is a one-sample location problem-based nonparametric test that is used to test the null hypothesis wherein the median of a distribution equals some value ($H_0 : \tilde{X} = 0$) for data that are skewed or otherwise (i.e., do not follow a normal distribution). It can be used instead of a one-sample t-test or paired t-test, or for ordered categorical data with a normal distribution. If ($p$-value $\leq \alpha$), the null hypothesis is rejected; this is strong evidence that the null hypothesis is invalid, i.e., the result of the median is significant. The Formula for the Wilcoxon Rank Sum Test ($W$) for $x_i$ independent random variable is:

$$W = \frac{W^* - \frac{\pi(\pi+1)}{4}}{\sqrt{\frac{\pi(\pi+1)(2\pi+1)}{24}}}$$  \hspace{1cm} (36)

where $\pi = \text{number of pairs whose difference is not 0}$. $W^* = \text{smallest of the absolute values of the sums of } x_i$. The symbols of these statistics are listed in Table 6. The details of the implementation of $\Theta_{2SAT}$ into DHNN is presented in Figure 3, which contains the probability logic, the learning and testing phases, and the evaluation metric in each phase.

<table>
<thead>
<tr>
<th>Table 6. Parameters List for $DHNN - \delta_{2SAT}$.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Parameter Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\bar{X}$</td>
<td>The arithmetic mean</td>
</tr>
<tr>
<td>$\bar{X}$</td>
<td>The median</td>
</tr>
<tr>
<td>$\sigma x$</td>
<td>Standard deviation</td>
</tr>
<tr>
<td>$Q_1$</td>
<td>First quartile</td>
</tr>
<tr>
<td>$Q_2$</td>
<td>Second quartile</td>
</tr>
<tr>
<td>$Q_3$</td>
<td>Third quartile</td>
</tr>
<tr>
<td>$IQR$</td>
<td>Interquartile range</td>
</tr>
<tr>
<td>$LF$</td>
<td>Lower fences</td>
</tr>
<tr>
<td>$UF$</td>
<td>Upper fences</td>
</tr>
<tr>
<td>$f(x)$</td>
<td>Probability density function for random variables</td>
</tr>
<tr>
<td>$W^*$</td>
<td>Smallest of absolute values of the sum of $x_i$ in Wilcoxon test</td>
</tr>
<tr>
<td>$W$</td>
<td>Wilcoxon test value (sum of smallest and largest absolute values of the sum $x_i$)</td>
</tr>
</tbody>
</table>
Figure 3. Flowchart of DHNN – δSAT and Experimental evaluation.

6. Results and Discussion

In this section, we describe the suggested logical output and evaluate it using a variety of evaluation metrics throughout all three phases to ensure that the addition of statistical tools to the RAN2SAT structure and the produced δ,2SAT logic was effective. Furthermore, the simulation
platform, assigned parameters, and the metrics’ performance are discussed in this section. It is important to note that we have not considered any optimization during the probability logic phase, as in Zamri et al.’s [24] work; the training phase, as proposed in [21,38]; or the testing phase, as proposed in [9,53].

6.1. Logic Structure Capability

The probability phases give us different models in terms of negative literals and second-order logic with respect to the two parameters $Y$ and $\rho$. Since both parameters fall within the $[0,1]$ interval, we can generate an endless (infinite) number of 2SAT models using both parameters. For the majority of the representations of 2SAT, we chose to use $Y (p(y_m))$ more frequently than $p(x_m)$ so that the results would be in the range (0.6–0.9). In this study, we chose values of $\rho$ greater than 0.5 in the range (0.6–0.9) of the probability logic phases to obtain a greater representation of the negative numbers in order to study the predominating attributes in the dataset, as we previously mentioned.

We selected the most significant differences from the two intervals and designated them as models, which are illustrated in Table 7, in order to examine the efficacy of the two parameters with different numbers of $\lambda_m$, where $5 < \lambda_1 < 50$ so at improve the benefits compared with other recently developed produced logic systems. Subsequently, we will test two $\delta_2SAT\rho$ types with different numbers of $\lambda_m$, $Y$, and $\rho$; these values are selected considering the significant change in probability and negative literals. Notably, values of $\rho = 1$ will be disregarded because we do not need all literals to be negative because the structure will not represent the Binomial distribution dataset. Moreover, the $DHNN - \delta_2SAT$ will give one satisfied interpretation of a first-order clause [54]; on the other hand, $Y = 1$ will give us second-order logic. It is important to emphasize that we do not consider a systematic $\delta_2SAT$ logical system in this study. Table 7 shows the names of two $\delta_2SAT\rho$ types for different possible models depending on the two parameters $Y$ and $\rho$, as well as other logic symbols.

### Table 7. The logical symbols in the experiment.

<table>
<thead>
<tr>
<th>$Y$</th>
<th>$\rho$</th>
<th>$\delta_2SAT\rho$</th>
<th>$\delta_2SAT\rho$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.6</td>
<td>0.6</td>
<td>A1</td>
<td>Q1</td>
</tr>
<tr>
<td></td>
<td>0.7</td>
<td>A2</td>
<td>Q2</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
<td>A3</td>
<td>Q3</td>
</tr>
<tr>
<td></td>
<td>0.9</td>
<td>A4</td>
<td>Q4</td>
</tr>
<tr>
<td>0.7</td>
<td>0.6</td>
<td>A5</td>
<td>Q5</td>
</tr>
<tr>
<td></td>
<td>0.7</td>
<td>A6</td>
<td>Q6</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
<td>A7</td>
<td>Q7</td>
</tr>
<tr>
<td></td>
<td>0.9</td>
<td>A8</td>
<td>Q8</td>
</tr>
<tr>
<td>0.8</td>
<td>0.6</td>
<td>A9</td>
<td>Q9</td>
</tr>
<tr>
<td></td>
<td>0.7</td>
<td>A10</td>
<td>Q10</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
<td>A11</td>
<td>Q11</td>
</tr>
<tr>
<td></td>
<td>0.9</td>
<td>A12</td>
<td>Q12</td>
</tr>
<tr>
<td>0.9</td>
<td>0.6</td>
<td>A13</td>
<td>Q13</td>
</tr>
<tr>
<td></td>
<td>0.7</td>
<td>A14</td>
<td>Q14</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
<td>A15</td>
<td>Q15</td>
</tr>
<tr>
<td></td>
<td>0.9</td>
<td>A16</td>
<td>Q16</td>
</tr>
</tbody>
</table>

The negativity representation: The PON measure in the different logic models has been tested by Equation (20). The PON represents the probability of the appearance of a negative literal in the entire logic system in all combinations with different $\lambda_1$. It is necessary to control the negative literals in order to determine the prevailing attributes in the dataset, as negative literals will ensure more negativity in the final neurons; then, we can ensure that the attribute will appear in the solution space by helping the DHNN find the optimal solution [24].

The Figure 4, a line representation, shows different layers of logic in different proportions for both types of $\delta_2SAT\rho$. At the same time, for other groups, $\rho = 0.5$ for $rSAT$ logic, and $\rho = \text{random}$ for other logic systems (YRAN2SAT, MAJ2SAT, RAN3SAT, 2SAT, and RAN2SAT). The reason why this is in the minimum levels of the proposed logic for the $\delta_2SAT\rho$ is because, as was already noted, the probability of receiving a negative literal for the SAT is incredibly low. The highest two layers were recorded as $\rho = 0.9$ and $\rho = 0.8$ in both types of $\delta_2SAT\rho$, respectively. By applying Equation (5), we obtain the best number of negative literals for all $\lambda_1$, which is similar to the third layer for the other two groups, where $\rho = 0.6$ and $\rho = 0.7$ were the lowest probabilities in both types of $\delta_2SAT\rho$, which, by the change in the proportional parameter $\rho$, indicates the success in terms of producing the
desired number of negative literals in the logic system, representing the predominate attributes in our dataset. Additionally, there was a direct correlation between the number of neurons in each class of the desired proportion and the proportions where a high PON recorded low probability when the number corresponded to $\lambda_1$. When the number is less than 17 and after 31 for $\lambda_1$, the PON becomes approximately stable. This is because the $d$ in Equation (6) in the sample size equation always selects the optimal sample that reflects the number of negative literals, even if the number of neurons is low. Table 8 provides detailed information on the PON in each proportion group for the two types of logic. Note that group ($\rho = 0.9$) recorded the maximum PON and highest mean value of the PON with low $\sigma$ in both types of $\delta_12SAT_{0.9}$; the small $\sigma$ indicates a different number of neurons $\lambda_1$, and this provides the nearest value of the PON means, and that result is highly similar within each group for all models and increases in accordance with the $Y$ increasing in the models for both types, namely, $\delta_12SAT_\rho$ and $\delta_22SAT_\rho$. The small $\sigma$ indicate, with different number of neuron $\lambda_1$, it provides the nearest value of PON mean’s, and that result is highly similar within each group for all models and increases in accordance with the $Y$ increasing in models for both types, namely, $\delta_12SAT_\rho$, $\delta_22SAT_\rho$. However, we can also note that the PON mean of the other logic systems is closest, indicating that the minimal PON value was recorded in YRANSAT with a minimum mean of 0.4966 and a low SD(\sigma) = (0.015), which indicates that it was also the lowest for different numbers of neurons, but we can also notice that the PON mean of other logic systems is closest, showing low values for different numbers of neurons that were less than or equal to 0.5. The PON results prove the flexibility of $\delta_12SAT_\rho$’s structure in terms of controlling the literals’ states.

![Figure 4](image-url)  
**Figure 4.** PON line representation for models in both types of logic (a) $\delta_12SAT_\rho$, (b) $\delta_22SAT_\rho$, and recently developed logic systems.

The accuracy of the models is evaluated by the NAE measure in Equation (21) in terms of the amount of error that is non-negative or the quantity of the negative literal status for the entire logic system in each proportion group for both types of $\delta_12SAT_\rho$ models. According to Figure 5, in the line representation, note that the effect of the proportional changes in the logic restructure guarantees that the best RANSAT is required, or effective of the prevailing attribute in the dataset where different proportions give us different layers. The details of Figure 5 can be found in Table 9, which shows the minimum values of the NAE that were recorded in a group $\rho = 0.9$ where $A_4$ in $\delta_22SAT_{0.9}$. was recorded as the lowest error by (0.1429). It should be observed that the median value (0.3090) was the lowest possible value, indicating that the $A_4$ for all $n$ neurons of $\lambda_1$ always had a lesser error in the middle sections. Additionally, it should be noted that all models in the same group, $A_6$, $A_12$, and $A_8$, have very similar median values (0.333, 0.31, and 0.320), which is because, as shown in the PON, this group has the highest probability for the representation of a negative literal, which is accomplished by the proportion $\rho = 0.9$. Similarly, in $\delta_22SAT_{0.9}$. $Q_4$ recorded the lowest error as 0.1429, but the least median was recorded by $Q_16$ (0.13125), which means the minimum error lies in the middle values with respect to the number of neurons $\lambda_1$. Moreover, it can be noted from Figure 5 that for a small number of neurons $\lambda_1$, $Q_4$ has fewer NAE values than $Q_8$, $Q_{12}$, and $Q_{16}$. However, the reverse is true for the middle values of $Q_{16}$ compared to $Q_{12}$, $Q_8$, and $Q_4$, as mentioned before regarding...
the effect of $Y$ in $\lambda_1$. However, in Table 9 the value of the median has very small differences from the model in group $\rho = 0.9$. As discussed in terms of the PON, this indicates the successfulness of the proportion of representation in the logic system. The highest NAE value was observed to be for r5SAT with a high median, where $r = 0.5$ with the nearest value of NAE for the other logic systems (YRAN2SAT, MAJ2SAT, RAN3SAT, 2SAT, and RAN2SAT); as previously mentioned, there was a lack of representation of the negative literals in the logic system, as they recorded the least degree of the probability of the appearance of negative literals.

Table 8. PNO results for models with both types of logic, $\delta_1 2SAT\rho$ and $\delta_2 2SAT\rho$, and recently developed logic systems’ details determined by Wilcoxon test for median divided by $\rho$ value.

<table>
<thead>
<tr>
<th>$\rho$</th>
<th>Model</th>
<th>Mean</th>
<th>SD</th>
<th>Min</th>
<th>Max</th>
<th>Model</th>
<th>Mean</th>
<th>SD</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.6</td>
<td>A1</td>
<td>0.5483</td>
<td>0.0294</td>
<td>0.5217</td>
<td>0.6250</td>
<td>Q1</td>
<td>0.5454</td>
<td>0.0318</td>
<td>0.5200</td>
<td>0.6250</td>
</tr>
<tr>
<td></td>
<td>A5</td>
<td>0.5503</td>
<td>0.0290</td>
<td>0.5208</td>
<td>0.6250</td>
<td>Q5</td>
<td>0.5392</td>
<td>0.0165</td>
<td>0.5200</td>
<td>0.5714</td>
</tr>
<tr>
<td></td>
<td>A9</td>
<td>0.5460</td>
<td>0.0211</td>
<td>0.5200</td>
<td>0.6000</td>
<td>Q9</td>
<td>0.5392</td>
<td>0.0115</td>
<td>0.5227</td>
<td>0.5556</td>
</tr>
<tr>
<td></td>
<td>A13</td>
<td>0.5368</td>
<td>0.0100</td>
<td>0.5208</td>
<td>0.5500</td>
<td>Q13</td>
<td>0.5319</td>
<td>0.0134</td>
<td>0.5200</td>
<td>0.5556</td>
</tr>
<tr>
<td>0.7</td>
<td>A2</td>
<td>0.5827</td>
<td>0.0253</td>
<td>0.5526</td>
<td>0.6364</td>
<td>Q2</td>
<td>0.5779</td>
<td>0.0215</td>
<td>0.5500</td>
<td>0.6250</td>
</tr>
<tr>
<td></td>
<td>A6</td>
<td>0.5812</td>
<td>0.0246</td>
<td>0.5556</td>
<td>0.6364</td>
<td>Q6</td>
<td>0.5850</td>
<td>0.0466</td>
<td>0.5500</td>
<td>0.7143</td>
</tr>
<tr>
<td></td>
<td>A10</td>
<td>0.5858</td>
<td>0.0405</td>
<td>0.5532</td>
<td>0.7143</td>
<td>Q10</td>
<td>0.5863</td>
<td>0.0353</td>
<td>0.5500</td>
<td>0.6667</td>
</tr>
<tr>
<td></td>
<td>A14</td>
<td>0.5829</td>
<td>0.0242</td>
<td>0.5500</td>
<td>0.6364</td>
<td>Q14</td>
<td>0.5668</td>
<td>0.0097</td>
<td>0.5526</td>
<td>0.5806</td>
</tr>
<tr>
<td>0.8</td>
<td>A3</td>
<td>0.6565</td>
<td>0.0524</td>
<td>0.6170</td>
<td>0.8000</td>
<td>Q3</td>
<td>0.6558</td>
<td>0.051</td>
<td>0.6170</td>
<td>0.8000</td>
</tr>
<tr>
<td></td>
<td>A7</td>
<td>0.6558</td>
<td>0.0501</td>
<td>0.6190</td>
<td>0.8000</td>
<td>Q7</td>
<td>0.6384</td>
<td>0.0257</td>
<td>0.6200</td>
<td>0.7143</td>
</tr>
<tr>
<td></td>
<td>A11</td>
<td>0.6481</td>
<td>0.0441</td>
<td>0.6170</td>
<td>0.8000</td>
<td>Q11</td>
<td>0.6410</td>
<td>0.0172</td>
<td>0.6170</td>
<td>0.6667</td>
</tr>
<tr>
<td></td>
<td>A15</td>
<td>0.6481</td>
<td>0.0441</td>
<td>0.6170</td>
<td>0.8000</td>
<td>Q11</td>
<td>0.6410</td>
<td>0.0172</td>
<td>0.6170</td>
<td>0.6667</td>
</tr>
<tr>
<td>0.9</td>
<td>A4</td>
<td>0.7753</td>
<td>0.0362</td>
<td>0.7429</td>
<td>0.8750</td>
<td>Q4</td>
<td>0.7716</td>
<td>0.0378</td>
<td>0.7400</td>
<td>0.8750</td>
</tr>
<tr>
<td></td>
<td>A8</td>
<td>0.7709</td>
<td>0.0360</td>
<td>0.7419</td>
<td>0.8750</td>
<td>Q8</td>
<td>0.7704</td>
<td>0.0349</td>
<td>0.7400</td>
<td>0.8751</td>
</tr>
<tr>
<td></td>
<td>A12</td>
<td>0.7730</td>
<td>0.0315</td>
<td>0.7400</td>
<td>0.8571</td>
<td>Q12</td>
<td>0.7678</td>
<td>0.0217</td>
<td>0.7447</td>
<td>0.8182</td>
</tr>
<tr>
<td></td>
<td>A16</td>
<td>0.7578</td>
<td>0.0206</td>
<td>0.7419</td>
<td>0.8182</td>
<td>Q16</td>
<td>0.7623</td>
<td>0.0183</td>
<td>0.7400</td>
<td>0.7895</td>
</tr>
<tr>
<td>random</td>
<td>2SAT</td>
<td>0.5041</td>
<td>0.0211</td>
<td>0.4870</td>
<td>0.5600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MAJ2SAT</td>
<td>0.5021</td>
<td>0.0159</td>
<td>0.4750</td>
<td>0.5286</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RAN2SAT</td>
<td>0.4982</td>
<td>0.0131</td>
<td>0.4829</td>
<td>0.5240</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RAN3SAT</td>
<td>0.5056</td>
<td>0.0121</td>
<td>0.4962</td>
<td>0.5288</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>YRAN2SAT</td>
<td>0.4966</td>
<td>0.0152</td>
<td>0.4625</td>
<td>0.5117</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>r5SAT</td>
<td>0.4900</td>
<td>0.0100</td>
<td>0.4700</td>
<td>0.5100</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: The yellow highlights indicate the highest number in the column and the green indicates the smallest number in the column.

Figure 5. NAE line representation for models in both types of logic (a) $\delta_1 2SAT\rho$, (b) $\delta_2 2SAT\rho$, and recently developed logic systems.
Table 9. Maximum and minimum NAE results for models with both types of logic $\delta_1 2SAT_\rho$, $\delta_2 2SAT_\rho$, and recently developed logic systems with details determined by Wilcoxon test for median divided by $\rho$ value.

<table>
<thead>
<tr>
<th>$\rho$</th>
<th>Model</th>
<th>Min</th>
<th>Max</th>
<th>Median</th>
<th>$W$</th>
<th>Model</th>
<th>Min</th>
<th>Max</th>
<th>Median</th>
<th>$W$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.6</td>
<td>A1</td>
<td>0.6</td>
<td>0.9167</td>
<td>0.8571</td>
<td>120</td>
<td>Q1</td>
<td>0.6</td>
<td>0.9231</td>
<td>0.875</td>
<td>120</td>
</tr>
<tr>
<td></td>
<td>A5</td>
<td>0.6</td>
<td>0.92</td>
<td>0.8333</td>
<td>120</td>
<td>Q5</td>
<td>0.75</td>
<td>0.9231</td>
<td>0.8634</td>
<td>78</td>
</tr>
<tr>
<td></td>
<td>A9</td>
<td>0.6667</td>
<td>0.9231</td>
<td>0.85</td>
<td>190</td>
<td>Q9</td>
<td>0.8</td>
<td>0.913</td>
<td>0.8571</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>A13</td>
<td>0.8182</td>
<td>0.92</td>
<td>0.8536</td>
<td>78</td>
<td>Q13</td>
<td>0.8</td>
<td>0.9231</td>
<td>0.9</td>
<td>45</td>
</tr>
<tr>
<td>0.7</td>
<td>A2</td>
<td>0.5714</td>
<td>0.8095</td>
<td>0.7417</td>
<td>105</td>
<td>Q2</td>
<td>0.6</td>
<td>0.8182</td>
<td>0.75</td>
<td>136</td>
</tr>
<tr>
<td></td>
<td>A6</td>
<td>0.5714</td>
<td>0.8</td>
<td>0.75</td>
<td>120</td>
<td>Q6</td>
<td>0.4</td>
<td>0.8182</td>
<td>0.753</td>
<td>78</td>
</tr>
<tr>
<td></td>
<td>A10</td>
<td>0.4</td>
<td>0.8077</td>
<td>0.75</td>
<td>190</td>
<td>Q10</td>
<td>0.5</td>
<td>0.8182</td>
<td>0.7333</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>A14</td>
<td>0.5714</td>
<td>0.8182</td>
<td>0.7361</td>
<td>78</td>
<td>Q14</td>
<td>0.7222</td>
<td>0.8095</td>
<td>0.76923</td>
<td>45</td>
</tr>
<tr>
<td>0.8</td>
<td>A3</td>
<td>0.25</td>
<td>0.6207</td>
<td>0.5635</td>
<td>105</td>
<td>Q3</td>
<td>0.25</td>
<td>0.6207</td>
<td>0.5635</td>
<td>136</td>
</tr>
<tr>
<td></td>
<td>A7</td>
<td>0.25</td>
<td>0.6154</td>
<td>0.56</td>
<td>120</td>
<td>Q7</td>
<td>0.4</td>
<td>0.6129</td>
<td>0.5848</td>
<td>78</td>
</tr>
<tr>
<td></td>
<td>A11</td>
<td>0.25</td>
<td>0.6207</td>
<td>0.5833</td>
<td>190</td>
<td>Q11</td>
<td>0.5</td>
<td>0.6207</td>
<td>0.5714</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>A15</td>
<td>0.4444</td>
<td>0.6087</td>
<td>0.5714</td>
<td>78</td>
<td>Q15</td>
<td>0.5333</td>
<td>0.6154</td>
<td>0.5862</td>
<td>45</td>
</tr>
<tr>
<td>0.9</td>
<td>A4</td>
<td>0.1429</td>
<td>0.3462</td>
<td>0.309</td>
<td>105</td>
<td>Q4</td>
<td>0.1429</td>
<td>0.3514</td>
<td>0.3274</td>
<td>136</td>
</tr>
<tr>
<td></td>
<td>A8</td>
<td>0.1428</td>
<td>0.3478</td>
<td>0.32</td>
<td>120</td>
<td>Q8</td>
<td>0.1667</td>
<td>0.3514</td>
<td>0.3191</td>
<td>78</td>
</tr>
<tr>
<td></td>
<td>A12</td>
<td>0.1667</td>
<td>0.3514</td>
<td>0.3125</td>
<td>190</td>
<td>Q12</td>
<td>0.2222</td>
<td>0.3429</td>
<td>0.3182</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>A16</td>
<td>0.2222</td>
<td>0.3478</td>
<td>0.3333</td>
<td>78</td>
<td>Q16</td>
<td>0.2667</td>
<td>0.3514</td>
<td>0.3125</td>
<td>45</td>
</tr>
<tr>
<td>random</td>
<td>2SAT</td>
<td>1.005</td>
<td>1.47</td>
<td>1.097</td>
<td>55</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MAJ2SAT</td>
<td>1.018</td>
<td>1.171</td>
<td>1.076</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RAN2SAT</td>
<td>0.999</td>
<td>1.34</td>
<td>1.086</td>
<td>36</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RAN3SAT</td>
<td>1.012</td>
<td>1.309</td>
<td>1.058</td>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>YRAN2SAT</td>
<td>1.039</td>
<td>1.2</td>
<td>1.062</td>
<td>28</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: The results highlighted in yellow indicate the highest number in the column and the green indicates the smallest number in the column, ($p$-value < 0.00), for all models in terms of Wilcoxon test, which means that $H_0$ should be rejected.

The probability of full negativity of second-order logic: We examined the ability of several models incorporating the two types of $\delta_2 2SAT_\rho$ to produce full-negativity second-order clauses with greater accuracy compared to other recently developed logic systems by manipulating two parameters, $Y$ and $\rho$, using the FNAE measure for the second-order clause in Equation (22). Obtaining full negativity second-order logic guarantees that the prevailing attribute in the desired logic structure is represented. Figure 6, a columnar representation, shows the result of the FNAE measure, the higher accuracy achieved by A8 and A4 in $\delta_1 2SAT_\rho$, and Q4 in $\delta_2 2SAT_\rho$ that obtained a value of (0) for FNAE. This is due to the effect of the two parameters in this model, for which the proportion of negative number is $\rho = 0.9$, with a lower probability than other models in second-order logic where $Y = 0.6, 0.7$, which means that all second-order clauses are satisfied by negative numbers because of the small representation of second-order clauses. Based on the same figure, the low accuracy obtained by A1 and Q1, which obtain the maximum number in terms of the FNAE logic (0.8930, 0.8650), is the reason for the low representation of the negative proportion in the logic system. Thus, if we need greater representation of the prevailing attributes in the desired logic structure, we should choose A8 and A16 from $\delta_1 2SAT_0.9$ and Q4 from $\delta_2 2SAT_0.9$. Model A4 recorded higher accuracy using the lowest value of the FNAE median (0.3995), which means the minimum error lies is in the middle values for all neuron quantities $\lambda_1$. We also note the proportion of negative literals is $\rho = 0.9$, which means there are more second-order negative clauses in the models in $\delta_1 2SAT_\rho$ recorded in model Q12, where the lowest FNAE median was (0.4147). The accurate results regarding the FNAE measure are listed in Table 10. It is evident that the ratios of the negative literals are $\rho = 0.9$ and $Y = 0.9$, indicating that the model has a higher fraction of negative, second-order representations. Comparing these results to those of other state-of-the-art logic systems, all of them provide low accuracy due to a higher median value, which indicates that the mean lacks the ability to accurately represent the full-negative second-order values in this model. RAN2SAT performs the best among the logic systems. The latest logic systems give higher errors because the fluctuation in predetermine for assigning second-order logic and low represent for negative literal that indicate the $\delta_1 2SAT_\rho$ and $\delta_2 2SAT_\rho$ is flexible more than the recent logic systems in controlling of two parameters.
Figure 6. FNAE column representation for models in both types of logic (a) $\delta_12SAT\rho$, (b) $\delta_22SAT\rho$, and recently developed logic systems.

Table 10. Maximum and minimum FNAE results for models in both types of logic $\delta_12SAT\rho$, $\delta_22SAT\rho$, and recently developed logic systems with details determined by Wilcoxon test for median.

<table>
<thead>
<tr>
<th>Model</th>
<th>Min</th>
<th>Max</th>
<th>Median</th>
<th>W</th>
<th>Model</th>
<th>Min</th>
<th>Max</th>
<th>Median</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>0.5650</td>
<td>0.8931</td>
<td>0.6186</td>
<td>120</td>
<td>Q1</td>
<td>0.6522</td>
<td>0.8650</td>
<td>0.7472</td>
<td>120</td>
</tr>
<tr>
<td>A2</td>
<td>0.5071</td>
<td>0.8683</td>
<td>0.7688</td>
<td>105</td>
<td>Q2</td>
<td>0.5722</td>
<td>0.8400</td>
<td>0.6993</td>
<td>136</td>
</tr>
<tr>
<td>A3</td>
<td>0.3333</td>
<td>0.8025</td>
<td>0.4825</td>
<td>105</td>
<td>Q3</td>
<td>0.3333</td>
<td>0.7633</td>
<td>0.5678</td>
<td>136</td>
</tr>
<tr>
<td>A4</td>
<td>0.0000</td>
<td>0.5525</td>
<td>0.3996</td>
<td>91</td>
<td>Q4</td>
<td>0.0000</td>
<td>0.7594</td>
<td>0.4527</td>
<td>120</td>
</tr>
<tr>
<td>A5</td>
<td>0.5756</td>
<td>0.8700</td>
<td>0.7186</td>
<td>120</td>
<td>Q5</td>
<td>0.6576</td>
<td>0.8141</td>
<td>0.7322</td>
<td>78</td>
</tr>
<tr>
<td>A6</td>
<td>0.5129</td>
<td>0.8560</td>
<td>0.6860</td>
<td>120</td>
<td>Q6</td>
<td>0.5763</td>
<td>0.7856</td>
<td>0.7000</td>
<td>78</td>
</tr>
<tr>
<td>A7</td>
<td>0.3333</td>
<td>0.7975</td>
<td>0.5950</td>
<td>120</td>
<td>Q7</td>
<td>0.4962</td>
<td>0.7043</td>
<td>0.6179</td>
<td>78</td>
</tr>
<tr>
<td>A8</td>
<td>0.0000</td>
<td>0.7488</td>
<td>0.4440</td>
<td>105</td>
<td>Q8</td>
<td>0.3333</td>
<td>0.7378</td>
<td>0.5017</td>
<td>78</td>
</tr>
<tr>
<td>A9</td>
<td>0.6570</td>
<td>0.8515</td>
<td>0.7775</td>
<td>190</td>
<td>Q9</td>
<td>0.6818</td>
<td>0.7880</td>
<td>0.7660</td>
<td>91</td>
</tr>
<tr>
<td>A10</td>
<td>0.5753</td>
<td>0.8050</td>
<td>0.6925</td>
<td>190</td>
<td>Q10</td>
<td>0.6475</td>
<td>0.7489</td>
<td>0.6938</td>
<td>91</td>
</tr>
<tr>
<td>A11</td>
<td>0.4712</td>
<td>0.7231</td>
<td>0.6047</td>
<td>190</td>
<td>Q11</td>
<td>0.5418</td>
<td>0.6840</td>
<td>0.6225</td>
<td>91</td>
</tr>
<tr>
<td>A12</td>
<td>0.3217</td>
<td>0.7510</td>
<td>0.4488</td>
<td>190</td>
<td>Q12</td>
<td>0.3538</td>
<td>0.4773</td>
<td>0.4148</td>
<td>91</td>
</tr>
<tr>
<td>A13</td>
<td>0.7068</td>
<td>0.7860</td>
<td>0.7651</td>
<td>78</td>
<td>Q13</td>
<td>0.7254</td>
<td>0.7615</td>
<td>0.7386</td>
<td>45</td>
</tr>
<tr>
<td>A14</td>
<td>0.6445</td>
<td>0.7594</td>
<td>0.7125</td>
<td>78</td>
<td>Q14</td>
<td>0.6745</td>
<td>0.7383</td>
<td>0.7018</td>
<td>45</td>
</tr>
<tr>
<td>A15</td>
<td>0.5445</td>
<td>0.6700</td>
<td>0.6245</td>
<td>78</td>
<td>Q15</td>
<td>0.6009</td>
<td>0.6467</td>
<td>0.6196</td>
<td>45</td>
</tr>
<tr>
<td>A16</td>
<td>0.3780</td>
<td>0.5044</td>
<td>0.4204</td>
<td>78</td>
<td>Q16</td>
<td>0.3965</td>
<td>0.4742</td>
<td>0.4177</td>
<td>45</td>
</tr>
<tr>
<td>2SAT</td>
<td>0.6850</td>
<td>0.7682</td>
<td>0.7502</td>
<td>55</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAJ2SAT</td>
<td>0.7350</td>
<td>0.7800</td>
<td>0.7505</td>
<td>36</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RAN2SAT</td>
<td>0.7300</td>
<td>0.7658</td>
<td>0.7441</td>
<td>36</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RAN3SAT</td>
<td>0.7100</td>
<td>0.7700</td>
<td>0.7530</td>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>YRAN2SAT</td>
<td>0.7225</td>
<td>0.7900</td>
<td>0.7514</td>
<td>45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rSAT</td>
<td>0.7400</td>
<td>0.8200</td>
<td>0.7500</td>
<td>78</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: The results highlighted in yellow indicate the highest number in the column and the green indicates the smallest number in the column, ($p$-value < 0.00) for all models in terms of Wilcoxon test, it means reject $H_0$.

A high result in the WFNAE measure in Equation (23) indicates that full-negative second-order logic is more greatly represented. By using this scale, the weight of the sentences in the logic has been evaluated, and the $Y$ parameter may be used to determine whether the model is desirable because the highest probability gives the highest weight. The maximum probability, as shown in Figure 7, is the highest weight represented and is obtained by A16, Q16 in $\delta_12SAT_{0.9}$, and $\delta_22SAT_{0.9}$, respectively, and 0 for YRAN2SAT, because it also produces first-order logic. In Table 11, note the highest significant
The median value was achieved by the A16 and Q16 models (0.4477 and 0.4691, respectively), and the lowest significant median value was achieved by the YRANSAT (0) WFNAE value. This would ensure that the prevailing attribute has the highest representation in our logic compared to other state-of-the-art logic systems, in addition to its ability to minimize and maximize changes in Y. In conclusion, it is evident that the two parameters, Y and ρ, have a direct impact on the probability distribution dataset in the δ2SATρ logic structure.

Figure 7. WFNA column representation for models in both types of logic (a) δ1SATρ, (b) δ2SATρ and recently developed logic systems.

Table 11. Maximum and minimum WFNAE results for models in both types of logic δ1SATρ, δ2SATρ and recently developed logic systems with details determined by Wilcoxon test for median.

<table>
<thead>
<tr>
<th>Model</th>
<th>Median</th>
<th>Min</th>
<th>Max</th>
<th>W</th>
<th>Model</th>
<th>Median</th>
<th>Min</th>
<th>Max</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>0.1162</td>
<td>0.0308</td>
<td>0.1373</td>
<td>120</td>
<td>Q1</td>
<td>0.0889</td>
<td>0.0270</td>
<td>0.1181</td>
<td>120</td>
</tr>
<tr>
<td>A2</td>
<td>0.0721</td>
<td>0.0374</td>
<td>0.1501</td>
<td>105</td>
<td>Q2</td>
<td>0.1058</td>
<td>0.0320</td>
<td>0.1369</td>
<td>136</td>
</tr>
<tr>
<td>A3</td>
<td>0.1506</td>
<td>0.0559</td>
<td>0.1849</td>
<td>105</td>
<td>Q3</td>
<td>0.1293</td>
<td>0.0832</td>
<td>0.1789</td>
<td>136</td>
</tr>
<tr>
<td>A4</td>
<td>0.1896</td>
<td>0.1100</td>
<td>0.2695</td>
<td>105</td>
<td>Q4</td>
<td>0.1904</td>
<td>0.0818</td>
<td>0.2400</td>
<td>136</td>
</tr>
<tr>
<td>A5</td>
<td>0.1024</td>
<td>0.0390</td>
<td>0.1486</td>
<td>120</td>
<td>Q5</td>
<td>0.1258</td>
<td>0.0782</td>
<td>0.1653</td>
<td>78</td>
</tr>
<tr>
<td>A6</td>
<td>0.1140</td>
<td>0.0448</td>
<td>0.1616</td>
<td>120</td>
<td>Q6</td>
<td>0.1353</td>
<td>0.1017</td>
<td>0.1977</td>
<td>78</td>
</tr>
<tr>
<td>A7</td>
<td>0.1409</td>
<td>0.0608</td>
<td>0.1943</td>
<td>120</td>
<td>Q7</td>
<td>0.1702</td>
<td>0.1380</td>
<td>0.2432</td>
<td>78</td>
</tr>
<tr>
<td>A8</td>
<td>0.2014</td>
<td>0.0910</td>
<td>0.2800</td>
<td>120</td>
<td>Q8</td>
<td>0.2304</td>
<td>0.1233</td>
<td>0.2931</td>
<td>78</td>
</tr>
<tr>
<td>A9</td>
<td>0.1141</td>
<td>0.0507</td>
<td>0.1646</td>
<td>190</td>
<td>Q9</td>
<td>0.1412</td>
<td>0.1056</td>
<td>0.1980</td>
<td>91</td>
</tr>
<tr>
<td>A10</td>
<td>0.1563</td>
<td>0.0520</td>
<td>0.2184</td>
<td>190</td>
<td>Q10</td>
<td>0.1760</td>
<td>0.1552</td>
<td>0.2233</td>
<td>91</td>
</tr>
<tr>
<td>A11</td>
<td>0.1906</td>
<td>0.1333</td>
<td>0.2708</td>
<td>190</td>
<td>Q11</td>
<td>0.2157</td>
<td>0.1685</td>
<td>0.2851</td>
<td>91</td>
</tr>
<tr>
<td>A12</td>
<td>0.2732</td>
<td>0.1262</td>
<td>0.3305</td>
<td>190</td>
<td>Q12</td>
<td>0.3517</td>
<td>0.2628</td>
<td>0.3945</td>
<td>91</td>
</tr>
<tr>
<td>A13</td>
<td>0.1630</td>
<td>0.1284</td>
<td>0.2131</td>
<td>78</td>
<td>Q13</td>
<td>0.2058</td>
<td>0.1744</td>
<td>0.2158</td>
<td>45</td>
</tr>
<tr>
<td>A14</td>
<td>0.1973</td>
<td>0.1673</td>
<td>0.2571</td>
<td>78</td>
<td>Q14</td>
<td>0.2386</td>
<td>0.1992</td>
<td>0.2563</td>
<td>45</td>
</tr>
<tr>
<td>A15</td>
<td>0.2676</td>
<td>0.2040</td>
<td>0.3245</td>
<td>78</td>
<td>Q15</td>
<td>0.3029</td>
<td>0.2544</td>
<td>0.3086</td>
<td>45</td>
</tr>
<tr>
<td>A16</td>
<td>0.3977</td>
<td>0.3244</td>
<td>0.4477</td>
<td>78</td>
<td>Q16</td>
<td>0.4414</td>
<td>0.4158</td>
<td>0.4691</td>
<td>45</td>
</tr>
<tr>
<td>2SAT</td>
<td>0.2300</td>
<td>0.1250</td>
<td>0.2447</td>
<td>55</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAJ2SAT</td>
<td>0.1734</td>
<td>0.0883</td>
<td>0.2141</td>
<td>36</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RAN2SAT</td>
<td>0.1287</td>
<td>0.0850</td>
<td>0.1393</td>
<td>36</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RAN3SAT</td>
<td>0.0987</td>
<td>0.0575</td>
<td>0.1088</td>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>YRAN2SAT</td>
<td>0.0610</td>
<td>0.0000</td>
<td>0.2131</td>
<td>36</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rSAT</td>
<td>0.1200</td>
<td>0.0600</td>
<td>0.1500</td>
<td>36</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: The results highlighted in yellow indicate the highest number in the column and the green indicates the smallest number in the column, (p-value < 0.00) for all models in terms of Wilcoxon test, it means reject H0.
6.2. Training Phase Capability

This phase’s objective is to evaluate the efficiency of various $\delta_2 2SAT_\rho$ structures produced in the probability logic phase, which were trained in a DHNN and minimize the logical inconsistencies using Equation (12), to obtain the correct synaptic weight. In this phase, ES obtained consistent interpretations for $\Theta_{\delta_2 2SAT_\rho}$ and derived the correct synaptic weight for the logic system. If the model arrived at an inconsistent interpretation ($E_{\Theta_{\delta_2 2SAT_\rho}} = 0$), the $DHNN - \delta 2SAT$ model will reset the whole search space and generate a new one until $\phi = \nu$. The error of the maximum fitness of logic, which is represented by the total clause from the achieved fitness, is calculated by employing $RMSE_{\text{train}}$ and $RMSE_{\text{weight}}$ to quantify the error in the training phase via Equations (24) and (25), respectively. Figures 8 and 9 show different $RMSE_{\text{train}}$, and $RMSE_{\text{weight}}$ results for both types of $\delta_2 2SAT_\rho$, when ($\nu = 100$); for both types of $\delta_2 2SAT_\rho$, RMSEtrain was described to undergo an exponential increase (logistic growth) with a rate of growth equal to $|F_i - F_{\text{desired}}|$ and a linear positive increase in $RMSE_{\text{weight}}$. According to [26], the error value in the training phase starts off low when the learning set is small because it is more difficult to fit the larger learning set. In this instance, as $\lambda_1$ rises, more iterations are required for the DHNN to locate SAT structures with satisfying interpretations, and the training phase metrics obtain 0 value when $\lambda_1$ is small. When the value of $Y$ is high, there is always low error because the structure of second-order logic helps ES by becoming satisfied ($F_i = F_{\text{desired}}$) to a greater extent than first-order logic and because the probability of finding a consistent interpretation for each $\delta_2 2SAT_\rho$ clause follows a binomial distribution, which measures the effect of flexible structure by changing in two parameters $Y$ and $\rho$ in terms of the $RMSE_{\text{train}}$ and $RMSE_{\text{weight}}$ results [24]. As shown in Figures 8 and 9, high probability of obtaining second-order $Y$ makes it easier to locate optimal interpretations [22], which means the WA method will derive the correct synaptic weight. On the other hand, when $Y$ decreases, it signifies that the probability of the first-order clauses being satisfied is very low for $2SAT$. Due to its limited number of interpretations, the non-systematic logical rule with first-order clauses reduces the cost function of the logic.

![Figure 8](image_url)

(a) (b)

**Figure 8.** RMSEtrain line representation for models in both types of logic (a) $\delta_1 2SAT_\rho$, (b) $\delta_2 2SAT_\rho$.

Table 12 records the values in Figure 8; in the line representation, it is noted for $\delta_2 2SAT_\rho$ large $RMSE_{\text{train}}$ reported for A4 (118.895) that follows group $Y = 0.6$, have the smallest number for $2SAT$ at the same time, the result of the $RMSE_{\text{train}}$ median gives us the more significant result reported by group $Y = 0.7$, whereas A8(68.5274) has a large $RMSE_{\text{train}}$ value without any effect by outlier for all $\lambda_1$; thus, when $Y$ decreases, the ES could not find a consistent interpretation for first-order logic. The low $RMSE_{\text{train}}$ median go for group $Y = 0.9$ were A14 (38.1665), which also indicates a large number of $2SAT$ that make it simpler for ES to achieve consistent interpretation. For $\delta_2 2SAT_\rho$ a large error was reported for the $Y = 0.6$ group in Q1(114.342) because of a small number of $2SAT$. 

For the median result, we note that Q3(64.7599) reported a high RMSEtrain in the same group, and group Y = 0.9 reported a lower value with respect to Q16 (41.0488), which indicates it has the same behavior for δ2SATρ; it is worth noting here that large Y and ρ have large fitness errors. It is clear in Q(4,8,12,16) that when ρ = 0.9 in both measures, that means it is difficult for ES to become satisfied for negative literals, because the extreme value for negative literal makes it difficult to achieve optimal fitness, as mentioned in [24]. Due to the limited room for searching, it is challenging for ES to be applied to large Y in small λ1. Finally, the mechanism of ES in the training phase of DHNN is only effective when λ1 is small and effected by a high number of neurons because of the non-randomized operator [24]. The training phase can be improved further by embedding a learning algorithm in a DHNN and using global and local search operators [26]. This approach may aid in the search for optimal Θδ,2SATρ interpretations and ensures that logical inconsistencies are minimized.

![Figure 9. RMSEweight column representation for models in both types of logic (a) δ12SATρ, (b) δ22SATρ.](image_url)

From Figure 9, column representation, the RMSEweight for two types δ2SATρ models help to better understand the fitness of the neuron state. Based on the results, the value of 0 was obtained in different quantities of λ1 in the interval [5,18] in different models in both types of δ2SATρ; then, the values started to fluctuate at large λ1—the maximum RMSEweight values were reported for A7 and Q3, where the values of the negative literals were large (ρ = 0.9) and where λ1 was large. In Table 13, which corresponds Figure 9, it is reported that the maximum RMSEweight values in terms of the median are A1(0.0791) and Q10(0.0548), wherein the ρ is small. In addition, a small result was reported for A16 (0.0075) and Q14(0.0048), where the negative numbers are large, which is clearly the result of the RMSEweight being affected by the fitness clauses that were measured by RMSEtrain, because the ES is could not find the interpretation for a clause with a high value of λ1 then the DHNN could not derive the correct synaptic weight by WA methods when the result was more than zero. The fluctuation in the result is because the DHNN is selected the random number for weight if EΘδ,2SATρ ≠ 0 after the number of iterations φ reaches the maximum. In conclusion, it is evident that two parameters, Y and ρ, have a direct impact on the probability distribution dataset during the testing phases.
Table 12. Maximum and minimum RMSE\textsubscript{train} results for models in both type of logic $\delta_2\text{SAT}\rho$, $\delta_2\text{SAT}\rho$ details by Wilcoxon test for median.

<table>
<thead>
<tr>
<th>Y</th>
<th>Model</th>
<th>Min</th>
<th>Max</th>
<th>Median</th>
<th>W</th>
<th>Model</th>
<th>Min</th>
<th>Max</th>
<th>Median</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.6</td>
<td>A1</td>
<td>0</td>
<td>114.965</td>
<td>65.6734</td>
<td>78</td>
<td>Q1</td>
<td>0.0000</td>
<td>114.342</td>
<td>61.3433</td>
<td>91</td>
</tr>
<tr>
<td>A2</td>
<td>0</td>
<td>113.345</td>
<td>62.2179</td>
<td>78</td>
<td>Q2</td>
<td>0.0000</td>
<td>113.745</td>
<td>59.2203</td>
<td>91</td>
<td></td>
</tr>
<tr>
<td>A3</td>
<td>0</td>
<td>116.34</td>
<td>62.4112</td>
<td>78</td>
<td>Q3</td>
<td>0.0000</td>
<td>110.381</td>
<td>64.7599</td>
<td>91</td>
<td></td>
</tr>
<tr>
<td>A4</td>
<td>0</td>
<td>118.895</td>
<td>56.8860</td>
<td>78</td>
<td>Q4</td>
<td>0.0000</td>
<td>106.132</td>
<td>64.1423</td>
<td>105</td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>A5</td>
<td>0</td>
<td>111.853</td>
<td>65.3299</td>
<td>91</td>
<td>Q5</td>
<td>0.0000</td>
<td>93.0699</td>
<td>59.5461</td>
<td>55</td>
</tr>
<tr>
<td>A6</td>
<td>0</td>
<td>113.982</td>
<td>67.6314</td>
<td>78</td>
<td>Q6</td>
<td>0.0000</td>
<td>93.8776</td>
<td>57.8688</td>
<td>55</td>
<td></td>
</tr>
<tr>
<td>A7</td>
<td>0</td>
<td>116.327</td>
<td>63.0476</td>
<td>91</td>
<td>Q7</td>
<td>0.0000</td>
<td>101.459</td>
<td>58.5859</td>
<td>91</td>
<td></td>
</tr>
<tr>
<td>A8</td>
<td>0</td>
<td>118.617</td>
<td>68.5274</td>
<td>91</td>
<td>Q8</td>
<td>0.0000</td>
<td>102.528</td>
<td>57.0719</td>
<td>55</td>
<td></td>
</tr>
<tr>
<td>0.8</td>
<td>A9</td>
<td>0</td>
<td>100.822</td>
<td>53.1695</td>
<td>120</td>
<td>Q9</td>
<td>0.0000</td>
<td>84.5754</td>
<td>41.8091</td>
<td>55</td>
</tr>
<tr>
<td>A10</td>
<td>0</td>
<td>101.922</td>
<td>50.3786</td>
<td>120</td>
<td>Q10</td>
<td>0.0000</td>
<td>85.4868</td>
<td>41.6653</td>
<td>66</td>
<td></td>
</tr>
<tr>
<td>A11</td>
<td>0</td>
<td>103.015</td>
<td>50.6162</td>
<td>120</td>
<td>Q11</td>
<td>0.0000</td>
<td>82.3286</td>
<td>43.8634</td>
<td>55</td>
<td></td>
</tr>
<tr>
<td>A12</td>
<td>0</td>
<td>103.388</td>
<td>51.0294</td>
<td>136</td>
<td>Q12</td>
<td>0.0000</td>
<td>87.8521</td>
<td>44.6654</td>
<td>55</td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td>A13</td>
<td>0</td>
<td>78.5366</td>
<td>38.3444</td>
<td>45</td>
<td>Q13</td>
<td>5.8309</td>
<td>74.3707</td>
<td>41.7971</td>
<td>45</td>
</tr>
<tr>
<td>A14</td>
<td>0</td>
<td>72.9383</td>
<td>38.16665</td>
<td>55</td>
<td>Q14</td>
<td>5.0000</td>
<td>71.2881</td>
<td>41.6413</td>
<td>45</td>
<td></td>
</tr>
<tr>
<td>A15</td>
<td>0</td>
<td>76.0526</td>
<td>41.35185</td>
<td>55</td>
<td>Q15</td>
<td>1.0000</td>
<td>68.2202</td>
<td>42.4853</td>
<td>55</td>
<td></td>
</tr>
<tr>
<td>A16</td>
<td>0</td>
<td>79.4921</td>
<td>41.86575</td>
<td>55</td>
<td>Q16</td>
<td>2.0000</td>
<td>73.2871</td>
<td>41.0488</td>
<td>45</td>
<td></td>
</tr>
</tbody>
</table>

Note: The results highlighted in yellow indicate the highest number in the column and the green indicates the smallest number in the column, ($p$-value < 0.00) for all models in terms of Wilcoxon test, it means reject $H_0$.

Table 13. Maximum and minimum RMSE\textsubscript{weight} results for models in both types of logic $\delta_2\text{SAT}\rho$, $\delta_2\text{SAT}\rho$ and details by Wilcoxon test for median.

<table>
<thead>
<tr>
<th>Y</th>
<th>Model</th>
<th>Min</th>
<th>Max</th>
<th>Median</th>
<th>W</th>
<th>Model</th>
<th>Min</th>
<th>Max</th>
<th>Median</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.6</td>
<td>A1</td>
<td>0</td>
<td>0.2458</td>
<td>0.0791</td>
<td>91</td>
<td>Q1</td>
<td>0.0000</td>
<td>0.1978</td>
<td>0.0295</td>
<td>55</td>
</tr>
<tr>
<td>A2</td>
<td>0</td>
<td>0.2567</td>
<td>0.0485</td>
<td>91</td>
<td>Q2</td>
<td>0.0000</td>
<td>0.2340</td>
<td>0.0241</td>
<td>55</td>
<td></td>
</tr>
<tr>
<td>A3</td>
<td>0</td>
<td>0.2943</td>
<td>0.0321</td>
<td>78</td>
<td>Q3</td>
<td>0.0000</td>
<td>0.2365</td>
<td>0.0145</td>
<td>91</td>
<td></td>
</tr>
<tr>
<td>A4</td>
<td>0</td>
<td>0.2642</td>
<td>0.0242</td>
<td>78</td>
<td>Q4</td>
<td>0.0000</td>
<td>0.2350</td>
<td>0.0193</td>
<td>105</td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>A5</td>
<td>0</td>
<td>0.1134</td>
<td>0.0364</td>
<td>136</td>
<td>Q5</td>
<td>0.0000</td>
<td>0.1700</td>
<td>0.0093</td>
<td>55</td>
</tr>
<tr>
<td>A6</td>
<td>0</td>
<td>0.0887</td>
<td>0.0397</td>
<td>78</td>
<td>Q6</td>
<td>0.0000</td>
<td>0.2077</td>
<td>0.0304</td>
<td>91</td>
<td></td>
</tr>
<tr>
<td>A7</td>
<td>0</td>
<td>0.3374</td>
<td>0.0261</td>
<td>91</td>
<td>Q7</td>
<td>0.0000</td>
<td>0.1389</td>
<td>0.0319</td>
<td>91</td>
<td></td>
</tr>
<tr>
<td>A8</td>
<td>0</td>
<td>0.2591</td>
<td>0.0138</td>
<td>78</td>
<td>Q8</td>
<td>0.0000</td>
<td>0.1598</td>
<td>0.0134</td>
<td>55</td>
<td></td>
</tr>
<tr>
<td>0.8</td>
<td>A9</td>
<td>0</td>
<td>0.2277</td>
<td>0.0175</td>
<td>91</td>
<td>Q9</td>
<td>0.0000</td>
<td>0.1268</td>
<td>0.0321</td>
<td>55</td>
</tr>
<tr>
<td>A10</td>
<td>0</td>
<td>0.1821</td>
<td>0.0110</td>
<td>120</td>
<td>Q10</td>
<td>0.0000</td>
<td>0.1012</td>
<td>0.0548</td>
<td>55</td>
<td></td>
</tr>
<tr>
<td>A11</td>
<td>0</td>
<td>0.2023</td>
<td>0.0228</td>
<td>120</td>
<td>Q11</td>
<td>0.0000</td>
<td>0.1079</td>
<td>0.0396</td>
<td>55</td>
<td></td>
</tr>
<tr>
<td>A12</td>
<td>0</td>
<td>0.1265</td>
<td>0.0135</td>
<td>120</td>
<td>Q12</td>
<td>0.0000</td>
<td>0.0368</td>
<td>0.0207</td>
<td>66</td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td>A13</td>
<td>0</td>
<td>0.0790</td>
<td>0.0389</td>
<td>45</td>
<td>Q13</td>
<td>0.0004</td>
<td>0.0639</td>
<td>0.0178</td>
<td>45</td>
</tr>
<tr>
<td>A14</td>
<td>0</td>
<td>0.0990</td>
<td>0.0250</td>
<td>55</td>
<td>Q14</td>
<td>0.0016</td>
<td>0.0358</td>
<td>0.0048</td>
<td>45</td>
<td></td>
</tr>
<tr>
<td>A15</td>
<td>0</td>
<td>0.0423</td>
<td>0.0213</td>
<td>55</td>
<td>Q15</td>
<td>0.0005</td>
<td>0.0246</td>
<td>0.0171</td>
<td>45</td>
<td></td>
</tr>
<tr>
<td>A16</td>
<td>0</td>
<td>0.0427</td>
<td>0.0075</td>
<td>55</td>
<td>Q16</td>
<td>0.0005</td>
<td>0.0885</td>
<td>0.0329</td>
<td>45</td>
<td></td>
</tr>
</tbody>
</table>

Note: The results highlighted in yellow indicate the highest number in the column and the green indicates the smallest number in the column, ($p$-value < 0.00) for all models in terms of Wilcoxon test, it means reject $H_0$.

6.3. Testing Phase Capability

Optimal testing phase is achieved when $E_{Theta} = 0$ retrieved optimal synaptic weight, after DHNN $\rightarrow \delta_2\text{SAT} \rho$ completing checking clause satisfaction and generating optimal synaptic weight through the WA method. The final state of the neuron will converge towards the global minimum energy. It is important to evaluate testing phase because DHNN frequently produces similar final...
neuron states as opposed to novel final neuron states [55]. Therefore, we compare the $\delta_2SAT\rho$ logic with the recent logic systems by global minima ratio matric. If the model is unable to reach a global solution, this indicates that it is trapped in a local solution, which makes it impossible to determine whether the proposed DHNN – $\delta 2SAT$ is satisfied or not.

Figure 10, column representation, shows the global minima ratio results, calculated by Equation (26) for two types $\delta_2SAT\rho$ and state-of-the-art logic systems, without considering any optimizer, to assess the actual testing phase capability for of DHNN – $\delta 2SAT$. Where the optimal result for global minima ratio $R_G$ is 1, we can note in Figure 10 that all are capable of retrieving the optimal synaptic weight values in small $\lambda_1$ and then it decreases linearity with large $\lambda_1$, because the ES is unable to manage synaptic weight in the training phase and will be susceptible to retrieving non-optimal neuron states and ensnared in local minima. A model’s ability to achieve maximum global minima ratio demonstrates that the suggested SAT is effectively integrated into DHNN. Maximum global minima ratio reported for YRAN2SAT, rSAT, and (A1, A11, Q11) models in $\delta_2SAT\rho$. The reason for YRAN2SAT recorded the high global minima ratio for small $\lambda_1$ [26] because the flexibility in the structure offers an accurate result. Table 14 gives the Figure 10 numerical result, from the $R_G$ median results without effect from the outliers, note both type $\delta_2SAT\rho$ achieve near result to other latest logic systems. High median goes for MAJ2SAT because the structure of logic that (2SAT,3SAT) [23], also the fare literals state represent in rSAT [24] make it achieved highly $R_G$. Based on Table 14, from $R_G$ median, there is a high effect for two parameters $Y$ and $\rho$ in $\delta_2SAT\rho$, small $\lambda_1$ in the DHNN for small $Y$ and $\rho$ can retrieval the right synaptic weight such as (A1,Q1), but from median, the high $Y$ and $\rho$ achieved more global minimum than other such as A(13,14,15), Q(9,10,13,14). It can say the proposed models showcased the efficiency of $\delta_2SAT\rho$ to control DHNN as a symbolic structure that causes network convergence. Since the local field in Equation (15) drives the neuron’s final state in accordance with the behavior of the second and first-order clause, it exhibits the same behavior as the non-systematic RAN2SAT structure presented by [20].

![Figure 10](image_url)

**Figure 10.** Column representation for models in both types of logic (a) $\delta_1SAT\rho$, (b) $\delta_2SAT\rho$ and recently developed logic systems.
Table 14. Maximum $R_G$ results for models in both type of logic $\delta_12SAT\rho$, $\delta_22SAT\rho$ and RAN2SAT details by Wilcoxon test for median.

<table>
<thead>
<tr>
<th>Model</th>
<th>Median</th>
<th>Max</th>
<th>W</th>
<th>Model</th>
<th>Median</th>
<th>Max</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>0.0200</td>
<td>0.9338</td>
<td>66</td>
<td>Q1</td>
<td>0.0197</td>
<td>0.9321</td>
<td>105</td>
</tr>
<tr>
<td>A2</td>
<td>0.0133</td>
<td>0.9125</td>
<td>105</td>
<td>Q2</td>
<td>0.0152</td>
<td>0.9191</td>
<td>136</td>
</tr>
<tr>
<td>A3</td>
<td>0.0194</td>
<td>0.7945</td>
<td>91</td>
<td>Q3</td>
<td>0.0097</td>
<td>0.7479</td>
<td>136</td>
</tr>
<tr>
<td>A4</td>
<td>0.0077</td>
<td>0.7468</td>
<td>91</td>
<td>Q4</td>
<td>0.0099</td>
<td>0.7603</td>
<td>105</td>
</tr>
<tr>
<td>A5</td>
<td>0.0061</td>
<td>0.8855</td>
<td>91</td>
<td>Q5</td>
<td>0.0185</td>
<td>0.8322</td>
<td>78</td>
</tr>
<tr>
<td>A6</td>
<td>0.0096</td>
<td>0.9008</td>
<td>120</td>
<td>Q6</td>
<td>0.0299</td>
<td>0.7276</td>
<td>78</td>
</tr>
<tr>
<td>A7</td>
<td>0.0092</td>
<td>0.7970</td>
<td>120</td>
<td>Q7</td>
<td>0.0170</td>
<td>0.7025</td>
<td>78</td>
</tr>
<tr>
<td>A8</td>
<td>0.0058</td>
<td>0.7551</td>
<td>105</td>
<td>Q8</td>
<td>0.0162</td>
<td>0.8546</td>
<td>78</td>
</tr>
<tr>
<td>A9</td>
<td>0.0563</td>
<td>0.9093</td>
<td>190</td>
<td>Q9</td>
<td>0.1084</td>
<td>0.7822</td>
<td>91</td>
</tr>
<tr>
<td>A10</td>
<td>0.0437</td>
<td>0.9009</td>
<td>153</td>
<td>Q10</td>
<td>0.1014</td>
<td>0.6888</td>
<td>91</td>
</tr>
<tr>
<td>A11</td>
<td>0.0200</td>
<td>0.9338</td>
<td>66</td>
<td>Q11</td>
<td>0.0740</td>
<td>0.6787</td>
<td>91</td>
</tr>
<tr>
<td>A12</td>
<td>0.0204</td>
<td>0.7479</td>
<td>171</td>
<td>Q12</td>
<td>0.0240</td>
<td>0.5382</td>
<td>91</td>
</tr>
<tr>
<td>A13</td>
<td>0.1576</td>
<td>0.7467</td>
<td>78</td>
<td>Q13</td>
<td>0.0945</td>
<td>0.5113</td>
<td>45</td>
</tr>
<tr>
<td>A14</td>
<td>0.1226</td>
<td>0.6243</td>
<td>78</td>
<td>Q14</td>
<td>0.0864</td>
<td>0.4968</td>
<td>45</td>
</tr>
<tr>
<td>A15</td>
<td>0.0815</td>
<td>0.6197</td>
<td>78</td>
<td>Q15</td>
<td>0.0560</td>
<td>0.4055</td>
<td>45</td>
</tr>
<tr>
<td>A16</td>
<td>0.0434</td>
<td>0.4152</td>
<td>78</td>
<td>Q16</td>
<td>0.0261</td>
<td>0.2097</td>
<td>45</td>
</tr>
<tr>
<td>2SAT</td>
<td>0.4190</td>
<td>0.8799</td>
<td>55</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAJ2SAT</td>
<td>0.5050</td>
<td>0.8076</td>
<td>36</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RAN2SAT</td>
<td>0.0172</td>
<td>0.8756</td>
<td>36</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RAN3SAT</td>
<td>0.2580</td>
<td>0.8213</td>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>YRAN2SAT</td>
<td>0.0178</td>
<td>0.9488</td>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rSAT</td>
<td>0.0000</td>
<td>0.9200</td>
<td>28</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: The Yellow highlighted to indicate the highest number in the column, ($p$-value < 0.00) for all models in terms of Wilcoxon test, it means reject $H_0$.

The purpose of finding the RMSEnergy in Equation (27) is to calculate the difference between the final energy and the absolute minimum energy, as stated in condition Equation (18). indicates whether or not the solutions produced by $DHNN - \delta_2SAT\rho$ are optimal, it must assess the flexibility of $\delta_2\gamma SAT\rho$ by determining the value of RMSEnergy. Based on Figure 11 column representation, was reported to small $\lambda_1$ achieve less RMAEenergy value for all models, which indicates a successful convergence towards the optimal final neuron state, after which the final energy difference fluctuates as the number of $\lambda_1$ increased. This phenomenon occurs as a result of the decreased probability of receiving cost function $E_{\Theta_2\delta_2SAT} = 0$, as clear in RMSEnergy which leads to higher energy, and $DHNN - \delta_2SAT$’s ineffective learning strategy. As the number of $\lambda_1$ increases, some synaptic weights become suboptimal, resulting in final neuron states stuck in local minimum energy. Additionally, Sathasivam [18] claims that during the DHNN testing phase, suboptimal neuron updates are what caused the local minimum energy to exist. Suboptimal neuron updates in this situation will result in more incomplete sentences, which raises the energy gap. When the logical formulation containing 2SAT was incorporated into $DHNN - \delta_2SAT$ we said the $\delta_2\rho SAT\rho$ behaved like the traditional non-systematic logical rule RAN2SAT. As shown in Figure 11 it can be observed that the adverse impact of negative literal with high number of $\lambda_1$ in logic where A4 Q12 recorded the highest value of RMSEnergy and A1 Q1 when number of $\lambda_1$ small is the opposite, from Table 15 gives from the Figure 11 the median of RMSEnergy gives us the accurate result where the small median go for A13, Q9 with low value parameter $\rho$ and A8, A16 with high value parameter $\rho$ give high RMSEnergy error. This demonstrates that when most neuron states are negative, tend to converge towards local minimum energy. In conclusion, it is evident that two parameters, Y and $\rho$, have a direct impact on the probability distribution dataset during the testing phases.
Figure 11. RMSEenergy column representation for models in both types of logic (a) $\delta_1 2SAT_\rho$, (b) $\delta_2 2SAT_\rho$.

Table 15. Maximum and minimum RMSEenergy results for models in both type of logic $\delta_1 2SAT_\rho$, $\delta_2 2SAT_\rho$ details by Wilcoxon test for median.

<table>
<thead>
<tr>
<th>Model</th>
<th>Median</th>
<th>Min</th>
<th>Max</th>
<th>W</th>
<th>Model</th>
<th>Median</th>
<th>Min</th>
<th>Max</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>1.8703</td>
<td>0.2573</td>
<td>3.0905</td>
<td>120</td>
<td>Q1</td>
<td>1.9596</td>
<td>0.2606</td>
<td>2.7905</td>
<td>120</td>
</tr>
<tr>
<td>A2</td>
<td>1.9854</td>
<td>0.2958</td>
<td>2.7368</td>
<td>105</td>
<td>Q2</td>
<td>1.9339</td>
<td>0.2844</td>
<td>2.7724</td>
<td>120</td>
</tr>
<tr>
<td>A3</td>
<td>2.1074</td>
<td>0.4555</td>
<td>3.1800</td>
<td>105</td>
<td>Q3</td>
<td>2.0517</td>
<td>0.5021</td>
<td>3.1297</td>
<td>136</td>
</tr>
<tr>
<td>A4</td>
<td>2.3208</td>
<td>0.5032</td>
<td>3.9210</td>
<td>105</td>
<td>Q4</td>
<td>2.1874</td>
<td>0.4896</td>
<td>3.4820</td>
<td>136</td>
</tr>
<tr>
<td>A5</td>
<td>1.9285</td>
<td>0.3384</td>
<td>2.7783</td>
<td>120</td>
<td>Q5</td>
<td>1.7077</td>
<td>0.4096</td>
<td>2.6531</td>
<td>78</td>
</tr>
<tr>
<td>A6</td>
<td>2.1017</td>
<td>0.3150</td>
<td>2.8161</td>
<td>120</td>
<td>Q6</td>
<td>1.8026</td>
<td>0.5526</td>
<td>2.9892</td>
<td>78</td>
</tr>
<tr>
<td>A7</td>
<td>2.2435</td>
<td>0.4680</td>
<td>3.0432</td>
<td>120</td>
<td>Q7</td>
<td>2.1137</td>
<td>0.5819</td>
<td>3.5014</td>
<td>78</td>
</tr>
<tr>
<td>A8</td>
<td>2.5302</td>
<td>0.4949</td>
<td>3.4379</td>
<td>120</td>
<td>Q8</td>
<td>2.2589</td>
<td>0.7982</td>
<td>2.7102</td>
<td>78</td>
</tr>
<tr>
<td>A9</td>
<td>1.6867</td>
<td>0.3012</td>
<td>2.2621</td>
<td>190</td>
<td>Q9</td>
<td>1.4306</td>
<td>0.4812</td>
<td>2.5811</td>
<td>91</td>
</tr>
<tr>
<td>A10</td>
<td>1.5883</td>
<td>0.3148</td>
<td>2.9905</td>
<td>190</td>
<td>Q10</td>
<td>1.5079</td>
<td>0.6127</td>
<td>2.7328</td>
<td>91</td>
</tr>
<tr>
<td>A11</td>
<td>1.7648</td>
<td>0.5055</td>
<td>3.0828</td>
<td>190</td>
<td>Q11</td>
<td>1.6259</td>
<td>0.6307</td>
<td>3.1733</td>
<td>91</td>
</tr>
<tr>
<td>A12</td>
<td>2.0895</td>
<td>0.5021</td>
<td>3.5483</td>
<td>190</td>
<td>Q12</td>
<td>2.4068</td>
<td>0.8465</td>
<td>3.6066</td>
<td>91</td>
</tr>
<tr>
<td>A13</td>
<td>1.4357</td>
<td>0.5445</td>
<td>2.3507</td>
<td>78</td>
<td>Q13</td>
<td>1.6090</td>
<td>0.8600</td>
<td>2.3148</td>
<td>45</td>
</tr>
<tr>
<td>A14</td>
<td>1.5348</td>
<td>0.7222</td>
<td>2.7120</td>
<td>78</td>
<td>Q14</td>
<td>1.7685</td>
<td>0.9484</td>
<td>2.4639</td>
<td>45</td>
</tr>
<tr>
<td>A15</td>
<td>1.7717</td>
<td>0.7171</td>
<td>3.0085</td>
<td>78</td>
<td>Q15</td>
<td>2.0352</td>
<td>1.1185</td>
<td>2.6995</td>
<td>45</td>
</tr>
<tr>
<td>A16</td>
<td>2.2963</td>
<td>1.0788</td>
<td>3.5845</td>
<td>78</td>
<td>Q16</td>
<td>2.5315</td>
<td>1.6824</td>
<td>3.3637</td>
<td>45</td>
</tr>
</tbody>
</table>

Note: The results highlighted in yellow indicate the highest number in the column and the green indicates the smallest number in the column, ($p$-value < 0.00) for all models in terms of Wilcoxon test, it means reject $H_0$.

6.4. Similarity Index Analysis

For final neurons’ quality states only compare both type of $\delta_2 2SAT_\rho$ with RAN2SAT because $\delta_2 2SAT_\rho$ consider the enhancement and developing for RAN2SAT, also, they have the same structure behavior, we tested the variation introduced by the testing phase for $\delta_2 2SAT_\rho$ models and final neurons’ quality state compared with RAN2SAT, where the degree of state redundancy for the DHNN...
model training phases is indicated by the similarity index of the final neuron state. A standard has been introduced indexing metrics, which is Sokal index, and consider the effective metric known as the ratio of total neurons variation $R_{tv}$.

Firstly, consider the lower Sokal in Equation (30) in the similarity index matrices indicates that the final neuron states obtained are highly distinct to the benchmark states. According to Figure 12, a column representation to both types of $\delta_2 2SAT\rho$ reported low values, which imply higher more variety solution than other recorded by $A_{16}, Q_{16m}$ but $Q_{1}, A_{5}$ recorded high value, due to parameter $\rho$. Table 16 translates Figure 12, which clarifies numerically, where the $A_{16}, Q_{16}$ reported low median value. Where all logic has the $\rho = 0.9$ and $Y = 0.9$ record low value, it indicates that there is more negative neuron and less first-order logic provides the final neuron state and the benchmark state distinction as shown in blue numbers in Table 16, $Q, A_{(4,8,12,16)}$. In other words, the low negativity and greater representation for first-order logic give us a high Sokal, as shown in $Q, A_{(1,5,9,13)}$ with a red number.

![Figure 12. Sokal column representation for models in both types of logic (a) $\delta_1 2SAT\rho$, (b) $\delta_2 2SAT\rho$ and RAN2SAT.](image)

Secondly, the effective parameter known as the ratio total variation of neurons $R_{tv}$. In Equation (31). From the Figure 13 its clearly a column representation to both types giving us different number of variation solution for different number of $\lambda_1$ because of the effect of two parameter $Y$ and $\rho$ in the training phase. The highly oscillation recorded for $\delta_1 2SAT\rho$ in $14 < \lambda_1 < 20$ and $14 < \lambda_1 < 26$ for $\delta_1 2SAT\rho$ models and the highest oscillation value is recorded for $A_{16}$ in $17 < \lambda_1 < 20$. For $\delta_2 2SAT\rho$. The highly oscillation recorded in $14 < \lambda_1 < 26$ for $\delta_1 2SAT\rho$ models and the highest oscillation value is traced for $Q_{15}$ in $13 < \lambda_1 < 23$. At the same time both type of $\delta_2 2SAT\rho$ models affected by a number of neurons, they start the ups and downs in different $\lambda_1$ according to the effect of two parameter $Y$ and $\rho$. The total oscillation for some of models is rich to zero when $\lambda_1 < 5, \lambda_1 > 39$ such as $A_{(1,3,4,5,8,10,12)}$ and so low for others models $\delta_1 2SAT\rho$, also $Q_{1}, Q_{4}$ when $\lambda_1 < 5, \lambda_1 > 35$ for $\delta_2 2SAT\rho$, we can be said there are no significant variations for high than 37, also we can note here the effect of $Y$, it can’t achieve the global solution for low $Y$ because the ES will disturb the $\delta_2 2SAT\rho$ model in order to reach the optimal training phase (known as learn inconsistent interpretation), from Figure 10, global solutions acquired by $\delta_2 2SAT\rho$ models grow as $\lambda_1$ decrease as introduce previously. Table 17 for numerical result for Figure 13 note the effect of increase $\rho$ where the logic has $\rho > 0.7$ recorded the highest high number of $R_{tv}$, where the highest variation go for $A_{16}$ (0.2149) and $Q_{15}$ (0.2084), and also can see the $\delta_2 2SAT\rho$ record high $R_{tv}$, than $\delta_2 2SAT\rho$ in general, the reason her the $\delta_2 2SAT\rho$ give less number of first-order logic than $\delta_1 2SAT\rho$ for the same $Y$ that was mention previously in Table 1, then the ES will deal with fewer numbers for first-order logic where it difficult to reach the optimal training phase. Moreover, Figure 13 presents the reason for the decrease when
\( \lambda_1 \) increases the hard achieved global solution. It was observed that the RAN2SAT behave similarly to the \( \delta_1 \text{SAT}_\rho \), with a high \( R_{tv} \) recorded of (0.1764) at the same time as its increase in the interval \( 13 < \lambda_1 < 42 \) and then decrease with a high \( \lambda_1 \). The impact of the global minimum solution \( R_{tv} \) is related to the number of neurons. As \( \lambda_1 \) rises, the probability of the number of global solutions reduces. We can conclude from the above results that \( R_{tv} \) is related to the occurrence of other neuron states that lead to global minimum solutions in other domain adaptations [22].

Table 16. Maximum and minimum Sokal results for models in both types of logic \( \delta_1 \text{SAT}_\rho \), \( \delta_2 \text{SAT}_\rho \) and RAN2SAT details by Wilcoxon test for median.

<table>
<thead>
<tr>
<th>Model</th>
<th>Median</th>
<th>Min</th>
<th>Max</th>
<th>W</th>
<th>Model</th>
<th>Median</th>
<th>Min</th>
<th>Max</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>0.6483</td>
<td>0.6367</td>
<td>0.7667</td>
<td>120</td>
<td>Q1</td>
<td>0.6601</td>
<td>0.6350</td>
<td>0.7225</td>
<td>120</td>
</tr>
<tr>
<td>A2</td>
<td>0.6771</td>
<td>0.6212</td>
<td>0.7512</td>
<td>105</td>
<td>Q2</td>
<td>0.6527</td>
<td>0.6337</td>
<td>0.6887</td>
<td>136</td>
</tr>
<tr>
<td>A3</td>
<td>0.6335</td>
<td>0.6020</td>
<td>0.7074</td>
<td>105</td>
<td>Q3</td>
<td>0.6341</td>
<td>0.5993</td>
<td>0.6747</td>
<td>136</td>
</tr>
<tr>
<td>A4</td>
<td>0.6286</td>
<td>0.5988</td>
<td>0.6447</td>
<td>105</td>
<td>Q4</td>
<td>0.6189</td>
<td>0.5884</td>
<td>0.6757</td>
<td>136</td>
</tr>
<tr>
<td>A5</td>
<td>0.6632</td>
<td>0.6360</td>
<td>0.7751</td>
<td>120</td>
<td>Q5</td>
<td>0.6509</td>
<td>0.6324</td>
<td>0.6810</td>
<td>78</td>
</tr>
<tr>
<td>A6</td>
<td>0.6556</td>
<td>0.6329</td>
<td>0.7509</td>
<td>120</td>
<td>Q6</td>
<td>0.6385</td>
<td>0.5965</td>
<td>0.6730</td>
<td>78</td>
</tr>
<tr>
<td>A7</td>
<td>0.6369</td>
<td>0.6000</td>
<td>0.7074</td>
<td>120</td>
<td>Q7</td>
<td>0.6271</td>
<td>0.5997</td>
<td>0.6423</td>
<td>78</td>
</tr>
<tr>
<td>A8</td>
<td>0.6232</td>
<td>0.6024</td>
<td>0.6790</td>
<td>120</td>
<td>Q8</td>
<td>0.6095</td>
<td>0.5593</td>
<td>0.6662</td>
<td>78</td>
</tr>
<tr>
<td>A9</td>
<td>0.6594</td>
<td>0.6374</td>
<td>0.7266</td>
<td>190</td>
<td>Q9</td>
<td>0.6536</td>
<td>0.6284</td>
<td>0.6709</td>
<td>91</td>
</tr>
<tr>
<td>A10</td>
<td>0.6493</td>
<td>0.5974</td>
<td>0.6915</td>
<td>190</td>
<td>Q10</td>
<td>0.6276</td>
<td>0.5960</td>
<td>0.6614</td>
<td>91</td>
</tr>
<tr>
<td>A11</td>
<td>0.6288</td>
<td>0.5987</td>
<td>0.6547</td>
<td>190</td>
<td>Q11</td>
<td>0.6051</td>
<td>0.5850</td>
<td>0.6322</td>
<td>91</td>
</tr>
<tr>
<td>A12</td>
<td>0.6027</td>
<td>0.5702</td>
<td>0.6702</td>
<td>190</td>
<td>Q12</td>
<td>0.5790</td>
<td>0.5468</td>
<td>0.6096</td>
<td>91</td>
</tr>
<tr>
<td>A13</td>
<td>0.6453</td>
<td>0.6377</td>
<td>0.6651</td>
<td>78</td>
<td>Q13</td>
<td>0.6385</td>
<td>0.6167</td>
<td>0.6459</td>
<td>45</td>
</tr>
<tr>
<td>A14</td>
<td>0.6277</td>
<td>0.6031</td>
<td>0.6475</td>
<td>78</td>
<td>Q14</td>
<td>0.6228</td>
<td>0.6058</td>
<td>0.6335</td>
<td>45</td>
</tr>
<tr>
<td>A15</td>
<td>0.6080</td>
<td>0.5770</td>
<td>0.6227</td>
<td>78</td>
<td>Q15</td>
<td>0.5977</td>
<td>0.5836</td>
<td>0.6060</td>
<td>45</td>
</tr>
<tr>
<td>A16</td>
<td>0.5783</td>
<td>0.5517</td>
<td>0.5890</td>
<td>78</td>
<td>Q16</td>
<td>0.5665</td>
<td>0.5458</td>
<td>0.5863</td>
<td>45</td>
</tr>
<tr>
<td>RAN2SAT</td>
<td>0.6379</td>
<td>0.6007</td>
<td>0.6564</td>
<td>55</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Note:** The results highlighted in yellow indicate the highest number in the column and the green indicates the smallest number in the column, (p-value < 0.00) for all models in terms of Wilcoxon test, it means reject \( H_0 \).

Figure 13. Column representation for models in both types of logic (a) \( \delta_1 \text{SAT}_\rho \), (b) \( \delta_2 \text{SAT}_\rho \) and RAN2SAT.
Table 17. Maximum and minimum $R_{tv}$ results for models in both types of logic $\delta_12SAT\rho$, $\delta_22SAT\rho$ and RAN2SAT.

<table>
<thead>
<tr>
<th>Model</th>
<th>Min</th>
<th>Max</th>
<th>Model</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>0.0000</td>
<td>0.0246</td>
<td>Q1</td>
<td>0.0000</td>
<td>0.0364</td>
</tr>
<tr>
<td>A2</td>
<td>0.0003</td>
<td>0.0573</td>
<td>Q2</td>
<td>0.0002</td>
<td>0.0712</td>
</tr>
<tr>
<td>A3</td>
<td>0.0000</td>
<td>0.0962</td>
<td>Q3</td>
<td>0.0001</td>
<td>0.1327</td>
</tr>
<tr>
<td>A4</td>
<td>0.0000</td>
<td>0.1469</td>
<td>Q4</td>
<td>0.0000</td>
<td>0.1840</td>
</tr>
<tr>
<td>A5</td>
<td>0.0000</td>
<td>0.0347</td>
<td>Q5</td>
<td>0.0003</td>
<td>0.0804</td>
</tr>
<tr>
<td>A6</td>
<td>0.0004</td>
<td>0.0624</td>
<td>Q6</td>
<td>0.0013</td>
<td>0.0926</td>
</tr>
<tr>
<td>A7</td>
<td>0.0001</td>
<td>0.1031</td>
<td>Q7</td>
<td>0.0006</td>
<td>0.1219</td>
</tr>
<tr>
<td>A8</td>
<td>0.0000</td>
<td>0.1423</td>
<td>Q8</td>
<td>0.0002</td>
<td>0.1587</td>
</tr>
<tr>
<td>A9</td>
<td>0.0005</td>
<td>0.0590</td>
<td>Q9</td>
<td>0.0028</td>
<td>0.0929</td>
</tr>
<tr>
<td>A10</td>
<td>0.0000</td>
<td>0.1067</td>
<td>Q10</td>
<td>0.0007</td>
<td>0.1225</td>
</tr>
<tr>
<td>A11</td>
<td>0.0002</td>
<td>0.1413</td>
<td>Q11</td>
<td>0.0003</td>
<td>0.1596</td>
</tr>
<tr>
<td>A12</td>
<td>0.0000</td>
<td>0.1880</td>
<td>Q12</td>
<td>0.0002</td>
<td>0.1979</td>
</tr>
<tr>
<td>A13</td>
<td>0.0027</td>
<td>0.0984</td>
<td>Q13</td>
<td>0.002</td>
<td>0.1581</td>
</tr>
<tr>
<td>A14</td>
<td>0.0009</td>
<td>0.1288</td>
<td>Q14</td>
<td>0.0047</td>
<td>0.1542</td>
</tr>
<tr>
<td>A15</td>
<td>0.0015</td>
<td>0.1678</td>
<td>Q15</td>
<td>0.0033</td>
<td>0.2084</td>
</tr>
<tr>
<td>A16</td>
<td>0.0008</td>
<td>0.2149</td>
<td>Q16</td>
<td>0.0018</td>
<td>0.1987</td>
</tr>
<tr>
<td>RAN2SAT</td>
<td>0.0004</td>
<td>0.1764</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: The results highlighted in yellow indicate the highest number in the column and the green indicates the smallest number in the column.

6.5. Synaptic Weight Analysis

The mean is important because it signifies the location of the dataset’s centre value, it contains information from each observation in a dataset. When a dataset is skewed or contains outliers, the mean may be untrue. We are utilizing various statistical tests to aid us comprehend the behaviors of synaptic weight to deduce information about the performance of logic in the training phases for further inquiry in synaptic weight distribution. The descriptive statistic of mean synaptic weight is a novel perspective in synaptic weight analysis, and we consider the mean of full logic to obtain a meaningful result in this analysis of our study by using the following formula:

\[
\text{Mean of } \delta_2\text{SAT} = \frac{\sum_{i=1}^{n} W_{ri} + \sum_{j=1}^{n} W_{rj} + \sum_{j=1}^{n} W_{rjrj+1}}{\lambda_1}
\]  

(37)

where $W_{ri} = \pm 0.5$ synaptic weight for first-order logic, $W_{rj} = \pm 0.25$ synaptic weight for second-order logic literals, $W_{rjrj+1} = \pm 0.25$ synaptic weight for second-order logic clauses. An example of the formula is shown as follows:

\[
\delta_2\text{SAT} = \neg a \land b \land (\neg e \lor \neg f) \land (\neg k \lor l)
\]

Mean of $\delta_2\text{SAT} = \frac{-5 \pm 5 \times (\neg 5 - 25 - 25) + (\neg 25 + 25 + 25)}{b} = 0.0833
\]

(38)

The center value located in a dataset is carries a piece of information from every observation in a dataset; accordingly the mean will give the information of the center value for all synaptic weight in logic, where they affect together in cos function on training phases, in this study the mean for 100 combinations been calculated in training phase as sampling size for each logic in both type of $\delta_22SAT\rho$, so we have 100 individual results for the means that have the same characteristic in two parameters $Y$ and $\rho$. It is worth noting that all means’ values were tested first using appropriate tests that yielded significant $p$-values to ensure a correct outcome. The logic value signifies that features will be statistically defined by the curve of probability density function $f(x)$, representing points and (boxplot and whiskers), denoted as (Raincloud Plot), and we want to achieve the following by using these figures:

(a) The probability density function $f(x)$ the curve will give an accurate result data behaviors (symmetric or skewness) so that we can determine if there is an outlier or if all value is
distributed normality in the $\delta_12SAT\rho$ and $\delta_22SAT\rho$ logic (a normal bell curve indicating there is no outlier, and this logic has a high probability of achieving satisfaction in terms of $Y$ and $\rho$).

(b) The representing points the spread of mean values, while the boxplot and whiskers explain the amount of spread around the median, along with the details of an outlier from the median value given by whiskers sides.

This investigation will look at the impact of mean value analysis in evaluating the $DHNN−\delta 2SAT$ during the training phase. We consider the highest $\lambda_1$ in each logic systems combination to calculate the mean, so we have $\lambda_1$ between 48,50 to obtain more accurate results. In the training phase, the synaptic mean value was determined using the ES effect to uncover inconsistent interpretations that offer us a basic understanding about the behavior of logic and achieving satisfied. There are 4 figures for both types of $\delta_12SAT\rho$, each figure includes a probability density function curve, the representing point, and (boxplot and whiskers), its classification depending on the $Y$ values in both types of $\delta_12SAT\rho$, where they have the same structure because it is the key affected parameter in the mean values discussed as follows: For both $\delta_12SAT\rho$ and $\delta_22SAT\rho$:

(a) When $Y = 0.6$ noted the following from Figure 14:

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{Figure14.png}
\caption{The Raincloud Plot analysis for (a) $\delta_12SAT\rho$, (b) $\delta_22SAT\rho$ synaptic weight means when $Y = 0.6$.}
\end{figure}
\(\delta_1 2 \text{SAT} \rho\) probability function curve shows thin-tailed on two sides, so it is fairly to be a symmetric shape, which indicates that outliers are infrequent (an observation is considered an outlier if it differs numerically from the rest of the data), and the values for the mean tend to be normal for A (1,2,3,4), whereas the probability function curve Q3,4 are similar in behavior for \(\delta_1 2 \text{SAT} \rho\). It is fairly to a symmetric ship, so it has thin tailed on two sides and rare outliers, but Q1,2 shows different results because it tends to be non-symmetric by the heavier tail on the left which means that there are a lot of outliers. This result will be supported by the boxplot and whiskers. When we look at interquartile ranges, IQR (the lengths of the boxes), the longer it is, the more dispersed the data are, and the shorter it is, the less dispersed the data are. It can be observed that the \(\delta_1 2 \text{SAT} \rho\) is highly dispersed from the median compared to \(\delta_2 2 \text{SAT} \rho\) since the IQR range is higher in A (1,2,3,4) than Q (1,2,3,4). In addition, in terms of outlier, when checking a box plot, an outlier is defined as a data point that lies outside the box plot’s whiskers, the \(\delta_1 2 \text{SAT} \rho\) and \(\delta_2 2 \text{SAT} \rho\) have the approximate behavior of a huge outlier, but it can be noted that the \(\delta_1 2 \text{SAT} \rho\) has more outlier than \(\delta_2 2 \text{SAT} \rho\) because ES could not achieve inconsistent interpretation in the training phase due to the \(\delta_2 2 \text{SAT} \rho\) models structure that leads to a random value for synaptic weight. Finally, the boxplot clearly shows that the distribution is nonsymmetric for \(\delta_1 2 \text{SAT} \rho\) and \(\delta_2 2 \text{SAT} \rho\), as previously explained (the distribution is symmetric when the median is in the center of the box and the whiskers are nearly the same on both sides of the box), in both logic systems. The reasons for these results are:

In terms of \(Y\) parameter, the number of first-order logic that has a \(p(x_m) = 0.4\) in logic value that pulls the logic curve to the sides because the suboptimal synaptic weight for first-order logic is clearly in the distribution tail and box-whiskers plot also \(\delta_2 2 \text{SAT} \rho\) has more 2SAT than \(\delta_1 2 \text{SAT} \rho\) for the same \(Y\) parameter, and that reflects in the spread of value in the boxplot, which is high in \(\delta_1 2 \text{SAT} \rho\). This indicates a high variation between the mean values ES failed to find a consistent interpretation. In terms of \(\rho\) parameter, from the boxplot also, we can observe that the effect of \(\rho\) gives more negative synaptic weight, but we should also consider the value of \(W_{BB}\) that was positive in cases \((-\xi_1 \lor r_j), (\xi_1 \lor -\xi_1)\) and \((\xi_1 \lor r_j)\) that affected 2SAT clauses mean values, because it is noted in \(\delta_2 2 \text{SAT} \rho\) there is no effect for \(\rho\), as mentioned previously, it has more 2SAT clauses than \(\delta_1 2 \text{SAT} \rho\) for the same \(Y\). Therefore, the ES tend to obtain consistent interpretation that is reflected in the mean values of whole synaptic weight logic. Conversely, for \(\delta_1 2 \text{SAT} \rho\) the effect of \(\rho\) is clearer in the mean values, with most points of the values located on the negative side.

(b) When \(Y = 0.7\) noted the following from Figure 15 as follows:

The probability function curve for \(\delta_1 2 \text{SAT} \rho\) exhibits the same behavior for \(Y = 0.6\), indicating that it is a symmetric ship with normal mean values. It has a thin tailed on two sides, so outliers are infrequent. For \(\delta_2 2 \text{SAT} \rho\) it is a little different, all Q (5,6,7,8) is symmetric. Then, the mean values tend to be normal and have a light tail, except for Q6, as we see in the curve, it is a fat tail, therefore there are a lot of outliers on both sides. The boxplot and whiskers tell the same story for \(Y = 0.6\). When we look at the box side, we can see that the \(\delta_2 2 \text{SAT} \rho\) is highly dispersed from the median compared than the \(\delta_2 2 \text{SAT} \rho\) because the value of IQR is higher in A (4,5,6,7) than the Q (4,5,6,7). Moreover, in terms of an outlier, we can observe that the \(\delta_1 2 \text{SAT} \rho\) and \(\delta_2 2 \text{SAT} \rho\) both have the approximate behavior of a huge outlier, but the \(\delta_1 2 \text{SAT} \rho\) is more outlier than \(\delta_2 2 \text{SAT} \rho\) except for Q6. Most logic has an outlier and, at the same time, is a short box (which implies that high-frequency data tends to be more fat-tailed). Finally, from the boxplot, the non-symmetric shape in both for \(\delta_1 2 \text{SAT} \rho\) and \(\delta_2 2 \text{SAT} \rho\) can be seen clearly. The reasons for these results are justified as follows:

In terms of \(Y\) parameter, the number of second-order logic clauses that have a \(p(x_m) = 0.3\) is considered a bit high, especially in high \(\lambda_1\) which generates ES_{\delta_2 \text{SAT} \rho} \neq 0 that pulls the logic curve to the two sides because the suboptimal synaptic weight is clearly in the tail of probability curve distribution and boxplot-whiskers. For \(\delta_2 2 \text{SAT} \rho\), it has more 2SAT clauses than \(\delta_1 2 \text{SAT} \rho\), for the same \(Y\) parameter. This reflects in the spread of value in the boxplot at its highest more than in \(\delta_1 2 \text{SAT} \rho\). Therefore, it shows a high variation between mean values because ES failed to find consistent interpretations. In terms of \(\rho\) parameter, boxplot in \(\delta_1 2 \text{SAT} \rho\) and \(\delta_2 2 \text{SAT} \rho\) are reflected in a negative synaptic weight value. Both models have the parameter of \(Y = 0.6\), the spread of data affected by \(\rho\) in 2SAT clauses and it affects the value of the mean which tends to be positive, as we mentioned previously. Finally, as seen in the Q6, the reasons for right fat-tailed the number of high second-order logic sentences that generate suboptimal synaptic weight, resulting in positive mean values.
Figure 15. The Raincloud Plot analysis for (a) $\delta_1 2SAT\rho$, (b) $\delta_2 2SAT\rho$ synaptic weight means when $Y = 0.7$.

(c) When $Y = 0.8$ is observed from Figure 16 as follows:

Where the 2SAT clauses are the common clauses, for $\delta_1 2SAT\rho$ the curve shows semi normal ship in A (9,11,12) with a semi skewed in A10, the light tail on the two sides with less outlier is in all $\delta_1 2SAT\rho$. On the other side, $\delta_2 2SAT\rho$ gives near result where Q (10,12) is fairly to be symmetric ship, the mean values tend to be normal, it has a thin-tailed in two sides, Q (9,11) tend to be non-symmetric, the light tail in the two sides with less outlier is on all $\delta_2 2SAT\rho$. The boxplot and whiskers for $\delta_1 2SAT\rho$ and $\delta_2 2SAT\rho$, is highly sparse from the median comparison because the IQR range is higher in A (9,10,11,12) than Q (9,11,12), and shorter in Q10. In the terms outlier, from a box plot whiskers, the $\delta_1 2SAT\rho$ and $\delta_2 2SAT\rho$ have approximate behavior of huge outliers on both sides, but we can note the Q11 is more outlier on the left than others and Q9 is more outlier on the right. Finally, based on the boxplot, it clarifies both logic systems have non symmetric curves. The reasons for these results are justified as follows:

In terms of $Y$ parameter, the number of first-order logic clauses that have a small appearance probability that makes the range values of mean is high in the two previous Figures 14 and 15. It is clear here in these figures the $\delta_1 2SAT\rho$, $\delta_2 2SAT\rho$ obtaining (0.5) synaptic weight is small, so most of the means value range is small that led to less spread curve line, on the other side, the high representation of 2SAT clauses makes the length of the box highest because the volatile in the mean
values of 2SAT clauses it gives a different result depending on negative literal, where \((¬r_i ∨ r_j)\), \((r_i ∨ ¬r_j)\) and \((r_i ∨ r_j)\) have the mean values different from \((¬r_i ∨ ¬r_j)\) the effect also by ES algorithm searching and that effect in cost function in Equation (12), that pull the logic curve and boxplot-whiskers into sides, that reflects in the spread of values in boxplot its highest than in \(Y = 0.6, 0.7\). In terms of \(ρ\) parameter, its high effect here, in boxplot in \(δ_12SATρ\) and \(δ_22SATρ\) is clearly in the range of values, most of it full in the negative side, more clearly in Q, A(11,12) because the mean values of full negative second-order logic clauses it is highest here as we clarify in FNAE matric. It is also noted for Q (9,10), A10 is in the positive side because the \(ρ\) is small therefore, the mean will be positive and ES algorithm searching tend to find inconsistent interpretation. This indicates the effect of the parameter \(ρ\) but A9 still has first-order logic, which makes the data spread in two sides with light tail. However, in Q10 and Q12, the tail because the extreme mean values that come from full negative clauses and first-order logic clauses.

**Figure 16.** The Raincloud Plot analysis for (a) \(δ_12SATρ\), (b) \(δ_22SATρ\) synaptic weight means when.

(d) When \(Y = 0.9\) is observed from Figure 17 as follows:
The $\delta_1 2SAT\rho$ probability function curve indicates that it is reasonable to be a symmetric shape in $A(15,16)$, but $A14$ tends to be non-symmetric, with a thin tail in two sides, implying that outliers are infrequent. Whereas $A13$ is left-right skewed and heavy-tailed, which implies that there a lot of outliers on the left, but in $\delta_2 2SAT\rho$, $Q(13,14,16)$ is symmetrical. While $Q15$ tends to be non-symmetric, they have a thin tailed on two sides, implying that outliers are infrequent. Moreover, $Q14$ is heavily tailed which indicates there is a lot of outliers, but $Q13, 16$ have light tails and outliers are infrequent. When we look at interquartile ranges, we can observe that $\delta_1 2SAT\rho$, $A(15,16)$ is considerably distributed from the median compared to $A(13,14)$ because the IQR range is similarly high in $\delta_2 2SAT\rho$. Meanwhile, $Q(13,15,16)$ is highly dispersed from the median compared to $Q(14)$ because the IQR range is highest in terms of an outlier. When reviewing box whiskers, the $\delta_1 2SAT\rho$ and $\delta_2 2SAT\rho$ have the approximate behavior of a huge outlier however, we can note that $Q, A(13,14)$ is more outlier than $Q, A(15,16)$. Finally, from the boxplot, it is clearly the non-symmetric for $\delta_1 2SAT\rho$ and $\delta_2 2SAT\rho$ as we previously mentioned. The reasons for these results are justified as follows:

In terms of $Y$ parameter, the number of second-order logic clauses that have the smallest appearance, so the mean values are high, is clear in the $\delta_1 2SAT\rho, \delta_2 2SAT\rho$ figures. Moreover, the majority of $2SAT$ clauses representing, make the spread in all length box highest in $\delta_1 2SAT\rho, \delta_2 2SAT\rho$
because of the volatility in the mean of 2SAT clauses, as mentioned previously. This effect in the
logic curve and pulling the logic curve into the two sides also for boxplot-whiskers, which reflect
in the dispersion of value in boxplot is more than \( \gamma = 0.6, 0.7 \). In terms of the \( p \) parameter, it has a
high effect as well, in the boxplot in \( \delta_22SATp \) and \( \delta_22SATp \), it is clearly in the range of value, most
of it falls in the negative side also more clearly in \( Q, A \) (15, 16) because the mean of full negative
second-order logic clauses is highest here. As we explain in the FNAE metric, for other logic A, Q (14,
13) still has more first-order logic, which causes the mean spread in two directions and a heavy tail in
Q14 and A13 due to the extrema value that occurs due to the full negative clauses and second-order
logic clauses.

From this result, we can note the significance of the synaptic weight analysis; it gives a summary
of the search space area for a specific algorithm in training phases, and it is clarified by the mean
synaptic weight results, which give the center of search space (optimal) and the wide by the range of
spread (suboptimal) from the previous result the mean synaptic weight gives a general perspective
for the mechanism of ES algorithm in this search space. Thus, we can observe the behavior of working
in this limited space, as well as the behaviors of obtaining a solution using optimal and suboptimal
synaptic weights. The ES has a unique search space that is heavily influenced by the number of
neurons and the structure of logic.

6.6. The Limitation of the DHNN-\( \delta2SAT \)

One of the limitations of \( DHNN - \delta2SAT \) in this study is that the proposed hybrid network
DHNN only considers propositional logic programming. The DHNN is unable to embed other variant
of logic, such as predicate logic, fuzzy logic, or probabilistic logic due to the nature of Hopfield Neural
Network proposed by Pinkas [56] that are limited to symmetric connectionist network, as well as the
DHNN’s low storage capacity and the cost function proposed by Wan Abdullah (1992), which only
considers bipolar neurons. Conversely, this study takes a number of neurons limits is less than 52
because of ES. Consequently, as we improve, will replace ES by metaheuristics such as Artificial Bee
Colony Algorithm [57] and Election Algorithm [58]. Despite DHNN flexibility, \( \delta2SAT \)'s the quality
of solutions offered needed to be improved. We can increase the iterations numbers required in our
simulations by increasing the number of learning. The proposed model may yield more variation
neurons, less errors, and a global minimum solution with more iterations.

6.7. Summary

In this section, we provide a brief summary of the beneficial properties of the logical structure
of the proposed model; moreover, we provide a simple summary of the most important accompli-
ishments of the proposed logic system, clarifying the findings given in the Results and Discussion
section with respect to the following points:

(a) Probability logic phases were applied to introduce various models to address dataset-related
requirements. Notably, one of the most significant advantages of \( \delta kSAT \) is that it can generate
multifarious models by controlling parameters that are revealed from the dataset features in
the logic system. It is a flexible logic system, but this is not discussed in this study. The
parameters can be used to generate models of logic that can be systematic when \( p(x_m) = 0 \),
transforming to 2SAT and when \( p(y_m) = 0 \), it becomes first-order logic or it can be high-
order non-systematic when \( k = 3 \), and it can be SRAN3SAT for order \( k = 1, 2, 3 \) or \( k = 2, 3 \)
or \( k = 1, 3 \) by adding a new parameter \( p(z_m) \). In this case, regarding the probability of third
classes, we consider the probability concept \( p(z_m) + p(y_m) + p(x_m) = 1 \), when \( p(y_m) = 0 \) and
\( p(x_m) = 0 \), it becomes 3SAT. The main differences between \( \delta kSAT \) and other logic systems such
as YRAN2SAT, RAN3SAT, and RAN2SAT, as well as other systematic logic systems such as
2SAT and 3SAT, are the factors of probability, wherein the dataset will choose the best structure
by controlling the probability parameter; in addition, the terms of negative literals determined
from the dataset and distributed in clauses depend on the probability parameter, whose two
main features render the \( \delta kSAT \) unique.

(b) The testing and the training phases were examined. By applying Equations (24) and (25) in the
testing phase, the results show that the proposed model obtained optimal synaptic weight after
checking the clauses’ satisfaction. It also generated optimal synaptic weight through the WA
method for the small number of neurons and high parameter values. Equations (26) and (27) in
the training phase showed that the efficiency of the probability logic phase produced various
logical structures in the DHNN compared to the current systems.

(c) A novel analysis of the synaptic weight for \( DHNN - \delta2SAT \) was introduced, which was
termed the descriptive statistic of mean synaptic weight. Previously, there have been various
statistical tests used to study the behaviors of synaptic weight to deduce information about the performance of a proposed logic system in the training phases. Whereas, in this study, the descriptive statistical method analyzed the synaptic weight distribution by obtaining the mean of the synaptic weight in the testing phase.

(d) Notably, in the Results and Discussion section, the sample size in Equation (5) gives us the best number of negative literals for the desired logic needed to obtain satisfaction. Of particular significance are the models $\delta_1 2SAT$ and $\delta_2 2SAT$, which have a high proportion ($0.9$) and high probability ($0.9$) introduced by probability logic phases, have the best structure as clarified by the measures used in the study (PON, NAE, and FNAE), and tended to be the best models in the training and testing phases, which are also shown by the similarity index measures. This result is the opposite of that obtained by Zamri et al. [24], which concluded a value of $r = 0.5$ for negative literal works efficiently in the logic phase and yielded a better structure than ($r = 0.1, 0.9$). The reason behind these contrary findings is that the proportion is dependent on the $d$ value in Equation (6), which gives a margin of error dependent on the $Z$ value; additionally, there is the probability of second-order logic $Y$ drawn from the dataset, which affected the $\delta_2 2SAT$ models—all these factors rendered it the best in terms of logic structure.

(e) In this study, the probability distribution from the contributed data set successfully generated an efficient, new logical structure for a DHNN. The discussion section considered the introduction of the comparative analysis of the $\delta 2SAT$ with other existing SATs, for which the proposed model was superior in several aspects, as shown in Table 18.

<table>
<thead>
<tr>
<th>Contribution</th>
<th>$\delta 2SAT$</th>
<th>rSAT</th>
<th>MAJ2SAT</th>
<th>2SAT</th>
<th>RAN3SAT</th>
<th>RAN2SAT</th>
<th>YRAN2SAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organized phase</td>
<td>√</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>System for selecting clauses</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>System for selecting negative literals</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Systematic structure</td>
<td>√</td>
<td>√</td>
<td></td>
<td></td>
<td>√</td>
<td></td>
<td>√</td>
</tr>
<tr>
<td>Non-Systematic structure</td>
<td>√</td>
<td></td>
<td></td>
<td>√</td>
<td></td>
<td>√</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### 7. Conclusions and Future Work

It is critical to create a non-systematic logical framework in a DHNN, employing parameters conducive to building a flexible final neuronal state. This study introduced a new probability logic phase that assigns the probability of the first- and second-order clauses and the desired negative literals appearing in each sentence, which helped to address the requirements of datasets. Statistical tools govern the creation of $\Theta 2SAT$ during the probability logic phase. The novel logic probability phase of the proposed $\delta 2SAT$ model provides a new enhancement with which to shape the logic structure according to the dataset, for which it was found that these models have high values in two parameters ($Y = 0.9, \rho = 0.9$) of two $\delta 2SAT$ types, which introduced efficient logic structures in the probability logic phase. The new logic was embedded in the DHNN $-\delta 2SAT$ by reducing the logical inconsistency of the corresponding zero-cost function’s logical rule. The achieved cost function that corresponds to satisfaction was used to calculate the synaptic weight of the DHNN’s effectiveness with a $\delta 2SAT$ logical structure, which was examined using three proposal metrics in comparison with state-of-the-art methods, such as 2SAT, MAJ2SAT, RAN2SAT, RAN3SAT, YRAN2SAT, and rSAT. The final neuron state was assessed based on various initial neuron states, statical method parameters, and various metric performances, such as learning errors, synaptic weight errors, energy profiles, testing errors, and similarity metrics, which were compared with existing benchmark works. To further demonstrate the efficiency and robustness of the proposed $\Theta 2SAT$, it was validated using four different second-order probability distributions with four different proportions of extensive simulations. Further, a new prospective logical investigation was introduced in this study, which consisted of the analysis of the mean of synaptic weight for DHNN $-\delta 2SAT$ to evaluate the existence of a flexible logical structure. The findings demonstrated that the proposed $\delta 2SAT$ was successful in achieving a flexible logical structure with a prevailing attribute dataset compared to other state-of-the-art SAT. For future work: (1) A metaheuristic analysis of the probability logic phase would aid the selection of the negative literals’ positions in a logic system. (2) A metaheuristic analysis of the training phases would aid the satisfaction of Equation (12). (3) A metaheuristic analysis of the testing phases would aid the generation of a vast range of space solutions. (4) Synaptic weight analysis can
be applied in the training phases to address the effects of the energy function and global solutions on
the synaptic weight. Moreover, we can add the measure of variability to address the deviation in the
results. Notably, the robust architecture of ANNs integrated with our proposed logic would serve
as a good foundation for real-life applications such as Natural Disaster prediction. In this context,
each neuron would represent the attributes from the data, such as rainfall trends, river levels, and
drainage and ground conditions. These attributes will be embedded into the logic-mining approach
proposed by [45], which will lead to the formation of induced logic, which, in turn, has predictive
and classificatory abilities. In other developments, the proposed logic system would be indispensable
in finding the optimal route in the Travelling Salesman Problem.

**Author Contributions:** Conceptualization, methodology, software, writing—original draft prepa-
ratin, S.A.; formal analysis, validation, N.E.Z.; supervision and funding acquisition, M.S.M.K.;
writing—review and editing, G.M.; visualization, N.A.; project administration, M.A.M. All authors
have read and agreed to the published version of the manuscript.

**Funding:** This research was supported by Ministry of Higher Education Malaysia for Transdisci-
plinary Research Grant Scheme (TRGS) with Project Code: TRGS/1/2022/USM/02/3/3.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** The authors would like to express special thanks to all researchers in the Artificial
Intelligence Research Development Group (AIRDG) for their continued support.

**Conflicts of Interest:** The authors declare no conflict of interest.

**Abbreviations**

<table>
<thead>
<tr>
<th>Notation</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AI</td>
<td>Artificial Intelligence</td>
</tr>
<tr>
<td>DHNN</td>
<td>Discrete Hopfield Neural Network</td>
</tr>
<tr>
<td>ANN</td>
<td>Artificial Neural Network</td>
</tr>
<tr>
<td>CAM</td>
<td>Content addressable memory</td>
</tr>
<tr>
<td>SAT</td>
<td>Satisfiability</td>
</tr>
<tr>
<td>HORNSAT</td>
<td>Horn Satisfiability</td>
</tr>
<tr>
<td>2SAT</td>
<td>2 Satisfiability</td>
</tr>
<tr>
<td>3SAT</td>
<td>3 Satisfiability</td>
</tr>
<tr>
<td>RAN2SAT</td>
<td>Random 2 Satisfiability</td>
</tr>
<tr>
<td>RAN3SAT</td>
<td>Random 3 Satisfiability</td>
</tr>
<tr>
<td>MAJ2SAT</td>
<td>Major 2 Satisfiability</td>
</tr>
<tr>
<td>YRAN2SAT</td>
<td>Y-Type Random 2 Satisfiability</td>
</tr>
<tr>
<td>GRAN3SAT</td>
<td>G-Type Random k Satisfiability</td>
</tr>
<tr>
<td>PSAT</td>
<td>Probabilistic Satisfiability Problem</td>
</tr>
<tr>
<td>rSAT</td>
<td>Weighted Random k Satisfiability</td>
</tr>
<tr>
<td>PMAXSAT</td>
<td>partial maximum satisfiability</td>
</tr>
<tr>
<td>GA</td>
<td>Genetic Algorithm</td>
</tr>
<tr>
<td>ES</td>
<td>Exhaustive Search</td>
</tr>
<tr>
<td>HTAF</td>
<td>Hyperbolic Tangent Activation Function</td>
</tr>
<tr>
<td>WA</td>
<td>Wan Abdullah method</td>
</tr>
<tr>
<td>CNF</td>
<td>Conjunctive Normal Form</td>
</tr>
<tr>
<td>RMSE</td>
<td>Root-Mean-Square Error</td>
</tr>
<tr>
<td>PON</td>
<td>Probability of total negative</td>
</tr>
<tr>
<td>NAE</td>
<td>Negativity Absolut Error</td>
</tr>
<tr>
<td>FNAE</td>
<td>Full negativity Absolut Error second clauses</td>
</tr>
<tr>
<td>WFNAE</td>
<td>Weight Full negativity Absolut Error</td>
</tr>
<tr>
<td>( \rho_0 )</td>
<td>pre-defined proportion range</td>
</tr>
<tr>
<td>( \rho )</td>
<td>negative literal proportion</td>
</tr>
<tr>
<td>( \alpha )</td>
<td>Significance Level</td>
</tr>
<tr>
<td>( Z )</td>
<td>the upper ( \alpha/2 ) point of the normal distribution</td>
</tr>
<tr>
<td>( w )</td>
<td>Number of learning in probability logic phases</td>
</tr>
</tbody>
</table>
\( \tau \) literal
\( T^{(1)} \) First-order clause
\( T^{(2)} \) Second order clause
\( Y \) Probability second-order logic range
\( \lambda_1 \) Number of literals/neurons
\( \lambda_2 \) Total clauses
\( x \) Number of the second-order clauses
\( y \) Number of the first-order logic clauses
\( \Theta_{2S\text{SAT}} \) General formula of \( 2\text{SAT} \)
\( W_{ij} \) Synaptic weight between \( i \) and \( j \)
\( W_{ii} \) Synaptic weight of neuron \( i \)
\( F_{\text{desired}} \) Maximum fitness
\( F_i \) Current fitness
\( W_E \) Expected Synaptic weight that obtained by Wan method
\( W_A \) Actual synaptic weight
\( \xi \) The Total of negative literal in logic
\( p(y_m) \) The probability of obtaining second-order clauses
\( \xi_{2\text{SAT}} \) The number full negativity second-order clauses
\( \lambda_{2\text{SAT}} \) The number of second-order
\( \lambda_{2\text{SAT}} \) The mean number of second-order clauses
\( H_{\Theta_{2\text{SAT}}} \) Minimum energy value
\( H_{\Theta_{2\text{SAT}}} \) Final energy
\( R_G \) Ratio of global minimum solutions
\( G_{\Theta_{2\text{SAT}}} \) Number of global minimum solutions
\( S_i \) Neuron state
\( S_{\text{max}} \) Benchmark neuron state
\( \text{Sokal} \) Sokal and Michener Index
\( R_{tv} \) The Ratio of cumulative neuronal variation
\( h_i \) local field
\( b, b^* \) Counter
\( \nu \) Number of Learning
\( \eta \) Number of neuron combination
\( \varphi \) Number of Trials
\( Tol \) Tolerance value
\( R \) Relaxation rate
\( \phi \) Learning iteration
\( \theta \) Threshold constraint of DHNN
\( E_{\Theta_{2\text{SAT}}} \) The cost function of the DHNN-YRAN2SAT
\( \bar{X} \) The arithmetic mean
\( \tilde{X} \) The median
\( \sigma x \) Standard deviation
\( Q_1 \) First quartile
\( Q_2 \) Second quartile
\( Q_3 \) Third quartile
\( IQR \) Interquartile range
\( LF \) Lower fences
\( UF \) Upper fences
\( f(x) \) Probability density function for random variables
\( W_s^* \) Smallest of absolute values of the sum of \( x_i \) in Wilcoxon test
\( W \) Wilcoxon test value (sum of smallest and biggest of absolute values of the sum \( x_i \))

**References**


5. Juan, N.P.; Valdecantos, V.N. Review of the application of Artificial Neural Networks in ocean engineering. *Ocean Eng.* **2022**, *259*, 111947. [CrossRef]


Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.