



SCIENTIFIC OASIS

Journal of Soft Computing and Decision Analytics

Journal homepage: www.jsdda-journal.org
ISSN: 3009-3481

Classification-Oriented Neural Networks with Granular Processing

Inas Shadoul^{1,*}, Rami Al-Hmouz², Majdi Mansouri^{1,*}, Mostefa Mesbah¹¹ Department of Electrical and Computer Engineering, Sultan Qaboos University, Alkhoud, Muscat, Oman² College of Computer Systems Engineering, Abdullah Al Salem University, Khaldiya, Kuwait

ARTICLE INFO

Article history:

Received 26 July 2026

Received in revised form 17 September 2026

Accepted 3 October 2026

Available online 26 October 2026

Keywords:

Granular Computing; Neural Network; Allocation of Information Granules; Interval Analysis

ABSTRACT

This paper presents a classification framework of granular neural networks (GNNs) designed for classification tasks. Built upon conventional neural network architectures, the framework introduces interval-valued (granular) connections, where each interval forms an information granule around a numeric weight. This design is proposed to utilize a granular approach in classification tasks. This representation provides a quantifiable measure of uncertainty and interpretability in classification decisions. By optimally distributing information granularity across network connections, the proposed model seeks to maximize classification confidence while preserving specificity. To achieve this objective, a single-objective particle swarm optimization (PSO) algorithm is employed, guided by classification coverage and specificity criteria. The proposed approach is validated using benchmark datasets and evaluated through multiple performance indices derived from the Interval Neutrosophic Set (INS) and granular computing domains.

1. Introduction

Traditional neural networks used for classification are typically designed as deterministic systems, which take inputs and produce definite categorical labels. While this approach works well with clean data, it struggles in situations filled with uncertainty, such as medical diagnostics, environmental monitoring, or remote sensing, where input features are often noisy, ambiguous, or incomplete. In such cases, it's not just about what the model predicts, but also how confident it is in those predictions. This is where Granular Neural Networks (GNNs) present a meaningful shift. By incorporating information granules and interval-valued representations into their structure, GNNs enable the inherent quantification of uncertainty. This enhances the interpretability and robustness of their predictions in the face of data variability. The existing research area focused on designing, optimizing, and applying GNNs for tasks, regression [1], prediction intervals [2], and time series analysis [3]. We will focus on a classification task. This review consolidates recent studies that advance both the theoretical understanding and practical implementation of GNNs.

GNNs extend conventional neural architectures by embedding information granules, intervals formed around numeric weights, into the model. These models produce interval-valued outputs that encapsulate the uncertainty associated with predictions. Since the pioneering framework introduced

*Corresponding author..

E-mail address: s121278@student.squ.edu.om, m.mansouri@squ.edu.om<https://doi.org/10.31181/jsdda412026105>© The Author(s) 2026 | [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/)

by [1], research in this domain has grown into a rich interdisciplinary field. [4] contributed a neural-fuzzy modelling scheme grounded in granular computing, while [5] took early steps toward integrating fuzzy regression with interval weights. More recent efforts have honed in on the optimization of granularity itself both in the input space and within model parameters [6–8] as well as adaptive granulation based on data characteristics [9, 10]. [11] introduced hyper-box iteration granulation to streamline structure and improve computational clarity. In parallel, applications have expanded: [12] unified granular architectures for improved classification, while others have pushed GNNs into forecasting [13, 14], compact modeling via optimized data subsets [15], and dynamic network analysis [16]. Even in limited-data regimes like few-shot learning, granularity has proven valuable for generalization [17, 18]. Meanwhile, foundational reviews [19] and conceptual expansions [20] continue to re-define what granular computing can bring to neural systems.

Although conventional neural networks achieve high classification accuracy, they typically provide point-valued predictions that fail to explicitly characterize predictive uncertainty. To overcome this limitation, this paper presents a classification-oriented Granular Neural Network (GNN) framework that produces interval-valued probabilistic outputs, enabling uncertainty-aware classification through granular information representation

- A new formulation of interval-valued class predictions that allows for simultaneous uncertainty estimation in classification tasks.
- The principle of justifiable granularity and the allocation of information granularity are combined, wherein interval bounds are optimized utilizing PSO to balance coverage and specificity.
- A new mapping from interval-valued GNN outputs to Interval Neutrosophic Sets (INS), enabling semantic interpretation through Truth, Intermediacy, and Falsity.
- A unified evaluation framework combining geometric (coverage/specificity) and semantic (INS) metrics for assessing uncertainty-aware classifiers.

It is important to emphasize that the proposed work departs from conventional objectives in machine learning that primarily focus on maximizing predictive accuracy. Instead, it aims to explicitly model and quantify uncertainty, particularly in settings where the underlying classifier is weak or sub-optimally trained.

The paper is structured as follows: Section 2 introduces the Granular Classifier and its neural network basis, along with the performance metrics used for evaluation. Section 3 shows experimental results on various public datasets, and Section 4 concludes the paper.

2. Granular Neural Network Classifier Based on Interval-Valued Architectures

We begin with a standard multilayer perceptron (MLP) classifier trained using categorical cross-entropy loss. The input vector $\mathbf{x} \in \mathbb{R}^n$ is propagated through layers using interval-valued weights and biases.

Interval Arithmetic Essentials Before delving deeper into the granular neural network structure, it is helpful to recall basic results from interval arithmetic. When variables are represented as intervals say, $X = [a, b]$ and $Y = [c, d]$ —the fundamental operations are defined as follows:

$$\begin{aligned} \text{Addition: } X + Y &= [a + c, b + d], \\ \text{Subtraction: } X - Y &= [a - d, b - c], \\ \text{Multiplication: } X \times Y &= [\min(ac, ad, bc, bd), \max(ac, ad, bc, bd)], \\ \text{Division: } \frac{X}{Y} &= X \times \left[\frac{1}{d}, \frac{1}{c} \right] \quad (\text{assuming } 0 \notin Y). \end{aligned}$$

Additionally, the behavior of functions applied to intervals depends on their monotonicity:

$$\begin{aligned} \text{If } f \text{ is non-decreasing: } f([a, b]) &= [f(a), f(b)], \\ \text{If } f \text{ is non-increasing: } f([a, b]) &= [f(b), f(a)]. \end{aligned}$$

These rules serve as the mathematical foundation for processing interval-valued data within the

GNN framework.

Given the input vector $x = [x_1, x_2, \dots, x_n]$, interval weights $W_{ji} = [w_{ji}^-, w_{ji}^+]$, interval bias $b_j = [b_j^-, b_j^+]$, and assuming c output classes, the hidden layer computation is defined as

$$z_j^- = \sum_{i=1}^n \min(w_{ji}^- x_i, w_{ji}^+ x_i) + b_j^-, \quad (1)$$

$$z_j^+ = \sum_{i=1}^n \max(w_{ji}^- x_i, w_{ji}^+ x_i) + b_j^+, \quad (2)$$

where z_j^- and z_j^+ denote the lower and upper bounds of the pre-activation value for the j th hidden neuron, respectively. Applying a monotonic activation function (e.g., sigmoid or ReLU) gives

$$h_j = [h_j^-, h_j^+] = [f(z_j^-), f(z_j^+)]. \quad (3)$$

Given interval weights $v_{kj} = [v_{kj}^-, v_{kj}^+]$ and interval bias $d_k = [d_k^-, d_k^+]$, the interval output of the k th output neuron is computed as

$$\tilde{y}_k^- = \sum_{j=1}^H v_{kj}^- h_j^- + d_k^-, \quad (4)$$

$$\tilde{y}_k^+ = \sum_{j=1}^H v_{kj}^+ h_j^+ + d_k^+, \quad (5)$$

where H is the number of hidden neurons. In actuality, for the posterity of the NN, the actual outputs $y_k = [y_k^-, y_k^+]$ are the result of:

$$y_k^- = \min(\tilde{y}_k^-, \tilde{y}_k^+), \quad y_k^+ = \max(\tilde{y}_k^-, \tilde{y}_k^+), \quad (6)$$

$$\text{which guarantees: } y_k^- \leq y_k^+ \quad (7)$$

The network therefore produces an interval prediction for each output class,

$$\mathbf{y} = \{[y_1^-, y_1^+], [y_2^-, y_2^+], \dots, [y_c^-, y_c^+]\}. \quad (8)$$

It is through this output calculated, that the classification is made. But due to the nature of the result, it is not possible to strictly call this result a probability, as it does not use the softmax function. The results produced through our experiment and described system will be referred to as a score as opposed to a probability. Figure 1 shows a granular neural network with three inputs, two hidden layers, and three output layers, featuring interval-based weights and biases, with the output also expressed as intervals. qi

2.1 Granular Allocation and Optimization

The process of forming interval weights involves allocating an information granularity level ϵ to each connection [1]. For a given weight w_{ji} and bias b_k , the corresponding interval bounds are defined as:

$$w_{ji}^- = w_{ji} - \epsilon |w_{ji}|, \quad (9)$$

$$w_{ji}^+ = w_{ji} + \epsilon |w_{ji}|, \quad (10)$$

$$b_k^- = b_k - \epsilon |b_k|, \quad (11)$$

$$b_k^+ = b_k + \epsilon |b_k|. \quad (12)$$

where $\sum_{i=1}^p \epsilon_i = p\epsilon$ and p represent the number of parameters in the neural network (total number of weights and biases)

The total available granularity is distributed among weights and biases. Different allocation protocols are considered:

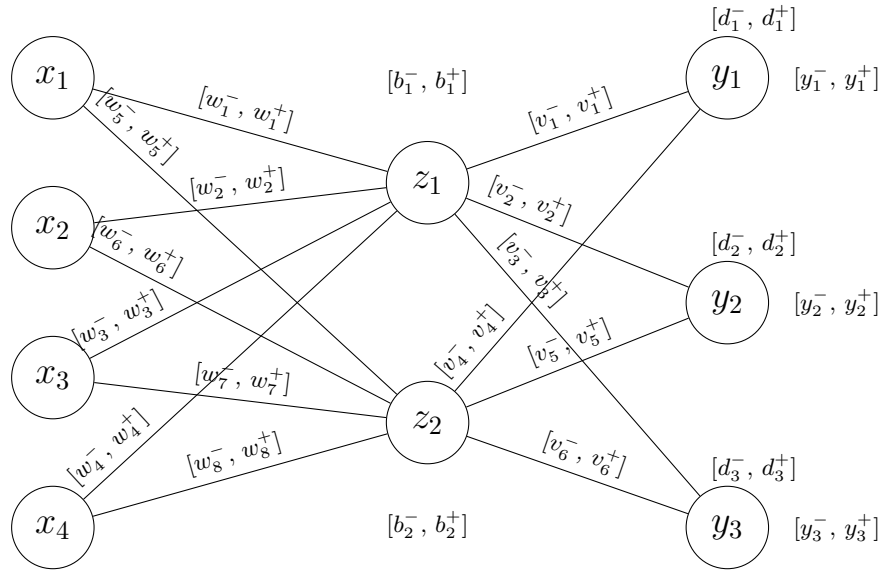


Fig. 1. Granular Neural Network

- **C1:** Uniform symmetric allocation where all connections use the same ϵ , split evenly as $\epsilon^- = \epsilon^+ = \epsilon/2$.
- **C2:** Uniform asymmetric allocation with same ϵ but different ϵ^- and ϵ^+ satisfying $(1 - \gamma)\epsilon^- + (\gamma)\epsilon^+ = \epsilon$. $\gamma \in [0, 1]$
- **C3:** Nonuniform symmetric allocation where each connection has its own ϵ_i , symmetrically split as $\epsilon_i^- = \epsilon_i^+ = \epsilon_i/2$.
- **C4:** Nonuniform asymmetric allocation where $\epsilon_i^- + \epsilon_i^+ = \epsilon_i$.
- **C5:** Random allocation used as a baseline to evaluate optimization improvements.

2.2 Interval Neutrosophic Sets (INS)

Interval Neutrosophic Sets (INSs) are an extension of fuzzy set concepts, aimed at addressing problems involving intervals of real numbers and the limitation of fuzzy sets in handling certain types of uncertainty [21]. As a generalization of multiple forms of fuzzy sets, INSs describe an item in the set with three values: Truth (T), Intermediacy (I), and Falsity. (F). An item, let it be A, can relate to a set with a value T_A showing how strongly it belongs to the set, a value I_A showing how indeterminate it is in the set and a value F_A to relay how falsely related it is to the set. As the values T, I, and F ultimately describe membership, they must obey the constraint:

$$0 \leq T_A + I_A + F_A \leq 3 \quad (13)$$

Allowing independent variation of the variables. The use of Neutrosophic set characterization can provide greater flexibility [22]. In the case of vague values, the decision of the class requires a decision based on a metric, which is where INS interpretations are usable. With the use of concepts of Truth, Falsity, and Intermediacy, the results can be better represented, and qualitative descriptions can be used [23]. Using an understanding of INS characteristics, the results of the granular Neural Network can be mapped to the INS constraints. The GNN produces two vectors of interval class prediction, namely the lower-bound prediction vector $\mathbf{y}^-$ and the upper-bound prediction vector $\mathbf{y}^+$, each containing one predicted score for every output class ($k = 1, \dots, c$):

$$\mathbf{y}^+ = [y_1^+, y_2^+, \dots, y_c^+],$$

$$\mathbf{y}^- = [y_1^-, y_2^-, \dots, y_c^-].$$

Each output class is therefore represented by an interval

$$[y_k^-, y_k^+], \quad k = 1, \dots, c.$$

Traditional crisp accuracy evaluates only the final predicted class and therefore cannot fully characterize the uncertainty represented by these interval predictions. By mapping the interval outputs into the Interval Neutrosophic Set (INS) framework, three complementary characteristics are obtained:

(a) **Truth (T):** Let

$$Y_t = [y_t^-, y_t^+]$$

be the interval prediction output of the target class, and let

$$Y_o = [y_o^-, y_o^+],$$

be the interval score of the strongest competing class, where

$$y_o^- = \max_{k \neq t} y_k^-, \quad y_o^+ = \max_{k \neq t} y_k^+.$$

The Truth component is defined directly as the target interval,

$$T = Y_t. \quad (14)$$

(b) **Intermediacy (I):**

Intermediacy represents the uncertainty shared between the target class and its strongest competing class. It is defined as the intersection of their interval output,

$$I = Y_t \cap Y_o = \begin{cases} [\max(y_t^-, y_o^-), \min(y_t^+, y_o^+)], & \text{if } \max(y_t^-, y_o^-) < \min(y_t^+, y_o^+), \\ \emptyset, & \text{otherwise.} \end{cases} \quad (15)$$

(c) **Falsity (F):**

Falsity quantifies the strongest evidence supporting the competing class against the target class. Since the shared uncertainty has already been captured by the Intermediacy component, Falsity is defined as the largest non-overlapping segment of the competing interval.

The left and right non-overlapping segments are

$$L = \begin{cases} [y_o^-, y_t^-], & y_o^- < y_t^-, \\ \emptyset, & \text{otherwise,} \end{cases}$$

$$R = \begin{cases} [y_t^+, y_o^+], & y_o^+ > y_t^+, \\ \emptyset, & \text{otherwise.} \end{cases}$$

where the interval length is defined as

$$\text{len}([a, b]) = b - a.$$

The falsity interval is then determined as

$$F = \begin{cases} \emptyset, & Y_o \subseteq Y_t, \\ Y_o, & Y_t \cap Y_o = \emptyset, \\ \arg \max_{S \in \{L, R\}} \text{len}(S), & \text{otherwise.} \end{cases} \quad (16)$$

Here, Truth represents the confidence assigned to the target class, Intermediacy captures

the uncertainty shared between the target and the strongest competing class, and Falsity represents the dominant non-overlapping evidence supporting the competing class. This mapping preserves the interval-valued outputs produced by the GNN while providing a consistent Interval Neutrosophic interpretation.

Unlike classical INS, where Truth, Indeterminacy, and Falsity are independently assigned, the proposed approach derives these three components directly from interval-valued class predictions generated by the GNN. Consequently, the proposed formulation constitutes a new INS mapping tailored for interval-based classification

Consider a GNN with four output classes ($c = 4$). For each class $k = 1, \dots, c$, the network produces an interval prediction $[y_k^-, y_k^+]$. The prediction corresponding to the target class t is mapped into the Interval Neutrosophic Set (INS) framework using the three components:

- Truth (T): confidence in the target class.
- Falsity (F): evidence supporting the strongest competing class.
- Intermediacy (I): uncertainty caused by overlap between the target and competing class intervals.

This representation distinguishes confident, ambiguous, and incorrect predictions, providing a richer uncertainty characterization than conventional crisp accuracy.

Example A (Target clearly stronger).

Suppose the GNN produces the interval-valued prediction vectors

$$\mathbf{y}^- = [0.10, \mathbf{0.36}, 0.20, 0.05], \quad \mathbf{y}^+ = [0.40, \mathbf{0.80}, 0.55, 0.30],$$

where the target class is $t = 2$. Therefore,

$$Y_t = [0.36, 0.80], \quad Y_o = [0.20, 0.55].$$

Since the two intervals overlap,

$$T = Y_t = [0.36, 0.80], \quad I = [0.20, 0.55], \quad F = \emptyset.$$

Using the interval midpoints as representative values,

$$T^* = 0.58, \quad I^* = 0.175, \quad F^* = 0,$$

which satisfy

$$0 \leq T^* + I^* + F^* = 0.755 \leq 3.$$

Example B (Strong competing evidence).

Suppose

$$Y_t = [0.20, 0.55], \quad Y_o = [0.36, 0.80].$$

The overlap between the two intervals is

$$I = Y_t \cap Y_o = [0.36, 0.55].$$

The left non-overlapping segment is

$$L = \emptyset,$$

while the right non-overlapping segment is

$$R = [0.55, 0.80].$$

Since

$$\text{len}(R) > \text{len}(L),$$

the falsity interval is

$$F = R = [0.55, 0.80].$$

Thus,

$$T = [0.20, 0.55], \quad I = [0.36, 0.55], \quad F = [0.55, 0.80].$$

Using interval midpoints,

$$T^* = 0.375, \quad I^* = 0.455, \quad F^* = 0.675,$$

which satisfy

$$0 \leq T^* + I^* + F^* = 1.505 \leq 3.$$

These examples illustrate how interval-valued GNN outputs are mapped into the proposed INS representation. In Example A, the target interval is completely separated from the strongest competing interval, resulting in high Truth and zero Intermediacy and Falsity, which indicates a confident prediction. In Example B, the target and competing intervals partially overlap, producing a non-empty Intermediacy interval together with a dominant Falsity interval, indicating both ambiguity and strong competing evidence.

Classification based on interval-valued outputs: The trained GNN produces two vectors creating and an interval-values score, $\mathbf{y}^-$ and $\mathbf{y}^+$, representing the lower and upper bounds of the class score, respectively. The final class prediction is determined according to two possible cases.

Consistent Prediction: If both score vectors assign their maximum score to the same class, that class is selected as the final prediction. For example,

$$\mathbf{y}^- = [0.10, \mathbf{0.36}, 0.20, 0.05], \quad \mathbf{y}^+ = [0.40, \mathbf{0.80}, 0.55, 0.30].$$

Both vectors assign the highest score to class $k = 2$; therefore, the final predicted class is class 2.

Conflicting Prediction: If the lower and upper bound score vectors assign their maximum score to different classes, the final prediction is determined by the upper bound score vector $\mathbf{y}^+$. For example,

$$\mathbf{y}^- = [0.10, 0.36, \mathbf{0.40}, 0.05], \quad \mathbf{y}^+ = [0.40, \mathbf{0.80}, 0.55, 0.30].$$

The lower bound score vector assigns the highest score to class $k = 3$, whereas the upper bound score vector assigns the highest score to class $k = 2$. Consequently, the final predicted class is class 2.

The proposed decision rule prioritizes the upper-bound score vector because it represents the maximum confidence supported by the interval-valued prediction. This conserves the most influential prediction, and provides a consistent decision strategy for resolving disagreements between the lower- and upper-bound predictions while preserving the semantic interpretation of the interval output.

2.3 Performance Indices of the Granular Classifier

The proposed Granular Neural Network (GNN) is evaluated using two complementary groups of performance measures. The first group assesses the quality of the generated interval predictions in terms of coverage and precision and is used for the formation of the GNN. The second group provides a semantic interpretation of the interval-valued predictions through the proposed INS mapping, enabling a richer characterization of prediction confidence and uncertainty.

2.3.1 Granular Classifier Performance Metrics

The interval predictions produced by the GNN are first evaluated using interval-based performance measures. These metrics assess the ability of the classifier to produce prediction intervals that are both reliable and informative. Specifically, the intervals should achieve high coverage while remaining as narrow as possible. These measures are also used as objective functions in optimizing the proposed GNN.

- **Coverage:** Coverage measures the proportion of target labels that are captured by their corresponding predicted score intervals. It reflects how well the generated intervals contain the correct class while accounting for prediction uncertainty.

The coverage metric is defined as

$$Coverage = \begin{cases} 1, & \max(y_m^+) = y_{t,m}^+ \quad \& \quad \max(y_m^-) = y_{t,m}^- \\ \frac{1}{M} \sum_{m=1}^M \mathcal{O}(Y_{k,m}, Y_{t,m}), & \text{otherwise,} \end{cases} \quad (17)$$

where

$$Y_{k,m} = [y_{k,m}^-, y_{k,m}^+], \quad Y_{t,m} = [y_{t,m}^-, y_{t,m}^+],$$

represent the predicted and target score intervals for sample m , respectively, and M is the total number of evaluation samples.

The overlap function is defined as

$$\mathcal{O}(Y_{k,m}, Y_{t,m}) = \max(0, \min(y_{k,m}^+, y_{t,m}^+) - \max(y_{k,m}^-, y_{t,m}^-)).$$

The overlap function simply determines whether the target class score lies within the predicted interval. Alternatively, partial overlap may be used to provide a graded measure of coverage instead of a binary decision.

- **Specificity:** Specificity quantifies the average width of the predicted score intervals. Higher specificity corresponds to narrower intervals and therefore more precise predictions. Specificity is defined as

$$Specificity = 1 - \frac{1}{M} \sum_{m=1}^M |y_m^+ - y_m^-|. \quad (18)$$

- **Coverage–Specificity Trade-off (V):**

Coverage and specificity are naturally conflicting objectives. Increasing coverage generally results in wider prediction intervals, whereas increasing specificity encourages narrower intervals. To balance these objectives, the following performance index is employed:

$$V = Coverage \times Specificity. \quad (19)$$

The product formulation is intended to represent the joint satisfaction of the two criteria rather than to optimize either criterion independently. Since $(Coverage, Specificity \in [0, 1])$, it follows that:

$$0 \leq V \leq 1. \quad (20)$$

Thus, (V) is monotonically non-decreasing with respect to either criterion when the other is fixed. The formulation also makes the indicator sensitive to poor performance in either component. Hence, a high value of one criterion alone is insufficient to produce a high joint-quality value.

The indicator should therefore be interpreted as a bounded scalar measure of joint coverage-specificity quality. It is not claimed to be the unique or universally optimal aggregation function; rather, it provides a simple and interpretable criterion for comparing granular representations in which both coverage and specificity are considered desirable simultaneously.

The following example illustrates the computation of these metrics.

$$\mathbf{y}_m^- = [y_1^-, y_2^-, y_3^-], \quad \mathbf{y}_m^+ = [y_1^+, y_2^+, y_3^+].$$

m	True class t	$\mathbf{y}_m^-$	$\mathbf{y}_m^+$
1	1	[0.7,0.1,0.0]	[0.9,0.1,0.2]
2	2	[0.3,0.4,0.1]	[0.5,0.8,0.3]
3	3	[0.2,0.2,0.5]	[0.4,0.3,0.7]
4	2	[0.6,0.2,0.3]	[0.8,0.7,0.6]

For complete interval coverage,

$$Cov = \frac{1}{4}(1 + 1 + 1 + 1) = 1.$$

Using partial overlap,

$$Cov = \frac{1}{4}(1 + 1 + 1 + 0.1) = 0.775.$$

The specificity is

$$Spec = 1 - \frac{1}{4}(0.2 + 0.4 + 0.2 + 0.2) = 0.75.$$

Therefore,

$$V = Cov \times Spec = 1 \times 0.75 = 0.75,$$

or, using partial coverage,

$$V = 0.775 \times 0.75 = 0.5813.$$

2.3.2 INS-Based Semantic Evaluation

Although Coverage and Specificity assess the quality of the generated interval predictions, they do not provide a semantic interpretation of the uncertainty associated with those predictions. To address this limitation, we introduce a new Interval Neutrosophic Set (INS) mapping that transforms the interval-valued class scores produced by the GNN into three complementary semantic components: Truth, Intermediacy, and Falsity. These components characterize the confidence assigned to the target class, the uncertainty arising from overlapping class predictions, and the strongest competing evidence against the target class.

Based on the proposed INS mapping, each interval-valued prediction generated by the GNN is represented by the corresponding Truth (Eq. 14), Intermediacy (Eq. 15), and Falsity (Eq. 16) intervals. These semantic components are subsequently used to determine the final class prediction and to provide an interpretable assessment of the prediction confidence and uncertainty.

INS Performance Measures. The semantic representation provided by the proposed INS mapping enables the use of Interval Neutrosophic performance measures to evaluate the quality of interval-valued predictions.

INS Score. The INS score jointly evaluates the Truth, Intermediacy, and Falsity components of each prediction. For a single prediction, the score is computed as [24]

$$\text{Score}_m = \frac{T_m^- + T_m^+ + (1 - I_m^-) + (1 - I_m^+) + (1 - F_m^-) + (1 - F_m^+)}{6}. \quad (21)$$

Where T_m , F_m , and I_m are the Truth, Falsity and Intermediacy of each sample m respectively. In the case of an $\emptyset$ F_m or I_m are calculated as zero. The average INS score over all M evaluation samples is

$$\text{Score} = \frac{1}{M} \sum_{m=1}^M \text{Score}_m. \quad (22)$$

Example. Using the interval predictions presented in Examples A and B, the corresponding INS scores are obtained directly from Eq. (21). For Example A (confident prediction), the INS Score is $\text{Score} = 0.86$, reflecting a highly confident prediction with no ambiguity or competing evidence. For Example B (competing prediction), the INS Score is $\text{Score} = 0.415$, indicating increased uncertainty and stronger competing evidence against the target class.

INS Accuracy.

While the INS Score provides an overall semantic evaluation by jointly considering the Truth, Intermediacy, and Falsity components, the INS-Accuracy measures the confidence of an interval prediction by balancing the evidence supporting the target class (Truth) against the strongest competing evidence (Falsity) [25]. The index is normalized to the range $[0, 1]$, where values close to 1 indicate high-confidence predictions and values close to 0 indicate weak or highly contested predictions.

For a single prediction, the INS-Accuracy is defined as:

$$\text{Accuracy}_m = \frac{(T_m^- + T_m^+) - (F_m^- + F_m^+)}{4}. \quad (23)$$

The average Accuracy over all M evaluation samples is

$$\text{Accuracy} = \frac{1}{M} \sum_{m=1}^M \text{Accuracy}_m. \quad (24)$$

Example. Using the interval predictions presented in Examples A and B, the corresponding Accuracies are computed from Eq. (23). For Example A, the value is $\text{Accuracy} = 0.29$, because the target class is strongly supported with no competing evidence. For Example B, the Accuracy is $\text{Accuracy} = -0.103$, indicating that the competing class provides substantial evidence against the target class, resulting in negative accuracy. The lower Accuracy indicates that the competing class provides substantial evidence against the target class, resulting in a less confident prediction.

The INS-based semantic evaluation complements the interval-performance measures by providing an interpretable characterization of prediction confidence, ambiguity, and competing evidence. The INS Score offers an overall semantic assessment by jointly considering Truth, Intermediacy, and Falsity, whereas the Accuracy quantifies the balance between supporting and competing evidence. Together, these measures provide a more comprehensive evaluation of interval-valued predictions than conventional classification accuracy.

Design Process Summary

The GNN development process involves the following steps:

1. Train a standard (numeric) neural network using labeled training data.
2. Assign an initial information granularity level ϵ to the network.
3. Select an allocation protocol and apply particle swarm optimization (PSO) to refine the distribution of ϵ .
4. Evaluate the GNN using the proposed interval performance metrics, including Coverage, Specificity, Truth, Intermediacy, Falsity, INS Score, and INS Accuracy.

Optimization via PSO Particle Swarm Optimization (PSO) [26] is employed to find optimal ϵ values for each C3 and C4. Each particle in the swarm represents a granularity vector, and its fitness is computed using the coverage objective Q .

The PSO update equations are:

$$v_i^{(t+1)} = \omega v_i^{(t)} + c_1 r_1 (p_i - x_i) + c_2 r_2 (g - x_i), \quad (25)$$

$$x_i^{(t+1)} = x_i^{(t)} + v_i^{(t+1)}, \quad (26)$$

where x_i and v_i are position and velocity, p_i is the best position of particle i , g is the global best, and $r_1, r_2 \sim \mathcal{U}(0, 1)$.

The proposed performance index V (19) provides a practical implementation of the principle of justifiable granularity [27, 28]. According to this principle, information granules should achieve a balance between experimental evidence, represented by high coverage, and semantic specificity, represented by narrow and discriminative intervals. The four allocation protocols (C1–C4) are therefore evaluated using the common objective V , while the optimization-based protocols (C3 and C4) employ PSO to identify interval bounds that best satisfy this balance. Consequently, the resulting information granules are both representative of the underlying uncertainty and sufficiently precise for reliable classification.

3. Experiments and Results

3.1 Datasets

A total of nine datasets were utilized in this experiment, each differing in the number of features and classes. The feature list include values both positive and negative values. The datasets employed are listed in Table 1 as follows:

Table 1

Experiment datasets

Name	Description
Breast Cancer Wisconsin (Diagnostic) [29]	Features from breast mass FNA digitized image of cell nuclei present in the image.
Iris [30]	Plant features set contains 3 classes of 50 instances each, where each class refers to a type of iris plant.
Yeast [31]	Contains biological data used for classifying protein cellular localization sites within a yeast cell.
human activity [32]	Matlab Dataset containing 24,075 observations of five physical human activities. Each observation has 60 features.
Heart failure [33]	Predicting heart disease events.
Adult Autism Screening [34]	Autistic Spectrum Disorder Screening Data for Adult. This dataset is related to classification and predictive tasks.
Weather type [35]	Dataset is synthetically generated to mimic weather data for classification tasks.
Crop Recommendation [36]	Dataset predict the most suitable crop to grow based on several agro-climatic parameters.
E coli [37]	Contains 336 protein samples used to predict their subcellular localization in E coli.

Name	# of Features
Breast Cancer Wisconsin (Diagnostic) [29]	30
Iris [30]	4
Yeast [31]	8
human activity [32]	60
Heart failure [33]	11

Table 1

Continued

Name	# of Features
Adult Autism Screening [34]	12
Weather type [35]	10
Crop Recommendation [36]	7
E coli [37]	7
Name	# of Classes
Breast Cancer Wisconsin (Diagnostic) [29]	2
Iris [30]	3
Yeast [31]	10
human activity [32]	6
Heart failure [33]	2
Adult Autism Screening [34]	2
Weather type [35]	4
Crop Recommendation [36]	20
E coli [37]	8

3.2 Experimental Setup and Implementation Details

The experiment conducted is detailed in Table 2. To ensure the rigidity of the results produced,

Table 2

Experimental setup and implementation details

Parameter	Configuration
Software	MATLAB R2022b
Random seed	<code>rng(seed)</code>
seed	[15, 70, 256, 389, 512, 641, 773, 881, 947, 1024]
Data split	70% training / 30% testing (<code>cvpartition</code>)
Baseline architecture	<code>patternnet(10)</code> ; 1 hidden layer, 10 neurons
Hidden activation	Hyperbolic tangent sigmoid (<code>tansig</code>)
Output activation	Softmax (baseline); linear (granular classifier)
Training algorithm	<code>trainscg</code>
Loss function	Cross-entropy
Maximum epochs	1000
Granularity protocols	C1–C5
PSO protocols	C3–C4
D	Number of PSO optimization variables
ϵ_i bounds	$[0, 1]$ $i = 1, \dots, D$,
Swarm size	$\min(100, 10D)$
Maximum iterations	$200D$
c_1, c_2	1.49, 1.49
Inertia range	$[0.1, 1.1]$
Function tolerance	10^{-6}
Objective	$V = \text{Cov} \times \text{Spec}$

they were reproduced with 10 different randomization seeds. The final results reported of any metric

x are in the form of $(\bar{x} \pm s)$ such that :

$$\bar{x} = \frac{1}{R} \sum_{r=1}^R x_r \quad (27)$$

$$s = \sqrt{\frac{1}{R-1} \sum_{r=1}^R (x_r - \bar{x})^2}, \quad (28)$$

3.3 Results and discussion

All experiments employed the same neural network architecture with one hidden layer containing 10 neurons. The data were randomly divided into training (70%) and testing (30%) sets. A fixed training protocol and identical hyperparameters were used across all experiments to ensure a fair comparison and to isolate the effect of the proposed granular information allocation strategy. The relatively compact architecture was intentionally selected to reduce the influence of model complexity, allowing the performance differences to be primarily attributed to the proposed GNN framework rather than increased network capacity. Network performance was evaluated using the cross-entropy loss function

The information granules of the neural network were formed after obtaining the parameters from the training phase using the allocation of information granules of several allocation protocols.

The performance of the GNN is first evaluated using coverage, specificity, and their product, which shows the overall balance achieved by the granular structure of the neural network. To provide a comprehensive measure of GNN performance, the area under the curve (AUC) of the coverage-specificity curve is calculated. This metric captures the combined performance of the model across different levels of granularity.

AUC represents the area between the evaluated performance curve and the (ϵ) -axis. Therefore, (ϵ) is the independent variable and the integration axis. For each calculated metric at every ϵ the AUC (Area Under the Curve) is calculated as:

$$\int_0^1 metric(\epsilon) \quad (29)$$

The area was then numerically approximated using the trapezoidal rule:

$$AUC \approx \sum_{i=1}^{M-1} \frac{metric(\epsilon_i) + metric(\epsilon_{i+1})}{2} (\epsilon_{i+1} - \epsilon_i), \quad (30)$$

where (M) is the number of evaluated $(\epsilon \in [0, 1])$ values. Thus, the AUC does not represent integration with respect to the metric. Rather, it quantifies the accumulated value of the plotted performance measure over the investigated range of the granularity parameter (ϵ) .

The GNNs were developed based on the four granularity allocation protocols described in section 2.1. To establish a reference for evaluation, a fifth protocol employing a random granularity allocation was implemented as a benchmark. Comparing the results of the systematic protocols against this random baseline enables a clear assessment of the effectiveness of structured granularity allocation in enhancing the overall performance and stability of the GNN.

In determining the predicted class, a particular challenge arises due to the probabilistic and interval-valued nature of the outputs. In conventional (crisp) neural networks, the class corresponding to the highest output score is directly assigned as the predicted label. However, in the GNN, the outputs are represented as intervals that capture the uncertainty associated with each class, with each interval constrained within the range $[0,1]$. These intervals are generated by two parallel sub-networks: one estimating the lower bounds and the other estimating the upper bounds of the class predictions.

In standard classification networks, the softmax layer produces normalized predictions emphasizing the most likely class. However, in the GNN, this behavior suppresses the interval variability

required to express uncertainty. To enable interval formation, the softmax activation was replaced with a purelin (linear) activation function, allowing the network outputs to preserve the true range of computed predictions without forced normalization. This modification enables the direct construction of interval-valued outputs that better represent the inherent uncertainty captured by the granular model.

Figure 2 presents the Coverage (partial coverage), Specificity, and their product, V , as functions of ϵ for the Crops dataset. For protocols C1 and C2, the optimal value of ϵ was identified as 0.45. For C3 and C4, the optimal allocation parameters were determined using the PSO algorithm, yielding the best-performing ϵ values of 0.7 and 0.9, respectively. To quantify the overall performance of each protocol, the Area Under the Curve (AUC) of the V curve was computed. For C2, the analysis indicates that $\gamma=0.7$ provides the best trade-off based on the AUC of the performance indicator V . Furthermore, the AUC results show that C4 achieved the highest overall performance, demonstrating that optimized granular information allocation, particularly through PSO, enhances the adaptability and informativeness of the generated intervals. It should be noted that using total coverage instead of partial coverage would increase the coverage values, leading to correspondingly higher values of V and its AUC.

Table 3
Granular allocation techniques at optimal ϵ (testing set)

Dataset	Metric	C1	C2	C3	C4	C5
Cancer	cov	.136 ± .01	.105 ± .035	.845 ± .099	.974 ± .018	.096 ± .039
	spec	.713 ± .025	.768 ± .065	.672 ± .281	.890 ± .116	.785 ± .077
	V	.083 ± .003	.068 ± .018	.519 ± .257	.868 ± .120	.062 ± .020
iris	cov	.145 ± .023	.142 ± .017	.924 ± .032	.966 ± .014	.127 ± .023
	spec	.689 ± .046	.690 ± .024	.969 ± .006	.971 ± .007	.720 ± .037
	V	.08 ± .007	.078 ± .007	.895 ± .031	.937 ± .015	.067 ± .012
Hu-Act	cov	.192 ± .038	.087 ± .045	.792 ± .120	.966 ± .010	.087 ± .045
	spec	.57 ± .07	.705 ± .073	.773 ± .078	.872 ± .030	.703 ± .074
	V	.082 ± .006	.051 ± .026	.570 ± .100	.839 ± .029	.050 ± .025
yeast	cov	.078 ± .029	.095 ± .022	.650 ± .033	.911 ± .015	.089 ± .020
	spec	.61 ± .025	.579 ± .029	.966 ± .004	.926 ± .006	.593 ± .045
	V	.038 ± .015	.041 ± .012	.626 ± .030	.840 ± .014	.040 ± .011
autism	cov	.0858 ± .0098	.085 ± .012	.935 ± .014	.957 ± .014	.087 ± .017
	spec	.8028 ± .0178	.809 ± .023	.958 ± .010	.962 ± .008	.806 ± .035
	V	.0618 ± .0052	.060 ± .007	.896 ± .016	.921 ± .013	.054 ± .009
heart	cov	.0997 ± .0086	.101 ± .008	.861 ± .011	.892 ± .013	.089 ± .014
	spec	.76 ± .0151	.759 ± .014	.939 ± .010	.943 ± .011	.785 ± .029
	V	.0672 ± .0041	.067 ± .004	.808 ± .018	.841 ± .010	.062 ± .007
weather	cov	.1112 ± .0157	.151 ± .021	.711 ± .037	.867 ± .014	.143 ± .025
	spec	.703 ± .0487	.614 ± .024	.928 ± .004	.916 ± .010	.632 ± .042
	V	.067 ± .0066	.072 ± .007	.658 ± .034	.793 ± .019	.067 ± .007
crops	cov	.018 ± .0147	.130 ± .025	.327 ± .070	.901 ± .026	.137 ± .027
	spec	.494 ± .0468	.453 ± .034	.973 ± .004	.959 ± .002	.458 ± .046
	V	.0058 ± .0051	.038 ± .005	.318 ± .068	.861 ± .027	.039 ± .006
ecoli	cov	.1281 ± .0307	.187 ± .033	.686 ± .059	.924 ± .012	.177 ± .034
	spec	.5412 ± .0535	.513 ± .029	.962 ± .003	.945 ± .009	.529 ± .050
	V	.0496 ± .0118	.065 ± .009	.659 ± .057	.871 ± .014	.059 ± .011

The performance indices for the remaining datasets are summarized in Table 3. The smoothness of the curves varied across the datasets; however, a general observation was that a distinct peak occurred at a specific value of ϵ , indicating the presence of an optimal granule structure. The average AUC value

across all datasets is listed in Table 4. Protocol C4 demonstrated the highest flexibility, as it permits simultaneous optimization of the interval size, bounds, and uniformity. Although using PSO adds extra computational overhead, the observed performance improvements are significant and justify its use for achieving enhanced granularity-based performance.

Table 4
Average AUC of Granular allocation techniques

Protocol	Coverage	Specificity	V
C1	0.1103 ± 0.006994	0.653 ± 0.0107	0.059 ± 0.00245
C2	0.1203 ± 0.00991	0.6545 ± 0.0168	0.06 ± 0.00396
C3	0.7485 ± 0.017	0.9033 ± 0.031	0.660 ± 0.0289
C4	0.9286 ± 0.00351	0.93 ± 0.0121	0.863 ± 0.0141
C5	0.115 ± 0.01087	0.668 ± 0.0189	0.0556 ± 0.0044

Another perspective on the performance of the protocols is provided by evaluating the Truth, Falsity, and Intermediacy measures using INS (refer to section 2.3.3). In our case, the cross-validation using INS characteristics further highlights the implications of free granular allocation. As shown in Figure 3, the widest intervals of Intermediacy and Falsity appear in protocols C3 and C4, where the granule allocation is most flexible. Interestingly, the Score values derived from INS characteristics do not always correspond directly to the indicator V presented in Figure 2. This divergence stems from the distinct nature of the metrics: while Coverage and Specificity quantify the size of a granule and how effectively it encloses the possible solution, the INS measures reflect the semantic quality of the granule, its interpretability, precision, and relevance to the target. Since the GNNs were optimized for the highest value of V , achieving a configuration that maximizes Truth (or overall INS accuracy) represents an ideal and desirable outcome. The classification performance, as reflected by the high Truth levels, thus remains robust and encouraging. Some variance in our results, however, can be explained by the small size of the network used in this experiment.

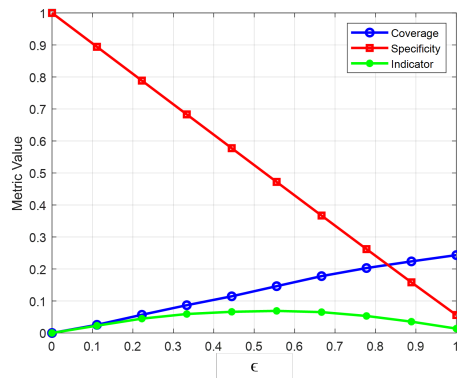
The INS-based evaluation offers a semantic interpretation of these results. In Table 5, all protocols appear to have similar performance, while protocols C1 and C5, despite showing higher Truth and Score, perform poorly on coverage and V . Meanwhile, by averaging results from all datasets as shown in Table 6, C3 and C4 achieve relatively lower Truth but significantly better balance between predictive Index and uncertainty representation. This reveals a meaningful trade-off: higher semantic certainty (Truth) does not always correspond to higher predictive quality. The results confirm that optimized granular protocols, particularly C4, produce intervals that are both protective and discriminative, capturing the underlying uncertainty while maintaining robust classification performance.

Pairwise differences between the five configurations were assessed using two-sided Wilcoxon signed-rank tests across the 90 paired observations. To account for multiple pairwise comparisons, p-values were adjusted using the Holm procedure (refer to Table 7). The rank-biserial correlation (RBC) were reported as measures of effect magnitude and direction. It is evident that over all the ranking from top to bottom is $C4 \rightarrow C3 \rightarrow C2 \rightarrow C1 \rightarrow C5$.

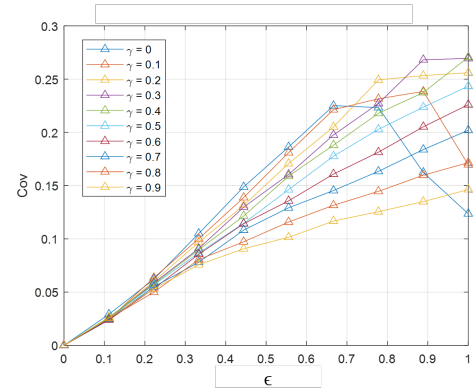
where the differences (d_i) for a comparison of (C_i Vs C_j) is $d_i = C_i - C_j$ such that more positive ranks reflect on a higher rank of C_i while more negative ranks reflect on a higher rank for C_j .

For comparison, two established interval/set-valued classification frameworks were evaluated on the same datasets: Conformal Predictors (CP) [38, 39] and the Least Ambiguous Set-Valued Classifier with Bounded Error Levels (LABEL) [40]. Both methods generate a prediction set of candidate class labels based on the probabilistic outputs of the underlying neural network.

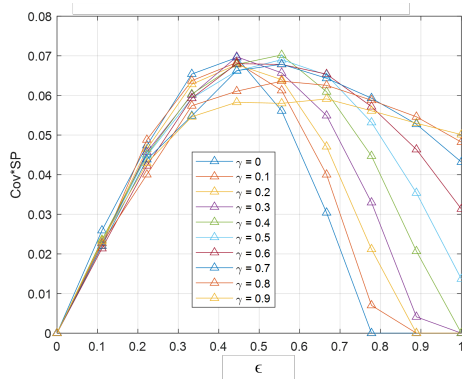
To ensure a fair comparison, the held-out data (30% of the original) were then divided into separate calibration and test subsets (15% of the original dataset each). The calibration subset was used to estimate the parameters required by the uncertainty-quantification methods, whereas the test subset was used exclusively for the final evaluation. Consequently, the test data were not used during model



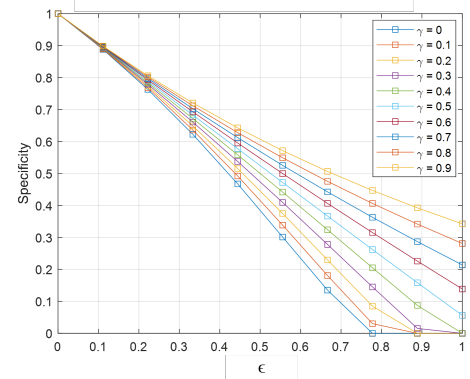
(a) C1



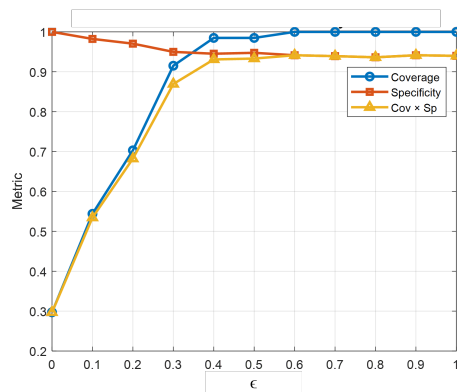
(b) C2 Coverage



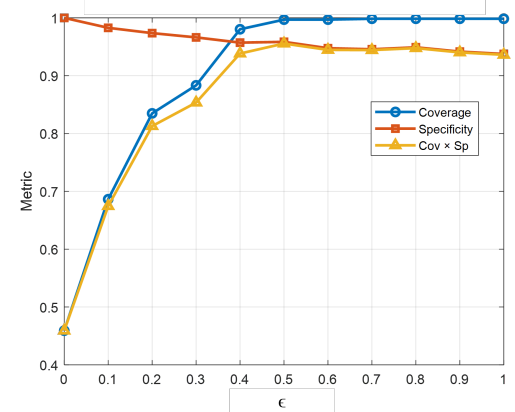
(c) C2 Indicator(V)



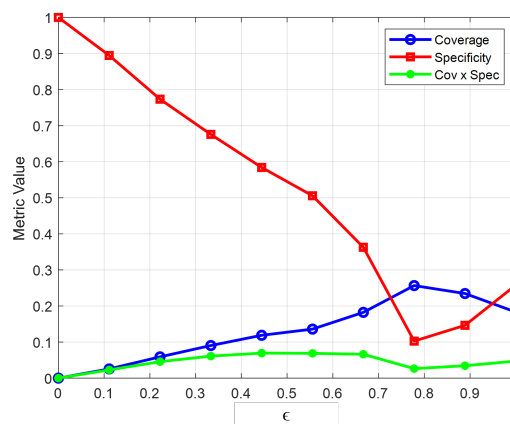
(d) C2 Specificity



(e) C3

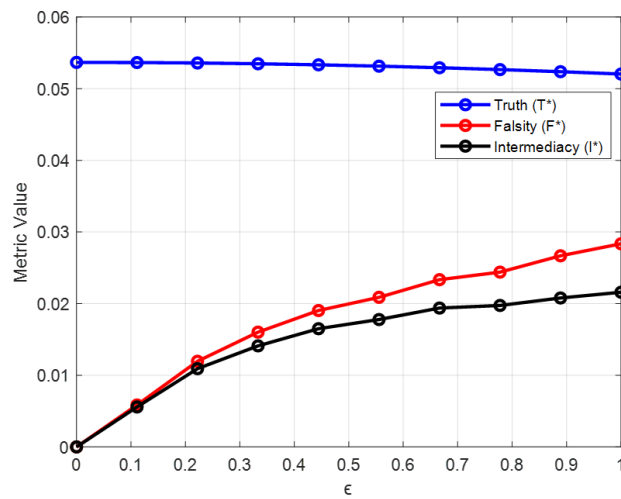


(f) C4

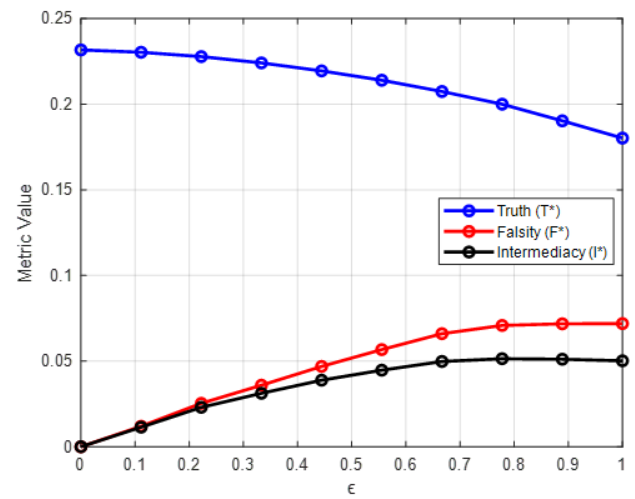


(g) C5

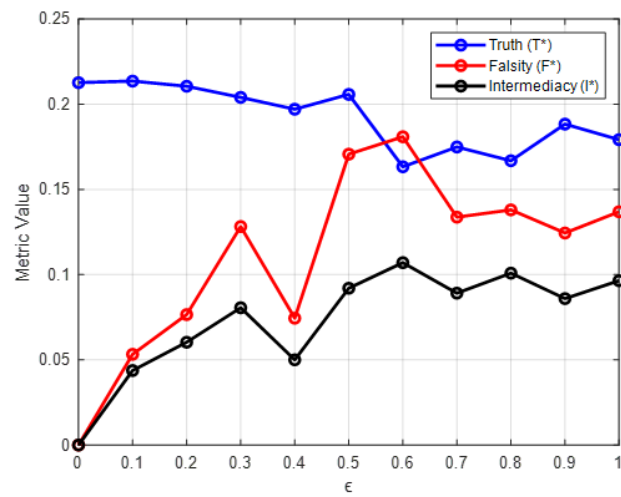
Fig. 2. Crops dataset results (testing set)



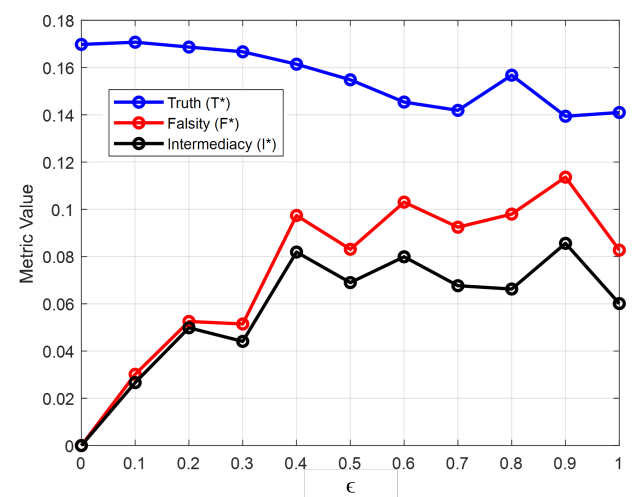
(a) C1



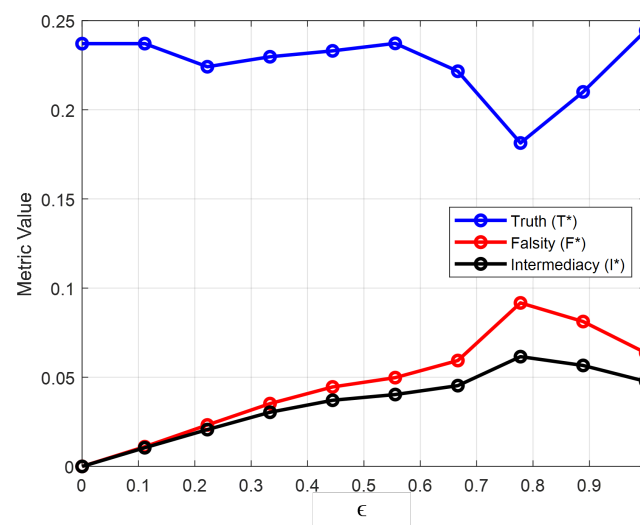
(b) C2



(c) C3



(d) C4



(e) C5

Fig. 3. Median INS characteristics of the Crops dataset

Table 5

AUC of Granular allocation techniques in terms of INS characteristics and metrics. (The midpoint of the T, F, and I is used to calculate AUC across all ϵ).

Dataset	Metric	C1	C2	C3	C4	C5
Cancer	T^*	.820 $\pm$.027	.537 $\pm$.020	.593 $\pm$.059	.611 $\pm$.047	.782 $\pm$.123
	F^*	.005 $\pm$.005	.035 $\pm$.004	.097 $\pm$.069	.120 $\pm$.071	.013 $\pm$.014
	I^*	.005 $\pm$.004	.030 $\pm$.004	.090 $\pm$.063	.115 $\pm$.065	.012 $\pm$.013
	Score	.937 $\pm$.011	.824 $\pm$.008	.802 $\pm$.055	.792 $\pm$.051	.919 $\pm$.049
	Accuracy	.408 $\pm$.015	.251 $\pm$.011	.248 $\pm$.054	.746 $\pm$.047	.385 $\pm$.068
iris	T^*	.670 $\pm$.035	.723 $\pm$.027	.696 $\pm$.038	.798 $\pm$.059	.782 $\pm$.027
	F^*	.013 $\pm$.007	.012 $\pm$.007	.023 $\pm$.011	.030 $\pm$.020	.010 $\pm$.006
	I^*	.012 $\pm$.006	.011 $\pm$.007	.023 $\pm$.011	.029 $\pm$.018	.009 $\pm$.006
	Score	.881 $\pm$.015	.900 $\pm$.010	.883 $\pm$.016	.913 $\pm$.030	.921 $\pm$.010
	Accuracy	.328 $\pm$.020	.355 $\pm$.014	.336 $\pm$.021	.884 $\pm$.037	.386 $\pm$.014
Hu-Act	T^*	.426 $\pm$.069	.303 $\pm$.069	.226 $\pm$.016	.221 $\pm$.014	.303 $\pm$.069
	F^*	.031 $\pm$.012	.036 $\pm$.024	.104 $\pm$.038	.108 $\pm$.012	.035 $\pm$.025
	I^*	.028 $\pm$.012	.033 $\pm$.023	.072 $\pm$.016	.084 $\pm$.010	.031 $\pm$.023
	Score	.789 $\pm$.029	.745 $\pm$.028	.683 $\pm$.021	.677 $\pm$.006	.746 $\pm$.027
	Accuracy	.198 $\pm$.039	.133 $\pm$.036	.061 $\pm$.024	.557 $\pm$.007	.134 $\pm$.035
yeast	T^*	.183 $\pm$.023	.238 $\pm$.013	.180 $\pm$.017	.198 $\pm$.013	.243 $\pm$.012
	F^*	.052 $\pm$.025	.067 $\pm$.028	.079 $\pm$.018	.123 $\pm$.017	.062 $\pm$.028
	I^*	.043 $\pm$.021	.053 $\pm$.025	.071 $\pm$.017	.098 $\pm$.014	.050 $\pm$.025
	Score	.696 $\pm$.014	.706 $\pm$.019	.676 $\pm$.010	.659 $\pm$.007	.711 $\pm$.019
	Accuracy	.066 $\pm$.014	.086 $\pm$.017	.050 $\pm$.009	.537 $\pm$.004	.091 $\pm$.017
autism	T^*	.817 $\pm$.016	.832 $\pm$.020	.864 $\pm$.015	.892 $\pm$.013	.845 $\pm$.022
	F^*	.007 $\pm$.003	.007 $\pm$.003	.012 $\pm$.002	.013 $\pm$.004	.004 $\pm$.003
	I^*	.006 $\pm$.003	.006 $\pm$.003	.012 $\pm$.002	.013 $\pm$.004	.004 $\pm$.003
	Score	.935 $\pm$.007	.940 $\pm$.006	.947 $\pm$.006	.955 $\pm$.006	.946 $\pm$.009
	Accuracy	.405 $\pm$.009	.413 $\pm$.010	.426 $\pm$.008	.939 $\pm$.008	.421 $\pm$.012
heart	T^*	.708 $\pm$.021	.705 $\pm$.025	.721 $\pm$.025	.762 $\pm$.029	.744 $\pm$.026
	F^*	.007 $\pm$.003	.078 $\pm$.011	.021 $\pm$.006	.021 $\pm$.005	.006 $\pm$.003
	I^*	.007 $\pm$.003	.072 $\pm$.011	.020 $\pm$.006	.020 $\pm$.005	.006 $\pm$.003
	Score	.898 $\pm$.007	.852 $\pm$.006	.893 $\pm$.012	.907 $\pm$.013	.911 $\pm$.010
	Accuracy	.351 $\pm$.011	.313 $\pm$.009	.350 $\pm$.015	.870 $\pm$.017	.369 $\pm$.014
weather	T^*	.363 $\pm$.045	.540 $\pm$.057	.364 $\pm$.045	.540 $\pm$.052	.548 $\pm$.061
	F^*	.039 $\pm$.010	.025 $\pm$.005	.068 $\pm$.012	.055 $\pm$.019	.020 $\pm$.007
	I^*	.036 $\pm$.009	.022 $\pm$.005	.061 $\pm$.012	.047 $\pm$.017	.018 $\pm$.006
	Score	.763 $\pm$.018	.831 $\pm$.021	.745 $\pm$.021	.813 $\pm$.029	.837 $\pm$.024
	Accuracy	.162 $\pm$.025	.258 $\pm$.030	.148 $\pm$.027	.743 $\pm$.035	.264 $\pm$.033
crops	T^*	.055 $\pm$.014	.223 $\pm$.036	.054 $\pm$.011	.223 $\pm$.036	.235 $\pm$.041
	F^*	.011 $\pm$.007	.050 $\pm$.015	.017 $\pm$.005	.106 $\pm$.019	.050 $\pm$.016
	I^*	.008 $\pm$.005	.036 $\pm$.011	.013 $\pm$.005	.082 $\pm$.016	.036 $\pm$.011
	Score	.679 $\pm$.003	.712 $\pm$.012	.675 $\pm$.002	.678 $\pm$.009	.716 $\pm$.014
	Accuracy	.022 $\pm$.005	.086 $\pm$.016	.019 $\pm$.004	.559 $\pm$.013	.093 $\pm$.020
ecoli	T^*	.288 $\pm$.040	.504 $\pm$.070	.286 $\pm$.038	.507 $\pm$.093	.529 $\pm$.076
	F^*	.026 $\pm$.015	.029 $\pm$.008	.044 $\pm$.014	.077 $\pm$.020	.018 $\pm$.006
	I^*	.022 $\pm$.012	.023 $\pm$.007	.040 $\pm$.013	.064 $\pm$.015	.015 $\pm$.006
	Score	.747 $\pm$.020	.818 $\pm$.024	.734 $\pm$.020	.788 $\pm$.041	.832 $\pm$.025
	Accuracy	.131 $\pm$.025	.238 $\pm$.036	.121 $\pm$.024	.715 $\pm$.055	.255 $\pm$.038

Table 6

Average AUC of Granular allocation techniques in terms of INS characteristics

Protocol	Truth	Falsity	Intermediacy	Score	INS Accuracy
C1	.481 ± .013	.021 ± .0037	.018 ± .003	.814 ± .005	.23 ± .007
C2	.538 ± .02	.035 ± .0045	.03 ± .004	.824 ± .008	.251 ± .011
C3	.443 ± .0087	.051 ± .007	.044 ± .0055	.783 ± .004	.196 ± .005
C4	.528 ± .016	.0725 ± .0079	.061 ± .0075	.798 ± .008	.728 ± .0095
C5	.5568 ± .0204	.024 ± .005	.02 ± .004	.838 ± .008	.266 ± .011

Table 7

Statistical comparison of granularity allocation protocols using the Wilcoxon signed-rank test

Comparison	Median difference	#Rank ⁺	#Rank ⁻	minRank	Holm-adjusted <i>p</i> – value	RBC
C1 vs C2	-0.00078	1904.5	2190.5	1904.5	0.565	-0.0698
C1 vs C3	-0.597	0	4095	0	1.744×10^{-15}	-1.0
C1 vs C4	-0.814	0	4095	0	1.744×10^{-15}	-1.0
C1 vs C5	+0.004	2445	1650	1650	0.2195	+0.1941
C2 vs C3	-0.597	0	4095	0	1.744×10^{-15}	-1.0
C2 vs C4	-0.806	0	4095	0	1.744×10^{-15}	-1.0
C2 vs C5	+0.0043	3378	717	717	2.588×10^{-7}	+0.65
C3 vs C4	-0.15	73	4022	73	7.782×10^{-15}	-0.96
C3 vs C5	+0.60	4095	0	0	1.744×10^{-15}	+1.0
C4 vs C5	+0.808	4095	0	0	1.744×10^{-15}	+1.0

training or calibration.

For Conformal Prediction, the trained NN produces a vector of class probabilities:

$$[p_{i,1}, p_{i,2}, \dots, p_{i,K}] \quad (31)$$

For each calibration sample (x_i), where (K) is the number of classes. For a calibration sample with true class (y_i), the nonconformity score used in our implementation is:

$$s_i = 1 - p_{i,y_i} \quad (32)$$

A calibration sample receives a smaller nonconformity score when the base classifier assigns a higher probability to its true class. The calibration scores are sorted, and for a specified non-conformity level (α), the conformal threshold is obtained at the index:

$$j = \lceil (N_{\text{cal}} + 1)(1 - \alpha) \rceil, \quad (33)$$

where (N_{cal}) is the number of calibration samples. In the experiments, ($\alpha = .05$), corresponding to a nominal coverage level of 95%. For a test sample (x_i), the conformal prediction set is consequently formed from the classes (k) whose nonconformity scores satisfy the calibrated threshold, i.e.,

$$[k : 1 - p_{i,k} \leq q_\alpha] \quad (34)$$

The empirical coverage is then evaluated exclusively on the independent test (the remaining 15%) subset as:

$$\text{Coverage}_{CP} = \frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} \mathbb{I}[y_i \in \Gamma(x_i)], \quad (35)$$

where ($\mathbb{I}[\cdot]$) is the indicator function. In addition to coverage, the Specificity is developed by calculating the predictions set size. The corresponding joint coverage-specificity indicator (V_{CP}) is calculated as the multiplication as defined previously.

The LABEL method is evaluated using the same trained base classifier and the same calibration/test

separation. For each class (k), the calibration procedure calculates a class-specific residual for every calibration sample:

$$|\mathbb{I}(y_i = k) - p_{i,k}|, \quad (36)$$

where $(\mathbb{I}(y_i = k))$ is equal to 1 when the calibration sample belongs to class (k), and 0 otherwise. The calculations produce the probability-interval procedure to construct an interval $[L_{i,k}, U_{i,k}]$.

For LABEL, empirical Coverage and Specificity are determined by whether the probability assigned to the true class lies within its corresponding predicted interval.

Using $Coverage_L$ and $Specificity_L$ the corresponding V_L is calculated.

Consequently, CP and LABEL are evaluated under the same overall experimental conditions, using the same trained base classifier, the same calibration/test framework, the same nominal confidence level, and the same test-set criteria of coverage, specificity, and the resulting joint indicator.

For all experiments, the confidence level was fixed at 95%.

Following the standard evaluation protocol adopted in the literature—empirically—coverage is considered successful if the prediction set contains the true class label, regardless of the number of labels in the set. In contrast, specificity penalizes prediction sets containing unnecessary labels; consequently, larger prediction sets yield lower specificity.

The comparative results are presented in Table 8. The primary difference between the methods is reflected in the performance index V , which combines coverage and specificity into a single measure of classification quality. The proposed framework consistently achieves higher V values than CP and LABEL, indicating a better balance between accurate coverage and informative predictions. Unlike the proposed method, which produces a single uncertainty-aware class label, CP and LABEL generate sets of candidate labels that, in some cases, include all possible classes, reducing their practical usefulness. Overall, the proposed framework provides higher-quality and more informative classifications while maintaining competitive coverage.

Table 8

Average AUC of Granular allocation techniques in addition to interval classification methods

Method	Coverage	Specificity	V
C3	0.7485 ± 0.017	0.9033 ± 0.031	0.660 ± 0.0289
C4	0.9286 ± 0.00351	0.93 ± 0.0121	0.863 ± 0.0141
Conformal Predictor	0.841 ± 0.2297	0.2571951 ± 0.267	0.1784 ± 0.172
LABEL predictors	1 ± 0	0.374 ± 0.1827	0.375 ± 0.183

The experimental results demonstrate that optimized granular information allocation substantially improves uncertainty quantification in neural network classification. The proposed GNN generates interval-valued predictions that capture predictive uncertainty while maintaining a favourable balance between coverage and specificity, as reflected by the consistently higher values of the performance index V . The performance of the PSO-based protocol (C4) indicates that adaptive granule allocation produces more informative and reliable uncertainty estimates than fixed or random allocation strategies. Furthermore, comparison with Conformal Prediction and LABEL classifiers shows that the proposed framework quantifies uncertainty without sacrificing decision informativeness, producing a single uncertainty-aware prediction rather than large sets of candidate labels. These findings demonstrate that the proposed GNN provides an effective and interpretable framework for uncertainty-aware classification when discussing empirical results.

4. Conclusion

This paper presented a Granular Neural Network (GNN) framework for uncertainty-aware classification, where interval-valued outputs explicitly quantify predictive uncertainty. By optimizing information granule allocation through several allocation protocols, including PSO-based strategies,

the proposed framework significantly improved the balance between coverage and specificity, as reflected by the performance index V. Among the evaluated protocols, C4 consistently achieved the highest overall performance, demonstrating that adaptive granule allocation produces more informative and reliable uncertainty estimates than fixed or random allocation schemes. Comparisons with Conformal Prediction and LABEL classifiers further showed that the proposed approach provides more discriminative predictions by generating a single uncertainty-aware class label rather than large prediction sets.

The complementary INS-based analysis revealed that semantic certainty does not necessarily correspond to superior predictive performance. While protocols C1 and C5 achieved higher Truth values, they produced less effective uncertainty estimates, whereas C3 and C4 provided a better balance between predictive quality and uncertainty representation. These findings highlight that effective uncertainty quantification requires not only semantically meaningful intervals but also an appropriate balance between interval coverage and precision. Overall, the proposed GNN offers an interpretable and robust framework for uncertainty-aware classification, making it well suited for applications involving noisy, ambiguous, or critical data.

Future work will extend the proposed framework to deep architectures, particularly Convolutional Neural Networks (CNNs), by incorporating interval-based representations within convolutional layers. In addition, information granule optimization will be investigated using INS characteristics, namely Truth, Falsity, and Intermediacy, alongside Coverage and Specificity, to develop more informative and semantically guided uncertainty-aware learning strategies.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgement

The authors acknowledge the Telecommunications Regulatory Authority (TRA) – Oman for supporting article publication costs through the UNESCO Chair on Artificial Intelligence at the Communications & Information Research Center, Sultan Qaboos University.

Funding

The publication of the work reported in this article was supported by the Telecommunications Regulatory Authority (TRA)-Oman, through its funding of the UNESCO Chair on Artificial Intelligence at The Communication and Information Research Center (CIRC), Sultan Qaboos University.

References

- [1] Ming Li Song and Witold Pedrycz. Granular Neural Networks: Concepts and Development Schemes. *IEEE Transactions on Neural Networks and Learning Systems*. (2013). Granular Neural Networks: Concepts and Development Schemes. *IEEE Transactions on Neural Networks and Learning Systems*, 24(4), 542-553. <https://doi.org/10.1109/tnnls.2013.2237787>
- [2] Yang Liu and Jun Zhao and Dianhui Wang and Witold Pedrycz. Prediction Intervals for Granular Data Streams Based on Evolving Type-2 Fuzzy Granular Neural Network Dynamic Ensemble. *IEEE Transactions on Fuzzy Systems*. (2021). Prediction Intervals for Granular Data Streams Based on Evolving Type-2 Fuzzy Granular Neural Network Dynamic Ensemble. *IEEE Transactions on Fuzzy Systems*, 29(4), 874-888. <https://doi.org/10.1109/tfuzz.2020.2966172>
- [3] Rami Al-Hmouz and Witold Pedrycz and Abdullah Saeed Balamash and Ali Morfeq. Description and classification of granular time series. *Soft Computing*. (2014). Description and classification of granular time series. *Soft Computing*, 19(4), 1003 - 1017. <https://doi.org/10.1007/s00500-014-1311-z>
- [4] George Panoutsos and Mahdi Mahfouf. A neural-fuzzy modelling framework based on granular computing: Concepts and applications. *Fuzzy Sets Syst*. (2010). A neural-fuzzy modelling framework based on granular computing: Concepts and applications. *Fuzzy Sets Syst*, 161(21), 2808-2830. <https://doi.org/10.1016/j.fss.2010.06.004>
- [5] Hisao Ishibuchi and Hideo Tanaka and Hidehiko Okada. An architecture of neural networks with interval weights and its application to fuzzy regression analysis. *Fuzzy Sets and Systems*. (1993). An architecture of neural networks with interval weights and its application to fuzzy regression analysis. *Fuzzy Sets and Systems*, 57(1), 27-39. [https://doi.org/10.1016/0165-0114\(93\)90118-2](https://doi.org/10.1016/0165-0114(93)90118-2)
- [6] Ming Li Song and Yukai Jing and Witold Pedrycz. Granular neural networks: A study of optimizing allocation of in-

- formation granularity in input space. *Appl. Soft Comput.* (2019). Granular neural networks: A study of optimizing allocation of information granularity in input space. *Appl. Soft Comput.*, 77, 67-75. <https://doi.org/10.1016/j.asoc.2019.01.013>
- [7] Ming Li Song and Yukai Jing. Granular neural networks: The development of granular input spaces and parameters spaces through a hierarchical allocation of information granularity. *Inf. Sci.* (2020). Granular neural networks: The development of granular input spaces and parameters spaces through a hierarchical allocation of information granularity. *Inf. Sci.*, 517, 148-166. <https://doi.org/10.1016/j.ins.2019.12.081>
- [8] Abdullah Saeed Balamash and Witold Pedrycz and Rami Al-Hmouz and Ali Morfeq. Granular classifiers and their design through refinement of information granules. *Soft Computing*. (2015). Granular classifiers and their design through refinement of information granules. *Soft Computing*, 21(10), 2745 - 2759. <https://doi.org/10.1007/s00500-015-1978-9>
- [9] Akhouni, E. (2014). Optimal Allocation of Information Granularity to the Inputs of Granular Neural Networks. Master's dissertation, University of Alberta, <https://doi.org/10.7939/r3j960j49>
- [10] Rami Al-Hmouz and Witold Pedrycz and Abdullah Saeed Balamash and Ali Morfeq. Granular representation schemes of time series: A study in an optimal allocation of information granularity. (2013). Granular representation schemes of time series: A study in an optimal allocation of information granularity. *2013 IEEE Symposium on Foundations of Computational Intelligence (FOCI)*. <https://doi.org/10.1109/foci.2013.6602454>
- [11] Wei Lu and Cong Ma and Witold Pedrycz and Jianhua Yang. Design of Granular Model: A Method Driven by Hyper-Box Iteration Granulation. *IEEE Transactions on Cybernetics*. (2021). Design of Granular Model: A Method Driven by Hyper-Box Iteration Granulation. *IEEE Transactions on Cybernetics*, 53(5), 2899-2913. <https://doi.org/10.1109/tcyb.2021.3124235>
- [12] Arun Kumar, D., Meher, S., Kanhar, D. & Padma Kumari, K. Unified granular neural networks for pattern classification. *Neurocomputing*. **216** pp. 109-125 (2016), <https://doi.org/10.1016/j.neucom.2016.07.034>
- [13] Xiubin Zhu and Witold Pedrycz and Zhiwu Li. Development and Analysis of Neural Networks Realized in the Presence of Granular Data. *IEEE Transactions on Neural Networks and Learning Systems*. (2019). Development and Analysis of Neural Networks Realized in the Presence of Granular Data. *IEEE Transactions on Neural Networks and Learning Systems*, 31(9), 3606-3619. <https://doi.org/10.1109/tnnls.2019.2945307>
- [14] Ming Li Song and Yan Li and Witold Pedrycz. Time series prediction with granular neural networks. *Neurocomputing*. (2023). Time series prediction with granular neural networks. *Neurocomputing*, 546, 126328. <https://doi.org/10.1016/j.neucom.2023.126328>
- [15] Lihong Dong and Guanchen Li and Dan Wang and Zhenhua Yu and Emad Abouel Nasr and Abdelatty E Abdelgawad and Emad Fararah. Granular data modeling and analysis based on optimized subsets of data. *Measurement and Control*. (2022). Granular data modeling and analysis based on optimized subsets of data. *Measurement and Control*, 56(1-2), 34 - 48. <https://doi.org/10.1177/00202940221083263>
- [16] Loia, V., Parente, M., Pedrycz, W., & Tomasiello, S. (2018). A Granular Functional Network with delay: Some dynamical properties and application to the sign prediction in social networks. *Neurocomputing*, 321, 61-71. <https://api.semanticscholar.org/CorpusID:53047063>
- [17] Yapeng Liu and Ming Li Song. Few samples learning based on granular neural networks. *Granular Computing*. (2021). Few samples learning based on granular neural networks. *Granular Computing*, 7(3), 577 - 589. <https://doi.org/10.1007/s41066-021-00285-z>
- [18] Song, M. L., & Liu, Y. (2021). A development framework of granular prototypes with an allocation of information granularity. *Inf. Sci.*, 573, 154-170. <https://doi.org/10.1016/J.INS.2021.06.001>
- [19] Qin, J., Martinez, L., Pedrycz, W., Ma, X. Y., & Liang, Y. (2023). An overview of granular computing in decision-making: Extensions, applications, and challenges. *Inf. Fusion*, 98, 101833. <https://doi.org/10.1016/j.inffus.2023.101833>
- [20] Ming Li Song and Yongbin Wang. A study of granular computing in the agenda of growth of artificial neural networks. *Granular Computing*. (2016). A study of granular computing in the agenda of growth of artificial neural networks. *Granular Computing*, 1(4), 247 - 257. <https://doi.org/10.1007/s41066-016-0020-7>
- [21] Smarandache, & Florentin. (2006). A unifying field in Logics: Neutrosophic Logic. In *Neutrosophic Probability (fifth Edition)* (pp. 157). Zenodo. <https://doi.org/10.48550/arXiv.math/0101228>
- [22] Qasim, A. R., abbas, Sajjad, Jumaah, H. N., Hussein, Khalaf, M., Khalid, & E, H. (2025). A Neutrosophic Approach to Improving Sentiment Classification Accuracy in Social Media Analytics. *Zenodo*. <https://doi.org/10.5281/zenodo.15825643>
- [23] Marcheco, Chacon, E., Washington Eduardo, Y. P., Cueva Salazar, N. M., Toro Molina, B. M., Silva Deley, L. M., & Dekhkonov, B. (2025). A proposed SWOT analysis method for integrating indeterminate Likert scale with the neutrosophic AHP. *International Journal of Neutrosophic Science (IJNS)*, 25(1). <https://doi.org/10.54216/ijns.250142>
- [24] Smarandache, F. (2020). The score, accuracy, and certainty functions determine a total order on the set of neutro-

- sophic triplets (T, I, F). Infinite Study. <https://doi.org/10.48550/arXiv.2012.10208>
- [25] Khatter, Kiran. Interval valued trapezoidal neutrosophic set: multi-attribute decision making for prioritization of non-functional requirements. *Journal of Ambient Intelligence and Humanized Computing*. (2021). Interval valued trapezoidal neutrosophic set: multi-attribute decision making for prioritization of non-functional requirements. *Journal of Ambient Intelligence and Humanized Computing*, 12(1), 1039–1055. <https://doi.org/10.1007/s12652-020-02130-8>
- [26] Kennedy, J. & Eberhart, R. Particle swarm optimization. *Proceedings Of ICNN'95 - International Conference On Neural Networks*. 4 pp. 1942-1948 vol.4 (2002), <https://api.semanticscholar.org/CorpusID:3114196>
- [27] Witold Pedrycz and Wladyslaw Homenda. Building the fundamentals of granular computing: A principle of justifiable granularity. *Applied Soft Computing*. (2013). Building the fundamentals of granular computing: A principle of justifiable granularity. *Applied Soft Computing*, 13(10), 4209-4218. <https://doi.org/10.1016/j.asoc.2013.06.017>
- [28] Pedrycz, W. (2021). The Principle of Justifiable Granularity. Springer International Publishing. https://doi.org/10.1007/978-3-030-52800-3_10
- [29] Wolberg, W., Mangasarian, O., Street, N., & Street, W. (1993). Breast Cancer Wisconsin (Diagnostic). <https://doi.org/10.24432/c5dw2b>
- [30] Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2), 179-188. <https://doi.org/10.1111/j.1469-1809.1936.tb02137.x>
- [31] Nakai, K. & Kanehisa, M. Expert system for predicting protein localization sites in gram-negative bacteria. *Proteins: Structure, Function, And Bioinformatics*. 11, 95-110 (1991), <https://doi.org/10.1002/prot.340110203>.
- [32] MathWorks. (2017). Data Sets for Deep Learning - MATLAB and Simulink. <https://www.mathworks.com/help/deeplearning/ug/data-sets-for-deep-learning.html>
- [33] Janosi & Detrano, R. Heart Disease. (UCI Machine Learning Repository,1989), <https://doi.org/10.24432/c52p4x>
- [34] Thabtah, F. Autism Screening Adult. (UCI Machine Learning Repository,2017), <https://doi.org/10.24432/C5F019>
- [35] Narayan, & Nikhil. (2024). Weather Type Classification. *Kaggle.com*. <https://www.kaggle.com/datasets/nikhil7280/weather-type-classification>
- [36] Bhagat, & Atmaram, M. (2024). Crop Recommendation Dataset. *Kaggle.com*. <https://www.kaggle.com/datasets/madhuraatmarambhagat/crop-recommendation-dataset>
- [37] Nakai, K. Ecoli. (UCI Machine Learning Repository,1996), <https://doi.org/10.24432/C5388M>
- [38] Matiz, Sergio and Barner, Kenneth E. Inductive conformal predictor for convolutional neural networks: Applications to active learning for image classification. *Pattern Recognition*. (2019). Inductive conformal predictor for convolutional neural networks: Applications to active learning for image classification. *Pattern Recognition*, 90, 172–182. <https://doi.org/10.1016/j.patcog.2019.01.035>
- [39] Li, M., Klamkin, M., Tanneau, M., Zandehshahvar, R. & Van Hentenryck, P. Conformal Prediction with Upper and Lower Bound Models. *ArXiv Preprint ArXiv:2503.04071*. (2025), <https://doi.org/10.48550/arXiv.2503.04071>
- [40] Sadinle, Mauricio and Lei, Jing and Wasserman, Larry. Least ambiguous set-valued classifiers with bounded error levels. *Journal of the American Statistical Association*. (2019). Least ambiguous set-valued classifiers with bounded error levels. *Journal of the American Statistical Association*, 114(525), 223–234. <https://doi.org/10.1080/01621459.2017.1395341>