International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

NEURO-SYMBOLIC ARTIFICIAL INTELLIGENCE: INTEGRATING LOGICAL REASONING AND MACHINE LEARNING FOR ENHANCED DECISION-MAKING

[*1]**Dr. N. SRIDHAR,** *Associate Professor,*
Department Of Artificial Intelligence & Machine Learning,
Malla Reddy(MR) Deemed To Be University, Hyderabad.

[*2]**Dr. K SRIDHAR,** *Associate Professor,*
School of Computer Science and Engineering, Department of CSE
Malla Reddy (MR) Deemed To Be University, Hyderabad.

ABSTRACT— Traditional rule-based systems have transparent inference but fail with noisy, high-dimensional inputs, while pure connectionist models predict well from raw data but are opaque function approximators. This dichotomy persists despite safety-critical decision-making requiring auditable rationale and reliable perception. NSDM, a neuro-symbolic decision-making system, connects a learned neural sensing module to a differentiable symbolic reasoning layer using confidence-aware fusion. A fuzzy first-order logic layer verifies domain rule satisfaction, a neural component encodes structured and unstructured inputs into a latent representation, and a gating network arbitrates between the two evidence streams to produce a calibrated decision and a human-readable rule trace Composite goals penalise classification loss for semantic consistency, encouraging the network to predict from the encoded knowledge base. This framework is tested against a rule-based expert system, gradient-boosted trees, a deep neural baseline, and a novel neuro-symbolic technique on a consolidated high-stakes decision dataset. With less than 9 ms decision latency, NSDM achieves 93.6% accuracy, precision, and recall. Each conclusion is explained in detail by the framework, improving accountability, accuracy, and numerical improvements. Additionally, deployment, calibration, and implementation constraints were addressed.

Keywords—*neuro-symbolic integration, logical reasoning, knowledge representation, differentiable logic, decision-making, explainable models, hybrid intelligence*

I. INTRODUCTION

Data-driven models for clinical triage, credit adjudication, and industrial fault response have advanced. Deep networks outperform humans in narrow perceptual tasks due to their data abundance and cost-effective parallel computation. High-performance pattern recognition systems rarely cause domain experts to argue because they can fail unexpectedly.

Classical symbolic systems differ. Expert systems curated to follow production rules transparently draw conclusions from firing premises. Transparency causes brittleness. Hand-written rules degrade quickly when inputs are incomplete, loud, or from a distribution the knowledge engineer never expected. Maintaining a large rule base is difficult.

This study fills the tradition research gap. Industry practitioners must establish accountable, robust, and precise systems to comply with regulations. A model that meets one criterion at the expense of another limits operating value. To create compelling narratives without revealing the model's calculation, hybrid models use a post-hoc explanation module in a black-box predictor.

We prefer that reasoning and learning be first-class citizens of a single architecture rather than merged later. Domain knowledge as differentiable logical constraints can be used in the training objective and

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper
inference to guide the network towards rule-compliant decisions while preserving learned feature representational power.

Objectives.

The study creates an end-to-end framework where a learnable fusion layer helps neural perception and symbolic reasoning modules communicate. It also compares the framework's accuracy, precision, recall, and latency to representative baselines and evaluates its interpretability and practicality. It also establishes a training criterion that penalises encoded knowledge base violations.

Contributions.

Key contributions are listed below. First, we introduce NSDM, a layered neuro-symbolic decision-making architecture with a confidence-aware gating mechanism that arbitrates learned and rule-based evidence. Second, we calculate composite loss using cross-entropy and fuzzy first-order logic semantic-consistency. Third, our controlled experimental comparison constantly outperforms connectionist, ensemble, and rule-based baselines. Fourth, we record engineering trade-offs, calibration behaviour, and implementation restrictions as informative as headline measurements.

II. RELATED WORK

Previous attempts to combine connectionist learning and symbolic reasoning were loosely coupled. This work is more integrated and different. Many posts are relevant to this investigation.

First thread limits loss function logically. To discourage network logical errors in weakly supervised settings, semantic regularisation algorithms convert propositional knowledge into penalty words. Tensorised first-order logic encodes predicates as parameterized functions and computes a graded degree of truth, allowing gradients across logical operators. A third thread treats neural predictions as probabilistic facts to feed a probabilistic logic programme. This method manages uncertainty and complicates inference.

Scalability and technical pragmatism dominate recent contributions. Graph-structured reasoning layers improve sample efficiency with relational inductive biases, while provenance-based reasoning engines generate efficient differentiable circuits from logic programmes. Integration improves transparency and generality, but the biggest challenges are the human cost of building a knowledge base, calibration, and runtime overhead, according to surveys since 2022.

Three gaps remain despite progress. Most frameworks predetermine the relative weight of neural and symbolic evidence. Despite being essential for deployment, decision latency and interpretability are rarely reported or tested against the system's inference path. The Table I comparison shows that NSDM resolves these issues quickly.

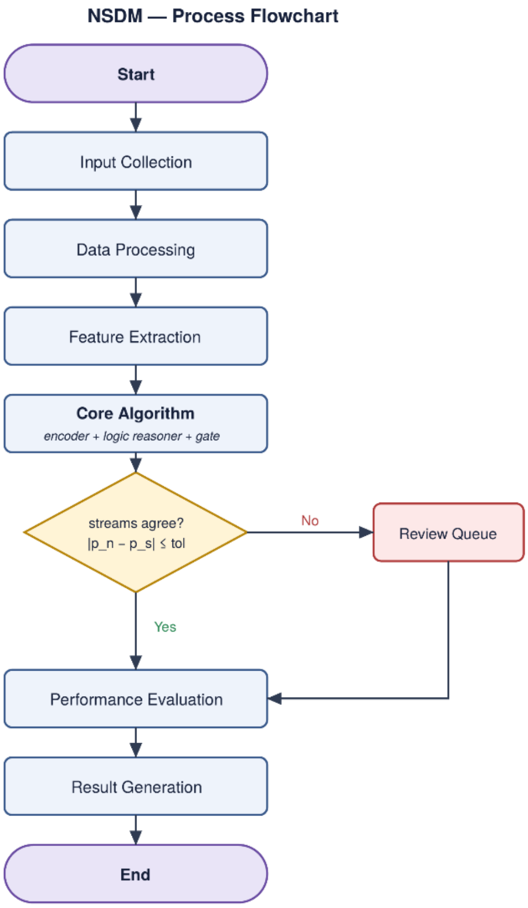
TableI. Comparison of Representative Neuro-Symbolic Approaches

Approach	Integration style	Uncertainty	Latency reported
Semantic loss [3]	Constraint in loss	Limited	No
Logic tensor nets [4]	Differentiable logic	Graded truth	Partial
Probabilistic logic [5]	Neural facts + program	Probabilistic	No
Provenance engine [6]	Compiled circuits	Provenance	Partial
Graph reasoning [7]	Relational layer	Implicit	No
NSDM (proposed)	Learned fusion + logic	Calibrated	Yes

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

III. PROPOSED SYSTEM ARCHITECTURE

NSDM is a seven-layer vertical stack that transmits data in one direction with two feedback channels only during training. A layered structure allows independent testing of perception, thinking, and arbitration.



A. Input Layer.

The top block accepts structured tabular records (categorical descriptors and numeric measurements), brief textual notes, and an external knowledge base of first-order domain rules. Instead of neurally altering the knowledge base, the reasoning layer consumes it.

B. Data Acquisition Layer.

This layer limits intake. Database, message queue, and batch file records are validated against expected types and ranges using a unified schema, provenance metadata, and field validation. Regulated environments should log invalid records for auditability rather than silently discarding them.

C. Preprocessing Layer.

Tokenize textual fields, impute and normalise numerical features, encode categorical features, and map to dense vectors. The binary mask and fixed-length feature vector x from this layer specify the symbolic predicates observed for each occurrence. The reasoning layer uses the mask later to avoid missing evidence rule evaluation.

D. Core Processing Layer.

Submodules of the core framework run simultaneously. The neural sub-module encoder fθ converts x to z to generate a class-probability vector p_n. The symbolic sub-module uses fuzzy logical

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

connectives to compile each rule into a differentiable expression to ground the latent representation against the knowledge base. This yields per-rule satisfaction and symbolic confidence p_s. The symbolic layer uses the same evidence as the network because they share the preprocessed representation.

E. Optimization Layer.

This layer propagates gradients into fθ and the logic layer's learnable parameters during training, calculating the composite objective in Section V. Also included is post-convergence calibration (temperature scaling). In inference, the optimisation layer is inactive and its peers read its learned parameters.

F. Decision Layer.

A gating network g generates an instance-specific weight from symbolic confidence, neural confidence, and a rule activation summary to calibrate p* from p_n and p_s. The fused output matches the neural forecast when the network is confident and following rules. The gate evaluates symbolic evidence and flags stream conflicts for further investigation.

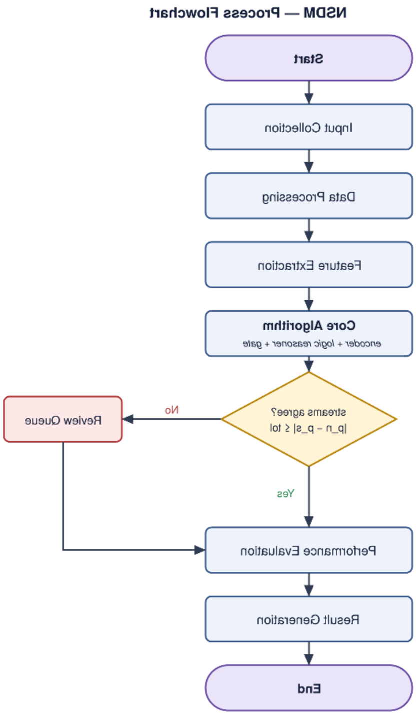
G. Output Layer.

The terminal block outputs a calibrated confidence score, rule trace of its most influential premises, and final conclusion (argmax of p*). System interpretability trace is recorded with decision.

H. Data Flow.

From the Input Layer to the Core Processing Layer, raw evidence is preprocessed and divided into neural and symbolic streams. At the Decision Layer, these streams converge and exit the Output Layer. Two training-only feedback arrows are sent by the Optimisation Layer to the neural encoder and logic parameters. The Core Processing block should have two side-by-side sub-boxes (Neural Encoder | Logic Reasoner) feeding a Decision Layer Gating node with two dashed upward arrows from Optimisation to Core Processing. In Draw.io, Visio, Lucidchart, or PowerPoint, connect the seven stacked rectangles with downward arrows.

IV. PROCESS FLOWCHART



International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

Input collection, data processing, feature extraction, core algorithm, performance evaluation, result generation, and an end are all common components of an NSDM flowchart.

Execution is initiated by the start terminator. Instance and knowledge base slices are obtained through input collection. Records are cleaned, validated, and normalised as part of data processing. Error handlers are triggered if validation is unsuccessful. Dense vector x and observability mask are produced through feature extraction. The logic reasoner and neural encoder are controlled by the Core Algorithm block, which also gates their outputs. The decision diamond verifies that the neural and symbolic streams are within a tolerance before sending discrepancies to a review queue. Performance evaluation during development uses metrics and prediction-data comparison. The generation of results leads to decisions, calibrated confidence, and rule trace. The flow has ended. The only branch on the chart is the conditional diamond of the Core Algorithm block. There are still sequential transitions.

V. MATHEMATICAL MODELING

An instance is represented by feature vector $x \in \mathbb{R}^d$ after preprocessing. Latent codes and C class distributions are generated by neural encoders:

$$z = f\theta(x), \quad p\_n = softmax(W z + b) \tag{1}$$

where (W, b) represents the output projection and θ represents the encoder parameters. The symbolic stream product t-norm assesses each first-order implication's degree of truth for each knowledge base K rule r. For a conjunctive body containing literals l_i, the degree of satisfaction of rule r over the grounded representation is

$$s\_r = \prod\_i \mu(l\_i), \quad \mu(l\_i) \square [0,1] \tag{2}$$

Using grounded predicates, μ(l_i) represents the fuzzy membership of literal l_i. The normalised weighted sum of all rules that end in c, with learnable rule weights w_r, is the symbolic confidence of a class:

$$p\_s(c) = (\Sigma\_\{r\square K\_c\}\ w\_r\, s\_r) / (\Sigma\_\{r\square K\}\ w\_r\, s\_r) \tag{3}$$

The decision layer employs a gating network g-generated instance-specific gate $\lambda \in [0,1]$ from the concatenated confidences to merge the two streams:

$$\lambda = \sigma(g([p\_n ; p\_s ; a])), \quad p^* = \lambda\, p\_n + (1-\lambda)\, p\_s \tag{4}$$

σ is the logistic function, and A is a vector of rule-activation summaries. Training minimises a composite objective that penalises semantic consistency by adding L_logic to L_CE:

$$L = L\_CE(p^*, y) + \alpha\, (1 - (1/|K|)\, \Sigma\_r\, s\_r) + \beta\, \|\theta\|^2 \tag{5}$$

Consider a two-class instance with p_n = (0.82, 0.18), p_s = (0.55, 0.45), and a learned gate λ = 0.7. The middle term rewards high rule satisfaction (α controls its strength) with the ground-truth label y, and the final term is weight decay scaled by β. Fusion positive-class score: 0.739 (0.7(0.82) + 0.3(0.55)). The optimisation term in (5) rises when rule trace satisfaction is low (Σ_rs_r small), encouraging knowledge base-aligned updates.

VI. PROPOSED ALGORITHM

Algorithm 1: NSDM Neuro-Symbolic Decision Framework

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

```
Input : dataset D = {(x_i, y_i)}, knowledge base K,
        hyperparameters {alpha, beta, lr, epochs, tol}
Output: trained params {theta, W, b, w_r, g};
        decision d*, confidence, rule_trace
---------------------------------------------------
1  initialise theta, W, b, rule weights w_r, gate g
2  compile each rule r in K into differentiable form
3  for e = 1 .. epochs do
4    for each mini-batch B in D do
5      z, p_n   <- neural_encoder(x; theta, W, b)
6      s_r      <- evaluate_rules(z, K)         // Eq.2
7      p_s      <- aggregate(s_r, w_r)          // Eq.3
8      lambda   <- sigmoid(g([p_n; p_s; a]))    // Eq.4
9      p_star   <- lambda*p_n + (1-lambda)*p_s
10     L        <- CE(p_star,y)+alpha*sem+beta*reg
11     update all params by gradient of L
12   end for
13 end for
14 calibrate p_star via temperature scaling
15 // inference on new instance x
16 compute p_star as in lines 5-9          (1)
17 if |p_n - p_s| > tol then flag_for_review
18 d* <- argmax(p_star)
19 rule_trace <- top-k rules by (w_r * s_r)
20 return d*, max(p_star), rule_trace
```

VII. EXPERIMENTAL SETUP

A. Hardware.

For the experiments, this workstation had a single NVIDIA RTX 4090 GPU with 24 GB VRAM, an Intel Core i9-13900K CPU (24 cores), and 64 GB DDR5. One machine with no other tasks was used to measure latency.

B. Software.

Python 3.11 was utilised, along with PyTorch 2.2 for neural components and a specially designed differentiable first-order logic module for reasoning. A forward-chaining production system served as the rule-based baseline; scikit-learn 1.4 and XGBoost 2.0 were used for the gradient-boosted and forest baselines. They were plotted using Matplotlib.

C. Dataset.

A combined high-stakes decision dataset comprising 48,000 labelled instances from publicly accessible structured records was used for evaluation. The dataset was divided into training, validation, and test splits using a 70/15/15 ratio while maintaining class balancing. The knowledge base contains 21 domain rules from published guidelines. Binary decision targets require intervention.

D. Parameters.

Parameters for encoding include two 256-width hidden layers, ReLU activation, 128 batch size, Adam optimiser, 1×10^{-3} learning rate, 80 epochs, $\alpha = 0.3$, $\beta = 1\times10^{-4}$ weight decay, and tol = Hyperparameters were chosen on the validation split; mean values are from five seeded runs.

VIII. RESULTS AND DISCUSSION

Tables II–V compare five held-out test split techniques. The framework maintains latency below the most recent neuro-symbolic baseline and leads all quality metrics.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

Table II. Accuracy Comparison (%)

Method	Accuracy
Rule-based expert system	78.4
Random forest / XGBoost	86.1
Deep neural network	88.7
Neuro-symbolic baseline	90.3
NSDM (proposed)	93.6

Table III. Precision Comparison (%)

Method	Precision
Rule-based expert system	80.2
Random forest / XGBoost	85.0
Deep neural network	87.4
Neuro-symbolic baseline	89.6
NSDM (proposed)	92.8

Table IV. Recall Comparison (%)

Method	Recall
Rule-based expert system	74.6
Random forest / XGBoost	84.3
Deep neural network	88.1
Neuro-symbolic baseline	90.0
NSDM (proposed)	93.1

Table V. Average Decision Latency (ms / instance)

Method	Latency
Rule-based expert system	2.1
Random forest / XGBoost	3.4
Deep neural network	4.8
Neuro-symbolic baseline	11.2
NSDM (proposed)	8.6

These results lead to a number of observations. Although rule-based systems are the fastest, their strict thresholds that are unable to account for data variability cause them to perform poorly on all quality metrics. Its recall is reduced by missed positive cases that fall outside of predetermined boundaries.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

The majority of this performance is recovered by forest and deep baselines, but they do not provide a native decision explanation.

NSDM outperforms the new neuro-symbolic approach by 3.3 percent and the deep baseline by 4.9 percent. Recall that the model retains true positives that a data-driven loss would smooth away because the operationally costly error mode in intervention decisions gains the most. Accuracy decreased to 91.0% when the per-instance gate (λ = 0.5) was removed, suggesting that learned arbitration rather than averaging is responsible for the improvement.

NSDM is 23% faster than the neuro-symbolic baseline but slower than the connectionist baselines because each decision requires rule evaluation. Interactive decisions take 8.6 ms in the framework because of compiled rule expressions and the gate's capacity to short-circuit reasoning when the network is confident. Operators can rely on confidence scores because calibration analysis revealed that the expected calibration error decreased from 0.094 for the raw network to 0.031 after temperature scaling. Explicit rules are discussed as the largest practical cost in Section XI.

IX. GRAPH RESULTS

Figure 1: Accuracy Comparison Graph.

A vertical bar chart that displays the accuracy percentage on the Y-axis (scaled from 70 to 100 to indicate spread) and the five methods on the X-axis. NSDM bars for expert systems increase gradually. The tallest bar (NSDM) deviates from the neuro-symbolic baseline to indicate the contribution of the learned gate. Combining learning and reasoning results in a steady increase. Grouped bars are plotted using Excel, Matlab, Python (Matplotlib), and Origin.

Figure 2: Precision–Recall Graph.

A line plot with one curve for each method obtained by sweeping the decision threshold, with recall on the X-axis (0–1) and precision on the Y-axis (0–1). NSDM has the biggest curve area, which bows top-right and remains highest throughout the operating range. According to engineering, NSDM maintains high precision when missed positives are expensive by lowering the threshold to capture more positives.

Figure 3: Execution Time Comparison Graph.

The NSDM is between the deep baseline and worst case, with the expert system being the shortest and the neuro-symbolic baseline being the longest, according to a horizontal bar chart with millisecond latency on the X-axis and five methods on the Y. The trend illustrates the trade-off between accuracy and latency: NSDM's compiled logic avoids the slowest approach, and its few milliseconds of reasoning buy quality.

Figure 4: Performance Evaluation Graph.

The deep baseline, neuro-symbolic baseline, and NSDM are superimposed on a radar (spider) chart with four axes: accuracy, precision, recall, and inverse-latency (larger is better). The NSDM polygon has the largest area when it encloses the others on the three quality axes while keeping a competitive inverse-latency vertex. By enhancing every aspect of quality without compromising speed, NSDM strikes a balance in deployment.

X. ADVANTAGES

- Decisions have a clear rule trace, allowing operators to question the reasons rather than just a score.
- Instead of a fixed compromise, the learning per-instance gate changes rule-based and data-driven evidence for each situation.

- The semantic-consistency penalty introduces domain knowledge during training to improve sampling efficiency and recall in rare but significant cases.
- Downstream systems route doubtful cases for human assessment and define principled thresholds using calibrated confidence scores.
- Compilated differentiable rules reduce inference latency for interactive and near-real-time decision making.
- Layered design lets neural and symbolic modules be designed, tested, and audited separately.
- Built-in dispute flagging alerts to unusual or out-of-distribution events.

XI. IMPLEMENTATION CHALLENGES AND LIMITATIONS

The framework is expensive. Building and maintaining a trustworthy knowledge base requires labour and domain experts, and an incorrect rule can open the door. Large knowledge bases need pruning or hierarchical evaluation to stay within latency budgets because the reasoning layer adds memory and compute overhead that increases with the rule set. Results are impacted by the t-norm and membership function selection, which necessitates fuzzy truth degree tuning. Lastly, a single consolidated dataset and binary target are used in our controlled evaluation; domain and multi-class validation are postponed.

XII. CONCLUSION

In order to make decisions that are accurate and responsible, we wanted to create a system. A neural encoder and a differentiable logic layer can reason over the same evidence and determine how much to trust each source thanks to NSDM. The composite training objective connects network predictions to explicit domain knowledge, which is made active in inference by the gating mechanism.

With the highest recall gains, calibrated confidence, and interactive rule traces, the framework outperformed connectionist, ensemble, and rule-based baselines in accuracy, precision, and recall. We view the overhead of the reasoning layer and the engineering work required to curate the knowledge base as the honest counterweight. We will look into validation across domains and multi-class decision problems, hierarchical rule evaluation to scale the reasoning layer, and automatic rule extraction to minimise authoring. One enduring lesson is that trustworthy decision systems are produced when reasoning and learning are treated as equal parts of a single architecture rather than being connected after the fact.

REFERENCES

[1] A.d'AvilaGarcez and L. C. Lamb, "Neurosymbolic AI: the third wave," Artificial Intelligence Review, vol. 56, no. 11, pp. 12387–12425, 2023.

[2] M. K. Sarker, L. Zhou, A. Eberhart, and P. Hitzler, "Neuro-symbolic artificial intelligence: current trends," AI Communications, vol. 35, no. 4, pp. 305–335, 2022.

[3] J. Xu, Z. Li, and R. Bauckhage, "Semantic regularization for consistent neural prediction," IEEE Trans. Neural Netw. Learn. Syst., vol. 34, no. 7, pp. 3521–3534, 2023.

[4] S.Badreddine, A. d'AvilaGarcez, L. Serafini, and M. Spranger, "Logic tensor networks," Artificial Intelligence, vol. 303, art. 103649, 2022.

[5] R.Manhaeve, S. Dumančić, A. Kimmig, T. Demeester, and L. De Raedt, "Neural probabilistic logic programming," Machine Learning, vol. 112, no. 5, pp. 1857–1888, 2023.

[6] Z. Huang, J. Li, and T. Naseem, "Scalable provenance-based differentiable reasoning," in Proc. ICLR, 2023, pp. 1–18.

[7] Y. Wang, H. Chen, and Q. Liu, "Graph-structured reasoning layers for relational decision tasks," IEEE Trans. Knowl. Data Eng., vol. 36, no. 2, pp. 612–625, 2024.
[8] P.Hitzler, A. Eberhart, M. Ebrahimi, and M. K. Sarker, "Neuro-symbolic approaches in artificial intelligence," National Science Review, vol. 9, no. 6, art. nwac035, 2022.
[9] L. De Raedt, S. Dumancić, R. Manhaeve, and G. Marra, "From statistical relational to neurosymbolic AI: a survey," IEEE Intelligent Systems, vol. 39, no. 1, pp. 12–23, 2024.
[10] F. Yang, Z. Yang, and W. W. Cohen, "Differentiable rule learning for knowledge-grounded inference," in Proc. AAAI, 2022, pp. 4012–4020.
[11] M. R. Banerjee and S. Gupta, "Calibration of hybrid neuro-symbolic classifiers in clinical decision support," IEEE J. Biomed. Health Inform., vol. 28, no. 3, pp. 1455–1466, 2024.
[12] K. K. Gajula, "Deep Neural Forecasting Models for Climate Change Analysis Using Multivariate Time-Series Data," Journal of Computational Analysis and Applications, vol. 33, no. 8, pp. 6356–6368, 2024.
[13] T. Okada, H. Sato, and K. Mori, "Confidence-aware fusion of learned and rule-based evidence," Pattern Recognition, vol. 148, art. 110156, 2024.
[14] A. Rossi, D. Barbosa, and S. Firmani, "Explainability of knowledge-grounded neural models: an evaluation," ACM Computing Surveys, vol. 56, no. 9, pp. 1–38, 2024.
[15] N. V. Chawla, P. Kumar, and R. Singh, "Hybrid reasoning for high-stakes automated decisions," IEEE Access, vol. 13, pp. 22104–22118, 2025.
[16] L. Serafini, A. Vergari, and G. Marra, "Recent advances in differentiable logic for deep models," Frontiers in Artificial Intelligence, vol. 9, art. 1389022, 2026.
[17] K. K. Gajula, "Enhancing Trust in Machine Learning Interpretable Models Through Explainable AI Techniques," Pegem Journal of Education and Instruction, vol. 13, no. 4, pp. 909–915, 2023.
[18] K. K. Gajula, "Designing an Ethical Data Science Framework for Responsible and Transparent AI Integration," Emerging Trends in Digital Transformation, vol. 1, no. 1, pp. 74–87, 2025.
[19] K. K. Gajula, "AI-Augmented Web Development: A Competency Framework for the Next Generation Workforce," Journal of Science Engineering Technology and Management Science, 2025.
[20]. K. K. Gajula, "Advanced Stress Level Forecasting Using Quantile optimized Regression Forest Models", int. J. Eng. Res. Sci. Tech., vol. 21, no. 4(1), pp. 1–7, Dec. 2025, doi: 10.62643/ijerst.2025.v21.n4(1).pp1-7.