FireGNN: Neuro-Symbolic Graph Neural Networks with Trainable Fuzzy Rules for Interpretable Medical Image Classification

Prajit Sengupta
BASIRA Lab
Department of Computing
Imperial College London
prajit.sengupta06@gmail.com

Islem Rekik*
BASIRA Lab
Department of Computing
Imperial College London
i.rekik@imperial.ac.uk

Abstract

Medical image classification requires not only high predictive performance but also interpretability to ensure clinical trust and adoption. Graph Neural Networks (GNNs) offer a powerful framework for modeling relational structures within datasets, however, standard GNNs often operate as black boxes, limiting transparency and usability particularly in clinical settings. In this work, we present an interpretable graph-based learning framework named **FireGNN** that integrates *trainable fuzzy rules into GNNs* for medical image classification. These rules embed topological descriptors - node degree, clustering coefficient, and label agreement - using learnable thresholds and sharpness parameters to enable intrinsic symbolic reasoning. Additionally, we explore *auxiliary self-supervised tasks* (e.g., homophily prediction, similarity entropy) as a benchmark to evaluate the contribution of topological learning. Our fuzzy-rule-enhanced model achieves strong performance across five MedMNIST benchmarks and the synthetic dataset MorphoMNIST, while also generating interpretable rule-based explanations. To our knowledge, this is the first integration of trainable fuzzy rules within a GNN.

1 Introduction

Medical image classification is fundamental to clinical decision-making and diagnostic workflows. While deep learning models, particularly Convolutional Neural Networks (CNNs), have demonstrated strong performance on medical imaging benchmarks, their lack of interpretability remains a key barrier to clinical adoption. In high-stakes domains like healthcare, it is crucial for models to not only make accurate predictions but also provide transparent, human-understandable reasoning behind those decisions [11, 18]. Graph Neural Networks (GNNs) have emerged as a compelling alternative by capturing topological and relational patterns across medical datasets [23]. By representing each image as a node and connecting similar samples via edges (e.g., using cosine similarity in feature space), GNNs can model global structure often overlooked by CNNs [1, 13]. However, standard GNN architectures such as Graph Convolutional Networks (GCNs) face major limitations [19]. **First**, GNNs often operate as black boxes, offering little insight into how predictions are made. For example, when a tumor patch is classified as malignant, it is unclear what node features or connections most influenced that decision - posing a major trust issue in clinical settings [3]. **Second**, many GNN pipelines use simplistic heuristics (e.g., spatial proximity) to construct graphs, especially

*Corresponding author: i.rekik@imperial.ac.uk, BASIRA Lab: https://basira-lab.com
Repository: https://github.com/basiralab/FireGNN

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: New Perspectives in Graph Machine Learning.

in histopathology [2, 26]. These hand-crafted rules may ignore biologically meaningful structures, leading to suboptimal or biased learning. **Third**, GNNs typically assume fixed graph topologies [4], making it difficult to adapt to new data without full retraining. This rigidity limits scalability in evolving clinical environments [11, 17]. These limitations motivate our proposed solution: **FireGNN** - a **Fuzzy Interpretable Rule Embedding for GNNs**. Unlike prior interpretability methods that work post-hoc, **FireGNN** integrates symbolic reasoning directly into the forward pass of a GNN. It does this by embedding interpretable fuzzy rules over key **node-level topological features** - specifically, the *node's degree*, *clustering coefficient*, and *2-hop label agreement*. Each fuzzy rule in FireGNN is parameterized by a trainable threshold and sharpness factor, enabling the model to define concepts like *"high connectivity"* or *"strong label consistency"* in a data-driven learnable way. These rules then *"fire"* with a certain strength, and are fused with learned graph embeddings through a gating mechanism. The result is a **transparent and expressive model** that can say, for example:

> *"This node is classified as a liver region due to its high degree and strong 2-hop label agreement with neighboring nodes."*

As an exploratory benchmark, we also evaluate whether **auxiliary self-supervised tasks**-such as prediction of local homophily and similarity entropy—can guide GNNs toward more topology-aware embeddings. However, we find that these tasks yield only modest gains compared to the symbolic reasoning introduced by fuzzy rules. FireGNN is evaluated on five MedMNIST datasets and one synthetic dataset (MorphoMNIST), where it achieves state-of-the-art accuracy while offering intrinsic interpretability. The key contributions of our paper can be summarized as follows:

1. *On a methodological level:* We present FireGNN that integrates trainable fuzzy rules over topological descriptors, enabling data-adaptive symbolic reasoning fused with learned embeddings for classification.

2. *On a clinical level:* Our intrinsically interpretable framework supports trustworthy medical AI, offering human-readable explanations (e.g., "high label agreement $\rightarrow$ liver node") alongside high performance, encouraging real-world deployment.

3. *On a generic level:* Our model generalizes across various datasets and GNN backbones *(GCN, GAT, GIN)*, and can be extended to any graph domain requiring explainable structural modeling. We also benchmarked against auxiliary tasks as an alternative way to encode topological structure but found limited improvement.

2 Related Work

Fuzzy logic in graph-based learning Fuzzy logic provides a framework for modeling uncertainty and interpretability using linguistic rules and soft membership functions. Classical systems like ANFIS [21, 10] combine fuzzy rule-based reasoning with trainable neural architectures. In the context of graphs, FGNN [22] leverages fuzzy inference for few-shot learning, and FL-GNN [8] integrates fuzzy systems into message-passing networks. Prior work such as [3] has explored combining fuzzy logic with GNNs to enhance model transparency. These methods have demonstrated the potential of fuzzy logic for enhancing interpretability, though they often rely on fixed rule templates.

Symbolic reasoning & structure-aware GNNs Incorporating symbolic rules into neural models has been explored as a path toward human-understandable AI [12]. However, many fuzzy and symbolic GNNs use pre-defined thresholds or rules that do not adapt to different datasets [20, 9]. Parallel to this, several studies have investigated topology-aware learning using auxiliary tasks or edge-type decoupling. [16] proposed using structural signals like homophily for improved node classification, while DuoGNN [15] decouples edges into homophilic vs. heterophilic sets and processes each through parallel GNN streams. Our model builds on this by embedding fuzzy logic over graphs.

Intrinsic vs post-hoc interpretability in GNNs Post-hoc explanation methods such as GNNExplainer [25] and PGExplainer aim to highlight important substructures or features after training. While useful, these approaches operate outside the model and may not faithfully reflect its internal reasoning [14]. Intrinsically interpretable models, by contrast, aim to embed explanation directly within the forward computation, offering more transparent and stable reasoning pipelines—especially critical in medical and safety-critical domains. Throughout this paper we are formulating **2 hypotheses**:

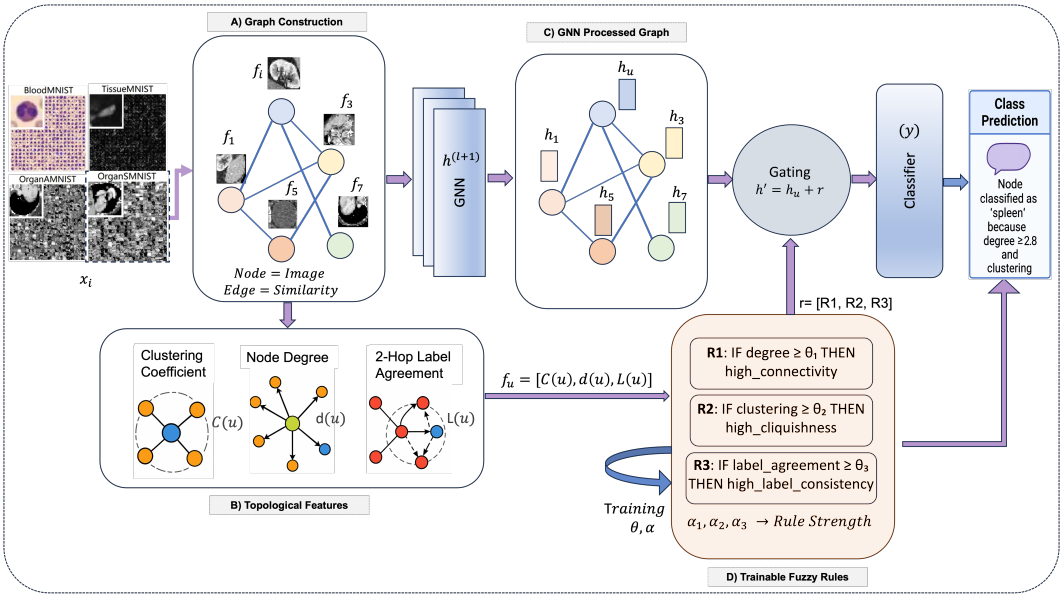

Figure 1: FireGNN architecture with four main components: (A) Graph construction, (B) Extraction of topological features, (C) GNN processing, and (D) Trainable fuzzy rules.

H1: Rule-based interpretability *Embedding fuzzy rules over node-level structural features—such as degree, clustering coefficient, and label agreement—enables the model to generate intrinsic, human-readable explanations.*

H2: Trainable symbolic reasoning *We hypothesize that learning fuzzy rule thresholds (θ) and sharpness parameters (α) from data allows the model to adapt boundaries to the specific graph structure, improving both generalization and semantic alignment.*

3 Methodology

Existing GNNs either ignore rich topological signals or treat them as fixed inputs, and prior fuzzy-GNN hybrids rely on hand-crafted thresholds.

Key challenge:

How can we learn *data-adaptive* symbolic conditions over graph-theoretic features (e.g., degree, clustering, etc) that both improve classification accuracy and yield *intrinsic interpretability*?

To this end, we propose **FireGNN**, a principled framework that (i) discovers, during training, the precise thresholds and sharpness for each rule per dataset, and (ii) integrates structural reasoning directly into the GNN's forward pass using symbolic rules over node degree, clustering coefficient, and 2-hop label agreement. An overview of the architecture as shown in **Fig. 1** is described below:

A. Graph construction We form a graph $G = (V, E)$ over all images: each node v_i has feature $f_i = F(x_i)$ and label y_i (Fig. 1-A). The adjacency matrix A is built via top-k cosine similarity. The base GCN processes the graph in the usual way as in (Fig. 1-C). Let $H^{(0)} = X$ be the initial node feature matrix. At each layer ℓ, we compute:

$$H^{(\ell+1)} = \sigma\left(\tilde{D}^{-1/2}\tilde{A}\tilde{D}^{-1/2}H^{(\ell)}W^{(\ell)}\right) \tag{1}$$

where $\tilde{A} = A + I$ (adjacency with self-loops), $\tilde{D}$ is the degree matrix of $\tilde{A}$, and $W^{(\ell)}$ are trainable weight matrices. After L layers, we obtain the final node embeddings h_v.

B. GNN with trainable fuzzy rules Our first model as shown in Fig. 1 augments the GNN with a neuro-symbolic fuzzy rule module. For each node u, we form a fact vector $f_u \in \mathbb{R}^3$ as in (Fig. 1-B)

containing its topological features:

$$f_u = [d(u),\, C(u),\, L(u)], \tag{2}$$

where $d(u)$ is the node degree, $C(u)$ is the clustering coefficient (the ratio of existing edges among v's neighbors to the maximum possible), and $L(u)$ is the 2-hop label agreement. We define three fuzzy rules corresponding to these features (Fig. 1-D). Each rule i has a learnable threshold θ_i and sharpness α_i, where θ_i controls the decision boundary and α_i modulates how sharply the rule activates near that threshold (i.e, how crisp or gradual the transition is). The activation of rule i on node u is computed:

$$r_i(u) = \sigma\left(\alpha_i(f_u[i] - \theta_i)\right) \in [0, 1] \tag{3}$$

where σ denotes the sigmoid function. Thus, $r_i(u)$ is close to 1 when the feature $f_u[i]$ significantly exceeds its threshold. Collecting all rule activations yields the firing strength vector: $r(u) = [r_1(u),\, r_2(u),\, r_3(u)] \in [0, 1]^3$. These activations serve as a symbolic explanation of the node's structural role. For instance, $r_1(u) \approx 1$ suggests "node has high degree". We fuse the fuzzy outputs with the GNN embedding via a gating mechanism. First, project the rule vector into the GNN embedding space:

$$e_u = W_r\, r(u) + b_r, \quad e_u \in \mathbb{R}^d \tag{4}$$

Then, concatenate $[h_u \parallel e_u] \in \mathbb{R}^{2d}$ and compute a gate vector:

$$g_u = \sigma(W_g[h_u \parallel e_u] + b_g) \in [0, 1]^d \tag{5}$$

The updated node embedding is given by:

$$h'_u = g_u \odot h_u + (1 - g_u) \odot e_u \tag{6}$$

where $\odot$ denotes elementwise multiplication. Intuitively, if rule-based information is highly relevant, g_u increases the influence of e_u; otherwise, it favors the original embedding h_u. The final embedding h'_u is passed to a linear classifier for label prediction. The entire model, including GCN weights, fuzzy rule parameters $\{\theta_i, \alpha_i\}$, and fusion weights (W_r, W_g), is trained end-to-end by minimizing the cross-entropy loss on labeled nodes. These rules offer interpretability: one can inspect which rules fired to explain a node's label prediction (e.g.,"high_connectivity" for bladder class).

C. GNN with auxiliary tasks as benchmark To assess alternative ways of injecting structural information into the GNN, we also experiment with two auxiliary prediction tasks which serves solely as a **benchmarking baseline** for comparison. Specifically, we supervise the model to estimate (1) *homophily* $h(v)$-the proportion of neighbors sharing the same label as node v, and (2) *similarity entropy* $S(v)$-a measure of uncertainty in edge-weight distributions over v's neighborhood. These quantities capture key graph properties: label agreement and connection sharpness, respectively. Detailed formulations, and the auxiliary-task training procedure are provided in **Appendix A.2**.

4 Results and Discussion

We conduct a comprehensive evaluation of our models on six inductive graph datasets as described in Table 2 (**Appendix A.1**): five derived from MedMNIST (OrganCMNIST, OrganAMNIST, OrganSMNIST, TissueMNIST, BloodMNIST) [24] and one synthetic benchmark (Morpho-MNIST) [5]. To ensure robust assessment, we perform 3-fold cross-validation with three random seeds (42,43,44), yielding a total of nine experiments per model variant. All results report mean±standard deviation across these runs. The complete end-to-end training procedure is as Algorithm 1 (**Appendix A.8**). All hyperparameters for the GNNs, fuzzy rule module, are detailed in **Appendix A.4** (Tables 4-7).

To enhance interpretability and transparency, we provide animated visualizations illustrating how fuzzy rule values evolve across epochs during training, along with a demo video showing the convergence of the learnable θ parameters for each dataset. These materials complement the reported quantitative results by offering an intuitive view of the model's self-reasoning and rule adaptation dynamics. Additionally, the repository includes all processed datasets and splits used in our experiments, enabling full reproducibility and ease of benchmarking. All supplementary videos, datasets, and resources are available at our project repository: `https://github.com/basiralab/FireGNN`.

Learned fuzzy rules Our model discovers dataset-driven fuzzy rule thresholds through training. On OrganCMNIST, the learned means (across 9 runs) are:

$$\theta_1 = 7.28, \quad \theta_2 = 0.18, \quad \theta_3 = 0.67.$$

Learned Fuzzy Rules (OrganCMNIST)

- **Rule 1:** IF degree ≥ 7.28 THEN high_connectivity
- **Rule 2:** IF clustering ≥ 0.18 THEN high_cliquishness
- **Rule 3:** IF 2-hop label agreement ≥ 0.67 THEN high_label_consistency

For example, a node predicted as *bladder* had a degree of 10, a clustering coefficient of 0.18, and a 2-hop label agreement of 0.73. Since:

- degree = 10 $\geq \theta_1 = 7.28 \Rightarrow$ Rule 1 activated (strength = 0.60),
- clustering = 0.18 = $\theta_2 \Rightarrow$ Rule 2 marginally activated (strength = 0.50),
- label agreement = 0.73 $\geq \theta_3 = 0.67 \Rightarrow$ Rule 3 strongly activated (strength = 0.56),

interprets this node as having high connectivity and strong label consistency, providing a human-readable justification for its *bladder* classification. This supports **H1** by offering intrinsic interpretability through explicit rule activations and **H2** by adapting thresholds to dataset-driven structural patterns.

Table 1: Comparison of GNN variants on six inductive datasets.

	OrganCMNIST				**OrganAMNIST**			
Method	ACC	F1	Sensitivity	ROC-AUC	ACC	F1	Sensitivity	ROC-AUC
GCN	88.20±0.61	86.03±0.89	86.13±0.06	99.09±0.08	91.85±0.30	91.19±0.33	91.18±0.03	99.51±0.03
GCN+ Aux	88.41±0.44	86.48±0.58	86.43±0.04	98.84±0.10	93.11±0.24	92.60±0.26	92.50±0.02	99.28±0.05
GCN+ FR	91.41±0.61	89.71±0.58	89.74±0.06	99.54±0.05	94.32±0.18	93.88±0.18	93.81±0.02	99.77±0.01
GAT	90.31±0.28	88.47±0.34	88.51±0.03	99.38±0.05	93.69±0.36	93.26±0.28	93.36±0.04	99.69±0.02
GAT+ Aux	90.88±0.49	89.07±0.59	88.93±0.62	99.45±0.08	93.70±0.46	93.36±0.44	93.33±0.04	99.54±0.02
GAT+ FR	91.66±0.48	90.02±0.52	90.12±0.05	99.56±0.05	94.52±0.31	94.08±0.30	94.13±0.03	99.78±0.02
GIN	87.96±0.59	85.61±0.75	85.56±0.75	98.86±0.09	91.54±0.71	90.45±0.73	90.50±0.69	99.41±0.08
GIN+ Aux	88.53±0.44	86.37±0.48	86.37±0.48	98.65±0.13	92.18±0.33	91.16±0.39	91.13±0.39	99.26±0.05
GIN+ FR	89.12±1.18	86.81±1.51	86.82±1.56	99.38±0.16	92.48±1.82	91.32±2.69	91.27±2.71	99.59±0.33

	OrganSMNIST				**TissueMNIST**			
Method	ACC	F1	Sensitivity	ROC-AUC	ACC	F1	Sensitivity	ROC-AUC
GCN	78.62±0.82	73.74±0.99	73.85±0.08	97.80±0.11	50.90±0.32	32.61±0.79	32.51±0.07	81.98±0.31
GCN+ Aux	79.19±0.74	74.21±0.84	74.28±0.07	97.56±0.14	52.70±0.22	35.67±0.36	35.60±0.04	82.99±0.19
GCN+ FR	85.05±0.43	80.56±0.61	80.74±0.04	98.95±0.05	65.73±0.88	51.63±1.22	51.59±0.09	93.56±0.36
GAT	81.80±0.68	77.22±0.73	77.16±0.07	98.39±0.09	51.53±0.35	33.10±0.56	33.06±0.07	83.11±0.12
GAT+Aux	81.69±0.68	77.33±0.73	77.06±0.07	98.28±0.09	OOM	OOM	OOM	OOM
GAT+FR	84.82±0.52	80.23±0.80	80.43±0.05	98.93±0.07	OOM	OOM	OOM	OOM
GIN	77.23±0.62	71.65±0.65	71.77±0.76	97.36±0.10	50.51±1.09	30.31±3.70	32.50±2.05	81.72±0.85
GIN+Aux	79.26±1.02	73.73±1.97	74.29±1.23	97.44±0.11	51.31±1.42	31.12±3.54	33.20±1.70	82.42±0.27
GIN+ FR	81.46±3.14	76.69±3.85	76.81±3.84	98.51±0.52	64.07±2.44	48.59±3.93	48.59±3.96	92.84±1.25

	BloodMNIST				**MorphoMNIST**			
Method	ACC	F1	Sensitivity	ROC-AUC	ACC	F1	Sensitivity	ROC-AUC
GCN	80.49±0.66	77.46±0.96	77.15±0.09	96.56±0.18	90.89±0.05	90.82±0.02	90.73±0.02	98.88±0.03
GCN+ Aux	80.17±0.66	77.07±0.84	76.96±0.10	96.05±0.13	92.84±0.10	92.83±0.10	92.78±0.03	99.00±0.04
GCN+ FR	88.31±0.37	86.36±0.45	86.22±0.04	99.15±0.05	94.76±1.02	94.76±1.03	94.58±0.34	99.63±0.25
GAT	82.02±0.31	79.38±0.40	79.18±0.04	97.45±0.08	91.50±0.15	91.48±0.16	91.33±0.05	98.73±0.04
GAT+ Aux	81.87±2.08	79.70±2.21	79.73±0.23	97.13±0.46	OOM	OOM	OOM	OOM
GAT+ FR	87.79±2.08	85.86±2.21	85.80±0.23	99.01±0.46	OOM	OOM	OOM	OOM
GIN	80.30±0.60	77.21±0.91	76.44±0.95	96.58±0.21	91.60±0.19	91.59±0.18	91.60±0.19	98.75±0.05
GIN+ Aux	80.47±0.20	77.13±1.31	76.83±0.38	96.30±0.42	92.30±0.24	92.11±0.32	92.50±0.43	99.25±0.12
GIN+ FR	84.83±3.04	82.41±3.60	82.39±3.81	98.36±0.94	93.72±1.55	93.73±1.54	93.72±1.55	99.55±0.22

Notes: **Aux** refers to Auxiliary Tasks; **FR** denotes Fuzzy Rules; Best Results in **Green**.

Classification performance Table 1 details accuracy (ACC), macro F_1, sensitivity, and ROC-AUC for each model. On OrganCMNIST, GCN+Fuzzy attains **91.41% ±0.61** ACC (+3.21% over GCN) and 89.71±0.58 F_1. GCN+Aux achieves a modest accuracy of 88.41±0.44, which even **sometimes**

decreases or remains unchanged, indicating that *auxiliary tasks alone yield only incremental improvements*. This supports our hypothesis that symbolic rule-based modeling is a more effective approach. GAT+Fuzzy further increases OrganCMNIST ACC to 91.66±0.48, confirming the generality of fuzzy reasoning across backbones. On low-homophily TissueMNIST, GCN+Fuzzy improves ACC by 14.83% (65.73±0.88 vs 50.90 ±0.32), underscoring its robustness where standard GCNs falter. Similarly, GCN+Aux enhances performance (52.50±0.26 ACC). We also tested the fuzzy rule methodology on Morpho-MNIST, a synthetic dataset, designed to evaluate representation learning. Morpho-MNIST results similarly favor GCN+ Fuzzy (94.76 ± 1.02 ACC), demonstrating generalizability. The **computational efficiency & dynamics** are described in Table 3(**Appendix A3**).

Conclusion We presented **FireGNN**, a simple yet effective framework that integrates trainable fuzzy rules for interpretable and accurate classification (Table 1). We also evaluated auxiliary structural tasks as a benchmarking tool for topology encoding. While our fuzzy rules capture key topological cues, future extensions could explore learning richer or more complex rule types. Expanding beyond the three structural features we use and applying this approach to dynamic or heterogeneous graphs offers promising directions for broader and more flexible interpretability **[see Appendix A.6 (Table 9) for learned rule thresholds, A.7 for a detailed exploration of other theoretical rules]**. Additionally, improving the efficiency of FireGNN particularly reducing its average epoch time and peak memory usage than the current values reported in Table 3 remains an important step toward sustainable model.

References

[1] Diego Ahmedt-Aristizabal, M. Armin Armin, Simon Denman, Clinton Fookes, and Lars Petersson. Graph-based deep learning for medical diagnosis and analysis: Past, present and future. *Sensors*, 21(14):4758, July 2021.

[2] Siemen Brussee, Giorgio Buzzanca, Anne M.R. Schrader, and Jesper Kers. Graph neural networks in histopathology: Emerging trends and future directions. *Medical Image Analysis*, 101:103444, 2025.

[3] Giovanna Castellano, Raffaele Scaringi, Gennaro Vessio, and Gianluca Zaza. Integrating graph neural networks and fuzzy logic to enhance deep learning interpretability. 11 2024.

[4] Deli Chen, Yankai Lin, Wei Li, Peng Li, Jie Zhou, and Xu Sun. Measuring and relieving the over-smoothing problem for graph neural networks from the topological view. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34:3438–3445, 04 2020.

[5] Daniel Coelho de Castro, Jeremy Tan, Bernhard Kainz, and Ben Glocker. Morpho-mnist: Quantitative assessment and diagnostics for representation learning. *Journal of Machine Learning Research*, 20, 10 2019.

[6] Thomas Dash, Ashwin Srinivasan, and Lovekesh Vig. Incorporating symbolic domain knowledge into graph neural networks. *Machine Learning*, 110(7):1609–1636, July 2021.

[7] Tirtharaj Dash, Sharad Chitlangia, Aditya Ahuja, and Ashwin Srinivasan. Incorporating domain knowledge into deep neural networks. *CoRR*, abs/2103.00180, 2021.

[8] Boyu Du, Jingya Zhou, Ling Liu, and Xiaolong She. *FL-GNN: Efficient Fusion of Fuzzy Neural Network and Graph Neural Network*. 10 2024.

[9] Samuel H. Huang and Hao Xing. Extract intelligible and concise fuzzy rules from neural networks. *Fuzzy Sets and Systems*, 132(2):233–243, 2002.

[10] J.-S.R. Jang. Anfis: adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man, and Cybernetics*, 23(3):665–685, 1993.

[11] Ruth Johnson, Michelle M. Li, Ayush Noori, Owen Queen, and Marinka Zitnik. Graph artificial intelligence in medicine. *Annual Review of Biomedical Data Science*, 7(Volume 7, 2024):345–368, 2024.

[12] Luís C. Lamb, Artur d'Avila Garcez, Marco Gori, Marcelo O.R. Prates, Pedro H.C. Avelar, and Moshe Y. Vardi. Graph neural networks meet neural-symbolic computing: a survey and perspective. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, IJCAI'20, 2021.

[13] Xiaoxiao Li, Yuan Zhou, Siyuan Gao, Nicha Dvornek, Muhan Zhang, Juntang Zhuang, Shi Gu, Dustin Scheinost, Lawrence Staib, Pamela Ventola, and James Duncan. Braingnn: Interpretable brain graph neural network for fmri analysis. *bioRxiv*, 2020.

[14] Ninghao Liu, Qizhang Feng, and Xia Hu. *Interpretability in Graph Neural Networks*, pages 121–147. Springer Nature Singapore, Singapore, 2022.

[15] Kevin Mancini and Islem Rekik. Duognn: Topology-aware graph neural network with homophily & heterophily interaction-decoupling. In *Graphs in Biomedical Image Analysis: 6th International Workshop, GRAIL 2024, Held in Conjunction with MICCAI 2024, Marrakesh, Morocco, October 6, 2024, Proceedings*, pages 129–140, Berlin, Heidelberg, 2025. Springer-Verlag.

[16] Franco Manessi and Alessandro Rozza. Graph-based neural network models with multiple self-supervised auxiliary tasks. *Pattern Recognition Letters*, 148:15–21, 08 2021.

[17] Zofia Rudnicka, Janusz Szczepanski, and Agnieszka Pregowska. Artificial intelligence-based algorithms in medical image scan segmentation and intelligent visual content generation—a concise overview. *Electronics*, 13(4), 2024.

[18] Zohaib Salahuddin, Henry C. Woodruff, Avishek Chatterjee, and Philippe Lambin. Transparency of deep neural networks for medical image analysis: A review of interpretability methods. *Computers in Biology and Medicine*, 140:105111, 2022.

[19] Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation, 11 2018.

[20] L. Wang and Jerry Mendel. Generating fuzzy rules by learning from examples. *Systems, Man and Cybernetics, IEEE Transactions on*, 22:1414 – 1427, 12 1992.

[21] Paul P. Wang, Da Ruan, and Etienne E. Kerre, editors. *Fuzzy Logic: A Spectrum of Theoretical & Practical Issues*, volume 216 of *Studies in Fuzziness and Soft Computing*. Springer-Verlag Berlin Heidelberg, Berlin, Heidelberg, 1 edition, 2007.

[22] Tong Wei, Junlin Hou, and Rui Feng. Fuzzy graph neural network for few-shot learning. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2020.

[23] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations (ICLR)*, 2019.

[24] Jiancheng Yang, Rui Shi, Donglai Wei, Xiangyu Li, Lingxi Xie, Xin Jin, Yuyin Zhou, Zhiwei Xie, Zequn Jie, Xingyu Li, Chi Harold Liu, Yutong Bai, Lin Gu, Shun Miao, and Alan Yuille. Medmnist v2 - a large-scale lightweight benchmark for 2d and 3d biomedical image classification. *Scientific Data*, 10(1):41, January 2023.

[25] Rex Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. Gnn explainer: A tool for post-hoc explanation of graph neural networks, 03 2019.

[26] Jiaxuan You, Rex Ying, and Jure Leskovec. Design space for graph neural networks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc.

A Technical Appendices and Supplementary Material

A.1 Dataset Overview

The datasets used in our experiments span both real biomedical image collections **MedMNIST v2** (MedMNIST v2 is available under the CC BY 4.0 license) which is a large-scale collection of lightweight 2D biomedical image datasets covering diverse organ types and a synthetic benchmark **Morpho-MNIST** (Morpho-MNIST is derived from MNIST, which is available under the CC BY-SA 3.0 license). Each dataset is converted into a graph where images serve as nodes and edges are formed via top-k cosine similarity in the feature space. This allows us to model global topological relationships between samples, which are often overlooked in purely CNN-based approaches.

Table 2: Graph dataset details created from images

	OrganCMNIST	**OrganAMNIST**	**OrganSMNIST**	**TissueMNIST**	**BloodMNIST**	**Morpho-MNIST**
# of Nodes	23583	58830	25211	236386	17092	280000
# of Edges	82315	205404	88951	826714	60116	970699
# of Features	512	512	512	512	512	512
# of Labels	11	11	11	8	8	4
Task Type	Multi-class	Multi-class	Multi-class	Multi-class	Multi-class	Multi-class
Training Type	Inductive	Inductive	Inductive	Inductive	Inductive	Inductive
Training Nodes	12975	34561	13932	165466	11959	216000
Validation Nodes	2392	6491	2452	23640	1712	24000
Test Nodes	8216	17778	8827	47280	3421	40000

As summarized in Table 2, dataset sizes vary substantially: BloodMNIST contains around 17K nodes, while TissueMNIST scales to over 230K nodes with nearly 830K edges. This diversity tests FireGNN's scalability. Morpho-MNIST, while synthetic, is deliberately challenging because its classes differ by morphological attributes (e.g., thickness - thick, thin; fragmentation - broken), contains 280K nodes, 970K edges, and 4 classes, requiring the model to rely on structural patterns rather than simple pixel statistics.

A.2 GNN with Auxiliary Tasks Mathematical Formalization

The purpose of these auxiliary tasks is to explicitly encourage the model to be aware of graph topology during training, beyond the main classification objective. The homophily term $h(v)$ captures whether a node shares labels with its neighbors, which reflects the level of semantic alignment in the graph. The similarity entropy $S(v)$, by contrast, quantifies the sharpness of a node's neighborhood distribution, distinguishing between nodes with a few strong connections versus many diffuse ones.

Given a node v, we define homophily as $h(v) = \frac{1}{|N(v)|} \sum_{u \in N(v)} \mathbb{I}(y_u = y_v)$, where $N(v)$ is the set of neighbors of v, and y_u denotes the label of node u. For similarity entropy, edge weights are computed as

$$w_{vu} = \frac{\exp(-\|x_v - x_u\|^2)}{\sum_{k \in N(v)} \exp(-\|x_v - x_k\|^2)}$$

followed by entropy computation:

$$S(v) = -\sum_{u \in N(v)} w_{vu} \log w_{vu}$$

Two lightweight MLP heads are added to the GNN to predict $\hat{h}(v)$ and $\hat{S}(v)$ from h_v, optimized via mean squared error:

$$\mathcal{L}_{\text{aux}} = \frac{1}{|\mathcal{V}_L|} \sum_{v \in \mathcal{V}_L} \left((\hat{h}(v) - h(v))^2 + (\hat{S}(v) - S(v))^2 \right) \tag{7}$$

where $\mathcal{V}_L$ denotes labeled nodes. Auxiliary loss is added to the main classification loss:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{cls}} + \lambda \mathcal{L}_{\text{aux}} \tag{8}$$

where λ is a tunable balancing hyperparameter. At inference time, the auxiliary heads are discarded, ensuring no additional computational overhead. Aux tasks regularize embeddings but do not yield explicit symbolic reasoning, limiting interpretability and performance, which FireGNN addresses easily.

A.3 Computational efficiency & learning dynamics

Table 3 highlights that fuzzy rule models like GCN+Fuzzy and GAT+Fuzzy exhibit higher computational costs, with epoch times of 17.96s and 44.50s, respectively, due to fuzzy logic computations, yet maintain modest memory usage (e.g., 519.28MB for GCN+Fuzzy). Experiments ran on an Apple M2 Air with 16GB RAM and MPS acceleration GPU, using an 80/20 train-validation split in a 3-fold cross-validation setup. Attention-based models like GAT+ Fuzzy occasionally hit out-of-memory (OOM) on large scale datasets like TissueMNIST and MorphoMNIST, reflecting the quadratic complexity of attention mechanisms.

Table 3: Comparison of **Average Epoch Time (s)** and **Peak Memory (MB)** across models and datasets.

Method	OrganCMNIST		OrganAMNIST		OrganSMNIST		TissueMNIST		BloodMNIST		MoprhoMNIST	
	Time	Memory	Time	Memory	Time	Memory	Time	Memory	Time	Memory	Time	Memory
GCN	0.48	758.95	1.15	1234.64	0.49	1024.92	2.76	2472.47	0.35	557.36	4.20	2648.61
GCN + Aux	1.90	944.25	1.78	1439.06	1.25	889.30	4.76	1781.39	0.35	718.44	4.71	1717.73
GCN + FR	**3.12**	519.28	**12.50**	1197.50	3.47	592.38	**17.96**	1035.08	**1.92**	547.02	4.86	1628.05
GAT	0.48	1895.61	1.49	1698.78	0.61	1486.59	**5.95**	1950.41	0.47	1576.44	3.79	2130.48
GAT+Aux	2.33	1486.59	2.89	1516.97	2.31	1425.03	OOM	OOM	1.55	1191.38	OOM	OOM
GAT + FR	**44.50**	827.34	**19.75**	864.91	**16.13**	803.56	OOM	OOM	**3.85**	1196.73	OOM	OOM
GIN	0.32	913.77	1.77	1293.22	2.26	801.11	3.22	1255.64	0.56	649.13	**5.49**	1526.40
GIN+ Aux	0.68	1004.95	1.98	1258.81	**4.36**	1022.61	3.87	1368.44	1.36	893.95	4.88	1480.20
GIN + FR	1.10	1221.91	3.31	1384.21	3.96	729.97	4.96	734.70	1.89	656.72	**16.61**	1660.84

Notes: Top two highest epoch times of each dataset are bolded.

A.4 Hyperparameters for FireGNN Model & Baselines

Core GNN Architecture. Table 4 lists the backbone hyperparameters for GCN, GAT, and GIN. These settings ensure a fair comparison across models while keeping complexity manageable.

Table 4: Core GNN architecture hyperparameters for FireGNN across different backbones.

Model	Parameter	Value
All (GCN/GAT/GIN)	Number of Layers (`num_layers`)	2
All (GCN/GAT/GIN)	Dropout Rate (`dropout`)	0.5
GAT	Attention Heads (`heads`)	8
GAT	Negative Slope (`negative_slope`)	0.2
GIN (MLP)	MLP Layers (`num_layers`)	2

Fuzzy Rule Layer. Table 5 specifies the design of the fuzzy rule module, including learnable parameters for rule centers, widths, and weights.

Table 5: Fuzzy rule layer hyperparameters in FireGNN.

Parameter	Value
Number of Rules (`num_rules`)	3
Topological Features (`num_features`)	3
Rule Centers (`centers`)	Learnable (initialized with `torch.randn`)
Rule Widths (`log_sigmas`)	Learnable (initialized with `torch.zeros`)
Rule Weights (`rule_weights`)	Learnable (initialized with `torch.ones`)

Graph Construction. Table 6 reports the hyperparameters used to construct the graphs from image features, including nearest-neighbor size, similarity metric, and feature extractor.

Table 6: Graph construction hyperparameters for FireGNN.

Parameter	Value
k Neighbors (`k`)	10
Distance Metric (`metric`)	cosine
Add Label Edges (`add_label_edges`)	True
Rewire Edges (`rewire_edges`)	True
Feature Extraction Batch Size (`batch_size`)	64
Feature Extraction Model	ResNet18 (pretrained)
Image Resize	(224, 224)

Training Setup. Table 7 summarizes the training hyperparameters, including number of epochs and cross-validation folds.

Table 7: Training hyperparameters for FireGNN experiments.

Parameter	Value
Epochs (`epochs`)	200
Cross-Validation Folds (`n_folds`)	3

A.5 Proposed FireGNN vs Limitations

Table 8: Limitations of standard GNNs and their solutions via the proposed model.

Limitations of Standard GNNs	Solution via Proposed Model
Black-box, non-interpretable decisions. [3]	Augment with fuzzy-rule layer that yields explicit logical conditions (e.g., "degree $\geq \theta \Rightarrow$ high connectivity"). Each rule's activation can be examined to explain predictions.
Fuzzy or interpretable rules are typically **fixed, hard-coded**, not data-adaptive. (e.g., degree $>$ k = high connectivity) [20, 9]	Our fuzzy rules are fully trainable: thresholds θ & sharpness α are optimized during training, allowing the model to discover boundaries (e.g., what node degree is 'high connectivity') for each dataset.
Cannot incorporate domain knowledge (e.g., **graph-theoretic rules**) into learning. [6, 7]	Fuzzy rules allow injecting human insight (e.g., "high clustering $\Leftrightarrow$ strong community"), bridging symbolic knowledge & GNNs. We blend neural features with rule-based signals.

A.6 Extended Showcase of Learned Fuzzy Rules

To further substantiate the claim that FireGNN learns data-adaptive symbolic conditions (Hypothesis H2), this section presents the fuzzy rule thresholds learned on two additional datasets: BloodMNIST and MorphoMNIST. These datasets possess distinct structural properties compared to OrganCMNIST, and as shown in Table 9, the model discovers unique rule boundaries for each, demonstrating its flexibility.

Table 9: Comparison of learned mean fuzzy rule thresholds (θ_i) across datasets. The values are learned during training and reflect the distinct topological characteristics of each graph.

Dataset	θ_1 (Degree)	θ_2 (Clustering)	θ_3 (Label Agreement)
OrganCMNIST	7.28	0.18	0.67
BloodMNIST	4.15	0.25	0.81
MorphoMNIST	11.52	0.09	0.92

Learned Rules for BloodMNIST BloodMNIST is a smaller graph with 8 classes. The model learns thresholds that reflect a potentially denser, more homophilic local structure compared to OrganCMNIST.

- **Rule 1:** IF degree ≥ 4.15 THEN `high_connectivity`
- **Rule 2:** IF clustering ≥ 0.25 THEN `high_cliquishness`
- **Rule 3:** IF 2-hop label agreement ≥ 0.81 THEN `high_label_consistency`

Example Interpretation: Consider a node representing a basophil cell image with a degree of 6, a clustering coefficient of 0.30, and a label agreement of 0.85. The model would reason:

- degree $= 6 \geq \theta_1 = 4.15 \Rightarrow$ Rule 1 strongly activated.
- clustering $= 0.30 \geq \theta_2 = 0.25 \Rightarrow$ Rule 2 activated.
- label agreement $= 0.85 \geq \theta_3 = 0.81 \Rightarrow$ Rule 3 activated.

The explanation would be that the node is classified as a basophil due to its high connectivity, cliquish structure, and very strong label consistency within its neighborhood.

Similarly, a *lung-left* node in OrganCMNIST datatset with degree 3, clustering 0.00, and label agreement 0.94 triggers:

- Rule 1: not activated ($3 \leq 7.28$),
- Rule 2: not activated ($0.00 \leq 0.18$),
- Rule 3: strongly activated ($0.94 \geq 0.67$, strength = 0.74).

This rule activation pattern reflects the node's sparse connectivity but high semantic consistency, guiding the model's prediction toward *lung-left*.

Learned Rules for MorphoMNIST MorphoMNIST is a large-scale synthetic graph where connections are based on morphological similarity. The model learns a high threshold for degree and label agreement, suggesting that for a node to be considered structurally significant, it must be exceptionally well-connected and consistent.

- **Rule 1:** IF degree ≥ 11.52 THEN `high_connectivity`
- **Rule 2:** IF clustering ≥ 0.09 THEN `high_cliquishness`
- **Rule 3:** IF 2-hop label agreement ≥ 0.92 THEN `high_label_consistency`

Example Interpretation: A node representing a "thick" digit with a degree of 15 and label agreement of 0.95 would strongly activate Rules 1 and 3. This provides a clear justification: the node's classification is driven by its status as a highly connected hub with near-perfect label consistency, a hallmark of a prototypical member of its morphological class.

A.7 Alternative Graph-Theoretic Rules for FireGNN

The current implementation of FireGNN leverages three fundamental and intuitive topological features: degree, clustering coefficient, and label agreement. These features primarily capture the *local* structure and semantic consistency of a node's immediate neighborhood. However, the FireGNN framework is extensible and can be enriched by incorporating a wider vocabulary of fuzzy rules derived from more complex graph-theoretic concepts. Exploring rules based on meso-scale and global network properties could enable the model to generate more sophisticated and nuanced explanations.

1. Rules from Node Centrality Measures While degree centrality is a local measure, other centrality metrics quantify a node's importance within the global network topology, capturing its role in information flow. Integrating these could provide explanations related to a node's structural influence.

- **Betweenness Centrality:** This measures how often a node lies on the shortest path between other nodes. A high betweenness centrality indicates a "bridge" or "bottleneck" node. In a medical context, such a node could represent a critical transitional state, such as a cell at the boundary of a tumor or a patient whose features bridge two distinct clinical subtypes.
  - *Proposed Rule:* `IF betweenness_centrality(u)` $\geq \theta_{bc}$ `THEN is_bridge_node`
- **Eigenvector Centrality:** This measures a node's influence based on the idea that connections to other highly important nodes contribute more than connections to peripheral nodes. In a graph of medical images, nodes with high eigenvector centrality could represent the most prototypical exemplars of a class or key hubs in a disease progression network.
  - *Proposed Rule:* `IF eigenvector_centrality(u)` $\geq \theta_{ec}$ `THEN is_influential_hub`

2. Rules from Community Structure Community detection algorithms partition a graph into densely connected subgraphs, revealing its meso-scale organization. In medical imaging graphs, communities might correspond to distinct tissue types, organ regions, or patient cohorts with similar characteristics. Rules based on a node's role within this community structure can offer powerful, context-aware explanations.

- **Participation Coefficient:** This metric quantifies how a node's edges are distributed among different communities. A node with a high participation coefficient is a "connector" hub, linking multiple communities. In a histopathology graph, this could identify a cell at the interface of stromal, epithelial, and immune cell communities - a location of significant biological interaction.
  - *Proposed Rule:* `IF participation_coefficient(u)` $\geq \theta_{pc}$ `THEN is_connector_hub`
- **Within-Module Degree Z-Score:** This measures how well-connected a node is to other nodes *within its own community*. A node with a high z-score is a "provincial" hub, central to its local community but not necessarily to the entire network. This could identify a core, representative component of a specific tissue type.

– *Proposed Rule:* `IF within_module_degree_zscore(u)` $\geq \theta_z$ `THEN provincial_hub`

3. Rules from Path-Based Metrics Metrics based on shortest paths can describe a node's integration and accessibility within the network, reflecting how efficiently it can interact with all other nodes.

- **Closeness Centrality:** This is the reciprocal of the average shortest path distance from a node to all other nodes in the graph. A node with high closeness centrality is "centrally accessible" and can propagate information efficiently throughout the network. Such nodes might represent the most "average" or canonical examples of a class, making them stable anchors for classification.
  – *Proposed Rule:* `IF closeness_centrality(u)` $\geq \theta_{cc}$ `THEN centrally_accessible`

By expanding FireGNN's rule set with these and other established graph-theoretic measures, future work can develop neuro-symbolic models that provide deeper, multi-faceted explanations, moving from "what" a node's local structure is to "why" it is important in the broader context of the entire dataset.

A.8 Algorithmic Details of FireGNN Training

Algorithm 1 formalizes the end-to-end procedure described in Section 3. The algorithm details the forward pass, including the computation of topological features and their fusion with GNN embeddings, followed by the standard backpropagation step to update all learnable parameters.

Algorithm 1 FireGNN End-to-End Training Procedure

Input: Graph $G = (V, E)$, node features $X \in \mathbb{R}^{|V| \times d_{in}}$, node labels Y, training node indices $\mathcal{V}_L$.
Parameters: GNN weights $\{W^{(\ell)}\}_{\ell=1}^{L}$, fuzzy rule parameters $\{\theta_i, \alpha_i\}_{i=1}^{3}$, fusion weights $\{W_r, b_r, W_g, b_g\}$.
Output: Trained FireGNN model parameters.
procedure TRAINFIREGNN($G, X, Y, \mathcal{V}_L$)
  Initialize all parameters.
  for each training epoch **do**
    // GNN Forward Pass
    $H^{(0)} \leftarrow X$
    for $\ell = 0$ to $L - 1$ **do**
      $H^{(\ell+1)} \leftarrow \sigma(\tilde{D}^{-1/2}\tilde{A}\tilde{D}^{-1/2}H^{(\ell)}W^{(\ell)})$ ▷ Eq. (1)
    end for
    $H \leftarrow H^{(L)}$ ▷ Final GNN embeddings
    // Fuzzy Rule and Fusion Pass
    Initialize final embedding matrix $H' \in \mathbb{R}^{|V| \times d}$
    for each node $u \in V$ **do**
      Compute degree $d(u)$, clustering coeff. $C(u)$, 2-hop label agreement $L(u)$.
      $f_u \leftarrow [d(u), C(u), L(u)]$ ▷ Fact vector, Eq. (2)
      for $i = 1$ to 3 **do**
        $r_i(u) \leftarrow \sigma(\alpha_i(f_u[i] - \theta_i))$ ▷ Rule activation, Eq. (3)
      end for
      $r(u) \leftarrow [r_1(u), r_2(u), r_3(u)]$ ▷ Firing strength vector
      $h_u \leftarrow H[u, :]$ ▷ GNN embedding for node u
      $e_u \leftarrow W_r r(u) + b_r$ ▷ Project rule vector, Eq. (4)
      $g_u \leftarrow \sigma(W_g[h_u \parallel e_u] + b_g)$ ▷ Compute gate, Eq. (5)
      $h'_u \leftarrow g_u \odot h_u + (1 - g_u) \odot e_u$ ▷ Fuse embeddings, Eq. (6)
      $H'[u, :] \leftarrow h'_u$
    end for
    // Loss Computation and Backpropagation
    $\hat{Y}_{\mathcal{V}_L} \leftarrow$ Classifier$(H'[\mathcal{V}_L, :])$
    $\mathcal{L} \leftarrow$ CrossEntropyLoss$(\hat{Y}_{\mathcal{V}_L}, Y_{\mathcal{V}_L})$
    $\mathcal{L}$.backward()
    Update all parameters using an optimizer (e.g., Adam).
  end for
  return All trained parameters.
end procedure

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction accurately frame the paper's contributions. The central claim of proposing an "interpretable graph-based learning framework named FireGNN that integrates trainable fuzzy rules" is thoroughly detailed in the Methodology (Section 3), which describes the architecture for learning and fusing these rules. The claim of achieving "strong performance" is substantiated by comprehensive empirical results in Table 1, where FireGNN variants consistently outperform baseline GNNs across six datasets, including a notable +14.83% accuracy improvement on the challenging TissueMNIST dataset. Finally, the claim of generating "interpretable rule-based explanations" is demonstrated with a concrete example in Section 4, where a node's classification is explicitly justified by the activation strengths of the learned fuzzy rules.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We explicitly discuss limitations in the Conclusion and detail computational costs in Appendix A.3 (Table 3) for example, FireGNN has higher epoch times, occasional out-of-memory errors on large-scale datasets when using GAT. Furthermore, Appendix A.5 (Table 8) directly contrasts the limitations of standard GNNs with the solutions provided by FireGNN, and we acknowledge that the current framework only supports three rule types (degree, clustering coefficient, label agreement) and that "future extensions could explore learning richer or more complex rule types". These constraints are acknowledged as future directions.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: Formal assumptions and derivations for the auxiliary tasks benchmark are provided in Appendix A.2 (Eqns. 7-8). The fuzzy rule module is defined with clear mathematical formulations (Eqns. 2–6). While we do not present formal theorems, all modeling assumptions are explicitly stated.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper provides extensive details for reproducibility. Section 4 specifies the exact datasets (five from MedMNIST and Morpho-MNIST), evaluation protocol (3-fold cross-validation with three random seeds: 42, 43, 44), and data splits (80/20 train-validation). The hardware is specified in the "Computational efficiency" paragraph of Appendix A.3. The core methodology, including all mathematical formulations for the GNN updates and fuzzy rule mechanism, is detailed in Section 3. Crucially, Appendix A.4 provides comprehensive tables (Tables 4-7) detailing all hyperparameters for the GNN architecture, fuzzy rule layer, graph construction, and training setup

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code to reproduce all experiments is made available in the following repository: `https://github.com/basiralab/FireGNN`. All datasets used are publicly available benchmarks (MedMNIST v2 and Morpho-MNIST), with sources and licenses explicitly cited and detailed in Appendix A.1.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

Justification: All experimental settings are detailed. Section 4 describes the datasets, evaluation protocol and cross-validation strategy. The core methodology is in Section 3 where the use of cross-entropy loss for optimization is mentioned. Furthermore, Appendix A.4 provides a full breakdown of all hyperparameters in Tables 4-7, covering the GNN backbones, fuzzy rule module, graph construction, and training parameters

Guidelines:

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: All primary experimental results in Table 1 are reported with mean and standard deviation (e.g., 91.41±0.61) averaged across 9 experiments, so the standard deviation acts as the error margin. The methodology in Section 4 clarifies that these statistics are computed over nine independent runs (3-fold cross-validation times 3 random seeds), which robustly captures variance from both data partitioning and model initialization, adhering to best practices for GNN evaluation

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The "Computational efficiency" section in the appendix specifies the exact hardware used: "an Apple M2 Air with 16GB RAM and MPS acceleration GPU". Furthermore, Table 3 in Appendix A.3 provides a detailed breakdown of the average epoch time and peak memory usage for every model on every dataset, allowing for accurate resource estimation.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification:The research aligns with the NeurIPS Code of Ethics. Its primary objective is to enhance the transparency and trustworthiness of AI in the critical domain of medicine, promoting beneficence. The work relies on publicly available, anonymized benchmark datasets (MedMNIST and MorphoMNIST), thereby avoiding ethical issues related to human subjects, data privacy, or consent.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper extensively discusses the potential positive societal impact by framing the work as a solution to the "key barrier to clinical adoption" of AI, aiming to build "clinical trust" through interpretable models that support "trustworthy medical AI". While the model improves interpretability, negative impacts may arise if clinicians over-rely on simplified fuzzy rules without considering full patient context, or if the learned thresholds inadvertently encode dataset biases, leading to skewed decision support. Mitigation strategies include careful dataset auditing and positioning FireGNN strictly as an assistive tool rather than an autonomous system.

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper introduces a novel modeling framework but does not release new datasets or large-scale pre-trained models that would pose a high risk of misuse. Therefore, this question is not applicable.

    Guidelines:

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

Justification: The datasets used are publicly available and properly credited. Appendix A.1 explicitly states that MedMNIST v2 is available under the CC BY 4.0 license and Morpho-MNIST is derived from MNIST (which uses a CC BY-SA 3.0 license)

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We do not release new datasets or benchmarks. Only model code will be provided.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The research does not involve crowdsourcing or experiments with human subjects.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: No human subjects are involved; datasets are synthetic or anonymized public biomedical datasets.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: LLMs were not used in the design of FireGNN's methodology or experiments. Only standard scientific writing tools were employed.