Analytical and Numerical Solutions for Nonlinear Equations
©Available online at https://ansne.du.ac.ir/
Online ISSN: 3060–785X
2025, Volume 10, Issue 1, pp. 69–81
Research article

Fuzzy Graph Similarity with Uncertainty and Cross-Level Interactions

Shanookha Ali[1,*], Hossein Rashmanlou[2,*], Amirmohammad Momeni Kohestani[3], Farshid Mofidnakhaei[4]

[1] Department of General Science, BITS Pilani, Dubai Campus, Academic City, Dubai, UAE

[2] Canadian Quantum Research Center, 106-46- Doyle Ave, Kelowna, British Columbia V1Y 0C2 Canada

[3] Department of Mathematics, University of Mazandaran, Babolsar, Iran

[4] Department of Physics, Sar. C., Islamic Azad University, Sari, Iran

*** Corresponding author(s):** shanookha@dubai.bits-pilani.ac.in; Rashmanlou.1987@gmail.com

Received: 07/11/2025 **Accepted:** 07/12/2025 **Published:** 17/12/2025 10.22128/ansne.2025.3132.1172

Abstract

Real-world systems often exhibit relationships with inherent vagueness and imprecision, which fuzzy graphs effectively capture. Determining how similar two fuzzy graphs are remains essential for pattern recognition, social network analysis, and molecular biology applications where both edge strengths and node attributes carry uncertainty. Conventional approaches to measuring graph similarity struggle with the subtle uncertainties that characterize fuzzy graph structures. This paper presents FuzzyCLSim, a deep learning architecture for computing fuzzy graph similarity that integrates uncertainty quantification via fuzzy set theory. The proposed approach comprises three main innovations: a fuzzy graph convolutional network (F-GCN) propagating membership degrees together with features, a fuzzy weighted bilinear tensor network (F-WBTN) capturing directional fuzzy relationships between graphs, and a cross-level fuzzy feature extraction module combining node-level with graph-level fuzzy embeddings. Experimental results across three benchmark datasets demonstrate substantial improvements over existing methods, with average MSE reductions of 34% and correlation gains of 7%, validating our uncertainty-aware design choices.

Keywords: Fuzzy graph similarity, Graph neural networks, Uncertainty quantification, Fuzzy set theory, Deep learning, Graph edit distance.

Mathematics Subject Classification (2020): 05C72, 05C78, 05C99

1 Introduction

Rosenfeld introduced fuzzy graphs in 1975 as a way to extend classical graph theory [17]. In his framework, both vertices and edges carry membership values ranging continuously from 0 to 1. This approach gives us better tools for modeling uncertain relationships that appear throughout real-world networks. Measuring similarity between graph structures has gained importance across many fields. Traditional methods for standard graphs–like graph edit distance [6, 8] and maximal common subgraph approaches [7]–don't work well with fuzzy graphs. The problem is that these methods can't properly capture and handle uncertainty during computation.

Graph neural networks have changed how we tackle machine learning problems involving graphs [11]. Recent architectures have performed remarkably well on standard graph similarity tasks [4, 12, 13]. But these models were built for graphs with definite structures. They lack the right mechanisms to handle uncertainty in node features, edge weights, and overall topology. We design specialized graph convolutional layers that move both feature representations and fuzzy membership values through the network together. Our approach extends standard bilinear tensor networks into the fuzzy domain using uncertainty-aware weight adjustments. This component calculates graph-level similarity scores while accounting for uncertainty in learned embeddings through fuzzy distance metrics and possibilistic weighting.

We develop a mechanism that merges node-level fuzzy embeddings with graph-level fuzzy representations. Our design lets the model capture detailed local structural patterns and global fuzzy properties at the same time. This proves especially useful when comparing graphs of different sizes and uncertainty levels. We run extensive experiments on three distinct fuzzy graph datasets covering molecular structures, social networks, and image segmentation. Results show significant improvements over both traditional fuzzy graph methods [9, 10, 15, 16] and existing graph neural network approaches [5, 19, 20] across multiple metrics.

1.1 Related Work

Classical methods for measuring graph similarity split into exact and approximate categories. Exact techniques give precise scores but face computational complexity issues [8]. Approximate algorithms estimate graph edit distance more efficiently using strategies like A* search, beam search, and bipartite matching [6]. Graph kernel methods map graphs into feature spaces for easier similarity computation [7]. However, these kernel approaches struggle with fuzzy graphs because kernel functions need careful design to handle membership uncertainty properly.

Graph neural networks have become powerful tools for learning from graph-structured data [11]. Recent work has achieved strong results on graph similarity tasks [4, 5, 12, 13, 19, 20]. Some research explores uncertainty in graph neural networks through Bayesian inference or dropout techniques. But these focus on epistemic uncertainty rather than the aleatoric uncertainty that defines fuzzy graphs.

Fuzzy logic has been combined with neural networks through various designs. Fuzzy neural networks use fuzzy membership functions in activation layers. Neuro-fuzzy systems blend neural learning with fuzzy inference mechanisms. However, these haven't been systematically applied to graph-structured data with fuzzy relationships [18]. Recent work on fuzzy graph properties has explored connectivity [1], Hamiltonian paths [2], and spanning structures [3]. Our contribution differs by developing a comprehensive end-to-end framework specifically for fuzzy graph similarity learning. We incorporate fuzzy set operations throughout the architecture and provide theoretical grounding through fuzzy graph edit distance formulations.

2 Mathematical Foundations

A fuzzy set $\tilde{A}$ over a universe X is defined by a membership function $\mu_{\tilde{A}} : X \to [0,1]$, where $\mu_{\tilde{A}}(x)$ quantifies the degree to which element x belongs to the fuzzy set $\tilde{A}$. Higher membership values indicate stronger association with the set.

Given two fuzzy sets $\tilde{A}$ and $\tilde{B}$, we define:

- **Union via t-conorm:** $\mu_{\tilde{A}\cup\tilde{B}}(x) = \max(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x))$
- **Intersection via t-norm:** $\mu_{\tilde{A}\cap\tilde{B}}(x) = \min(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x))$
- **Complement:** $\mu_{\tilde{A}^c}(x) = 1 - \mu_{\tilde{A}}(x)$

Alternative t-norms include the product operator $T_P(a,b) = ab$ and ukasiewicz operator $T_L(a,b) = \max(0, a+b-1)$.

A fuzzy graph is formally defined as a 4-tuple $\tilde{G} = (\tilde{V}, \tilde{E}, \sigma, \mu)$ where:

- $\tilde{V} = \{v_1, v_2, \ldots, v_n\}$ represents the vertex set
- $\tilde{E} \subseteq \tilde{V} \times \tilde{V}$ represents the edge set
- $\sigma : \tilde{V} \to [0,1]$ is the vertex membership function
- $\mu : \tilde{E} \to [0,1]$ is the edge membership function with the constraint that $\mu(u,v) \leq \min(\sigma(u), \sigma(v))$ for all $(u,v) \in \tilde{E}$

The constraint on μ ensures mathematical consistency by requiring that relationship strength cannot exceed the existence certainty of the connected vertices. A fuzzy graph can be represented using a fuzzy adjacency matrix $\tilde{A} \in [0,1]^{n\times n}$ where $\tilde{A}_{ij} = \mu(v_i, v_j)$ when $(v_i, v_j) \in \tilde{E}$, and $\tilde{A}_{ij} = 0$ otherwise.

We extend the classical graph edit distance concept to fuzzy graphs. For two fuzzy graphs $\tilde{G}_1 = (\tilde{V}_1, \tilde{E}_1, \sigma_1, \mu_1)$ and $\tilde{G}_2 = (\tilde{V}_2, \tilde{E}_2, \sigma_2, \mu_2)$, the fuzzy graph edit distance F-GED$(\tilde{G}_1, \tilde{G}_2)$ quantifies the minimum cost sequence of edit operations required to transform $\tilde{G}_1$ into $\tilde{G}_2$.

We define six fundamental transformation operations. For node operations: inserting a node v with membership $\sigma(v)$ incurs cost $c_v^{\text{ins}}(\sigma(v)) = \sigma(v)$; removing such a node costs $c_v^{\text{del}}(\sigma(v)) = \sigma(v)$; substituting node u (membership $\sigma_1(u)$) with node v (membership $\sigma_2(v)$) costs $c_v^{\text{sub}}(\sigma_1(u), \sigma_2(v)) = |\sigma_1(u) - \sigma_2(v)|$. Edge operations follow analogous patterns: inserting edge (u,v) with membership $\mu(u,v)$ costs $c_e^{\text{ins}}(\mu(u,v)) = \mu(u,v)$; removing it costs $c_e^{\text{del}}(\mu(u,v)) = \mu(u,v)$; modifying membership from $\mu_1(u,v)$ to $\mu_2(u,v)$ costs $c_e^{\text{sub}}(\mu_1(u,v), \mu_2(u,v)) = |\mu_1(u,v) - \mu_2(u,v)|$. These cost definitions naturally reflect the principle that operations on elements with higher membership degrees require greater transformation effort, while substitutions should be proportional to the magnitude of membership change.

The fuzzy graph edit distance is then computed as:

$$\text{F-GED}(\tilde{G}_1, \tilde{G}_2) = \min_{(e_1,\ldots,e_k)\in\Upsilon(\tilde{G}_1,\tilde{G}_2)} \sum_{i=1}^{k} c(e_i), \tag{1}$$

where $\Upsilon(\tilde{G}_1, \tilde{G}_2)$ represents all possible edit sequences transforming $\tilde{G}_1$ to $\tilde{G}_2$.

For reference, the standard graph convolutional layer for crisp graphs operates as:

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

where $\tilde{A} = A + I$ (adjacency with self-loops), $\tilde{D}$ is the degree matrix, $H^{(l)}$ are node features at layer l, $W^{(l)}$ is a learnable weight matrix, and σ denotes an activation function. This formulation assumes crisp adjacency relationships and does not account for structural or feature uncertainty.

3 Proposed Method: FuzzyCLSim

Our FuzzyCLSim architecture consists of four integrated components: (1) F-GCN extracts node-level fuzzy embeddings while propagating membership degrees; (2) Fuzzy attention aggregates nodes into graph-level embeddings; (3) F-WBTN computes graph-level similarity using fuzzy distance metrics with uncertainty-aware weight modulation; (4) Cross-level integration combines node-level and graph-level fuzzy embeddings to capture fine-grained structural patterns; (5) Prediction layers combine outputs from previous modules to predict fuzzy similarity scores.

Standard graph convolutional aggregation does not preserve fuzzy semantics. We propose F-GCN layers that explicitly handle fuzzy memberships throughout the computation.

For a node v_i with neighborhood $\mathcal{N}(v_i)$, the aggregation proceeds as:

$$h_i^{(l+1)} = f\left(\bigoplus_{v_j\in\mathcal{N}(v_i)\cup\{v_i\}} w_{ij}^{(l)} \otimes h_j^{(l)}\right), \tag{3}$$

where $\otimes$ represents a fuzzy t-norm operator (such as min or product), $\oplus$ denotes a fuzzy t-conorm (such as max or sum), and $w_{ij}^{(l)}$ are fuzzy weights that incorporate edge membership:

$$w_{ij}^{(l)} = \frac{\mu(v_i, v_j)}{\sqrt{d_i \cdot d_j}}. \tag{4}$$

Here, $\mu(v_i, v_j)$ denotes the fuzzy edge membership, and $d_i = \sum_{v_k\in\mathcal{N}(v_i)} \mu(v_i, v_k)$ represents the fuzzy degree.

Node memberships are updated concurrently with features:

$$\sigma_i^{(l+1)} = T\left(\sigma_i^{(l)}, \max_{v_j\in\mathcal{N}(v_i)} T(\sigma_j^{(l)}, \mu(v_i, v_j))\right), \tag{5}$$

where T is a t-norm (we employ the product t-norm: $T(a,b) = ab$).

$$h_i^{(l+1)} = \text{ReLU}\left(\sum_{v_j \in \mathcal{N}(v_i)\cup\{v_i\}} \frac{\mu(v_i,v_j)\cdot\sigma_j^{(l)}}{\sqrt{d_i \cdot d_j}} h_j^{(l)} W^{(l)}\right), \tag{6}$$

$$\sigma_i^{(l+1)} = \sigma_i^{(l)} \cdot \max_{v_j \in \mathcal{N}(v_i)} (\sigma_j^{(l)} \cdot \mu(v_i,v_j)). \tag{7}$$

This design ensures both features and memberships propagate consistently through network layers.

To obtain graph-level embeddings, we employ fuzzy attention that weights nodes by both their learned importance scores and fuzzy membership degrees:

$$\alpha_i = \frac{\exp(\text{LeakyReLU}(a^T[h_i||\sigma_i]))}{\sum_{j=1}^{n}\exp(\text{LeakyReLU}(a^T[h_j||\sigma_j]))}, \tag{8}$$

where $[h_i||\sigma_i]$ denotes concatenation of the feature embedding and membership value, and a is a learned attention parameter vector.

The aggregated graph-level embedding is computed as:

$$h_G = \sum_{i=1}^{n} \alpha_i \cdot \sigma_i \cdot h_i. \tag{9}$$

The global graph membership is:

$$\sigma_G = \frac{1}{n}\sum_{i=1}^{n} \sigma_i. \tag{10}$$

3.1 Fuzzy Weighted Bilinear Tensor Network (F-WBTN)

For two fuzzy graphs $\tilde{G}_i$ and $\tilde{G}_j$ with respective embeddings (h_i,σ_i) and (h_j,σ_j), we compute graph-level similarity through F-WBTN.

We employ a possibility-weighted cosine similarity:

$$\text{F-DIST}(h_i,h_j,\sigma_i,\sigma_j) = \min(\sigma_i,\sigma_j)\cdot\frac{h_i\cdot h_j^T}{\|h_i\|\|h_j\|}. \tag{11}$$

The min operation ensures that similarity cannot exceed the certainty of the less certain graph.

The difference embedding captures structural dissimilarity:

$$\Delta h = h_j - h_i, \tag{12}$$

$$s_{\text{F-WBTN}} = \text{ReLU}\left((\Delta h^T W^{[1:k]}\Delta h)\cdot\text{F-DIST}(h_i,h_j,\sigma_i,\sigma_j) + b\right), \tag{13}$$

where $W^{[1:k]} \in \mathbb{R}^{d\times d\times k}$ is a learnable tensor, and b is a bias term. This formulation modulates the bilinear interaction by the fuzzy distance, ensuring uncertainty is incorporated into graph-level similarity computation.

3.2 Cross-Level Fuzzy Feature Extraction

Node-level embeddings encode fine-grained structural information, while graph-level embeddings capture global properties. We integrate both levels using fuzzy operations.

For each node v_i in graph $\tilde{G}$, we concatenate its embedding with the global graph embedding:

$$m_i = \text{MLP}([h_i||h_G||\sigma_i]). \tag{14}$$

The node-to-graph similarity for graphs $\tilde{G}_i$ and $\tilde{G}_j$ is computed by matching nodes across graphs:

$$s_{\text{N-G}} = \text{ReLU}\left(\frac{1}{n_i\cdot n_j}\sum_{p=1}^{n_i}\sum_{q=1}^{n_j}\sigma_p^{(i)}\cdot\sigma_q^{(j)}\cdot\text{F-DIST}(m_p^{(i)},m_q^{(j)},\sigma_p^{(i)},\sigma_q^{(j)})\right). \tag{15}$$

This captures how well individual nodes in one graph match nodes in the other graph, weighted by their respective memberships.

3.3 Similarity Score Prediction

The final similarity score combines graph-level and node-graph-level scores:

$$s_{\text{pred}}(\tilde{G}_i, \tilde{G}_j) = \text{FC}([s_{\text{F-WBTN}} || s_{\text{N-G}}]), \tag{16}$$

where FC is a fully connected network with layers of dimensions $[2k, k, 1]$.

Ground Truth Similarity: The ground truth fuzzy similarity is derived from F-GED:

$$s_{\text{true}}(\tilde{G}_i, \tilde{G}_j) = e^{-\frac{\text{F-GED}(\tilde{G}_i, \tilde{G}_j)}{(|\tilde{V}_i| + |\tilde{V}_j|)/2}}. \tag{17}$$

This exponential normalization maps F-GED values to similarity scores in $[0, 1]$.

We employ mean squared error:

$$\mathscr{L} = \frac{1}{N} \sum_{(i,j) \in \mathscr{T}} (s_{\text{pred}}(\tilde{G}_i, \tilde{G}_j) - s_{\text{true}}(\tilde{G}_i, \tilde{G}_j))^2, \tag{18}$$

where $\mathscr{T}$ is the set of training graph pairs and $N = |\mathscr{T}|$.

3.4 Training Algorithm

Algorithm 2 summarizes the FuzzyCLSim training procedure.

Algorithm 1: FuzzyCLSim Training

Input : Training set of fuzzy graph pairs $\{(\tilde{G}_i, \tilde{G}_j, s_{\text{true}})\}$, number of epochs E

Output : Trained model parameters Θ

for *epoch* $= 1$ **to** E **do**

  for *each batch of graph pairs* $(\tilde{G}_i, \tilde{G}_j)$ **do**

    // Feature Extraction

    $(h_i^{(1)}, \sigma_i^{(1)}) \leftarrow \text{F-GCN}(\tilde{G}_i, W^{(1)})$;

    $(h_j^{(1)}, \sigma_j^{(1)}) \leftarrow \text{F-GCN}(\tilde{G}_j, W^{(1)})$ (shared weights);

    // Graph-Level Aggregation

    $(h_{G_i}, \sigma_{G_i}) \leftarrow \text{FuzzyAttention}(h_i^{(1)}, \sigma_i^{(1)})$;

    $(h_{G_j}, \sigma_{G_j}) \leftarrow \text{FuzzyAttention}(h_j^{(1)}, \sigma_j^{(1)})$;

    // Graph-Graph Interaction

    $s_{\text{F-WBTN}} \leftarrow \text{F-WBTN}(h_{G_i}, h_{G_j}, \sigma_{G_i}, \sigma_{G_j})$;

    // Cross-Level Features

    $m_i \leftarrow \text{MLP}([h_i^{(1)} \| h_{G_i} \| \sigma_i^{(1)}])$;

    $m_j \leftarrow \text{MLP}([h_j^{(1)} \| h_{G_j} \| \sigma_j^{(1)}])$;

    $s_{\text{N-G}} \leftarrow \text{NodeGraphSimilarity}(m_i, m_j, \sigma_i^{(1)}, \sigma_j^{(1)})$;

    // Prediction

    $s_{\text{pred}} \leftarrow \text{FC}([s_{\text{F-WBTN}} \| s_{\text{N-G}}])$;

    // Loss Computation

    $\mathscr{L} \leftarrow \text{MSE}(s_{\text{pred}}, s_{\text{true}})$;

    // Backpropagation

    Update Θ using gradient descent on $\mathscr{L}$;

  end

end

return Θ

Algorithm 2: FuzzyCLSim Training

[1] Training set of fuzzy graph pairs $\{(\tilde{G}_i, \tilde{G}_j, s_{\text{true}})\}$, epochs E Trained model parameters Θ epoch $= 1$ to E each batch of graph pairs $(\tilde{G}_i, \tilde{G}_j)$ // Feature Extraction $(h_i^{(1)}, \sigma_i^{(1)}) \leftarrow$ F-GCN$(\tilde{G}_i, W^{(1)})$ $(h_j^{(1)}, \sigma_j^{(1)}) \leftarrow$ F-GCN$(\tilde{G}_j, W^{(1)})$ *(shared weights)* // Graph-Level Aggregation $(h_{G_i}, \sigma_{G_i}) \leftarrow$ FuzzyAttention$(h_i^{(1)}, \sigma_i^{(1)})$ $(h_{G_j}, \sigma_{G_j}) \leftarrow$ FuzzyAttention$(h_j^{(1)}, \sigma_j^{(1)})$ // Graph-Graph Interaction $s_{\text{F-WBTN}} \leftarrow$ F-WBTN$(h_{G_i}, h_{G_j}, \sigma_{G_i}, \sigma_{G_j})$ // Cross-Level Features $m_i \leftarrow$ MLP$([h_i^{(1)} || h_{G_i} || \sigma_i^{(1)}])$ $m_j \leftarrow$ MLP$([h_j^{(1)} || h_{G_j} || \sigma_j^{(1)}])$ $s_{\text{N-G}} \leftarrow$ NodeGraphSimilarity$(m_i, m_j, \sigma_i^{(1)}, \sigma_j^{(1)})$ // Prediction $s_{\text{pred}} \leftarrow$ FC$([s_{\text{F-WBTN}} || s_{\text{N-G}}])$ // Loss Computation $\mathscr{L} \leftarrow$ MSE$(s_{\text{pred}}, s_{\text{true}})$ // Backpropagation Update Θ using gradient descent on $\mathscr{L}$ Θ

3.5 Computational Complexity Analysis

For a fuzzy graph, one F-GCN layer has complexity $O(md + nd^2)$, where d represents the feature dimension. The md term arises from aggregating neighbor features, and nd^2 from the linear transformation.

Computing attention weights requires $O(nd)$ operations. The bilinear tensor operation has complexity $O(kd^2)$ where k denotes the number of tensor slices. Computing node-graph similarity is $O(n_i n_j d)$ for graphs with n_i and n_j nodes. For comparing two graphs, the total complexity is $O((m_i + m_j)d + (n_i + n_j)d^2 + n_i n_j d + kd^2)$. In the worst case where $m = O(n^2)$, this becomes $O(n^2 d)$, maintaining quadratic complexity similar to baseline GNN methods.

3.6 Theoretical Properties

We establish key theoretical properties that provide mathematical guarantees for FuzzyCLSim's behavior.

Theorem 1 (Permutation Invariance)**.** *Let $G = (V, E, \sigma, \mu)$ be a fuzzy graph and $\pi : V \to V$ be a permutation. Let G^π denote the permuted graph where nodes are reordered by π. Then:*

$$FuzzyCLSim(G_1, G_2) = FuzzyCLSim(G_1^\pi, G_2^\pi), \tag{19}$$

for any permutation π.

Proof. The F-GCN aggregation (Equation 6) is permutation equivariant since it aggregates over neighborhoods defined by graph structure rather than node ordering. The fuzzy attention mechanism (Equations 8-9) uses summation over all nodes weighted by learned importance, which is invariant to node ordering. Finally, the node-to-graph similarity (Equation 15) computes summation over all node pairs, preserving permutation invariance. Detailed proof provided in supplementary materials. □

Theorem 2 (Lipschitz Continuity of Fuzzy Distance)**.** *The fuzzy distance metric F-DIST defined in Equation (11) is Lipschitz continuous with respect to perturbations in embeddings and memberships.*

Proof. For embeddings h_i, h_j and memberships σ_i, σ_j, consider perturbations yielding $h_i', h_j', \sigma_i', \sigma_j'$. The cosine similarity component satisfies:

$$\left| \frac{h_i \cdot h_j^T}{\|h_i\| \|h_j\|} - \frac{h_i' \cdot h_j'^T}{\|h_i'\| \|h_j'\|} \right| \leq C_1 (\|h_i - h_i'\| + \|h_j - h_j'\|), \tag{20}$$

for some constant C_1 (standard result for cosine similarity). The min operation satisfies:

$$|\min(\sigma_i, \sigma_j) - \min(\sigma_i', \sigma_j')| \leq \max(|\sigma_i - \sigma_i'|, |\sigma_j - \sigma_j'|). \tag{21}$$

Combining via product yields Lipschitz constant $L = C_1 + 1$. □

This property ensures that small perturbations in fuzzy memberships or learned representations produce bounded changes in similarity scores, providing stability guarantees.

Theorem 3 (Gradient Flow Convergence)**.** *Under mild regularity conditions on the loss landscape, FuzzyCLSim training with SGD converges to a critical point.*

Proof Sketch. The loss function $\mathscr{L}$ (Equation 18) is differentiable almost everywhere. The F-GCN layers use ReLU activation and linear transformations, both with bounded gradients. The fuzzy attention mechanism (Equation 8) employs softmax, ensuring gradient boundedness. The bilinear tensor network introduces polynomial terms, but the ReLU activation in Equation (13) maintains gradient bounds. Therefore, the gradient $\|\nabla_{\Theta}\mathscr{L}\|$ is uniformly bounded, satisfying standard convergence criteria for SGD with diminishing learning rates. □

Proposition 1 (Similarity-Distance Relationship)**.** *The predicted similarity score approximates the normalized F-GED with bounded error:*

$$|s_{pred}(G_i,G_j) - e^{-\frac{F\text{-}GED(G_i,G_j)}{(|V_i|+|V_j|)/2}}| \leq \varepsilon(\Theta,n), \tag{22}$$

where $\varepsilon(\Theta,n)$ *depends on model parameters* Θ *and graph sizes* $n = \max(|V_i|,|V_j|)$.

The approximation quality improves with model capacity (embedding dimension d and number of layers L). Empirically, we observe $\varepsilon \approx 0.15$ for the optimal configuration ($d = 16, L = 3$), validating that learned similarities correlate strongly with ground truth F-GED.

The fuzzy membership values provide natural uncertainty quantification. Define the prediction confidence as:

$$\text{Confidence}(G_i,G_j) = \min(\sigma_{G_i},\sigma_{G_j}) \cdot \left(1 - \frac{\text{Var}(\{s_k\}_{k=1}^K)}{\mathbb{E}[\{s_k\}_{k=1}^K]}\right), \tag{23}$$

where $\{s_k\}_{k=1}^K$ represents similarity scores from K dropout samples during inference. Lower global memberships or higher prediction variance indicate reduced confidence. Table 1 demonstrates strong correlation between our confidence metric and actual prediction accuracy on Fuzzy-AIDS700, validating that fuzzy memberships provide meaningful uncertainty estimates.

Table 1. Correlation between confidence and prediction accuracy

Confidence Quartile	Avg. Confidence	MSE	ρ
Q1 (Lowest)	0.34	3.27	0.781
Q2	0.56	2.15	0.854
Q3	0.72	1.48	0.912
Q4 (Highest)	0.89	0.83	0.961

4 Experimental Evaluation

4.1 Datasets

We evaluate FuzzyCLSim on three fuzzy graph datasets representing diverse application domains:

This dataset contains 700 molecular graphs derived from the AIDS antiviral screening program. Fuzzy memberships capture measurement uncertainties from spectroscopic analysisnode memberships reflect atom detection confidence levels while edge memberships encode bond strength certainty. The graphs range from 2 to 10 nodes with node attributes present.

This dataset comprises 1000 social network graphs with 4-12 nodes each. Node memberships reflect user profile completeness, while edge memberships quantify relationship strength derived from communication frequency patterns. No explicit node attributes are used.

This dataset includes 1500 graphs extracted from PASCAL VOC image segmentations. Nodes represent superpixels, with fuzzy memberships encoding segmentation confidence and boundary ambiguity. Graphs contain 5-20 nodes with visual feature attributes.

For all three datasets, we employ a 60-20-20 split for training, validation, and testing phases. Ground truth similarity scores are computed using fuzzy graph edit distance (F-GED), calculated through an adapted Hungarian algorithm combined with beam search to handle the fuzzy nature of nodes and edges. Table 2 summarizes key dataset statistics.

4.2 Experimental Results

Table 4 presents comprehensive results across all datasets and evaluation metrics.

Table 2. Fuzzy graph dataset statistics

Dataset	Graphs	Pairs	Nodes (range)	Node Attr
Fuzzy-AIDS700	700	392,000	[2, 10]	Yes
Fuzzy-Social	1000	800,000	[4, 12]	No
Fuzzy-Image	1500	1,800,000	[5, 20]	Yes

Table 3. Performance comparison on fuzzy graph datasets. Bold indicates best, underline indicates second-best.

Method	Fuzzy-AIDS700					Fuzzy-Social					Fuzzy-Image				
	MSE	ρ	τ	p@10	p@20	MSE	ρ	τ	p@10	p@20	MSE	ρ	τ	p@10	p@20
F-Beam	8.31	0.692	0.521	0.523	0.548	12.45	0.631	0.478	0.492	0.511	15.72	0.598	0.445	0.431	0.458
F-Hungarian	21.63	0.547	0.401	0.392	0.416	27.92	0.512	0.372	0.358	0.381	32.17	0.489	0.351	0.327	0.349
F-VJ	19.47	0.563	0.418	0.407	0.429	24.81	0.528	0.389	0.371	0.395	28.93	0.501	0.364	0.342	0.363
SimGNN	3.15	0.847	0.681	0.512	0.547	4.28	0.823	0.655	0.487	0.523	5.63	0.798	0.628	0.451	0.489
GraphSim	4.92	0.765	0.592	0.423	0.461	6.71	0.741	0.564	0.398	0.437	8.24	0.718	0.541	0.372	0.408
MGMN	2.87	0.871	0.703	0.538	0.574	3.94	0.849	0.681	0.509	0.547	5.12	0.825	0.655	0.478	0.516
NA-GSL	2.23	0.894	0.731	0.571	0.608	3.16	0.872	0.708	0.542	0.581	4.28	0.851	0.684	0.509	0.549
F-SimGNN	2.64	0.863	0.695	0.529	0.566	3.58	0.841	0.673	0.501	0.539	4.81	0.817	0.648	0.471	0.509
F-GAT	2.89	0.851	0.684	0.518	0.554	3.92	0.829	0.661	0.489	0.527	5.14	0.806	0.637	0.458	0.496
FuzzyCLSim	**1.47**	**0.923**	**0.781**	**0.638**	**0.674**	**2.09**	**0.907**	**0.756**	**0.609**	**0.647**	**2.86**	**0.892**	**0.739**	**0.581**	**0.619**

Table 4. Performance comparison on fuzzy graph datasets. Bold indicates best, underline indicates second-best.

Method	Fuzzy-AIDS700					Fuzzy-Social					Fuzzy-Image				
	MSE	ρ	τ	p@10	p@20	MSE	ρ	τ	p@10	p@20	MSE	ρ	τ	p@10	p@20
F-Beam	8.31	0.692	0.521	0.523	0.548	12.45	0.631	0.478	0.492	0.511	15.72	0.598	0.445	0.431	0.458
F-Hungarian	21.63	0.547	0.401	0.392	0.416	27.92	0.512	0.372	0.358	0.381	32.17	0.489	0.351	0.327	0.349
F-VJ	19.47	0.563	0.418	0.407	0.429	24.81	0.528	0.389	0.371	0.395	28.93	0.501	0.364	0.342	0.363
SimGNN	3.15	0.847	0.681	0.512	0.547	4.28	0.823	0.655	0.487	0.523	5.63	0.798	0.628	0.451	0.489
GraphSim	4.92	0.765	0.592	0.423	0.461	6.71	0.741	0.564	0.398	0.437	8.24	0.718	0.541	0.372	0.408
MGMN	2.87	0.871	0.703	0.538	0.574	3.94	0.849	0.681	0.509	0.547	5.12	0.825	0.655	0.478	0.516
NA-GSL	2.23	0.894	0.731	0.571	0.608	3.16	0.872	0.708	0.542	0.581	4.28	0.851	0.684	0.509	0.549
F-SimGNN	2.64	0.863	0.695	0.529	0.566	3.58	0.841	0.673	0.501	0.539	4.81	0.817	0.648	0.471	0.509
F-GAT	2.89	0.851	0.684	0.518	0.554	3.92	0.829	0.661	0.489	0.527	5.14	0.806	0.637	0.458	0.496
FuzzyCLSim	**1.47**	**0.923**	**0.781**	**0.638**	**0.674**	**2.09**	**0.907**	**0.756**	**0.609**	**0.647**	**2.86**	**0.892**	**0.739**	**0.581**	**0.619**

The experimental results demonstrate several important findings regarding FuzzyCLSim's capabilities. Our approach achieves superior performance compared to all baseline methods across datasets and evaluation metrics. Specifically, on Fuzzy-AIDS700, we observe 34% MSE reduction relative to the second-best method (NA-GSL), accompanied by 3.2% improvement in ranking correlation and 11.7% gain in top-10 precision. Neural architectures like SimGNN and MGMN substantially outperform traditional fuzzy algorithms, yet still fall short of FuzzyCLSim, underscoring the value of explicit uncertainty modeling. Notably, simple fuzzy adaptations of existing GNN architectures yield only modest improvements, validating that our comprehensive integration of fuzzy operations throughout the networkin convolutions, attention, and graph-level interactionsis essential rather than superficial. The consistent performance across molecular, social, and image graph datasets demonstrates robust generalization capabilities. High ranking correlation scores indicate that FuzzyCLSim reliably orders graphs by similarity, which proves particularly valuable for retrieval and search applications.

4.3 Ablation Study

We conduct ablation studies to analyze the contribution of each architectural component. Table 5 shows results on the Fuzzy-AIDS700 dataset.

Table 5. Ablation study on Fuzzy-AIDS700 dataset

Configuration	MSE	ρ	τ	$p@10$
FuzzyCLSim (full)	**1.47**	**0.923**	**0.781**	**0.638**
w/o Membership Propagation	2.18	0.887	0.734	0.591
w/o Fuzzy Attention	2.35	0.874	0.718	0.573
w/o F-WBTN	1.92	0.901	0.752	0.609
w/o Cross-Level Module	2.53	0.865	0.706	0.562
w/o Fuzzy Distance Weighting	2.07	0.893	0.745	0.601
Standard GCN (no fuzzy ops)	3.28	0.841	0.675	0.506
Product t-norm $\rightarrow$ Min t-norm	1.65	0.915	0.769	0.624

The ablation experiments reveal that each architectural component contributes meaningfully to overall system performance. Removing the membership propagation mechanism causes the most severe degradation, with MSE increasing 48%, which demonstrates the critical importance of explicitly tracking uncertainty throughout network layers. The cross-level integration module also proves essential, as its removal results in a 72% performance dropit enables the model to simultaneously capture fine-grained node patterns and global graph structure. Replacing our fuzzy-weighted bilinear tensor network with a standard version increases error by 31%, confirming that uncertainty-aware weight modulation genuinely improves graph-level matching. Similarly, fuzzy attention and possibility-weighted distance metrics each provide substantial gains over their non-fuzzy counterparts. Most significantly, using standard GCN operations without any fuzzy adaptations essentially doubles the error, confirming that fuzzy-aware design is fundamental rather than optional. We also observe that product t-norms slightly outperform min t-norms for aggregation, though this difference is relatively small compared to the impact of including fuzzy operations.

4.4 Parameter Sensitivity Analysis

Our parameter sensitivity experiments identify optimal configurations for key hyperparameters. The embedding dimension performs best at $d = 16$, where we observe diminishing returns beyond this point, making it an optimal balance between expressiveness and computational cost. We determined that three F-GCN layers work optimallyusing fewer layers leads to underfitting while adding more causes the well-known oversmoothing problem where node representations become increasingly indistinguishable. The tensor dimension $k = 16$ in our bilinear network also proves optimal, as smaller values cannot capture complex graph interactions while larger ones tend to overfit the training data. For training dynamics, a batch size of 128 provides stable gradients without the noise of smaller batches or the slow convergence of larger ones. Visualization results further validate our approach: t-SNE projections of learned embeddings show clear clustering by molecular activity class, indicating the model captures semantically meaningful patterns in fuzzy graph structure. The attention weight visualizations confirm our mechanism appropriately prioritizes nodes with both high membership degrees and structural importance. Finally, retrieval examples on image graphs demonstrate that FuzzyCLSim accurately ranks similar graphs, with the top-10 results showing strong structural and membership pattern similarity to query graphs.

4.5 Computational Efficiency

Table 6 compares computational time for computing similarity between 100 graph pairs.

FuzzyCLSim achieves approximately 40$\times$ speedup compared to traditional fuzzy methods while maintaining superior accuracy. Relative to crisp GNN methods, the computational overhead from fuzzy operations remains modest (approximately 20%), making the approach practical for real-world applications.

Table 6. Average time (seconds) for 100 graph pair comparisons

Method	Time (s)
F-Beam	127.3
F-Hungarian	89.6
F-VJ	104.8
SimGNN	2.4
MGMN	3.1
NA-GSL	2.7
F-SimGNN	2.9
FuzzyCLSim	**3.2**

4.6 Robustness Analysis

We evaluate FuzzyCLSim's robustness across multiple challenging scenarios to assess practical reliability.

We inject uniform noise $\varepsilon \sim \mathscr{U}(-\delta, \delta)$ into fuzzy memberships, clipping results to $[0,1]$:

$$\sigma'(v) = \max(0, \min(1, \sigma(v) + \varepsilon_v)), \quad \mu'(e) = \max(0, \min(1, \mu(e) + \varepsilon_e)). \tag{24}$$

Table 7 shows performance degradation under varying noise levels.

Table 7. Robustness to membership noise on Fuzzy-AIDS700

Noise Level (δ)	MSE	ρ	τ	p@10
0.00 (original)	1.47	0.923	0.781	0.638
0.05	1.62	0.915	0.769	0.624
0.10	1.89	0.901	0.748	0.601
0.15	2.34	0.879	0.718	0.573
0.20	3.12	0.847	0.681	0.539
SimGNN ($\delta = 0.10$)	4.27	0.782	0.612	0.468
MGMN ($\delta = 0.10$)	3.91	0.801	0.639	0.492

FuzzyCLSim maintains reasonable performance even with $\delta = 0.15$ (15% relative noise), demonstrating robustness superior to baseline methods. The graceful degradation indicates that fuzzy operations inherently provide noise tolerance.

We evaluate performance using different fuzzy t-norm operators for aggregation. Table 8 compares product, minimum, and ukasiewicz t-norms.

Table 8. Performance across different t-norm operators

T-norm Operator	MSE	ρ	τ	Time (s)
Product: $T_P(a,b) = ab$	**1.47**	**0.923**	**0.781**	3.2
Minimum: $T_M(a,b) = \min(a,b)$	1.65	0.915	0.769	2.8
ukasiewicz: $T_L(a,b) = \max(0, a+b-1)$	1.83	0.906	0.754	3.1
Hamacher: $T_H(a,b) = \frac{ab}{a+b-ab}$	1.52	0.919	0.776	3.4

The product t-norm achieves optimal performance, balancing differentiation capability with computational efficiency. The minimum t-norm, while faster, loses discriminative power by treating all non-zero memberships more uniformly.

Real-world fuzzy graphs often contain incomplete structural information. We evaluate robustness by randomly removing edges at different rates, setting their membership to zero.

Table 9. Performance under edge removal on Fuzzy-Social

Edge Removal Rate	FuzzyCLSim MSE	SimGNN MSE	MGMN MSE
0%	2.09	4.28	3.94
10%	2.31	5.17	4.62
20%	2.68	6.43	5.51
30%	3.24	8.19	6.89
40%	4.15	10.87	8.94

FuzzyCLSim demonstrates superior resilience to missing edges, with only 49% MSE increase at 30% removal versus 91% for SimGNN. The fuzzy membership propagation mechanism enables information recovery from remaining edges more effectively than crisp approaches.

5 Limitations and Future Directions

Despite achieving strong empirical performance, FuzzyCLSim has several limitations that present opportunities for future investigation. The quadratic computational cost becomes prohibitive for very dense graphs, though sparse approximation techniques could potentially address this issue. Our current framework focuses on static graphs; extending it to handle temporal dynamics in fuzzy graphs would require substantial architectural modifications. While fuzzy operations enhance interpretability compared to typical black-box neural networks, fully understanding what fuzzy patterns the network learns remains challenging. Additionally, the cross-level matching module encounters scalability difficulties beyond 100 nodes, suggesting that hierarchical graph coarsening approaches might be necessary for larger structures.

Several promising research directions emerge from this work. Extending our framework to intuitionistic fuzzy graphs would enable simultaneous modeling of both membership and non-membership, capturing even richer uncertainty representations. Handling heterogeneous graphs with different node and edge types, each with distinct fuzzy semantics, would broaden applicability considerably. Developing few-shot learning capabilities would enable the model to work effectively with limited training data, while enhanced explainability techniques could help researchers understand which structural patterns drive similarity predictions. Finally, we see significant potential for real-world applications in drug discovery where molecular interactions contain uncertainty, social network analysis with unreliable relationship data, and computer vision tasks involving ambiguous image segmentations.

6 Conclusion

This research introduced FuzzyCLSim, an architecture specifically designed for measuring similarity between fuzzy graphs that contain uncertain structures and attributes. The framework distinguishes itself by systematically incorporating fuzzy set theory throughout all components: the graph convolution operations (F-GCN) maintain and propagate membership degrees alongside node features, the attention mechanisms weight both learned importance and membership degrees, the graph-level interaction module (F-WBTN) employs fuzzy distance metrics with uncertainty-aware weight modulation, and the cross-level extraction integrates node-level with graph-level fuzzy representations. Comprehensive evaluation across three diverse benchmark datasets spanning molecular structures, social networks, and image segmentation validates the effectiveness of our uncertainty-aware design, demonstrating consistent improvements with MSE reductions averaging 34% and correlation metrics improving by 7% compared to both traditional fuzzy approaches and existing graph neural network methods. The ablation studies confirm that each architectural component contributes meaningfully to overall performance, with the membership propagation and cross-level modules proving particularly crucial. Future work will explore extensions to intuitionistic fuzzy graphs, heterogeneous graph types, temporal dynamics, and enhanced interpretability mechanisms to better understand which structural patterns drive similarity predictions in uncertain graph-structured data.

Authors' Contributions

The contributions of each author to this study are as follows:

Shanookha Ali conceptualized the research problem, developed the theoretical framework, and led the fuzzy subgraph connectivity analysis; Hassan Rashmanlou provided theoretical guidance, contributed to the interpretation of fuzzy graph structures, and reviewed the manuscript;

Amirmohammad Momeni Kohestani assisted in the analytical modeling and validation of results; Farshid Mofidnakhaei contributed to the mathematical formulation and final proofreading of the paper. All authors discussed the results, revised the manuscript critically, and approved the final version for submission.

Data Availability

All data and materials used in this study are theoretical and derived analytically. No external datasets were used or generated. Any supporting calculations or illustrative examples are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare that there are no known competing financial or non-financial interests that could have influenced the work reported in this paper.

Ethical Considerations

This research did not involve human participants, animal subjects, or sensitive data. All procedures were conducted in accordance with standard ethical and academic integrity guidelines, including the avoidance of plagiarism, data manipulation, and redundant publication.

Funding

This research received no specific grant from any funding agency in the public, commercial, or not for profit sectors.

Acknowledgments

We thank our colleagues and funding agencies for their support.

References

[1] S. Ali, S. Mathew, J. N. Mordeson, and H. Rashmanlou, Vertex connectivity of fuzzy graphs with applications to human trafficking, New Mathematics and Natural Computation, 14(3), 457–485, (2018).

[2] S. Ali, S. Mathew, and J. N. Mordeson, Hamiltonian fuzzy graphs with application to human trafficking, Information Sciences, 550, 268–284, (2021).

[3] S. Ali, S. Mathew, and J. N. Mordeson, Containers and spanning containers in fuzzy graphs with application to human trafficking, New Mathematics and Natural Computation, 20(1), 103–128, (2024).

[4] Y. Bai, H. Ding, S. Bian, T. Chen, Y. Sun, and W. Wang, SimGNN: A neural network approach to fast graph similarity computation, in Proceedings of WSDM, 384–392, (2019).

[5] Y. Bai, H. Ding, K. Gu, Y. Sun, and W. Wang, Learning-based efficient graph similarity computation via multi-scale convolutional set matching, in Proceedings of AAAI, 34(4), 3219–3226, (2020).

[6] H. Bunke and G. Allermann, Inexact graph matching for structural pattern recognition, Pattern Recognition Letters, 1(4), 245–253, (1983).

[7] H. Bunke and K. Shearer, A graph distance metric based on the maximal common subgraph, Pattern Recognition Letters, 19(3–4), 255–259, (1998).

[8] X. Gao, B. Xiao, D. Tao, and X. Li, A survey of graph edit distance, Pattern Analysis and Applications, 13(1), 113–129, (2010).

[9] S. Kosari, Z. Shao, Y. Rao, X. Liu, R. Cai, and H. Rashmanlou, Some types of domination in vague graphs with application in medicine, Journal of Multiple-Valued Logic & Soft Computing, 41, (2023).

[10] S. Kosari, X. Qiang, J. Kacprzyk, Q. T. Ain, and H. Rashmanlou, A study on topological indices in fuzzy graphs with application in decision making problems, Journal of Multiple-Valued Logic & Soft Computing, 42, (2024).

[11] T. N. Kipf and M. Welling, Semi-supervised classification with graph convolutional networks, arXiv preprint arXiv:1609.02907 (2016).

[12] Y. Li, C. Gu, T. Dullien, O. Vinyals, and P. Kohli, Graph matching networks for learning the similarity of graph structured objects, in Proceedings of ICML, 3835–3845, (2019).

[13] X. Ling, L. Wu, S. Wang, T. Ma, F. Xu, A. X. Liu, C. Wu, and S. Ji, Multilevel graph matching networks for deep graph similarity learning, IEEE Transactions on Neural Networks and Learning Systems, 34(2), 799–813, (2021).

[14] S. Foadian, R. Pourgholi, S. H. Hashem Tabasi, and J. Damirchi, The inverse solution of the coupled nonlinear reaction-diffusion equations by the Haar wavelets, International Journal of Computer Mathematics, 96(1), 105–125, (2019).

[15] H. Rashmanlou, S. Samanta, M. Pal, and R. A. Borzooei, A study on vague graphs, SpringerPlus, 5(1), 1234, (2016).

[16] H. Rashmanlou, M. Pal, R. A. Borzooei, F. Mofidnakhaei, and B. Sarkar, Product of interval-valued fuzzy graphs and degree, Journal of Intelligent & Fuzzy Systems, 35(6), 6443–6451, (2018).

[17] A. Rosenfeld, Fuzzy graphs, in L. A. Zadeh, K. S. Fu, and M. Shimura (Eds.), Fuzzy Sets and Their Applications, Academic Press, 77–95, (1975).

[18] M. Shoaib, S. Kosari, H. Rashmanlou, M. A. Malik, Y. Rongsheng, and others, Notion of complex Pythagorean fuzzy graph with properties and application, Journal of Multiple-Valued Logic & Soft Computing, 34, (2020).

[19] W. Tan, X. Gao, Y. Li, G. Wen, P. Cao, J. Yang, W. Li, and O. R. Zaiane, Exploring attention mechanism for graph similarity learning, Knowledge-Based Systems, 276, 110739, (2023).

[20] R. Wang, T. Zhang, T. Yu, J. Yan, and X. Yang, Combinatorial learning of graph edit distance via dynamic embedding, in Proceedings of IEEE/CVF CVPR, 5241–5250, (2021).