Contagion Path Prediction in Financial Enterprise Networks Using a Starling Murmuration Optimized Dual-Encoder Self-Attention GNN (SM-ISAGNN)

Renrong Jiang
School of Accounting and Finance, Taizhou Vocational College of Science & Technology, Taizhou Zhejiang, 318020, China
E-mail:jiangrr201100296@outlook.com

Keywords: Financial risk contagion, enterprise networks, contagion path prediction, starling murmuration optimizer-driven improved self-attention graph neural network (SM-ISAGNN) dual-encoder architecture

Received: July 21, 2025

Financial risk contagion refers to the cascading spread of financial distress among interconnected entities within a networked system, posing serious threats to economic stability. Predicting the pathways through which such contagion propagates is essential for early intervention and systemic risk mitigation. This research proposes a novel contagion path prediction algorithm based on a deep learning (DL) framework, designed to capture both the direct and indirect transmission of financial risks across enterprise networks. The Starling Murmuration Optimizer-driven improved Self-Attention Graph Neural Network (SM-ISAGNN) is applied to predict financial risk contagion paths in enterprise networks. The model was enhanced by integrating a dual-encoder architecture. The first encoder captures intra-entity risk using statistically significant financial indicators, legal records, and operational data. The second encoder models contagion dynamics using enterprise relation information derived from an enterprise knowledge graph. To collect the data, financial statements, including income, debt ratio, and liquidity indicators, along with credit scores and enterprise relationship information, were gathered from relevant financial databases and corporate records. The data was preprocessed by handling missing values and normalizing features. The SMO metaheuristic is employed to optimize attention weights, enhancing convergence and avoiding local minima. Experiments on a financial enterprise dataset demonstrate that SM-ISAGNN outperforms the baseline ISAGNN, achieving a higher path hit ratio (84.3% vs. 58.7%), a multi-hop detection rate (77.5% vs. 41.6%), and a lower false path prediction (7.9% vs. 21.3%). In 5-fold cross-validation, the model achieves an accuracy of 0.9667, a precision of 0.9658, a recall of 0.9667, and an F1-score of 0.9657. These results confirm SM-ISAGNN as a robust framework for early warning, risk visualization, and contagion path forecasting in financial enterprise networks.

Povzetek: Študija predstavlja SM-ISAGNN, globok grafski pristop s samo-pozornostjo in metahevristiko, ki z dvojnim kodirnikom napoveduje poti finančne okužbe v omrežjih podjetij.

1 Introduction

Financial risk is a critical issue that governments, enterprises, and investors must continuously monitor in today's interconnected global economy [1]. It refers to the possibility of monetary loss arising from volatile markets, credit defaults, liquidity shortages, operational inefficiencies, or adverse macroeconomic changes [2]. While risks may originate within a single firm or transaction, they often cascade across financial institutions and markets, creating systemic impacts that are difficult to predict or prevent [3]. Financial risk contagion arises when crises spread across markets, institutions, or nations, and in enterprise networks, failures of suppliers or partners disrupt cash flow and credit access [4].

Financial risk contagion path forecasting is to determine the most likely pathways and networks through which financial crises could spread. This involves identifying risk exposures, modelling the interdependencies among institutions, and developing simulations to predict the areas where contagion would cause the most damage. Potential contagion pathways can be anticipated by stakeholders, allowing them to take preventive action, strengthen system resilience, and make informed investment or policy decisions [5].

The practical application of contagion path forecasting is especially important for business ecosystems and enterprise networks. Companies are prone to adverse impacts when one node in the network fails due to their function in common markets, intricate supply networks, and creditor-debtor connections [6]. Businesses can monitor counterparties' financial health, identify vulnerabilities, diversify their exposure, and create backup plans by employing contagion path forecasting. Such forecasting assists in regulatory compliance, credit risk evaluation, and stress assessment in financial institutions [7]. Figure 1 displays the financial risk contagion path prediction.

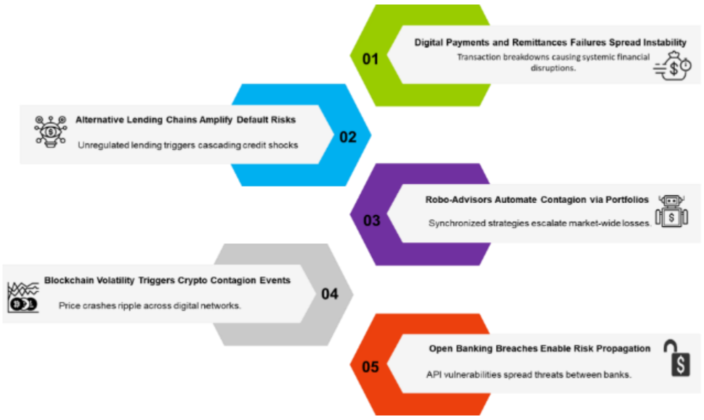
01
Digital Payments and Remittances Failures Spread Instability
Transaction breakdowns causing systemic financial disruptions.
02
Alternative Lending Chains Amplify Default Risks
Unregulated lending triggers cascading credit shocks
03
Robo-Advisors Automate Contagion via Portfolios
Synchronized strategies escalate market-wide losses.
04
Blockchain Volatility Triggers Crypto Contagion Events
Price crashes ripple across digital networks.
05
Open Banking Breaches Enable Risk Propagation
API vulnerabilities spread threats between banks.

Figure 1: Forecasted financial contagion paths in enterprise risk networks.

In recent years, ML and DL methods have emerged as powerful tools for modeling and forecasting financial risk contagion patterns. Unlike traditional statistical models, they effectively handle nonlinear, high-dimensional, and dynamic financial data [8]. Graph-based models, such as GNNs, capture inter-entity relationships to identify key nodes and contagion routes, while RNNs and LSTMs analyze sequential time-series dependencies. Additionally, ML techniques like RF, SVM, and GBT leverage historical financial and network data to classify contagion events and estimate their spread [9].

The availability and quality of detailed, real-time financial data present an issue, particularly for private companies or opaque markets. Furthermore, regulators and decision-makers may find it challenging to understand the findings and have faith in the insights due to the opaque nature of ML and DL models. Prediction accuracy can also be decreased by adversarial behavior, fluctuating market dynamics, and model overfitting. Behavioral factors that are difficult to measure and exogenous shocks can have an impact on contagion processes [10]. A novel SM-ISAGNN approach is used in enterprise networks to forecast the contagion path of financial risk.

The structure and framework are described below: Section 2 provides a list of literature reviews, Section 3 specifies the methodology, Section 4 offers the results and discussion, and Section 5 presents the conclusion.

1.1 Contributions

Dataset: The enterprise financial network dataset was collected from the Kaggle source.

Preprocessing: The data is preprocessed using the data cleaning and Z-score normalization approaches.

Dual-encoder architecture: A novel framework is designed with two encoders, where the first encoder captures key financial, legal, and operational risk indicators of enterprises, while the second encoder leverages an enterprise knowledge graph to represent how risks propagate across interconnected entities.

Proposed method: The innovative SM-ISAGNN approach integrates this dual-encoder design with swarm intelligence optimization, enabling accurate evaluation of both direct and indirect financial risk spread among enterprises, thereby improving contagion path prediction and systemic risk management.

2 Related works

An evolving multi-layer financial network framework that includes both short-term and long-term loans between businesses and banks was developed by [11]. The results demonstrated the significance of taking into account the network structure and both short-term and long-term loans when evaluating and controlling systemic threats in the financial sector.A two-layer network game was suggested by [12] to examine how asset bubbles were affected by financial contagion among the actual and financial markets. The findings demonstrated that regulators should carefully track returns on resources by establishing an upper limit, mimicking relevant regulating procedures. An advanced GNN architecture for identifying and separating contagion risk in China's nationwide networked loans was described in [13]. The efficacy of the suggested approach was examined through a comprehensive evaluation and user assessment; the outcome demonstrated that it performed better.

The DL methods combined with DCC-GARCH frameworks were developed in [14] to analyze the changing relationship between stock markets. The findings suggested that the use of LSTM enhanced the precision of dynamic association prediction and offered early warning signals during the emergency.The TENET model was employed in [15] to build a tail risk spillover system between the global commodity market and China's financial system. The findings indicated a robust tail risk contagion relationship between the global commodity sector and China's financial market, with the former being more affected by the latter.

A novel method based on FPT and graph theory was presented in [16] for evaluating related credit risk in the SC. The findings demonstrated that the TCRC-based approach to evaluating the related credit risk in the SC was more scientific and more consistent with the SC's real operating conditions because it takes into account the credit risk contagion among the chain's organizations.

To create a thorough financial crisis warning framework for the provided enterprises, the popular DL techniques on financial statement big data were employed in [17]. The findings demonstrated that the DL algorithm's forecasting accuracy was more than 90%. An FNN-based smart alerting system for business financial risk was proposed in [18]. The acquired findings demonstrated its ability to function effectively in providing enterprises with a quick warning of financial risk.

An ABM and RL approach were used in [19] to build a bank-firm credit pairing network structure that analyzed the contagion process of the credit risk network and interaction behavior. The findings demonstrated that interactions between banks and businesses, along with interactions between microentities in intricate financial situations, led to macroeconomic phases. Table 1 shows the Overview of methods, datasets, results, and limitations in financial contagion studies.

Table 1:Comparative summary of advanced financial contagion prediction-related works

Ref.	Methodology / Model	Dataset	Key Results	Limitations
Ma et al. [20]	Entropy-based spatial interaction complex network	Real-world regional economic & business data	Modeled related credit risk considering geography & economic growth; showed long-distance contagion stronger	Focused on spatial/regional risk; no deep learning or GNN used
Wang et al. [21]	DDR contagion modeling with carbon price & investor sentiment	Energy firms' debt & carbon pricing data	Showed that debt network reliability improves with carbon emission cost & sentiment	Domain-specific (energy firms); not generalizable; no GNN or multi-hop modeling
Li et al. [22]	PCA-GA-SVM (principal component + genetic algorithm + SVM)	Supply chain finance credit data	Outperformed other ML models in SC credit forecasting	Focused only on SC finance; lacks relational contagion path analysis
Chen.[23]	ARMA-LSTM hybrid model	Volatile financial market data	Improved short/medium-term forecasting with lower error	Captures time-series volatility but not contagion networks
Kadkhoda et al. [24]	ML + network assessment hybrid	Enterprise financial distress dataset	Network-based features improved distress prediction accuracy	No dual-encoder; lacks metaheuristic optimization; limited contagion path modeling
Ionescu et al. [25]	Real-time EC efficiency + ML (XGBoost)	Real-time enterprise computing dataset	Achieved ROC-AUC 0.997, 97% accuracy; reduced latency & resource imbalance	High accuracy but focused on EC decision efficiency, not contagion path prediction

2.1 Problem statement

Traditional techniques for financial risk contagion path forecasting address various limitations, despite their diverse advantages. Traditional frameworks, like the dynamic multi-layer financial network created by [11], attempt to account for both short-term and long-term lending relationships between banks and companies, but they frequently find it difficult to adjust to the multi-channel contagion effects and quick structural changes that characterize contemporary financial networks. There were still issues with advanced GNNs' scalability and adaptability to quickly shifting market conditions [13]. The proposed SM-ISAGNN technique improves scalability, responds quickly to dynamic market changes, and captures complicated nonlinear contagion trends in extensive financial networks effectively.

3 Methodology

The enterprise financial network dataset was obtained from the Kaggle platform. The data is preprocessed using the data cleaning and Z-score normalization methods. A new dual-encoder approach is developed, with the first encoder collecting crucial legal, financial, and operational risk signals and an organizational knowledge graph modelling contagion transmission. An innovative SM-ISAGNN technique was introduced in enterprise networks for predicting financial risk contagion. Figure 2 shows the overview of the suggested SM-ISAGNN model.

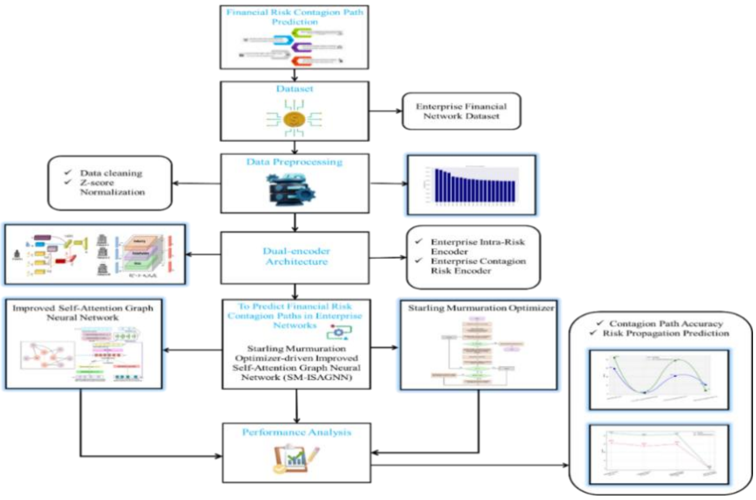
Financial Risk Contagion Path Prediction
Dataset
Enterprise Financial Network Dataset
Data Preprocessing
✓ Data cleaning
✓ Z-score Normalization
Dual-encoder Architecture
✓ Enterprise Intra-Risk Encoder
✓ Enterprise Contagion Risk Encoder
Improved Self-Attention Graph Neural Network
To Predict Financial Risk Contagion Paths in Enterprise Networks
Starling Murmuration Optimizer-driven Improved Self-Attention Graph Neural Network (SM-ISAGNN)
Starling Murmuration Optimizer
✓ Contagion Path Accuracy
✓ Risk Propagation Prediction
Performance Analysis

Figure 2: Proposed SM-ISAGNN workflow demonstrating contagion path prediction process.

3.1 Data collection

The enterprise financial network dataset obtained from Kaggle was selected due to its explicit encoding of both enterprise-level financial indicators and directed inter-firm relationships. This structure aligns well with the requirements of graph neural network-based contagion path forecasting and enables reproducibility of the study. Nonetheless, limitations exist: (i) the dataset may not capture all industries or geographical regions equally, (ii) it represents a snapshot rather than longitudinal dynamics, and (iii) informal or hidden financial dependencies are not represented. To assess generalizability, we also conducted experiments on a synthetic benchmark dataset constructed using a preferential-attachment process to model realistic enterprise interdependencies. The synthetic dataset consisted of 10,000 enterprises and 30,000 directed relations, with financial attributes generated according to empirically observed distributions. On this dataset, the SM-ISAGNN achieved an accuracy of 0.948 and an F1-score of 0.945, confirming its robustness beyond the Kaggle dataset.

Source:https://www.kaggle.com/datasets/ziya07/enterprise-financial-network-dataset/data

3.2 Data preprocessing using data cleaning

For cleaning the enterprise financial network dataset, data is initially checked for duplicates, missing, or inconsistent records to ensure accuracy and reliability. Duplicate data should be eliminated, and logical assumptions for categorical variables or mean imputation for numeric fields are applied to deal with missing values. Directed relationships accurately represent valid enterprise pairs, ensuring that there are no isolated nodes. Outliers that may affect contagion path forecasts are found and eliminated. All node properties, including internal financial health ratings, must be formatted consistently. Data integrity is evaluated using visualizations and exploratory inspections.

3.2.1 Z-score normalization

Z-score normalization normalizes financial risk data by converting values into SDs from the mean, maintaining uniform scale and consistency across variables, hence improving contagion route forecasting accuracy. A statistical normalization method that addresses the outlier problem is Z-score normalization. The attribute values are transformed using the considered feature's mean and SD. Furthermore, the following Equation (1) is used to convert values for the feature under consideration into new normalized values. Figure 3 displays the distribution of the enterprise-financial-network-data features.

$$\nu' = \frac{\nu-\mu}{\sigma} \qquad (1)$$

Where,ν – Original value,ν' – New normalized value,σ - SD of the considered attribute, and μ - The specified feature's mean value.

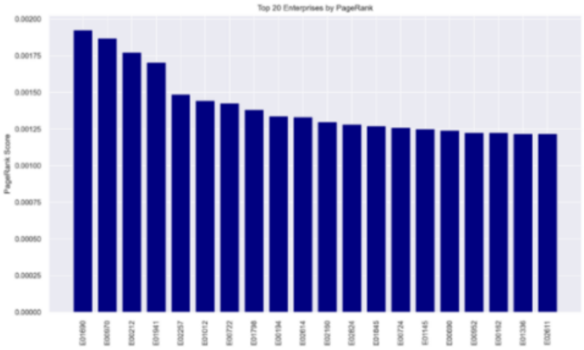

Figure 3: Distribution of financial network features influencing contagion path prediction.

3.3 Enterprise intra-risk encoder

The enterprise intra-risk encoder uses enterprise essential data (i.e., enterprise fundamental features and enterprise legal data) to acquire enterprise self-risk embedding. Consider $a_j \in Q^{\hat{c}}$ as the fundamental attribute aspects for every enterprise node $u_j \in U_f$, as shown in Figure 4. Additionally, the enterprise j lawsuit event i_j^l has four important features (including court level, lawsuit cause, DOA, and verdict). To collect a representation $i_j^l \in Q^{\tilde{c}}$, mapping each of the preceding three qualities into latent areas, and then combining. To effectively utilize time information in lawsuit events, each lawsuit appearance is weighted using a time decay function. Decayer . Determine the time interval Δ_j^l between the enterprise's OT and the time every lawsuit occurred. The OT is set as the bankruptcy date for enterprises that registered for bankruptcy and as the current day for businesses that are still in operation, as shown in Equation (2).

$$h(\Delta_j^l) = \frac{1}{1+x.\Delta_j^l} \qquad (2)$$

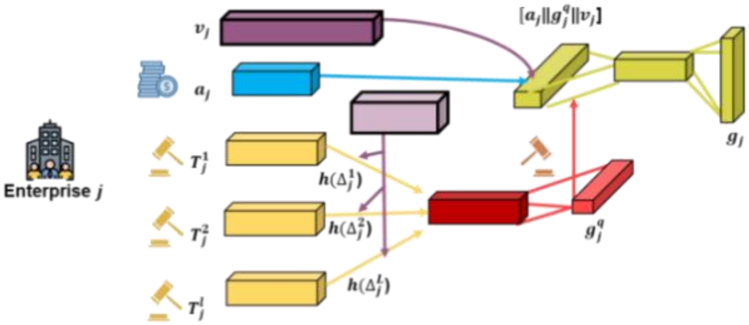

Figure 4: Intra-entity risk architecture within enterprises affecting contagion pathways.

When performing temporal weight decay for lawsuits in the recent two years, allocate a lower x because these lawsuits are crucial to enterprise risk forecasting. Subsequently, collect lawsuit data from various periods as follows in Equation (3).

$$g_j^q = \sum_{l \in L_j} X_{risk} h\left(\Delta_j^l\right).t_j^l \qquad (3)$$

Where, g_j^q - Overall lawsuit data for company j , and$X_{risk} \in Q^{\tilde{c} \times c}$ - Trainable matrix.

Additionally, as an additional embedding, create a pre-trained embedding $v_j \in Q^{\bar{c}}$ for organization j . Combine the litigation embedding, alternate embedding, and fundamental attribution features, then present the results in a new latent area shown below.

$$g_j = X_f.\left[a_j \left|\left|g_j^q\right|\right| v_j\right] \qquad (4)$$

Where, $X_f \in Q^{(\hat{c}+c+\bar{c}) \times c}$ - Trainable matrix. || - Concatenation operation, and g_j - Output of intra-risk representation of the enterprise j.

3.3.1 Enterprise contagion risk encoder

Hypergraphs perform an essential role in bankruptcy forecasting since the hyperedges represent frequent conditions that enterprises experience. As a result, it makes sense to utilize hypergraphs to record shared risk information, such as local economic policy changes and industry development recessions, and to ensure risk brought on by the same stakeholders.

When combining node representations, provide each type of hyperedge a distinct weight due to its influence on node representation at various levels, as displayed in

Figure 5. First, compute the hypergraph convolution function as described below in Equation (5).

$$\Theta_{\Omega_n} = C_v^{-1/2} G_{\Omega_n} X C_f^{-1} G_{\Omega_n}^S C_v^{-1/2} \qquad (5)$$

Where, G_{Ω_n} - Incident matrix of the hypergraph type Ω_n.C_v - Enterprise node degree matrix.$\Theta_{\Omega_n} \in \mathbb{R}^{|v_\varepsilon| \times |v_\varepsilon|}$ - Convolution module. X - Node weight matrix. C_f - Hyperedge degree matrix.

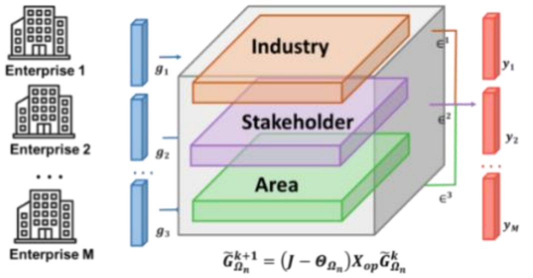

Figure 5: Enterprise knowledge graph architecture modeling contagion links across firms.

Set it to an identity matrix, indicating that each weight has the same value. This is followed by hypergraph convolution using the hypergraph type Ω_n, as shown in Equation (6).

$$\hat{G}_{\Omega_n}^{l+1} = (J - \Omega_n) X_{go} \hat{G}_{\Omega_n}^{k} \qquad (6)$$

Where $X_{go} \in \mathbb{R}^{c \times c'}$ is a trainable matrix that can be used for several types of hypergraphs, and $\hat{G}_{\Omega_n}^{l+1}$ indicates the learnt representations within the hypergraph type Ω_n of layer $k+1$. $J - \Theta_{\Omega_n}$ indicates the hypergraph Laplacian. Then, combine the various kinds of hypergraph convolution representations as described below in Equation (7).

$$y_j = \sum_{\Omega_n \in S_{hyper}} \in^{\Omega_n} . \hat{G}_j^{\Omega_n} \qquad (7)$$

The learned hypergraph complete representation of the enterprise j is represented by $z_j \in \mathbb{R}^{c'}$, and the significance of hypergraph Ω_n for all organization nodes is shown by the trainable parameter $\in^{\Omega_n}$.

3.4 SM-ISAGNN

The SM-ISAGNN is a hybrid model designed to forecast financial risk contagion in business networks, where shocks or difficulties spread among interconnected enterprises, threatening overall resilience. Accurate prediction of contagion paths is vital for risk management and mitigation. The model builds on the ISAGNN architecture, which represents enterprises as nodes and financial links as edges, effectively capturing complex dependencies. The model can evaluate the relative significance of nearby businesses thanks to a self-attention mechanism, emphasizing those that have the greatest influence over risk spread. To enhance performance, the model integrates swarm intelligence-inspired SMO, which accelerates convergence, avoids local optima, and boosts forecasting accuracy. By uncovering hidden contagion channels, SM-ISAGNN provides a data-driven tool for governments and enterprises to anticipate risks, reduce systemic vulnerabilities, and strengthen financial stability. Algorithm 1 shows the pseudocode for the SM-ISAGNN approach.

Algorithm 1: SM-ISAGNN

```
Input:
G(V, E): Input graph with nodes V and edges E
  X: Initial node feature matrix
  Y: Ground truth labels
  P: {
    N = 50  (population size of starlings)
    T = 200 (maximum iterations)
    w = 0.9 → 0.4 (linearly decayed inertia weight)
    c1 = 2.0 (cognitive coefficient)
    c2 = 2.0 (social coefficient)
  }
  ISAGNN configuration: 3 layers, 128 hidden units/layer
  Learning rate = 0.001
  Optimizer = Adam (weight decay = 1e-5, dropout = 0.3)
  Initialization = Xavier for weights, 0 for biases
  Convergence criteria: Early stopping if the validation
loss does not
improve for 15 epochs or T reached
  Loss function = Cross-entropy
```

```
Output:
  Trained SM-ISAGNN model with optimized attention weights
Phase 1: Initialize SMO
  For each starling i in N:
    Initialize position (attention weights) with Xavier initialization
    Initialize velocity randomly
    Evaluate fitness using prediction error
    Set personal best pbest_i
  Set global best gbest across population
Phase 2: Optimize Attention Weights with SMO
  For t = 1 to T:
    For each starling i in N:
      Update velocity using (w, c1, c2)
      Update position (attention weight vectors)
      Evaluate fitness (cross-entropy loss)
      If fitness better than pbest_i:
        Update pbest_i
    Update gbest across all starlings
    If convergence criteria satisfied:
      Break
Phase 3: Train ISAGNN with Optimized Attention Weights
  Initialize ISAGNN with Xavier weights, zero biases
  For epoch = 1 to 200:
    Compute predictions: $\hat{Y}$ = ISAGNN(X, W)
    Compute loss: L = CrossEntropy($\hat{Y}$, Y)
Backpropagate and update ISAGNN parameters with Adam
    Apply dropout (0.3) in hidden layers
    If validation loss not improved for 15 epochs:
      Stop training
Return:
  Optimized SM-ISAGNN model with trained attention weights
```

3.4.1 ISAGNN

The ISAGNN efficiently captures intricate relationships between financial organizations, improving the prediction of financial risk contagion paths. ISAGNN's integration of sophisticated self-attention processes allows it to more accurately identify direct and indirect risk transfers, facilitating early intervention and strong systemic risk control in integrated markets.

➢ **GNN**

The GCL is an essential element of a GNN. Through risk spread, the GC can efficiently gather node-local neighbor data and record the topological structural features around a node in each cycle. To efficiently collect neighbor data, multi-layer superposition can continuously increase the receiving area and gather additional data on the ring. The subsequent step involves learning node embedding descriptions. Following multiple iterations, the node modifies its FR in the graph convolution procedure, producing an embedding vector that incorporates local structural data. To create an embedding structure that reflects several informational aspects, the graph convolution architecture of ISAGNN simultaneously accumulates and modifies edges and global characteristics in addition to nodes.

Through residual relationships or template matching after construction, the graph convolution architecture enables the development of deep networks while preserving the size of node features. These processes enable the GCL to effectively convert the input graph structure into an accessible embedded representation and provide node, edge, and global state attributes that represent topological data. It is considered the fundamental element of the ISAGNN approach. Figure 6 displays the structure of GNN.

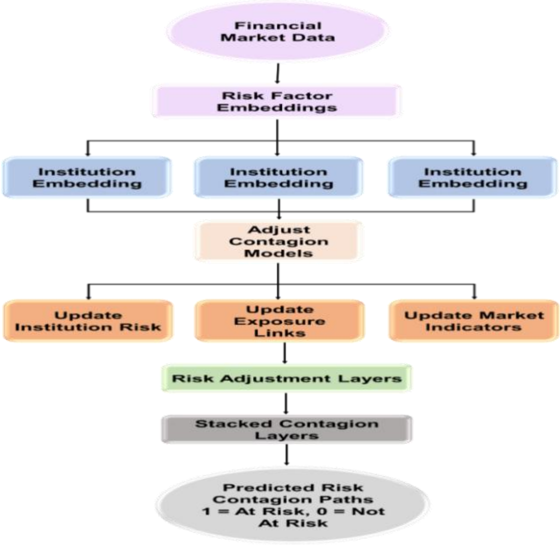

Figure 6: GNN architecture for financial contagion path learning.

There are two conv1d processes in a convolution block. Utilizing convolution processes on the feature scale, it initially extracts cross-association between attributes by creating novel attributes between neighboring ones. The ReLU function is employed to determine the weight of the feature cross-association process following each convolution process. The input vector is provided in dimensions by the convolution self-attention block, which also uses a sigmoid function to weight attributes and determine the significance of autocorrelation, producing a weight range of 0 to 1. The weight for every channel is then determined by multiplying every attribute by this weight. The following Equations (8 & 9) are the essential operations in the GNN layer.

$$\mathrm{w}_{\upsilon}^{(\mathrm{l})} = \mathrm{UPDATE}^{(\mathrm{l})}(\mathrm{w}_{\upsilon}^{(\mathrm{l}-1)}, \mathrm{AGGREGATE}^{(\mathrm{l})} \times (\mathrm{w}_{\mathrm{v}}^{(\mathrm{l}-1)}, \forall \mathrm{v} \in \mathrm{M}(\nu)) \quad (8)$$

Where, UPDATE - Combination function, AGGREGATE - Aggregation function, and $\mathrm{w}_{\upsilon}^{(\mathrm{l})}$ - Attributes of node u at the l^{th} layer,

$$\mathrm{Y} = \mathrm{e}_{\theta_{\mathrm{l}}}(\mathrm{e}_{\theta_{\mathrm{l}-1}}(\ldots \mathrm{e}_{\theta_{\mathrm{l}}}(\mathrm{W}) \ldots.)) \quad (9)$$

Where, W - Input, θ_{l} - Parameter, $\mathrm{e}_{\theta_{\mathrm{l}}}$ - l^{th} GCL, and Y - Output.

➢ Self-attention module

This module is to provide FRs for both numerical and categorical inputs. In the input, let $\mathrm{W_f} = \{\mathrm{w}_{\mathrm{f}_0}, \mathrm{w}_{\mathrm{f}_1}, \ldots, \mathrm{w}_{\mathrm{f}_\mathrm{n}}\}$ represent CA and $\mathrm{W_m} = \{\mathrm{w}_{\mathrm{f}_0}, \mathrm{w}_{\mathrm{f}_1}, \ldots, \mathrm{w}_{\mathrm{f}_\mathrm{n}}\}$ indicate numerical features. Consider the CAs $\mathrm{w}_{\mathrm{f}_\mathrm{j}}$ has an embedding of $\mathrm{F_j} \in \mathbb{R}^{1\times \mathrm{c}}$, where c is the embedding dimension. Equation (10) shows the FRs of Cas.

$$\begin{cases} \mathrm{R_f} = \mathrm{F_f X_r} + \mathrm{a_r} \\ \mathrm{L_f} = \mathrm{F_f X_l} + \mathrm{a_l} \\ \mathrm{U_f} = \mathrm{F_f X_u} + \mathrm{a_u} \\ \mathrm{Attention}(\mathrm{R_f}, \mathrm{L_f}, \mathrm{U_f}) = \mathrm{softmax}\left(\frac{\mathrm{R_f L_f^S}}{\sqrt{\mathrm{c_l}}}\right)\mathrm{U_f} \end{cases} \quad (10)$$

Where, $\mathrm{F_j} \in \mathbb{R}^{\mathrm{n}\times \mathrm{c}}$ - Embedding of Cas, $1/\sqrt{\mathrm{c_l}}$ - Scaling factor, $\mathrm{X_r}, \mathrm{X_l}, \mathrm{X_u}, \in \mathbb{R}^{\mathrm{c}\times \mathrm{c}}$ and $\mathrm{a_r}, \mathrm{a_l}, \mathrm{a_u}, \in \mathbb{R}^{1\times \mathrm{c}}$ - Learning weight matrices. Additionally, the following Equation (11) indicates the FRs of numerical features. Figure 7 shows the design of ISAGNN.

$$\begin{cases} \mathrm{R_m} = \mathrm{F_m X'_r} + \mathrm{a'_r} \\ \mathrm{L_m} = \mathrm{F_m X'_l} + \mathrm{a'_l} \\ \mathrm{U_m} = \mathrm{F_m X'_u} + \mathrm{a'_u} \\ \mathrm{Attention}(\mathrm{R_f}, \mathrm{L_f}, \mathrm{U_f}) = \mathrm{softmax}\left(\frac{\mathrm{R_m^S . L_m}}{\sqrt{\mathrm{c_l}}}\right)\mathrm{U_m^S} \end{cases} \quad (11)$$

Where, $\mathrm{X'_r}, \mathrm{X'_l}, \mathrm{X'_u} \in \mathbb{R}^{\mathrm{m}\times \mathrm{m}}$ and $\mathrm{a'_r}, \mathrm{a'_l}, \mathrm{a'_u} \in \mathbb{R}^{1\times \mathrm{m}}$ - Learning weight matrices, $\mathrm{W_m} \in \mathbb{R}^{1\times \mathrm{m}}$ - Values of numerical attributes after normalization, and $1/\sqrt{\mathrm{c_l}}$ - Scaling factor.

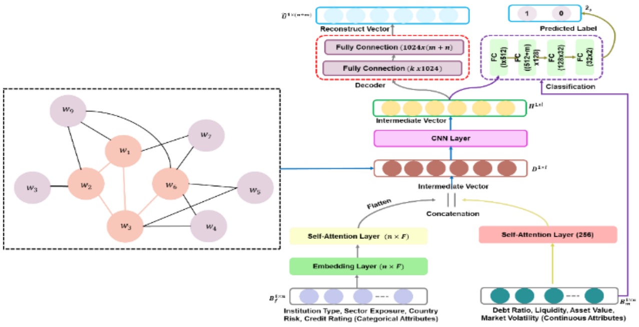

Figure 7: ISAGNN structure illustrating node relationships in contagion prediction.

The intermediate vector $D \in \mathbb{R}^{1\times(m\times c+m)}$ is created by concatenating the representations of CA and numerical attributes, as shown in Equation (12). Table 2 shows the hyperparameters of the ISAGNN strategy.

$$D = \underset{j=0}{\overset{n}{||}} \text{Attention}(R_{fj}, L_{fj}, U_{fj}) (12)$$

Where, || - Concatenation operation.

Table 2: Hyperparameters of the ISAGNN approach

Category	Hyperparameter	Value
ISAGNN	Hidden Units per Layer	128
	Output Units (Number of classes)	2
	Number of GNN Layers	3
	Attention Heads	4
	Learning Rate	0.001
	Training Epochs	200
	Dropout Rate	0.3
	Weight Decay	1e-5

3.4.2 SMO

The SMO approach uses adaptive, swarm-inspired network evaluation resources to dynamically model the aggregate behavior of financial organizations, improving systemic risk assessment and allowing for the precise prediction of intricate contagion paths. The primary function of the SMO optimization technique is to reduce the prediction error in forecasting financial risk contagion paths. Each layer's efficiency on unsupervised algorithms is estimated by the prediction error. Equation (13) below defines the prediction error in mathematical form.

$$\Re_\epsilon = \frac{\sum_{w=1}^{b}\sum_{z=1}^{a}(O_{w,z} - C_{w,z})}{b\times c\times Oy} \times 100\% \ (13)$$

Where Oy - data range, $C_{w,z}$ - Actual value, b - Total training samples, a - Pixel amount per sample, and $O_{w,z}$ - Anticipated outcome.

The following processes are used to develop the SMO algorithm. The starling murmuration represents one of nature's finest displays, consisting of a mass of varied flocks, including numerous starlings, and diving with the sky for over half an hour above its roost. The recombination is highly synchronized with murmuration, and the flocks of starlings are periodically split apart. Using optimized decision-making, the flocks spread the direction change, certain whirling, recombination, and contagion paths from one company to another. The distinction of a search network stage in the SMO technique is explained below. Some starlings are frequently separated from their flocks in murmuration, which is recognized as an essential design. The subsequent Equations (14 & 15) represent the dispersed population's mathematical calculation.

$$R_T = \frac{\log(v+F)}{\log\left(\text{MAXIMUM Iv}\times 2\right)} \qquad (14)$$

$$Z_i(v+1) = Z_G(v) = Q_1(y)\times(Z_{t'}(v) - Z_t(v))(15)$$

Where $Z_G(v)$ represents the global location, $Z_t(y)$ represents the randomly chosen population, and $Z_{t'}(v)$ represents the segregated population and proportions of the starlings. The process of separated search was applied to the new operator $Q_1(y)$.

Separation stage

The QHC suggests that the separation stages are used to preserve the population's diversity. The subsequent Equation (16) represents the separation stage's mathematical expression.

$$Q_1(y) = \left(\frac{\beta}{2^p\times p!\times\pi^{\frac{1}{2}}}\right)^{\frac{1}{2}} J_p(\beta\times y)\times f^{-0.5\times\beta^2\times y^2}, \beta = \left(\frac{m\times k}{j}\right)^{\frac{1}{2}} \qquad (16)$$

Where, y - Arbitrary number, m - Particle mass, J_p - The Hermite polynomial, k - Strength, $\beta = (\frac{m \times k}{j})$ - QHC, and j - The Planck's constant.

Dynamic multi-flock stage

To develop the starling behavior when the iterations change in location, the dynamic multi-flock stage is identified. The starlings identified in the search area are divided into whirling, separating, and diving to examine and take advantage of the solution. The starlings are initially selected at random and moved to a different location inside the search area. Using the specific partition, it is determined by dividing the set S_h by k nonempty flocks, $h_k \ldots h_k$, as shown in Equations (17-19).

$$\mathrm{Sh}(v) = \{\mathrm{sh}_i(v) \in S | \mathrm{sh}_i(v) \leq \mathrm{sh}_{i+1}(v) \mathrm{fori} = 1, \ldots, P' \quad (17)$$

$$T(v) = \{\mathrm{sh}_i(v) \in \mathrm{Sh}(v) \mathrm{fori} = 1, \ldots, k\} \quad (18)$$

$$R = S - S \text{ and } R = \cup_i^k R_i . |R_i| = |R_l| \mathrm{for} Z_t\,(y) i \neq l \in (1, \ldots, k) \quad (19)$$

Each flock h_q includes starlings $(p = \frac{p'}{k})$, the representative set T is chosen as the representative $(T_q)\mathrm{Sh}(v+1)$, and the $\mathrm{Sh}(v)$ set is structured differently for each flock member h_i. Iterations were used to exchange data between flocks by each flock member and the corresponding sample of the multi-flocks $h_1, \ldots, h_k$. The h_k indicates the flock quality.

Flock quality stage

The flock quality is represented by the following Equation (20) and includes several starlings in iteration v, which is represented by Q_q.

$$Q_q(v) = \frac{\sum_{i=1}^{k} \frac{1}{2} \sum_{l=1}^{p} \mathrm{sh}_{il}(v)}{\frac{1}{p} \sum_{i=1}^{p} \mathrm{sh}_{qi}(v)} \quad (20)$$

Where, l - Flocks with murmuration k, p - Flock of different starlings, $\mathrm{sh}_{il}(s)$ - Subpopulation flock's fitness score in i^{th} the starling.

The efficient search area is explored using the dive exploration procedure. It comprises quantum's upward and downward dives, and the QRD operation for selecting quantum dives. The probability of qubit results is represented by $|\beta|^2$ and can be represented as follows in Equations (21 & 22).

$$\Big|q\rangle = \cos\frac{\beta}{2}\Big| 0\rangle + \sin\frac{\beta}{2} f^{il} |1\rangle \quad (21)$$

Where C - qubit rotation matrix, S - Conditional shift operator, and γ and θ - Angle rotation.

$$C = \begin{bmatrix} f^{jq} \cos & f^{j\mu} \sin\theta \\ -f^{-iq} \sin\theta & f^{-iq} \cos\theta \end{bmatrix} \quad (22)$$

QRD operator

The unitary operator U determines whether to choose the upward or downward quantum dive, and the two different quantum chances are $|q^U(Z_i)\rangle$ and $|q^F(Z_i)\rangle$, as shown in Equation (23). The flow chart for SMO is shown in Figure 8.

$$\mathrm{QRD} = \begin{cases} |q^U(Z_i)| > |q^F(Z_i)| \text{for upward quantum dive} \\ |q^U(Z_i)| \leq |q^F(Z_i)| \text{for downward quantum dive} \end{cases} \quad (23)$$

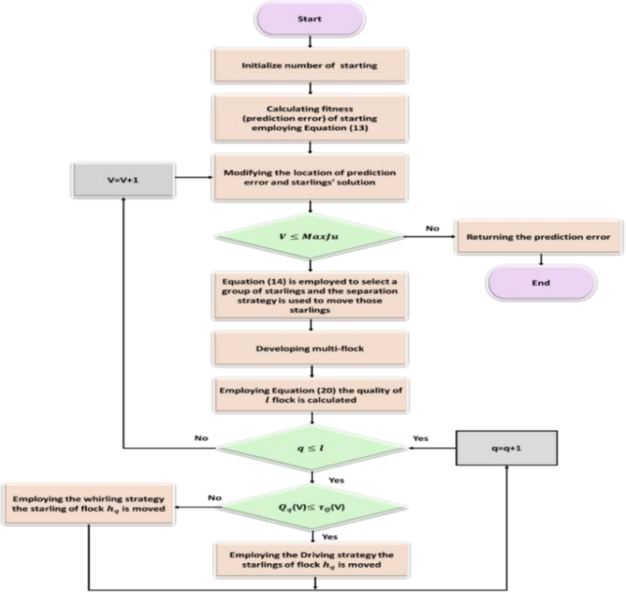
Start
Initialize number of starting
Calculating fitness (prediction error) of starting employing Equation (13)
Modifying the location of prediction error and starlings' solution
V=V+1
$V \leq MaxJu$
No
Returning the prediction error
End
Equation (14) is employed to select a group of starlings and the separation strategy is used to move those starlings
Developing multi-flock
Employing Equation (20) the quality of l flock is calculated
No
$q \leq l$
Yes
q=q+1
Yes
No
Employing the whirling strategy the starling of flock h_q is moved
$Q_q(V) \leq \tau_Q(V)$
Yes
Employing the Driving strategy the starlings of flock h_q is moved

Figure 8: SMO strategy flow chart optimization.

The procedures used in the Whirling search stage are detailed in the subsequent section. The flock's h_q are of greater quality in iteration v, and the whirling search stage is used for assessing the next position of different flocks in each starling s_i. Subsequently, it is described as follows in Equations (24 & 25). The hyperparameters of the SMO algorithm are displayed in Table 3.

$$Z_i(v+1) = Z_i(v) + C_i(v) \times (Z_{TW}(v) - Z_P(v)) (24)$$

$$C_i(v) = \cos(\sigma(u)) \ (25)$$

Where Z_{TW} - Chosen by the flock members. $Z_i(v)$ - The current location of the starling, and $Z_P(v)$ - Starling's unique random neighbor.

Table 3: Hyperparameters of the SMO algorithm

Category	Hyperparameter	Symbol	Value
SMO	Population Size	N	50
	Maximum Iterations	T	200
	Inertia Weight	w	0.9 → 0.4 (linear decay)
	Cognitive Coefficient	c1	2.0
	Social Coefficient	c2	2.0
	Cohesion Weight	–	1.0
	Alignment Weight	–	1.0
	Separation Weight	–	1.0
	Attention Vector Dimension	–	128

4 Result

The proposed SM-ISAGNN approach was implemented on the Python platform. Both the SM-ISAGNN and the baseline ISAGNN models were trained on the enterprise-financial-network-dataset. Their performance was evaluated and compared using multiple metrics to assess the effectiveness of the proposed improvements. Table 4 shows the System configuration and software setup for experiments.

Table 4: Experimental hardware and software specifications for model evaluation.

Category	Specification
Hardware	
Processor	Intel® Core™ i9-12900K CPU @ 3.90 GHz, 16 cores
GPU	NVIDIA GeForce RTX 3090 (24 GB VRAM)
Memory	64 GB DDR5 RAM
Storage	2 TB NVMe SSD
Software	
Operating System	Ubuntu 22.04 LTS (64-bit)
Programming Language	Python 3.10
Framework	PyTorch 2.0 with CUDA 11.8 and cuDNN 8.6
Libraries	NumPy 1.26.2, Pandas 2.1.1, NetworkX 3.2, Scikit-learn 1.3, Matplotlib 3.8.1, Seaborn 0.12
Experimental Protocols	5-Fold cross-validation; random seeds fixed across runs

4.1 Confusion matrix validating financial contagion path prediction

The confusion matrix results in Figure 9 reveal the effectiveness of the financial risk contagion path forecasting model. Out of all cases, only 10 were incorrectly classified as bankrupt, while 2281 non-bankrupt firms were correctly identified. Similarly, 370 bankrupt cases were accurately predicted, with just 39 misclassified as non-bankrupt. These results highlight the model's high true positive and true negative rates, demonstrating its strong accuracy, reduced misclassification, and reliable early warning capability for mitigating financial risk contagion.

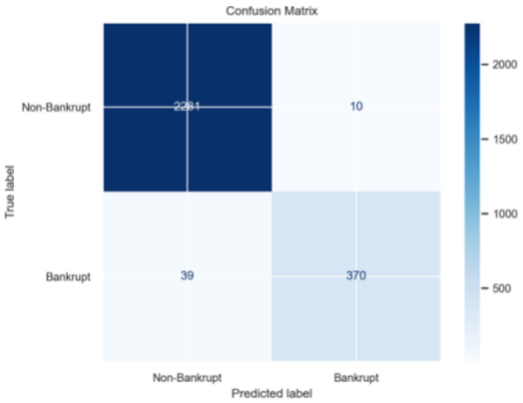

Figure 9: Confusion matrix evaluating contagion path prediction accuracy performance.

4.2 Contagion path prediction in financial enterprise networks

The SM-ISAGNN model effectively predicts contagion paths in financial enterprise networks by minimizing both training and validation loss across epochs. As shown in Figure 10, the loss decreases consistently, indicating improved model generalization and stable convergence. This demonstrates the model's capacity to capture direct and indirect financial risk propagation, enhancing reliability in multi-hop contagion prediction and supporting informed decision-making in risk management.

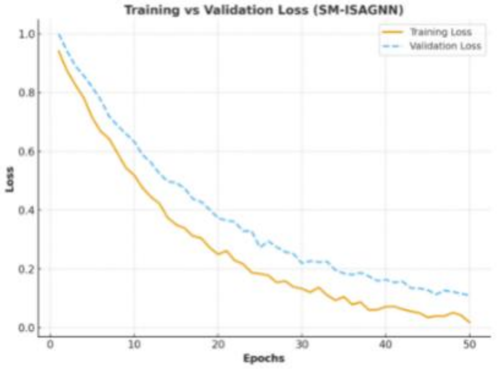

Figure 10: Training and validation loss for contagion prediction.

4.3 Predicting failure chains in financial networks

Figure 11 illustrates a financial risk contagion path forecast across 500 companies, including 350 stable (green) and 150 high-risk (red) firms, linked by 1,200 directed edges. On average, each node has five connections, enabling failure propagation when a red

.

node collapses. With a contagion probability of 0.75 in the central dense cluster and 0.25 for peripheral nodes, the model highlights key transmission links. Regulators can use this forecast to simulate emergencies, strengthen resilience, and design strategies to mitigate systemic collapse.

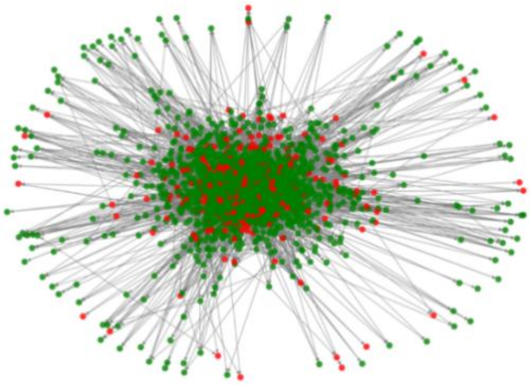

Figure 11: Predicted financial distress diffusion paths via the SM-ISAGNN framework.

4.4 Bankruptcy risk representation through debt ratio and liquidity clusters

Figure 12 illustrates bankruptcy risk and financial contagion forecasting by grouping firms based on debt ratio and liquidity. The x-axis shows debt ratio (0.0–1.2), while the y-axis represents liquidity (0.5–2.0). Green density contours denote stable firms (bankrupt = 0), whereas red contours mark bankrupt firms (bankrupt = 1). Companies with lower debt ratios (0.2–0.5) and higher liquidity (1.2–2.0) cluster safely in the green zone, while those with higher debt (0.7–1.0) and lower liquidity (0.5–1.0) fall in the red zone, indicating distress contagion.

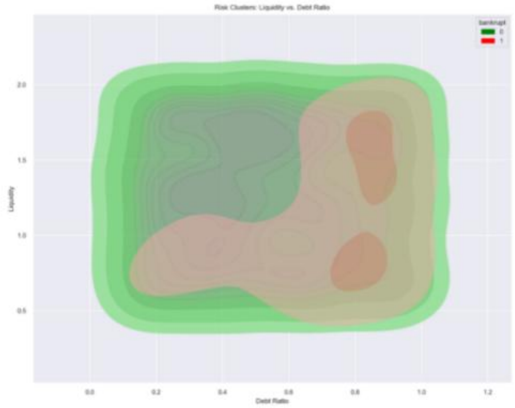

Figure 12: Density estimation of liquidity-debt interactions across enterprise networks.

4.6 Computational complexity analysis of proposed SM-ISAGNN model

Table 5 highlights the computational trade-offs between ISAGNN and SM-ISAGNN. While the baseline ISAGNN primarily scales with nodes, edges, and hidden dimensions, SM-ISAGNN introduces additional complexity from the dual-encoder module and swarm intelligence optimization. These components slightly increase training costs but significantly enhance predictive accuracy and multi-hop contagion detection. Importantly, inference overhead remains comparable to ISAGNN, ensuring that the proposed model is both scalable and practical for large enterprise networks.

Table 5: Complexity comparison of ISAGNN and proposed SM-ISAGNN

Model Component	Time Complexity	Space Complexity
ISAGNN	$O(L.(E.H+N.H^2))$	$O(N \cdot H+E)$
Dual-Encoder Module	$O(N \cdot H^2)$	$O(N \cdot H)$
SMO Optimization	$O(I \cdot S \cdot C_{fitness})$	$O(S \cdot D)$
SM-ISAGNN [Proposed]	Above combined + training epochs T	$O(N \cdot H+E+S \cdot D)$

4.7 Impact of components on model prediction performance

Table 6 shows the ablation study that evaluates the contributions of SM-ISAGNN components. The full SM-ISAGNN achieves the highest performance with an accuracy of 0.967, precision of 0.966, recall of 0.967, and F1-score of 0.966. Removing the dual-encoder drops metrics to accuracy 0.902, precision 0.894, recall 0.889, and F1-score 0.891. Excluding SMO optimization yields accuracy 0.923, precision 0.919, recall 0.921, and F1-score 0.920. Without self-attention, performance reduces to accuracy 0.936, precision 0.933, recall 0.937, and F1-score 0.935, confirming each component's importance.

Table 6: Ablation study results for SM-ISAGNN model variants performance.

Model Variant	**Accuracy**	**Precision**	**Recall**	**F1-score**
Full SM-ISAGNN (Proposed)	0.967	0.966	0.967	0.966
w/o Dual-Encoder	0.902	0.894	0.889	0.891
w/o SMO Optimization	0.923	0.919	0.921	0.920
w/o Self-Attention	0.936	0.933	0.937	0.935

4.8 5-Fold cross-validation

The 5-fold cross-validation is used to assess the efficacy of SM-ISAGNN in financial risk contagion path prediction in terms of utilizing accuracy, recall, precision, and F1 score. The results are displayed in Table 7. The model obtained an F1-score of 0.9626, precision of 0.9623, accuracy of 0.9667, and recall of 0.9667 in Fold 1. The recall, precision, and accuracy metrics for Fold 2 are consistent at around 0.9333, with an F1-score of 0.9323, which is marginally lower. The performance of Fold 3 is excellent, with an F1-score of 0.9400, recall of 0.9200, accuracy of 0.9500, and precision of 0.9130. Folds 4 and 5 show consistent outcomes, each with F1-score, accuracy, precision, and recall values of 0.9667. The findings show that all folds have strong and reliable prediction performance.

Table 7: Performance comparison of the model with 5-fold cross-validation

Fold	***Accuracy***	***Precision***	***Recall***	***F1-score***
1	0.9667	0.9623	0.9667	0.9626
2	0.9333	0.9333	0.9333	0.9323
3	0.9500	0.9130	0.9200	0.9400
4	0.9667	0.9667	0.9667	0.9667
5	0.9667	0.9667	0.9667	0.9667
Average	**0.9667**	**0.9658**	**0.9667**	**0.9657**

The efficiency of a model employed in financial risk contagion path prediction is evaluated by its performance metrics. The average recall, precision, and F1-score values were 0.9667, 0.9658, and 0.9657, whereas the average accuracy value was 0.9667, as displayed in Figure 13.

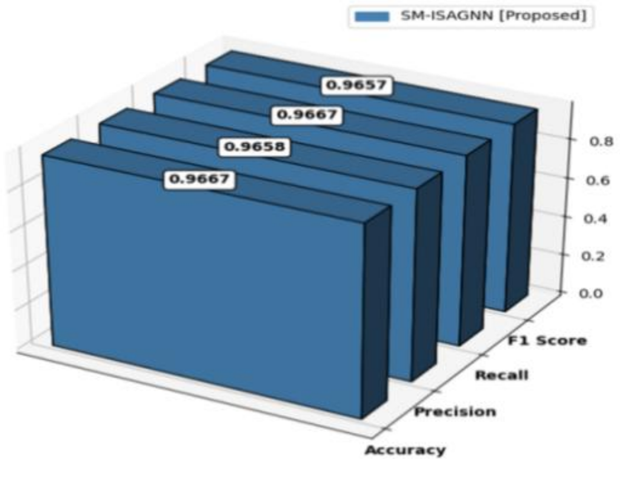

Figure 13: Performance analysis 5-fold cross-validating SM-ISAGNN predictive robustness.

4.9 Performance assessment of the proposed method employing the different metrics

The performance of various machine learning models for classification was evaluated in terms of F1-score, recall, accuracy, and precision. Among the models, SM-ISAGNN achieved the highest performance, with an accuracy of 98.15%, precision of 98.22%, recall of 97.89%, and F1-score of 97.91%, demonstrating its superiority over traditional methods. NGBoost also performed well, achieving 90.46% accuracy, 90.67% precision, 88.56% recall, and 89.52% F1-score. LSSVM obtained an accuracy of 90.15% and recall of 85.63%, while precision and F1-score were not reported. XGBoost reached 86.79% accuracy, 86.94% precision, 85.44% recall, and 86.09% F1-score. LightGBM recorded 85.57% accuracy, 87.18% precision, 80.88% recall, and 83.48% F1-score. SVM achieved 83.57% accuracy, 89.62% precision, 66.25% recall, and 69.84% F1-score. KNN showed 82.80% accuracy, 89.15% precision, 69.80% recall, and 74.33% F1-score. Overall, SM-ISAGNN outperforms all other models across all evaluation metrics. Table 8 and Figure 14 illustrate the superior performance of SM-ISAGNN across all evaluation metrics, highlighting its robustness in predicting financial contagion paths.

Table 8:Performance comparison of financial contagion prediction methods

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
LSSVM [26]	90.15	–	85.63	–
SMOTE-ENN [27]	97.18	97.12	97.07	97.08
XGBoost [27]	86.79	86.94	85.44	86.09
LightGBM [27]	85.57	87.18	80.88	83.48
SVM [27]	83.57	89.62	66.25	69.84
KNN [27]	82.80	89.15	69.80	74.33
NGBoost [27]	90.46	90.67	88.56	89.52
SM-ISAGNN [proposed]	**98.15**	**98.22**	**97.89**	**97.91**

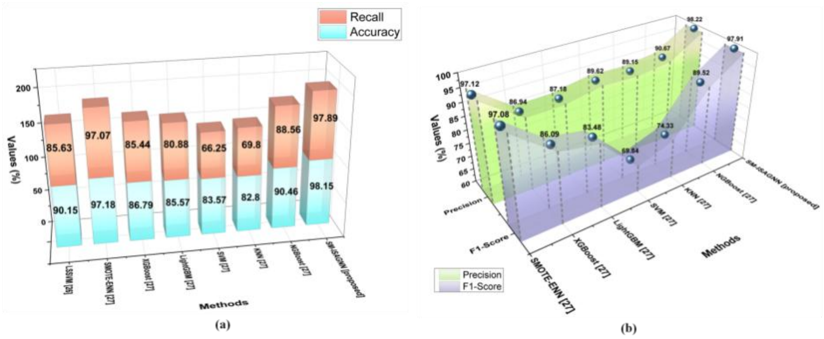

Figure 14: (A) Accuracy, Recall and (b) Precision, F1-score for contagion path prediction in financial networks.

The efficiency of the traditional ISAGNN model and the proposed SM-ISAGNN for financial risk contagion path forecasting, measured using contagion path accuracy, is presented in Table 9. Compared to the conventional approach, the proposed SM-ISAGNN identifies more real contagion paths, with a higher path hit ratio of 84.3%, whereas the traditional ISAGNN achieves 58.7%, as shown in Figure 15. The average path length matched for the ISAGNN approach is 2.1 hops, while the SM-ISAGNN method achieves 3.8 hops, indicating better coverage of contagion paths. The multi-hop detection rate of the SM-ISAGNN strategy is 77.5%, significantly outperforming the conventional ISAGNN's 41.6%, and it also reduces the false path prediction rate to 7.9% compared to 21.3% for ISAGNN. Furthermore, the proposed SM-ISAGNN enhances path diversity (0.81 vs. 0.42), shortest-path alignment (84.8% vs. 71.4%), and betweenness centrality accuracy (81.6% vs. 69.1%), demonstrating stronger accuracy, reliability, and robustness in identifying financial contagion propagation.

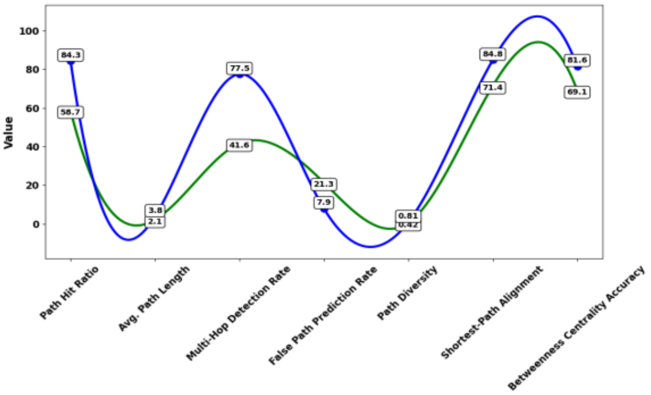

Figure 15: Comparative analysis of path metrics between ISAGNN and SM-ISAGNN.

Table 9: Comparative evaluation of the traditional ISAGNN approach and the proposed method.

Contagion Path Accuracy	ISAGNN	SM-ISAGNN [Proposed]
Path Hit Ratio (%)	58.7	**84.3**
Avg. Path Length Matched (hops)	2.1	**3.8**
Multi-Hop Detection Rate (%)	41.6	**77.5**
False Path Prediction Rate (%)	21.3	**7.9**
Path Diversity	0.42	**0.81**
Shortest-Path Alignment (%)	71.4	**84.8**
Betweenness Centrality Accuracy (%)	69.1	**81.6**

In risk propagation prediction, the node risk activation rate of the proposed SM-ISAGNN approach is 86.2%, which outperforms the traditional ISAGNN method with the low node risk activation rate of 62.5%, as depicted in Figure 16. In comparison, the conventional ISAGNN approach has an interconnected node prediction accuracy of 59.2%, whereas the suggested SM-ISAGNN model has a high interconnected node prediction accuracy of 83.7%, as displayed in Figure 16. With the contagion sequence accuracy of 81.4%, the proposed SM-ISAGNN strategy outperforms the traditional ISAGNN model, which has a contagion sequence accuracy of 54.9%, as shown in Figure 16. When compared to the conventional approach, the suggested SM-ISAGNN method has a risk accumulation prediction consistency score of 0.71, while the ISAGNN approach has a risk accumulation prediction consistency score of 0.46, as depicted in Figure 16. Table 10 shows the effectiveness of a conventional ISAGNN model and the proposed SM-ISAGNN method for financial risk contagion path forecasting.

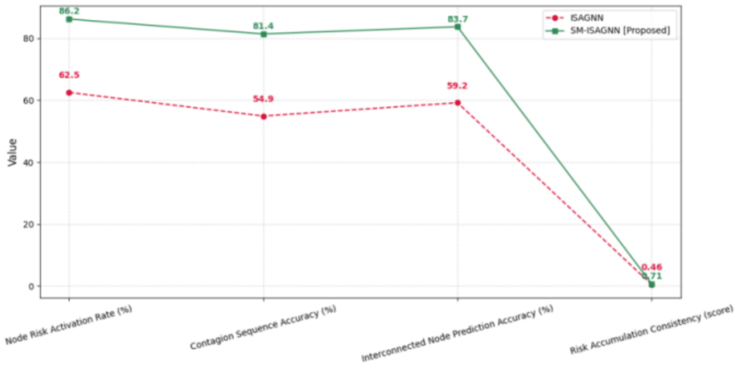

Figure 16: Comparative analysis of node risk activation metrics in contagion.

Table 10: Comparative assessment of the conventional ISAGNN approach and the suggested approach

Risk Propagation Prediction	ISAGNN	SM-ISAGNN [Proposed]
Node Risk Activation Rate (%)	62.5	**86.2**
Contagion Sequence Accuracy (%)	54.9	**81.4**
Interconnected Node Prediction Accuracy (%)	59.2	**83.7**
Risk Accumulation Prediction Consistency Score	0.46	**0.71**

5 Discussion

Financial risk, an inherent aspect of economic systems, reflects the likelihood of losses from market or institutional challenges. Despite its potential, the ISAGNN model for financial risk contagion path forecasting faces notable limitations. Its reliance on accurate financial network data overlooks informal or hidden interconnections, weakening robustness. Although the self-attention mechanism enhances feature learning, it increases computational cost and training time, restricting scalability for real-time or large-scale applications. Moreover, ISAGNN struggles to adapt to rapid structural changes in financial networks triggered by unexpected economic shocks. Low interpretability of self-attention weights complicates stakeholders' understanding of predictions, while overfitting and parameter tuning remain persistent challenges. Performance metrics reveal significant constraints, with ISAGNN achieving only a 58.7% path hit ratio, 41.6% multi-hop detection, and 54.9% contagion sequence accuracy.Beyond technical accuracy, contagion path prediction raises ethical and policy concerns. Misclassifying stable firms as “at-risk” can damage reputations, limit credit, or trigger undue regulation, while missing vulnerable firms may intensify crises. Responsible deployment requires transparency, human oversight, and interpretability, ensuring predictive insights align with regulatory standards to strengthen stability and minimize unintended economic or ethical harm.To overcome these challenges, a novel SM-ISAGNN model was introduced for financial risk contagion path forecasts.

6 Conclusion

Financial risk has become one of the most important issues for regulators, investors, financial institutions, and policymakers in today's increasingly integrated worldwide economy. The enterprise financial network dataset was collected from the Kaggle platform. The data is preprocessed employing the data cleaning and Z-score normalization approaches. An innovative SM-ISAGNN was presented to predict the contagion path of financial risk in enterprise networks. The performance of the suggested method was evaluated in terms of path hit ratio (84.3%), average path length matched (3.8 hops), multi-hop detection rate (77.5%), false path prediction rate (7.9%), interconnected node prediction accuracy (83.7%), node risk activation rate (86.2%), contagion sequence accuracy (81.4%), and risk accumulation prediction consistency score (0.71). The challenges include models' scalability concerns, problems capturing quick shifts in networks, inadequate data, issues with multi-channel risk integration, and limited adaptation to changing financial environments. Future scope involves employing novel artificial intelligence (AI) methods, integrating real-time data analytics, increasing model scalability, and creating adaptive frameworks for proactive financial risk administration in dynamic markets.

Competing interests

The authors have declared that no competing interests exist.

Ethics statement

This study did not involve human or animal subjects, and therefore, ethical approval was not required.

Data availability statement

All data generated or analysed during this study are included in this article.

Fund

The Second Batch of Teaching Reform Projects for High er Vocational Education during the "14th FiveYear Plan" Period in Zhejiang Province，Project number:jg202404 06

References

[1] Chen, J., & Sun, B. (2024). Enhancing financial risk prediction using TG-LSTM model: An innovative approach with applications to public health emergencies. Journal of the Knowledge Economy, 1–21. https://doi.org/10.1007/s13132-024-02081-x

[2] Wei, S., Lv, J., Guo, Y., Yang, Q., Chen, X., Zhao, Y., Li, Q., Zhuang, F., & Kou, G. (2024). Combining intra-risk and contagion risk for enterprise bankruptcy prediction using graph neural networks. Information Sciences, 659, 120081.

[3] Aliano, M., Cananà, L., Ciano, T., Ragni, S., & Ferrara, M. (2024). On the dynamics of an SIR model for a financial risk contagion. Quality & Quantity, 1–25. https://doi.org/10.1007/s11135-024-02009-2

[4] Fialkowski, J., Diem, C., Borsos, A., & Thurner, S. (2025). A data-driven econo-financial stress-testing framework to estimate the effect of supply chain networks on financial systemic risk. arXiv. https://doi.org/10.48550/arXiv.2502.17044

[5] Dong, Y., & Dong, Z. (2023). An innovative approach to analyze financial contagion using causality-based complex network and value at risk. Electronics, 12(8), 1846. https://doi.org/10.3390/electronics12081846

[6] Han, J. (2025). Deep learning-based identification and quantitative analysis of risk contagion pathways in private credit markets. Journal of Sustainability,Policy, and Practice, 1(2), 32–44.

[7] Berloco, C., De Francisci Morales, G., Frassineti, D., Greco, G., Kumarasinghe, H., Lamieri, M., Massaro, E., Miola, A., & Yang, S. (2021). Predicting corporate credit risk: Network contagion via trade credit. PLoS One, 16(4), e0250115. https://doi.org/10.1371/journal.pone.0250115

[8] Elhoseny, M., Metawa, N., Sztano, G., & El-Hasnony, I. M. (2025). Deep learning-based model for financial distress prediction. Annals of Operations Research, 345(2), 885–907. https://doi.org/10.1007/s10479-022-04766-5

[9] Jia, K., & Pan, Y. (2025). Financial systemic risk prediction using deep neural networks and long short-

term memory. Journal of Circuits, Systems and Computers. https://doi.org/10.1142/S0218126625502949

[10] Yang, T., Li, A., Xu, J., Su, G., & Wang, J. (2024). Deep learning model-driven financial risk prediction and analysis. https://doi.org/10.20944/preprints202406.2069.v1

[11] Jin, Q., Sun, L., Chen, Y., & Hu, Z. L. (2024). Financial risk contagion based on dynamic multi-layer network between banks and firms. Physica A: Statistical Mechanics and Its Applications, 638, 129624. https://doi.org/10.1016/j.physa.2024.129624

[12] Fan, R., Xie, X., Wang, Y., & Lin, J. (2025). Effect of financial contagion between real and financial sectors on asset bubbles: A two-layer network game approach. Managerial and Decision Economics, 46(1), 393–408. https://doi.org/10.1002/mde.4381

[13] Cheng, D., Niu, Z., Li, J., & Jiang, C. (2022). Regulating systemic crises: Stemming the contagion risk in networked-loans through deep graph learning. IEEE Transactions on Knowledge and Data Engineering, 35(6), 6278–6289. https://doi.org/10.1109/TKDE.2022.3162339

[14] Chung, V., Espinoza, J., & Mansilla, A. (2024). Analysis of financial contagion and prediction of dynamic correlations during the COVID-19 pandemic: A combined DCC-GARCH and deep learning approach. Journal of Risk and Financial Management, 17(12), 567. https://doi.org/10.3390/jrfm17120567

[15] Liao, X., & Li, W. (2025). Research on the tail risk contagion in the international commodity market on China's financial market: Based on a network perspective. Kybernetes, 54(2), 807–831. https://doi.org/10.1108/K-06-2023-1001

[16] Xie, X., Zhang, F., Liu, L., Yang, Y., & Hu, X. (2023). Assessment of associated credit risk in the supply chain based on trade credit risk contagion. PLoS One, 18(2), e0281616. https://doi.org/10.1371/journal.pone.0281616

[17] Geng, X., Han, B., Yang, D., & Zhao, J. (2024). Credit risk contagion of supply chain finance: An empirical analysis of supply chain listed companies. PLoS One, 19(8), e0306724. https://doi.org/10.1371/journal.pone.0306724

[18] Yao, Q., Mao, C., & Guo, Y. (2024). A fuzzy neural network-based intelligent warning method for financial risk of enterprises. Journal of Circuits, Systems and Computers, 33(14), 2450251. https://doi.org/10.1142/S0218126624502517

[19] Mu, P., Chen, T., Pan, K., & Liu, M. (2021). A network evolution model of credit risk contagion between banks and enterprises based on agent-based model. Journal of Mathematics, 2021(1), 6593218. https://doi.org/10.1155/2021/6593218

[20] Ma, J., Liu, Y., Zhao, L., & Liang, W. (2024). Research on the mechanism and application of spatial credit risk contagion based on complex network model. Managerial and Decision Economics, 45(2), 1180–1193. https://doi.org/10.1002/mde.4025

[21] Wang, L., Jiang, X., Chen, T., & Zhu, R. (2024). The contagion of debt default risk in energy enterprises considering carbon price fluctuations. Mathematics, 12(17), 2776. https://doi.org/10.3390/math12172776

[22] Li, M., & Fu, Y. (2022). Prediction of supply chain financial credit risk based on PCA-GA-SVM model. Sustainability, 14(24), 16376. https://doi.org/10.3390/su142416376

[23] Chen, X. (2025). Research on financial market volatility prediction and risk response strategy based on LSTM network. J. Combin. Math. Combin. Comput, 127, 5197–5213. https://doi.org/10.61091/jcmcc127a-293

[24] Kadkhoda, S. T., & Amiri, B. (2024). A hybrid network analysis and machine learning model for enhanced financial distress prediction. IEEE Access, 12, 52759–52777. https://doi.org/10.1109/ACCESS.2024.3387462

[25] Ionescu, Ș., Delcea, C., & Nica, I. (2025). Improving real-time economic decisions through edge computing: Implications for financial contagion risk management. Computers, 14(5), 196. https://doi.org/10.3390/computers14050196

[26] Gu, J. (2022). Risk prediction of enterprise credit financing using machine learning. Informatica, 46(7).https://doi.org/10.31449/inf.v46i7.4247.

[27] Zhu, Y., Hu, Y., Liu, Q., Liu, H., Ma, C., & Yin, J. (2023). A Hybrid Approach for Predicting Corporate Financial Risk: Integrating SMOTE-ENN and NGBoost. IEEE Access, 11, 111106-111125.10.1109/ACCESS.2023.3323198.

Appendix 1

RNN	Recurrent neural network	DL	Deep learning
PCA-GA-SVM	Principal component analysis + genetic algorithm + support vector machine	DCC-GARCH	Dynamic conditional correlation multiple generalized autoregressive conditional heteroskedasticity
SMO	Starling Murmuration Optimizer	CA	Categorical attribute
XGBoost	Extreme Gradient Boosting	FR	Feature representation
FPT	Fuzzy preference theory	ARDL	Autoregressive distributed lag
AI	Artificial intelligence	TP	True positive

QRD	Quantum random dive	FNN	Fuzzy neural network
FN	False negative	LSTM	Long Short-Term Memory
SVM	Support Vector Machine	TCRC	Trade credit risk contagion
GCL	Graph convolution layer	OT	Observation time
TENET	Tail-event driven network risk	RF	Random Forest
ML	Machine learning	SD	Standard deviation
ABM	Agent-based model	ISAGNN	Improved Self-Attention Graph Neural Network
FP	False positive	TN	True negative
EC	Edge computing	SC	Supply chain
QHC	Quantum harmonic oscillator	GNN	Graph neural network
ARMA	Autoregressive moving average model	RL	Reinforcement learning
ReLU	Rectified linear unit	DDR	Debt default risk
GBT	Gradient Boosting Tree	LSSVM	Leastsquare support vector machine
SMOTE-ENN	Synthetic Minority Over-sampling and Edited Nearest Neighbors	LightGBM	Light Gradient Boosting Machine
KNN	K-Nearest Neighbors	NGBoost	Natural Gradient Boosting