

Sensor Data Fusion and Processing Using Temporal Graph Neural Network for Real-time Cost Variance Analysis in Production Lines

Yuenan Liu,¹ Mingyu Zhao,^{2*} and Xuehong Zhao¹

¹Harbin Finance University; Harbin 150030, China

²Heilongjiang Academy of Sciences; Harbin 150001, China

(Received July 8, 2026; accepted October 20, 2026)

Keywords: cost variances, financial analysis, manufacturing enterprises, counterfactual costs, time series network, deep learning

Manufacturing cost anomalies occur through interactions among production processes, resource consumption, equipment conditions, and product quality. Conventional data-driven soft sensors lack interpretable links between sensor measurements and financial outcomes and do not incorporate manufacturing cost relationships. To address these limitations, we developed a counterfactual temporal graph neural network (CF-TGNN) for sensor-data fusion and its use for cost-variance analysis. The model integrates a domain-constrained sensor–cost embedding layer, a production-line similarity graph, temporal graph propagation, and counterfactual normal-cost generation with structural consistency constraints. These elements enable the transformation of heterogeneous sensor data into structured cost representations, capturing relationships among production lines and identifying anomalies through deviations from estimated normal costs. The model was evaluated using 35,464 weekly observations from 341 production lines across 64 manufacturing companies for 104 weeks in China. The CF-TGNN model showed an F1-score of 0.840 and an area under the receiver operating characteristic curve of 0.963, outperforming conventional machine learning, deep learning, and graph neural network baselines. It also achieved an average early-warning horizon of 2.36 weeks, with one-period and two-period early-warning rates of 76.84 and 58.19%, respectively, and a missing-alarm rate of 17.10%. Robustness test results presented consistent performance under alternative anomaly definitions, graph configurations, and selected industry-specific evaluations. By integrating manufacturing cost knowledge, temporal graph learning, and counterfactual inference, we developed a model that conducts sensor-data-driven cost-variance analysis. Using finer-grained data and through deployment in different industries, we can further enhance the model performance.

*Corresponding author: e-mail: zhaomingyu2794@dingtalk.com
<https://doi.org/10.18494/SAM6476>

1. Introduction

In modern smart manufacturing, numerous sensors are deployed to monitor production processes in real time. Inline power analyzers, photoelectric counters, pneumatic and hydraulic pressure sensors, and machine-vision inspection modules are used to generate high-frequency data to obtain information on machine health, energy consumption, and product quality. In addition to these sensing devices, data-driven soft sensors have been developed to transform low-level measurements into higher-level operational indicators for the efficient monitoring and control of production-line performance.⁽¹⁾ Multi-source sensor data are used to analyze production cost variations. In manufacturing, cost variance is not caused by a single operational abnormality. It results from the cumulative effects of multiple interacting factors, such as fluctuations in energy consumption, equipment degradation, material waste, production interruptions, and reduced product quality.

Conventional threshold-based monitoring methods frequently generate false alarms and fail to identify combined anomalies because individual sensor data are insignificant. For example, a slight voltage fluctuation, minor tool wear, or intermittent micro-stoppages might not trigger conventional alarms, although their combined effects can substantially reduce production efficiency and increase manufacturing costs. Therefore, effective cost variance analysis requires the integration and interpretation of heterogeneous sensor signals to identify latent operational issues before they escalate into financially significant losses.^(2,3)

To address this challenge, various machine learning and deep learning models, including long short-term memory (LSTM), gated recurrent unit (GRU), and graph convolutional network (GCN)-based models, are used for industrial fault detection, energy forecasting, and equipment failure prediction using sensor data.^(4,5) These models extract complex patterns from high-dimensional data, enhancing predictive performance. However, they are mainly applied in operational monitoring, with the limited consideration of the financial implications of production anomalies. In addition, interpretability, domain knowledge integration, robustness to sensor data noise, and the modeling of relationships among production lines are challenges to address in sensor-data-driven production monitoring. Existing deep learning models generally process sensor data as numerical features without incorporating manufacturing cost-information mechanisms into the learning process.⁽⁶⁾ As a result, their outputs are too complex to interpret from operational and financial perspectives, although they show satisfactory predictive accuracy. Most models also fail to examine functional similarities among peer manufacturing lines. This reduces the models' capability to distinguish genuine process abnormalities from localized disturbances caused by sensor data drift, calibration errors, communication interruptions, or environmental noise.⁽⁷⁾ In addition, cost decomposition relationships and conservation constraints are rarely included within existing learning frameworks, limiting model interpretability and practical applicability.^(8–10)

To overcome these limitations, we developed a counterfactual temporal graph neural network (CF-TGNN) for sensor-data-driven cost variance analysis in manufacturing environments.⁽¹¹⁾ The developed CF-TGNN model incorporates manufacturing cost-information mechanisms into a temporal graph-learning architecture, which is different from conventional deep learning

models that focus on pattern recognition. Heterogeneous sensor data are transformed into structured data on material, labor, energy, and overhead costs to be linked to operational conditions and financial performance. By generating counterfactual normal-cost baselines, the model identifies anomalies caused by process deviations and quantifies their contributions to cost variance. In the model, graph-based relational learning is employed to capture structural similarities among 341 production lines, sharing operational information across comparable production environments and improving robustness against localized sensor disturbances and noise.^(12,13) Through the integration of cost-aware embeddings, temporal learning, and counterfactual inference, the model extends the role of sensor systems for intelligent financial decision support and proactive cost management in smart manufacturing environments.^(14,15)

In addition to cost variance analysis, the model contributes to the advancement of sensing technologies and functional sensor materials. By identifying situations in which existing physical data cannot explain observed cost deviations, the model informs sensing limitations that require improvements in sensitivity, response speed, durability, or environmental robustness.⁽¹⁶⁾ This enables the development of next-generation sensing materials, including metal-oxide nanocomposites, flexible piezoelectric films, and other advanced functional materials designed for demanding industrial environments.⁽¹⁷⁾ The modular graph-based architecture of the model is compatible with edge AI-enabled smart sensors, distributed wireless sensor networks, and multimodal industrial data systems.⁽¹⁸⁾ Therefore, the model can be used for data-driven manufacturing analytics and for future sensor design and deployment strategies.

2. Method

2.1 Data collection

Data were collected from 341 production lines operated by 64 manufacturing companies in Heilongjiang Province, China, over a period of 104 weeks, yielding 35,464 weekly production-line observations. The participating companies represented diverse industrial sectors, including equipment manufacturing, food and agricultural processing, electronic component manufacturing, chemical processing, and light manufacturing. The collected data were organized into company-level, production-line, sensor-acquisition, and temporal dimensions (Table 1).

Table 1
Companies and their production lines used for data collection.

Type	Number of companies	Number of production lines	Ratio to total number of production lines
Equipment manufacturing	14	83	24.34%
Food processing	12	61	17.89%
Agricultural products	10	49	14.37%
Electronic component	9	45	13.20%
Chemical materials	8	43	12.61%
Light manufacturing	11	60	17.59%
Total	64	341	100.00%

Data were collected through a multi-tier cyber-physical architecture integrating physical sensors, programmable logic controllers (PLCs), manufacturing execution systems (MES), and enterprise resource planning platforms. Each production line was equipped with sensors to continuously collect data on power consumption, machine operating status, material flow, and product quality. To ensure interoperability among heterogeneous devices and enable real-time monitoring across manufacturing sites, sensor data were transmitted through standardized industrial communication protocols and consolidated within a unified data management framework. Sensing devices included Schneider PM800 and Siemens PAC3200 power meters, Siemens S7-1200 and Mitsubishi FX5U PLCs, HBM PW15 load cells, Zebra FX9600 radio-frequency identification (RFID) scanners, and Basler Ace and Keyence CV-X machine-vision systems. In addition to these representative devices, equivalent sensors from other vendors, including Omron, SICK, Honeywell, and WIKA, were deployed according to site-specific operational requirements.

Power consumption was monitored using Schneider PM800 (Schneider Electric, France) and Siemens PAC3200 (Germany) digital power meters, which recorded three-phase voltage and current measurements at a sampling frequency of 1 Hz using the Modbus-RTU protocol. Machine operating status was monitored through PLC registers, including Siemens S7-1200 (Germany) and Mitsubishi FX5U (Japan) controllers. These devices logged equipment startup, shutdown, and downtime events and transmitted operational information through the Open Platform Communications Unified Architecture (OPC UA) protocol. Material flow was tracked using strain-gauge load cells, such as the HBM PW15 from Germany, and industrial RFID scanners, such as the Zebra FX9600 from the United States. These devices transmitted data through RS-485 or Ethernet connections and interfaced directly with MES databases. Quality inspection was performed using automated optical inspection (AOI) cameras, including the Basler Ace series from Germany and the Keyence CV-X from Japan. These cameras captured images at 30 frames per second (FPS) and transmitted defects over Gigabit Ethernet Vision. Additional production variables were aggregated on a weekly basis.

Output was measured using MES counters and production line sensors. Unit material consumption was calculated using RFID systems and weighing devices. Unit labor hours were derived from MES work reporting terminals. Unit energy consumption was determined using power meters and energy management systems. Downtime duration was recorded in PLC logs, and utilization rate was calculated from MES and PLC records. Yield was measured using AOI systems and quality control terminals (Table 2). Raw sensor signals were transformed into structured process variables, which were then aggregated into weekly panel data. This dataset shows the data transmission path from sensor data to cost information and finally to financial indicator changes, forming the foundation for CF-TGNN modeling.

2.2 Anomaly identification

In this study, anomalies were defined by statistical irregularities and their financial significance. Unusual sensor data, such as higher energy use, longer downtime, or lower yield, were converted into abnormal production costs. In production networks, anomalies must be

Table 2
Sensor specifications and data acquisition protocols

Data	Sensor (model, manufacturer, country)	Sampling frequency/ protocol	Description
Power consumption	Schneider PM800 (Schneider Electric, France); Siemens PAC3200 (Germany)	1 Hz/Modbus-RTU	Total electricity consumption per reporting period
Machine status	Siemens S7-1200 PLC (Germany); Mitsubishi FX5U PLC (Japan)	Event-driven/OPC-UA	Startup, shutdown, failure, idle state
Material flow	HBM PW15 load cell (Germany); Zebra FX9600 RFID (USA)	10 Hz/RS-485, Ethernet	Material input and consumption
Quality inspection	Basler Ace AOI camera (Germany); Keyence CV-X (Japan)	30 FPS/Gigabit Ethernet Vision	Records of qualified, defective, and reworked products
Output	MES counters, production line sensors	Aggregated weekly	Qualified outputs
Unit material consumption	RFID + weighing devices	Aggregated weekly	Material consumption/output
Unit labor hours	MES work reporting terminals	Aggregated weekly	Labor hours/output
Energy consumption	Power meters, energy management system	Aggregated weekly	Energy consumption/output
Downtime duration	PLC logs	Event-driven	Equipment downtime
Utilization rate	MES + PLC	Aggregated weekly	Runtime/planned runtime
Yield	AOI + quick connect terminals	Aggregated weekly	Qualified/total products

evaluated simultaneously at the financial and sensor levels. Traditional fault detection methods separate these domains, resulting in unquantified operational alarms or lagging financial audits. To solve such problems, we established a unified, bi-directional thresholding model. Macro-level cost variances function as enterprise-level soft sensors when structural operational budgets are violated.

An anomaly was verified to evaluate shop-floor state vectors against baseline boundary conditions. A production line was classified as anomalous if it satisfied any of the following macro-level operational violations:

- Budget deviation rate: The computed unit product cost exceeds its historical moving average, yielding a structural budget variance greater than 10%.
- Manufacturing overhead ratio: The allocation of auxiliary manufacturing overhead surges significantly beyond its statistically established historical baseline.
- Gross profit margin: The realized gross profit margin falls below its historical mean.

To ensure interpretable violations, the model couples the following events with its corresponding sensor data:

1. Electrical power anomalies: Micro-scale spikes in unit energy consumption logged by inline digital power meters isolate localized electrical or thermodynamic degradation.
2. Asset utilization anomalies: Extended equipment downtime or micro-stoppages parsed from PLC register state transitions identify mechanical failures or line balancing constraints.
3. Quality degradation anomalies: Sudden drops in line yield rates captured by AOI terminals isolate machine tool wear or raw material defects.

4. Material volumetric anomalies: Material mass overages recorded via inline strain-gauge weighing sensors and RFID component tracking indicate excessive physical scrap or process waste.
5. Labor capacity anomalies: Increased operational run-times registered across MES logging terminals point to process inefficiencies or manual assembly delays.

By fusing high-frequency sensor data with discrete financial metrics, the model ensured that production anomalies were technically traceable and economically significant. In application, sensors were used to issue early warnings and provide root-cause evidence, whereas financial structures were used to validate the macro-scale impact on factory cost efficiency. In this study, yield is used to represent first-pass yield.⁽¹⁹⁾ It is defined as the proportion of qualified products produced that meet specification standards on the first attempt without requiring rework or resulting in scrap, expressed as

$$Yield_{i,t} = \frac{N_{qualified,i,t}}{N_{total,i,t}}. \quad (1)$$

Here, $N_{total,i,t}$ is the total volume of units processed on line i during week t (tracked using MES counters) and $N_{qualified,i,t}$ represents the volume of non-defective, non-reworked outputs verified by AOI terminals and quality control logs.

3. CF-TGNN Model

The CF-TGNN model was constructed, comprising the following interconnected layers: domain-knowledge constrained sensor layer, production line similarity graph construction layer, temporal graph propagation layer, and counterfactual normal cost generation layer. The model transforms raw sensor data into structured costs and builds a similarity graph across production lines, propagates temporal information through graph-based encoders, and finally generates counterfactual normal costs. Anomalies are identified by comparing actual sensor-derived costs with these counterfactual baselines. Figure 1 shows the architecture of the CF-TGNN model. Sensor-derived input features (energy, downtime, yield, and material flow) are embedded into cost vectors, connected through a similarity graph, propagated across time, and finally used to generate counterfactual costs (a calculated baseline that represents what a production line's operational expenses should have been under normal, steady-state, and optimized conditions for a given output volume) and decisions on cost variances.

1. Domain-knowledge constrained sensor layer

Instead of feeding raw sensor data into the neural network, this layer applies deterministic engineering rules to project sensor telemetry into a structured four-dimensional cost vector.

$$c_{i,t}^{str} = [MC_{i,t}, LC_{i,t}, EC_{i,t}, OC_{i,t}] \quad (2)$$

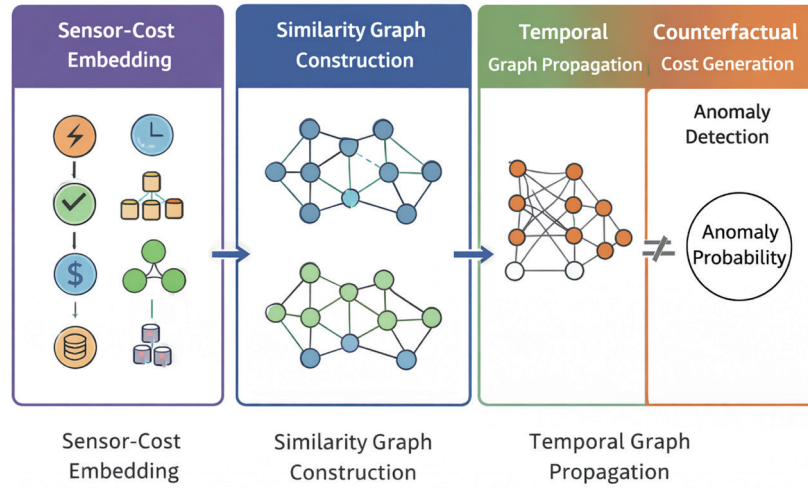


Fig. 1. (Color online) Architecture of CF-TGNN model. In the second panel from the left, blue nodes represent equipment or electronics, green nodes represent food and agricultural processing, and light blue nodes are active target nodes. In the third panel, orange nodes indicate active hidden states and white nodes represent the baseline of input states.

Here, $MC_{i,t}$ is the material cost derived from production output, unit material consumption, and sensor-tracked material prices; $LC_{i,t}$ is the labor cost calculated from output, unit labor hours, and wage rates; $EC_{i,t}$ is the energy cost estimated on the basis of unit energy consumption from power meters and prevailing energy tariffs; and $OC_{i,t}$ is the overhead cost estimated from a function of downtime, yield rate, and equipment utilization. $OC_{i,t}$ is derived as a function of equipment downtime, line utilization rate, and $Yield_{i,t}$. A decrease in first-pass yield reduces the effective volume of saleable units, increasing the unit product cost through raw material loss and overhead absorption inflation.

$$OC_{i,t} = f\left(Downtime_{i,t}, Utilization_{i,t}, \frac{1}{Yield_{i,t}}\right) \quad (3)$$

The layer is responsible for the physical state-space transformation to ensure that the model learns statistical correlations and the logic of cost information. The structured vector is concatenated with raw sensor variables and passed through a nonlinear mapping function (multi-layer perceptron) with a rectified linear unit (ReLU).

2. Production line similarity graph construction layer

In this layer, production lines are connected through an unsupervised functional sensor topology graph G , not treated as isolated units.

$$G = (V, E, A) \quad (4)$$

Here, V represents the nodes of production lines, E represents edges with similarity relationships, and A is the adjacency matrix computed using the cosine similarity of cost-structure vectors. If two lines show highly correlated behavior (similar thermodynamic regimes or mechanical loading rates), they share an edge. This allows the model to borrow spatial context from peer lines, smoothing out localized sensor noise. Figure 2 shows the production line cost structure diagram. Nodes are color-coded by industry (equipment manufacturing, food processing, and chemical materials) and how cross-industry similarities are captured, enabling the graph to reflect functional sensor relationships rather than organizational charts.

3. Temporal graph propagation layer

To capture time-dependent behavior, the similarity graph is combined with a temporal encoder as follows.

$$h_{i,t}^g = \sigma \left(\sum_{j \in N(i)} \tilde{A}_{ij} W_g z_{j,t}^0 \right) \quad (5)$$

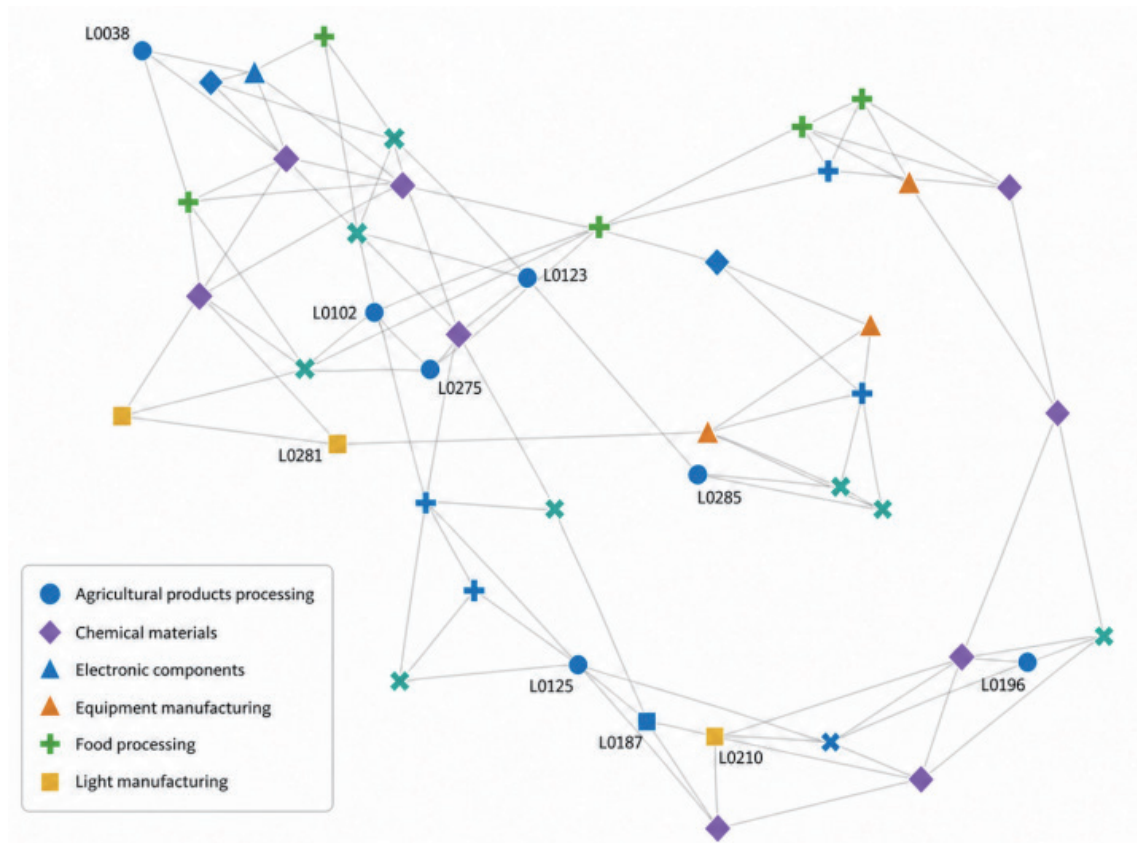


Fig. 2. (Color online) Cost structure diagram (labels indicate production line numbers).

$$h_{i,t}^s = GRU(h_{i,t}^g, h_{i,t-1}^s) \quad (6)$$

Here, $h_{i,t}^g$ is the graph-propagated feature vector for production line i at time t , representing how line i 's sensor profile is affected by its neighbors in the similarity graph, $\tilde{A}_{ij}$ is the normalized adjacency weight between i and j , determining how much effect neighbor j has on i , W_g is the trainable weight matrix including the raw sensor features transformed into higher-level representations, $z_{j,t}^0$ is the initial embedded feature vector of j at t , derived from its sensor inputs and cost embedding, and $\sigma(\cdot)$ is the nonlinear activation function (ReLU), which introduces nonlinearity so that the model can capture complex operational patterns.

On the basis of the equations, spatial similarity is captured across production lines, whereas the gated GRU temporal encoder captures time-dependent behavior within each line. The model detects spatially and temporally structured anomalies on the basis of the equations.

3. Counterfactual normal cost generation layer

The counterfactual generator acts as a predictive digital twin. This layer estimates whether the cost vector is under optimal, steady-state operating conditions, using the following equation:

$$\hat{C}_{i,t}^{cf} = MLP(h_{i,t}^s). \quad (7)$$

Here, $\hat{C}_{i,t}^{cf}$ is the counterfactual normal cost for production line i at t , representing the cost the line incurs under optimal, steady-state operating conditions, and MLP is a multilayer perceptron, which is a feedforward neural network that takes the temporal hidden state as input and outputs the predicted counterfactual cost, and $h_{i,t}^s$ is the temporal hidden state of production line i at t , produced by the GRU in the temporal graph propagation layer. It encodes both the current sensor features and historical behavior. The deviation between actual and counterfactual costs is defined as

$$D_{i,t} = \frac{C_{i,t}^{real} - \hat{C}_{i,t}^{cf}}{\hat{C}_{i,t}^{cf} + \varepsilon}. \quad (8)$$

Here, $D_{i,t}$ is the cost deviation score for production line i at t , measuring how far the actual cost deviates from the counterfactual normal cost, $C_{i,t}^{real}$ is the actual observed cost of production line i at t , calculated from real financial and sensor data, $\hat{C}_{i,t}^{cf}$ is the counterfactual normal cost predicted by the model [Eq. (5)], and ε is a constant added to the denominator to prevent division by zero. If the deviation is large, the system flags the production line as anomalous. The anomaly classifier then outputs both the probability of anomaly and interpretable metrics such as deviation magnitude.

The loss function of the CF-TGNN model is formulated as Eq. (7) to balance anomaly detection loss to determine binary cross-entropy, normal cost reconstruction loss to ensure the model learns normal cost behavior, and cost structure consistency loss to enforce conservation rules (material + labor + energy + overhead), and graph smoothing loss is used to maintain similarity among connected production lines. The total loss (L) is calculated by combining prediction error functions, including classification loss ($\mathcal{L}_{cls}$) and reconstruction loss ($\mathcal{L}_{rec}$) using weighting coefficients (λ).

$$L = L_{cls} + \lambda_1 L_{rec} + \lambda_2 L_{str} + \lambda_3 L_{graph} + \lambda_4 \|\Theta\|_2^2 \quad (9)$$

Here, L_{cls} is the classification binary cross-entropy loss used to train the anomaly detector to ensure that the model correctly distinguishes between normal and abnormal costs, L_{rec} is the reconstruction loss with which the difference between the actual cost and the counterfactual normal cost is measured for normal samples, and the true cost information mechanism is learned rather than fitting anomalies, L_{str} is the cost structure consistency loss, making the counterfactual cost equal to the sum of material, labor, energy, and overhead components, and L_{graph} is the graph smoothing loss. This keeps the learned representations of connected production lines similar, reflecting the idea that lines with similar sensor profiles should behave alike under normal conditions, $\lambda_1, \lambda_2, \lambda_3$, and λ_4 are hyperparameters that control the relative importance of each loss term, adjusted to enhance anomaly detection accuracy against interpretability and stability, and Θ is the set of model parameters (weights and biases). The term $\|\Theta\|_2^2$ is an L2 regularization penalty (weight decay or Euclidean norm penalty) and is used to ensure that the CF-TGNN model is accurate, interpretable, and stable in cost related to sensor formation logic and to prevent overfitting by discouraging excessively large parameter values. The L_2 regularization penalty is calculated using the squared Euclidean norm of all trainable model weights.⁽²⁰⁾

4. Experiment

To evaluate the performance of the CF-TGNN model, experiments were conducted on a high-performance computing cluster using PyTorch.⁽²¹⁾ The dataset containing 104 weeks of observations from 341 production lines was used to construct a heterogeneous multi-sensor data matrix that integrates electrical, mechanical, operational, and material-flow data. The dataset was used for the evaluation of sensor-data fusion and anomaly-detection performance under manufacturing conditions. The CF-TGNN model's performance was compared with representative machine learning, deep learning, graph neural network, and dynamic graph neural network models.

1. Machine learning models

Extreme gradient boosting (XGBoost)⁽²²⁾ and random forest (RF)⁽²³⁾ were selected as conventional machine learning baselines. These models process multimodal sensor

measurements, such as power consumption, PLC status information, and material-flow indicators, as static feature vectors. Consequently, they are unable to capture temporal dependencies or structural relationships among production lines.

2. Deep learning models

LSTM⁽²⁴⁾ and GRU⁽²⁵⁾ models were used to apply representative sequence-learning approaches. These models capture temporal patterns within individual production lines but treat each line independently, limiting their capability to exploit spatial correlations and making them susceptible to localized sensor noise and drift.

3. Spatiotemporal GNN (STGCN) models

STGCN⁽²⁶⁾ and temporal graph convolutional network (T-GCN)⁽²⁷⁾ models were used to apply representative graph-based approaches. These models incorporate graph structures to capture spatial dependencies among neighboring nodes, simultaneously modeling temporal dynamics. However, they assume fixed graph topologies and do not explicitly incorporate manufacturing-domain constraints, which may lead to operational states that are physically or economically implausible.

4. Dynamic GNN models

To compare the performance of CF-TGNN with that of other models, two temporal GNN models were experimented with. The evolving TGNN (E-TGNN) updates graph representations as node relationships evolve over time,⁽²⁸⁾ enabling the adaptive modeling of changing production environments. The dynamic self-attention network (DySAT) employs structural and temporal self-attention mechanisms to learn dynamic node embeddings from evolving graph sequences.⁽²⁹⁾ Although these models effectively capture temporal changes in graph topology, they conduct representation learning and lack explicit manufacturing knowledge, cost-information logic, and conservation constraints. As a result, their predictions may violate physical operating conditions or financial cost-structure relationships. In contrast, the CF-TGNN model embeds manufacturing cost-information mechanisms directly into the graph-learning process through sensor-cost embeddings, counterfactual cost generation, and structural consistency constraints. With the integration of domain-specific cost conservation principles with temporal graph learning, CF-TGNN ensures that anomaly predictions remain both operationally meaningful and financially interpretable, improving robustness to sensor disturbances and changing production conditions.

The CF-TGNN model embeds domain knowledge constraints directly into the graph topology and cost-information logic, ensuring that predicted states remain physically and financially plausible. The anomaly detection capability across all sensor nodes was quantified using standardized instrumentation evaluation metrics: precision, recall, F1-score, and area under the receiver operating characteristic curve (AUC).⁽³⁰⁾ These metrics are defined as follows.

$$Precision = \frac{TP}{TP + FP} \quad (10)$$

$$Recall = \frac{TP}{TP + FN} \quad (11)$$

$$F1\text{-score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (12)$$

Here, *TP*, *FP*, and *FN* denote true positives, false positives, and false negatives, respectively, relative to verified physical anomalies on the shop floor. The AUC is used to evaluate the tradeoff between detection sensitivity and false alarm rate across varying thresholds. To ensure reproducibility, all models were trained under identical data partitioning and normalization schemes, with hyperparameters optimized through Bayesian search.⁽³¹⁾ Evaluation was conducted following best practices for industrial soft sensor validation.⁽³²⁾

Precision represents the reliability of the soft sensor when an alarm is triggered, avoiding false alarms that cause unnecessary inspection downtime, and recall represents the system's sensitivity in detecting true operational faults to avoid missed alarms that allow costly efficiency losses to go unnoticed. F1-score represents the harmonic mean of precision and recall, providing a single balanced measure of overall detection performance.⁽³³⁾ In industrial soft sensors, when normal operating records far outnumber abnormal samples (class imbalance), a high F1-score ensures the model's high fault-detection sensitivity, simultaneously maintaining a low rate of false alarms.

5. Results

5.1 Descriptive statistics

Table 3 shows the descriptive statistics of the data collected: unit energy consumption [mean = 2.37; standard deviation (SD) = 0.82; range, 0.64–5.96], downtime duration (mean = 5.84; SD = 4.92; range, 0.1–37.8), and manufacturing cost ratio (the ratio of a production line's manufacturing overhead to its total production cost in a given period) (mean = 0.322;

Table 3
Descriptive statistics of data.

Variable	Mean	SD	Minimum	Maximum
Unit material consumption	1.215	0.284	0.512	2.166
Unit labor cost	0.0618	0.0241	0.0185	0.144
Unit energy consumption	2.37	0.82	0.64	5.96
Downtime duration	5.84	4.92	0.1	37.8
Yield	0.928	0.034	0.742	0.987
Unit product cost	56.82	18.77	18.94	143.6
Manufacturing cost ratio	0.322	0.083	0.125	0.574
Budget deviation rate	0.023	0.092	−0.198	0.417
Gross profit rate	0.171	0.061	−0.031	0.357
Cost deviation	0.0116	0.045	−0.082	0.284

SD = 0.083; range, 0.125–0.574). These findings indicate that production line costs are not determined solely by corporate-level factors but are strongly affected by the operational status of individual lines. Fluctuations in yield rates (mean = 0.928, SD = 0.034) and equipment utilization affect unit costs and the dilution of manufacturing overhead. The descriptive statistics provide the baseline for identifying anomalies at the production line level.

Among the 35,464 observations, 31,638 were classified as normal and 3,826 as anomalies, corresponding to an overall anomaly rate of 10.79%. The distribution of anomaly types is summarized in Table 4. The most frequent anomalies were energy spikes (986 cases, 25.77%) and downtime events (812 cases, 21.22%). These anomalies account for nearly half (46.99%) of all abnormal samples. This indicates the role of resource consumption efficiency and equipment operational stability as early-warning indicators of cost variance in manufacturing enterprises. Other anomalies observed include yield drops (735 cases, 19.21%), material waste (611 cases, 15.97%), and labor inefficiency (421 cases, 11.00%). Although mixed anomalies represent a smaller share (261 cases, 6.82%), they often correspond to larger cost deviations and sharper declines in gross profit margins.

5.2 Data partitioning and hyperparameter configuration

To evaluate the predictive stability and robustness of the CF-TGNN model, avoiding data leakage or look-ahead bias, the dataset was randomly partitioned into 70% training, 15% validation, and 15% testing subsets. This ensured that anomaly detection accuracy was verified on unseen data. The temporal encoder was used to construct the graph architecture, mapping continuous operational profiles and machine statuses to predict localized cost variances. Model optimization was performed using the Adam optimizer with a learning rate of 0.001 and a batch size of 64, as shown in Table 5.

The latent features produced by the spatial graph and temporal gated layers were regularized with a dropout rate of 0.2, which reduces feature co-adaptation and enhances generalization. To balance spatial context with computational efficiency, the neighbor restriction parameter was set to 8, meaning each production line aggregated information only from its eight most relevant peers.

Overfitting was controlled using an early-stopping mechanism, which stopped training if the

Table 4
Identified anomalies in data.

Anomaly	Description	Number	Proportion of anomaly
Energy spike	Abnormal increase in energy consumption	986	25.77%
Downtime	Device shutdown anomaly	812	21.22%
Yield drop	Decreased yield rate	735	19.21%
Material waste	Abnormal material consumption	611	15.97%
Labor inefficiency	Decreased labor efficiency	421	11.00%
Mixed anomaly	Mixed anomaly	261	6.82%
Total	—	3826	100.00%

Table 5
Model training parameters.

Parameter	Value	Description
Time window	12 weeks	Defines the temporal span of input data, ensuring that seasonal and medium-term cost fluctuations are captured.
Number of hidden dimensions	128	Sets the size of latent feature representations, allowing the model to encode complex spatiotemporal patterns
Optimizer	Adam	Provides efficient gradient based optimization with adaptive learning rates, accelerating convergence.
Learning rate	0.001	Controls the step size during optimization, balancing convergence speed and stability
Batch size	64	Determines the number of samples processed per iteration, improving training efficiency and stability.
Neighbor restriction parameter	8	Limits each production line to its eight most functionally correlated peers, ensuring meaningful spatial aggregation without high computational cost
Dropout	0.2	Regularizes latent features by randomly deactivating neurons, reducing co-adaptation and improving generalization
Early stopping	AUC stops improving after 10 consecutive epochs.	Prevents overfitting by halting training when performance plateaus, ensuring convergence to a stable solution

validation set’s AUC failed to improve for 10 consecutive epochs. This prevented unnecessary computation and ensured that the model converged to a stable solution. These hyperparameters, including the 12-week time window, 128 hidden dimensions, Adam optimizer, learning rate of 0.001, and batch size of 64, form a transparent and reproducible training configuration. The parameters were set up to ensure that CF-TGNN effectively captures temporal dynamics and spatial correlations in production line cost data.

5.3 Model comparison

The performance of different models is compared in Table 6. The CF-TGNN model showed the highest overall performance, with a precision of 0.849, a recall of 0.832, an F1-score of 0.840, and an AUC of 0.963. These metric values exceeded those of other models, including conventional machine learning, sequence-based deep learning, spatiotemporal graph neural network, and dynamic graph neural network models.

XGBoost and RF showed F1-scores of 0.718 and 0.709 with AUC values of 0.841 and 0.835, respectively. Their relatively low performance is attributed to the treatment of heterogeneous multi-sensor measurements as static feature vectors, which prevents the models from capturing temporal dependencies and structural relationships among production lines. Consequently, important operational patterns associated with cost-variance anomalies are not adequately represented. Sequence-based deep learning models showed enhanced performance. LSTM and GRU showed F1-scores of 0.752 and 0.760, respectively, demonstrating their capability to capture temporal dependencies in production processes. However, they are unable to exploit cross-line operational similarities because these models treat each production line as an independent

Table 6

Performance comparison of different models for sensor data processing capability.

Model	Precision	Recall	F1-score	AUC
XGBoost	0.742	0.695	0.718	0.841
RF	0.731	0.688	0.709	0.835
LSTM	0.774	0.732	0.752	0.889
GRU	0.781	0.74	0.76	0.894
STGCN	0.812	0.784	0.798	0.923
T-GCN	0.819	0.791	0.805	0.931
DySAT	0.815	0.792	0.803	0.932
E-TGNN	0.823	0.8	0.811	0.938
CF-TGNN (this study)	0.849	0.832	0.84	0.963

sequence. As a result, localized sensor disturbances, such as temporary signal drifts, measurement noise, or communication errors, may be misinterpreted as genuine production anomalies.

Graph structures enable the enhancement of anomaly-detection performance. STGCN and T-GCN showed F1-scores of 0.798 and 0.805 with AUC values of 0.923 and 0.931, respectively. The results showed the benefits of integrating spatial relationships among production lines. Despite this, these models use statistical correlations within graph structures and do not incorporate manufacturing-domain knowledge or cost-information mechanisms. Consequently, their predictions do not satisfy physical operating conditions or economic cost-composition relationships.

The dynamic GNN models DySAT and E-TGNN showed F1-scores of 0.803 and 0.811 and AUC values of 0.932 and 0.938, respectively. These results indicate the effectiveness of dynamic graph learning in capturing evolving relationships among production-line nodes. By continuously updating graph representations over time, TGNN models reflect changing production conditions better than static graph architectures. Despite their superior representation-learning capability, these models are data-driven and do not identify physical and financial constraints. As a result, anomalies are identified using statistical deviations rather than operationally interpretable cost-generation mechanisms.

CF-TGNN consistently outperformed all benchmark models by integrating sensor-cost embeddings, temporal graph propagation, counterfactual baseline generation, and structural consistency constraints in a unified framework. Whereas conventional graph models classify node states, CF-TGNN generates a counterfactual normal-cost baseline representing expected operating conditions and then evaluates deviations from this baseline. This mechanism enhances robustness to localized sensor disturbances, preserving the physical and financial consistency of predicted states. The superior F1-score (0.840) and AUC (0.963) indicate that the integration of graph-based relational learning with manufacturing cost-information knowledge leads to a more reliable and interpretable solution for sensor-driven cost variance analysis than purely data-driven approaches.

5.4 Early warning capability and financial management implications

To evaluate the operational and predictive capability of the CF-TGNN model as a real-time early warning system, its capability to anticipate systemic cost-variance anomalies was compared with those of the other models. In manufacturing, sensor data anomalies, such as micro-stoppages, mechanical degradation, or material over-consumption, manifest well before they are formally compiled into delayed monthly or quarterly corporate financial ledgers. An experiment was conducted to quantify how many weeks in advance a front-end data-driven soft-sensor model can alert operators to downstream financial anomalies.

The CF-TGNN model achieved the longest average advance detection, identifying anomalies 2.36 weeks before occurrence, which outperformed machine-learning models and conventional deep-learning models (Table 7). It also showed high reliability under strict time-to-detection constraints, with a one-period early warning rate of 76.84%, a two-period rate of 58.19%, and the lowest missing alarm rate (17.10%) among all models. XGBoost and random forest models failed to capture sequential dependencies, achieving horizons of only 1.1 weeks and suffering from high missing alarm rates (>31%). They detected anomalies only after significant degradation is visible.

The LSTM and GRU models shortened the period to 1.7 weeks by tracking continuous process signatures. However, without spatial connectivity, localized sensor noise masks early fault signals, keeping missing alarm rates relatively high (24%). The STGCN, T-GCN, and CF-TGNN models extended the period to about 2 weeks by leveraging spatial context. The CF-TGNN model showed the longest period by integrating counterfactual digital-twin generation with structural conservation rules, suppressing environmental noise and detecting subtle, compounding variations across heterogeneous sensor streams. This yields the lowest missing alarm rate (17.10%) and the most reliable early-warning capability.

5.5 Ablation study results

To validate the effectiveness of each layer in the CF-TGNN architecture, an ablation experiment was conducted (Table 8). This analysis incrementally incorporated the four structural

Table 7
Early-warning capabilities of different models.

Model	Average detection period in advance (weeks)	One-period early warning rate (%)	Two-period early warning rate (%)	Missing alarm rate (false negative rate) (%)
XGBoost	1.12	54.36	32.15	31.25
Random Forest	1.05	51.84	29.4	33.1
LSTM	1.68	67.93	44.52	24.8
GRU	1.72	68.45	45.1	23.95
STGCN	1.94	71.2	50.85	20.15
T-GCN	2.01	72.8	52.1	19.42
CF-TGNN	2.36	76.84	58.19	17.1

Table 8
Ablation experiment results.

Model Variant	Sensor-cost embedding layer	Similarity graph layer	Counterfactual cost generation layer	Structural consistency loss layer	F1-score	AUC
LSTM (without all layers)	×	×	×	×	0.759	0.927
Without similarity graph layer	✓	×	×	×	0.787	0.939
Without counterfactual layer	✓	✓	×	×	0.812	0.949
Without consistency loss layer	✓	✓	✓	×	0.829	0.957
CF-TGNN (all layers included)	✓	✓	✓	✓	0.84	0.963

layers, sensor-cost embedding, similarity graph construction, temporal graph propagation with counterfactual cost generation, and structural consistency loss layers, to evaluate their individual and combined contributions.

The sensor-cost embedding layer improved the F1-score from 0.759 to 0.787 and the AUC from 0.927 to 0.939 by incorporating financial cost-structure information. This confirms that the integration of embedding sensor and cost features enhances anomaly detection accuracy. In the similarity graph layer, by adding production line similarity graphs, the F1-score increased to 0.812, demonstrating that cross-line spatial relationships provide reliable benchmarks for detecting cost variances. With the introduction of counterfactual modeling, the counterfactual cost generation layer increased the F1-score to 0.829, showing that identifying anomalies through deviations between actual and counterfactual costs is more effective than direct classification. Consistency between counterfactual and real cost structures yielded the highest performance (F1 = 0.840, AUC = 0.963) in the structural consistency loss layer, ensuring that anomaly prediction results align with financial decomposition rules, enhancing stability and interpretability.

The ablation study results show that the CF-TGNN model's superiority stems from the integration of all four layers rather than any single component. Its architecture follows a sequential design: sensor-cost embedding → similarity graph construction → counterfactual cost generation → structural consistency enforcement. This layered architecture enables the CF-TGNN model to capture physical shop-floor dynamics and financial cost structures, ensuring robust anomaly detection and meaningful cost interpretation.

5.6 Counterfactual cost deviation

A key output of the CF-TGNN framework is the counterfactual normal cost ($\hat{C}_{i,t}^{cf}$). The cost deviation ($D_{i,t}$) is calculated by comparing actual costs with counterfactual costs. When the actual costs of a production line's consistently exceed the counterfactual normal cost, the variance is not attributable to random fluctuations but instead reflects persistent inefficiencies or

structural issues in the production process. The results show that the average cost deviation for normal samples was 0.0068 and that for abnormal samples was 0.0513. Actual costs in manufacturing normal products fluctuated slightly around the counterfactual benchmark, whereas those in manufacturing abnormal products were 5.13% higher on average. This shows that cost deviation is an effective discriminator between normal and abnormal operating conditions (Table 9). The average gross margin of abnormal products (0.1395) was lower than that of normal products (0.1750). This indicates that cost variances exert a tangible impact on corporate profitability. From a financial perspective, higher cost deviations correspond to increased unit product costs, wider budget variances, and compressed gross profit margins.

The results presented in Fig. 3 show the managerial and operational significance of counterfactual cost deviation analysis. Figure 3(a) shows that actual costs generally track the

Table 9
Cost deviations of manufacturing normal and abnormal products.

Product type	Number	Average cost deviation	Ratio of average gross profit
Normal	31638	0.0068	0.1750
Abnormal	3826	0.0513	0.1395

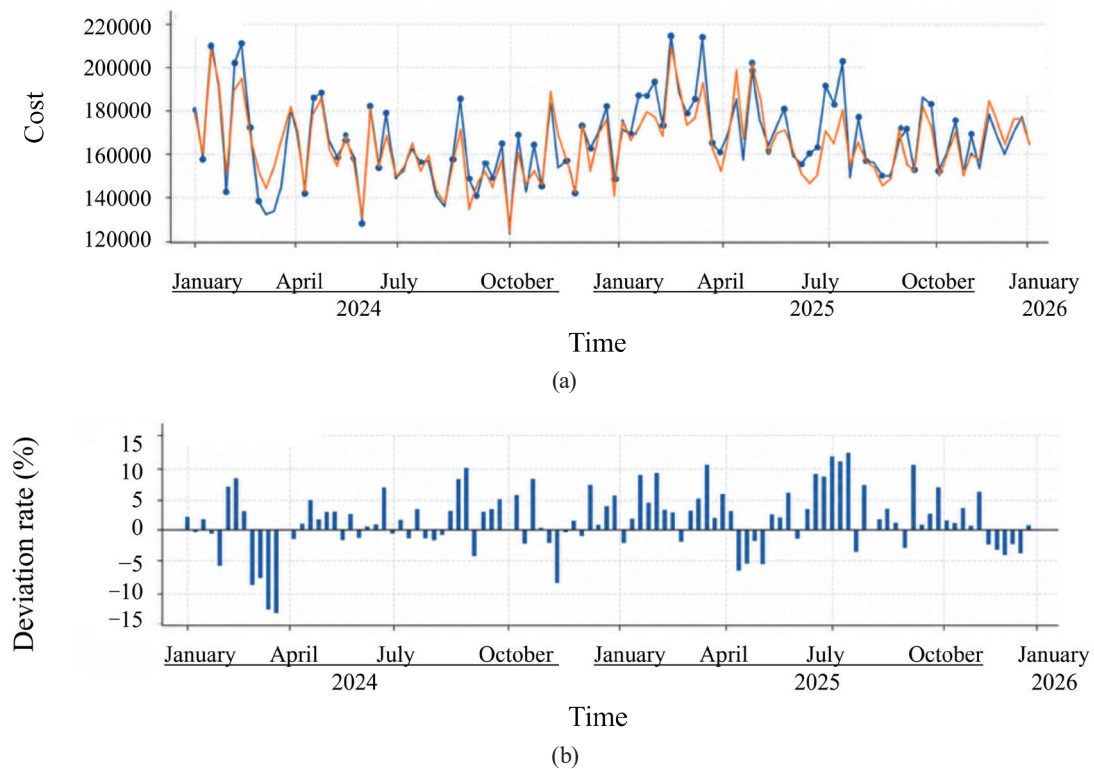


Fig. 3. (Color online) (a) Comparison of actual costs (blue line) with counterfactual normal costs (orange line) and (b) deviation rate (blue dot: abnormal cost observed).

counterfactual benchmark. Persistent divergences in abnormal weeks reveal inefficiencies that are structural rather than stochastic. Figure 3(b) shows these divergences, indicating that abnormal samples deviate by an average of 5.13% compared with 0.68% for normal samples. From a managerial perspective, this result enables a diagnostic tool: deviations that consistently exceed the counterfactual baseline signal underlying process inefficiencies that warrant targeted intervention. Operationally, the capability to identify abnormal cost behavior enables managers to prioritize corrective actions, such as refining resource allocation, adjusting production scheduling, or redesigning workflows to mitigate inefficiencies. Financially, the relationship between higher deviations and compressed gross margins highlights the tangible impact of operational inefficiency on profitability. Therefore, counterfactual cost deviation analysis discriminates between normal and abnormal operating conditions and equips decision-makers with actionable insights to improve cost control, enhance process stability, and safeguard corporate margins.

5.7 Robustness

To ensure the reliability of the model performance, three robustness tests were conducted.

1. Alternative anomaly test: The original anomaly criteria (based on 1.5 SD and budget deviation thresholds) were replaced with definitions using the top 10 or 15% cost deviation percentile.
2. Graph adjustments: Cosine similarity was substituted with Gaussian kernel similarity, and the neighbor restriction parameter was varied to test stability.
3. Industry-specific subsample validation: Model performance was tested across equipment manufacturing, food processing, and agricultural product processing sectors.

When the anomaly definition was modified to use the top 10 or 15% cost deviation percentile, the CF-TGNN model maintained stable performance, with F1-scores of 0.831 and 0.846 and AUC values of 0.958 and 0.961, respectively (Table 10). This confirms that the model is not overly sensitive to the choice of anomaly labeling criteria. When changing the graph initialization step by substituting a Gaussian kernel graph, the F1-score and AUC were 0.835 and 0.960,

Table 10
Robustness test results.

Verification method	F1-score	AUC
Change anomaly to top 10% cost deviation percentile	0.831	0.958
Change anomaly to top 15% cost deviation percentile	0.846	0.961
Gaussian kernel graph	0.835	0.96
Neighbor restriction parameter = 5	0.832	0.957
Neighbor restriction parameter = 10	0.841	0.964
Equipment manufacturing	0.836	0.959
Food processing	0.828	0.954
Agricultural product processing	0.821	0.951

respectively. These values were identical to the baseline performance, proving that the temporal propagation layer remains robust under different topological definitions. When adjusting the neighbor restriction parameter from 5 (a sparse setting) to 10 (a dense setting), a negligible performance change was observed. This stability shows the model's spatial feature-aggregation capability from neighboring nodes without introducing localized noise or suffering from oversmoothing. The cross-industry evaluation result showed strong domain generalization across distinct manufacturing sectors. For agricultural product processing, which typically features variable material inputs and cyclical processing, the model achieved an F1-score of 0.821 and an AUC of 0.951. This performance indicates that the CF-TGNN model avoided overfitting to any single factory environment by structuring deep learning layers with physical process rules, such as mass balance and energy conservation constraints. The model operates as a reliable, physics-bounded soft-sensor framework across diverse smart manufacturing environments.

6. Discussion

6.1 Significance of CF-TGNN

GNNs have been applied to capture nonlinear spatial interactions across industrial layouts, reducing manual parameter tuning.⁽³⁴⁾ Temporal attention mechanisms dynamically weight historical sensor states to adapt to changes in production modes.⁽³⁵⁾ CF-TGNN is built on the advances by integrating spatial dependencies with counterfactual baseline generation, providing a steady-state reference for variance analysis. Statistical models are criticized for their black-box nature. Recently, conservation laws have been embedded into loss functions,⁽³⁶⁾ showing that boundary conditions such as mass balances or thermodynamic constraints improve stability and prevent unrealistic outputs.⁽³⁷⁾ We extended the concept by demonstrating that financial cost-decomposition metrics can serve as macro-level boundary constraints, linking physical anomalies to economic outcomes.

6.2 Industrial implications and financially aware manufacturing

Managing diverse shop-floor data requires robust real-time data fusion. Edge-computing frameworks are used to process vibration and acoustic telemetry locally before transmitting aggregated states.⁽³⁸⁾ Multi-modal data fusion methods are used to synchronize heterogeneous protocols, such as Modbus, OPC UA, and Ethernet Vision, to ensure temporal consistency across events.⁽³⁹⁾ CF-TGNN validates this approach by showing that combining enterprise data with continuous edge telemetry produces a traceable diagnostic path from machine components to the financial balance sheet. CF-TGNN demonstrates how sensor technology is used in intelligent manufacturing. When embedded in advanced architectures, sensors become part of a larger diagnostic and financial management system. With the fusion of sensor data with physics-based rules and enterprise metrics, CF-TGNN shows how sensor networks can evolve into intelligent, financially aware systems that guide proactive decision-making across the factory floor.

The results of this study confirm the effectiveness of the CF-TGNN model as a high-fidelity soft sensor framework. More than achieving strong predictive accuracy, the model shows how sensor technology can be repositioned in intelligent manufacturing: from isolated measurement devices to integrated diagnostic systems that connect shop-floor telemetry directly to enterprise-level financial outcomes.

Traditional soft sensors often operate as stand-alone tools, translating local measurements into narrow outputs such as fault classes or temperature estimates. They rarely account for the consequences of multi-sensor deviations across an entire facility. CF-TGNN addresses this issue by converting sensor data to industrial metrics. This shows the increasing role of sensor technology in intelligent manufacturing systems that are proactive, financially aware, and resilient to real-world disturbances.

The developed model represents a physics-informed, domain-constrained sensor architecture. Conventional deep neural networks and spatiotemporal graph convolution models are vulnerable to hardware drift, electromagnetic noise, and packet loss. Under these conditions, standard models accumulate errors or generate unrealistic machine states, limiting their reliability in industrial environments. In contrast, CF-TGNN mitigates the limitation through its sensor–cost embedding layer and structural consistency loss. By mapping raw signals such as power consumption streams or register state changes into vector spaces governed by engineering conservation rules, the model enforces physical boundaries within its learning process. If one sensor drifts, the spatial propagation mechanism draws context from neighboring production lines to reconstruct the missing state. This resilience transforms the soft sensor from a simple pattern-matching tool into a self-correcting diagnostic system, strengthening the role of sensor technology in intelligent manufacturing.

Multi-sensor data fusion, combined with advanced graph topologies derived from that fusion, is essential for modern manufacturing analytics. Failures in automated production are rarely caused by a single machine or a clear threshold violation. Instead, combined small anomalies, such as tool wear, voltage drops, or micro-stoppages, reduce efficiency, although they appear normal in isolation. Detecting these subtle anomalies requires dense deployments of sensors operating with soft sensors to interpret spatial relationships. The sensor topology graph used in CF-TGNN shows that production lines must not be modeled as isolated streams. By mapping similarities across production lines, the network detects variations invisible to single-line models. This spatial context enables CF-TGNN to recognize 2.36 weeks of anomalies, demonstrating how sensor networks can evolve from reactive fault logging to proactive operational control in intelligent manufacturing.

6.3 Performance comparison

DySAT and E-TGNN enable adaptive representation learning by capturing temporal changes in graph topology, showing enhanced performance in various graph-based prediction tasks.^(5,13) However, their application is limited to industrial soft sensing, and manufacturing cost variance analysis using them is limited because they primarily rely on statistical correlations among

graph nodes. In complex manufacturing environments, operational states are governed by temporal and spatial dependencies as well as by physical and economic constraints. Standard temporal GNNs are not incorporated into manufacturing cost-information mechanisms or conservation principles. Therefore, model predictions might not be within operational boundaries or cost estimates might not align with the underlying relationships among material, labor, energy, and overhead components. Their learned node representations are sensitive to localized sensor disturbances, such as signal drift, transient noise, calibration errors, or communication interruptions, which propagate through the graph and adversely affect anomaly-detection performance.

The CF-TGNN model addresses these limitations through the integration of sensor-cost embeddings, counterfactual baseline generation, and the structural consistency loss term. Rather than classifying operational states, CF-TGNN generates a counterfactual steady-state cost baseline that represents expected operating conditions under normal production behavior. In the model, anomalies are identified on the basis of deviations from this counterfactual reference to distinguish genuine process abnormalities from transient sensor disturbances, maintaining consistency with manufacturing cost-information logic.

CF-TGNN's superior performance to other models indicates that domain-specific manufacturing knowledge integration into temporal graph learning enables advantages beyond dynamic topology modeling. Through the combination of counterfactual reasoning and structural cost-conservation constraints, CF-TGNN has greater robustness, interpretability, and predictive reliability for sensor-driven cost variance analysis in intelligent manufacturing environments.

6.4 Sensor modalities and materials

The CF-TGNN model is used for feature-attribution sensitivity analysis and the formulation of guidance for sensor hardware engineers and material scientists.⁽¹⁶⁾ By isolating operational states that show uncertainty in cost variance, the CF-TGNN model enables the identification of parameters that show low signal-to-noise ratios or measurement fluctuations under harsh industrial conditions. For instance, localized energy spikes caused by friction-induced thermal losses indicate the need for advanced thin-film thermal sensors or high-temperature piezoelectric materials capable of continuous inline monitoring in dynamic mechanical environments.⁽¹⁷⁾ Soft sensing transitions from a passive monitoring tool to an active diagnostic engine that leads to the determination of design criteria for functional materials and smart sensors.

As edge computing, multi-modal sensor data fusion, and wireless sensor networks are widely applied in smart manufacturing, soft-sensor architectures require proof-of-concept. CF-TGNN enables the decoupling of hardware interfaces from graph representation learning. Advanced hardware, such as embedded edge-AI camera modules, high-frequency acoustic emission transducers, or millimeter-wave spatial tracking systems, must be integrated into domain-constrained smart manufacturing.⁽¹⁸⁾ The spatiotemporal graph structure of the model can be used to adjust adjacency weights to incorporate new high-frequency data streams without requiring structural model redesign.

6.5 Cross-industry applicability

In soft-sensor development, applicability and generalizability in different industrial sectors must be considered.⁽⁴⁰⁾ A model specifically designed for a single production environment has less flexibility in operating conditions and measurements, but model training becomes easier, and anomaly-detection accuracy can be increased. However, the multi-industry design provides important advantages. First, the similarity graph is constructed according to operational characteristics derived from sensor data rather than predefined industrial characteristics. This enables the model to identify common production behaviors across different sectors. For example, manufacturing processes with similar energy-consumption patterns or equipment-utilization characteristics can share useful information even if they belong to different industries.

A diverse graph structure enhances robustness against sensor disturbances. When a production line experiences temporary sensor drifts, hardware malfunction, or communication noise, the model uses information from other production lines with similar operating conditions to compensate for the corrupted measurements. This peer manufacturing line data-based learning enables the reduction of the impact of localized sensing errors and improves detection reliability. The model that can be applied to various industries enhances deployment flexibility. Instead of developing separate soft-sensor models for each manufacturing sector, the CF-TGNN model enables a unified digital-twin architecture that can be applied across different industrial environments. The model can be adapted to specific applications through transfer learning or fine-tuning, reducing development costs while retaining strong predictive performance.

The robustness of the developed model in this study supports such flexibility. CF-TGNN maintains consistently high performance across equipment manufacturing, food processing, and agricultural product processing. The model also balances domain-specific knowledge with cross-industry generalization, enabling reliable sensor-driven cost variance analysis across diverse manufacturing environments.

6.6 Limitations and model deployment

Although the model showed high detection accuracy, online, real-time deployment testing must be conducted. Model performance results using historical data might have constraints in live shop-floor implementation.⁽⁴¹⁾ In the industrial IoT, communication jitter, network congestion, and temporary packet loss occur since different protocols, such as the Modbus remote terminal unit, OPC UA, and Gigabit Ethernet for machine vision, are used. Although the model enables the reduction of missing data, interruptions in data transmission degrade soft-sensor performance.

In manufacturing systems, production lines frequently undergo updates, product switching, or layout reconfigurations. Such structural changes alter baseline cost-information dynamics, requiring adaptive online retraining mechanisms to prevent false alarms. The inference time of CF-TGNN (14.2 ms per graph forward pass) is sufficiently fast for real-time monitoring. However, large-scale TGNN must be applied to edge gateways or PLCs through considerable

compression in edge-cloud collaborative processing architectures.⁽⁴²⁾ Therefore, the CF-TGNN model must be tested in MES to evaluate stability in data transmission, online learning, and alert response times.

7. Conclusion

A CF-TGNN model is developed in this study for sensor-data fusion, cost-variance analysis, and anomaly detection in manufacturing production lines. The model integrates domain-constrained sensor–cost embedding, production-line similarity graph construction, temporal graph propagation, and counterfactual normal-cost generation. By converting heterogeneous sensor measurements into structured material, labor, energy, and overhead cost representations, the model connects operational conditions with financial outcomes. Its counterfactual mechanism estimates normal production costs, and structural consistency constraints ensure that the predicted cost components remain consistent with manufacturing cost relationships.

Experiment results using 35,464 weekly observations from 341 production lines across 64 companies showed that the CF-TGNN model achieved an F1-score of 0.840 and an AUC of 0.963, outperforming machine learning, deep learning, and graph neural network models. The ablation study results indicated that sensor–cost embedding, similarity graph construction, counterfactual cost generation, and structural consistency constraints contributed to the overall detection performance. Robustness tests showed that the model maintained competitive performance under alternative anomaly definitions and graph configurations and across selected manufacturing sectors. CF-TGNN enabled an average early-warning horizon of 2.36 weeks, with one-period and two-period early-warning rates of 76.84 and 58.19%, respectively. Its missing-alarm rate was 17.10%, the lowest among the models evaluated in this study. In addition, the differences in average cost deviation between normal and abnormal observations indicate that counterfactual cost comparisons enable the accurate detection of operating fluctuations that lead to costly production inefficiencies. The results support the use of domain-constrained temporal graph learning for interpretable, sensor-driven cost monitoring and early warning. The model needs further development to be deployed in different intelligent manufacturing systems that connect shop-floor sensing with enterprise-level cost management.

Acknowledgments

This research was funded by the Basic Scientific Research Business Project for Provincial Undergraduate Universities of Heilongjiang "Research on the Construction of a Digital-Intelligent Financial System for Manufacturing Enterprises in Heilongjiang under New Quality Productive Forces" (Project No. 2026-KYYWF-E015).

References

- 1 L. Kong, X. Peng, Y. Chen, P. Wang, and M. Xu: *Int. J. Extreme Manuf.* **2** (2020) 022001. <https://doi.org/10.1088/2631-7990/ab7ae6>
- 2 J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu: *J. Manuf. Syst.* **48** (2018) 144. <https://doi.org/10.1016/j.jmsy.2018.01.003>
- 3 J. Lee, Y. C. Lee, and J. T. Kim: *J. Manuf. Syst.* **57** (2020) 357. <https://doi.org/10.1016/j.jmsy.2020.10.013>
- 4 G. Pang, C. Shen, L. Cao, and A. Van Den Hengel: *ACM Comput. Surv.* **54** (2021) 38. <https://doi.org/10.1145/3439950>
- 5 Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu: *IEEE Trans. Neural Networks Learn. Syst.* **32** (2021) 4. <https://doi.org/10.1109/TNNLS.2020.2978386>
- 6 W.-H. Choi and J. Kim: *Appl. Syst. Innov.* **7** (2024) 18. <https://doi.org/10.3390/asi7020018>
- 7 P. Ding, S. Q. Liu, R. Chiong, S. Dhakal, D. Chen, D. Li, H.-L. Ma, and S.-H. Chung: *Comput. Ind.* **174** (2026) 104398. <https://doi.org/10.1016/j.compind.2025.104398>
- 8 J. Dai and M. A. Vasarhelyi: *J. Emerg. Technol. Account.* **20** (2023) 1. <https://doi.org/10.2308/JETA-2023-009>
- 9 K. Abbas: *Br. Account. Rev.* **58** (2026) 101551. <https://doi.org/10.1016/j.bar.2025.101551>
- 10 J. Ou and Y. Wang: *Chemom. Intell. Lab. Syst.* **271** (2026) 105656. <https://doi.org/10.1016/j.chemolab.2026.105656>
- 11 S. Latif, M. Driss, W. Boulila, Z. E. Huma, S. S. Jamal, Z. Idrees, and J. Ahmad: *Sensors* **21** (2021) 7518. <https://doi.org/10.3390/s21227518>
- 12 J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun: *AI Open* **1** (2020) 57. <https://doi.org/10.1016/j.aiopen.2021.01.001>
- 13 H. Qiao, H. Tong, B. An, I. King, C. Aggarwal, and G. Pang: *IEEE Trans. Knowl. Data Eng.* **37** (2025) 5106. <https://doi.org/10.1109/TKDE.2025.3581578>
- 14 Y. Liu, S. Garg, J. Nie, Y. Zhang, Z. Xiong, J. Kang, and M. S. Hossain: *IEEE Internet Things J.* **8** (2021) 6348. <https://doi.org/10.1109/JIOT.2020.3011726>
- 15 H. Nizam, S. Zafar, Z. Lv, F. Wang, and X. Hu: *IEEE Sens. J.* **22** (2022) 22836. <https://doi.org/10.1109/JSEN.2022.3211874>
- 16 Y. Jiang, S. Yin, J. Dong, and O. Kaynak: *IEEE Sens. J.* **21** (2021) 12868. <https://doi.org/10.1109/JSEN.2020.3033153>
- 17 N. Reeh, G. Manthei, and P. J. Klar: *Appl. Syst. Innov.* **7** (2024) 39. <https://doi.org/10.3390/asi7030039>
- 18 E. Karapidakis, G. Konstantinidis, N. Vidakis, and S. Yfanti: *Appl. Syst. Innov.* **5** (2022) 78. <https://doi.org/10.3390/asi5040078>
- 19 V. V. Shenvi and A. Sharma: *Res. Intell. Manuf. Assem.* **4** (2025) 200. <https://doi.org/10.25082/RIMA.2025.01.007>
- 20 M. Mohammadagha, F. Bigdeli, S. Sharifi, M. Deldadehasl, and S. Ataei: *Preprints* (2025) 2025110501. <https://doi.org/10.20944/preprints202511.0501.v1>
- 21 A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala: *Proc. 33rd Int. Conf. Neural Information Processing Systems* (2019) 8026–8037. <https://dl.acm.org/doi/10.5555/3454287.3455008>
- 22 T. Chen and C. Guestrin: *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (ACM, 2016)* 785–794. <https://doi.org/10.1145/2939672.2939785>
- 23 L. Breiman: *Mach. Learn.* **45** (2001) 5. <https://doi.org/10.1023/A:1010933404324>
- 24 S. Hochreiter and J. Schmidhuber: *Neural Comput.* **9** (1997) 1735. <https://doi.org/10.1162/neco.1997.9.8.1735>
- 25 K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio: *Proc. 2014 Conf. Empirical Methods in Natural Language Processing (Association for Computational Linguistics, 2014)* 1724–1734. <https://doi.org/10.3115/v1/D14-1179>
- 26 B. Yu, H. Yin, and Z. Zhu: *Proc. 27th Int. Joint Conf. Artificial Intelligence (AAAI Press, 2018)* 3634–3640. <https://dl.acm.org/doi/10.5555/3304222.3304273>
- 27 D. Oliveira, J. Pinheiro, L. Neto, V. H. Pinto, and G. Gonçalves: *Sensors* **22** (2022) 7694. <https://doi.org/10.3390/s22197694>

- 28 D. Calderón, F. J. Folgado, I. González, and A. J. Calderón: *Sensors* **24** (2024) 8074. <https://doi.org/10.3390/s24248074>
- 29 C. Li, Y. Li, Q. Song, J. Gao, H. Duan, and Q. Zeng: *Blockchain: Res. Appl.* (2026) 100553. <https://doi.org/10.1016/j.bcr.2026.100553>
- 30 T. Saito and M. Rehmsmeier: *PLoS One* **10** (2015) e0118432. <https://doi.org/10.1371/journal.pone.0118432>
- 31 J. Snoek, H. Larochelle, and R. P. Adams: *Proc. Advances in Neural Information Processing Systems* 25 (Curran Associates, 2012) 2951–2959. https://papers.nips.cc/paper_files/paper/2012/hash/05311655a15b75fab86956663e1819cd-Abstract.html
- 32 Y. Jiang, S. Yin, J. Dong, and O. Kaynak: *IEEE Sens. J.* **21** (2021) 12868. <https://doi.org/10.1109/JSEN.2020.3033153>
- 33 J. Dembski, B. Wiszniewski, and A. Kołakowska: *Sensors* **25** (2025) 5526. <https://doi.org/10.3390/s25175526>
- 34 J. Long, B. Wang, H. Peng, and H. Zhang: *Adv. Eng. Inf.* **71** (2026) 104395. <https://doi.org/10.1016/j.aei.2026.104395>
- 35 L. Zhou, H. Wang, and S. Xu: *Reliab. Eng. Syst. Saf.* **262** (2025) 111152. <https://doi.org/10.1016/j.res.2025.111152>
- 36 M. Faegh, S. Ghungrad, J. P. Oliveira, P. Rao, and A. Haghighi: *J. Manuf. Processes* **133** (2025) 524. <https://doi.org/10.1016/j.jmapro.2024.11.066>
- 37 J. Rao, T. Yang, C. Ji, C. Han, J. Wang, and W. Sun: *Chem. Eng. Sci.* **327** (2026) 123602. <https://doi.org/10.1016/j.ces.2026.123602>
- 38 G. Spencer, P. M. B. Torres, V. H. Pinto, and G. Gonçalves: *Machines* **14** (2026) 270. <https://doi.org/10.3390/machines14030270>
- 39 R. Elsas, D. Van Leemput, J. Hoebeke, and E. De Poorter: *Wireless Pers. Commun.* **140** (2025) 1. <https://doi.org/10.1007/s11277-023-10213-w>
- 40 J. Zhu, Y. Wang, W. Zhu, Y. Liu, and C. Yang: *Chem. Eng. Sci.* **331** (2026) 123969. <https://doi.org/10.1016/j.ces.2026.123969>
- 41 X. Wu, C. Pan, M. Cao, and K. Sun: *Appl. Syst. Innov.* **9** (2026) 175. <https://doi.org/10.3390/asi9090175>
- 42 X. Chen, H. Wang, X. Yang, H. Huang, and C. Wei: *Discover Internet Things* **6** (2026) 127. <https://doi.org/10.1007/s43926-026-00356-0>