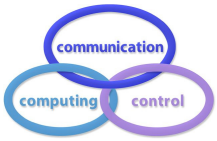
INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 21, Issue: 6, Month: December, Year: 2026
Article Number: 7694, https://doi.org/10.15837/ijccc.2026.6.7694



CCC Publications



A dual-stream multitask deep learning model for supply chain demand forecasting with statistically derived early-warning signals

K. Park

Kyoungjong Park
Department of Business Administration
Gwangju University, South Korea
Corresponding author: kjpark@gwangju.ac.kr

Abstract

Accurate demand forecasting is crucial in supply chain management (SCM) as global supply networks become increasingly complex and exposed to disruptions. Traditional models that rely mainly on historical sales may miss demand shifts driven by heterogeneous auxiliary signals and often do not provide explicit early warning of abnormal events. This study proposes a dual-stream multitask deep-learning model that integrates demand forecasting and statistically derived early-warning signal estimation within a unified framework. The model adopts a dual-input architecture: (i) an LSTM branch that captures temporal dynamics from historical sales sequences and (ii) a Transformer encoder branch that models supplementary auxiliary features and statistically derived anomaly signals computed from that same sales series, including STL residual–based indicators, change-point flags, and calendar variables. The fused representation is jointly optimized to predict both the demand two days ahead of the input window and the probability of a statistically defined anomaly event on that day. Experiments are conducted on the public Kaggle Store Item Demand Forecasting dataset (10 stores, 50 items, daily sales from 2013 to 2017), using the most recent 90 days in each series for validation and synthetic anomaly labels derived from statistical residual analysis. Compared with a unimodal LSTM baseline, the proposed model reduces RMSE by approximately 7.23% and MAE by approximately 6.86%, and the bootstrap confidence intervals of both gaps exclude zero. Under an identical evaluation protocol the model also outperforms a naive forecast, a seasonal naive model and AutoARIMA, whereas a global gradient-boosted tree baseline attains a slightly lower point error (RMSE 0.5574 versus 0.5798). Because ground-truth incident labels are unavailable for this dataset, the classification task is supervised by weak labels derived from the residuals of the same series; the resulting discrimination is limited (ROC–AUC between 0.578 and 0.589 and PR–AUC between 0.126 and 0.141 against a positive rate of 0.100), and the anomaly head is therefore presented as a calibrated, cost-aware screening signal rather than as a validated anomaly detector. Under a 5:1 false-negative-to-false-positive cost ratio the proposed model attains the lowest expected cost among the configurations compared, and temperature scaling yields well-calibrated alert probabilities, so that a single multitask model can support both replenishment decisions and screening-level early warning.

Keywords: Supply Chain Management (SCM), Multitask Learning, Anomaly Detection, Long Short-Term Memory (LSTM), Transformer Encoder, Early-warning

1 Introduction

The importance of accurate demand forecasting has intensified as the global supply chain environment has become increasingly uncertain. Forecasting errors can lead to stockouts or excess inventory, increasing costs and reducing service levels for firms. Unexpected disruptions, exemplified by the COVID-19 pandemic, triggered abrupt shifts in consumer behavior and caused significant logistical delays, making demand forecasting highly challenging [20]. Against this backdrop, recent work increasingly leverages artificial intelligence (AI), particularly deep learning, to predict demand from complex supply chain data. Deep learning models learn complex, nonlinear patterns from large-scale datasets and often outperform traditional statistical forecasting methods. Recurrent neural network (RNN) architectures, such as LSTM networks, have been proposed to capture complex patterns in time-series demand data and improve forecasting accuracy [11].

Supply chain operational data comprises multiple modalities beyond sales history, including promotions, weather conditions, economic indicators, and social media streams. Consequently, there is growing interest in multimodal learning approaches that integrate these diverse data sources within forecasting models. Leveraging deep learning, demand forecasting models can jointly exploit visual quality cues and textual information from these heterogeneous data sources. Indeed, prior studies detect product defects from images while forecasting sales and use social media data to capture changes in consumer trends and incorporate them into demand forecasts [13, 43, 46].

From a supply chain management perspective, early detection of anomaly signals is essential to minimize potential losses. Here, an anomaly signal denotes deviations from expected operating conditions, such as sudden demand surges or drops, production disruptions, delivery delays, or quality issues, and identifying such signals in advance enables proactive responses. Traditionally, anomaly detection has been conducted separately from demand forecasting, using statistical quality control techniques or outlier-detection models. More recently, researchers have attempted to perform forecasting and anomaly detection in a unified manner [24]. Deep learning models can automatically identify supply chain disruptions by learning complex patterns from large volumes of sensor data and operational indicators. For example, Tang et al. [37] proposed a deep learning model to automatically detect industrial chain disruption events in China's manufacturing supply chain, and various studies have demonstrated the potential of machine learning for supply chain anomaly detection [2, 29]. However, when forecasting and anomaly detection are conducted separately, models may under-capture demand fluctuations under anomalous regimes, leading to larger forecasting errors. Therefore, integrated forecasting models that incorporate anomaly-signal information into the predictor are needed to improve demand-forecasting accuracy while simultaneously detecting anomalies.

This study proposes a dual-stream multitask deep learning framework that performs both supply chain demand forecasting and statistically derived early-warning signal estimation within a single model. The proposed model takes as input the historical sales time series (primary stream) together with auxiliary variables and anomaly-signal time series (auxiliary stream) and performs regression to predict future demand while concurrently conducting a binary classification to determine whether an anomalous pattern is present in the series. It should be emphasised that every auxiliary input is a deterministic statistical transformation of the sales series itself; the architecture is therefore a single-source, multi-feature (dual-stream) model rather than a multimodal model in the strict sense of fusing heterogeneous data types. The multimodal studies reviewed in Section 2.2 motivate the design, and extending it to genuinely external data sources is left to future work. Following the standard taxonomy of multimodal machine learning [1], a modality is the channel through which a phenomenon is recorded, so that genuinely multimodal learning fuses heterogeneous channels such as text, images and sensor readings. By that definition the two input streams used here belong to a single modality represented by two feature groups, and the term "multimodal" is retained in this paper only where it refers to the prior literature. This integrated design enables the model to learn both normal and anomalous regimes, maintaining robust forecast performance when anomalies occur.

The main contributions are threefold.

(1) This study proposes a dual-input, multitask model tailored to the supply chain domain, in which the sales time series and auxiliary/anomaly signals are processed in parallel branches and then

fused, enabling demand forecasting and anomaly-signal classification to be jointly learned in a single network.

(2) Using a publicly available multi-location, multi-product sales dataset, this study empirically demonstrates a statistically significant reduction in forecasting error relative to a unimodal LSTM baseline, benchmarks the model against external classical and machine-learning baselines under an identical protocol, and reports transparently that the gains on the auxiliary classification task are limited under weak supervision.

(3) This study analyses both the advantages and the limitations of dual-stream multitask integration by interpreting the experimental results of the integrated model, including cost-sensitive threshold selection and probability calibration, and discusses the feasibility of applying the approach as a screening-level early-warning layer in real-world SCM environments.

Because public datasets rarely include ground-truth anomaly labels, this study generates auxiliary variables and anomaly signals directly from the sales time series using statistical techniques. Specifically, this study constructs anomaly-signal features and auxiliary inputs from a robust z-score–based anomaly intensity derived from STL-decomposition residuals; change-point flags obtained with the Ruptures package; calendar variables (day-of-week, month, end-of-month, weekend indicators); and a 7-day rolling sales-volatility index [8, 39]. This procedure yields reproducible anomaly-signal data without requiring separate labels.

The remainder of this study is organized as follows. Section 2 reviews related work on demand forecasting and anomaly detection, Section 3 describes the proposed model architecture and training method, Section 4 presents the experimental setup and results, Section 5 discusses the implications of the main findings, and Section 6 concludes the study with directions for future research.

2 Literature review

2.1 Deep learning approaches for demand forecasting

Research on supply chain demand forecasting has traditionally relied on time-series statistical models such as ARIMA and exponential smoothing. However, these approaches are limited in capturing complex nonlinear patterns and multidimensional drivers. With advances in machine learning and deep learning, data-driven methods that automatically learn demand patterns from large-scale datasets have gained prominence [22]. In particular, LSTM-based RNNs are known to model long-range temporal dependencies effectively, thereby improving forecasting accuracy. Douaioui et al. [11] reported that LSTM models reduced forecasting error by approximately 15-20% compared with classical statistical techniques, and Ni et al. [31] showed that Bi-LSTM outperformed unidirectional LSTM. While these studies demonstrate that deep learning can finely capture complex demand volatility, they also highlight drawbacks such as the need for large training data, high computational cost, and limited interpretability [26]. Table 1 summarizes representative approaches and trade-offs.

Table 1: Comparison of demand forecasting methods

Approach	Key methods	Strengths	Limitations	Representative studies
Traditional statistical models	ARIMA, exponential smoothing	Simplicity; fast computation	Limited in capturing nonlinear and multidimensional factors	[22]
Classical ML models	Linear regression, support vector machine (SVM), random forest	Can incorporate multiple features	Weak in handling long-term dependencies	[26]
Deep learning (RNN/LSTM)	LSTM, Bi-LSTM	Captures long-term dependencies; improved accuracy	Requires large datasets, high computation, and has interpretability constraints	[11, 31]

2.2 Multimodal and auxiliary-information approaches

Research that integrates multimodal data and auxiliary factors has also increased, as actual demand is shaped by promotions, weather, consumer behavior, social media, and other external drivers. Khedr and Rani [20] noted that combining an image-based anomaly-detection model with an LSTM forecaster can improve both demand prediction and quality-related anomaly handling, while Ni et al. [31] leveraged weather and temporal information for fresh-food sales forecasting to enhance performance. Although multimodal learning has strong potential to reflect complex causal structures, it raises challenges in data consistency (scale differences and synchronization), model complexity, training cost, and explainability [45]. Table 2 outlines typical multimodal configurations in SCM, their benefits, and implementation challenges.

Table 2: Applications of multimodal learning in SCM

Data-type combination	Application example	Benefits	Challenges	Representative studies
Time-series + weather	Sales volume + weather conditions	Better capture of demand fluctuations; accuracy gains	Data alignment and consistency issues	[31]
Time-series + image	Sales volume + quality inspection (CCTV)	Simultaneous defect detection and demand forecasting	Cross-modal synchronization requirements	[20]
Time-series + social	Sales volume + consumer reviews/trends	Early detection of consumer-pattern shifts	Noisy and unstructured data	[45]

2.3 Anomaly detection in supply chains

In anomaly detection for SCM, post hoc methods based on Statistical Process Control (SPC) and classical outlier detection have been prevalent. More recently, real-time anomaly detection using deep learning has gained traction. Tang et al. [37] demonstrated automatic monitoring of disruption events in China's manufacturing supply chains with a deep learning model, and de la Cruz Madrigal et al. [9] improved anomaly-detection accuracy using a parallel neural architecture on simulated disruption data. However, most such studies either execute anomaly detection separately from demand forecasting or remain limited to simulation. By contrast, research in other domains, including power systems and predictive maintenance, has proposed multitask neural network architectures that jointly learn forecasting and anomaly detection within a single framework [35]. In the SCM domain, fully integrated models remain scarce. Table 3 summarizes major research streams and their limitations. In contrast to prior work that treats the two tasks separately, this study processes demand forecasting and anomaly detection together within one dual-stream multitask deep learning architecture; Section 2.4 positions this design within the literature on integrated forecasting and anomaly detection frameworks.

Table 3: Research streams in anomaly detection and their limitations

Approach	Key methods	Characteristics	Limitations	Representative studies
Traditional SPC	Statistical Process Control	Simple and fast	Low sensitivity to abnormal patterns	[5]
Independent DL models	Convolutional neural network (CNN) / RNN-based anomaly detection	Enables real-time automatic detection	Operated separately from demand forecasting	[9, 37]
Integrated forecasting models	Combined demand forecasting + anomaly detection	Maintains robust prediction under disruptions	Higher model complexity; limited interpretability	This study

2.4 Integrated forecasting and anomaly detection frameworks

Work that couples forecasting and anomaly detection can be grouped into three patterns. In the forecast-then-threshold pattern, a model is trained to predict normal behaviour and an alert is raised when the prediction residual exceeds a threshold; Hundman et al. [16] combine LSTM forecasters with nonparametric dynamic thresholding for spacecraft telemetry, and reconstruction-based encoder–decoder variants follow the same logic [28]. In the second pattern, two separate models are deployed side by side: in a supply chain setting, Nguyen et al. [30] pair an LSTM sales forecaster with an LSTM-autoencoder and one-class support vector machine detector for fashion retail, which improves decision support but shares no representation between the two tasks. The third pattern is joint multitask learning, in which a single network with a shared representation produces both outputs, following the general premise that related tasks act as mutual inductive biases [4]; Seman et al. [35] instantiate this pattern for predictive maintenance, and Liu et al. [24] outline an integrative supply chain framework that couples forecasting, anomaly detection and resource allocation. Table 4 summarises these studies along the dimensions that matter for the present work.

Two observations follow. First, joint multitask models remain scarce in demand-oriented SCM applications: the closest supply chain study keeps the two tasks separate [30], while the integrated architectures reported so far originate in other domains [35]. This study therefore contributes a dual-stream multitask architecture in which demand forecasting and anomaly-signal classification share a fused representation and are trained jointly with an imbalance-aware loss, complemented by cost-sensitive threshold selection and post hoc probability calibration. Second, the studies that report strong detection performance are supervised by independently recorded incidents [16], whereas the supervision available here is weak and is derived from the very series being modelled (Section 3.1). The anomaly head of the present model is therefore evaluated and presented as a calibrated screening signal, and no claim is made that it detects validated real-world disruptions.

Table 4: Integrated forecasting and anomaly detection frameworks

Study	Coupling of the two tasks	Anomaly supervision	Application domain
Hundman et al. [16]	Forecast-then-threshold: LSTM forecaster plus nonparametric dynamic thresholding of residuals	Independently recorded incident labels	Spacecraft telemetry
Malhotra et al. [28]	Reconstruction-then-threshold: LSTM encoder–decoder reconstruction error	Labelled anomalous sequences	Multi-sensor time series
Nguyen et al. [30]	Two separate models: LSTM forecaster and LSTM-autoencoder with one-class SVM	Anomalies observed in operational sales data	Supply chain (fashion retail)
Seman et al. [35]	Joint multitask network obtained by differentiable architecture search	Operational condition-monitoring data	Predictive maintenance
Liu et al. [24]	Integrative framework coupling forecasting, anomaly detection and resource allocation	Not specified for a single benchmark	Supply chain resilience
This study	Joint multitask network with a shared fused representation and an imbalance-aware loss	Weak labels derived from the residuals of the same sales series	Supply chain (multi-store retail demand)

Because several of the ingredients used here are established, it is important to state precisely which parts of the architecture are new. Table 5 separates the components that are adopted unchanged from prior work, those that are adapted, and the element that this study claims as its own contribution.

No individual building block is claimed to be novel. What the study contributes is the combination and, equally, an explicit account of what that combination can and cannot support. Three advantages over conventional arrangements follow from the results reported later. Relative to deploying two separate models [30], a single network is trained once and the auxiliary stream improves the forecasting task itself by 7.23% in RMSE (Table 7), which a separate detector cannot do. Relative to forecast-then-threshold designs [16, 28], the alerting decision is driven by a calibrated probability, with an expected calibration error of at most 0.0044 (Table 15), so it can be fed directly into an expected-cost computation instead of requiring a separate residual-thresholding rule. Relative to conventional

Table 5: Component-level provenance of the proposed architecture

Component	Status	Source or basis	Role in this study
Stacked LSTM encoder for the sales stream	Adopted unchanged	Hochreiter and Schmidhuber [15]; standard demand-forecasting practice [11, 31]	Encodes trend, seasonality and autocorrelation of the primary stream
Transformer encoder for the auxiliary stream	Adopted unchanged	Vaswani et al. [40]	Encodes variable lags and time-varying importance of the anomaly-signal features
Late fusion by concatenation with a shared head	Adopted unchanged	Standard multitask practice [4]	Produces one representation that feeds both output heads
Two-head objective combining MSE and weighted BCE	Adapted	Multitask learning [4]; class-imbalance weighting	Fixed weights of 0.5 and 2.0, positive-class weight and logit-based bias initialisation for a positive rate near 0.10
Statistically derived auxiliary stream	Adapted	STL decomposition [8]; change-point detection [39]	Supplies an auxiliary stream when no exogenous data are available, at the cost of self-referential supervision
Cost-aligned selection of the alerting threshold	Adapted	Cost-sensitive learning [34]	Aligns θ with the assumed FN:FP cost structure (Tables 12 and 13)
Temperature scaling of the anomaly head	Adopted unchanged	Guo et al. [14]	Makes alert probabilities usable directly in expected-cost computations
The specific combination of the above in one demand-forecasting model	Claimed as new	This study (Section 2.4)	Joint forecasting and weak-label screening with cost-aligned thresholds and calibrated probabilities in a single SCM network

multitask frameworks, the addition is the cost-aligned decision layer of Tables 12 and 13. It should be stated with equal clarity that, under the evidence presented here, the advantage of the combination is demonstrated for the forecasting task only; the classification head does not outperform its counterparts in ranking terms, for the reasons analysed in Section 5.

3 Methodology

This study proposes a single, multitask framework that jointly learns standard demand forecasting (regression) and anomaly detection (binary classification). The inputs comprise the historical sales time series (primary stream) and the time series of auxiliary/anomaly signals (auxiliary stream); both are derived from the same source, as explained in Section 1. The method learns branch-specific representations and then fuses them to jointly produce the demand two days ahead of the input window and the probability of an anomaly on that day. The loss is a weighted sum of regression MSE and classification WBCE, with positive-class weighting to address class imbalance. In addition, the classification threshold is tuned on the validation set to maximize the F1-score, thereby reducing sensitivity to the operating threshold during deployment.

3.1 Problem definition

The analysis unit is the store–item level; all data construction, windowing, standardization, and evaluation are performed per group. From an input window of length L, $X = \{x_{t-L+1}, \ldots, x_t\}$, the model jointly estimates the next-step demand y_{t+1} and the anomaly label $z_{t+1} \in \{0, 1\}$, where $z_{t+1} = 1$ denotes that the series exhibits a statistically defined anomalous pattern, formalised below.

After aligning each group's time series, this study constructs sliding windows with $L = 30$. The regression target is the z-scored demand $\widehat{y}_{t+1}$, standardized by the training-window mean and standard deviation (replacing a zero standard deviation with 1). Each window spans the 30 days ending two days before the target day, so the day immediately preceding the target is not part of the input and

the task is a two-day-ahead forecast; this corresponds to a one-day reporting lag in the sales feed. To prevent temporal leakage, the last 90 days of each group are held out as an independent validation set.

To respect the constraints of a public dataset, no external data are collected; instead, this study derives signals statistically from the sales series. Specifically, this study computes an anomaly score from the residuals of STL decomposition and create a threshold-based weak label (anomaly_label) [8]. The decomposition uses a weekly period (period = 7) with robust fitting. The anomaly score is the absolute robust z-score of the remainder, s = |r − median(r)| / (1.4826 · MAD(r)), set to zero where the MAD vanishes, and the label is anomaly_label = 1 when s $\geq \tau$ with $\tau = 3.5$ and 0 otherwise. The threshold was fixed a priori and applied unchanged to every store–item series; it yields a positive rate of 0.0999 on the validation split (Table 14), so that on average one day in ten is labelled anomalous. Because the label is a deterministic function of the same series that the model observes, it constitutes weak, self-referential supervision; the classification results are interpreted accordingly throughout this paper. This study also performs change-point detection to obtain change-point flags, using binary segmentation with an L2 cost function and three breakpoints per series, with the day before and the day of each detected breakpoint flagged [39]; includes calendar features (dow, month, is_weekend, is_month_start, is_month_end); and computes rolling_vol_z as the z-score of the rolling seven-day standard deviation of sales (minimum two observations), with non-finite values set to zero. All signals are aligned to the sales series by the (store–item–date) key. The auxiliary signals are computed once per series over the whole span covered by the public dataset, which extends 90 days beyond the last day of the training period. The STL decomposition, the change-point search and the normalisation of the volatility index therefore draw on observations that follow the day being described, so the auxiliary stream is not strictly causal. The consequence for the labels is not negligible: recomputing them from the training portion alone changes 3.5% of the validation labels and reduces the number of positives from 4,498 to 3,226. Section 5 returns to this point.

For clarity, Table 6 fixes the terminology used for all anomaly-related quantities. These terms are used in this fixed sense in the remainder of the paper, and the earlier expression “anomaly indicator” is no longer used.

Table 6: Terminology for anomaly-related quantities

Term	**Definition**
Anomaly score (s)	Continuous anomaly intensity: the absolute robust z-score of the remainder of the weekly STL decomposition of the sales series, s = |r − median(r)| / (1.4826 · MAD(r)) [8].
Anomaly label (z)	Weak binary training target: z = 1 when s $\geq \tau$, with $\tau = 3.5$. It is self-referential because it is computed from the sales series itself.
Anomaly-signal features	The auxiliary input vector at each time step: anomaly score, change-point flag [39], calendar variables (dow, month, is_weekend, is_month_start, is_month_end) and rolling_vol_z.
Anomaly probability (p)	The sigmoid output of the classification head: the model-estimated probability that the next-step label equals 1.
Alert indicator	The binary decision issued when p $\geq \theta$, where θ is selected on the validation set, either F1-optimal (Table 11) or cost-optimal (Table 12).

The model outputs $\widehat{y}_{t+1}$ (regression) and p_{t+1} (anomaly probability). The objective is $J = 0.5 \cdot MSE(y, \widehat{y}) + 2 \cdot WBCE(z, p)$, where WBCE applies a positive-class weight $w_+ = \frac{1-\pi}{\max(\pi, 10^{-6})}$ based on the positive rate π. The classification bias is initialized to $logit(\pi)$ to account for class sparsity. This study uses Adam (lr = 10^{-3}), batch size = 512, max epoch = 100, and early stopping (patience = 5). The classification threshold is selected on the validation set by grid search $\theta \in [0.05, 0.95]$ with step 0.05 to maximize F1-score. That grid holds the 19 values from 0.05 to 0.95, whereas the cost-sensitive search of Section 3.4 uses 99 values from 0.01 to 0.99 in steps of 0.01, which is why the cost-optimal thresholds reported in Table 12 (0.13, 0.15 and 0.17) do not lie on the coarser grid. Figure 1 summarizes the setup.

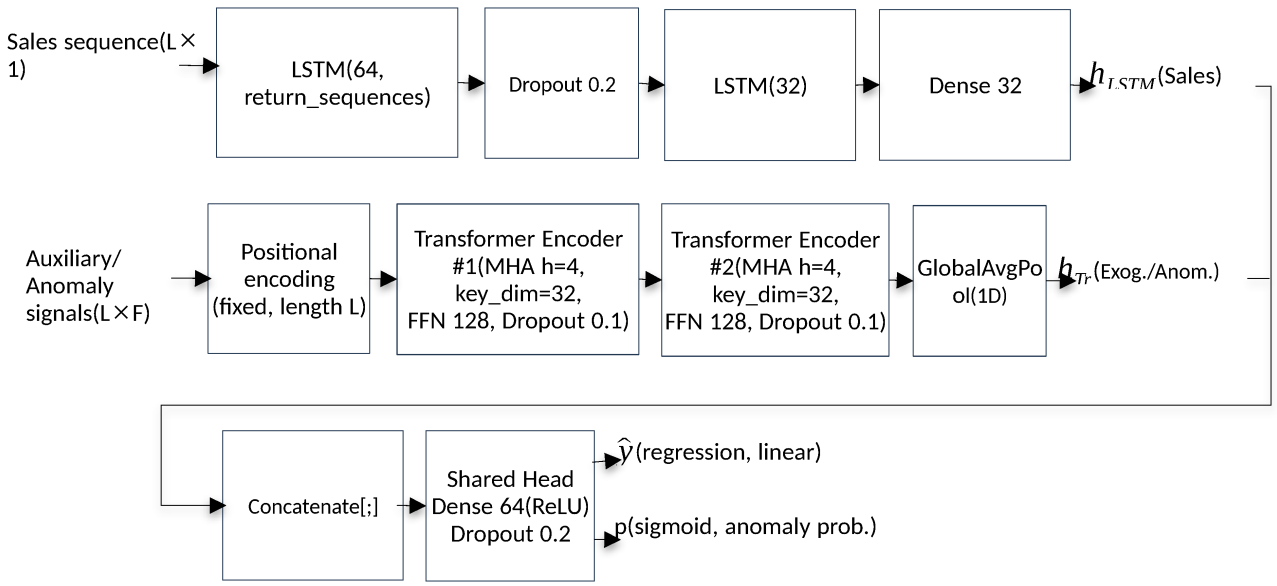


Fig. 1: Proposed_MM architecture: a dual-input model that fuses an LSTM branch for sales with a transformer-encoder branch for auxiliary/anomaly signals to jointly predict demand ($\widehat{y}$) and anomaly probability (p); training minimizes 0.5MSE + 2WBCE with class-imbalance weighting and F1-score based threshold tuning.

3.2 Dual-stream multitask architecture

The model is a dual-input multitask architecture that processes two synchronized time-series inputs in parallel and then fuses them so that a single head performs both regression and classification. The two streams are aligned to the same length L: the sales series (primary stream) and the auxiliary/anomaly signals (auxiliary stream). As defined in §3.1, the auxiliary stream includes the STL-residual–based anomaly score/label, change-point flags, calendar-derived features, and volatility features, all aligned by (store–item–date) [8, 39]. This subsection details the branch-specific encoders, fusion, and the baseline/ablation/proposed variants.

To robustly capture short- and long-term dependencies, the sales sequence ($L \times 1$) is embedded with stacked LSTMs [15]: LSTM(64, return_sequences=True) → Dropout(0.2) → LSTM(32). A subsequent Dense(32, ReLU) projects the final hidden state to the sales latent vector h_{LSTM}. This branch is designed to preserve trend/seasonality/autocorrelation structures in a compact representation [22].

This study implements two alternatives: Specifically, in Proposed_MM, a Transformer encoder [40] models variable lags and time-varying importance. A positional encoding with the same shape as the input feature matrix ($L \times F$), drawn once when the network is constructed and held fixed thereafter, is added to it, followed by two encoder blocks. Each block applies layer normalization, multi-head self-attention (num_heads=4, key_dim=32), a residual connection, and a position-wise FFN (Dense(128, ReLU) → Dropout(0.1) → Dense) with another residual connection. GlobalAverage-Pooling1D summarizes the sequence to produce h_{Tr}.
In Ablation_Exog variant, this study replaces the Transformer with causal Conv1D layers (filters 64 and 32, kernel size 3, ReLU) followed by GlobalAveragePooling1D to extract local temporal patterns.

The latent vectors h_{LSTM} and h_{Tr} are concatenated and passed through a shared head (Dense(64, ReLU) → Dropout(0.2)). The head bifurcates into: (i) a linear Dense(1) regression head outputting $\widehat{y}_{t+1}$; and (ii) a sigmoid Dense(1) classification head outputting p_{t+1}. Bias initialization and class-imbalance handling follow §3.1. The decision threshold is selected on the validation set to maximize the F1-score.

Baseline_LSTM connects only the sales branch to the head. Ablation_Exog fuses the LSTM sales branch with the Conv1D auxiliary branch. Proposed_MM uses the Transformer encoder in the auxiliary branch to learn long-range interactions and variable lags.

Temporal dependencies in sales are learned by the LSTM branch, while complex temporal patterns in auxiliary/anomaly signals are encoded by the Transformer (or Conv1D in the ablation). The two representations are integrated via late fusion, and the shared head jointly outputs the regression and

classification targets. To address label noise and class imbalance, this study implements alternative classification losses as configurable options (LOSS_TYPE); unless otherwise stated, weighted BCE is used.

3.3 Evaluation metrics

This study assesses demand forecasting accuracy and anomaly-discrimination ability over the validation set using RMSE (root mean squared error) and MAE (mean absolute error).

For anomaly detection, the predicted anomaly probability is thresholded to produce a binary alert. This study reports precision, recall, and F1-score. Precision measures the fraction of predicted anomalies that are true anomalies (low false positives), recall measures the fraction of true anomalies detected (low false negatives), and F1 is their harmonic mean. Threshold-independent metrics, such as ROC-AUC and PR-AUC, are also reported.

Because missed anomalies often incur higher costs in SCM, recall is important; however, excessive alerts must be avoided. This study, therefore, selects θ on the validation set to maximize F1-score as a balanced compromise. A grid search over $\theta \in [0.05, 0.95]$ (step 0.05) is used. Alongside the resulting confusion matrix (TP, FP, TN, FN), this study reports the actual positive rate π and the predicted positive rate (alert rate) at the chosen θ to analyze alert–performance trade-offs. This study considers asymmetric error costs; when the cost of FN exceeds that of FP, it defines a cost matrix and selects θ minimizing the expected cost on the validation set.

In addition, threshold independent metrics, namely ROC-AUC and PR-AUC, are reported. ROC-AUC summarizes sensitivity vs. (1–specificity), while PR-AUC is more informative under class imbalance. Values closer to 1 indicate better discrimination. These metrics complement the F1-score at a specific θ.

During training, this study applies the positive-class weight, $w_+ = \frac{1-\pi}{\pi}$, in the classification loss to correct the imbalance. At evaluation, all metrics follow their standard (unweighted) definitions, and the test-time decision rule uses the θ selected on the validation set for consistency.

3.4 Extended evaluation and training strategies

To assess the statistical significance of performance differences between models, this study employs additional analyses. For RMSE/MAE, this study uses a nonparametric bootstrap with 5,000 resamples of error differences to estimate the sampling distribution of mean gaps; the 2.5th and 97.5th percentiles form 95% confidence intervals [12]. Differences in ROC-AUC are evaluated using the DeLong test for correlated ROC curves, yielding standard errors and p-values to judge significance [10].

This study also aligns threshold selection with operational cost structures. While the default θ maximizes F1, this additionally defines a cost matrix with the cost of FN set to five times that of FP and search for θ that minimizes expected cost on the validation set [34]. The 5:1 ratio is adopted as an illustrative base case, reflecting that in many supply chain settings the cost of missing a disruption (stockouts, expediting, service-level penalties) exceeds the cost of investigating a false alert. Because the appropriate ratio is industry- and firm-specific, Section 4 additionally reports how the cost-optimal policy changes under ratios of 2:1, 5:1 and 10:1 (Table 13). This study then compares whether the cost-optimal θ yields more operationally favorable outcomes than the F1-optimal θ.

To mitigate data imbalance and label noise at the loss level, this study experiments with three alternatives in addition to weighted BCE: Focal loss $(1 - p_t)^\gamma$ down-weights easy examples [23]; Label smoothing replaces hard targets with softened labels (e.g., $1 - \varepsilon$ and ε) to improve calibration [36]; and Symmetric cross-entropy adds a reverse-CE term to the standard CE to improve robustness under noisy labels [41]. The active loss is selected via LOSS_TYPE.

To improve probability calibration, this study applies temperature scaling: a scalar temperature T is fitted on the validation set to rescale logits z/T before the sigmoid, aligning predicted probabilities with observed frequencies [14]. Reliability diagrams visualize pre/post-calibration alignment.

Finally, this study conducts a stratified error analysis by demand volatility. Using rolling_vol_z as a continuous volatility indicator, this study partitions validation instances into quantile bins and report RMSE, MAE, and F1-score per bin to examine how forecasting and detection performance vary

with volatility. Finally, because a single fixed-origin split can understate evaluation variability in time series [3, 38], Section 4 quantifies how sensitive the pooled error metrics are to the choice of validation origin by re-evaluating the external reference baselines over three consecutive, non-overlapping 90-day windows.

4 Experiments and results

This study uses the public Store Item Demand Forecasting Challenge dataset from Kaggle [18]. The dataset contains daily sales from 2013 to 2017 for 10 stores and 50 items within a single retail chain. For each store–item series, the most recent 90 days are held out as the validation set to prevent data leakage during training. As described in Section 3, auxiliary variables and anomaly signals are generated directly from the sales series [39]: robust z-scores of STL residuals (anomaly score), change-point flags via the ruptures library, calendar features (day-of-week, month, weekend, month-start, month-end), and a short-horizon volatility indicator computed from the rolling 7-day standard deviation. These auxiliary variables are aligned with sales by the (store–item–date) key and used as model inputs.

The three configurations are implemented in TensorFlow/Keras and were executed on the hosted Google Colab runtime with the platform's default library versions; because that runtime is allocated dynamically and its hardware was not recorded, wall-clock timings are not reported. The networks are small: Baseline_LSTM has 30,434 trainable parameters, Ablation_Exog 41,378 and Proposed_MM 46,530. Training uses Adam with a learning rate of 0.001, a batch size of 512, a cap of 100 epochs and early stopping on the validation loss with a patience of 5 and restoration of the best weights; the random seed is fixed at 42 for TensorFlow, NumPy and the Python standard library, and the same seed is used for the bootstrap resampling of Section 3.4. Sliding windows yield 852,000 training and 45,000 validation instances. The auxiliary input of each window is a 30×8 matrix holding the five calendar variables, the anomaly score, the change-point flag and the rolling-volatility index; the anomaly label is used only as the classification target and is never supplied as an input. One qualification belongs here rather than in the discussion: early stopping selects the epoch on the same validation split that produces every reported metric, so the validation figures are selected on that split rather than estimated on data unseen by the selection procedure. The external reference baselines of Table 8 were computed on a single CPU core with statsforecast 2.1.1 (AutoARIMA) and XGBoost 3.4.0.

Table 7: Metrics summary (validation)

Model	RMSE	MAE	Precision	Recall	F1	ROC-AUC	PR-AUC
Baseline_LSTM	0.6250	0.4844	0.1249	0.5207	0.2014	0.5776	0.1264
Ablation_Exog	0.6109	0.4783	0.1313	0.4627	0.2046	0.5835	0.1330
Proposed_MM	0.5798	0.4511	0.1245	0.5602	0.2037	0.5892	0.1409

To place these results against references that are independent of the proposed architecture, Table 8 reports four external baselines evaluated under exactly the protocol of Section 3.1: per-group standardisation with training-split statistics, the same final 90 days of each series held out, the same two-day-ahead information set, in which the most recent observation available for a target day is the value two days earlier, and metrics pooled over all 45,000 validation points. The baselines are a naive forecast that repeats the last available observation, a seasonal naive model with a weekly period, AutoARIMA with weekly seasonality (orders selected on the training split using the approximation-based search that the reference implementation applies by default to series longer than 150 observations [17], then held fixed and applied at a two-day horizon), and a global gradient-boosted tree model (XGBoost [6]) trained across all 500 store–item groups on the same 30-day window together with the calendar and rolling-volatility features of Section 3.1, with the 90 days preceding the validation window reserved for early stopping. Proposed_MM improves on all three classical baselines, reducing RMSE by 16.9% relative to AutoARIMA, but the gradient-boosted tree baseline attains a slightly lower point error (RMSE 0.5574 against 0.5798, a difference of 3.9%). This comparison narrows what can be claimed: the contribution of the proposed architecture is not universal superiority in point accuracy, but the joint provision within a single network of competitive forecasts and calibrated, cost-aware alert

probabilities, which a gradient-boosting pipeline would deliver only by adding a separate detection model and a separate calibration stage.

Table 8: External reference baselines under the evaluation protocol of Section 3.1 (standardised scale, 45,000 pooled validation points)

Model	RMSE	MAE
Baseline_LSTM (this study, Table 7)	0.6250	0.4844
Ablation_Exog (this study, Table 7)	0.6109	0.4783
Proposed_MM (this study, Table 7)	0.5798	0.4511
Naive (last available observation)	1.0266	0.8237
Seasonal naive (7-day)	0.8367	0.6500
AutoARIMA (weekly seasonality)	0.6978	0.5480
XGBoost (global; 30 lags, calendar and volatility features)	0.5574	0.4352
DLinear (global; decomposition-linear, 30-day look-back)	0.6658	0.5218

Because recent work on time-series forecasting is dominated by attention-based and decomposition-based architectures, Table 8 also reports DLinear [44], a decomposition-linear model that was shown to match or exceed Transformer-based long-horizon forecasters on standard benchmarks. Fitted globally under the protocol of Section 3.1, it reaches an RMSE of 0.6658, which improves on AutoARIMA but remains worse than all three configurations of this study. This is consistent with its design purpose: DLinear targets long-horizon forecasting from long look-back windows, whereas the task here is a two-day-ahead prediction from a 30-day window, a regime in which recent lags carry most of the signal and a gradient-boosted tree model on the same lags is the stronger competitor. A full benchmark against architectures such as the Temporal Fusion Transformer [21] and PatchTST [32] would require a separate tuning and training budget for each model and is left for future work; the comparative claims of this paper are therefore bounded to the configurations listed in Table 8, and no claim of state-of-the-art point accuracy is made anywhere in the paper.

Because all of these results rest on a single validation origin, Table 9 re-evaluates the external baselines over three consecutive, non-overlapping 90-day windows, recomputing the standardisation statistics and re-fitting each model on the data preceding each window. Two findings matter for the interpretation of this study. First, the ordering of the baselines is identical in all three windows, so comparative conclusions drawn from a single origin are stable for these models. Second, the final window used throughout the paper is the easiest of the three for every baseline: pooled RMSE on that window is between 5.8% and 17.4% lower than the average over the two earlier windows, and the coefficient of variation of RMSE across origins ranges from 3.5% for the seasonal naive model to 10.7% for the naive forecast. The absolute error levels reported in Table 7 should therefore be read as specific to this validation window rather than as origin-independent quantities, even though the relative comparison between models is the quantity the study actually claims. The three deep configurations could not be re-trained under this protocol within the revision, so Table 9 characterises the evaluation window rather than the architecture; this restriction is stated as a limitation in Section 5.

Table 9: Origin sensitivity of the external baselines: pooled RMSE over three consecutive 90-day validation windows

Model	Fold 1 (2017-04-06 – 2017-07-04)	Fold 2 (2017-07-05 – 2017-10-02)	Fold 3 (2017-10-03 – 2017-12-31; main split)	Mean ± SD
Naive (last available observation)	1.2584	1.2257	1.0266	1.1702 ± 0.1254
Seasonal naive (7-day)	0.8943	0.8827	0.8367	0.8713 ± 0.0305
XGBoost (global)	0.6429	0.6320	0.5574	0.6108 ± 0.0465
DLinear (global)	0.7231	0.6935	0.6658	0.6941 ± 0.0287

To assess the statistical significance of the regression improvements, this study computes bootstrap confidence intervals and p-values. Table 10 reports 95% confidence intervals and p-values for

RMSE/MAE differences relative to Baseline_LSTM. The 95% confidence interval for the RMSE gap between Baseline_LSTM and Proposed_MM is [0.0419, 0.0486], excluding zero ($p < 0.001$), which indicates statistical significance. For ROC–AUC, the archived output of the DeLong procedure [10] recorded an AUC difference of exactly 0.000 together with $p = 1.000$ for all three model pairs. That output cannot be reconciled with the AUC estimates of Table 7, whose pairwise differences are 0.0059, 0.0117 and 0.0057. Inspection of the routine that produced it identified the cause: the area was accumulated from midranks computed within the positive class alone rather than from midranks over the pooled sample, a quantity that equals zero for any input, so the procedure returned a null difference and a unit p-value by construction rather than as a finding about the models. The archived values are retained in Table 10 for transparency but are not used: the AUC differences are reported descriptively, and no significance claim is made for them. This does not affect any conclusion of the study, because all three AUC values lie between 0.578 and 0.589 and no claim of superior anomaly discrimination is made in any case.

Table 10: Significance analysis (bootstrap confidence intervals; the archived DeLong output is reported for transparency only)

Model Pair	RMSE confidence interval	RMSE p-value	MAE confidence interval	MAE p-value	AUC difference	DeLong p-value
Baseline_LSTM vs Ablation_Exog	0.0112 - 0.0171	<0.001	0.0039 – 0.0081	<0.001	0.0059	1.0
Baseline_LSTM vs Proposed_MM	0.0419 - 0.0486	<0.001	0.0310 – 0.0356	<0.001	0.0117	1.0
Ablation_Exog vs Proposed_MM	0.0293 - 0.0329	<0.001	0.0256 – 0.0288	<0.001	0.0057	1.0

From a practical standpoint the forecasting improvement is the primary contribution of the architecture. The 7.23% RMSE and 6.86% MAE reductions relative to Baseline_LSTM are obtained without any external data, purely by routing statistically derived auxiliary features through a dedicated encoder, and the ablation isolates where the gain originates: the same features processed by a causal convolutional branch recover a 2.27% RMSE reduction, so approximately two thirds of the total improvement is attributable to the attention-based encoding of the auxiliary stream rather than to the presence of the features alone. In supply chain operations a lower short-horizon error at the store–item level translates directly into tighter safety-stock and replenishment decisions, and the multitask design delivers this together with calibrated alert probabilities without maintaining a second model.

The classification threshold (θ) plays a central role in the alert policy. A validation-set grid search maximizing the F1-score yielded the same value for all three models, $\theta = 0.10$. Table 11 reports, for each model, the threshold at which the F1-score is maximized, along with the corresponding Precision, Recall, and F1-score. In supply chain operations, where must-catch events are prioritized, it is generally reasonable to select a decision threshold below 0.5 to increase recall, as confirmed by the empirical results of this study.

Table 11: Threshold Tuning (F1-score Optimal)

Model	F1-score best threshold	Best precision	Best recall	Best F1-score
Baseline_LSTM	0.10	0.1249	0.5207	0.2014
Ablation_Exog	0.10	0.1313	0.4627	0.2046
Proposed_MM	0.10	0.1245	0.5602	0.2037

This study further incorporates operational cost asymmetry by setting the cost of a false negative (FN) to five times that of a false positive (FP) and re-optimizing θ. Table 12 reports the cost-sensitive thresholds and expected costs together with two trivial reference policies that any useful alerting rule must beat: never alert, whose expected cost is $5 \times 4{,}498 = 22{,}490$, and always alert, whose expected cost is 40,502. Two observations follow. First, the cost-optimal threshold for Baseline_LSTM ($\theta =$

0.15) produces no predicted positives at all (FP = 0 and FN = 4,498), so the cost-optimal policy for the baseline degenerates into the never-alert rule and its expected cost of 22,490 is exactly the cost of issuing no alerts; under this cost structure the baseline probabilities are not informative enough for any operating point to improve on doing nothing. Second, Proposed_MM attains the lowest expected cost, 21,860 at $\theta = 0.17$ with 543 true positives, which is 2.8% below the never-alert reference, while Ablation_Exog reaches 22,306, or 0.8% below it. The absolute margin over the trivial policy is therefore modest and should be read as screening-level value consistent with the limited discrimination reported above, rather than as evidence of a strong stand-alone detector. Nevertheless, only the dual-stream configurations deliver any positive value over the trivial policy, which supports both the contribution of the auxiliary stream and the importance of cost-aligned threshold selection for supply-chain operations.

Table 12: Cost-sensitive Thresholds (C_FN = 5·C_FP)

Model	Best_cost_ threshold	Expected_cost	False_negatives	False_positives
Baseline_LSTM	0.15	22,490	4,498	0
Ablation_Exog	0.13	22,306	4,276	926
Proposed_MM	0.17	21,860	3,955	2,085
Never alert (reference policy)	—	22,490	4,498	0
Always alert (reference policy)	—	40,502	0	40,502

Because the appropriate cost ratio is industry- and firm-specific, Table 13 recomputes the expected cost of every operating point reported in this study under FN:FP ratios of 2:1, 5:1 and 10:1, alongside the two trivial reference policies. The exercise changes the conclusion qualitatively rather than merely quantitatively. At 2:1 no evaluated operating point improves on never alerting (8,996), so under a mildly asymmetric cost structure the alerting layer carries no value. At 5:1 the cost-optimal operating point of Proposed_MM is the best of all evaluated policies. At 10:1 the ranking shifts again: for every model the F1-optimal threshold $\theta = 0.10$ becomes preferable to the threshold that was cost-optimal at 5:1, and Proposed_MM at $\theta = 0.10$ attains the lowest cost of all evaluated policies (37,500), 16.6% below never alerting and 7.4% below always alerting. Reporting the full grid therefore makes explicit that the operating point, and indeed whether alerting is worthwhile at all, depends on the assumed cost asymmetry. Because only the two thresholds reported in this study were available for each model, Table 13 evaluates those operating points rather than re-optimising θ over the full grid for each ratio; a complete re-optimisation requires the stored validation probability vectors for every model and is left to future work.

Table 13: Expected cost of every operating point reported in this study under FN:FP cost ratios of 2:1, 5:1 and 10:1 (lowest cost per ratio: 8,996, 21,860 and 37,500)

Policy	θ	FN	FP	Cost at 2:1	Cost at 5:1	Cost at 10:1
Never alert (reference)	—	4,498	0	8,996	22,490	44,980
Always alert (reference)	—	0	40,502	40,502	40,502	40,502
Baseline_LSTM, F1-optimal	0.10	2,156	16,415	20,727	27,195	37,975
Ablation_Exog, F1-optimal	0.10	2,417	13,767	18,601	25,852	37,937
Proposed_MM, F1-optimal	0.10	1,978	17,720	21,676	27,610	37,500
Baseline_LSTM, cost-optimal at 5:1	0.15	4,498	0	8,996	22,490	44,980
Ablation_Exog, cost-optimal at 5:1	0.13	4,276	926	9,478	22,306	43,686
Proposed_MM, cost-optimal at 5:1	0.17	3,955	2,085	9,995	21,860	41,635

Analysis of confusion matrices (Figs. 2–4) shows that all models generate predicted positive rates far exceeding the actual positive rate during validation. While the actual positive rate is about 10%, Baseline_LSTM predicts positives 41.7% of the time, with 24,087 TN, 16,415 FP, 2,342 TP, and 2,156 FN. The low threshold reduces missed anomalies but triggers many alerts under normal conditions.

Ablation_Exog, which adds a Conv1D-based auxiliary branch to the sales pathway, reduces FPs

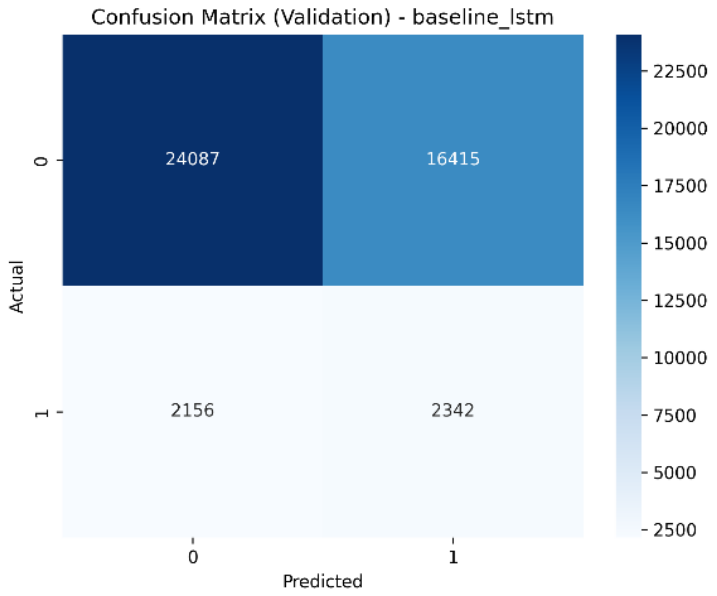


Fig. 2: Confusion matrix – Baseline_LSTM

by 2,648 (16,415 → 13,767) compared with Baseline_LSTM. Its predicted positive rate declines to 35.2% and Precision improves slightly, but FNs increase to 2,417 (up by 261), indicating that patterns extracted from auxiliary signals help suppress unnecessary alerts but also miss more true anomalies.

By contrast, Proposed_MM introduces a Transformer encoder in the auxiliary branch and achieves the highest Recall. With 2,520 TP and 1,978 FN, it reduces FNs by 178 versus Baseline_LSTM. Although FPs (17,720) exceed those of Ablation_Exog, they are only ~1,305 higher than Baseline_LSTM. Its predicted positive rate is 45.0%, providing more alerts while reducing missed anomalies. This operating-point difference should not be over-interpreted: it is obtained at a common fixed threshold ($\theta = 0.10$), the 178 avoided false negatives are accompanied by 1,305 additional false positives, and the three models differ by at most 0.012 in ROC–AUC (Table 10). Under cost structures in which missed anomalies are substantially more expensive than false alerts this recall-oriented profile is nevertheless operationally relevant, as quantified by the cost-sensitive analysis below.

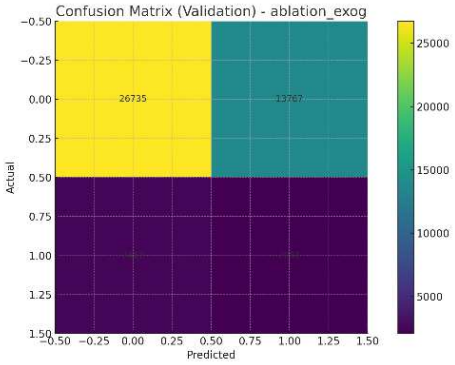


Fig. 3: Confusion matrix – Ablation_Exog

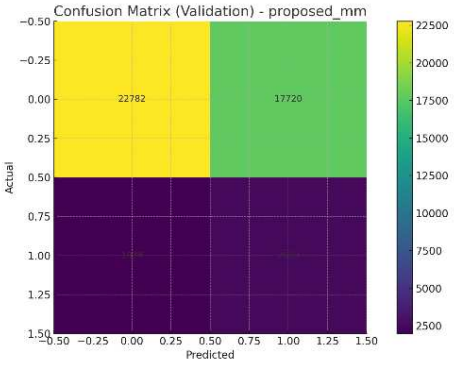


Fig. 4: Confusion matrix – Proposed_MM

The confusion-matrix analysis makes the precision–recall trade-off explicit. Baseline_LSTM triggers many false positives; Proposed_MM yields more alerts (FPs) than Baseline_LSTM, while Ablation_Exog reduces FP at the expense of higher FN. Because this comes with more alerts, practical deployments should manage alert load via cost-sensitive thresholding or downstream filtering. Table 14 consolidates the validation-set actual positive rate (π), the predicted positive rate, and confusion-matrix counts and metrics at $\theta = 0.10$. All models predicted positive rates that greatly exceed $\pi \approx 0.10$ (0.4168/0.3522/0.4498 for Baseline/Ablation/Proposed), reflecting the typical pattern under class imbalance whereby higher recall at low thresholds trades off against precision. At the same θ, Proposed_MM achieves the highest recall (0.5602), increasing predicted positives by 1,483 vs

Baseline_LSTM (20,240 vs 18,757) and by 4,392 vs Ablation_Exog (20,240 vs 15,848), while decreasing FNs by 178 (−8.3%) and 439 (−18.2%), respectively. Ablation_Exog attains the highest Precision (0.1313) but lower Recall (0.4627). Baseline_LSTM remains in between.

Table 14: Diagnostics at θ=0.10

Model	val_ positive_ rate	pred_ positive_ rate@0.10	TP	FP	TN	FN	Precision @0.10	Recall @0.10	F1@0.10
Baseline_LSTM	0.099956	0.4168	2,342	16,415	24,087	2,156	0.1249	0.5207	0.2014
Ablation_Exog	0.099956	0.3522	2,081	13,767	26,735	2,417	0.1313	0.4627	0.2046
Proposed_MM	0.099956	0.4498	2,520	17,720	22,782	1,978	0.1245	0.5602	0.2037

Probability calibration results are shown in Figs. 5–7 and Table 15. The reliability diagrams illustrate the alignment between predicted probabilities and empirical frequencies on the validation set. The expected calibration error (ECE) in Table 15 aggregates bin-wise accuracy–probability gaps; lower values indicate better alignment. All three models concentrate probabilities in the 0.00–0.30 range, with few samples at high probabilities. Under this distribution, temperature scaling yields very small ECE values on the order of 10^{-3}–10^{-2} (Baseline_LSTM ECE = 0.0018, T = 1.0329; Ablation_Exog ECE = 0.0032, T = 1.0426; Proposed_MM ECE = 0.0044, T = 1.0019). Thus, predicted probabilities align well with observed frequencies; notably, Proposed_MM has T ≈ 1, suggesting minimal distortion of the probability scale.

Across the three models the deviations from the diagonal are small and confined to the low-probability region, with mild over-confidence for Proposed_MM around 0.20–0.30. Given temperatures close to 1 and an ECE of at most 0.0044, these deviations are operationally acceptable, and because calibration uses a single scalar temperature it is inexpensive to repeat under distribution shift. The calibrated probabilities can therefore be used directly for the expected-cost computations and threshold searches reported above.

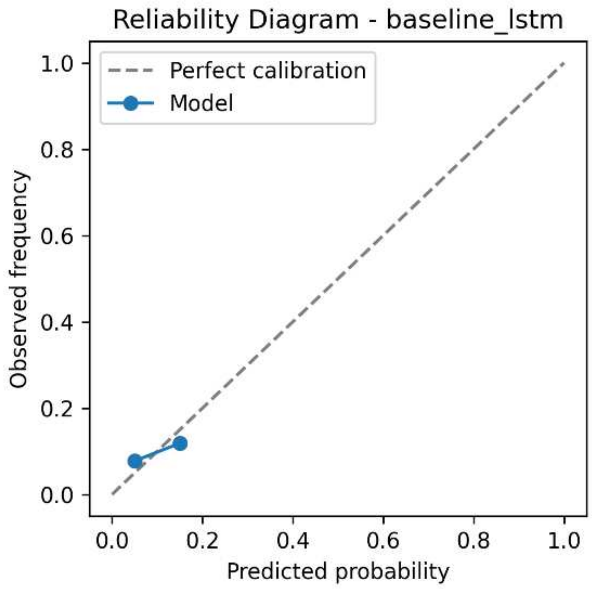


Fig. 5: Reliability diagrams of Baseline_LSTM

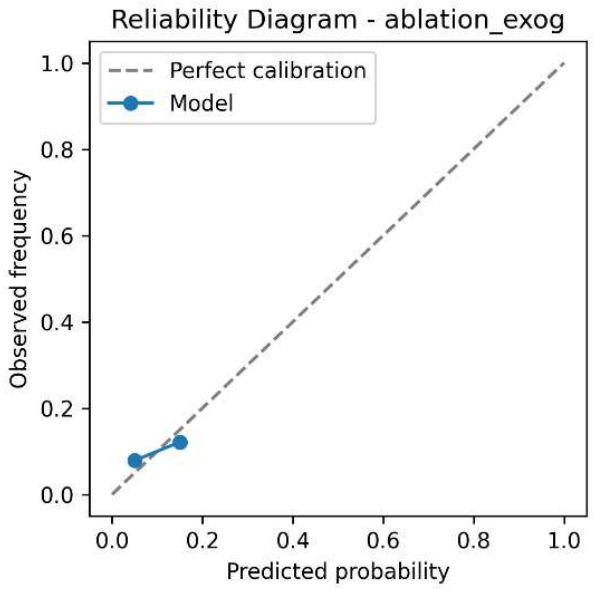


Fig. 6: Reliability diagrams of Ablation_Exog

Proposed_MM significantly reduces regression error and outperforms Baseline_LSTM and Ablation_Exog under cost-sensitive detection. Bootstrap analysis confirms that RMSE/MAE improvements are statistically significant, while the differences in ROC–AUC are small, at most 0.012, and are reported descriptively only. Against external references the model outperforms the classical baselines but does not surpass a global gradient-boosted tree baseline in point accuracy (Table 8), which is consistent with its positioning as an integrated forecasting and screening architecture rather than as a pure point-accuracy maximiser. The F1-score optimal threshold differs from the cost-optimal threshold,

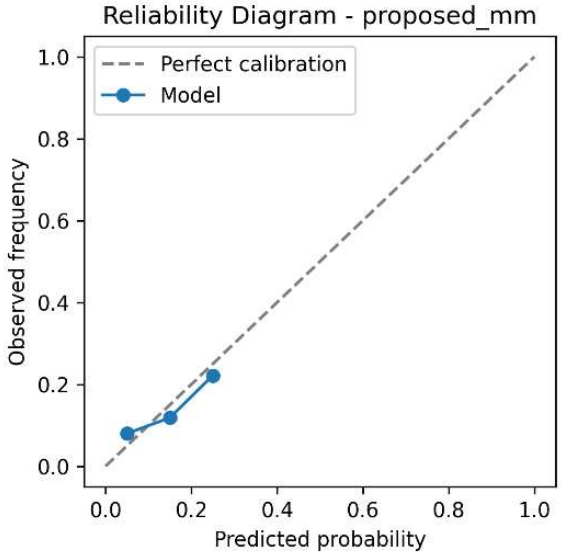


Fig. 7: Reliability diagrams of Proposed_MM

Table 15: Calibration summary (temperature scaling)

Model	Temperature	ECE
Baseline_LSTM	1.0329	0.0018
Ablation_Exog	1.0426	0.0032
Proposed_MM	1.0019	0.0044

highlighting the need for threshold adjustment tailored to operational objectives. Temperature scaling further enhances probability reliability across all models, yielding calibrated probabilities that can be directly leveraged in supply-chain decision-making

5 Discussion

This section discusses four practical implications from the empirical results.

First, regression gains are statistically reliable. RMSE and MAE reductions achieved by Proposed_MM over Baseline_LSTM exclude zero in bootstrap confidence intervals with very small p-values, confirming statistical significance (Table 10). This indicates that encoding the auxiliary/anomaly stream with a Transformer encoder and fusing it with the sales stream contributed materially to forecasting-accuracy improvements.

Second, classification uplifts are limited under imbalance and weak labels. The three models yield broadly similar F1-scores (~0.20) and ROC–AUC values that differ by at most 0.012 (Tables 7 and 10). More fundamentally, the supervision is weak and self-referential: the labels encode a transparent statistical rule computed from the same series that the model observes, so the ceiling of the classification task is to approximate that rule rather than to detect independently verified incidents. The PR–AUC values make the size of the effect explicit: against a positive rate of 0.100 they correspond to a lift of only 1.26 to 1.41 over a random ranking. The classification head should therefore be interpreted as a calibrated screening signal whose operational value is realised through cost-aligned thresholding (Tables 12 and 13) rather than through ranking metrics, and improving this component requires better labels before better losses: curating incident labels from operational logs is the first-order need, after which noise-robust loss functions and additional probability-calibration procedures can be compared meaningfully without altering the forecasting objective.

It is worth setting out explicitly why the classification performance remains modest despite the architectural changes, because the answer is structural rather than incidental. Four factors compound. First, the target is a deterministic function of the very series that the model observes, so the achievable

ceiling is the reproducibility of a threshold rule applied to STL residuals, and residual threshold crossings are close to unpredictable from earlier values of the same residual process by construction. Second, the positive rate is approximately 0.100, and against that baseline the PR–AUC values of 0.126 to 0.141 represent a lift of only 1.26 to 1.41; the ranking signal is real but weak. Third, the predicted probabilities are strongly compressed: all three models place their probability mass in the 0.00–0.30 range, and for Baseline_LSTM no validation probability reaches 0.15, which is precisely why its cost-optimal policy collapses onto the never-alert rule in Table 12. Fourth, the class imbalance forces a low operating threshold, so at $\theta = 0.10$ every model raises alerts on 35% to 45% of the validation instances (Table 14), which caps precision near the positive rate irrespective of the encoder used. Architectural refinement cannot lift a ceiling set by the label definition; changing the labels can.

Third, cost-sensitive thresholding provides operational benefits. Even for a fixed model, the F1-optimal threshold differs from the cost-optimal threshold. Under an asymmetric cost structure where FN $\gg$ FP, Proposed_MM lowers expected cost the most by reducing missed anomalies (Table 12). This empirically supports the premise that threshold selection must reflect operational loss functions; in deployment, threshold policies should be tuned to business costs rather than solely to F1.

Fourth, probability calibration is effective and lightweight. Temperature scaling yields low expected calibration error (ECE), indicating that predicted probabilities align well with observed frequencies and can serve as reliable decision inputs for early warning (Figs. 5–7; Table 15). Proposed_MM shows mild over-confidence in the lower-probability region; periodic re-validation with additional samples or cross-validation can confirm stability while preserving the observed calibration benefits.

Four limitations bound the scope of these findings. First, the anomaly component is evaluated against labels that the same series generates, so the reported detection metrics measure agreement with a statistical screening rule and support no claim about detecting real-world disruptions. Second, the headline comparison rests on a single validation origin; the origin-sensitivity analysis of Table 9 shows that this particular window is the easiest of the three tested for every external baseline, and although the ordering of models is stable across origins for those baselines, the three deep configurations were not re-trained under a rolling-origin protocol, so their stability across origins remains unverified. Third, deriving all auxiliary features from the sales series guarantees their availability in practice but caps the information the auxiliary stream can add; genuinely external signals may change the relative merits of the compared architectures. Fourth, the archived DeLong output was traced to a defect in the routine that computed it and has been withdrawn, so the comparison of AUC values is descriptive only. Fifth, two properties of the experimental pipeline qualify the validation figures. The auxiliary signals, and therefore the labels, were derived once from the entire span of the public dataset rather than from the training portion alone, so the auxiliary stream is not strictly causal and the validation labels themselves depend on that choice (Section 3.1). In addition, early stopping selected the epoch on the same validation split that produced every reported metric, which makes those metrics selected rather than held out. Neither issue affects the relative comparison between the three configurations, since all three were trained and evaluated identically, but both mean that the absolute levels in Table 7 are optimistic, and a re-run with causal feature construction and a separate model-selection split is the first correction a follow-up study should make.

A concrete route to removing the first limitation exists in operational settings. Inventory-shortage records, supplier-disruption notices, transportation-delay logs, promotion calendars and recorded demand surges are all externally observed events that can be aligned to the (store–item–date) key and substituted for the statistical rule used here. With such labels the architecture can be re-trained unchanged, and the classification head could then be assessed as a detector rather than as a screening signal. Until that assessment is carried out, the anomaly component of this study should be read strictly as a learned reproduction of a statistical screening rule, and the operational value claimed for it is limited to the cost-aligned screening reported in Tables 12 and 13.

The choice of dataset compounds this restriction. The Kaggle Store Item Demand Forecasting dataset contains sales history alone, with no promotions, prices, weather, logistics or supplier variables, which is exactly why every auxiliary feature in this study had to be derived from the sales series and why the model is single-source. The benefit of an auxiliary encoder therefore cannot be evaluated here under the conditions in which it would matter most. Public datasets that carry such variables

are available, including the M5 competition data [27] and the Favorita, Walmart and Rossmann retail datasets, and validating the architecture on them, with the auxiliary stream carrying genuinely exogenous information, is the most informative single extension of this work.

Two methodological choices were fixed a priori rather than tuned, and this bounds the strength of the conclusions. The task weights of the combined objective, 0.5 for the regression loss and 2.0 for the weighted cross-entropy, were set in advance and no sweep over alternative weightings was performed; principled alternatives exist, since homoscedastic uncertainty weighting learns the weights jointly with the network parameters [19] and GradNorm balances gradient magnitudes across tasks [7], and either could be adopted without changing the architecture. Related to this, because all three configurations share the same two-head objective, the experiments isolate the contribution of the auxiliary stream but not the contribution of multitask learning itself: no single-task variant was trained, so neither a benefit nor an interference effect between forecasting and classification can be asserted from these results. The remaining hyperparameters, namely the window length of 30, the Adam learning rate of 0.001, the batch size of 512, the early-stopping patience of 5, the layer widths of the two branches and the dropout rates, were likewise fixed rather than searched, so the reported figures should be read as the performance of one specific configuration and not as the best attainable performance of the architecture family. A single-task ablation, a task-weight sweep and a sensitivity analysis over window length and learning rate together form the natural next study.

The framework is intended to support supply chain decisions, and interpretability is therefore a legitimate requirement rather than an optional extra. Two forms of transparency are already available in the present results. At the architectural level the ablation attributes approximately two thirds of the forecasting gain to the attention-based encoding of the auxiliary stream rather than to the presence of the auxiliary features themselves, which localises where the improvement originates. At the decision level the alert probability is calibrated, so a reported value of 0.17 corresponds to an empirical frequency close to 0.17 and can be defended to an operator as such. What is missing is feature-level attribution: neither SHAP values [25] nor inspection of the attention maps of the auxiliary encoder was computed, so the paper cannot state which of the anomaly score, change-point flag, calendar variables or rolling volatility drives a particular forecast or a particular alert. Adding both is a direct and inexpensive extension, and it is the extension most likely to make the alerting layer acceptable to practitioners.

Finally, several considerations arise if the framework is operated continuously rather than evaluated offline. Alert volume is the binding practical constraint: at the F1-optimal threshold of 0.10 the model raises alerts on about 45% of store–item days, which no planning team can process, whereas the cost-optimal threshold of 0.17 reduces this to 2,628 alerts out of 45,000 instances, or 5.8%, which is a workable volume. This is a further argument for selecting thresholds by cost rather than by F1 in deployment. Recalibration is cheap, since temperature scaling fits a single scalar and can be repeated on a rolling window without touching the network, and the expected calibration error together with the realised alert rate provide two inexpensive drift monitors: a rising ECE indicates that the probability scale has shifted, while an alert rate departing from the value implied by the chosen threshold indicates that the score distribution has moved. Because the cost-optimal threshold depends on the assumed FN:FP ratio, and Table 13 shows that the preferred operating point changes between 2:1 and 10:1, the threshold should be re-derived whenever the underlying cost structure is revised rather than treated as a fixed model constant. Inference cost is not a limiting factor, as the model is a small network scored once per store–item and day, but the retraining cadence, the handling of new store–item combinations and the audit trail for issued alerts all require specification before deployment.

Collectively, these findings suggest that (i) dual-stream fusion improves point forecasts in a statistically reliable manner, (ii) lifting classification performance requires addressing imbalance and weak supervision beyond architecture changes, (iii) thresholds should be optimized to operational costs, and (iv) calibrated probabilities enable more trustworthy alerting in supply-chain early-warning systems.

6 Conclusion

This study proposed a unified dual-stream multitask deep-learning approach that integrates demand forecasting and statistically derived early-warning signal estimation within a single framework for supply-chain management. The sales time series and the auxiliary/anomaly series are encoded by an LSTM and a Transformer encoder, respectively; their branch representations are concatenated and jointly optimized via multitask learning to exploit cross-task information. The proposed dual-stream model (Proposed_MM) achieved statistically significant improvements over Baseline_LSTM and Ablation_Exog in demand forecasting, reducing both RMSE and MAE; under an identical protocol it outperformed classical univariate baselines, while a global gradient-boosted tree baseline retained a small advantage in point accuracy; and it attained the lowest expected cost among the compared configurations under cost-sensitive thresholding, although the margin over a never-alert reference policy is modest and disappears when the cost asymmetry is milder than 5:1. Although improvements in classification metrics were limited and the anomaly component is validated only against weak, self-referential labels, temperature scaling produced well-calibrated predictive probabilities, so the anomaly head can serve as a screening-level early-warning signal. Consequently, the proposed model can maintain stable demand forecasts under anomalous conditions while providing a foundation for cost-aware early-warning systems in supply-chain management.

Future work will focus on curating higher-quality anomaly labels from operational logs and systematically comparing noise-robust loss functions and data-augmentation strategies. In addition, this study will investigate an online setting in which dynamic thresholding and adaptive cost optimization are combined with probability calibration to assess whether expected cost can remain stable under pronounced distribution shifts. Several further directions follow from the limitations identified above. First, graph neural networks [42] can encode the relational structure of supply networks, such as store–item hierarchies, shared suppliers and logistics links, allowing anomaly signals to propagate across related series instead of being modelled independently for each series. Second, genuinely multimodal external sources, including promotions, weather, macroeconomic indicators and social-media streams, can replace the current self-referential auxiliary stream and would allow the early-warning component to be validated against externally observable events. Third, extending the regression head to probabilistic demand forecasting, for example with autoregressive likelihood models such as DeepAR [33] or with quantile outputs, would yield full predictive distributions that integrate naturally with the cost-sensitive alerting framework, since expected costs could then be computed against the entire forecast distribution rather than a point estimate. Finally, re-training all configurations under a rolling-origin protocol and validating the framework on operational data with expert-labelled incidents remain the decisive tests of practical value.

Two further extensions follow from the requirements of practice. Feature-level attribution through SHAP values [25] and inspection of the attention maps would explain why a given warning is raised, which is a precondition for managerial acceptance of an alerting system. A specification of the online operating regime, covering retraining cadence, recalibration frequency and drift monitoring, would in turn determine whether the calibrated screening layer remains dependable over time.

Acknowledgement

This study was done by research funds from Gwangju University in 2026.

References

[1] Baltrušaitis, T., Ahuja, C., Morency, L.-P. (2019). Multimodal machine learning: A survey and taxonomy, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 41, No. 2, 423-443, doi: 10.1109/TPAMI.2018.2798607.

[2] Baryannis, G., Dani, S., Antoniou, G. (2019). Predicting supply chain risks using machine learning: The trade-off between performance and interpretability, Future Generation Computer Systems, Vol. 101, 993-1004, doi: 10.1016/j.future.2019.07.059.

[3] Bergmeir, C., Benítez, J.M. (2012). On the use of cross-validation for time series predictor evaluation, Information Sciences, Vol. 191, 192-213, doi: 10.1016/j.ins.2011.12.028.

[4] Caruana, R. (1997). Multitask learning, Machine Learning, Vol. 28, No. 1, 41-75, doi: 10.1023/A:1007379606734.

[5] Chandola, V., Banerjee, A., Kumar, V. (2009). Anomaly detection: A survey, ACM Computing Surveys, Vol. 41, No. 3, 1-58, doi: 10.1145/1541880.1541882.

[6] Chen, T., Guestrin, C. (2016). XGBoost: A scalable tree boosting system, In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, USA, 785-794, doi: 10.1145/2939672.2939785.

[7] Chen, Z., Badrinarayanan, V., Lee, C.-Y., Rabinovich, A. (2018). GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks, In: Proceedings of the 35th International Conference on Machine Learning, Stockholm, Sweden, Proceedings of Machine Learning Research, Vol. 80, 794-803.

[8] Cleveland, R.B., Cleveland, W.S., McRae, J.E., Terpenning, I. (1990). STL: A seasonal-trend decomposition procedure based on Loess, Journal of Official Statistics, Vol. 6, No. 1, 3-73.

[9] de la Cruz Madrigal, V.H., Avelar Sosa, L., Mejía-Muñoz, J.M., García Alcaraz, J.L., Jiménez Macías, E. (2025). Dynamical system modeling for disruption in supply chain and its detection using a data-driven deep learning-based architecture, Logistics, Vol. 9, No. 2, Article No. 51, doi: 10.3390/logistics9020051.

[10] DeLong, E.R., DeLong, D.M., Clarke-Pearson, D.L. (1988). Comparing the areas under two or more correlated receiver operating characteristic curves: A nonparametric approach, Biometrics, Vol. 44, No. 3, 837-845, doi: 10.2307/2531595.

[11] Douaioui, K., Oucheikh, R., Benmoussa, O., Mabrouki, C. (2024). Machine learning and deep learning models for demand forecasting in supply chain management: A critical review, Applied System Innovation, Vol. 7, No. 5, Article No. 93, doi: 10.3390/asi7050093.

[12] Efron, B., Tibshirani, R. (1997). Improvements on cross-validation: The .632+ bootstrap method, Journal of the American Statistical Association, Vol. 92, No. 438, 548-560, doi: 10.1080/01621459.1997.10474007.

[13] Fu, Y., Fisher, M.L. (2023). The value of social media data in fashion forecasting, Manufacturing & Service Operations Management, Vol. 25, No. 3, 1136-1154, doi: 10.1287/msom.2023.1193.

[14] Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q. (2017). On calibration of modern neural networks, In: Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 1321-1330, doi: 10.48550/arXiv.1706.04599.

[15] Hochreiter, S., Schmidhuber, J. (1997). Long short-term memory, Neural Computation, Vol. 9, No. 8, 1735-1780, doi: 10.1162/neco.1997.9.8.1735.

[16] Hundman, K., Constantinou, V., Laporte, C., Colwell, I., Soderstrom, T. (2018). Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding, In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, London, UK, 387-395, doi: 10.1145/3219819.3219845.

[17] Hyndman, R.J., Khandakar, Y. (2008). Automatic time series forecasting: The forecast package for R, Journal of Statistical Software, Vol. 27, No. 3, 1-22, doi: 10.18637/jss.v027.i03.

[18] Kaggle. Demand forecasting – Store item demand forecasting challenge (Kernels only), from https://www.kaggle.com/c/demand-forecasting-kernels-only/data, accessed December 10, 2025.

[19] Kendall, A., Gal, Y., Cipolla, R. (2018). Multi-task learning using uncertainty to weigh losses for scene geometry and semantics, In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, USA, 7482-7491, doi: 10.1109/CVPR.2018.00781.

[20] Khedr, A.M., Rani, S. (2024). Enhancing supply chain management with deep learning and

machine learning techniques: A review, Journal of Open Innovation: Technology, Market, and Complexity, Vol. 10, No. 4, Article No. 100379, doi: 10.1016/j.joitmc.2024.100379.

[21] Lim, B., Arık, S.Ö., Loeff, N., Pfister, T. (2021). Temporal fusion transformers for interpretable multi-horizon time series forecasting, International Journal of Forecasting, Vol. 37, No. 4, 1748-1764, doi: 10.1016/j.ijforecast.2021.03.012.

[22] Lim, B., Zohren, S. (2021). Time-series forecasting with deep learning: A survey, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, Vol. 379, No. 2194, Article No. 20200209, doi: 10.1098/rsta.2020.0209.

[23] Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollár, P. (2017). Focal loss for dense object detection, In: Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 2999-3007, doi: 10.1109/ICCV.2017.324.

[24] Liu, Z., Zhao, N., Kim, S.H., Tadres, K., Dejanov, N., Rasasane, N. (2025). Integrative framework to enhance supply chain resilience through advanced forecasting, anomaly detection, and optimized resource allocation, Preprints, doi: 10.20944/preprints202505.0049.v1.

[25] Lundberg, S.M., Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, Vol. 30, 4765–4774.

[26] Makridakis, S., Spiliotis, E., Assimakopoulos, V. (2018). Statistical and machine learning forecasting methods: Concerns and ways forward, PLOS ONE, Vol. 13, No. 3, Article No. e0194889, doi: 10.1371/journal.pone.0194889.

[27] Makridakis, S., Spiliotis, E., Assimakopoulos, V. (2022). M5 accuracy competition: Results, findings, and conclusions, International Journal of Forecasting, Vol. 38, No. 4, 1346-1364, doi: 10.1016/j.ijforecast.2021.11.013.

[28] Malhotra, P., Ramakrishnan, A., Anand, G., Vig, L., Agarwal, P., Shroff, G. (2016). LSTM-based encoder-decoder for multi-sensor anomaly detection. *arXiv preprint*, arXiv:1607.00148. doi: 10.48550/arXiv.1607.00148.

[29] Moradi, F., Tarif, M., Homaei, M. (2025). Semi-supervised supply chain fraud detection with unsupervised pre-filtering, arXiv preprint arXiv:2508.06574, doi: 10.48550/arXiv.2508.06574.

[30] Nguyen, H.D., Tran, K.P., Thomassey, S., Hamad, M. (2021). Forecasting and anomaly detection approaches using LSTM and LSTM autoencoder techniques with the applications in supply chain management, International Journal of Information Management, Vol. 57, Article No. 102282, doi: 10.1016/j.ijinfomgt.2020.102282.

[31] Ni, S., Peng, Y., Liu, Z. (2022). Logistics demand forecast of fresh food e-commerce based on Bi-LSTM model, Journal of Computer and Communications, Vol. 10, No. 9, 51-65, doi: 10.4236/jcc.2022.109004.

[32] Nie, Y., Nguyen, N.H., Sinthong, P., Kalagnanam, J. (2023). A time series is worth 64 words: Long-term forecasting with transformers. In: *The Eleventh International Conference on Learning Representations (ICLR 2023)*, Kigali, Rwanda.

[33] Salinas, D., Flunkert, V., Gasthaus, J., Januschowski, T. (2020). DeepAR: Probabilistic forecasting with autoregressive recurrent networks, International Journal of Forecasting, Vol. 36, No. 3, 1181-1191, doi: 10.1016/j.ijforecast.2019.07.001.

[34] scikit-learn developers. Post-tuning the decision threshold for cost-sensitive learning, from https://scikit-learn.org/stable/auto_examples/model_selection/plot_cost_sensitive_learning.html, accessed December 10, 2025.

[35] Seman, L.O., Scapinello Aquino, L., Stefenon, S.F., Yow, K.-C., Mariani, V.C., dos Santos Coelho, L. (2025). Simultaneously anomaly detection and forecasting for predictive maintenance using a zero-cost differentiable architecture search-based network, Computers & Industrial Engineering, Vol. 208, Article No. 111412, doi: 10.1016/j.cie.2025.111412.

[36] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z. (2016). Rethinking the Inception architecture for computer vision, In: Proceedings of the IEEE Conference on Computer Vision

and Pattern Recognition, Las Vegas, USA, 2818-2826, doi: 10.1109/CVPR.2016.308.

[37] Tang, H., Yuan, Y., Gong, T. (2023). Industrial chain disruption events monitoring with deep learning methods: A practical application in China, International Journal of Intelligent Systems, Vol. 2023, Article No. 3338184, doi: 10.1155/2023/3338184.

[38] Tashman, L.J. (2000). Out-of-sample tests of forecasting accuracy: An analysis and review, International Journal of Forecasting, Vol. 16, No. 4, 437-450, doi: 10.1016/S0169-2070(00)00065-0.

[39] Truong, C., Oudre, L., Vayatis, N. (2020). Selective review of offline change point detection methods, Signal Processing, Vol. 167, Article No. 107299, doi: 10.1016/j.sigpro.2019.107299.

[40] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I. (2017). Attention is all you need, Advances in Neural Information Processing Systems, Vol. 30, 5998-6008, doi: 10.48550/arXiv.1706.03762.

[41] Wang, Y., Ma, X., Chen, Z., Luo, Y., Yi, J., Bailey, J. (2019). Symmetric cross entropy for robust learning with noisy labels, In: Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 322-330, doi: 10.1109/ICCV.2019.00041.

[42] Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., Yu, P.S. (2021). A comprehensive survey on graph neural networks, IEEE Transactions on Neural Networks and Learning Systems, Vol. 32, No. 1, 4-24, doi: 10.1109/TNNLS.2020.2978386.

[43] Xu, W., Cao, Y., Chen, R. (2024). A multimodal analytics framework for product sales prediction with the reputation of anchors in live streaming e-commerce, Decision Support Systems, Vol. 177, Article No. 114104, doi: 10.1016/j.dss.2023.114104.

[44] Zeng, A., Chen, M., Zhang, L., Xu, Q. (2023). Are transformers effective for time series forecasting?, Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 37, No. 9, 11121-11128, doi: 10.1609/aaai.v37i9.26317.

[45] Zhang, A., Lipton, Z.C., Li, M., Smola, A.J. (2023). Dive into deep learning, Cambridge University Press, Cambridge, UK.

[46] Zhang, X., Guo, C. (2024). Research on multimodal prediction of e-commerce customer satisfaction driven by big data, Applied Sciences, Vol. 14, No. 18, Article No. 8181, doi: 10.3390/app14188181.



Copyright ©2026 by the authors. Licensee Agora University, Oradea, Romania.
This is an open access article distributed under the terms and conditions of the Creative Commons Attribution-NonCommercial 4.0 International License.
Journal's webpage: http://univagora.ro/jour/index.php/ijccc/



This journal is a member of, and subscribes to the principles of,
the Committee on Publication Ethics (COPE).
https://publicationethics.org/members/international-journal-computers-communications-and-control

Cite this paper as:
Park, K. (2026). A dual-stream multitask deep learning model for supply chain demand forecasting with statistically derived early-warning signals, *International Journal of Computers Communications & Control*, 21(6), 7694, 2026.
https://doi.org/10.15837/ijccc.2026.6.7694