ORIGINAL ARTICLE

A Comparative Study of Forecasting Models Using the Temporal Fusion Transformer in Pharmacy Chain Sales Systems

Chin-Sheng Chen[1], Nien-Tsu Hu[1*], and Li-Wen Feng[1]

[1]*Graduate Institute of Automation Technology, National Taipei University of Technology, Taipei, Taiwan*
(**Received** 08 January 2026, **Revised** 23 March 2026, 11 April 2026, **Accepted** 12 April 2026)
*Corresponding author: nthu@ntut.edu.tw;
DOI: 10.5875/5psfnf83

Abstract: This study investigates the application of the Temporal Fusion Transformer (TFT) model for multi-product daily sales forecasting in pharmacy chain sales systems. Using real transactional data from a pharmacy chain, the study constructs a forecasting framework based on the top 20 best-selling products. Historical sales records from 2025, comprising 327,726 transactions, are divided into training and validation sets, while sales data from January 2026, comprising 25,284 transactions, are reserved as an unseen test set for multi-horizon evaluation. To ensure a fair comparison, TFT is benchmarked against Linear Regression, Random Forest, XGBoost, and standard LSTM models. The experimental design incorporates daily aggregation, temporal feature derivation, rolling-mean smoothing, and sequence construction to model heterogeneous retail information, including static attributes and time-varying inputs. Forecasting performance is evaluated using Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and the Coefficient of Determination (R^2). Experimental results show that TFT achieves the best overall performance, with MAE = 104.165, RMSE = 149.717, and R^2 = 0.986, outperforming all baseline models. These findings indicate that TFT provides a robust and interpretable forecasting framework for pharmacy retail automation, inventory planning, and replenishment decision support.

Keywords: Automation; Machine Learning; Deep Learning; Temporal Fusion Transformer; Sales Forecasting; Retail Supply Chain.

Introduction

The retail industry is undergoing rapid digital transformation, and demand forecasting has become a central component of inventory planning, operational coordination, and decision support. Recent studies in retail supply chain management and retail demand forecasting have shown that machine learning and deep learning methods are increasingly being adopted to improve forecasting performance in data-rich business environments [2], [5]. More broadly, forecasting research has emphasized that modern time-series applications often involve heterogeneous inputs, long-range temporal dependencies, and the need to balance predictive accuracy with interpretability for practical deployment [18], [20], [24].

However, sales forecasting in pharmacy chain systems remains challenging because the data are inherently heterogeneous. A typical pharmacy retail

dataset includes static attributes such as store identifiers and product categories, known future variables such as weekdays and calendar effects, and time-varying historical observations such as past sales quantities. Conventional machine learning models, including Random Forest and XGBoost, are often effective on structured tabular data but are not specifically designed to model sequential dependence over forecasting horizons [9],[10]. Likewise, Long Short-Term Memory (LSTM) networks are well-suited to sequential modeling, yet they do not natively integrate static covariates and known future inputs within a unified, interpretable framework [3],[8].

To address these limitations, this study investigates the Temporal Fusion Transformer (TFT), an interpretable multi-horizon forecasting architecture that integrates recurrent sequence modeling, variable selection, gating mechanisms, and attention-based long-range dependency learning within a unified framework [1]. Recent studies on Transformer-based forecasting and time-series modeling have further shown that attention-based architecture has become an important direction in modern forecasting research [6],[17], [21], [23]. Accordingly, this paper compares TFT with Linear Regression, Random Forest, XGBoost, and standard LSTM models using real pharmacy chain sales data, with the goal of evaluating whether TFT can deliver superior predictive performance and greater practical value for retail automation and inventory-related decision-making [1], [18], [20].

Related Work

Deep Learning Applications in Retail Forecasting

Retail demand forecasting has evolved from conventional regression-based approaches toward machine learning and deep learning frameworks that can handle high-dimensional, heterogeneous business data. In the retail supply chain domain, deep learning has been recognized as an important enabler of digital transformation and operational improvement [2]. Recent retail-oriented studies have further shown that machine learning and data-driven forecasting can improve demand prediction by leveraging richer transactional and contextual information [4],[5]. In addition, large-scale temporal and contextual data have been shown to play an important role in supply-chain forecasting, particularly when forecasts must reflect dynamic demand patterns and external signals [16].

Another important development in forecasting research is the growing emphasis on interpretability. In real-world business settings, managers require not only accurate forecasts but also transparent explanations regarding the variables and historical patterns that drive model outputs. This issue has been highlighted in recent explainable artificial intelligence research on time-series data, which argues that model transparency is increasingly important for trustworthy operational decision-making [7],[19].

Broader Time-Series Forecasting Methodologies

From a broader methodological perspective, time-series forecasting includes classical statistical methods, machine learning ensembles, recurrent neural networks, and Transformer-based architectures [20], [24]. Ensemble learning approaches such as Random Forest and XGBoost remain strong baselines for structured tabular data because of their robustness and ability to model nonlinear feature interactions [9], [10]. Recurrent neural networks, especially LSTM, have also been widely adopted for sequential modeling and long-range dependency learning in temporal data [8].

Deep learning has substantially broadened the scope of forecasting research. Survey studies have shown that deep learning has become a major paradigm for time-series forecasting due to its ability to learn complex temporal patterns from large-scale data [18]. Representative deep forecasting architectures include DeepAR for probabilistic forecasting across related time series [13] and N-BEATS for interpretable neural forecasting [12],[15]. Transformer-based models have further expanded this literature. Autoformer introduced decomposition-based Transformer forecasting for long-horizon prediction [14], while recent surveys and comparative studies have examined the effectiveness of Transformer architectures for time-series analysis more generally [17], [21], [22],[23].

Among these models, the Temporal Fusion Transformer (TFT) is particularly relevant to the present study because it was explicitly designed for interpretable multi-horizon forecasting with static covariates, known future inputs, and observed historical variables [1]. By integrating variable selection networks, gating mechanisms, recurrent local processing, and interpretable attention layers, TFT provides a unified architecture that addresses both predictive accuracy and model interpretability [1].

Research Gap and Positioning of the Present Study

Although prior studies have discussed retail supply chain deep learning [2], retail demand forecasting [5], temporal big-data forecasting in operational settings [16], and broader deep learning and Transformer-based forecasting methodologies [17], [18], [20], [21], [23], [24], relatively limited research has directly compared TFT with

conventional regression models, tree-based ensemble models, and recurrent neural networks in a pharmacy chain sales context using real transactional retail data under a unified experimental framework. Accordingly, the present study positions itself at the intersection of retail automation, demand forecasting, and interpretable deep learning by systematically comparing Linear Regression, Random Forest, XGBoost, LSTM, and TFT on pharmacy chain sales data.

Methodology

The proposed forecasting framework is based on the Temporal Fusion Transformer (TFT), which was originally developed for interpretable multi-horizon time-series forecasting under heterogeneous input conditions [1]. Unlike conventional forecasting models that process all covariates uniformly, TFT explicitly distinguishes static attributes, known future inputs, and observed historical variables, and integrates them through variable selection, recurrent sequence modeling, and interpretable attention. Such a design is particularly appropriate for pharmacy chain sales forecasting because demand is jointly influenced by product identity, calendar regularities, and recent demand momentum rather than by a single homogeneous signal [1], [18], [20].

Forecasting Task Formulation

Let i denote product $i-th$ product series, t denote the current day, and $y_{i,t}$ denote the observed daily sales quantity of product i at time t. The objective of this study is to estimate future sales over a 31-day forecasting horizon using the most recent 90 days of historical observations. The forecasting problem can be expressed as

$$\hat{y}_{i,t+\tau}^{(q)} = f_q\left(S_i, X_{i,t-L+1:t+\tau}, Z_{i,t-L+1:t}\right), \tau = 1, \dots, H \qquad (1)$$

where $\hat{y}_{i,t+\tau}^{(q)}$ denotes the predicted value for quantile q at future forecasting step τ, $f_q(\cdot)$ denotes the forecasting function for quantile q, L is the encoder length, H is the prediction horizon, $\mathbf{s}_i$ denotes the static covariates of product i, $\mathbf{x}_{i,t}$ denotes the time-varying known inputs, $\mathbf{z}_{i,t}$ denotes the time-varying observed inputs, and τ indexes the future forecasting step. In this study, $L = 90$ and $H = 31$, corresponding to a 90-day look-back window and a 31-day multi-horizon forecasting task [1].

Chin-Sheng Chen e-mail: saint@ntut.edu.tw

Nien-Tsu Hu e-mail: nthu@ntut.edu.tw

Li-Wen Feng e-mail: t111618510@ntut.edu.tw

Data Pre-processing and Feature Extraction

The original pharmacy sales data consist of raw transactional records and, therefore, cannot be directly used as model-ready temporal sequences. To construct a structured forecasting dataset, all transactions were first aggregated at the product-day level, where the daily sales quantity of each product was obtained by summing transaction quantities on the same calendar day. This temporal aggregation is theoretically necessary because it converts noisy event-level observations into stable daily demand signals and aligns the data representation with the monthly replenishment-oriented forecasting horizon.

To preserve temporal continuity, a complete date-product grid was constructed and missing observations were filled with zero sales. A continuous integer time index, $time\_idx$, was then assigned to each day to represent the chronological order of the sequence. This design allows the model to explicitly distinguish temporal distance and order, which is essential for sequence encoding and attention-based dependency learning.

To capture seasonal and cyclical effects, calendar-derived features were extracted from the transaction date, including *month*, *day_of_month*, and *day_of_week***.** These variables were treated as known future inputs, since they are available for all future horizons before prediction time. In addition, a 7-day rolling mean of daily sales was computed for each product:

$$rolling\_mean_7(t) = \frac{1}{7}\sum_{k=0}^{6} y_{i,t-k} \qquad (2)$$

where $rolling_mean_7(t)$denotes the 7-day rolling mean at time step t, $y_{i,t-k}$ denotes the observed sales quantity of product i at lag k ,and $k = 0, \dots, 6$ represents the seven daily observations included in the smoothing window. This feature was included to summarize recent demand momentum and reduce short-term noise. The theoretical rationale is that pharmacy retail data often exhibit abrupt fluctuations caused by promotions, temporary stock-outs, and isolated demand spikes. The rolling mean suppresses such high-frequency volatility while preserving the underlying short-term demand trend, thereby improving the robustness of feature extraction.

Accordingly, the input variables were organized into TFT-compatible groups as follows:

Static categoricals:

$product\_no$, $product\_category$

Time-varying known categoricals:

$month$, day_of_month, day_of_week

Time-varying known reals:

$time\_idx$, $relative\_time\_idx$

Time-varying unknown reals:

$daily\_quantity$, $rolling\_mean\_7$

This decomposition is theoretically consistent with TFT, which explicitly models static background information, known future drivers, and observed historical dynamics through separate processing pathways [1].

Retail Data Source, Variables, and Characteristics

The retail dataset used in this study was obtained from an anonymized pharmacy-chain operational database authorized by the collaborating company. The database integrates pharmacy transaction records generated by the pharmacy information platform developed by SJEN Health Technology Co., Ltd. The original database consisted of transaction-level sales records generated during the routine operation of the pharmacy sales information system. Two data extracts were used in this study. The first covered the period from 2025-01-01 to 2025-12-31 and contained 327,726 transaction records, while the second covered the period from 2026-01-01 to 2026-01-31 and contained 25,284 transaction records. All records were de-identified prior to analysis, and no personally identifiable information was included in model training.

The raw database contained multiple field types as shown in Table 1, including store identifier (*storeNo*), product identifier (*product_no*), product name (*product_name*), product category (*product_category*), transaction timestamp (*transaction_date*), sales quantity (*quantity*), transaction amount (*total_amount*), payment method (*payment_method*), and membership identifier (*memberNo*). These fields jointly represent a heterogeneous retail data environment involving identifier variables, textual descriptors, datetime information, and numerical transaction attributes. Among them, *product_no*, *product_category*, *transaction_date*, and *quantity* were the core variables used to construct the forecasting dataset.

A key characteristic of the original dataset is that it was recorded at the transaction level rather than as pre-structured temporal sequences. Consequently, the observations were irregularly distributed across time and could not be directly used as input to a deep time-series forecasting model. To address this issue, the transaction records were first aggregated into daily sales observations at the product-day level. Specifically, quantities were summed for each product on each date to generate daily demand values. Next, the top 20 products were selected according to cumulative sales quantity in 2025, and only these products were retained for subsequent modeling. To preserve temporal continuity, a complete date-product grid covering the period from 2025-01-01 to 2026-01-31 was constructed, and missing observations were filled with zero sales. This process transformed sparse and irregular transaction records into a balanced daily panel suitable for multi-horizon forecasting.

Another important characteristic of the dataset is its heterogeneous structure. It simultaneously contains static product attributes, deterministic calendar information, and time-varying sales observations. This property makes the dataset particularly suitable for the Temporal Fusion Transformer framework, which was explicitly designed to integrate static covariates, known future inputs, and observed historical variables within a unified architecture. In addition, the retail demand series exhibited short-term volatility, intermittent zero-sales days, and calendar-related fluctuations. Therefore, derived temporal features such as *day_of_week*, *month*, and *day_of_month*, together with a 7-day rolling mean (*rolling_mean_7*), were incorporated into the final forecasting pipeline.

Table 1. Summary of Variables in the Raw Retail Dataset

Variable	Type	Description
storeNo	Numeric	Store identifier
product_no	Numeric	Unique product identifier
product_name	Text	Product name
product_category	Numeric	Product category code
transaction_date	Datetime	Transaction timestamp
quantity	Numeric	Sales quantity per transaction
total_amount	Numeric	Total sales amount
payment_method	Text	Payment method
memberNo	Numeric	Membership identifier

Based on the above variable structure and data characteristics, the forecasting pipeline was designed to transform transaction-level retail records into model-ready daily sequences for multi-horizon prediction.

Technical Description of the Proposed TFT Architecture

As shown in Figure 1, the TFT architecture combines recurrent local processing with attention-based global dependency learning [1]. The technical logic of the proposed model can be described in four stages: input encoding, variable selection, sequential encoding-decoding, and temporal fusion.

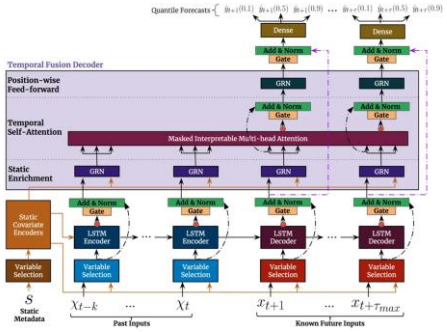

Figure 1. TFT Architecture [1]

Input Encoding and Static Conditioning

Categorical inputs were transformed into trainable embedding vectors, while continuous variables were normalized using a group-wise normalization strategy on a per-product basis. This step is theoretically justified because products in pharmacy retail systems often have very different demand scales. Without product-level normalization, high-volume products would dominate the optimization objective and bias the gradient updates. Group-wise normalization stabilizes training while preserving the relative temporal patterns within each product sequence.

Static variables such as product number and product category were processed by static covariate encoders, which transform time-invariant metadata into context vectors. These context vectors are not treated as simple auxiliary inputs; instead, they condition downstream modules, including variable selection and temporal feature transformation. This design is theoretically preferable because static information influences the full demand trajectory of a product and should therefore modulate the temporal representation learning process rather than merely appear as additional columns.

Variable Selection Networks

A central component of TFT is the Variable Selection Network (VSN), which dynamically determines the most informative variables at each time step [1]. Let $x_{j,t}$ denote the j-th candidate feature at time t. Each candidate variable is first transformed by a feature-specific Gated Residual Network (GRN):

$$\tilde{x}_{j,t} = GRN_j(x_{j,t}, c_s) \tag{3}$$

where $\tilde{x}_{j,t}$ denotes the transformed representation of the $j-th$ input feature, $GRN_j(\cdot)$ denotes the feature-specific Gated Residual Network, and c_s denotes the static context vector derived from the static covariates. The transformed variables are then weighted through a softmax gating mechanism:

$$\alpha_{j,t} = \frac{\exp(e_{j,t})}{\sum_{k=1}^{m} \exp(e_{k,t})} \tag{4}$$

and the fused temporal representation is obtained by

$$v_t = \sum_{j=1}^{m} \alpha_{j,t} \tilde{x}_{j,t} \tag{5}$$

where $\alpha_{j,t}$ denotes the importance weight assigned to the j-th variable at time t, $e_{j,t}$ denotes the pre-softmax relevance score, m is the number of candidate input variables, and v_t denotes the fused temporal representation after variable selection. The theoretical significance of this mechanism is that the predictive relevance of retail covariates is not constant across time. For example, recent sales dynamics may dominate during stable demand periods, whereas weekday-related signals may become more influential under strong weekly seasonality. By adaptively weighting candidate features, the VSN suppresses irrelevant noise and provides an interpretable measure of variable importance [1], [19].

LSTM Encoder-Decoder

After variable selection, the model applies an LSTM encoder-decoder to capture local sequential structure. The encoder processes the previous 90 days of selected historical inputs, while the decoder processes the known future inputs over the 31-day prediction horizon:

$$(h_t^{enc}, c_t^{enc}) = LSTM_{enc}(v_t^{hist}, h_{t-1}^{enc}, c_{t-1}^{enc}) \tag{6}$$

$$(h_{t+\tau}^{dec}, c_{t+\tau}^{dec}) = LSTM_{dec}(v_{t+\tau}^{fut}, h_{t+\tau-1}^{dec}, c_{t+\tau-1}^{dec}) \tag{7}$$

where h_t^{enc} and c_t^{enc} denote the hidden state and cell state of the encoder at time t, respectively, and $h_{t+\tau}^{dec}$ and $c_{t+\tau}^{dec}$ denote the hidden state and cell state of the decoder at forecasting step τ. Moreover, v_t^{hist} denotes the selected historical input representation, and $v_{t+\tau}^{fut}$ denotes the selected known future input representation. The theoretical role of the recurrent block is to preserve short-term order, local temporal continuity, and autoregressive demand structure. This is particularly relevant in pharmacy demand forecasting, where daily sales are frequently associated with short replenishment cycles, repeated customer purchasing behavior, and recent demand momentum [8].

Temporal Fusion Decoder and Interpretable Attention

Although LSTM can encode local sequence dependencies, it is less effective at capturing long-range interactions across the full temporal context. TFT therefore introduces a Temporal Fusion Decoder with interpretable multi-head self-attention [1]. The attention mechanism can be expressed as

$$Attention(Q, K, V) = softmax\left(\frac{QK^{\top}}{\sqrt{d_k}} + M\right)V \tag{8}$$

where Q, K, and V denote the query, key, and value matrices, respectively, d_k is the dimensionality of the key vectors, and M *is* a causal mask that prevents information leakage from future observations. The resulting attention output represents a weighted combination of temporally relevant representations for each forecasting step. This masking operation is essential because a valid forecasting model must not access unknown future target values during inference.

The theoretical advantage of multi-head attention is that it enables the model to simultaneously learn different temporal dependency patterns, including short-term momentum, weekly cycles, and broader long-range interactions [17], [21], [23]. For pharmacy retail demand, such multi-scale dependencies often coexist, and their relative importance may vary across products and time periods. In addition, the resulting attention weights provide time-step-level interpretability, allowing the model to identify which historical intervals contribute most strongly to a given forecast.

Quantile Output Layer and Loss Function

Instead of producing only a single point estimate, the proposed TFT generates **quantile forecasts** to represent predictive uncertainty. Let $\hat{y}_{n,\tau}^{(q)}$ denote the predicted value for the q-th quantile of the n-th training sample at the τ-th forecasting step. The model is trained using **Quantile Loss**, also known as pinball loss:

$$\rho_q(u) = q\max(u,0) + (1-q)\max(-u,0), \qquad u = y - \hat{y}^{(q)} \quad (9)$$

where $\rho_q(u)$ denotes the quantile loss at quantile q, u is the prediction error, y is the actual target value, and $\hat{y}^{(q)}$ is the predicted value for quantile q.

The overall training objective is defined as

$$\mathcal{L} = \frac{1}{N}\sum_{n=1}^{N}\sum_{\tau=1}^{H}\sum_{q\in Q}\rho_q\left(y_{n,\tau} - \hat{y}_{n,\tau}^{(q)}\right) \quad (10)$$

where N denotes the total number of training sequences, H denotes the forecasting horizon, τ indexes the future prediction step, and Q denotes the set of target quantiles. In addition, $y_{n,\tau}$ represents the ground-truth value of the n-th training sample at forecasting step τ, while $\hat{y}_{n,\tau}^{(q)}$ represents the corresponding prediction for quantile q. In this study, $H = 31$, corresponding to a 31-day multi-horizon forecasting task. This probabilistic formulation is theoretically preferable to mean-only prediction because pharmacy demand is often skewed and heteroscedastic, and the operational costs of over-forecasting and under-forecasting are asymmetric. Quantile outputs therefore provide more informative decision support for inventory planning and replenishment management [1].

Experimental Setup and Implementation Details

The experiments were implemented in Python using PyTorch Forecasting and PyTorch Lightning. Model development and preliminary debugging were performed on a local workstation equipped with an Intel Core i7-13620H CPU, 32 GB RAM, and an NVIDIA GeForce RTX 4060 Laptop GPU, whereas final model training and benchmarking were conducted in Google Colab using an NVIDIA A100-SXM4-40GB GPU. The software environment comprised Python 3.12.11, PyTorch 2.8.0, PyTorch Lightning 2.6.1, and PyTorch Forecasting 1.6.1. To enhance reproducibility, the random seed was fixed at 42. In addition, adaptive gradient-based optimization strategies, exemplified by the Adam framework, are widely recognized as an effective foundation for stable neural network training in modern deep learning implementations [11].

The forecasting workflow began with transaction preprocessing. Raw *transaction_date* values were converted into datetime format, after which the sales records were aggregated into daily observations at the product-day level. A continuous integer time index (*time_idx*) was then generated to represent the sequence's absolute chronological order. In addition, a relative time index (*relative_time_idx*) was introduced during TimeSeriesDataSet construction to indicate each time step's relative position with respect to the forecasting origin within the encoder–decoder window. Unlike *time_idx*, which represents the global temporal order across the full dataset, *relative_time_idx* provides sequence-level positional information, helping the model distinguish historical context from future prediction steps in multi-horizon forecasting. Calendar-derived features, including *day_of_week*, *month*, and *day_of_month*, were extracted from the date field. In addition, a 7-day rolling mean of daily sales (*rolling_mean_7*) was computed for each product in order to capture recent demand momentum and reduce short-term fluctuations.

The final input variables were organized according to the TFT architecture's heterogeneous-input design. *product_no* and *product_category* were treated as static categorical variables. *day_of_week*, *month*, and *day_of_month* were treated as time-varying known categorical variables because they are deterministically available for future dates. *time_idx* and *relative_time_idx* were treated as time-varying known real variables. Here, *time_idx* represents the absolute chronological order of observations across the full dataset, whereas *relative_time_idx* represents the relative temporal position of each step within the encoder–decoder sequence for multi-horizon forecasting. *daily_quantity* and *rolling_mean_7* were treated as time-varying, real-

valued unknown variables. All categorical variables were encoded using *NaNLabelEncoder*, and the target variable was normalized by product using *GroupNormalizer* with a softplus transformation to reduce scale heterogeneity across products.

A strictly chronological split was adopted to avoid information leakage. Data from 2025-01-01 to 2025-11-30 were used for training, the final 31 days of 2025 (2025-12-01 to 2025-12-31) were reserved for validation, and data from 2026-01-01 to 2026-01-31 served as a fully unseen out-of-sample test period. The *TimeSeriesDataSet* was configured with a maximum encoder length of 90 days, a minimum encoder length of 45 days, and a maximum prediction length of 31 days, thereby enabling direct 31-day multi-horizon forecasting. This study adopts a 90-day historical context with a chronologically separated validation and test design.

The final TFT model was trained with a learning rate of 0.005, hidden size of 48, attention head size of 8, dropout rate of 0.3, hidden continuous size of 8, and output size of 7 for quantile forecasting. The batch size was set to 64, and the maximum number of training epochs was 40. Model training was managed through the PyTorch Lightning Trainer with gradient clipping set to 0.1. Early stopping based on validation performance with a patience of 7 epochs, together with learning-rate monitoring, was applied to improve convergence stability and reduce overfitting. After training, the fitted TFT model was used to generate forecasts for the unseen January 2026 period, and forecasting performance was evaluated using MAE, RMSE, and R^2.

The overall implementation procedure of the proposed forecasting framework is summarized in Figure 2.

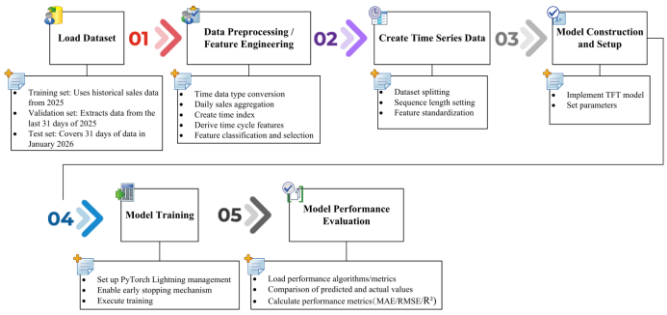

Figure 2. Processing Flow

Evaluation Metrics:

Performance was quantified using three standard metrics: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and the Coefficient of Determination (R^2) [20], [24].

Mean Absolute Error (MAE):

MAE measures the average magnitude of the errors in a set of predictions, without considering their direction [24]. It is defined as:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{11}$$

Root Mean Squared Error (RMSE):

RMSE is the square root of the average squared difference between the predicted and actual values. Because the errors are squared before averaging, RMSE assigns a relatively larger penalty to large deviations [24]. It is defined as:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \tag{12}$$

Coefficient of Determination (R^2):

The coefficient of determination measures the proportion of variance in the observed values that is explained by the prediction model [20], [24]. It is defined as

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2} \tag{13}$$

where n denotes the number of samples, y_i denotes the actual value, $\hat{y}_i$ denotes the predicted value, and $\bar{y}$ denotes the mean of the actual values. Lower values of MAE and RMSE indicate better predictive accuracy, whereas an R^2 value closer to 1 indicates stronger explanatory performance.

Results and Discussion

Learning Dynamics and Convergence Analysis

To evaluate the optimization stability of the proposed TFT model, Figure 3 shows the training and validation loss curves, and Table 2 summarizes the corresponding numerical values over epochs. As shown in Figure 3, both curves decrease sharply during the early training stage, indicating that the model rapidly captures the dominant temporal structure embedded in the pharmacy sales data. More specifically, Table 2 shows that the training loss decreases from 23.64 at Epoch 0 to 2.62 at Epoch 7, whereas the validation loss decreases from 16.55 to 2.19 over the same interval. This rapid convergence suggests that the adopted feature engineering strategy, including calendar-derived variables and the 7-day rolling mean, provides highly informative inputs for sequence learning.

Table 2. Training/validation loss values of the TFT model across epochs.

Epoch	Training Loss	Validation Loss
0.00	23.64	16.55
1.00	12.98	6.91
2.00	7.58	4.01
3.00	5.54	3.13
4.00	4.25	2.42
5.00	3.73	2.27
6.00	3.22	2.17
7.00	2.62	2.19

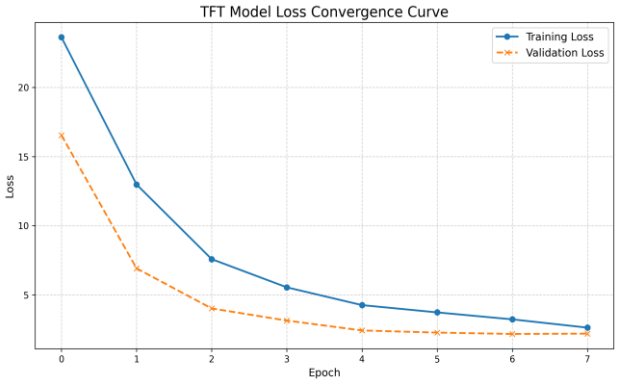

Figure 3. Training and validation loss curves of the TFT model

Another important observation from Figure 3 is that the validation loss remains close to the training loss throughout the fitting process, without severe divergence. This behavior indicates that the model does not merely memorize the training data but instead learns temporal representations that generalize satisfactorily. The relatively small generalization gap further suggests that the current regularization design is effective. In this study, the TFT model is trained with a learning rate of 0.005, hidden size of 48, attention head size of 8, dropout rate of 0.3, batch size of 64, and a maximum of 40 epochs, while early stopping with a patience of 7 epochs is used to enhance convergence stability and reduce overfitting. Therefore, the evidence from Figure 3 and Table 2 supports the conclusion that the proposed forecasting framework achieves stable, efficient convergence in the current pharmacy forecasting setting.

Predictive Performance on the Unseen Test Set

To assess the practical forecasting capability of the proposed framework, the trained TFT model is further evaluated on the fully unseen January 2026 test period. Figure 4 compares predicted and actual sales, while Table 3 reports product-level actual totals, predicted totals, and their differences. As illustrated in Figure 4, the predicted trajectory closely follows the actual demand pattern, indicating that the model preserves not only the overall sales trend but also the major short-term fluctuations in the target series. This visual consistency is also consistent with the TFT model's quantitative performance, which achieves MAE = 104.165, RMSE = 149.717, and R^2 = 0.986 on the unseen test horizon. These results demonstrate both high predictive accuracy and strong explanatory performance.

Table 3. Product-level comparison of actual totals, predicted totals, and differences for the top 20 products.

Product_ID	Actual	Predicted	Difference
2004134	7150	7428.30	278.30
2004151	3210	3266.90	56.90
2021163	2200	2433.90	233.90
2000810	2156	2257.30	101.30
2003851	2116	2121.00	5.00
2010280	2000	2165.70	165.70
2014263	2000	2380.30	380.30
2005133	1980	1985.70	5.70
2005797	1920	2076.00	156.00
2004927	1710	1712.60	2.60
2003039	1700	1700.60	0.60
2013190	1670	1720.10	50.10
2008240	1641	1638.50	-2.50
2021699	1594	1651.00	57.00
2021700	1497	1619.30	122.30
2006102	1355	1348.30	-6.70
2002137	1322	1567.80	245.80
2002090	1280	1256.10	-23.90
2001085	1196	1232.90	36.90
2006931	1169	1320.80	151.80

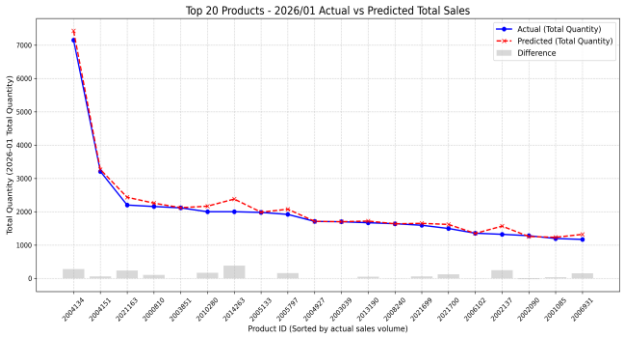

Figure 4. Comparison between actual and predicted sales values of the TFT model on the January 2026 test set

The detailed values in Table 3 further show that the deviations between actual and predicted monthly totals remain small for several products. For example, product 2003039 shows a difference of only 0.60, product

2004927 shows a difference of 2.60, and product 2003851 shows a difference of 5.00. These examples indicate that the proposed model can produce highly accurate estimates for multiple high-volume products. At the same time, some products still exhibit relatively large positive deviations, such as product 2014263 and product 2002137, suggesting that certain stock-keeping unit (SKU)-specific irregularities and abrupt sales changes are not fully captured under the current feature setting. Nevertheless, the combined evidence from Figure 4 and Table 3 confirms that the TFT model exhibits a forecasting pattern that remains highly consistent with actual retail demand, thereby providing meaningful support for inventory planning and replenishment decisions.

Comparative Results Across Benchmark Models

To compare the overall forecasting effectiveness of different methods, Table 4 summarizes the MAE, RMSE, R^2, and total execution time for Linear Regression, Random Forest, XGBoost, LSTM, and TFT. In addition, Figures 5-7 visualize the results separately for MAE, RMSE, and R^2. According to Table 4, TFT achieves the best overall performance among all evaluated models, with the lowest MAE (104.165), the lowest RMSE (149.717), and the highest R^2 (0.986). By contrast, Linear Regression, Random Forest, XGBoost, and LSTM all yield inferior results to different degrees. The visual comparisons in Figures 5-7 further confirm that TFT's superiority is consistent across all three evaluation perspectives, not limited to a single metric.

Table 4. Performance comparison of Linear Regression, Random Forest, XGBoost, LSTM, and TFT

Models	MAE	RMSE	R^2	Execution time (seconds)
Linear Regression	130.337	194.283	0.735	21.8
Random Forest	162.953	227.749	0.635	61.6
XGBoost	140.702	164.903	0.809	55.4
LSTM	130.442	176.006	0.782	222.6
TFT	**104.165**	**149.717**	**0.986**	**118.1**

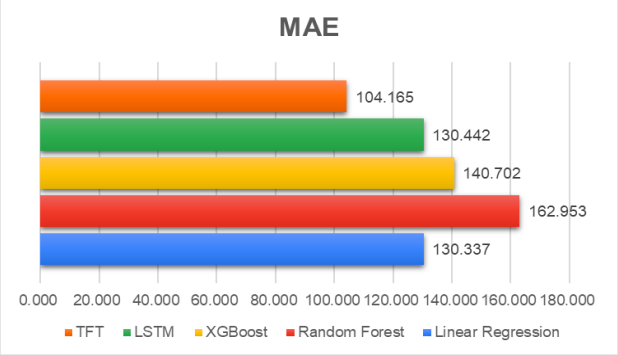

Figure 5. MAE comparison across forecasting models

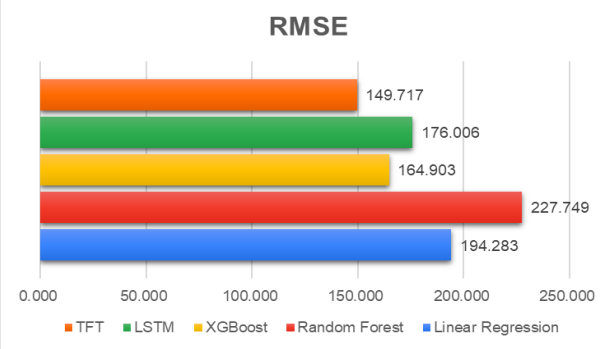

Figure 6. RMSE comparison across forecasting models

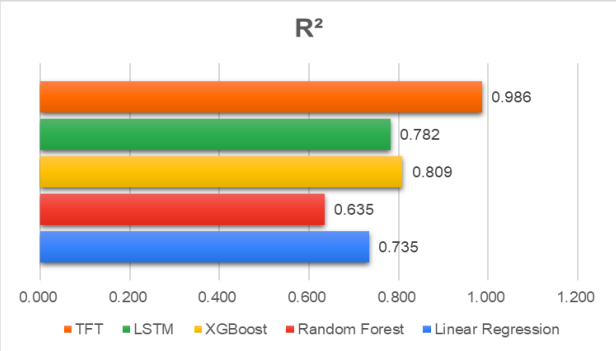

Figure 7. R^2 comparison across forecasting models

More specifically, Linear Regression yields MAE = 130.337, RMSE = 194.283, and R^2 = 0.735, indicating limited ability to model nonlinear retail demand behavior. Random Forest and XGBoost improve nonlinear fitting, but their performance remains lower than that of TFT, with Random Forest producing MAE = 162.953, RMSE = 227.749, and R^2 = 0.635, and XGBoost producing MAE = 140.702, RMSE = 164.903, and R^2 = 0.809. LSTM demonstrates stronger temporal modeling capability than the conventional machine learning baselines, yet it still underperforms TFT, with MAE = 130.442, RMSE = 176.006, and R^2 = 0.782. These findings indicate that TFT's performance advantage is substantial rather than marginal.

From the computational perspective, TFT requires 118.1 seconds of total execution time. Although it is not the fastest model, it remains considerably more efficient than LSTM, which requires 222.6 seconds to train and delivers markedly better predictive accuracy.

Therefore, the combined evidence from Table 4 and Figures 5-7 suggests that TFT achieves a favorable balance between forecasting quality and computational cost, making it particularly attractive for pharmacy chain decision-support systems where forecast reliability is more critical than minimal runtime alone.

Interpretability and Architecture Validation

Although the comparative results demonstrate that TFT achieves the best overall forecasting performance among all evaluated models, aggregate metrics alone are insufficient to fully substantiate the proposed architecture's validity. Because TFT is designed not only for multi-horizon forecasting but also for interpretability, additional technical evidence is required to verify whether the model learns meaningful internal representations from heterogeneous pharmacy sales data. Therefore, this subsection further evaluates the proposed model through feature importance analysis, temporal attention analysis, and product-level MAE–RMSE trade-off analysis. These complementary results provide stronger evidence that the superiority of TFT in the present study arises not merely from numerical optimization, but from structurally meaningful representation learning aligned with the retail forecasting task.

Global Feature Importance Analysis

Figure 8 presents the learned feature-importance structure of the proposed TFT model from the static, encoder, and decoder pathways. These analyses are important because they reveal how the model allocates predictive emphasis to different categories of inputs rather than treating all covariates uniformly. In the static pathway, product identity is most important, followed by product category. This result indicates that product-level identity plays a major conditioning role in pharmacy demand forecasting. Such a finding is theoretically consistent with the retail setting, where different stock-keeping units (SKUs) may exhibit distinct demand scales, replenishment cycles, and purchasing regularities.

In the encoder pathway, the most influential variables are *day_of_month*, *daily_quantity*, and *rolling_mean_7*. This pattern suggests that the model primarily relies on three sources of information when encoding the historical window: intra-month calendar position, raw recent sales dynamics, and smoothed short-term demand momentum. The relatively high contribution of *rolling_mean_7* is especially meaningful because it indicates that the smoothing-based feature engineering strategy adds forecasting value beyond merely duplicating the raw target signal. This interpretation is also consistent with the current forecasting pipeline, in which static covariates, known future inputs, and observed historical variables are explicitly organized into separate functional groups.

In the decoder pathway, *day_of_month* becomes the dominant known future input, while *time_idx* and *relative_time_idx* provide secondary contributions. This result indicates that the model relies strongly on the monthly calendar position when forming multi-step future forecasts. For pharmacy sales systems, this is an operationally plausible finding because recurring refill behavior, monthly consumption rhythms, and calendar-timed purchasing may be more strongly associated with intra-month position than with broader monthly labels alone. Therefore, the variable-importance results collectively support the claim that the proposed TFT architecture effectively differentiates static metadata, observed historical variables, and known future inputs in a functionally coherent manner.

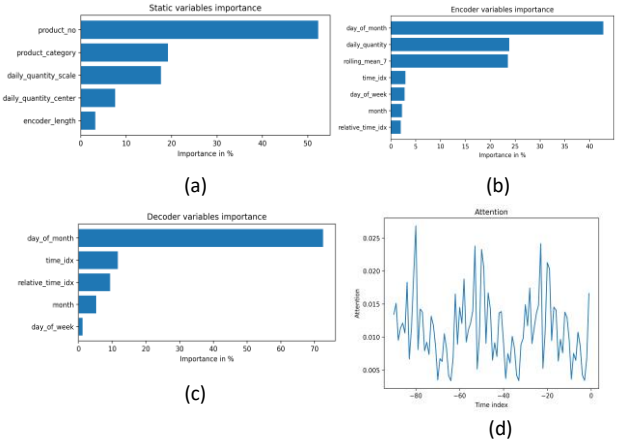

Figure 8. Interpretability analysis of the proposed TFT model: (a) static variable importance, (b) encoder variable importance, (c) decoder variable importance, and (d) temporal attention pattern over the historical input window.

Temporal Attention Analysis

Figure 8(d) further illustrates the learned temporal attention pattern over the historical input window. The attention distribution is clearly non-uniform, exhibiting multiple peaks across historical lags rather than collapsing exclusively on the most recent observations. This behavior indicates that the TFT model does not operate merely as a short-memory predictor. Instead, it selectively retrieves information from multiple historical intervals when generating future forecasts, thereby capturing temporally distributed dependencies over the encoder horizon.

This empirical behavior closely aligns with the architectural rationale described in the Methodology section and already explains how TFT integrates static covariates, known future inputs, and observed historical variables through variable selection networks, recurrent sequence modeling, and interpretable attention. Accordingly, the attention pattern serves as direct evidence that the model is not only theoretically capable of learning long-range temporal interactions but is also empirically observed to do so in the present pharmacy forecasting task.

From an application perspective, such a result is also highly plausible because pharmacy demand is often influenced simultaneously by immediate recent sales momentum, recurrent short-cycle purchasing behavior,

and broader temporal rhythms. In other words, historical information relevant to future demand is not concentrated in a single recent time step but is distributed across multiple prior intervals. Accordingly, the temporal attention analysis further substantiates the internal validity of the proposed architecture and helps explain why TFT is better aligned with the intrinsic characteristics of heterogeneous pharmacy sales forecasting than conventional machine learning baselines and standard LSTM models.

MAE–RMSE Trade-off and Product-Level Error Heterogeneity

Figure 9 complements the aggregate forecasting metrics by examining the MAE–RMSE trade-off at the product level. Most products are clustered within a relatively low-to-moderate error region, indicating that the proposed TFT model maintains stable predictive behavior for the majority of top-selling products. This finding is consistent with the product-level results previously reported in Table 3, where several products exhibit only very small deviations between actual and predicted totals. By contrast, a limited number of products appear as high-error outliers, suggesting that product-specific irregularities remain for certain items.

This pattern is analytically important because MAE and RMSE capture different aspects of forecasting quality. MAE reflects the average absolute deviation, whereas RMSE places greater emphasis on large prediction errors. Accordingly, products located close to the main diagonal pattern indicate relatively consistent error behavior, while products with disproportionately large RMSE suggest occasional sharp misses or abrupt demand variations. This interpretation is consistent with the current test-set results, where some products, such as 2003039, 2004927, and 2003851, show very small deviations, whereas others, such as 2014263 and 2002137, exhibit relatively larger gaps between actual and predicted totals.

Such outliers may reflect intermittent demand, promotion-driven sales spikes, stock-out effects, or other exogenous factors not explicitly encoded in the current feature set. Therefore, the MAE–RMSE trade-off not only complements the overall evaluation metrics but also provides a diagnostically useful view of where the current forecasting framework is robust and where additional feature augmentation may still be beneficial. This point is particularly meaningful because the current manuscript already demonstrates that TFT achieves the strongest overall predictive performance among all benchmark models, with MAE = 104.165, RMSE = 149.717, and R^2 = 0.986, yet the product-level trade-off analysis reveals that some residual heterogeneity remains beneath the aggregate performance summary.

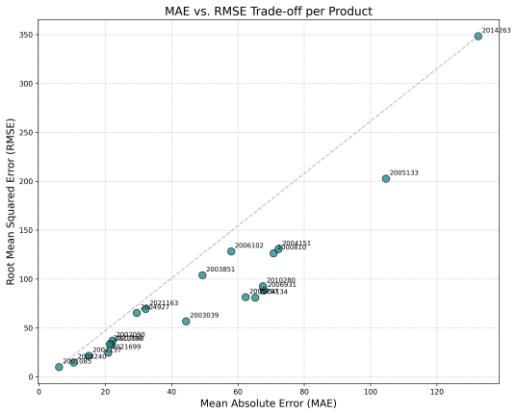

Figure 9. Product-level MAE–RMSE trade-off of the TFT model on the unseen January 2026 test set.

Interpretation Summary and Architectural Validity

Taken together, the combined evidence from variable importance, temporal attention, and product-level error trade-off substantiates the validity of the proposed TFT architecture from three complementary perspectives. First, the static feature analysis confirms that product-level metadata meaningfully conditions the forecasting process. Second, the encoder–decoder importance results verify that the model differentiates between observed historical variables and known future temporal cues in a functionally coherent way. Third, the temporal attention pattern demonstrates that the model does not depend solely on the most recent data point but selectively integrates information across multiple historical intervals.

Therefore, the added interpretability analysis strengthens the manuscript in two ways. First, it provides technical evidence for the internal validity of the proposed architecture. Second, it enhances the practical credibility of forecasting results for inventory planning, replenishment scheduling, and pharmacy retail decision support. Under this interpretation, the proposed TFT framework should be viewed not only as a high-performing predictive model but also as an interpretable analytical framework that supports more transparent demand forecasting in heterogeneous pharmacy retail environments.

Table 5 summarizes the main interpretability findings of the proposed TFT model and their corresponding technical and operational implications for pharmacy forecasting and inventory-related decision support.

Table 5. Summary of interpretability findings and their implications for pharmacy forecasting and inventory management.

Analytical component	Main finding	Technical implication	Operational implication
Static variable importance	**product_no** contributes the largest static importance, followed by **product_category** variables.	The model meaningfully uses product-level metadata to condition downstream temporal representation learning instead of treating all products uniformly.	Different products exhibit distinct demand behaviors; therefore, product-specific replenishment and inventory policies are justified.
Encoder variable importance	**day_of_month**, **daily_quantity**, and ***rolling_mean_7*** are the most influential encoder variables.	Historical encoding is mainly driven by intra-month calendar position, recent raw sales dynamics, and short-term smoothed demand momentum.	The model captures practical short-term purchasing patterns that are relevant to retail operations and near-term replenishment planning.
Decoder variable importance	**day_of_month** dominates the known future inputs, while **time_idx, relative_time_idx** provide secondary contributions.	Multi-step forecasting depends strongly on deterministic calendar structure, especially intra-month temporal position.	Monthly purchasing rhythm and refill timing are important drivers of pharmacy demand planning.
Temporal attention pattern	Attention is distributed across multiple historical lags rather than concentrated only on the most recent time step.	The model captures selectively distributed long-range temporal dependencies over the encoder horizon.	Forecasting decisions reflect more than immediate recent sales, improving robustness for replenishment scheduling and inventory control.
MAE–RMSE trade-off	Most products show relatively stable error behavior, whereas a limited number of products remain high-error outliers.	The forecasting framework is broadly robust, but some product-specific irregularities are not fully captured by the current feature set.	Additional features, such as promotions, stock-out indicators, or holiday effects, may further improve forecasting performance for difficult products.

Discussion

The results of this study indicate that the superior performance of the proposed TFT framework can be interpreted from three complementary perspectives: data structure compatibility, temporal dependency modeling, and interpretability-oriented decision support. First, the pharmacy forecasting problem addressed in this study is inherently heterogeneous because it simultaneously involves static product identifiers, deterministic calendar variables, and time-varying historical sales observations. Under such conditions, TFT is structurally more appropriate than conventional regression-based and tree-based baselines, because it was specifically designed to integrate static covariates, known future inputs, and observed historical variables within a unified multi-horizon forecasting architecture.

Second, the strong empirical performance of TFT is closely related to its ability to combine local sequential continuity with selective learning of long-range dependencies. In the present study, the encoder–decoder design preserves short-term demand momentum and recent sales dynamics, while the temporal attention mechanism allows the model to identify which historical intervals are most relevant for future prediction. This combination is particularly advantageous in pharmacy retail environments, where demand may be shaped simultaneously by recent purchase behavior, short replenishment cycles, and broader temporal rhythms. The results of the interpretability analysis further support this interpretation by showing that the model assigns distinct importance to static variables, historical observations, and known future temporal inputs, rather than processing all covariates uniformly.

Third, the present findings suggest that TFT should not be viewed only as a high-accuracy predictive model, but also as a practically informative analytical framework. The feature-importance analysis indicates that product-level identity and calendar-related variables play central roles in forecasting, while the temporal attention analysis shows that relevant predictive information is distributed across multiple historical lags rather than concentrated in the most recent observations. In addition, the product-level MAE–RMSE trade-off analysis reveals that most high-volume products are forecast with relatively stable error behavior, although a limited number still exhibit larger deviations. This means that the proposed framework is broadly robust, while also revealing where additional feature augmentation may further improve forecasting reliability.

Conclusion

The principal contribution of this study does not lie in reintroducing TFT itself as a novel algorithm, since TFT has already been established in the forecasting literature. Rather, the originality of the present work lies in demonstrating how TFT can be rigorously positioned, validated, and interpreted in a real pharmacy-chain forecasting context. Specifically, this study contributes in three ways. First, it applies TFT to real pharmacy-chain transactional data rather than to synthetic or generic benchmark series. Second, it constructs a consistent retail forecasting pipeline that converts irregular transaction-

level records into structured daily sequences suitable for multi-horizon prediction. Third, it evaluates TFT against Linear Regression, Random Forest, XGBoost, and standard LSTM under the same data split, feature engineering process, forecasting horizon, and evaluation criteria. Therefore, the study provides a practically grounded comparison that clarifies why TFT is especially suitable for heterogeneous pharmacy retail forecasting problems.

The interpretability analysis further strengthens this conclusion. The variable-importance results show that the model meaningfully distinguishes static product-level metadata, observed historical variables, and known future calendar inputs, while the temporal attention analysis shows that relevant predictive information is distributed across multiple historical intervals rather than concentrated exclusively in the most recent observations. In addition, the product-level MAE–RMSE trade-off analysis indicates that the proposed framework is broadly robust across most high-demand products, although a limited number of items remain more difficult to forecast. Taken together, these findings show that the advantage of TFT in the present study is not merely reflected in lower aggregate forecasting errors, but also in its interpretable internal learning behavior and its closer structural alignment with the intrinsic characteristics of pharmacy demand data.

Acknowledgment

Research supported by the National Science and Technology Council under Grant NSTC 114-2221-E-027-104.

References

[1] B. Lim, S. Ö. Arık, N. Loeff, and T. Pfister, “Temporal Fusion Transformers for interpretable multi-horizon time series forecasting,” International Journal of Forecasting, vol. 37, no. 4, pp. 1748–1764, 2021.

[2] S. K. Lee and M.-J. Rah, “Deep learning in retail supply chain management: an evolution,” World Journal of Economics and Business Research, vol. 2, no. 2, pp. 43–47, 2024.

[3] S. Hartanto and A. A. S. Gunawan, “Temporal Fusion Transformers for Enhanced Multivariate Time Series Forecasting of Indonesian Stock Prices,” International Journal of Advanced Computer Science and Applications, vol. 15, no. 7, 2024.

[4] K. Jeaab, Y. Saoudi, and M. E. M. Falloul, “A comparison of LSTM, GRU, and XGBoost for forecasting Morocco’s yield curve,” Mathematical Modeling and Computing, vol. 11, no. 3, pp. 674–681, 2024, doi: 10.23939/mmc2024.03.674.

[5] P. Yadav, “Demand forecasting in retail using machine learning and big data,” International Journal of Advanced Research in Computer and Communication Engineering, vol. 14, no. 2, 2025, doi: 10.17148/IJARCCE.2025.14235.

[6] P. Niu, T. Zhou, X. Wang, L. Sun, and R. Jin, “Attention as robust representation for time series forecasting,” arXiv preprint arXiv:2402.05370, 2024.

[7] A. Vaswani et al., “Attention is all you need,” in Advances in Neural Information Processing Systems (NeurIPS), vol. 30, 2017.

[8] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[9] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 785–794.

[10] L. Breiman, “Random forests,” Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.

[11] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in Proceedings of the International Conference on Learning Representations (ICLR), 2015.

[12] B. N. Oreshkin, D. Carpov, N. Chapados, and Y. Bengio, “N-BEATS: Neural basis expansion analysis for interpretable time series forecasting,” in Proceedings of the International Conference on Learning Representations (ICLR), 2020.

[13] D. Salinas, V. Flunkert, J. Gasthaus, and T. Januschowski, “DeepAR: Probabilistic forecasting with autoregressive recurrent networks,” International Journal of Forecasting, vol. 36, no. 3, pp. 1181–1191, 2020.

[14] H. Wu, J. Xu, J. Wang, and M. Long, “Autoformer: Decomposition transformers for long-term series forecasting,” in Advances in Neural Information Processing Systems (NeurIPS), vol. 34, 2021, pp. 22419–22430.

[15] N. Kitaev, L. Kaiser, and A. Levskaya, “Reformer: The efficient transformer,” in Proceedings of the International Conference on Learning Representations (ICLR), 2020.

[16] Y. R. Sagaert, E.-H. Aghezzaf, N. Kourentzes, and B. Desmet, “Temporal Big Data for Tactical Sales Forecasting in the Tire Industry,” Interfaces, vol. 48, no. 2, pp. 121–129, 2018.

[17] Q. Wen, T. Zhou, C. Zhang, W. Chen, Z. Ma, J. Yan, and L. Sun, “Transformers in time series: A survey,” in Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence (IJCAI-23), 2023, pp. 6778–6786.

[18] K. Benidis et al., “Deep learning for time series forecasting: Tutorial and literature survey,” ACM Computing Surveys, vol. 55, no. 6, pp. 1–36, 2022.

[19] I. Rojat et al., “Explainable artificial intelligence (XAI)

on time series data: A survey," arXiv preprint arXiv:2104.00950, 2021.
[20] F. Petropoulos et al., "Forecasting: theory and practice," International Journal of Forecasting, vol. 38, no. 3, pp. 705–871, 2022.
[21] A. Zeng, M. Chen, L. Zhang, and Q. Xu, "Are transformers effective for time series forecasting?," in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, no. 10, pp. 11121–11128, 2023.
[22] T. Makridakis, E. Spiliotis, and V. Assimakopoulos, "The M4 Competition: 100,000 time series and 61 forecasting methods," International Journal of Forecasting, vol. 36, no. 1, pp. 54–74, 2020.
[23] Y. Zhang and J. Yan, "Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting," in Proceedings of the International Conference on Learning Representations (ICLR), 2023.
[24] R. J. Hyndman and G. Athanasopoulos, Forecasting: Principles and Practice, 3rd ed. Melbourne, Australia: OTexts, 2021.