IJAAF; Vol. 10, No. 2, Spring 2026, Serial 35

Ferdowsi University of Mashhad

Iranian Journal of
Accounting, Auditing & Finance

Quarterly

RESEARCH ARTICLE

Predicting Stock Market Returns Using Temporal Fusion Transformer: A Comprehensive Data-Driven Approach

Somayeh Mohebi*
School of Industrial Engineering and Management, Shahrood University of Technology, Shahrood, Iran
Mohammad Reza Mohebbi
Department of Computer Science, University of Passau, Passau, Germany
Javad Mohebbi Najm Abad
Department of Computer Engineering, Imam Reza International University, Mashhad, Iran

How to cite this article:
Mohebi, S., Mohebi, M. R. and Mohebi Najm Abad, J. (2026). Predicting Stock Market Returns Using Temporal Fusion Transformer: A Comprehensive Data-Driven Approach. Iranian Journal of Accounting, Auditing and Finance, 10(2), 125-151. doi: 10.22067/ijaaf.2026.47513.1562
https://ijaaf.um.ac.ir/article_47513.html

ARTICLE INFO

Article History
Received: 2025-06-14
Accepted: 2025-12-06
Published online: 2026-04-12

Keywords:
Deep Learning, Dimensionality Reduction, Feature Selection, Stock Market Prediction, Temporal Fusion Transformer

Abstract

Stock market return prediction remains a highly challenging task due to the complex, dynamic, and noisy nature of financial markets. Although machine learning and deep learning models have been widely explored, many approaches struggle to capture long-term temporal dependencies and provide interpretable feature selection. In tackling these issues, the Temporal Fusion Transformer (TFT) is applied for multi-horizon time series forecasting of daily returns in the Tehran Stock Exchange (TSE). Accordingly, the study proposes a new feature selection method called Initial Selected Features–Mutual Information Difference (ISF-MID), which enhances the traditional Minimum Redundancy Maximum Relevance (mRMR) algorithm by demonstrating a superior capability of handling redundancy and focusing on determining important parameters. Dual preprocessing works on ISF-MID, mRMR and PCA that improve the predictive performance, while also allowing for interpretable machine learning by working with a better representation of input. Mean Absolute Error (MAE), Mean Squared Error (MSE) and the coefficient of determination (R^2) were used to conduct extensive comparisons with benchmark methods such as Long Short-Term Memory LSTM, Multi-Layer Perceptron MLP, and Random Forest RF. Thus, the proposed method achieved competitive performance against benchmark architectures (i.e. TFT achieves R^2 of 98.9%, MAE 0.00043 and MSE 0.000018 on out-of-sample test, as well as high directional accuracy). Overall, the synergy of TFT with the ISF-MID feature selection strategy provides methodological novelty and practical significance, extending a potential framework for evidence-based return prediction and informed risk-central investment decisions.

https://doi.org/10.22067/ijaaf.2026.47513.1562

NUMBER OF REFERENCES
45

NUMBER OF FIGURES
10

NUMBER OF TABLES
7

Homepage: https://ijaaf.um.ac.ir
E-Issn: 2717-4131
P-Issn: 2588-6142

*Corresponding Author: Somayeh Mohebi
Email: mhd_moradi@um.ac.ir
Tel: 09153215094
ORCID: 0000-0003-2559-7440

1. Introduction

The prediction of market returns is probably the essential and extremely challenging issue in financial analysis, as markets are mostly dynamic, nonlinear and noisy—especially in real-world data for time series (Sharma & Mehta, 2024). Generating accurate forecasts helps with portfolio management, risk control, and investment strategy design. However, it is still challenging because of the complexity and high dimensionality of these influencing factors. Interactions among macroeconomic factors, industry dynamics, company performance, and investor behavior are difficult to capture with reliable models (Dos Santos et al., 2025). These difficulties have led to considerable research on predictive techniques of some kind to identify regularities in complex phenomena of financial data (Sharma and Mehta, 2024).

Conventional Machine Learning (ML) and Deep Learning (DL) methods have been extensively studied as part of financial forecasting. Support Vector Regression (SVR), Random Forest (RF), and recurrent neural networks such as Long Short-Term Memory (LSTM) have demonstrated excellent capability to model certain nonlinearities of financial time series despite essential drawbacks (Tulcanaza-Prieto and Lee, 2019). Recurrent structures are typically unable to describe long-term relationships, ensemble and kernel approaches are hard to interpret, and most of them are still vulnerable to irrelevant or redundant variables, which may impair prediction precision at high dimensionalities of datasets (Alam et al., 2024).

The latest progress of sequence modeling, such as transformer architecture, provides the potential to overcome these limitations. The Temporal Fusion Transformer (TFT) is arguably one of the best DL models introduced so far, designed for multi-horizon forecasting (Lim et al., 2021). The use of attention mechanisms allows the model itself to automatically assign flexible importance to input features over time, making it particularly efficient under conditions of temporal complexity and signal noise (Hartanto and Gunawan, 2024). Although other applications have flourished with transformers, the usage of the transformers applied to financial return prediction is limited. Specifically, limited studies have concentrated on the integration of TFT with systematic feature engineering that has been tailored based on the unique features of financial markets (Davoud Abadi, 2025).

Feature selection and dimension reduction are known to be important factors contributing to the performance of models in financial predictions. The models are more accurate and by eliminating redundancy, they are much more interpretable. Minimum Redundancy Maximum Relevance (mRMR) and Principal Component Analysis (PCA) are typical methods that are widely adopted to improve data quality before model training (Mohebi et al., 2022a). Although most of these methods are useful, they are usually used in isolation, and only a few studies have investigated hybrid pipelines, combining feature selection and feature extraction. Additionally, existing work has rarely evaluated the establishment of new feature selection methods that are explicitly designed for financial time series data.

This study proposes a forecasting framework, filling these gaps by introducing a powerful framework that predicts the daily returns of the Tehran Stock Exchange (TSE). Accordingly, the Initial Selected Features–Mutual Information Difference (ISF-MID) approach is proposed as a newly developed feature selection framework, which generalizes the mRMR method to enhance relevance and reduce redundancy. ISF-MID is used as a preprocessing technique to improve prediction accuracy and interpretability, with its performance also compared to that of PCA. The TFT is used as the main forecasting framework and compared with several ML and DL baselines, such as Multi-Layer Perceptron (MLP), Radial Basis Function (RBF), SVR, Decision Trees (DT), RF, Deep Neural Networks (DNN), and LSTM. The overall research pipeline is illustrated in Figure 1, covering dataset construction, preprocessing, feature selection and extraction, and forecasting with TFT and comparative baselines. The next sections describe each step in detail. The principal contributions of

this paper are as follows:

We developed a rich set of candidate features at scale by combining market indices, technical indicators and macroeconomic variables. Based on this, secondary features were produced that enhance the primary input by adding more dependencies to create a richer representation of the market behavior. We provided a detailed comparison of the performance of the TFT with popular classical ML and modern DL models for daily stock return forecasting in the TSE. We proposed a new feature selection approach termed "ISF-MID", which utilizes the mRMR principle of maximizing relevance while reducing redundancy, leading to compact and efficient input representation. We compared alternative preprocessing strategies, including an mRMR method and its two variants, MID and FCD, as well as a PCA strategy to assess their relative impact on predictive performance, establishing that the ISF-MID variant is a more effective extension of mRMR. We showed a TSE index prediction case study that targeted feature selection framework ranks and validates the most impactful features for predicting the TSE index. In using the proposed ISF-MID algorithm, this pipeline pushes methodological innovation whilst giving more interpretable financial forecasting.

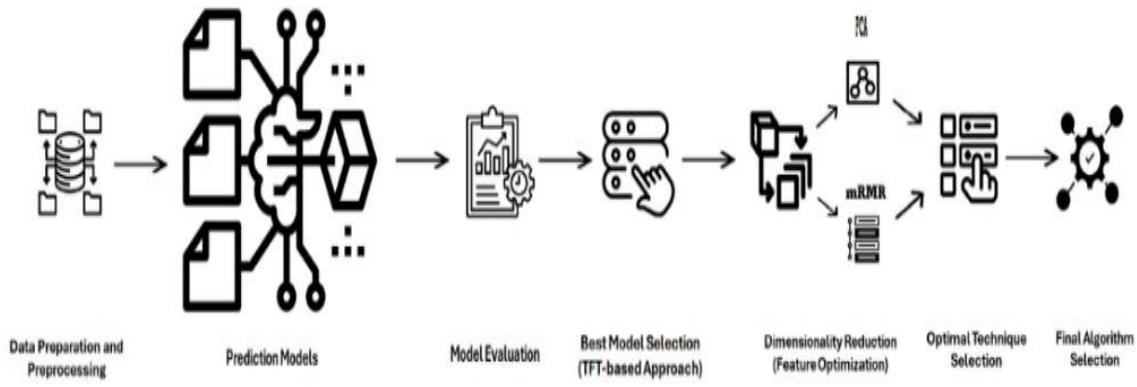

Figure 1. Proposed framework for stock index return prediction

2. Related work

This prediction of stock market returns is a very difficult problem because financial data is inherently dynamic, nonlinear and noisy (Zheng et al., 2024). Proper prediction assists investment decisions, risk management, and portfolio allocation. Reported literature contains numerous approaches from typical ML models to more complex DL, hybrid forms, RL and transformer-based methods (Li et al., 2020; Sezer et al., 2020). However, there still exist limitations in these studies regarding the ability to learn long-term dependencies, interpret feature importance and high dimensionality. To demonstrate the overview in a structured manner, we grouped related work into three categories: (i) ML and DL models; (ii) hybrid and RL approaches; and (iii) transformer-based frameworks focused on more recent TFT applications.

2.1. Machine learning and deep learning models

Traditional ML methods have been popular in stock market prediction and are also able to capture structured and high-dimensional financial data. As a result, SVR, DT, RF, KNN (k-Nearest

Neighbors) and GBM (Gradient Boosting Machines) are all well proven to have the ability to model short-term relationships and cross-sectional appearances in financial indicators (Kumbure et al., 2022). Feature engineering and dimensionality reduction are often key complements: e.g., Wei et al. (2021) found that PCA reduces the redundancies of these features, retaining important information in the market. We have also proven a useful ensemble-based model; the adaptation of AdaBoost with DTs is associated with increasing predictive precision and computational efficiency (Guo et al., 2017), while Zhang et al. (2018) reduced complexity without sacrificing accuracy using combined feature engineering with ensemble learning strategies. While some domains were successful, such methods tend to fall short of adequately learning the complex sequential and nonlinear structures present within financial time series.

Simultaneously, DL models have become a popular tool to handle temporal dependencies and nonlinear relationships in financial data. Their effectiveness, however, is limited mainly due to the presence of long-term dependencies, upon which LSTM networks and Gated Recurrent Units (GRU) were developed as recurrent architectures that are widely used. Nabipour et al. (2020) showed that LSTM outperformed classical ML algorithm classes for trend forecasting at the TSE consistently, while Wei et al. (2024) found the same thing. Unlike the recurrent models, other data representations, such as the use of bidirectional networks and autoencoder based architectures, have been proposed to capture complex feature interactions and latent structures (Gandhudi et al., 2024; Bao et al., 2017). Moreover, where inputs were often restricted to numbers, adding textual sentiment from news and social networks transformed the forecast in DL frames. This was accomplished by combining LSTM or MLP networks with natural language processing modules (Awad et al., 2023).

Yet the ML and DL approaches suffer from significant caveats. Traditional ML models usually do not consider temporal dynamics. Although DL architectures are more powerful but are subject to overfitting, need high computation resources, and they also behave like “black boxes” as they have less interpretability. Additionally, the two model classes remain fundamentally sensitive to redundant or irrelevant variables, a very relevant issue in high-dimensional financial data sets. These limitations underscore the need for systematic feature selection and/or dimensionality reduction to improve predictive robustness as well as interpretability. As a result, more recent work has focused on sophisticated architectures, and in particular transformer-based models, that are not only able to capture long-term dependencies but can also be enhanced with principled feature engineering strategies.

2.2. Hybrid and reinforcement learning approaches

A number of research-based frameworks have been proposed that integrate the complementary paradigms of stand-alone ML and DL models to mitigate their individual drawbacks, thus improving forecasting performance. These methods generally combine statistical preprocessing and deep sequence models or optimization techniques with non-linear predictors. For example, Wu et al. included the work of Yi and Shen (2023), who queried historical stock data using Triple Q-learning in conjunction with the Growing Neural Gas (GNG) algorithm. John and Latha (2023) proposed a Hybrid Recurrent Neural Network (HyRNN), which incorporated financial news sentiment into BiLSTM and GRU. Feature decomposition methods have also been widely applied. Liang et al. (2019) adopted Wavelet Transform (WT) and LSTM for the separation of informative frequency components, whereas Niu et al. (2020) utilized VMD, presenting more stable prediction results. Hybridization has also found its way into optimization-based approaches, with an instant example being the Bayesian optimization carcass algorithm (Liwei et al., 2021). SVR mollusk with Particle Swarm Optimization (PSO)-based hyperlloid-adding model (Jaiwang and Jeatrakul, 2018) allows for on-the-fly hyperparameter changes appropriate for ever-fluctuating market environments.

RL is this brand new, promising direction for the entire world of financial forecasting and algorithmic trading that goes beyond Hybrids. Deep Q-Networks (DQN) and variants with attention features and sentiment features have shown a lot of achievement in navigating SEO Bid environments that are dynamic and non-deterministic (Awad et al., 2023). Recent works leverage multimodal signals in RL to train models on numerical and linguistic data streams together for improved strategy learning. Concurrently, federated learning has proposed frameworks to improve privacy and mitigate data diversity issues. Pourroostaei Ardakani et al. (2023) showed RF and SVM models improved by federated learning to protect privacy across institutions, while Kumarappan et al. (2024) introduced LSTM models integrated with federated learning, balancing superior predictive effectiveness with stringent privacy safeguards. These approaches highlight the adaptability of RL and federated settings to practical financial environments where both robustness and security are critical.

Although hybrid RL and federated approaches represent important advances, they also introduce added architectural complexity and often shift the focus to trading strategy optimization rather than systematic feature engineering. In their dependence upon black box deep structures, they sacrifice interpretability and often end up with redundant or irrelevant features in the input space, undermining robustness. Such gaps drive the investigation of models, which can maintain predictive power, interpretable features and high feature efficiency. The current work bridges this gap by integrating the interpretability and temporal modeling capabilities of TFT with state-of-the-art feature selection and dimensionality reduction techniques to deliver accurate yet interpretable financial forecasts.

2.3. Transformer-based models for financial forecasting

Recently, transformer architectures have been applied in financial forecasting due to their capabilities of capturing long-range dependencies and adaptively weighting relevant signals. To make them work for high-frequency trading data, sparse or decomposition-based attention has been used (Wu, 2023) in informer and outformer. They proposed transformer layers were adapted to multivariate financial series, resulting in improvements over recurrent models (Wang et al., 2022). Emami Gohari et al. (2024) proposed a multimodal transformer architecture that jointly processes numeric time-series and textual inputs (e.g., economic reports) through intra-, inter- and target-modal attention layers, demonstrating improved performance over baseline models in a financial forecasting setting. In addition to numeric features, other models like FinSentiBERT and multimodal transformers demonstrated the ability of textual sentiment (in particular from headlines/analyst reports) to extract predictive signals (Sun et al., 2025). The results of these studies support the acclaimed usefulness of the transformer structure in the main financial features.

More recently, researchers have applied the TFT specifically to financial markets. Hartanto and Gunawan (2024) applied TFT to Indonesian stock prices and found gains over baseline transformer variants, as well as naïve methods in volatility-sensitive settings (consult Hartanto and Gunawan (2024) for datasets, baselines, and metrics). Davoud Abadi (2025) conducted a comparison between TFT and LSTM, SMA combined with SVR on both the S&P 500 and Tehran indices, and reported that for the Tehran index, TFT outperformed the results. Yang et al. (2025) moreover extended TFT into a multi-sensor framework for Sharpe Ratio (SR=expected return/risk) optimization in what is called here as the Tuning Framework Transfer for ASset Risk Optimization (or TFT-ASRO), gaining an 18% risk-adjusted forecasting performance. Dashtaki et al. (2025) developed a cross-modal temporal fusion framework that integrates textual and numerical features, leading to enhanced inference accuracy and efficiency. Lim et al. (2021) discovered that TFT is interpretable in forecasting tasks and validated the transparency of TFT compared to other black-box models.

Although TFT has shown great promise for financial forecasting, existing works typically use raw or less engineered inputs like prices, volumes and sentiment embeddings, which opens the models to

redundant features, noise, and has lower robustness in high-dimensional settings. To the best of our knowledge, no prior work systematically integrates TFT with a feature optimization framework in which selection methods (mRMR, ISF-MID) and extraction techniques (PCA) are systematically evaluated as alternative preprocessing strategies. This study provides a novel approach to overcome this by proposing ISF-MID for redundancy-sensitive feature selection and investigating its improvement on both TFT performance measures, predictive accuracy and interpretability with respect to stock index prediction.

3. Methodological background

This part briefly outlines the theoretical underpinnings deployed in this study. We first introduce the Temporal Fusion Transformer (TFT), a state-of-the-art interpretable multi-horizon forecasting model that can utilize both temporal and static covariates. By focusing on the proposed ISF-MID and well-established mRMR and PCA techniques, we then discuss feature selection and dimensionality reduction methods suited for high-dimensional, correlated financial inputs. In combination, these components form the foundation of our forecasting framework.

3.1. Temporal fusion transformer (TFT)

The TFT is a deep learning model, which focuses on interpretable multi-horizon time series forecasting (Lim et al., 2021) and generalizes well with heterogenous datasets containing temporal and static covariates, thus allowing for financial settings in which the underlying dynamics of any market vary according to colliding technical and macroeconomic factors (Lim et al., 2021).

3.1.1. Input representation and embedding

Temporal covariates $\mathrm{x}_t \in \mathbb{R}^{d_t}$ at time t and static covariates $\mathrm{x}^{\text{static}} \in \mathbb{R}^{d_s}$ are mapped to dense embeddings via learnable functions $f_{\text{embed}}^{(t)}(\cdot)$ and $f_{\text{embed}}^{(s)}(\cdot)$:

$$\mathrm{z}_t = f_{\text{embed}}^{(t)}(\mathrm{x}_t), \quad \mathrm{z}^{\text{static}} = f_{\text{embed}}^{(s)}\left(\mathrm{x}^{\text{static}}\right), \tag{1}$$

where $\mathrm{z}_t \in \mathbb{R}^{p_t}$ and $\mathrm{z}^{\text{static}} \in \mathbb{R}^{p_s}$ are latent representations. In financial datasets, this places indicators (e.g., moving averages, exchange rates) in a consistent latent space.

3.1.2. Variable selection networks (VSNs)

VSNs select informative variables at each step [3]. Let $\mathrm{x}_t \in \mathbb{R}^{d_t}$ be the (possibly concatenated) temporal features at time t. The network produces a *weight vector* $\mathrm{w}_t \in \mathbb{R}^{d_t}$: and the bias vector b:

$$\mathrm{w}_t = \mathrm{softmax}(\mathrm{W}\mathrm{x}_t + \mathrm{b}), \tag{2}$$

where $\mathrm{W} \in \mathbb{R}^{d_t \times d_t}$ and $\mathrm{b} \in \mathbb{R}^{d_t}$ are learnable parameters. The i-th entry $[\mathrm{w}_t]_i$ indicates the importance of variable i at time t, improving interpretability by ranking indicators (e.g., volume, oil price).

3.1.3. Gating and residual connections

Gating mechanisms regulate information flow and stabilize training (Hartanto and Gunawan, 2024):

$$\mathrm{h}_t = f_{\text{gate}}\left(\mathrm{z}_t, \mathrm{z}^{\text{static}}\right), \tag{3}$$

where z_t is the main input representation at time step t. Also, z^{static} is time-invariant information. Finally, $f_{gate}(0)$ denotes a learned gating function. Gates suppress irrelevant/noisy inputs, while residual connections maintain gradient flow.

3.1.4. Multi-head self-attention

Long-range temporal dependencies are modeled via multi-head self-attention (Lim et al., 2021):

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^{\top}}{\sqrt{d_k}}\right)V, \quad (4)$$

with query Q, key K, value V, and key dimension d_k. This uncovers lagged effects (e.g., macroeconomic shocks).

3.1.5. Decoder and forecast generation

Context vectors are decoded to produce multi-horizon forecasts. TFT yields both accurate predictions and post-hoc interpretability via attention and variable selection, attributing importance to covariates and temporal patterns (Lim et al., 2021).

Transition.

Given the high dimensionality and correlation structure of financial inputs, we next outline feature selection and dimensionality reduction methods used to construct compact, informative representations.

3.2. Feature selection and dimensionality reduction

Financial datasets often include redundant variables that can impair generalization. We therefore consider ISF-MID and mRMR (feature selection) and PCA (feature extraction) to improve efficiency and interpretability.

3.2.1. Initial selected features–mutual information difference

Initial Selected Features–Mutual Information Difference (ISF-MID) extends the mRMR-MID criterion by seeding the selection with a domain-informed initial set S_0 and then iteratively adding features from the candidate pool F (the full set of candidate features) based on a redundancy-aware score (Mohebi et al., 2022b). Let S denote the current selected subset (initialized as S_0), y the target variable, and $I(0;0)$ mutual information. At each step, f^*, which is the next feature to select, is calculated by the following equation:

$$f^* = \arg\max_{f \in F \setminus S} \left[I(f;y) \; - \; \alpha \cdot \frac{1}{|S|} \sum_{s \in S} I(f;s) \right], \quad (5)$$

where $\alpha \geq 0$ controls redundancy penalization. $F \setminus S$ are the remaining features, which are not yet selected. Seeding with S_0 injects expert knowledge, the MI term favors relevance, and the penalty discourages selecting near-duplicate indicators (e.g., overlapping moving averages).

In contrast to ISF-MID, which emphasizes redundancy-sensitive feature selection, the traditional mRMR method has been widely applied as a baseline for balancing feature relevance and redundancy.

3.2.2. Minimum redundancy maximum relevance (mRMR)

According to Mohebbi et al. (2025(a)), the mRMR goal strikes a balance between y-relevant information and pairwise redundancy within S:

$$\max_{S\subseteq F,\,|S|=k}\left[\frac{1}{|S|}\sum_{f\in S} I\,(f;y)\;-\;\frac{1}{|S|^2}\sum_{f_1,f_2\in S} I\,(f_1;f_2)\right]. \qquad (6)$$

Here, F is the full feature set and k is the desired subset size. The result is a predictive yet diverse set of variables.

Dimensionality reduction techniques, such as principal component analysis (PCA), offer an alternative to feature selection methods like ISF-MID and mRMR by converting features into orthogonal components, which preserve variance while minimizing redundancy.

3.2.3. Principal component analysis (PCA)

Orthogonal components that account for the most variance are projected by PCA from correlated inputs (Pourroostaei Ardakani et al., 2023). Let Σ be the covariance of centered data, with eigenpairs $(\lambda_i, \mathrm{v}_i)$:

$$\Sigma \mathrm{v}_i = \lambda_i \mathrm{v}_i, \qquad \mathrm{z} = \mathrm{V}^{\top}(\mathrm{x} - \bar{\mathrm{x}}), \qquad (7)$$

where $\mathrm{V} = [\mathrm{v}_1, \ldots, \mathrm{v}_r]$ collects the top r eigenvectors, x is an input vector, $\bar{\mathrm{x}}$ its mean, and z is the r-dimensional representation.

Summary.

TFT provides a flexible temporal modeling backbone, while ISF-MID (with mRMR and PCA as comparators) yields compact, informative inputs by mitigating redundancy and emphasizing relevant signals. This methodological foundation motivates the dataset construction and experimental design in the next sections.

3.3. Methodological contributions

This study follows a methodological design with several key contributions that improve the predictive performance and interpretability:

- **ISF-MID**: The Initial Selected Features–Mutual Information Difference (ISF-MID) as a redundancy-sensitive feature selection strategy adapted to the nature of financial time series data.
- **TFT Integration**: To make the best use of temporal and static covariates, ISF-MID is integrated into a TFT-based forecasting framework.
- **Comparative Benchmarking**: The introduced framework is systematically benchmarked with a range of classical ML (e.g., SVR, RF, DT) and advanced DL (e.g., LSTM, DNN)-based models along with alternative preprocessing methods (mRMR, PCA).
- **Assessment through Multiple Metrics**: A diverse range of evaluation metrics is used, from MAE, MSE and RMSE to MAPE, R^2, DA and SR, enabling a thorough examination of the model performance.
- **Improved Interpretability**: With integrated variable selection, dimensionality reduction, and attentional mechanisms in place, our framework offers transparent insights regarding the comparative significance of input features that influence performance as well as dynamic

transformations learned from market interactions.

3.4. Dataset specification

3.4.1. Dataset description

The empirical analysis uses 1,108 consecutive trading days of daily observations from the TSE. The dataset includes technical, market, and macroeconomic indicators that jointly characterize the Iranian equity market. The predictive task is to estimate the next-day return of the Tehran Exchange Dividend and Price Index (TEDPIX), serving as the target variable.

3.4.2. Feature overview

To emphasize the complementary roles of predictor variables in return prediction, we classify predictors into three categories:

- ***Market Indices***: Historical Returns of TEDPIX, the Price Index of 50 More Active Companies (PIC50), and the 30 Largest Companies Index (LCI30), which would represent market-wide momentum and capture sectoral trends.
- ***Technical Indicators***: Trend and volume indicators such as Trading Volume (TV), Moving Averages (MA), Exponential Moving Averages (EMA), Relative Difference in Percentage (RDP) and others, which are extensively used in short-horizon forecasting.
- ***Macroeconomic Variables****:* External drivers, including OPEC oil prices (OP), USD/IRR (DP) and EUR/IRR (EP) exchange rates, and gold coin prices (GP), reflecting macro and commodity influences.

3.4.3. Tools and preprocessing environment

Feature construction follows Zhong and Enke (2017), which is implemented in R (4.3.2) and Excel. Preprocessing and analysis were performed in MATLAB (R2024b) and Python (3.12.1) (Table 1).

This diverse set of features provides a comprehensive representation of internal market dynamics and external shocks, which also introduces redundancy and collinearity, motivating the feature reduction strategies (ISF-MID, mRMR, PCA) described earlier.

3.4.4. Data preparation

3.4.4.1. Normalization

Continuous variables are rescaled using Min–Max normalization to [0,1]:

$$v_i^{(\mathrm{norm})} = \frac{v_i - \min(V)}{\max(V) - \min(V)}. \qquad (8)$$

Definitions: v_i is the i-th observation of a given variable V; min(V) and max(V) denote the minimum and maximum over the training set for V, which prevents variables with large magnitudes (e.g., exchange rates) from dominating others (e.g., relative indices).

Table 1. Features included in the dataset with descriptions and calculation methods

Feature	Description	Calculation Method
Tehran Exchange Dividend and Price Index (TEDPIX)	Return on the current day	(p(t)-p(t-1))/p(t-1)
	Return on day t-1	(p(t-1)-p(t-2))/p(t-2)
	Return on day t-2	(p(t-2)-p(t-3))/p(t-3)
	Return on day t-3	(p(t-3)-p(t-4))/p(t-4)
Trading Volume (TV)	Relative change on day t-1	(p(t-1)-p(t-2))/p(t-2)
	Relative change on day t-2	(p(t-2)-p(t-3))/p(t-3)
	Relative change on day t-3	(p(t-3)-p(t-4))/p(t-4)
Index of 50 More Active Companies (PIC50)	Return on day t-1	(p(t-1)-p(t-2))/p(t-2)
	Return on day t-2	(p(t-2)-p(t-3))/p(t-3)
	Return on day t-3	(p(t-3)-p(t-4))/p(t-4)
Index of 30 large companies (LCI30)	Return on day t-1	(p(t-1)-p(t-2))/p(t-2)
	Return on day t-2	(p(t-2)-p(t-3))/p(t-3)
	Return on day t-3	(p(t-3)-p(t-4))/p(t-4)
Exponential Moving Average (EMA)	Four distinct EMAs of closing prices: 10, 20, 50, and 200 days	p(t)×(2/(10+1))+EMA10(t-1)×(1-2/(10+1)) p(t)×(2/(20+1))+EMA20(t-1)×(1-2/(20+1)) p(t)×(2/(50+1))+EMA50(t-1)×(1-2/(50+1)) p(t)×(2/(200+1))+EMA200(t-1)×(1-2/(200+1))
Moving Average (MA)	Three distinct MAs: 5, 10, and 30 days	$MA_{t-1} = \frac{\sum_1^n p_{t-1}}{n}$
Relative Difference Percentage (RDP)	5-day relative difference (%)	(p(t)-p(t-5))/p(t-5)×100
	10-day relative difference (%)	(p(t)-p(t-10))/p(t-10)×100
	15-day relative difference (%)	(p(t)-p(t-15))/p(t-15)×100
	20-day relative difference (%)	(p(t)-p(t-20))/p(t-20)×100
Support/Resistance Index (MPP)	levels at 30 and 120 days	$MPP = \frac{p - p_{min}}{p_{max} - p_{min}}$
OPEC Oil Prices (OP)	Price on the current day and the past three days and return on day t-1	(p(t-1)-p(t-2))/p(t-2)
	Return on day t-2	(p(t-2)-p(t-3))/p(t-3)
	Return on day t-3	(p(t-3)-p(t-4))/p(t-4)
USD/IRR Exchange Rate (DP)	Price on the current day and the past three days and return on day t-1	P(t), P(t-1), P(t-2), P(t-3)
	Return on day t-2	(p(t-1)-p(t-2))/p(t-2)
	Return on day t-3	(p(t-2)-p(t-3))/p(t-3)
		(p(t-3)-p(t-4))/p(t-4)
EUR/IRR Exchange Rate (EP)	Price on the current day and the past three days and return on day t-1	P(t), P(t-1), P(t-2), P(t-3)
	Return on day t-2	(p(t-1)-p(t-2))/p(t-2)
	Return on day t-3	(p(t-2)-p(t-3))/p(t-3)
		(p(t-3)-p(t-4))/p(t-4)
Gold Coin Price (GP)	Price on the current day and the past three days and return on day t-1	P(t), P(t-1), P(t-2), P(t-3)
	Return on day t-2	(p(t-1)-p(t-2))/p(t-2)
	Return on day t-3	(p(t-2)-p(t-3))/p(t-3)
		(p(t-3)-p(t-4))/p(t-4)

3.4.4.2. Temporal windowing

To model sequential dependencies, the input samples are reshaped into sliding windows of length $L = 5$. Let d be the number of predictors and N the number of days; the resulting tensor has shape $(N - L + 1) \times L \times d$, enabling the TFT to capture short-term market memory.

3.4.4.3. Handling missing data

Missing values are imputed by forward fill, preserving temporal continuity without introducing synthetic noise, appropriate in financial time series, where the most recent value often carries a

predictive signal.

3.4.4.4. Rolling window splitting

To mimic realistic trading conditions, a rolling (forward-expanding) split is used. Let $X_t \in \mathbb{R}^d$ denote the predictor vector at day t and $y_t \in \mathbb{R}$ the target return variable. For a split index t and horizon K,

$$\text{Training: } \{(X_1, y_1), \dots, (X_t, y_t)\}, \text{Testing: } \{(X_{t+1}, y_{t+1}), \dots, (X_{t+K}, y_{t+K})\}.$$

Evaluation is thus conducted on unseen, forward-looking data, avoiding information leakage under non-stationarity.

3.5. Methodology

This section presents the methodological design of the study, beginning with the proposed forecasting framework, followed by the baseline models, the hyperparameter tuning strategy, and finally the evaluation metrics. The structure ensures transparency, reproducibility, and alignment with best practices in financial forecasting.

3.5.1. Proposed framework

The proposed forecasting framework integrates ISF-MID and TFT. ISF-MID reduces the dimensionality of features by iteratively identifying predictors with strong mutual information with the target variable while penalizing redundancy. The selected features are then processed by the TFT, which models both short- and long-term dependencies and integrates static and dynamic covariates through attention and gating mechanisms. This hybrid design specifically addresses two challenges in financial forecasting: (i) noisy and high-dimensional predictors; and (ii) nonlinear temporal dependencies.

3.5.2. Baseline models for comparison

To validate the effectiveness of the ISF-MID+TFT framework, several ML and DL models were implemented as baselines:

- *MLP:* basic non-linear neural model;
- *RBF Network:* kernel-based approximation of local dynamics;
- *SVR:* robust kernel regression model;
- *DT and RF:* interpretable and ensemble tree-based learners;
- *DNN:* multi-layer feedforward architecture;
- *LSTM:* recurrent network for temporal dependencies.

This comparison ensures that TFT is evaluated against both classical ML and strong DL baselines, as summarized later in Table 2.

3.5.3. Model implementation and hyperparameter tuning

Hyperparameter optimization was performed using Optuna (Akiba et al., 2019), configured with the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) sampler [20], an evolutionary optimization algorithm. The objective function minimized RMSE on the validation set over 100 trials. The hyperparameter search space was defined as follows:

- Learning rate: $[10^{-5}, 10^{-2}]$;
- Batch size: $\{32,64,128\}$;
- Hidden dimension: $[64,256]$;

- Dropout rate: [0.0,0.5];
- Attention heads (TFT): {2,4,8}.

To ensure reproducibility, experiments were executed with fixed random seeds for both data shuffling and model initialization. Early stopping was applied with a patience of 15 epochs to prevent overfitting. The search converged after approximately 70 trials, with marginal gains thereafter, confirming that 100 trials provided a sufficient budget for reliable convergence.

The optimal configuration for the TFT was found to be as follows: learning rate = 0.001; batch size = 64; hidden dimension = 128; dropout = 0.1; and four attention heads. Comparable search spaces were defined for the baseline models. All experiments were implemented in Python (3.12.1) with PyTorch (2.2) on a workstation with an Intel i7 processor, 32GB RAM, and an NVIDIA GPU with 16GB memory.

3.5.4. Evaluation metrics

To ensure a rigorous and balanced assessment, we employed both statistical error measures (RMSE, MAE, MSE, MAPE, R^2) and domain-specific indicators (DA, SR), together with computational efficiency (training time). This combination captures accuracy, explanatory power, relevance of decisions, and practical feasibility. All error metrics are computed on Min–Max normalized returns, making them dimensionless and directly comparable across features.

The notation used in this section is as follows: $O_i \in \mathbb{R}$: observed value at index i; $P_i \in \mathbb{R}$: predicted value at index i; N: number of samples; $\bar{O} = 1/N \sum_{i=1}^{N} O_i$: mean of observed values; sign(0): sign function; $\mathbb{1}\{0\}$: indicator function (1 if condition holds, 0 otherwise); μ_p: mean portfolio return per period; r_f: risk-free rate per period; σ_p: standard deviation of portfolio returns; T_{epoch}: training time per epoch (seconds).

3.5.5. Root mean squared error (RMSE)

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(O_i - P_i)^2} \tag{9}$$

Capturing the average magnitude of errors quadratically penalizes large deviations. It is useful when big mistakes are especially costly (Mohebbi et al., 2024).

3.5.6. Mean Absolute Error (MAE)

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|O_i - P_i| \tag{10}$$

Equation 10 represents the mean absolute deviation, which is more robust to outliers than RMSE and directly interpretable in normalized units (Mohebbi et al., 2024).

3.5.7. Mean squared error (MSE)

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(O_i - P_i)^2 \tag{11}$$

Large errors are highlighted in Equation 11 due to squaring, forming the basis of RMSE and many least-squares objectives (Mohebbi et al., 2024).

3.5.8. Mean absolute percentage error (MAPE)

$$\text{MAPE} = \frac{100}{N}\sum_{i=1}^{N}\left|\frac{O_i - P_i}{O_i}\right| \quad (12)$$

Equation 12 provides a scale-free percentage error for cross-series comparison, and its interpretability is advantageous, though instability may arise when $O_i \approx 0$ (Mohebbi et al., 2025(a)).

3.5.9. Coefficient of determination (R^2)

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(O_i - P_i)^2}{\sum_{i=1}^{N}(O_i - \bar{O})^2} \quad (13)$$

The proportion of variance is indicated in observations explained by the model; values closer to 1 imply stronger explanatory power (Mohebbi et al., 2025(b)).

3.5.10. Training time per epoch

Training time per epoch reports computational efficiency as T_{epoch} (s/epoch), reflecting scalability and deployability for real-time or large-scale applications (Mohebbi et al., 2025(a)).

3.5.11. Directional accuracy (DA)

$$\text{DA} = \frac{1}{N-1}\sum_{i=2}^{N}\mathbb{1}\{\text{sign}(O_i - O_{i-1}) = \text{sign}(P_i - P_{i-1})\} \quad (14)$$

Mean Directional Accuracy (MDA) measures the proportion of correctly predicted movement directions, which is particularly important in financial contexts, where the direction of change often outweighs precise error minimization (Costantini et al., 2016).

3.5.12. Sharpe ratio (SR)

$$\text{SR} = \frac{\mu_p - r_f}{\sigma_p} \quad (15)$$

Sharpe ratio (SR) evaluates risk-adjusted returns by comparing excess portfolio gains to volatility. Widely used in finance, values above 1 generally indicate favorable risk–return trade-offs and practical profitability (Bieganowski and Ślepaczuk, 2025).

4. Experimental results

4.1. Notation and units

O_i: observed return at index i; P_i: predicted return at index i; N: number of test samples; $\bar{O}$: mean of observed returns; MAE, RMSE: reported in the same unit as the output variable (Tehran Stock Exchange Price Index, TEPIX); MSE: reported in squared TEPIX units; MAPE: reported as a fraction in [0,1]; R: unitless; DA: reported in percent; SR: unitless; Time: seconds per epoch (s/epoch).

4.2. Benchmarking and proposed model

To determine the most effective model for predicting daily returns of the TSE index, several ML and DL models were compared. These included MLP, RBF networks, SVR, DT, RF, DNN, LSTM,

and TFT. All these models have been widely applied in the financial forecasting literature.

Figure 2 presents an initial performance comparison, where the TFT demonstrated the lowest prediction error in multiple sets of characteristics. LSTM ranked second, consistent with its established effectiveness for financial time series. However, the TFT's capability to model complex temporal dependencies and to leverage both static and dynamic covariates yields higher accuracy.

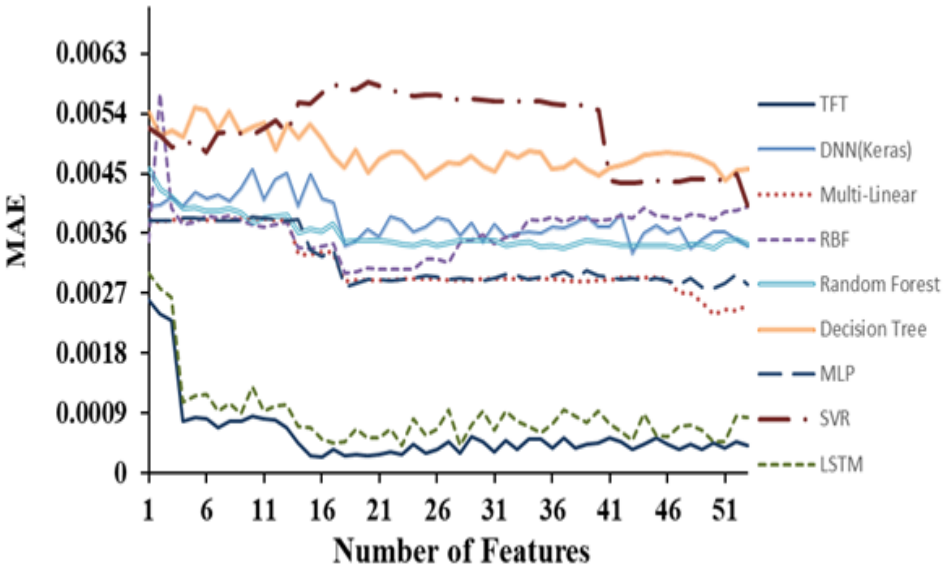

Figure 2. Comparative analysis of soft computing models for next-day stock index prediction

These findings validate the selection of the TFT as the proposed model for predicting the daily returns of the TSE index.

4.3. Performance comparison of TFT with other models

To evaluate its performance in predicting stocks, its performance is compared with cutting-edge deep and conventional machine learning algorithms, including LSTM, MLP, Random Forest, DNN, SVR, RBF, and DT. All of them have been compared with a range of performance metrics, which are MAE, MSE, RMSE, MAPE, R^2 Score, training time per epoch, DA, and SR. All are presented in a concise form in Table 2.

The results in Table 2 reveal that TFT achieves the best performance across all evaluation metrics, with LSTM as the closest competitor but still less efficient, while simpler models such as RF, MLP, and SVR show notably higher errors with minimum values for errors (MAE, RMSE) and a high value for R^2 (0.989), representing its high potential to track market trends. In addition, TFT outperforms in terms of DA (83.2%) and significantly outperforms models such as DT and SVR, which struggle to go beyond random chance. To avoid overestimation concerns, we note that DA is computed strictly on out-of-sample test windows, ensuring no look-ahead bias or data leakage. This provides a conservative but reliable measure of the accuracy of direction. Its value for SR (4.95) also confirms its performance in terms of risk-adjusted returns, making it more reliable for financial forecasting. Here, the SR was calculated under an idealized daily trading strategy (long/short at the close, risk-free rate set to zero, no transaction costs or slippage). Thus, the reported SR should be interpreted as an upper bound of achievable profitability; in practice, market frictions would slightly lower the ratio.

Table 2. Performance comparison of different models based on various evaluation metrics (errors reported in actual TEPIX units; MSE in squared units; MAPE as a fraction; DA in percent; Time in s/epoch).

Model	MAE	RMSE	MAPE	R^2	DA (%)	SR	Time (s/epoch)
TFT	0.000	0.001	0.021	0.989	83.200	4.950	120
LSTM	0.000	0.001	0.029	0.976	78.500	3.610	110
MLP	0.002	0.003	0.052	0.945	72.300	2.350	70
RF	0.002	0.004	0.058	0.938	69.800	1.250	30
DNN	0.003	0.004	0.065	0.926	68.200	1.850	90
RBF	0.003	0.005	0.071	0.918	66.700	0.670	25
SVR	0.003	0.005	0.073	0.912	65.900	0.610	20
DT	0.004	0.006	0.082	0.895	63.400	0.400	5

Although the TFT model has the longest computation time per epoch (120s), its performance in all key evaluation metrics is worth its long training period. Computational intensity comes with the use of a self-attention mechanism and sequential feature extraction, but it can also detect complex trends in the market more effectively than traditional approaches. DT (5 s/epoch) and SVR (20 s/epoch) are more efficient computationally, but their predictive accuracy is lower than that of TFT. Ultimately, the trade-off between computation time and predictive power favors TFT, as its substantial accuracy gains outweigh the additional training cost.

4.4. Feature compact analysis: selection and reduction

Feature selection and reduction are significant steps in enhancing the performance of predictive models. In the case of this research work, two advanced statistical methods, mRMR and PCA, were used to choose the most relevant features of the data.

4.4.1. Feature selection using mRMR

As introduced in Section 3, mRMR was adopted for feature selection to maximize relevance to the target variable and minimize redundancy between feature inputs. Due to the high dimensions of financial information, mRMR in its traditional form was computationally expensive; for that reason, two approximation techniques, Mutual Information Difference (MID) and F-Test Correlation Difference (FCD), have been adopted for efficiency gains (Mohebi et al., 2022a).

For discrete variables, the mRMR criterion is approximated by using MID (Mohebi et al., 2022a):

$$MID(x_i) = I(x_i; y) - \frac{1}{|S|}\sum_{x_j \in S} I\left(x_i; x_j\right) \tag{16}$$

where:

- $I(x_i; y)$ is the mutual information between the feature x_i and the target variable y, measuring how much knowledge x_i reduces the uncertainty about y.
- $I(x_i; x_j)$ is the mutual information between a feature x_i and another selected feature x_j, capturing redundancy.
- $|S|$ is the number of selected features in the subset S.

Mutual information $I(A; B)$ is defined as (Mohebi et al., 2022b):

$$I(A; B) = \sum_{a \in A}\sum_{b \in B} p\,(a, b)\log\frac{p(a, b)}{p(a)p(b)} \tag{17}$$

where $p(a,b)$ is the joint probability distribution of A and B, while $p(a)$ and $p(b)$ are their respective marginal probabilities.

For continuous variables, mutual information is replaced by an F-test score, leading to the FCD approximation (Mohebbi Najm Abad and Soleimani, 2018):

$$FCD(x_i) = F(x_i, y) - \frac{1}{|S|}\sum_{x_j \in S} F\left(x_i, x_j\right) \quad (18)$$

where:

- $F(x_i, y)$ is the F-statistic, which measures the correlation between the feature x_i and the target variable y.
- $F(x_i, x_j)$ is the F-statistic that measures the correlation between a feature x_i and another selected feature x_j, indicating redundancy.
- $|S|$ is the number of selected features in the subset S.

The features were prioritized by the MID and FCD methods based on their relevance to the target variable and mutual independence. Prioritizations were preserved in the feature vector ζ. The subsets of characteristics ($1 \leq j \leq |\zeta|$) were evaluated using the TFT model to determine the optimal number of characteristics to forecast. The MAE was used as the performance metric.

The results, as illustrated in Figure 3, show that the accuracy of the model is initially enhanced by adding more selected features. However, after the 16th feature, the accuracy drops due to the addition of redundant or less significant features. Between the two mRMR approximation methods, MID outperformed FCD up to the 16th feature. Consequently, MID was selected as the preferred method for feature selection.

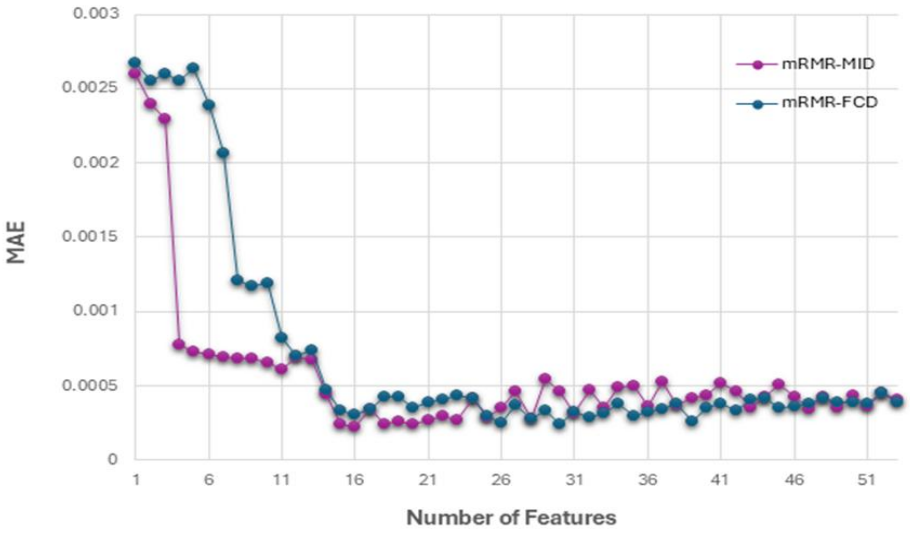

Figure 3. Impact of feature selection on prediction accuracy

The names of the first 16 best priority features using MID, which result in the highest accuracy in the prediction model, are presented in Table 3.

Table 3. Features prioritization based on importance scores

Rank	Feature Name	Description
1	RDP5	Relative Percentage Difference of Index Returns (5 Days)

2	U3	Euro Price Change (3 Days)
3	OP1	Crude Oil Price Change (1 Day)
4	PICS01	1-Day Return of Top 50 Active Stock Index
5	TV2	Relative Volume Change (2 Days)
6	RDP15	Relative Percentage Difference of Index Returns (15 Days)
7	D2	USD Price Change (2 Days)
8	D1	USD Price Change (1 Day)
9	D3	USD Price Change (3 Days)
10	OP3	Crude Oil Price Change (3 Days)
11	TV1	Relative Volume Change (1 Day)
12	RDP10	Relative Percentage Difference of Index Returns (10 Days)
13	GPd2	Gold Coin Price Change (2 Days)
14	TEDPIX3	3-Day Return of Overall Stock Index
15	OP2	Crude Oil Price Change (2 Days)
16	TEDPIX1	1-Day Return of Overall Stock Index

4.4.2. Feature extraction using PCA

PCA was used as a feature extraction algorithm to transform correlated input variables into an orthogonal principal component with preserved variance. The number of dimensions of the PCA was systematically varied to determine the optimal number of components, and the predictive performance of the TFT model was evaluated for each setting.

The results, as plotted in Figure 4, show that the accuracy of the model improves with additional PCA components to 30 dimensions. Beyond this, further increases in dimensionality yield diminishing returns in predictive capability. This suggests that PCA effectively reduces dimensionality while preserving significant information up to a certain threshold.

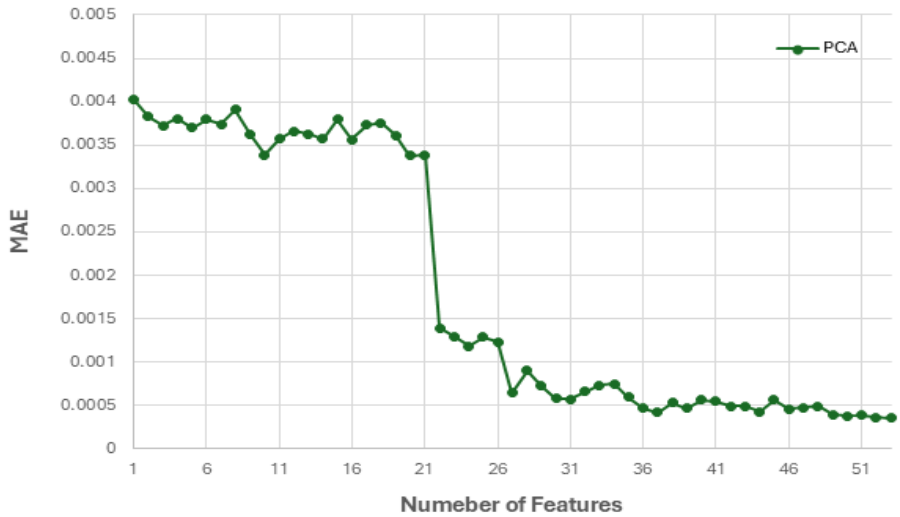

Figure 4. Impact of PCA Dimensionality on TFT Model Performance

4.4.3. Comparative analysis of mRMR and PCA

To compare the effectiveness of feature selection (mRMR) and feature extraction (PCA) techniques, their respective impacts on the predictive capacity of the TFT model were analyzed. The results of this comparison are shown in Figure 5.

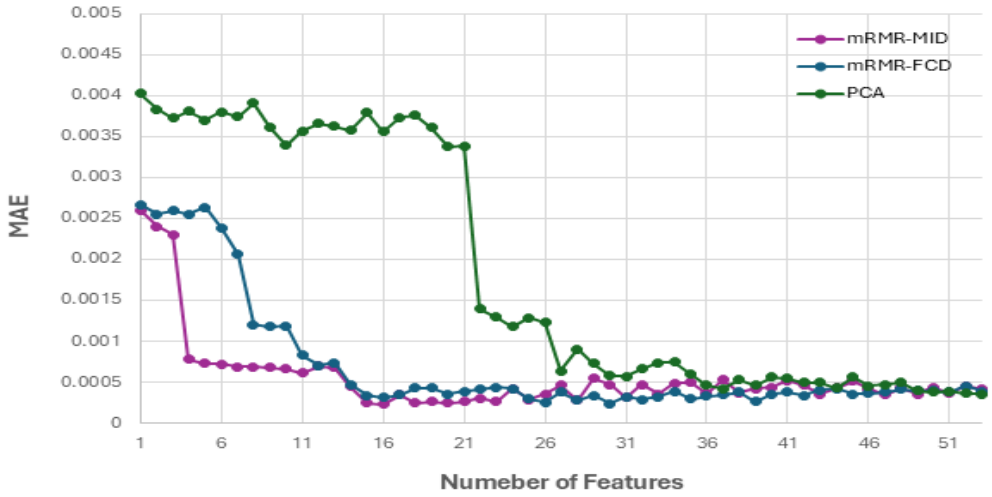

Figure 5. Comparison of mRMR and PCA techniques on TFT model performance

The findings indicate that while both techniques enhance model accuracy compared to using raw features, the mRMR-based feature selection method consistently outperforms PCA in terms of predictive accuracy. This advantage comes from the ability of mRMR to select the most relevant features explicitly. In contrast, PCA transforms the features into new components that may not necessarily capture the most predictive aspects of the data for the target variable. Based on this analysis, mRMR with the MID approximation was chosen as the preferred preprocessing technique for this study.

4.5. Impact of input feature selection on TFT model performance

To evaluate the contribution of individual input features to the precision of the prediction in the TFT model, three scenarios with a uniform and consistent network configuration were considered.

- ***Effective Features****:* In this scenario, only the effective features that have been proven to be significant for the prediction of the stock index were used as input to the model. In total, 25 such effective features have been considered.
- ***Effective Features and Statistical Features****:* Building on the same scenario, in this model, additional statistical characteristics derived from values of the historical parameters were included. There were additional features with information of historical perspective on parameter variations, increasing the total number of input features to 53.
- ***Selected Features with mRMR-MID****:* In the last scenario, to optimize the set of features, a subset of the most important features was derived using the mRMR-MID algorithm. A comparison of results showed that increasing the number of features to 16 increased the prediction accuracy, but no contribution to the accuracy was noticed for feature values over 16. Hence, the selection of 16 features was identified as the most effective approach to accurately predict the stock index. A comparison of these three scenarios is presented in Table 4.

Table 4. Results of the TFT model evaluation using different feature sets

Model Input Features	Number of Features	MAE	MSE	RMSE
Selected Influential Features	25	0.003	0.000	0.005
Influential Statistical Features	53	0.000	0.000	0.001
Optimally Selected Features	16	0.000	0.000	0.000

The results illustrate that the incorporation of statistical features significantly improves the precision of the prediction. Regardless of its success in representing long-term dependencies through self-attention mechanisms, including historical values of the stock index and other parameters with current values, leads to more reliable predictions. Furthermore, according to the error values in Table 4, selecting the 16 most relevant and highest-priority features, as determined by the mRMR-MID, generated a minimum model error.

4.6. Proposed algorithm for feature selection

In the proposed feature selection algorithm, an initial selection of feature sets was derived through a manual selection by the researcher. By carefully examining the variation in the pattern errors in Figure 5, an improvement in the accuracy of the prediction was observed with a feature count that increased to 16, and no further improvement was observed with any subsequent addition of features. Among the 16 initial features, the five most significant played the most important role in reducing model errors, which are:

- One-day lagged return of the top 50 active stocks index (PIC501)
- Two-day lagged USD price change (D2)
- Three-day lagged return of the overall index (TEDPIX3)
- Two-day lagged oil price change (OP2)
- One-day lagged return of the overall index (TEDPIX1)

Based on these observations, these five features were first selected, and then prioritization of the remaining features with the mRMR-MID algorithm was conducted. After that, these selected factors have been sequentially fed into the TFT model to predict the daily stock index return in Tehran. The flowchart of the proposed algorithm is presented in Figure 6.

In the implementation of ISF-MID, the redundancy penalty parameter α was empirically set to 0.5 as a balanced trade-off between the relevance of the features and the redundancy. Sensitivity tests with $\alpha \in \{0.3,0.5,0.7\}$ showed stable results, confirming robustness. To ensure comparability between features, all inputs were pre-processed using Min-Max normalization to $[0,1]$, as detailed in Section 4.2. This prevented scale disparities from biasing the mutual information estimates.

4.7. Comparative analysis for optimal TFT feature set

The following performance confirms the effectiveness of a larger number of prioritized features in improving the accuracy of the TFT model, which can be observed in Figure 7.a. The feature prioritization extracted via the proposed algorithm, known as the ISF-MID, is also displayed. Initially, the first five of them did not appear to be sufficient to provide a significant improvement in the accuracy of the prediction. However, when the sixth feature was incorporated, the necessity of the first five features was realized, resulting in a substantial improvement in the accuracy of the model, with lower error than the best result achieved in the previous phase.

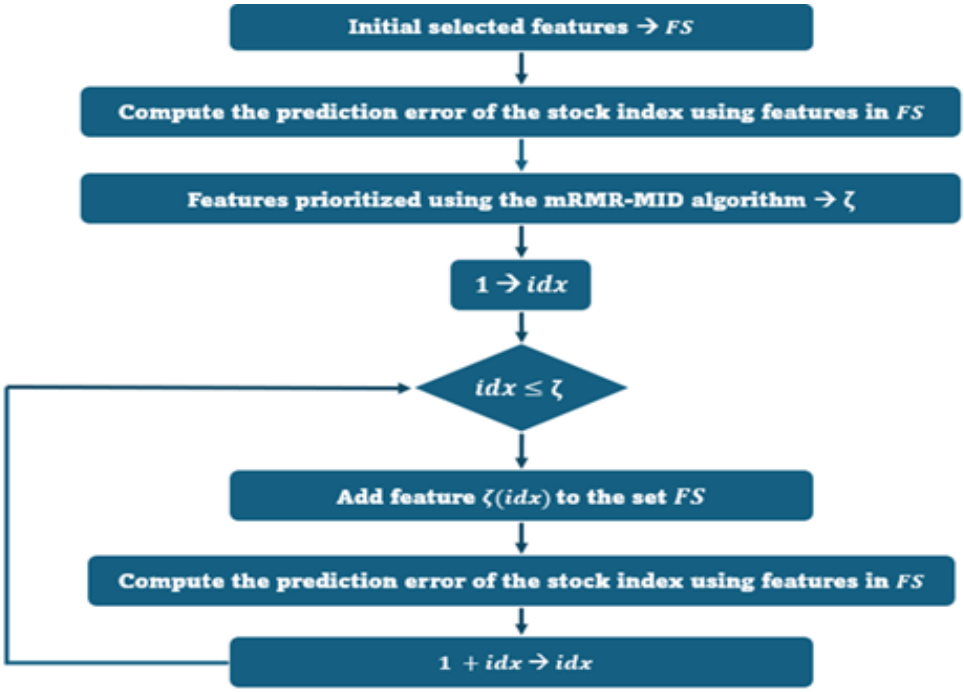

Figure 6. Flowchart of the proposed feature selection algorithm

This approach was further extended by introducing additional features to refine the stock index prediction model. As can be seen in Figure 7.a, both the sixth and eleventh characteristics, the difference in the five-day return in stocks and the three-day lagged oil price change, proved useful in reducing model error. Consequently, these two additional features were added to the initial selection, bringing the total to seven. The remaining were ranked with the mRMR-MID algorithm and incorporated into the TFT model. The evaluation result from this stage is shown in Figure 7.b.

As demonstrated in Figure 7.b, increasing the number of initial features to eight resulted in a substantial reduction in error, beyond which no further improvements were observed. Hence, the eighth feature, the relative change in the trading volume the previous day, was included in the selection at the last stage. Since no additional improvement in prediction accuracy could be noticed with an additional feature, these eight features were determined as the most effective features to predict the TSE index using the TFT model. The following is the ultimate selection of the chosen features:

- One-day lagged return of the top 50 active stocks index
- Two-day lagged USD price change
- Three-day lagged return of the overall index
- Two-day lagged oil price change
- One-day lagged return of the overall index
- Relative difference in five-day index return
- Three-day lagged oil price change
- Relative change in trading volume over one day

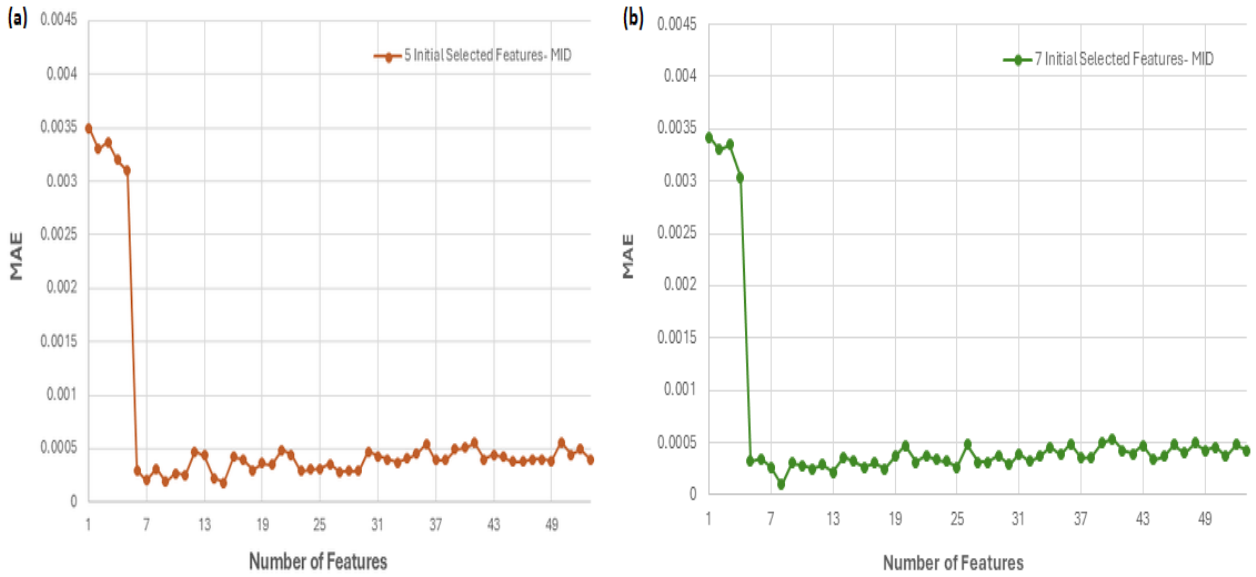

Figure 7. Evaluation of the TFT model with feature prioritization: (a) performance with five initial features (ISF-MID), (b) performance with seven initial features

A comparison was made with the mRMR-MID algorithm to assess the effectiveness of the proposed feature selection algorithm in reducing dimensions while improving prediction accuracy. Figure 8 illustrates the output of both feature selection algorithms. By comparing curves in Figure 8, a comparative analysis shows that the best performance of the TFT model with eight dimensions was achieved when the ISF-MID algorithm was utilized, confirming its effectiveness in feature selection. The prediction error values for feature selection cases for the prediction model with TFT are presented in Table 5. Experimental observations significantly improve predictive accuracy when using the proposed feature selection scheme. Specifically, when all 53 features are considered, an MAE value of 0.00042 is generated. However, when mRMR-MID feature selection is adopted, an improvement of 0.000239 is observed in the error value. Furthermore, the MAE decreased to 0.000230 by applying the proposed ISF-MID approach with a compact subset of six features. While this demonstrates the robustness of ISF-MID in extracting informative features, the six-feature configuration did not surpass the final eight-feature configuration. The best performance was ultimately achieved with ISF-MID using eight selected features, which produced the lowest MAE of 0.000097 and the highest R^2 (0.989). This establishes the eight-feature ISF-MID subset as the definitive optimal configuration for the TFT model in the study.

Table 5. Performance evaluation of the TFT model with different feature selection methods

Feature Selection Approach	Number of Features	MAE
All Features	53	0.000
mRMR-MID Selected Features	16	0.000
Proposed ISF-MID (6 Features)	6	0.000
Proposed ISF-MID (8 Features)	8	0.000

4.8. Evaluating training stability and prediction accuracy

A critical objective in model training is to have an optimal balance between underfitting and overfitting. The underfit and overfit balance is measured through observation of training and validation loss curves, and such plots illustrate the generalizability of a model to new observations and unseen data. A well-trained model has a decreasing trend in both losses up to a point where they

stabilize, meaning convergence. Moreover, a minimal gap between training and validation loss implies that the model has effectively captured trends in the data without overfitting, as shown in Figure 9.

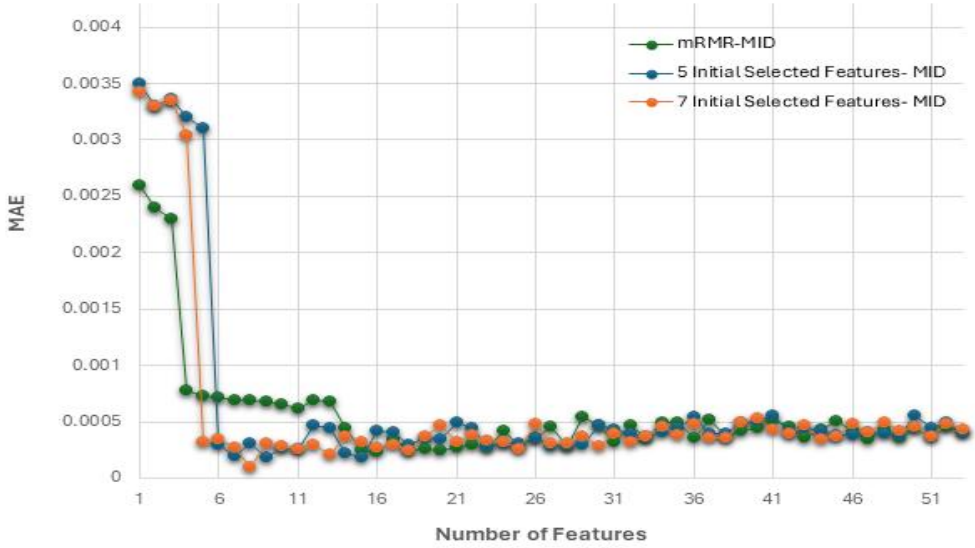

Figure 8. Comparative analysis of feature selection algorithms: ISF-MID vs. mRMR-MID in the TFT model

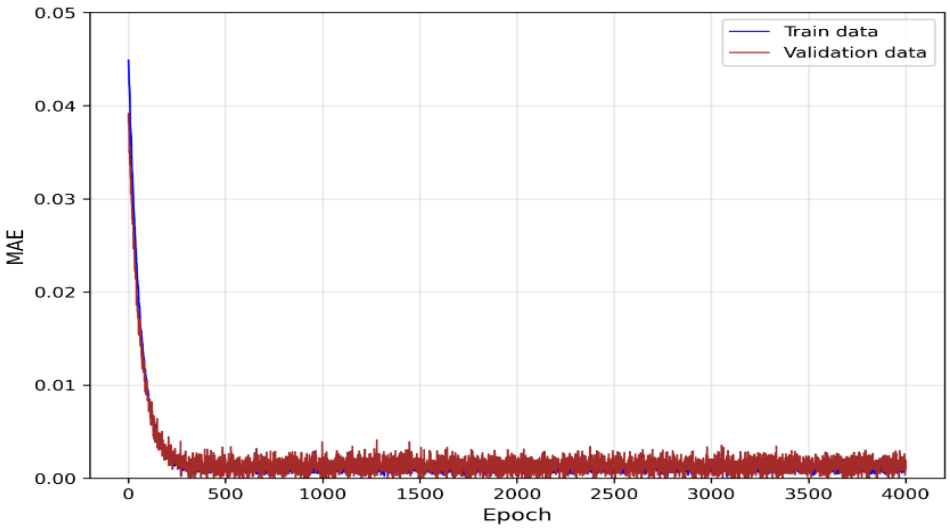

Figure 9. Training and validation loss curves for assessing model generalizability

Beyond loss curves, the predictive performance of the model should also be evaluated by comparing actual and predicted values for the stock index. To achieve this, the predicted values for the TFT model were compared with actual observations in several test samples. Figure 10.a shows a regression plot of test samples, illustrating how closely the model output aligns with the actual trends in the marketplace. Figure 10.b provides a comprehensive performance analysis in all test samples, further validating the accuracy of the model. The results show that with the optimally chosen features, the TFT model is most effective at predicting the real values of the stock market indices. This is corroborated by the minimal deviations within the prediction, which highlights the effectiveness of the model in describing market fluctuations. That means that the TFT model could be considered a

powerful and reliable tool for financial time series prediction, as it manages to balance both training stability and predictive accuracy.

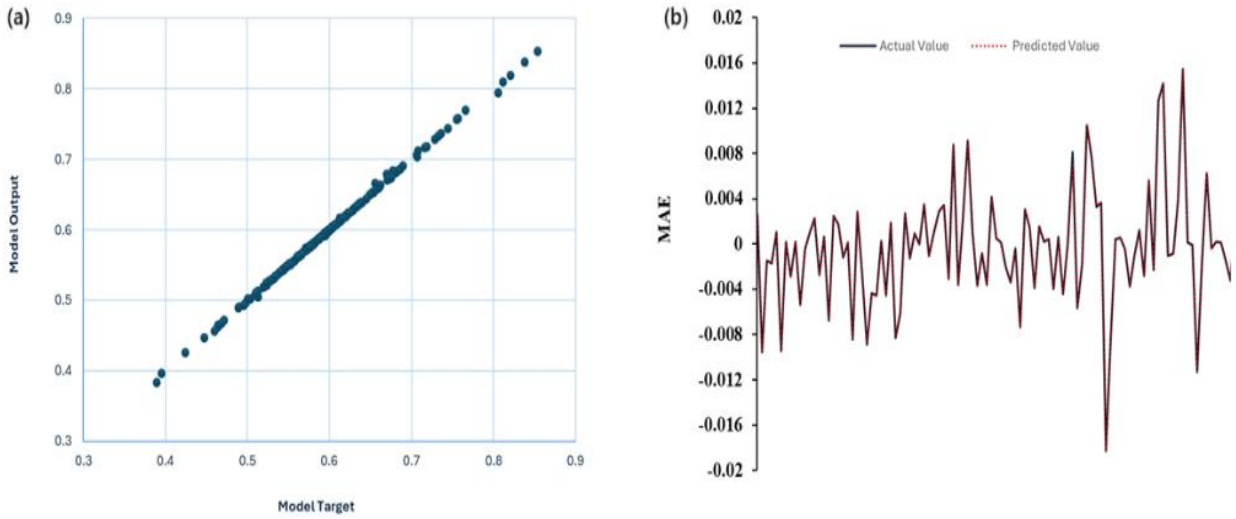

Figure 10. Evaluation of TFT model predictions: (a) Regression plot of predicted vs. actual values, (b) performance analysis across the full test set.

4.9. Statistical evaluation of the proposed model

The Friedman test is used to assess the significance of the results obtained from the predictions. It is a statistical method applied for two-way analysis of variance (for non-parametric data) using a ranking approach. In addition, the Friedman test can be used to compare the mean rankings of different groups. Table 6 presents the descriptive statistics of the errors in the models examined under the rolling window approach. As observed, the mean error in the TFT and LSTM models is lower than that of the other models, respectively. The Friedman test is a nonparametric test, equivalent to repeated measures analysis of variance (ANOVA), which is used to compare mean ranks between k variables (groups). In this test, a significance level less than 0.05 indicates a statistically significant difference in the errors of the models studied.

The results of the Friedman test are presented in Table 7. The findings indicate that the TFT model has a lower error compared to the other models examined. Since the significance level is below 0.05 and based on the mean ranks obtained, the TFT and LSTM models have the lowest mean ranks, respectively.

Table 6. Descriptive statistics of model errors in the rolling window approach

Prediction Model	Mean	Standard Deviation	Minimum	Maximum	Kurtosis	Skewness
MLP	0.002	0.003	0.000	0.056	22.733	3.617
SVR	0.003	0.003	0.000	0.062	47.313	4.771
DNN	0.003	0.003	0.000	0.040	18.916	3.256
Linear Regression	0.002	0.003	0.000	0.044	69.103	6.389
DT	0.004	0.005	0.000	0.055	19.705	3.472
RF	0.003	0.004	0.000	0.050	32.382	4.185
LSTM	0.000	0.001	0.000	0.002	156.975	10.054
TFT	0.000	0.001	0.000	0.025	464.283	20.063

Table 7. Comparison of median error in studied models based on the Friedman test

Model	Median Error	
MLP	5.070	
SVR	5.690	
DNN	5.290	
Linear Regression	4.510	
DT	5.870	
RF	5.100	
LSTM	2.840	
TFT	1.640	
Friedman Test Results	Chi-Square Statistic	1730.029
	Degrees of Freedom	7.000
	Significance Level	0.000

5. Conclusion

This study proposed a forecasting framework for daily returns of the TSE by combining domain-informed feature selection with deep learning. Among the candidate models, the TFT achieved the strongest performance by capturing temporal dependencies and nonlinear patterns in TEPIX data. An extensive set of 53 candidate features was initially constructed from 12 technical, market, and macroeconomic indicators. To reduce redundancy and improve efficiency, the mRMR-MID method was applied, resulting in 16 relevant features. Building on this, we introduced the ISF-MID algorithm, which integrates expert-informed initialization with redundancy-aware selection. ISF-MID achieved its best performance with only 8 features, producing the lowest MAE (0.000) and the highest R^2 (0.989). These results established TFT combined with ISF-MID (8 features) as the optimal configuration, demonstrating that a compact and well-prioritized subset can deliver highly accurate and interpretable forecasts in complex financial environments. In general, the integration of ISF-MID with TFT not only improved predictive accuracy but also provided a reproducible benchmark for researchers and a practical guideline for practitioners seeking efficient and interpretable models in financial forecasting.

Future work could extend this framework by incorporating transaction costs, liquidity constraints, and market frictions for realistic profitability assessment; adapting ISF-MID to nonlinear dependency measures or hybrid optimization strategies; and applying the framework to multi-market and cross-asset settings to evaluate its generalizability.

Declarations:

On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. Akiba, T., Sano, S., Yanase, T., Ohta, T. and Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. In Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining (pp. 2623-2631). New York, NY. https://doi.org/10.1145/3292500.3330701
2. Alam, K., Bhuiyan, M. H., Haque, I. U., Monir, M. F. and Ahmed, T. (2024). Enhancing stock market prediction: A robust LSTM-DNN model analysis on 26 real-life datasets. *IEEE Access*, 12, pp. 122757-122768. https://doi.org/10.1109/ACCESS.2024.3434524
3. Awad, A. L., Elkaffas, S. M. and Fakhr, M. W. (2023). Stock market prediction using deep reinforcement learning. *Applied System Innovation*, 6(6), p. 106. https://doi.org/10.3390/asi6060106
4. Bao, W., Yue, J. and Rao, Y. (2017). A deep learning framework for financial time series using stacked autoencoders and long-short term memory. *PloS One,* 12(7), A. e0180944.

https://doi.org/10.1371/journal.pone.0180944
5. Bieganowski, B. and Ślepaczuk, R. (2025). Supervised autoencoder MLP for financial time series forecasting. *Journal of Big Data*, 12(1), p. 207. https://doi.org/10.1186/s40537-025-01267-7
6. Costantini, M., Cuaresma, J. C. and Hlouskova, J. (2016). Forecasting errors, directional accuracy and profitability of currency trading: The case of EUR/USD exchange rate. *Journal of Forecasting,* 35(7), pp. 652-668. https://doi.org/10.1002/for.2398
7. Dashtaki, S. M., Chagahi, M. H., Bahadori, A., Moshiri, B., Piran, M. J. and Montazeri, A. (2025). HSIF: A transformer-based cross-attention framework for cryptocurrency trend forecasting via multimodal sentiment–market fusion. *IEEE Access*, 13, pp. 156600-156612. https://doi.org/10.1109/ACCESS.2025.3605522
8. Davoud Abadi, M. (2025). Forecasting financial market indices using a transformer-based model (Master's thesis). Allameh Tabataba'i University, Tehran, Iran (In Persian).
9. Dos Santos, J. P. G., Moutinho, V. M. F., Madaleno, M. and Teixeira, Z. M. D. S. S. (2025). Examining the effect of macroeconomic, institutional, and capital market drivers on infrastructure investment. *Economic Analysis and Policy*, 86, pp. 165-190. https://doi.org/10.1016/j.eap.2025.02.038
10. Emami Gohari, H., Dang, X. H., Shah, S. Y. and Zerfos, P. (2024). Modality-aware transformer for financial time series forecasting. In Proceedings of the 5th ACM International Conference on AI in Finance (pp. 677-685). Association for Computing Machinery. https://doi.org/10.1145/3677052.3698654
11. Gandhudi, M., Alphonse, P. J. A., Velayudham, V., Nagineni, L. and Gangadharan, G. R. (2024). Explainable causal variational autoencoders based equivariant graph neural networks for analyzing the consumer purchase behavior in E-commerce. *Engineering Applications of Artificial Intelligence*, 136, A. 108988. https://doi.org/10.1016/j.engappai.2024.108988
12. Guo, Z., Ye, W., Yang, J. and Zeng, Y. (2017). Financial index time series prediction based on bidirectional two-dimensional locality preserving projection. In Proceedings of the 2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA) (pp. 934 –938). IEEE, New York. https://doi.org/10.1109/ICBDA.2017.8078775
13. Hartanto, S. and Gunawan, A. A. S. (2024). Temporal fusion transformers for enhanced multivariate time series forecasting of Indonesian stock prices. *International Journal of Advanced Computer Science and Applications (IJACSA),* 15(7), pp. 1-22. https://doi.org/10.14569/IJACSA.2024.0150713
14. https://arxiv.org/abs/2503.18096
15. Jaiwang, G. and Jeatrakul, P. (2018). Enhancing support vector machine model for stock trading using optimization techniques. In 2018 International Conference on Digital Arts, Media and Technology (ICDAMT) (pp. 23-28). Phayao, Thailand. https://doi.org/10.1109/ICDAMT.2018.8376489
16. John, A. and Latha, T. (2023). Stock market prediction based on deep hybrid RNN model and sentiment analysis. *Automatika*, 64(4), pp. 981-995. https://doi.org/10.1080/00051144.2023.2217602
17. Kumarappan, J., Rajasekar, E., Vairavasundaram, S., Kotecha, K. and Kulkarni, A. (2024). Federated learning enhanced MLP–LSTM modeling in an integrated deep learning pipeline for stock market prediction. *International Journal of Computational Intelligence Systems*, 17(1), p. 267. https://doi.org/10.1007/s44196-024-00680-9
18. Kumbure, M. M., Lohrmann, C., Luukka, P. and Porras, J. (2022). Machine learning

techniques and data for stock market forecasting: A literature review. *Expert Systems with Applications,* 197, A. 116659. https://doi.org/10.1016/j.eswa.2022.116659

19. Li, Y., Ni, P. and Chang, V. (2020). Application of deep reinforcement learning in stock trading strategies and stock forecasting. *Computing*, 102(6), pp. 1305-1322. https://doi.org/10.1007/s00607-019-00773-w
20. Liang, X., Ge, Z., Sun, L., He, M. and Chen, H. (2019). LSTM with wavelet transform based data preprocessing for stock price prediction. *Mathematical Problems in Engineering,* 2019(1), A. 1340174. https://doi.org/10.1155/2019/1340174
21. Lim, B., Arık, S. Ö., Loeff, N. and Pfister, T. (2021). Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International journal of forecasting*, 37(4), pp. 1748-1764. https://doi.org/10.1016/j.ijforecast.2021.03.012
22. Liwei, T., Li, F., Yu, S. and Yuankai, G. (2021). Forecast of LSTM-XGBoost in Stock Price Based on Bayesian Optimization. *Intelligent Automation & Soft Computing*, 29(3), pp. 855–868. https://doi.org/10.32604/iasc.2021.016805
23. Mohebbi Najm Abad, J. and Soleimani, A. (2018). Novel feature selection algorithm for thermal prediction model. *IEEE Transactions on Very Large-Scale Integration (VLSI) Systems,* 26(10), pp. 1831-1844. https://doi.org/10.1109/TVLSI.2018.2841318
24. Mohebbi, M. R., Kafash, E. and Döller, M. (2025a). Multi-Agent Trajectory Prediction for Urban Environments with UAV Data Using Enhanced Temporal Kolmogorov-Arnold Networks with Particle Swarm Optimization. In ICAART (2) (pp. 586-597). ScitePress, Setúbal, Portugal. https://doi.org/10.5220/0013243100003890
25. Mohebbi, M. R., Klinger, J., Abad, J. M. N., Döller, M. and Tavasoli, M. (2025b). Advanced driving behavior analysis through Kolmogorov-Arnold network and UAV traffic data. In International Conference on Machine Learning and Computing (pp. 305-322). Cham: Springer Nature Switzerland, Switzerland. https://doi.org/10.1007/978-3-031-94892-3_23
26. Mohebbi, M. R., Zada, M. J. H., Rostami-Shahrbabaki, M., Döller, M. and Yamnenko, I. (2024). Network traffic co-movement assessment via oriented basis signal processing and ensemble decision trees. In 2024, IEEE 27th International Conference on Intelligent Transportation Systems (ITSC) (pp. 1948-1955). IEEE, Piscataway, NJ. https://doi.org/10.1109/ITSC58415.2024.10919555
27. Mohebi, S., Fadaeinejad, M. E., Osoolian, M. and Hamidizadeh, M. R. (2022a). Feature Selection for the Prediction Model of the Tehran Stock Exchange Index by Dimensionality Reduction Techniques. *Financial Research Journal,* 24(4), pp. 577-601. https://doi.org/10.22059/frj.2021.325675.1007202
28. Mohebi, S., Fadaeinezhad, M. and Osoolian, M. (2022b). Predicting the Tehran Stock Exchange Index using support vector regression; Based on the dimension reduction technique. *Financial Management Strategy*, 10(3), pp. 1–26 (In Persian). https://doi.org/10.22051/jfm.2022.35543.2528
29. Nabipour, M., Nayyeri, P., Jabani, H., Mosavi, A., Salwana, E. (2020). Deep learning for stock market prediction. *Entropy*, 22(8), p. 840. https://doi.org/10.3390/e22080840
30. Niu, H., Xu, K. and Wang, W. (2020). A hybrid stock price index forecasting model based on variational mode decomposition and LSTM network: A hybrid stock price index forecasting model based on variational mode decomposition and LSTM network. *Applied Intelligence*, 50(12), pp. 4296-4309. https://doi.org/10.1007/s10489-020-01814-0
31. Pourroostaei Ardakani, S., Du, N., Lin, C., Yang, J. C., Bi, Z. and Chen, L. (2023). A federated learning-enabled predictive analysis to forecast stock market trends. *Journal of*

Ambient Intelligence and Humanized Computing, 14(4), pp. 4529-4535. https://doi.org/10.1007/s12652-023-04570-4

32. Sezer, O. B., Gudelek, M. U. and Ozbayoglu, A. M. (2020). Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Applied Soft Computing*, 90(1), A. 106181. https://doi.org/10.1016/j.asoc.2020.106181
33. Sharma, R. and Mehta, K. (2024). Stock market predictions using deep learning: developments and future research directions. *Deep Learning Tools for Predicting Stock Market Movements*, 1(1), pp. 89-121. https://doi.org/10.1002/9781394214334.ch4
34. Sun, Y., Yuan, H. and Xu, F. (2025). Financial sentiment analysis for pre-trained language models incorporating dictionary knowledge and neutral features. *Natural Language Processing Journal*, 11(1), A. 100148. https://doi.org/10.1016/j.nlp.2025.100148
35. Tulcanaza Prieto, A. B. and Lee, Y. H. (2019). Determinants of stock market performance: VAR and VECM designs in Korea and Japan. *Global Business & Finance Review (GBFR),* 24(4), pp. 24-44. https://doi.org/10.17549/gbfr.2019.24.4.24
36. Wang, C., Chen, Y., Zhang, S. and Zhang, Q. (2022). Stock market index prediction using deep Transformer model. *Expert Systems with Applications*, 208, A. 118128. https://doi.org/10.1016/j.eswa.2022.118128
37. Wei, D. (2019). Prediction of stock price based on an LSTM neural network. In the 2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM) (pp. 544-547). IEEE, New York. https://doi.org/10.1109/AIAM48774.2019.00113
38. Wei, X., Chen, W. and Li, X. (2021). Exploring the financial indicators to improve the pattern recognition of economic data based on machine learning. *Neural Computing and Applications,* 33(2), pp. 723-737. https://doi.org/10.1007/s00521-020-05094-0
39. Wei, Y., Gu, X., Feng, Z., Li, Z. and Sun, M. (2024). Feature extraction and model optimization of deep learning in stock market prediction. *Journal of Computer Technology and Software*, 3(4), 1-22. https://doi.org/10.5281/zenodo.13622489
40. Wu, Y. (2023). Comparison between transformer, informer, autoformer and non-stationary transformer in financial market. *Applied and Computational Engineering*, 29, pp. 68-78. https://doi.org/10.54254/2755-2721/29/20230874
41. Wu, Y., Fu, Z., Liu, X. and Bing, Y. (2023). A hybrid stock market prediction model based on GNG and reinforcement learning. *Expert Systems with Applications*, 228, A. 120474. https://doi.org/10.1016/j.eswa.2023.120474
42. Yang, J., Li, P., Cui, Y., Han, X. and Zhou, M. (2025). Multi-sensor temporal fusion transformer for stock performance prediction: An adaptive Sharpe ratio approach. *Sensors*, 25(3), p. 976. https://doi.org/10.3390/s25030976
43. Zhang, J., Cui, S., Xu, Y., Li, Q. and Li, T. (2018). A novel data-driven stock price trend prediction system. *Expert Systems with Applications,* 97(1), pp. 60-69. https://doi.org/10.1016/j.eswa.2017.12.026
44. Zheng, H., Wu, J., Song, R. and Guo, L. (2024). Predicting financial enterprise stocks and economic data trends using machine learning time series analysis. *Applied and Computational Engineering*, 87(1), pp. 26–32. https://doi.org/10.20944/preprints202407.0895.v1
45. Zhong, X. and Enke, D. (2017). Forecasting daily stock market return using dimensionality reduction. *Expert Systems with Applications*, 67(2), pp. 126-139. https://doi.org/10.1016/j.eswa.2016.09.027