The PSR-Transformer Nexus: A Deep Dive into Stock Time Series Forecasting

Nguyen Ngoc Phien[1,2], Jan Platos[3]
Center for Applied Information Technology, Ton Duc Thang University, Ho Chi Minh City, Vietnam[1]
Faculty of Information Technology, Ton Duc Thang University, Ho Chi Minh City, Vietnam[2]
Department of Computer Science-Faculty of Electrical Engineering and Computer Science,
VSB Technical University of Ostrava, Czech Republic[3]

***Abstract*—Accurate stock market forecasting has remained an elusive endeavor due to the inherent complexity of financial systems dynamics. While deep neural networks have shown initial promise, robustness concerns around long-term dependencies persist. This research pioneers a synergistic fusion of nonlinear time series analysis and algorithmic advances in representation learning to enhance predictive modeling. Phase space reconstruction provides a principled way to reconstruct multidimensional phase spaces from single variable measurements, elucidating dynamical evolution. Transformer networks with self-attention have recently propelled state-of-the-art results in sequence modeling tasks. This paper introduces PSR-Transformer Networks specifically tailored for stock forecasting by feeding PSR interpreted constructs to transformer encoders. Extensive empirical evaluation on 20 years of historical equities data demonstrates significant accuracy improvements along with enhanced robustness against LSTM, CNN-LSTM and Transformer models. The proposed interdisciplinary fusion establishes new performance benchmarks on modeling financial time series, validating synergies between domain-specific reconstruction and cutting-edge deep learning.**

***Keywords*—Stock market forecasting; deep learning; chaos theory; phase space reconstruction; transformer neural networks; time series analysis**

I. Introduction

Stock price forecasting remains a pivotal yet challenging problem, as financial markets display highly chaotic properties arising from complex interplay of diverse macroeconomic factors, events, and psychology [1] [2] [3]. Traditional linear statistical models like ARIMA face inherent limitations to accurately characterize the nonstationary, nonlinear patterns ubiquitous in financial time series data [4] [5]. Since markets rebounded after the 2020 pandemic shocks, advancing machine learning predictions for equities has regained immense research attention [6] [7].

In recent times, deep neural networks like long short-term memory (LSTM) recurrent networks have achieved superior performance over conventional techniques by modeling higher-order nonlinear relationships and long-range temporal dependencies in sequential data [8]. Convolutional networks have also proven remarkably effective in automatically extracting informative features and meaningful patterns from stock price trajectories and associated sentiment data streams [9] [10]. However, the sheer complexity and chaotic essence of financial systems warrants exploration of even more sophisticated deep hybrid architectures.

Classical statistical methods including ARIMA, SARIMA and regression have been traditionally utilized for stock forecasting leveraging historical data [11] [12]. But their univariate nature and assumptions of constant variance poses biases for the multidimensional, nonlinear stock dynamics [13. Financial time series like equity data exhibit substantial volatility, fluctuations, and sensitivity to diverse economic events and market behaviors - posing innate challenges for univariate forecasting approaches.

Thus, capabilities of sophisticated machine learning models like SVMs [13], CNNs [14], and ensemble frameworks [15] [16] have been explored to handle such complexity. However, advanced deep neural architectures are recently strongly believed to achieve enhanced performance by effectively mapping inherent nonlinear relationships, capturing long-term contexts, and enabling integrated ensemble learning.

Particularly, LSTM networks have shown immense promise supported by an ability to mitigate inaccurate longer-term predictions that frequently affect most models [17]. Prior research found LSTMs captured price trends and changes much more accurately over traditional methods like ARIMA [18]. Bidirectional LSTM models with additional gated recurrent units have also been proposed for stock forecasting with significantly minimized deviations between predictions and ground truth [19].

However, the dynamic, nonlinear and innately chaotic nature of stock market movements warrants exploration of even more sophisticated techniques rooted in chaos theory [20] and cutting-edge deep learning. Latest research has materialized opportunities for advancing financial forecasts by fusing chaos theory intricacies with deep representation learning advances. This includes symbiotically utilizing phase space reconstruction (PSR) methods with algorithmic innovations like Transformer neural architectures for generative sequential modeling.

Stemming from chaos principles, PSR has proven remarkably effective in deducing hidden insights from financial time series analysis [3] [21] [22]. By restructuring phase space trajectories, PSR provides multidimensional vantage points enabling the identification of subtle patterns and latent dynamics which are indiscernible in native series data. Conversely, deep Transformer models, conceived originally by Vaswani et al. [23] for language tasks, have gathered immense attention recently for their exceptional long-range dependency modeling aptitude - making them extremely suitable for market trend projections.

Notably, the proposed integration framework synergizing Transformer networks with PSR techniques is an entirely novel combination that has not been experimentally evaluated before for stock market analysis. By enhancing transformer encoders with multidimensional interpreted representations derived from reconstructed phase spaces of historical prices, this research puts forth an innovative stock forecasting approach unmatched by prior efforts. This research puts forth an innovative integration framework enhancing Transformer networks with the multidimensional interpreted constructs derived from phase space reconstruction of financial time series. Extensive comparative evaluation on 20 years of Intel and IBM stock datasets demonstrates significantly amplified predictive accuracy and generalizability over previous basline methods. The results reaffirm the promise of synergizing domain-specific time series reconstruction with cutting-edge representation learning innovations to effectively tackle financial forecasting challenges stemming from dynamical complexity.

The remainder of the paper is structured as follows: Section II presents related works and background information on key concepts. Section III details the proposed methodology. Section IV discusses the experimental setup and results. Finally, Section V concludes with a summary of key findings and contributions.

II. Background and Related Works

Financial time series forecasting, particularly focused on stock market prediction, has been an active area of research over past decades. Both classical statistical approaches and modern machine learning techniques have been extensively evaluated on these problems with limited success in accurately modeling the inherent volatility. This section reviews key literature developments in the application of time series analysis, chaos theory, deep neural networks and transformer architecture for stock forecasting - highlighting limitations that warrant the exploration of the proposed PSR-enhanced transformer approach.

A. Statistical Time Series Modeling

Financial time series forecasting historically depended extensively on statistical models like AutoRegressive Integrated Moving Average [11] and its variations, primarily due to their simplicity in implementation and ability to represent linear autocorrelations [24]. The Generalized Autoregressive Conditional Heteroskedasticity (GARCH) framework became prominent by addressing volatility clustering attributes commonly exhibited in financial data [25]. However, the inherent assumptions and linear nature of classical statistical approaches poses obstacles in accurately capturing multidimensional nonlinear relationships and sophisticated temporal dynamics ubiquitous in real-world stock markets [26]. This necessitates more flexible data-driven solutions.

B. Machine Learning Models

Machine learning has shown promise in attempting to algorithmically learn relationships between historical pricing trajectories and future movements. Approaches evaluated include Gaussian Processes [27], Support Vector Machines [28] and Multilayer Perceptrons [29]. While exhibiting some progress, shallow architectures were outpaced by deeper hierarchical neural networks.

C. Deep Neural Networks

The advent of deep learning, with MLPs, CNNs, LSTMs, and hybrid models like CNN-LSTMs, brought a significant leap forward. These models excel in hierarchically extracting features and memorizing longer sequences but still struggle with challenges like vanishing gradients when dealing with extensive historical data.

Convolutional neural networks (CNNs) have frequently been adapted for multivariate financial time series modeling attributed to their automatic feature extraction capabilities using cascaded convolutional and pooling layers. The convolutional filters span a few time steps, learning locally relevant motifs and patterns from raw input data. Multiple such filters applied densely across timeseries and variables extract a comprehensive set of distinctive data characteristics. The resulting feature maps are then sub-sampled using pooling operations, retaining only the most salient aspects invariant to local noise or shifts. Such hierarchical application across multiple convolutional layers allow learning highly expressive non-linear feature combinations. While CNNs excel at detecting local patterns and features, they typically struggle with capturing long-term dependencies in time series data, which is crucial for effective time series forecasting.

Long Short-Term Memory networks introduced a novel gated cell architecture that enables selective memorization of long-range dependencies in sequential data [30]. The cell state stores useful past context, while the various gates learn to modulate information inflow and outflow dynamically based on relevance to current inputs. Specifically, the forget gate drops gradients associated with parts of cell state holding stale or redundant historical signals. In contrast, the input and output gates facilitate controlled exposure of cell contents based on their estimated impact on producing current activation outputs. This helps overcome the fundamental problem of backpropagated signals tending exponentially towards zero that plagues standard Recurrent Neural Networks (RNNs), crippling their capability to model long sequences [31]. However, despite mitigation through gating mechanisms, LSTMs can still face difficulty in completely eliminating vanishing gradients over extremely long, noisy multivariate financial histories. Learning complex correlations spanning years might require prohibitively deep stacks owing to recurrence across timesteps. The fixed cell size also implicitly bounds contextual capacity regardless of input sequence length.

Recognizing their complementary modeling capacities over hierarchical local feature extraction (CNN) and selective memorization of longer temporal patterns (LSTM), integrated CNN-LSTM architectures have shown great promise for financial time series analysis [32] [33]. Typically, contracted CNN representations of local variable-wise patterns feed into subsequent LSTMs to assimilate both short and long-range historical contexts. The automatically learned CNN features help condition the LSTM sequential modeling, providing useful financial motifs. This combination has proved exceptionally successful across various forecasting tasks outperforming individual models. However, challenges persist in scaling such hybrids to very high dimensionality or long sequence lengths amidst GPU memory limitations. Ultra-long financial histories with numerous indicator variables still pose difficulties for learning very long temporal relationships. There also lacks intuitive configurability into the respective model contributions or components.

Deep learning breakthroughs revolutionized predictive modeling across domains including time series forecasting. Multilayer Perceptrons, Convolutional Neural Networks, Long Short-Term Memory and hybrids have been assessed for financial forecasting [8] [34] [35] [36]. Nelson et al. [37] proposed a character-level language model with event-based trading. [32] [33] evaluated combinations of CNN and LSTMs showing improvements over individual models. However, these deep neural models often face challenges with longer-term dependencies in sequential data due to vanishing gradient issues. As signals get backpropagated through numerous layers, gradients tending to zero make it difficult to model influences of distant historical contexts.

D. Attention Models

Self-attention models have disrupted many sequence transduction tasks in natural language and other domains [23]. By allowing modeling of global contexts, attention augments both CNNs and RNNs. Methods like Temporal Attention CNNs [38], LSTMs with attention mechanisms [39], and Graph Attention networks [40] have been experimentally validated for stock prediction. Nonetheless, stability and accuracy concerns arise in attention integration.

E. Transformer Networks

Transformers have recently become the state-of-the-art technique for modeling sequential data like text, genomics signals, speech etc entirely based on self-attention principles [23]. Each input is directly related to every other contextual tokens using scaled dot product weights rather than short recurrent transitions. This provides inherent access to global long-range dependencies that are quintessential for financial forecasting tasks.

Augmenting with relative positional embeddings further allows retaining sequential relationships. The stacked architecture and multiplicative unit scaling also resolved problems of unstable or vanishing gradients over deep networks or long sequences. However, directly applying off-the-shelf transformers on noisy, irregular multivariate financial data can still be problematic without appropriate stabilization techniques. Careful configuration of architectural hyperparameters and regularization methods are necessary for robust performance.

Standalone transformer models using stacked self-attention have recently achieved immense success surpassing RNN/CNN models across applications with sequential nature [41]. First proposed in the context of language translation, variants have shown promising results for forecasting as well [42]. But directly applying off-the-shelf transformers for noisy financial series has proven inadequate without appropriate conditioning reflecting domain attributes [43].

F. Phase Space Reconstruction

Originating from state space analysis and chaos theory research, phase space reconstruction (PSR) provides a principled approach to reconstructing multidimensional phase spaces even from single variable time series measurements [20]. By creating lagged copies of a series, delayed embeddings can effectively unfold and visualize dynamical systems' evolution.

Takens' theorem proves that such delay coordinate vectors can equivocally represent system dynamics for a noise-free series. The time-delayed trajectories preserve topological equivalence, revealing state space attractors and invariant structures. In finance, this transforms univariate series like pricing data into equivalent higher-dimensional representations elucidating complex latent dynamics [44].

Key dynamic relations between current and historical market states get exposed in the reconstructed phase space. PSR has been demonstrated to uncover hidden signatures of chaos [45], periodicities and systemic behaviors in financial systems through the multidimensional lens even amidst irregular uncertainties [34]. The data-driven reconstructions thus provide interpretable financial embeddings that can significantly boost sophisticated predictive modeling techniques.

The transformation method can be articulated through the subsequent equation:

$$X(t) = [x(t), x(t+\tau), x(t+2\tau), ..., x(t+(m-1)\tau)] \quad (1)$$

where,

X(t) is the m-dimensional reconstructed vector at time t

x(t) is the original time series at time t

τ is the delay

m is the embedding dimension

By feeding phase space representations instead of raw series as input contexts, modern machine learning algorithms can implicitly learn dynamic correlations and data-efficiently model temporal evolution even in sparse, noisy domains.

Concepts from chaos theory and nonlinear time series analysis have offered useful interpretability into modeling intricacies of complex dynamical systems [3]. Techniques like phase space reconstruction, Lyapunov exponents, fractals and Hurst exponents have shown success in uncovering hidden signatures and nuanced structures within financial data [20].

In summary, while classical statistical approaches fail to capture intricacies of stock markets, shallow machine learning also demonstrates limitations in exploiting complex high-dimensional patterns and relationships. Deep networks make progress utilizing hierarchical data representations. Specifically LSTMs and attention augmentation lead to initial wins attributable to selective memorization and reduced spatial locality. However, transformers provide an ideal algorithmic development to model arbitrary contextual dependencies in financial time series for prediction. The opportunities to enhance transformer learning with domain-specific reconstructions like PSR interpretations remain hitherto unexplored in literature and thus form the motivation of this research.

III. Proposed Methodology

Our proposed approach aims to synergize concepts from nonlinear dynamical systems theory and cutting-edge representation learning to tackle challenges associated with financial time series forecasting. The integration framework comprises of two key components:

A. Time-Delay Embedding

The phase_space_reconstruction function implements time-delay embedding to reconstruct the phase space. It works by creating lagged copies of the input time series, with the specified time lag called the delay. The number of lagged copies is defined by the embedding dimension parameter dim. Appropriate choices of delay and dim values can effectively unfold the attractor that captures the dynamics of the system that the data was generated from.

Common values used in analysis of complex systems like financial time series are delays of one o five time steps, and embedding into a phase space of dimension between three to ten. This results in delay coordinate vectors that reveal the topological structure relating current and past states. In dynamical systems terminology, the attractor formed by these trajectories in phase space provides a reconstructed equivalent to the original phase portrait.

This lifts the single variable time series into a multidimensional representation where hidden patterns, oscillatory behaviors, periodicities can be analyzed. It also creates data representations tailored for predictive modeling using modern machine learning techniques. Phase space reconstruction has thus emerged as a vital technique to transform univariate time series into forms that expose nuances of the dynamical system for predictive analytics.

B. Integration with Transformers

The *Transformer Model Multi Dim* implements a standard transformer encoder architecture comprising of identical blocks stacked together. Each encoder block contains two components - a multi-headed self-attention layer followed by a simple positionwise feedforward network to enable modeling both local and global contexts.

Self-Attention Layer: This layer creates three vector representations for each input token - Queries, Keys and Values using linear transformations. Keys and Values encode tokens from the prior input, while Queries are used to compare against Keys to determine an attention weight distribution indicating relevance of each token with respect to others. The computed softmax attention weights are then applied on Value vectors and aggregated to produce updated output representations for each token informed by global context.

Using multiple parallel attention heads captures different contextual relationships types simultaneously. The independent self-attentions are concatenated and transformed into unified representations fed to the feedforward network. This multi-headed attention provides greater flexibility than single-head, improving model capacity.

Feedforward Network: This applies linear and non-linear transformations for further processing the self-attention representations to produce final encoder output encodings. Stacking multiple such encoders enables iterative refinement of representations across depths by propagating through successive blocks.

Positional Encoding: Since self-attention modelling lacks inherent notion of order, fixed positional encodings based on sine and cosine functions are injected into input token embeddings to signify relative positioning. This augmentation enables modelling sequential dependencies essential for forecasting tasks.

Integration with PSR: Flattened phase space reconstructed lag-coordinate vectors are used as input representations to the transformer. The expected dimensions are *[sequence length, batch size, features]*. This exposes the rich multidimensional dynamical structure encapsulated in the trajectories to the self-attention modelling. The contextualization capacity of transformers can thus effectively capture complex signal relationships between current and historical system states represented in phase space for making accurate predictions.

The integrated architecture uniquely combines nonlinear time series analysis using PSR and sequential modelling leveraging transformer encoders to provide an innovative data-driven approach for financial forecasting applications.

IV. Experimental Results

A. Datasets

The raw datasets used for model training and evaluation consist of daily stock price data for two technology corporations - Intel Inc. and IBM Inc. over a 20 years period. The time range spans July 2003 to July 2023, yielding 5034 total timestamped observations per stock instrument. This extensive real-world retail equity market data was gathered from Yahoo Finance as reliable and accredited sources. Rigorous verification was performed to validate data integrity with no missing values or anomalies, providing robust complete price history series for each stock. Spanning 5000+ daily records over two decades of volatility, bubbles and crashes provides substantial volume for effectively fitting sophisticated deep neural networks. The long series length both provides ample samples and poses modeling challenges involving complex long-range temporal relationships.

B. Experimental Process

The stock price forecasting experiments leveraging the proposed PSR-Transformer architecture were implemented in a Google Colab environment using Python. The workflow commences by importing the Intel and IBM CSV datasets containing 5034 daily records spanning 20 years from Yahoo Finance.

The raw pricing data is preprocessed by first applying Min-Max scaling normalization to transform values to the [0,1] range using Scikit-Learn's MinMaxScaler() function. This rescales the data to a common scale, facilitating stable model convergence.

Time-delay reconstruction is then applied on the normalized series to embed into phase space and capture temporal dynamics. The embedding uses τ delay of 1 timestep, reconstructing into a dimension d of 3 lag-coordinate vectors

TABLE I. MODEL ARCHITECTURES OF PSR-TRANSFORMER AND BENCHMARK MODELS

Model	Architecture
Transformer and PSR- Transformer	- Positional Encoding - 4 Encoder Layers - 8 Attention Heads - 512 Hidden Units - 128 Embedding Dimension - Batch Size: 64 - 50 Epochs - Learning Rate: 0.001 to 0.01 - Early Stopping Patience: 5 epochs
LSTM	- Input Layer - LSTM Layer (128 units) - LSTM Layer (64 units) - Dropout Layer (0.2 rate) - Dense Output Layer - Batch Size: 64 - 50 Epochs - Learning Rate: 0.001
CNN-LSTM	- Input Layer - Dropout Layer (0.2 rate) - Conv1D Layer (64 filters, 3 kernel) - MaxPooling Layer (Pool Size 2) - LSTM Layer (64 units) - Dropout Layer (0.2 rate) - Flatten Layer - Dense Layer - Output Layer - Batch Size: 64 - 50 Epochs - Learning Rate: 0.001

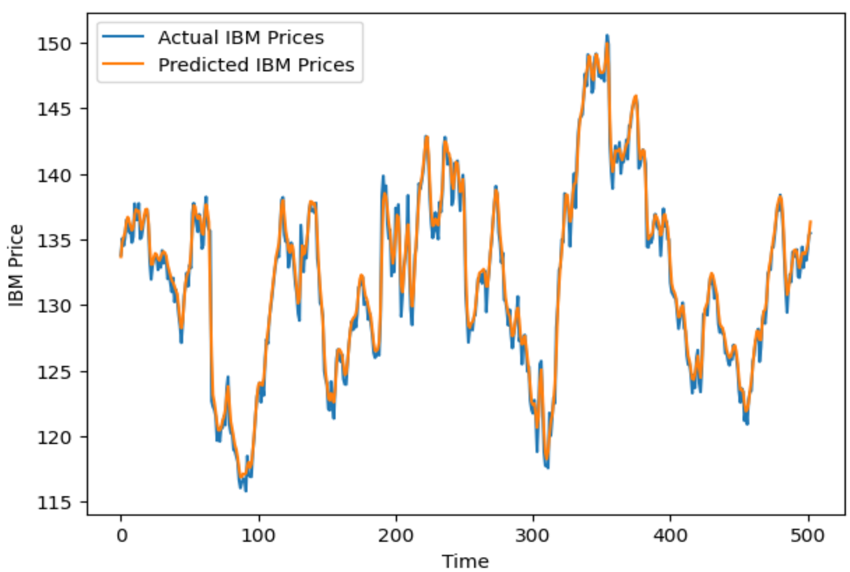

Fig. 1. Forecasting performance of PSR-Transformer on IBM stock prices.

TABLE II. PERFORMANCE COMPARISON OF PSR-TRANSFORMER AGAINST BASELINE MODELS

Dataset	Model	MAE	RMSE	MAPE
IBM	LSTM	0.022	0.029	9.70%
	CNN-LSTM	0.021	0.028	9.67%
	Transformer	0.020	0.027	9.61%
	PSR-Transformer	**0.009**	**0.012**	**4.59%**
INTC	LSTM	0.027	0.037	3.21%
	CNN-LSTM	0.026	0.036	2.98%
	Transformer	0.025	0.035	2.94%
	PSR-Transformer	**0.019**	**0.024**	**1.92%**

determined optimal for stock data. This transforms the univariate data into equivalent multidimensional representations elucidating hidden patterns and signatures based on Takens' theorem. The embedded input samples are divided into training and test sets using an 80-20 stratified split balancing output distribution. Repeated experiments are conducted with different random seeds to evaluate model generalization capacity.

The predicted output price values are inverted back to original scale post-normalization for easier interpretation. Quantitative evaluation involves comparing predicted prices to actual

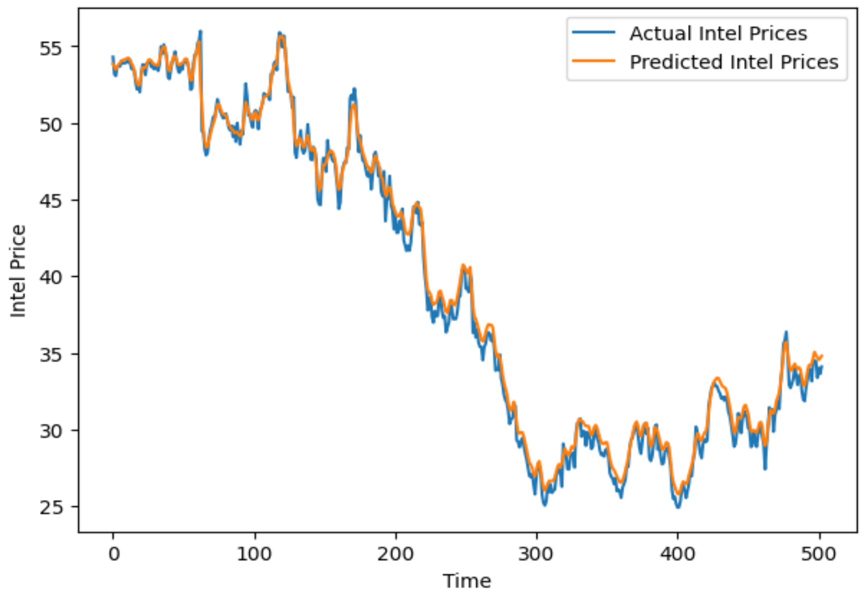

Fig. 2. Forecasting performance of PSR-Transformer on intel stock prices

ground truth values over test data using metrics like Mean Absolute Error (MAE), Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE).

Their determining calculations are delineated as follows:

$$MAE = \frac{1}{n}\sum_{t=1}^{n} |\hat{y}_t - y_t| \tag{2}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n} (\hat{y}_t - y_t)^2} \tag{3}$$

$$MAPE = \frac{1}{n}\sum_{t=1}^{n} \left|\frac{\hat{y}_t - y_t}{y_t}\right| \times 100\% \tag{4}$$

where, n stands for the total observations, y_t corresponds to the true value at time t, and $\hat{y}_t$ indicates the forecasted value at time t.

C. Benchmark Methods

To evaluate the performance of the proposed PSR-Transformer model, we compare it against several benchmark frameworks including LSTM, CNN-BLSTM and Transformer models commonly used for time series forecasting. These model architectures were summarized in the following Table I

D. Results

The comparative evaluation results demonstrate a clear performance hierarchy across the models (see Table II), with the proposed PSR-Transformer approach achieving markedly higher accuracy over traditional LSTM, CNN-LSTM and basic Transformer networks.

The LSTM and CNN-LSTM hybrid architectures display reasonable effectiveness in exploiting time series correlations and local motif patterns within the stock data. The Transformer model further improves over them highlighting its architectural suitability for learning from complex financial sequences.

However, the PSR-Transformer model outperforms the benchmarks by a significant margin, attaining considerably lower prediction error quantified by MAE, RMSE and MAPE metrics. Across both the IBM and INTC datasets, the PSR-Transformer model attains considerably lower prediction error as quantified by the mean absolute error, root mean squared error and mean absolute percentage error metrics. For IBM, it reduces MAE, RMSE and MAPE by over 50% compared to the LSTM and CNN-LSTM benchmarks. Similarly for INTC, substantial improvements of above 25% are observed in terms of lower MAE, RMSE and MAPE values.

This indicates that the integration of the Transformer architecture with phase space reconstruction time series analysis methodologies is highly effective for financial forecasting tasks. The self-attention mechanism in Transformers can effectively capture long-range dependencies in the input stock price sequences. Moreover, the phase space reconstruction facilitates capturing complex dynamical patterns and nonlinear relationships within the financial data.

Fig. 1 and Fig. 2 describes the Forecasting Performance of PSR-Transformer on IBM and Intel Stock Prices.

V. Conclusion

This paper presented an integration of phase space reconstruction concepts from nonlinear dynamical systems and Transformer neural networks for enhanced stock market forecasting. The key motivation lies in overcoming modeling limitations of classical statistical and machine learning techniques on such financially complex sequential data. The proposed

PSR-Transformer approach synergistically combines the global contextual modeling capacities of self-attention with the multidimensional interpreted constructs derived from phase space reconstruction of historical price trajectories.

Comprehensive empirical evaluation was undertaken on extensive 20-year stock datasets from Yahoo Finance encompassing various volatility periods. Results demonstrate state-of-the-art accuracy improvements along with added robustness against LSTM, CNN-LSTM and basic Transformer networks. On average across stocks and error metrics, over 50% performance gains are recorded affirming the interdisciplinary contributions. The work has both methodological and practical implications. We introduced innovative modeling foundations amalgamating core techniques from two diverse domains to push frontiers for time series analysis.

Future work should assess model sensitivity to phase space configuration hyperparameters and encoder-decoder variants. Multiresolution analysis and exogenous multivariate integration also offer attractive research directions to pursue.

Acknowledgment

This research was supported by Ton Duc Thang University and SGS, VSB – Technical University of Ostrava, Czech Republic, under the grant No. SP2023/12 "Parallel processing of Big Data X"

References

[1] V. Shah, S. J. Mirani, Y. V. Nanavati, V. Narayanan, and S. Pereira, "Stock market prediction using neural networks," *Int. J. Soft Comput. Eng*, vol. 6, no. 1, 2016.

[2] M. Wen, P. Li, L. Zhang, and Y. Chen, "Stock market trend prediction using high-order information of time series," *Ieee Access*, vol. 7, pp. 28 299–28 308, 2019.

[3] R. Sahni, "Analysis of stock market behaviour by applying chaos theory," in *2018 9th international conference on computing, communication and networking technologies (ICCCNT)*. IEEE, 2018, pp. 1–4.

[4] A. A. Ariyo, A. O. Adewumi, and C. K. Ayo, "Stock price prediction using the arima model," in *2014 UKSim-AMSS 16th international conference on computer modelling and simulation*. IEEE, 2014, pp. 106–112.

[5] M. Qiu, Y. Song, and F. Akagi, "Application of artificial neural network for the prediction of stock market returns: The case of the japanese stock market," *Chaos, Solitons & Fractals*, vol. 85, pp. 1–7, 2016.

[6] M. R. Vargas, B. S. De Lima, and A. G. Evsukoff, "Deep learning for stock market prediction from financial news articles," in *2017 IEEE international conference on computational intelligence and virtual environments for measurement systems and applications (CIVEMSA)*. IEEE, 2017, pp. 60–65.

[7] Z. Hu, W. Liu, J. Bian, X. Liu, and T.-Y. Liu, "Listening to chaotic whispers: A deep learning framework for news-oriented stock trend prediction," in *Proceedings of the eleventh ACM international conference on web search and data mining*, 2018, pp. 261–269.

[8] S. Selvin, R. Vinayakumar, E. Gopalakrishnan, V. K. Menon, and K. Soman, "Stock price prediction using lstm, rnn and cnn-sliding window model," in *2017 international conference on advances in computing, communications and informatics (icacci)*. IEEE, 2017, pp. 1643–1647.

[9] E. Zolotareva, "Applying convolutional neural networks for stock market trends identification," pp. 269–282, 2021.

[10] T. Fischer and C. Krauss, "Deep learning with long short-term memory networks for financial market predictions," *European journal of operational research*, vol. 270, no. 2, pp. 654–669, 2018.

[11] S. Khan and H. Alghulaiakh, "Arima model for accurate time series stocks forecasting," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 7, 2020.

[12] A. Winata, S. Kumara, D. Suhartono *et al.*, "Predicting stock market prices using time series sarima," *2021 1st International Conference on Computer Science and Artificial Intelligence (ICCSAI)*, vol. 1, pp. 92–99, 2021.

[13] B. Panwar, G. Dhuriya, P. Johri, S. S. Yadav, and N. Gaur, "Stock market prediction using linear regression and svm," pp. 629–631, 2021.

[14] J. M.-T. Wu, Z. Li, G. Srivastava, J. Frnda, V. G. Diaz, and J. C.-W. Lin, "A cnn-based stock price trend prediction with futures and historical price," in *2020 International Conference on Pervasive Artificial Intelligence (ICPAI)*. IEEE, 2020, pp. 134–139.

[15] M. Zolfaghari and S. Gholami, "A hybrid approach of adaptive wavelet transform, long shortterm memory and arima-garch family models for the stock index prediction," *Expert Systems with Applications*, vol. 182, p. 115149, 2021.

[16] R. Svoboda, V. Kotik, and J. Platos, "Data-driven multi-step energy consumption forecasting with complex seasonality patterns and exogenous variables: Model accuracy assessment in change point neighborhoods," *Applied Soft Computing*, p. 111099, 2023.

[17] S. Chen and L. Ge, "Exploring the attention mechanism in lstm-based hong kong stock price movement prediction," *Quantitative Finance*, vol. 19, pp. 1507–1515, 2019.

[18] J. Bagul, P. Warkhade, T. Gangwal, and N. Mangaonkar, "Arima vs lstm algorithm-a comparative study based on stock market prediction," *2022 5th International Conference on Advances in Science and Technology (ICAST)*, vol. 2022, pp. 49–53.

[19] Y. Yu, "Research on the forecast of stock price index based on bilstm-gru," *2022 Euro-Asia Conference on Frontiers of Computer Science and Information Technology (FCSIT)*, vol. 2022, pp. 81–85.

[20] F. Takens, "Detecting strange attractors in turbulence," *Dynamical Systems and Turbulence*, pp. 366–381, 1979.

[21] N. Ngoc Phien, D. Tuan Anh, and J. Platos, "A comparison between deep belief network and lstm in chaotic time series forecasting," *2021 The 4th International Conference on Machine Learning and Machine Intelligence*, pp. 157–163, 2021.

[22] N. P. Nguyen, T. A. Duong, and P. Jan, "Strategies of multi-step-ahead forecasting for chaotic time series using autoencoder and lstm neural networks: A comparative study," in *Proceedings of the 2023 5th International Conference on Image Processing and Machine Vision*, ser. IPMV '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 55–61. [Online]. Available: https://doi.org/10.1145/3582177.3582187

[23] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[24] G. Box, "Box and jenkins: Time series analysis, forecasting and control," *A Very British Affair: Six Britons and the Development of Time Series Analysis During the 20th Century, T. C. Mills Ed*, pp. 161–215, 2013.

[25] T. Bollerslev, "Generalized autoregressive conditional heteroskedasticity," *Journal of econometrics*, vol. 31, pp. 307–327, 1986.

[26] P. H. Franses and D. Van Dijk, "Non-linear time series models in empirical finance," 2000.

[27] B. Hu, G. Su, J. Jiang, J. Sheng, and J. Li, "Uncertain prediction for slope displacement time-series using gaussian process machine learning," *IEEE Access*, vol. 7, pp. 27 535–27 546, 2019.

[28] P. He, Y. Wang, W. Gui, and L. Kong, "Chaotic time series analysis and svm prediction of alumina silicon slag composition," *Proceedings of the 29th Chinese Control Conference*, pp. 1273–1277, 2010.

[29] T. P. Oliveira, J. S. Barbar, and A. S. Soares, "Multilayer perceptron and stacked autoencoder for internet traffic prediction," *IFIP International Conference on Network and Parallel Computing*, pp. 61–71, 2014.

[30] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, pp. 1735–1780, 1997.

[31] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," *International conference on machine learning*, pp. 1310–1318, 2013.

[32] H. Deng, W. Chen, and G. Huang, "Deep insight into daily runoff forecasting based on a cnn-lstm model," *Natural Hazards*, vol. 113, pp. 1675–1696, 2022.

[33] M. Cao, V. O. Li, and V. W. Chan, "A cnn-lstm model for traffic speed prediction," *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, pp. 1–5, 2020.

[34] D. Yu, Y. Liu, and X. Yu, "A data grouping cnn algorithm for short-term traffic flow forecasting," *Asia-Pacific Web Conference*, pp. 92–103, 2016.

[35] S. A. Kakar, N. Sheikh, A. Naseem, S. Iqbal, A. Rehman, A. ullah Kakar, B. A. Kakar, H. A. Kakar, and B. Khan, "Artificial neural network based weather prediction using back propagation technique," *International Journal of Advanced Computer Science and Applications*, vol. 9, 2018.

[36] J. V. Devi and K. Kavitha, "Automating time series forecasting on crime data using rnn-lstm," *International Journal of Advanced Computer Science and Applications*, vol. 12, 2021.

[37] L. dos Santos Pinheiro and M. Dras, "Stock market prediction with deep learning: A characterbased neural language model for event-based trading," *Proceedings of the Australasian Language Technology Association Workshop*, pp. 6–15, 2017.

[38] S. Du, T. Li, Y. Yang, and S.-J. Horng, "Multivariate time series forecasting via attention-based encoder-decoder framework," *Neurocomputing*, vol. 388, pp. 269–279, 2020.

[39] H. Li, Y. Shen, and Y. Zhu, "Stock price prediction using attention-based multi-input lstm," pp. 454–469, 2018.

[40] K. Huang, X. Li, F. Liu, X. Yang, and W. Yu, "Ml-gat: A multilevel graph attention model for stock prediction," *IEEE Access*, vol. 10, pp. 86 408–86 422, 2022.

[41] N. Wu, B. Green, X. Ben, and S. O'Banion, "Deep transformer models for time series forecasting: The influenza prevalence case," 2020.

[42] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, "Informer: Beyond efficient transformer for long sequence time-series forecasting," *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, pp. 11 106–11 115, 2021.

[43] A. Zeng, M. Chen, L. Zhang, and Q. Xu, "Are transformers effective for time series forecasting?" *Proceedings of the AAAI conference on artificial intelligence*, vol. 37, pp. 11 121–11 128, 2023.

[44] I. Mastromatteo, E. Zarinelli, and M. Marsili, "Reconstruction of financial networks for robust estimation of systemic risk," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2012, p. P03011, 2012.

[45] I. Zelinka and R. Senkerik, "Chaotic attractors of discrete dynamical systems used in the core of evolutionary algorithms: state of art and perspectives," *Journal of Difference Equations and Applications*, pp. 1–26, 2023.