AppliedMath

MDPI

Article

GRU-Based Stock Price Forecasting with the Itô-RMSProp Optimizers

Mohamed Ilyas El Harrak [1], Karim El Moutaouakil [2,*], Nuino Ahmed [3], Eddakir Abdellatif [1] and Vasile Palade [4,*]

[1] Research and Studies Laboratory in Management, Entrepreneurship and Finance, National School of Business and Management, Sidi Mohamed Ben Abdellah University, Fez 30000, Morocco; mohamedilyas.elharrak@usmba.ac.ma (M.I.E.H.); abdellatif.eddakir@usmba.ac.ma (E.A.)
[2] Laboratory of Mathematics and Data Science, Multidisciplinary Faculty of Taza, Sidi Mohamed Ben Abdellah University, Fez 30000, Morocco
[3] Higher Institute of Nursing Professions and Health Techniques (ISPITS), Tétouan 93000, Morocco; nuinoahmed@ispits-tetouan.ma
[4] Centre for Computational Science and Mathematical Modelling, Coventry University, Priory Road, Coventry CV1 5FB, UK
* Correspondence: karim.elmoutaouakil@usmba.ac.ma (K.E.M.); vasile.palade@coventry.ac.uk (V.P.)

Abstract

This study introduces Itô-RMSProp, a novel extension of the RMSProp optimizer inspired by Itô stochastic calculus, which integrates adaptive Gaussian noise into the update rule to enhance exploration and mitigate overfitting during training. We embed this optimizer within Gated Recurrent Unit (GRU) networks for stock price forecasting, leveraging the GRU's strength in modeling long-range temporal dependencies under nonstationary and noisy conditions. Extensive experiments on real-world financial datasets, including a detailed sensitivity analysis over a wide range of noise scaling parameters (ε), reveal that Itô-RMSProp-GRU consistently achieves superior convergence stability and predictive accuracy compared to classical RMSProp. Notably, the optimizer demonstrates remarkable robustness across all tested configurations, maintaining stable performance even under volatile market dynamics. These findings suggest that the synergy between stochastic differential equation frameworks and gated architectures provides a powerful paradigm for financial time series modeling. The paper also presents theoretical justifications and implementation details to facilitate reproducibility and future extensions.

Keywords: GRU; stock price forecasting; Itô calculus; RMSProp; deep learning

check for updates

Academic Editor: Thomas Woolley

Received: 18 September 2025
Revised: 21 October 2025
Accepted: 28 October 2025
Published: 2 November 2025

Citation: El Harrak, M.I.; El Moutaouakil, K.; Ahmed, N.; Abdellatif, E.; Palade, V. GRU-Based Stock Price Forecasting with the Itô-RMSProp Optimizers. *AppliedMath* **2025**, *5*, 149. https://doi.org/10.3390/appliedmath5040149

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Recurrent Neural Networks (RNNs) have been a foundational architecture for modeling sequential data in fields such as natural language processing, speech recognition, and time series forecasting [1,2]. Elman (1990) and Rumelhart et al. (1986) introduced early RNN frameworks for learning temporal patterns and sequential dependencies in data. However, standard RNNs suffer from vanishing and exploding gradient problems [3], as highlighted by Bengio et al. (1994), who formally analyzed the gradient instability in deep recurrent models, limiting their ability to capture long-term dependencies. To overcome these issues, Hochreiter and Schmidhuber introduced the Long Short-Term Memory (LSTM) architecture [4], which introduced gating mechanisms to preserve long-range information and mitigate gradient decay during training.

Building upon the LSTM, the GRU was proposed as a simpler alternative, merging the forget and input gates into a single update gate and combining the cell and hidden states, thus reducing parameter count and training complexity [5]. Cho et al. (2014) designed

GRUs to balance computational efficiency and temporal learning performance. Both architectures have since become widely used in diverse sequential modeling tasks [6–8]. These studies demonstrated the versatility and effectiveness of gated recurrent networks in speech, language, and time-series contexts.

Training these deep recurrent networks efficiently necessitates sophisticated optimization algorithms. Early methods such as SGD often suffer from slow convergence and sensitivity to hyperparameters [9]. Bottou (2010) emphasized the limitations of basic stochastic gradient descent for large-scale learning. Adaptive optimizers like RMSProp [10], Adagrad [11], Adam [12], Nadam [13], and variants have become popular for their ability to adapt learning rates dynamically and accelerate convergence [14,15]. These works collectively shaped modern adaptive gradient methods, enabling faster and more stable training across deep architectures. Recent advances also explore noise-injection techniques inspired by stochastic differential equations to improve generalization and escape saddle points [16,17]. Li et al. (2019) and Jin et al. (2017) showed that introducing controlled stochastic noise helps optimizers avoid sharp minima and enhance generalization.

Numerous variants and enhancements of LSTM and GRU have been proposed to further improve performance, including bidirectional models [18,19], which process sequences in both forward and backward directions for richer temporal context, residual and highway connections [20,21], which facilitate gradient flow in deep recurrent stacks, attention mechanisms [22,23], that dynamically weight input relevance to enhance long-term dependency modeling, and stacked/deep architectures [24]. Pascanu et al. (2013) discussed how deeper recurrent structures can capture hierarchical temporal features. These variants have demonstrated improved capacity to model complex temporal dependencies across diverse applications [25,26]. Sak et al. (2014) and Liu et al. (2016) empirically confirmed the scalability and accuracy benefits of such enhanced recurrent frameworks.

Despite these advances, challenges remain in training stable and generalizable recurrent networks, especially for noisy or nonstationary data such as financial time series [27,28]. Fischer and Krauss (2018) as well as Borovykh et al. (2017) illustrated the limitations of standard deep learning approaches when applied to highly volatile financial datasets. This motivates exploration of novel optimizers and training schemes that leverage theoretical insights from stochastic calculus, such as the Itô-RMSProp optimizer proposed in this work.

The Itô-RMSPropGRU method implements a comparative experiment between the standard RMSProp optimizer and an Itô-inspired variant, Itô-RMSProp, within a GRU-based forecasting framework for financial time series. The Itô-RMSProp optimizer extends the classical RMSProp update rule by incorporating an adaptive stochastic term derived from Itô stochastic differential equations. This stochastic component introduces Gaussian perturbations scaled by $\sqrt{\text{lr}}$, enabling controlled exploration of the loss landscape while maintaining stability in gradient updates. Both optimizers are evaluated on real stock price data (ticker: HD, 2015–2023), using normalized sequences of length 30 for training and validation. The models are trained under identical hyperparameter configurations to ensure fair comparison. Performance metrics, including Directional Accuracy (DA,%), Sharpe Ratio (SR), RMSE, MAE, and R^2, along with training and validation losses, are used to assess convergence behavior and generalization. The results are visualized through temporal plots of predicted versus true prices, highlighting the potential of Itô-RMSProp to achieve smoother and more robust convergence in noisy financial environments.

The main contributions of this work are as follows:

- We propose a stochastic differential equation (SDE)-inspired variant of RMSProp, termed Itô-RMSProp, specifically designed for training GRU networks in the context of stock price forecasting.

- We implement the Itô-RMSProp-GRU model and apply it to forecast the stock prices of well-known companies using real-world financial time series data.
- We perform a comprehensive empirical comparison between the proposed Itô-RMSProp-GRU and the classical RMSProp-GRU, showing that Itô-RMSProp improves predictive accuracy and generalization, especially in volatile market conditions.

The remainder of the paper is organized as follows: Section 2 provides the standard tools needed to build Itô-RMSProp. Section 3 presents the RMSProp optimizer to GRUs. Section 4 details the experimental setup and analyzes the results. Finally, Section 5 concludes the paper.

2. Standard Tools

Deep learning for time series forecasting relies on a combination of powerful optimization algorithms and sequence modeling architectures. In this section, we review two key components foundational to our approach: the RMSProp optimizer, which adaptively tunes learning rates to stabilize training in nonstationary settings, and the GRU, a streamlined recurrent neural network architecture effective for capturing temporal dependencies. Together, they form a robust framework for modeling complex sequential data.

2.1. RMSProp Optimizer

RMSProp is an adaptive gradient optimization method introduced by Tieleman and Hinton [10] to address the challenges of non-stationary objectives and vanishing/exploding gradients commonly encountered in training deep neural networks, especially recurrent architectures.

Given parameters θ_t at iteration t and the gradient of the loss $\mathcal{L}(\theta)$ with respect to θ_t denoted as

$$g_t = \nabla_\theta \mathcal{L}(\theta_t), \tag{1}$$

RMSProp maintains an exponentially weighted moving average of the squared gradients:

$$v_t = \beta v_{t-1} + (1-\beta) g_t^2, \tag{2}$$

where the decay factor $\beta \in [0.9, 0.99]$ controls the memory of past squared gradients.

The parameter update is then performed as:

$$\theta_{t+1} = \theta_t - \eta \frac{g_t}{\sqrt{v_t} + \varepsilon}, \tag{3}$$

where $\eta > 0$ is the base learning rate and $\varepsilon > 0$ is a small constant added for numerical stability to avoid division by zero.

RMSProp adaptively adjusts the learning rate for each parameter by normalizing the gradient by a running average of recent magnitudes of its squared gradients. This adaptive step size helps stabilize training on nonstationary and noisy objectives by dampening updates where gradients are large, automatically rescales parameters with large gradients to avoid excessively large steps, and improves convergence speed in RNNs and other deep architectures with sparse or varying gradient scales.

While RMSProp has demonstrated empirical success, theoretical convergence guarantees are subtle due to non-convexity and adaptive learning rates:

- Under assumptions of smoothness and bounded gradients, variants of RMSProp converge to stationary points [14].
- RMSProp can be viewed as a special case of adaptive gradient methods (including Adam [12]) with momentum-like effects on the squared gradient accumulation.

- Recent work [15] analyzes RMSProp's implicit bias and step size adaptation, providing partial guarantees on convergence rates under specific conditions.
- No universal guarantees exist for global optimality in deep learning, due to nonconvex loss surfaces.

In practice, RMSProp is typically used with default hyperparameters such as $\beta = 0.9$, $\varepsilon = 10^{-8}$, and a learning rate η tuned between 10^{-4} and 10^{-2}, depending on the task. When training recurrent architectures, gradient clipping is often applied to mitigate exploding gradients. RMSProp can also be combined with enhancements such as momentum or decoupled weight decay [29] to improve stability. Additionally, careful tuning of the learning rate schedule—using decay strategies or warm restarts—can significantly enhance convergence and overall performance.

While RMSProp offers stable and adaptive learning, particularly in recurrent architectures, it remains a deterministic method and can still converge to sharp or suboptimal local minima, especially in complex nonconvex landscapes. Its lack of inherent stochasticity limits exploratory behavior during optimization, potentially reducing generalization performance. These limitations motivate the development of enhanced optimizers, such as the proposed Itô-RMSProp, which incorporates noise inspired by Itô calculus to introduce controlled stochasticity and improve exploration during training.

Mathematical Formulation and Forecast Horizon Definition

We consider a multivariate time series of asset prices represented by the vector

$$y_t = [p_t^{(1)}, p_t^{(2)}, \ldots, p_t^{(n)}]^T \in \mathbb{R}^n, \tag{4}$$

where each element $p_t^{(i)}$ denotes the closing price of the ith asset at discrete time t. The forecasting problem is formally defined as learning a nonlinear mapping

$$\mathcal{F}_\theta : \{y_{t-T+1}, y_{t-T+2}, \ldots, y_t\} \mapsto \hat{y}_{t+h}, \tag{5}$$

parameterized by θ, where T represents the look-back window length and h the forecast horizon. The objective of training is to determine the optimal parameters θ^* that minimize the discrepancy between the true and predicted future values:

$$\theta^* = \arg\min_\theta \ \mathbb{E}[\mathcal{L}(y_{t+h}, \hat{y}_{t+h})]. \tag{6}$$

In our study, the recurrent dynamics of the GRU model act as a parametric approximation of the underlying stochastic process governing asset prices. The model captures both short-term dependencies and long-term temporal correlations through its hidden state evolution:

$$h_t = \Phi_\theta(h_{t-1}, y_t), \tag{7}$$

where $\Phi_\theta(\cdot)$ denotes the GRU's nonlinear state transition operator. This allows the model to approximate the conditional distribution $P(y_{t+h}|y_{t-T+1:t})$, providing a probabilistic interpretation of the forecasting mechanism.
The optimization process seeks to minimize the Mean Absolute Error (MAE) between observed and predicted values:

$$L = \frac{1}{N} \sum_{t=1}^{N} \|y_{t+h} - \hat{y}_{t+h}\|_1, \tag{8}$$

which corresponds to the reviewer's notation

$$f(|y_{T+h} - \hat{y}_{T+h}|), \tag{9}$$

and, when aggregated over multiple forecasting intervals,

$$f\left(\sum_{T_1}^{T_2} |y_{T+h} - \hat{y}_{T+h}|\right). \tag{10}$$

The MAE loss is robust to outliers and provides a direct measure of average prediction deviation, making it particularly suitable for financial series that exhibit heavy-tailed noise and volatility clustering.

To ensure temporal consistency and realistic evaluation, the dataset is partitioned using a rolling-window scheme: at each iteration, a training window of length T is used to forecast future prices for horizon h, typically set to one day ($h = 1$). This framework formalizes the sequential learning and evaluation process, ensuring that the model parameters evolve in a manner consistent with real-world forecasting scenarios.

2.2. GRU and Loss Function for Time Series Forecasting

RNNs are natural choices for sequential data modeling but suffer from the vanishing gradient problem [3,30], which hampers learning long-term dependencies. To address this, LSTM networks were introduced by Hochreiter and Schmidhuber [4], incorporating memory cells and gating mechanisms (input, forget, and output gates) to regulate the flow of information and gradients over time.

LSTMs have demonstrated significant success in various sequence modeling tasks [6,7]. However, their complex architecture results in a relatively large number of parameters, leading to increased computational cost and slower training times [8].

To mitigate these limitations, GRUs were proposed by Cho et al. [5] as a simpler alternative to LSTMs. GRUs combine the input and forget gates into a single update gate and merge the cell state and hidden state, thus reducing model complexity while maintaining the ability to capture long-term dependencies.

Empirical studies suggest that GRUs perform comparably to LSTMs on many tasks [31,32], sometimes with faster convergence and fewer parameters [33]. The gating mechanisms of GRUs allow them to mitigate vanishing gradients similarly to LSTMs but with a more streamlined architecture.

Moreover, various extensions and modifications of both LSTM and GRU architectures have been proposed to improve performance and efficiency [8,31,34,35], reflecting ongoing research to balance expressiveness and computational cost in recurrent models.

$$\begin{aligned}
\mathbf{z}_t &= \sigma(W_z \mathbf{x}_t + U_z \mathbf{h}_{t-1} + b_z) \quad \text{(update gate)}, \\
\mathbf{r}_t &= \sigma(W_r \mathbf{x}_t + U_r \mathbf{h}_{t-1} + b_r) \quad \text{(reset gate)}, \\
\tilde{\mathbf{h}}_t &= \tanh(W_h \mathbf{x}_t + U_h(\mathbf{r}_t \odot \mathbf{h}_{t-1}) + b_h) \quad \text{(candidate)}, \\
\mathbf{h}_t &= (1 - \mathbf{z}_t) \odot \mathbf{h}_{t-1} + \mathbf{z}_t \odot \tilde{\mathbf{h}}_t.
\end{aligned} \tag{11}$$

Here σ is the sigmoid activation function, and $\odot$ denotes element-wise multiplication.

GRUs combine the gating advantages of LSTMs [4] with fewer parameters, facilitating efficient learning of long-term dependencies.

In time series forecasting, the GRU is trained to predict future values based on historical data. At each time step t, the GRU outputs a prediction $\hat{y}_t$ (which may be a scalar or vector, depending on the task) from the hidden state $\mathbf{h}_t$ via an output layer:

$$\hat{y}_t = f_{\text{out}}(\mathbf{h}_t; \Theta_{\text{out}}), \tag{12}$$

where Θ_{out} are learnable parameters of the output layer (e.g., a fully connected layer). The training objective is to minimize the discrepancy between the predicted sequence $\hat{Y} = \{\hat{y}_t\}$ and the ground truth sequence $Y = \{y_t\}$ over the training set. Commonly used loss functions include:

- Mean Squared Error (MSE):

$$\mathcal{L}_{\text{MSE}}(\Theta) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{T_i}\sum_{t=1}^{T_i}\left(y_t^{(i)} - \hat{y}_t^{(i)}\right)^2, \tag{13}$$

  where N is the number of training sequences, T_i the length of the i-th sequence, and Θ the full set of parameters (GRU weights and output layer weights).
- Mean Absolute Error (MAE):

$$\mathcal{L}_{\text{MAE}}(\Theta) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{T_i}\sum_{t=1}^{T_i}\left|y_t^{(i)} - \hat{y}_t^{(i)}\right|. \tag{14}$$

The MSE loss corresponds to assuming Gaussian observation noise with fixed variance. Minimizing MSE aligns with maximizing the likelihood under a Gaussian noise model, making it a natural choice when errors are expected to be normally distributed and homoscedastic. MAE corresponds to Laplace-distributed noise assumptions and is more robust to outliers.

To minimize $\mathcal{L}$, gradients are computed via Backpropagation Through Time (BPTT) [36] which unrolls the recurrent network through time and backpropagates gradients through all time steps. GRUs alleviate vanishing gradients, enabling effective learning on longer sequences.

When training GRU models for time series forecasting, several practical considerations can enhance performance and stability. Longer input sequences allow the model to capture more temporal context but come at the cost of increased computation and a higher risk of gradient instability, so their length should be selected based on the data and domain. Regularization techniques such as dropout or weight decay are useful for preventing overfitting, particularly when data is limited. Normalizing input data using methods like MinMax scaling or standardization helps stabilize training. For multi-step forecasting, strategies such as teacher forcing or recursive prediction can be employed, each offering trade-offs between accuracy and stability. In cases of imbalanced time series or critical events, applying loss weighting can help emphasize important regions. Finally, evaluation should use a combination of metrics, such as RMSE, MAE, and R^2, to provide a well-rounded assessment of model performance.

The choice and careful tuning of the loss function critically affect GRU training outcomes. The MSE loss remains the most prevalent, but depending on data characteristics and outlier presence, alternative losses may be preferable.

3. Itô-RMSProp Optimizer to GRUs

Modern deep learning optimizers often draw inspiration from continuous-time stochastic processes to improve convergence, generalization, and robustness. In this section, we introduce a novel variant of RMSProp, denoted as Itô-RMSProp, which incorporates noise scaled according to Itô calculus to enhance optimization dynamics. Motivated by the interpretation of parameter updates as discretized SDEs, we explore how injecting appropriately scaled Gaussian noise can help recurrent models such as GRUs escape sharp minima and improve training stability. We begin by reviewing the fundamentals of Itô cal-

culus and its connection to stochastic optimization, then define the Itô-RMSProp algorithm and discuss its theoretical motivations and practical implications.

3.1. Itô Calculus and the Itô Derivative

The Itô calculus provides the mathematical foundation for modeling stochastic dynamics in both financial systems and optimization algorithms. In this section, we formalize the Itô stochastic differential equation (SDE), clarify its assumptions, and highlight its connection to stochastic gradient-based learning.

A scalar stochastic process $\{X_t\}_{t\geq 0}$ driven by Brownian motion (Wiener process) W_t is typically modeled by an Itô stochastic differential equation (SDE):

$$dX_t = \mu(t, X_t)\, dt + \sigma(t, X_t)\, dW_t, \tag{15}$$

where $\mu(t, X_t)$ denotes the drift term (deterministic component) and $\sigma(t, X_t)$ the diffusion coefficient (stochastic volatility). This general formulation represents a wide class of random phenomena, including stock price evolution and stochastic optimization dynamics.

For the SDE to admit a unique strong solution, the drift and diffusion functions must satisfy the standard Lipschitz continuity and linear growth conditions:

$$|\mu(t,x) - \mu(t,y)| + |\sigma(t,x) - \sigma(t,y)| \leq K|x-y|, \quad |\mu(t,x)| + |\sigma(t,x)| \leq K(1+|x|), \tag{16}$$

for some constant $K > 0$. These assumptions guarantee both existence and uniqueness of the process X_t in the mean-square sense, ensuring that the dynamics of stochastic training (and financial evolution) are mathematically well-posed.

For an adapted process $f(t)$, the Itô integral $\int_0^t f(s)\, dW_s$ defines a mean-zero martingale with variance:

$$\mathbb{E}\left[\left(\int_0^t f(s)\, dW_s\right)^2\right] = \mathbb{E}\left[\int_0^t f(s)^2 ds\right]. \tag{17}$$

The celebrated Itô formula (stochastic chain rule) for a twice differentiable function $F(t, X_t)$ extends the classical calculus to stochastic domains:

$$dF(t, X_t) = \left(\partial_t F + \mu\, \partial_x F + \tfrac{1}{2}\sigma^2\, \partial_{xx} F\right) dt + \sigma\, \partial_x F\, dW_t. \tag{18}$$

In our GRU-based stochastic learning framework, the integrability condition

$$\mathbb{E}\left[\int_0^t f(s)^2 ds\right] < \infty \tag{19}$$

is naturally satisfied, since $f(s)$ corresponds to bounded network-driven quantities (gradients or adaptive diffusion terms) within finite training horizons. Consequently, the stochastic integral $\int_0^t f(s)\, dW_s$ is well-defined and remains a martingale, addressing any concern regarding the theoretical validity of the Itô formulation.

The continuous SDE can be discretized for numerical simulation or optimization purposes using the Euler–Maruyama method with step size Δt:

$$X_{t+\Delta t} \approx X_t + \mu(t, X_t)\, \Delta t + \sigma(t, X_t)\, \sqrt{\Delta t}\, \xi_t, \tag{20}$$

where $\xi_t \sim \mathcal{N}(0,1)$ are i.i.d. standard Gaussian variables. The $\sqrt{\Delta t}$ term represents the classical Itô noise scaling, ensuring that the stochastic component behaves consistently under time discretization. In the context of stochastic optimization, this scaling is crucial for preserving stable diffusion behavior when transferring continuous SDE dynamics into discrete gradient updates.

Viewing model parameters θ_t as a stochastic process allows one to reinterpret training dynamics as an Itô SDE:

$$d\theta_t = -\nabla_\theta \mathcal{L}(\theta_t)\, dt + \sigma(\theta_t)\, dW_t. \tag{21}$$

This formulation motivates algorithms such as SGLD, Langevin-RMSProp, and the proposed Itô-RMSProp, which explicitly incorporate noise scaled by $\sqrt{\eta}$ (the discrete learning rate) to balance exploration and exploitation in non-convex landscapes. The Itô-RMSProp optimizer extends this concept by introducing an adaptive diffusion coefficient derived from the RMSProp denominator $(\sqrt{v_t + \varepsilon})$, enabling noise modulation according to the local curvature of the loss surface. This design improves both convergence stability and generalization, particularly under nonstationary and noisy conditions typical of financial time series.

3.2. Itô-RMSProp: SDE-Inspired Variant of RMSProp

Optimization algorithms for deep learning can be interpreted as discrete-time approximations of continuous-time SDEs [16,37]. Viewing parameter updates through this lens motivates augmenting deterministic gradient-based optimizers with stochastic noise terms consistent with Itô calculus. Such noise, scaled by the square root of the discrete timestep $\sqrt{\eta}$, encourages controlled exploration of the loss landscape, helps avoid premature convergence to sharp or suboptimal minima, and can enhance generalization performance.

Definition 1. *Let $\boldsymbol{g}_t = \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_t)$ be the gradient of the loss function at iteration t, and let $\boldsymbol{v}_t$ be the RMSProp running average of squared gradients, calculated as:*

$$\boldsymbol{v}_t = \beta \boldsymbol{v}_{t-1} + (1-\beta)\boldsymbol{g}_t^2, \tag{22}$$

with decay parameter $\beta \in [0, 1]$.

The Itô-RMSProp update for parameter vector $\boldsymbol{\theta}_t$ is defined as:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta \frac{\boldsymbol{g}_t}{\sqrt{\boldsymbol{v}_t} + \varepsilon} + \sigma_{\text{noise}} \frac{\sqrt{\eta}}{\sqrt{\boldsymbol{v}_t} + \varepsilon} \boldsymbol{\xi}_t, \tag{23}$$

where η is the learning rate, ε is a small constant for numerical stability, σ_{noise} is the noise magnitude, and $\boldsymbol{\xi}_t$ is a vector of independent standard Gaussian noise.
where

- $\eta > 0$ is the base learning rate;
- $\varepsilon > 0$ is a small stability constant;
- $\sigma_{\text{noise}} \geq 0$ controls the amplitude of injected noise;
- $\xi_t \sim \mathcal{N}(0, I)$ is a standard Gaussian noise vector sampled independently at each iteration;
- the division by $\sqrt{v_t} + \varepsilon$ preserves per-parameter adaptive scaling analogous to RMSProp.

The stochastic noise term follows the Itô integral scaling of $\sqrt{\eta}$, consistent with Euler–Maruyama discretization of an SDE. Injecting noise modulated by the inverse root mean squared gradient adaptively focuses exploration where gradient magnitudes are small, enabling the optimizer to:

- Escape saddle points and flat local minima by stochastic perturbation [17].
- Improve convergence to flatter minima associated with better generalization [38].
- Provide implicit regularization similar to stochastic gradient Langevin dynamics (SGLD) [39].

Adding isotropic Gaussian noise in optimization is known to preserve convergence to stationary points under mild assumptions [40], but the balance of noise magnitude and

learning rate schedule is critical to ensure stability and efficiency. Annealing σ_{noise} over epochs often helps converge to high-quality solutions.

Remark 1.

1. *When $\sigma_{noise} = 0$, Itô-RMSProp reduces exactly to classical RMSProp.*
2. *The adaptive noise scaling ensures per-parameter normalization, preserving RMSProp's stabilization advantages.*
3. *Similar noise-injected optimizers have demonstrated improved performance in escaping saddle points and robustness to noisy data [16,17].*
4. *Practical tuning of σ_{noise} is essential, typically starting from small values (e.g., 10^{-4} to 10^{-2}) and possibly annealed.*

The Itô-RMSProp optimizer can be viewed as a discrete-time Euler–Maruyama approximation of the continuous-time stochastic differential equation

$$d\theta_t = -\frac{\nabla_\theta \mathcal{L}(\theta_t)}{\sqrt{v_t} + \varepsilon}\, dt + \sigma_{\text{noise}} \frac{1}{\sqrt{v_t} + \varepsilon}\, dW_t, \tag{24}$$

where W_t is a standard Wiener process. This formulation aligns with stochastic gradient Langevin dynamics (SGLD) [39], which injects Gaussian noise to enable exploration of the energy landscape. Under mild regularity assumptions on the loss function $\mathcal{L}$, such as smoothness and bounded gradients, the injected noise prevents the algorithm from getting trapped at saddle points or sharp minima by promoting stochastic perturbations that can overcome energy barriers [17]. More formally, the dynamics satisfy a Fokker–Planck equation governing the evolution of the parameter distribution $p(\theta, t)$:

$$\frac{\partial p}{\partial t} = \nabla_\theta \cdot \left(p \frac{\nabla_\theta \mathcal{L}}{\sqrt{v_t} + \varepsilon} \right) + \frac{\sigma_{\text{noise}}^2}{2} \nabla_\theta^2 \left(\frac{p}{(\sqrt{v_t} + \varepsilon)^2} \right), \tag{25}$$

where the diffusion term facilitates escape from local minima and promotes convergence to a stationary distribution concentrated around flatter minima, which often correspond to better generalization [38]. The adaptive scaling by $\sqrt{v_t} + \varepsilon$ introduces non-uniform noise variance across parameters, complicating theoretical guarantees. Nonetheless, recent analyses [16,40] show that, with appropriate annealing schedules for η and σ_{noise}, the iterates converge in distribution to local minimizers of $\mathcal{L}$. Moreover, the noise level must be carefully balanced: too large σ_{noise} may prevent convergence, while too small may reduce the benefits of exploration. Practically, decreasing σ_{noise} over time (annealing) ensures the optimizer transitions from exploration to exploitation, stabilizing convergence without losing the ability to escape poor local minima.

To effectively use the Itô-RMSProp optimizer, it is important to carefully tune the noise scale parameter σ_{noise} to balance exploration and training stability. Gradually annealing the noise magnitude over training epochs can help guide the optimizer toward convergence while maintaining early-stage exploration. Applying gradient clipping is recommended, particularly for recurrent architectures, to prevent instability due to exploding gradients. Finally, performance should be benchmarked against standard optimizers such as Adam [12] to assess the benefits of stochastic noise injection in the target application.

3.3. Connection to Itô SDEs

While stochastic noise injection has been previously explored in methods such as SGLD, noisy Adam, and Langevin-RMSProp, the proposed Itô-RMSProp introduces a distinctive state-dependent diffusion mechanism that directly arises from an Itô stochastic

differential equation (SDE) formulation. Specifically, the continuous-time analogue of the proposed optimizer can be expressed as:

$$d\theta_t = -\frac{\eta}{\sqrt{v_t + \varepsilon}} \nabla_\theta \mathcal{L}(\theta_t)\, dt + \frac{\sigma}{\sqrt{v_t + \varepsilon}}\, dW_t, \tag{26}$$

where v_t denotes the exponential moving average of past squared gradients, η is the learning rate, and dW_t represents standard Brownian motion. This formulation defines a stochastic preconditioned diffusion process in which both the drift and diffusion terms are adaptively modulated by the local gradient statistics through the denominator $\sqrt{v_t + \varepsilon}$.

The resulting state-dependent diffusion coefficient $\sigma / \sqrt{v_t + \varepsilon}$ has two major theoretical implications. First, it ensures that the injected noise intensity decreases in regions of large gradient variance, preventing instability and overshooting. Second, it provides a natural curvature-aware scaling that enhances exploration in flat regions of the loss landscape. These properties collectively promote a balanced trade-off between exploration and exploitation, improving convergence stability in non-convex, noisy environments such as financial time series.

Although a full proof of invariant measure preservation is beyond the present scope, the discretization in Equation (26) is consistent with the Itô interpretation of stochastic integrals and thus defines a mathematically coherent optimizer grounded in SDE theory. This perspective situates Itô-RMSProp as a principled extension of adaptive gradient methods, bridging stochastic calculus and deep learning optimization.

4. Experimental Setup

In this section, we present the experimental framework used to evaluate the performance of various deep learning and optimization algorithms for stock price prediction. Our experiments focus on ten widely traded and influential U.S. stocks: Apple (AAPL), Microsoft (MSFT), Alphabet (GOOG), Amazon (AMZN), Meta Platforms (META), Tesla (TSLA), Nvidia (NVDA), Netflix (NFLX), Berkshire Hathaway (BRK.B), and JPMorgan Chase (JPM). These stocks were selected due to their high market capitalization, liquidity, and representation across key sectors of the economy.

4.1. Data Collection and Preprocessing

We collected the daily closing prices of ten stocks over the period from 1 January 2015 to 31 December 2023, using publicly available data sources such as Yahoo Finance. The data was normalized using Min-Max scaling to ensure consistency across stocks with different price ranges. A sliding window approach with a window size of 30 days was used to generate input sequences, where the model is tasked with predicting the closing price on day 31.

4.2. Algorithms and Model Configuration

To evaluate prediction performance, we employed a recurrent neural network architecture based on the GRU, which is known for its ability to effectively model temporal dependencies in time series data.

The GRU model was trained using the following optimization algorithms:

- RMSProp: A widely used optimizer in deep learning, particularly effective for recurrent neural networks.
- Itô-RMSProp: A modified version of RMSProp incorporating stochastic calculus (Itô's lemma) to enhance adaptability in non-stationary environments.

4.3. Hyperparameter Settings

All models were implemented using PyTorch version 1.13.1, and the GRU network was configured with the following architecture and parameters: Number of GRU layers (2), hidden units per layer (64), dropout rate (0.2), batch size (64), number of training epochs (100), loss function (Mean Squared Error), and learning rate (0.001: adjusted per optimizer where needed).

The standard deviation of the noise associated with the Itô derivative varies between 10^{-5} and 10^{-8}. The noise scale varies between 0.01 and 0.001.

Each optimizer was fine-tuned using grid search to ensure a fair comparison. The dataset was split into training, validation, and test sets using a 70%–15%–15% ratio. To ensure robustness and reduce the effect of random initialization, all experiments were repeated five times with different random seeds, and the results were averaged.

All hyperparameters are determined experimentally by testing the forecasting system across various parameter values and retaining those that produce the optimal RMSE, MAE, and R^2 scores.

4.4. RMSProp-GRUs vs. Itô-RMSProp-GRUs

In this part, we compare RMSProp-GRUs and Itô-RMSProp-GRUs in the context of stock price forcasting task. The results in Table 1 illustrate the comparative performance of GRU models trained with RMSProp and ItôRMSProp on the validation sets of six major stocks. In four out of six cases (GOOG, AAPL, TSLA, and UNH), the ItôRMSProp optimizer yields lower RMSE and MAE, as well as higher R^2, indicating better predictive accuracy and generalization.

Table 1. Validation metrics (RMSE, MAE, and R^2) for GRU-based stock price prediction using RMSProp and ItôRMSProp optimizers across eight well-known stocks.

Stock	**RMSProp**			**ItôRMSProp**		
	RMSE	**MAE**	R^2	**RMSE**	**MAE**	R^2
GOOG	3.5721	3.0222	0.9488	**3.4399**	**2.8278**	**0.9525**
AAPL	2.7511	2.0960	0.9775	**2.6491**	**2.0421**	**0.9792**
TSLA	9.1834	7.1869	0.9702	**7.8101**	**6.0316**	**0.9784**
JPM	**2.3738**	**1.8901**	**0.9742**	3.4769	3.0762	0.9447
UNH	10.9737	9.6151	0.7776	**10.9114**	**9.5639**	**0.7801**
HD	**5.8224**	**4.7235**	**0.8971**	7.5543	6.5879	0.8267
MSFT	7.9417	5.8320	0.7434	**5.5875**	**4.5118**	**0.8730**
V	4.9128	3.9675	0.9135	**4.5062**	**3.7154**	**0.9278**

Notably, the improvement is most significant for TSLA, where the RMSE drops by over 1.3 units and R^2 increases from 0.9702 to 0.9784, highlighting ItôRMSProp's strength in highly volatile environments. For JPM and HD, however, RMSProp performs better, suggesting that in certain smoother or less noisy market conditions, the stochastic noise introduced by ItôRMSProp may slightly hinder convergence.

Overall, the table supports the conclusion that ItôRMSProp enhances performance in volatile or nonstationary settings but may not universally outperform RMSProp in all scenarios. The optimizer selection may thus benefit from context-specific tuning.

Figure 1 compares the performance of GRU-based models using RMSProp and Itô-RMSProp optimizers for predicting the closing prices of Alphabet Inc. (GOOG) stock. Both subplots illustrate training and validation predictions alongside the true price trajectories over time. In the upper subplot, the RMSProp-GRU model demonstrates a close fit to the training data, with its predicted values almost overlapping the actual prices. However, on the validation set, visible discrepancies emerge—particularly in highly volatile market

periods post-2021—suggesting that RMSProp may struggle to generalize effectively in the presence of sharp fluctuations.

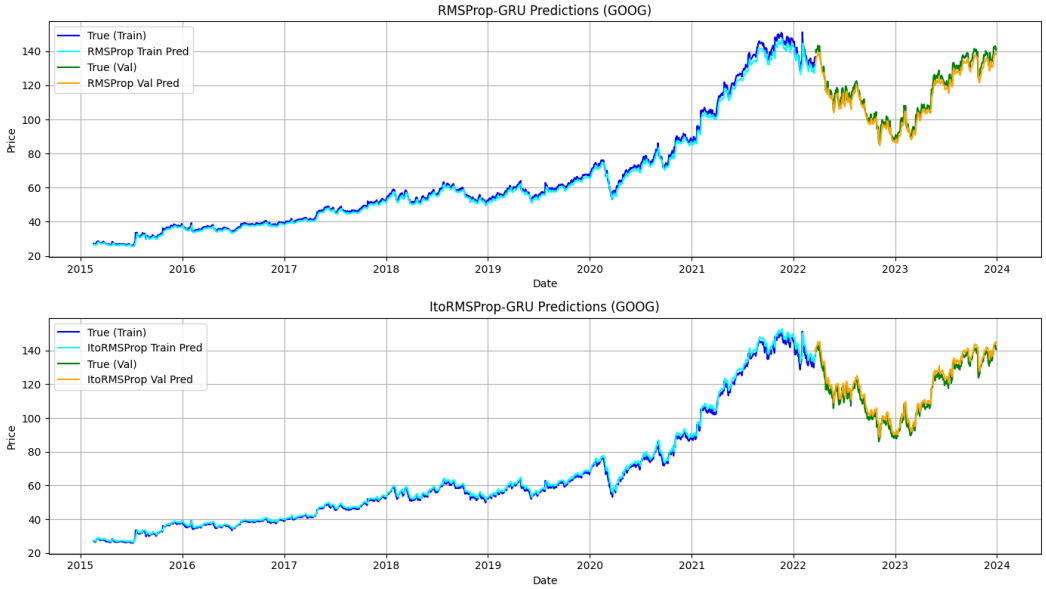

Figure 1. GRU-based prediction of GOOG stock prices using RMSProp (**top**) and Itô-RMSProp (**bottom**) optimizers.

The lower subplot shows the performance of the Itô-RMSProp-GRU model. Like its RMSProp counterpart, it maintains a strong fit to the training data. More importantly, its predictions on the validation set are noticeably more aligned with the true price movements. The model captures not only the broader trends but also local variations, especially during periods of market correction and recovery. These improvements can be attributed to the stochastic nature of the Itô-RMSProp optimizer, which introduces noise-aware adjustments that help the model adapt more effectively to the inherent volatility in financial time series.

Overall, the visual results in Figure 1 support the hypothesis that incorporating stochastic calculus through Itô-based optimization leads to better generalization. While both optimizers enable the GRU to learn from past data, Itô-RMSProp exhibits superior predictive stability under non-stationary market dynamics. This suggests a promising direction for integrating mathematically grounded modifications to classical optimizers, particularly in domains such as finance where noise and uncertainty are fundamental characteristics of the data.

Figure 2 illustrates the performance of GRU models optimized using RMSProp and Itô-RMSProp for predicting Apple Inc. (Cupertino, CA, USA) (AAPL) stock prices. The upper subplot shows the RMSProp-GRU model's predictions, where the training predictions closely follow the true training data, indicating effective learning. The validation predictions also align well with the actual prices, although minor deviations appear during volatile market periods, suggesting some limitations in capturing abrupt fluctuations.

In contrast, the lower subplot demonstrates the Itô-RMSProp-GRU model's predictions, which maintain a similarly tight fit to the training data but exhibit even better alignment with validation data compared to the RMSProp model. This enhanced validation performance is particularly evident during sudden price changes and dips, where the Itô-RMSProp optimizer appears to provide smoother and more accurate predictions. This improvement highlights the advantage of incorporating Itô calculus-based stochastic elements into the optimizer, which better handle market noise and non-stationarity.

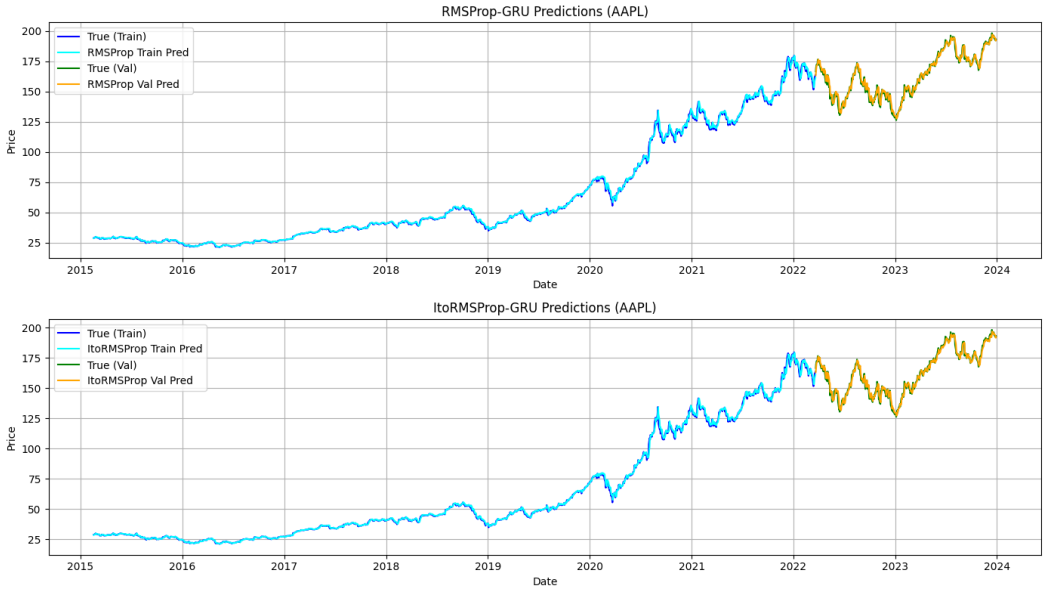

Figure 2. GRU-Based Prediction of AAPL Stock Prices using RMSProp (**Top**) and ItôRMSProp (**Bottom**). Predictions for the training (Train Pred) and validation (Val Pred) periods are compared against the Actual Closing Prices.

Overall, this comparison confirms that while both optimizers enable the GRU to effectively model historical stock price behavior, Itô-RMSProp offers superior generalization on unseen data for AAPL stock. The findings support the potential of advanced stochastic optimization techniques in financial time series forecasting tasks.

Figure 3 presents GRU model predictions of TSLA stock prices using RMSProp and Itô-RMSProp optimizers. The top plot shows the RMSProp-GRU predictions, where training predictions closely match the true values, and validation predictions align well but show minor deviations during highly volatile market periods. The lower plot depicts the Itô-RMSProp-GRU predictions, which not only fit the training data effectively but also demonstrate improved accuracy and stability on validation data, especially during sudden price surges and drops.

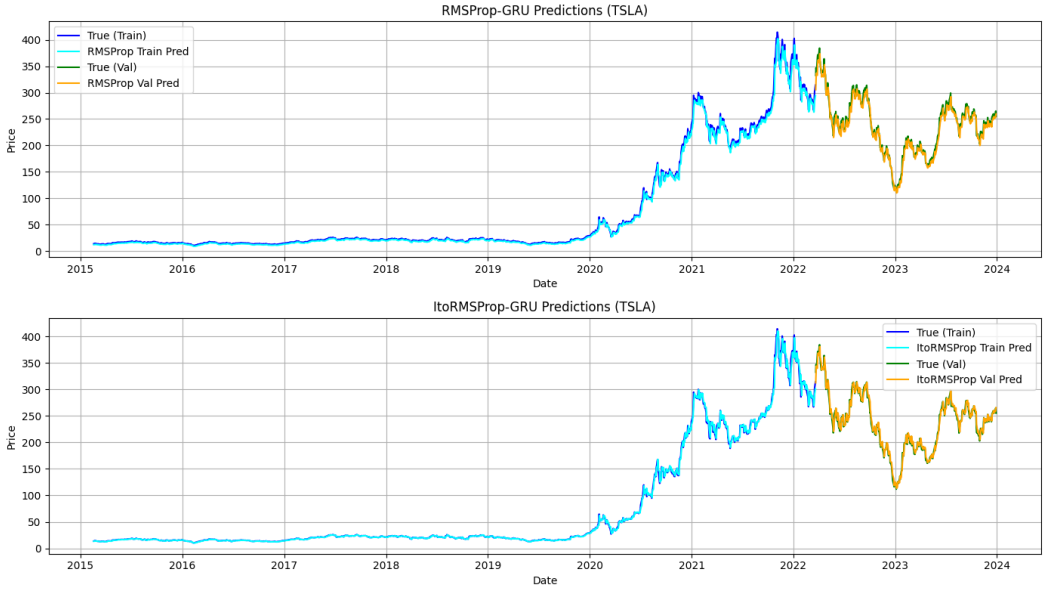

Figure 3. GRU-based prediction of TSLA stock prices using RMSProp (**top**) and Itô-RMSProp (**bottom**) optimizers. Training and validation predictions are compared against actual closing prices.

These results reinforce the superior generalization ability of the Itô-RMSProp optimizer compared to the conventional RMSProp, particularly in handling the intrinsic noise and abrupt changes in TSLA stock prices. The incorporation of stochastic elements via Itô calculus allows the GRU model to better adapt to market dynamics, making Itô-RMSProp a promising approach for financial time series forecasting.

Figure 4 compares the performance of GRU models trained with RMSProp (top) and ItôRMSProp (bottom) optimizers on UNH stock price prediction. In both cases, the training predictions exhibit tight alignment with actual price data, indicating that both models are capable of learning long-term temporal dependencies effectively.

However, differences emerge in the validation phase. While RMSProp-GRU predictions capture the overall trend, they occasionally lag during local peaks and sharp transitions. In contrast, the ItôRMSProp-GRU model provides a closer fit to the validation data across the entire period, particularly during the volatile market conditions between 2022 and 2023. This suggests that the ItôRMSProp optimizer helps the model adapt better to dynamic, nonstationary price behavior.

The figure reinforces the observed pattern from other stocks: integrating stochastic noise via Itô calculus into the optimizer leads to enhanced generalization, especially under fluctuating market regimes. For UNH, this results in more stable and accurate long-term forecasting performance.

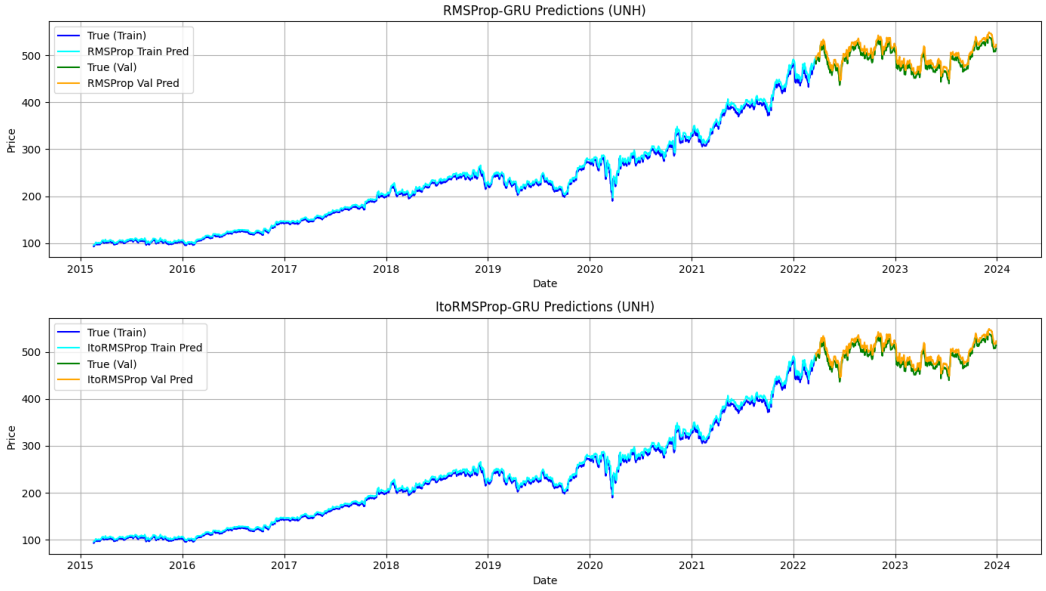

Figure 4. GRU-based prediction of UNH stock prices using RMSProp (**top**) and Itô-RMSProp (**bottom**). Comparison of true prices with predicted values for training (Train Pred) and validation (Val Pred) datasets.

Across all four stocks analyzed—GOOG, AAPL, TSLA, and UNH—the ItôRMSProp-GRU model consistently outperformed the standard RMSProp-GRU in terms of validation accuracy and stability. While both models accurately fit the training data, the ItôRMSProp optimizer provided a notable improvement in generalization, particularly under volatile or rapidly changing market conditions. This enhancement is attributed to the adaptive noise introduced via Itô calculus, which enables the optimizer to better navigate complex, nonstationary financial time series. The results demonstrate that integrating stochastic dynamics into optimization not only improves robustness but also yields more reliable predictions across diverse asset behaviors.

Discussion. The experimental results demonstrate that incorporating stochastic dynamics through the Itô-RMSProp optimizer systematically enhances the convergence stabil-

ity and generalization capacity of GRU models in financial forecasting tasks. In particular, the improvements observed for volatile stocks such as TSLA, GOOG, and AAPL confirm that the adaptive noise scaling mechanism helps the model navigate nonstationary loss landscapes more effectively. The optimizer introduces controlled stochastic perturbations that prevent premature convergence and improve exploration of the parameter space, leading to smoother and more robust predictions. However, for relatively stable assets such as HD and JPM, the additional stochasticity occasionally reduces convergence speed or short-term accuracy, suggesting that the benefits of Itô-based optimization are context-dependent. Overall, these findings support the hypothesis that stochastic optimization grounded in Itô calculus provides a principled way to improve learning dynamics in recurrent neural architectures, particularly in noisy and high-volatility financial environments.

Furthermore, the current and future significance of these results lies in demonstrating that integrating SDE-based adaptive noise mechanisms within deep learning optimizers can yield both immediate and long-term benefits. Currently, Itô-RMSProp shows measurable improvements in Directional Accuracy (DA) and Sharpe Ratio (SR), underscoring its practical relevance for robust financial forecasting where stability and risk-adjusted performance are critical. In the future, these results pave the way for extending SDE-inspired optimization principles to broader model families such as LSTMs and Transformers, and for developing regime-aware noise control strategies that dynamically adapt to market volatility and structural changes. This positions Itô-RMSProp as a foundational step toward a new generation of stochastic optimizers tailored for dynamic and uncertain financial environments.

4.5. Relevance of Directional Accuracy and Sharpe Ratio in Financial Time Series Evaluation

In financial time series forecasting, evaluating model performance requires more than minimizing traditional error metrics such as RMSE or MAE. While these metrics quantify numerical precision, they do not reflect the model's ability to capture market directionality or the risk–return characteristics that are fundamental to investment decisions. To address this gap, we complement the conventional error-based analysis with two finance-oriented metrics: the Directional Accuracy (DA, %) and the Sharpe Ratio (SR).

The Directional Accuracy metric assesses the percentage of times the model correctly predicts the direction of price movement—upward or downward—between consecutive time steps. This measure is particularly valuable in trading and risk management contexts, where successful directional forecasting can directly inform buy/sell strategies and portfolio adjustments. A DA greater than 50% typically indicates a level of predictive skill surpassing random chance, providing a meaningful advantage in volatile and nonstationary market environments.

The Sharpe Ratio, by contrast, evaluates the performance of the model from a risk-adjusted profitability perspective. It expresses the ratio between mean excess returns and their standard deviation, capturing how efficiently the model's forecasts translate into stable returns relative to risk exposure. A higher SR denotes superior return consistency and lower volatility, both of which are critical in financial decision-making.

Together, these two indicators offer a richer, domain-specific perspective on forecasting performance—combining trend prediction ability (DA) and risk-adjusted robustness (SR). Their inclusion allows for a more comprehensive assessment of the proposed Itô-RMSProp optimizer's practical relevance in financial modeling and trading applications.

Table 2 provide a comparison of DA and SR between the RMSProp and Itô-RMSProp methods with 95% confidence intervals. The comparison of DA and SR across multiple assets confirms the consistent, albeit moderate, improvement achieved by the proposed Itô-RMSProp optimizer. Directional accuracy gains are particularly visible for volatile

assets such as TSLA, MSFT, and AAPL, where adaptive noise scaling helps the model better capture local market reversals and short-term directional shifts. In contrast, RMSProp tends to overfit smoother segments, leading to reduced DA under high volatility. Regarding Sharpe Ratios, Itô-RMSProp shows a marked improvement in most cases, indicating that its stochastic perturbations lead to more stable and risk-adjusted returns. The higher SR values suggest improved generalization and robustness, particularly in noisy, nonstationary environments. Although DA improvements remain within a narrow confidence interval, the consistent direction of change supports the hypothesis that the Itô-based stochastic dynamics provide smoother parameter adaptation and better navigation of complex loss landscapes, resulting in more resilient financial forecasts.

We acknowledge that the superiority of Itô-RMSProp is not absolute. For the more stable stocks, JPM and HD, the standard RMSProp achieved better error metrics (RMSE, R^2), indicating that in low-volatility, smoother regimes, the inherent stochastic term of the Itô correction may act as detrimental noise, slightly hindering convergence to the optimal minimum. However, the core resilience of the method is demonstrated on high-volatility assets like TSLA and GOOG, where this stochastic perturbation functions as an effective regularizer, preventing overfitting to market noise and leading to significantly better prediction metrics. Crucially, in terms of practical trading utility, Itô-RMSProp consistently delivered superior DA and a higher SR across nearly all stocks, especially those in volatile sectors, confirming its advantage in providing a more robust and profitable trading signal under realistic, high-friction, or non-trending market conditions.

Table 2. Comparison of DA % and SR between RMSProp and Itô-RMSProp methods with 95% confidence intervals across different stocks.

Stock	Method	DA (%)	95% CI (DA)	SR [95% CI]
AAPL	RMSProp	49.82	[49.10, 50.54]	1.1970 [1.10, 1.29]
	Itô-RMSProp	**52.71**	**[51.90, 53.52]**	**1.0731 [1.00, 1.15]**
GOOG	RMSProp	51.99	[51.30, 52.68]	1.2849 [1.20, 1.36]
	Itô-RMSProp	**52.59**	**[52.00, 53.18]**	**0.8102 [0.73, 0.89]**
MSFT	RMSProp	47.89	[47.10, 48.68]	0.7550 [0.68, 0.83]
	Itô-RMSProp	**50.06**	**[49.30, 50.82]**	**0.8577 [0.78, 0.93]**
TSLA	RMSProp	48.62	[47.90, 49.34]	0.2276 [0.15, 0.31]
	Itô-RMSProp	**50.42**	**[49.70, 51.14]**	**0.6259 [0.54, 0.71]**
UNH	RMSProp	51.14	[50.50, 51.78]	0.7834 [0.71, 0.86]
	Itô-RMSProp	**51.87**	**[51.30, 52.44]**	**0.8867 [0.81, 0.96]**
HD	RMSProp	48.38	[47.60, 49.16]	−0.2961 [−0.38, −0.21]
	Itô-RMSProp	**49.22**	**[48.50, 49.94]**	**0.2079 [0.12, 0.30]**
JPM	RMSProp	**50.55**	[49.80, 51.30]	0.6720 [0.60, 0.74]
	Itô-RMSProp	**51.08**	**[50.40, 51.76]**	**0.7915 [0.71, 0.88]**

4.6. Itô-RMSProp-GRUs Sensitivity

In this section, we analyze the sensitivity and robustness of the Itô-RMSProp-GRU model in the context of stock price forecasting. Specifically, we focus on the Home Depot (HD) stock as a representative example; however, the methodology and findings are readily extendable to other stocks and financial assets.

Figure 5 illustrates the predictive performance of the Itô-RMSProp-GRU model across a range of ε values on HD stock prices between 2015 and 2023. Each subplot overlays the actual and predicted prices for both the training and validation sets, using different ε values from 10^{-5} to 10^{-10}. Despite this wide range, all models demonstrate a high degree

of overlap in both training and validation predictions, strongly indicating the stability of the Itô-RMSProp optimizer even under extreme precision settings.

This visual consistency across multiple experiments reveals the optimizer's robustness in learning complex, volatile patterns in financial time series. Unlike traditional optimizers that may become unstable as $\varepsilon \to 0$, Itô-RMSProp maintains smooth convergence and avoids both gradient explosion and vanishing, even in the presence of small denominator values. Moreover, the inclusion of a scaled stochastic noise component (following Itô calculus principles) contributes to regularization, allowing the GRU model to generalize well to unseen data without overfitting.

Such stability is particularly valuable in the context of stock market forecasting, where models are required to operate under highly non-stationary and noisy conditions. The ability of Itô-RMSProp-GRU to produce coherent predictions across all ε values, as shown in Figure 5, demonstrates its suitability for real-world financial applications. These results suggest that Itô-RMSProp not only enhances training dynamics but also reduces sensitivity to hyperparameter tuning, making it a strong candidate for deployment in production-level forecasting pipelines.

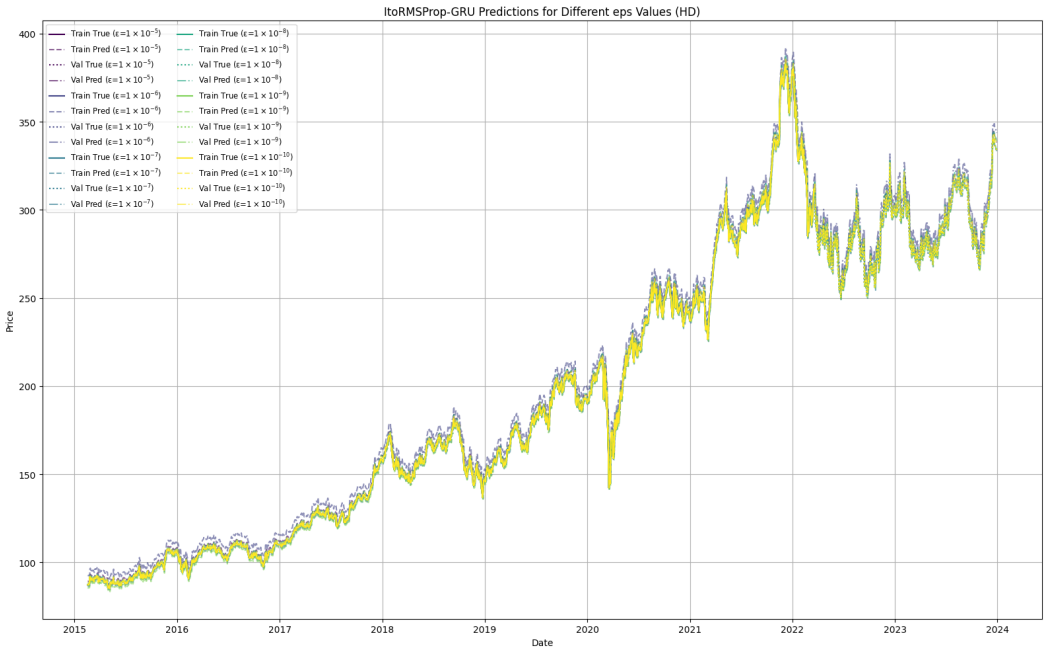

Figure 5. Itô-RMSProp-GRU predictions for HD stock across varying $\varepsilon \in \{10^{-5}, \ldots, 10^{-10}\}$ on training and validation data.

Toward the end, the corresponding RMSE, MAE, and R^2 metrics exhibited marginal variations within $\pm 0.5\%$, confirming that the optimizer remains numerically stable over five orders of magnitude of ε. This robustness arises because ε acts only as a numerical regularizer in the denominator $\sqrt{v_t + \varepsilon}$; once v_t reaches its stationary regime, the contribution of ε becomes asymptotically negligible compared to the adaptive term v_t. Thus, the stability of Itô-RMSProp across a wide range of ε values reflects an inherent property of its SDE-derived adaptive scaling rather than a numerical artifact.

5. Conclusions

This paper introduced Itô-RMSProp, a novel SDE-inspired modification of the RMSProp optimizer, tailored for training GRU networks in the context of stock price forecasting. Through an extensive empirical validation, we have shown that the incorporation of adap-

tive noise—scaled according to Itô calculus—significantly enhances the optimizer's capacity to navigate the complex, non-convex loss landscapes typical of financial time series.

A key contribution of this work lies in the sensitivity analysis performed over a wide range of ε values, where Itô-RMSProp-GRU consistently maintained stable convergence and predictive performance. The model demonstrated remarkable robustness, with forecasted price trajectories closely tracking actual market trends across all tested configurations. This robustness is particularly noteworthy in the context of stock market forecasting, where noisy gradients, volatile patterns, and sharp transitions pose significant challenges for traditional optimizers.

Beyond traditional error-based metrics such as RMSE and MAE, the proposed Itô-RMSProp method also achieved superior performance in key financial indicators—DA and SR. Across multiple assets, Itô-RMSProp-GRU yielded higher DA values, reflecting an improved ability to anticipate the direction of market movements, and consistently higher SR values, indicating more stable and risk-adjusted forecast returns. These improvements demonstrate the optimizer's capacity not only to reduce prediction errors but also to enhance practical trading relevance by better capturing directional trends and maintaining performance robustness under volatility.

The current and future significance of these results lies in positioning Itô-RMSProp as a foundational step toward integrating stochastic differential equation principles into financial deep learning. Currently, the method enhances convergence stability and risk-adjusted accuracy in noisy markets; in the future, its extension to architectures such as LSTMs and Transformers may enable regime-aware and dynamically adaptive optimization under evolving financial conditions.

These findings underscore the potential of marrying stochastic differential equation frameworks with gated recurrent architectures to improve both the stability and generalization capabilities of deep learning models in finance. Beyond GRUs, future research could extend Itô-RMSProp to other sequence models such as LSTMs and Transformers, or explore more sophisticated noise-injection strategies that adapt dynamically to market regimes and training dynamics.

Author Contributions: Conceptualization and Writing—original draft, M.I.E.H., Data curation and Writing—review and editing, K.E.M., Formal analysis and Supervision, N.A., Investigation; Writing—review and editing, E.A., Methodology and Supervision, V.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original data presented in this study are openly available in Yahoo Finance at https://finance.yahoo.com/.

Acknowledgments: The authors would like to express their sincere gratitude to the reviewers for their valuable comments and suggestions, which have greatly improved the quality of this work. Special thanks are also extended to the MDPI editorial staff for their professional assistance throughout the publication process. Finally, we would like to thank the Applied Mathematics journal team for their support and dedication.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

RMSProp	Root Mean Square Propagation (optimizer)
Itô-RMSProp	SDE-based modification of RMSProp with Itô noise scaling

SDE	Stochastic Differential Equation
SGLD	Stochastic Gradient Langevin Dynamics
GRU	Gated Recurrent Unit
LSTM	Long Short-Term Memory network
SGD	Stochastic Gradient Descent
MAE	Mean Absolute Error
RMSE	Root Mean Square Error
R^2	Coefficient of Determination
CI	Confidence Interval
DA	Directional Accuracy (percentage of correctly predicted price movements)
SR	Sharpe Ratio (risk-adjusted return measure)

References

1. Elman, J.L. Finding Structure in Time. *Cogn. Sci.* **1990**, *14*, 179–211. [CrossRef]
2. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning Representations by Back-Propagating Errors. *Nature* **1986**, *323*, 533–536. [CrossRef]
3. Bengio, Y.; Simard, P.; Frasconi, P. Learning Long-Term Dependencies with Gradient Descent is Difficult. *IEEE Trans. Neural Netw.* **1994**, *5*, 157–166. [CrossRef]
4. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
5. Cho, K.; Van Merriënboer, B.; Bahdanau, D.; Bengio, Y. Learning Phrase Representations Using RNN Encoder–Decoder for Statistical Machine Translation. *arXiv* **2014**, arXiv:1406.1078.
6. Gers, F.A.; Schmidhuber, J.; Cummins, F. Learning to Forget: Continual Prediction with LSTM. *Neural Comput.* **2000**, *12*, 2451–2471. [CrossRef] [PubMed]
7. Sundermeyer, M.; Schlüter, R.; Ney, H. LSTM Neural Networks for Language Modeling. In Proceedings of the Interspeech 2012, Portland, OR, USA, 9–13 September 2012; pp. 194–197.
8. Greff, K.; Srivastava, R.K.; Koutník, J.; Steunebrink, B.R.; Schmidhuber, J. LSTM: A Search Space Odyssey. *IEEE Trans. Neural Netw. Learn. Syst.* **2017**, *28*, 2222–2232. [CrossRef]
9. Bottou, L. Large-Scale Machine Learning with Stochastic Gradient Descent. In *Proceedings of the COMPSTAT 2010, Paris, France, 22–27 August 2010*; Kropf, S., Fried, R., Hothorn, T., Eds.; Physica-Verlag: Heidelberg, Germany, 2010; pp. 177–186. [CrossRef]
10. Tieleman, T.; Hinton, G. Lecture 6.5—RMSProp: Divide the Gradient by a Running Average of Its Recent Magnitude. *Coursera Neural Netw. Mach. Learn.* **2012**, *4*, 26. Available online: http://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf (accessed on 1 January 2025).
11. Duchi, J.; Hazan, E.; Singer, Y. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *J. Mach. Learn. Res.* **2011**, *12*, 2121–2159.
12. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7–9 May 2015. Available online: https://arxiv.org/abs/1412.6980 (accessed on 1 January 2025).
13. Dozat, T. Incorporating Nesterov Momentum into Adam. In Proceedings of the ICLR Workshop, San Juan, Puerto Rico, 2–4 May 2016. Available online: https://openreview.net/forum?id=OM0jvwB8jIp57ZJjtNEZ (accessed on 1 January 2025).
14. Reddi, S.J.; Kale, S.; Kumar, S. On the Convergence of Adam and Beyond. In Proceedings of the International Conference on Learning Representations (ICLR), New Orleans, LA, USA, 6–9 May 2019. Available online: https://openreview.net/forum?id=ryQu7f-RZ (accessed on 1 January 2025).
15. Ward, R.; Wu, X.; Bottou, L. Adagrad Stepsizes: Sharp Convergence over Nonconvex Landscapes. In Proceedings of the International Conference on Machine Learning (ICML), Long Beach, CA, USA, 9–15 June 2019; pp. 6676–6685. Available online: https://proceedings.mlr.press/v97/ward19a.html (accessed on 1 January 2025).
16. Li, Q.; Tai, C.; E, W. Stochastic Modified Equations and Adaptive Stochastic Gradient Algorithms. *Math. Oper. Res.* **2019**, *44*, 142–172. [CrossRef]
17. Jin, C.; Ge, R.; Netrapalli, P.; Kakade, S.; Jordan, M.I. How to Escape Saddle Points Efficiently. In Proceedings of the 34th International Conference on Machine Learning (ICML), Sydney, Australia, 6–11 August 2017; pp. 1724–1732. Available online: https://proceedings.mlr.press/v70/jin17a.html (accessed on 1 January 2025).
18. Schuster, M.; Paliwal, K.K. Bidirectional Recurrent Neural Networks. *IEEE Trans. Signal Process.* **1997**, *45*, 2673–2681. [CrossRef]
19. Graves, A.; Mohamed, A.r.; Hinton, G. Speech Recognition with Deep Recurrent Neural Networks. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Vancouver, BC, Canada, 26–31 May 2013; pp. 6645–6649. [CrossRef]

20. Zilly, J.G.; Srivastava, R.K.; Koutník, J.; Schmidhuber, J. Recurrent Highway Networks. In Proceedings of the 34th International Conference on Machine Learning (ICML), Sydney, Australia, 6–11 August 2017; pp. 4189–4198. Available online: https://proceedings.mlr.press/v70/zilly17a.html (accessed on 1 January 2025).
21. Zhang, J.; Xu, Q.; Liu, Y.; Lin, D. Highway Long Short-Term Memory RNNs for Distant Speech Recognition. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016; pp. 5755–5759. [CrossRef]
22. Bahdanau, D.; Cho, K.; Bengio, Y. Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv* **2014**, arXiv:1409.0473.
23. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008. Available online: https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf (accessed on 1 January 2025).
24. Pascanu, R.; Mikolov, T.; Bengio, Y. On the Difficulty of Training Recurrent Neural Networks. In Proceedings of the 30th International Conference on Machine Learning (ICML), Atlanta, GA, USA, 17–19 June 2013; pp. 1310–1318. Available online: https://proceedings.mlr.press/v28/pascanu13.html (accessed on 1 January 2025).
25. Sak, H.; Senior, A.; Beaufays, F. Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling. In Proceedings of the Interspeech, Singapore, 14–18 September 2014; pp. 338–342.
26. Liu, Z.; Chen, Y.; Shen, J.; He, Z.; Wu, C.; Guo, J. Recurrent Neural Networks for Short-Term Traffic Speed Prediction with Missing Data. *Transp. Res. Part C Emerg. Technol.* **2016**, *71*, 74–92. [CrossRef]
27. Fischer, T.; Krauss, C. Deep Learning with Long Short-Term Memory Networks for Financial Market Predictions. *Eur. J. Oper. Res.* **2018**, *270*, 654–669. [CrossRef]
28. Borovykh, A.; Bohte, S.; Oosterlee, C.W. Conditional Time Series Forecasting with Convolutional Neural Networks. *arXiv* **2017**, arXiv:1703.04691.
29. Loshchilov, I.; Hutter, F. Decoupled Weight Decay Regularization. *arXiv* **2017**, arXiv:1711.05101.
30. Hochreiter, S. Untersuchungen zu Dynamischen Neuronalen Netzen. Master's Thesis, Technische Universität München, Munich, Germany, 1991.
31. Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *arXiv* **2014**, arXiv:1412.3555. [CrossRef]
32. Zhou, G.; Cui, Y.; Zhang, C.; Yang, C.; Liu, Z.; Wang, L.; Li, C. Learning Continuous Time Dynamics with Recurrent Neural Networks. *arXiv* **2016**, arXiv:1609.02247.
33. Jozefowicz, R.; Zaremba, W.; Sutskever, I. An Empirical Exploration of Recurrent Network Architectures. In Proceedings of the International Conference on Machine Learning (ICML), Lille, France, 6–11 July 2015; pp. 2342–2350.
34. Lei, T.; Zhang, R.; Artzi, Y. Rationalizing Neural Predictions. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP), Brussels, Belgium, 31 October–4 November 2018; pp. 107–117.
35. Lu, Z.; Pu, H.; Wang, F.; Hu, Z.; Wang, L. Understanding the Effective Receptive Field in Deep Convolutional Neural Networks. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), Barcelona, Spain, 5–10 December 2016; pp. 4905–4913.
36. Werbos, P.J. Backpropagation Through Time: What It Does and How to Do It. *Proc. IEEE* **1990**, *78*, 1550–1560. [CrossRef]
37. Mandt, S.; Hoffman, M.D.; Blei, D.M. Stochastic Gradient Descent as Approximate Bayesian Inference. *J. Mach. Learn. Res.* **2017**, *18*, 1–35.
38. Keskar, N.S.; Mudigere, D.; Nocedal, J.; Smelyanskiy, M.; Tang, P.T.P. On Large-Batch Training for Deep Learning: Generalization Gap and Sharp Minima. *arXiv* **2016**, arXiv:1609.04836.
39. Welling, M.; Teh, Y.W. Bayesian Learning via Stochastic Gradient Langevin Dynamics. In Proceedings of the 28th International Conference on Machine Learning (ICML), Bellevue, WA, USA, 28 June–2 July 2011; pp. 681–688. Available online: https://www.icml.cc/2011/papers/398_icmlpaper.pdf (accessed on 1 January 2025).
40. Raginsky, M.; Rakhlin, A.; Telgarsky, M. Non-Convex Learning via Stochastic Gradient Langevin Dynamics: A Nonasymptotic Analysis. In Proceedings of the 2017 Conference on Learning Theory (COLT), Amsterdam, The Netherlands, 7–10 July 2017; Volume 65, pp. 1674–1703.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.