Trade-off Analysis of Efficiency and Accuracy in GRU vs LSTM

Bowen Yang[1*]

[1]College of Science, Yanbian University, Yanji 133002, China

Abstract. While both Gated Recurrent Unit (GRU) and Long Short-Term Memory (LSTM) networks effectively address the vanishing gradient problem in RNNs, a clear trade-off exists between their computational efficiency and predictive accuracy in the absence of clear quantitative guidelines. for model selection. This study presents a systematic evaluation of this trade-off through mathematical analysis and controlled experiments across financial and meteorological forecasting tasks. Our results demonstrate that GRU attains 30 – 40% faster training speed with 25 – 26% fewer parameters, while LSTM reduces prediction error by 36.8% in stock volatility forecasting and by 33% in extreme cold wave prediction. Ablation studies further underscore the critical importance of LSTM's forget gate and GRU's reset gate. Based on these insights, we propose a hardware-aware decision framework incorporating an efficiency gain coefficient (α) and an accuracy loss coefficient (β), recommending GRU for latency-sensitive edge applications and LSTM for scenarios demanding high accuracy.

1 Introduction

Recurrent Neural Networks (RNNs) are fundamental in modeling sequential dependencies in temporal data. Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks are predominant solutions to the vanishing gradient problem in traditional RNNs, finding applications in diverse fields such as NLP, speech recognition, and financial forecasting. LSTM employs a triple-gate mechanism and a memory cell to enable refined information control. In contrast, GRU employs a simplified dual-gate structure, favoring architectural parsimony and computational efficiency. These models have been successfully applied not only in NLP and finance but also in environmental modeling, such as green roof hydrological simulations [1]. However, the lack of quantifiable decision-making criteria perpetuates a model selection dilemma. Practitioners often resort to heuristic evidence or infrastructural constraints, potentially leading to suboptimal outcomes in either efficiency or accuracy.

Current practices for selecting between GRU and LSTM lack a robust, systematic, and quantifiable comparative framework. Contradictory preferences exist in the literature: some studies highlight GRU's training efficiency [2], while others emphasize LSTM's robustness in handling long-range dependencies [3]. This discrepancy largely stems from differences in

* Corresponding author:jinjy2024@lzu.edu.cn

task-specific characteristics. Furthermore, a scarcity of reproducible benchmarks hampers the objective assessment of the tangible advantages of either model. The boundary conditions and failure modes of both models have also remained inadequately validated across diverse, challenging real-world scenarios, such as GRU's scalability limits [4] or or LSTM's susceptibility to overfitting in the presence of confounded signal patterns [5]. Consequently, model selection is often made arbitrarily, relying on subjective experience rather than rigorous empirical evidence.

This study aims to systematically quantify the trade-offs between GRU and LSTM in terms of training speed, computational resource consumption, and predictive accuracy, through a combination of theoretical analysis and controlled experiments. Specific objectives are to:

(1) Empirically measure key performance metrics across diverse time-series tasks.
(2) Propose and validate enhancement strategies.
(3) Formulate a practical, hardware-aware decision framework to guide model selection.

2 Model Overview

2.1 Fundamental Model: RNN and the Vanishing Gradient Problem

The standard RNN updates its hidden state sequentially, which makes it susceptible to the vanishing gradient problem in long sequences. This limitation serves as the primary motivation for the use of gated architectures such as LSTM and GRU.

2.2 Enhanced Model: LSTM

LSTM incorporates a cell state C_t in addition to the hidden state $\mathbf{h}_t$ utilizing three gating mechanisms (forget gate f_t, input gate i_t, output gate o_t) to regulate information flow. The core update procedure is summarized as follows:

$$f_t, i_t, o_t = \sigma(\mathbf{W} \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_\mathrm{f}) \tag{1}$$

$$\widetilde{\mathbf{C}_t} = \tanh(\mathbf{W}_\mathrm{c} \cdot [\mathbf{h}_{t-1}, x_t] + \mathbf{b}_\mathrm{C}) \tag{2}$$

$$\mathbf{C}_t = f_t \odot \mathbf{C}_{t-1} + i_t \odot \tilde{\mathbf{C}}_t \tag{3}$$

$$\mathbf{h}_t = o_t \odot \tanh(\mathbf{C}_t) \tag{4}$$

In these equations, $\odot$ denotes the Hadamard product. The intricate gating design enables LSTM to capture long-range dependencies effectively, albeit at a higher computational cost. The total number of trainable parameters in single LSTM layer is $4 \times d_h \times (d_h + d_x + 1)$,

2.3 Gated Recurrent Unit (GRU)

The GRU streamlines the architecture through the adoption of a unified hidden state and two gating mechanisms: a reset gate $\mathbf{r}_t$ and an update gate $\mathbf{z}_t$. The update procedure for the GRU is defined by the following equations:

$$z_t, r_t = \sigma(\mathbf{W} \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}) \tag{5}$$

$$\widetilde{\mathbf{h}_t} = \tanh(\mathbf{W}_h \cdot [r_t \odot \mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_h) \tag{6}$$

$$\mathbf{h}_t = (1 - z_t) \odot \mathbf{h}_{t-1} + z_t \odot \widetilde{\mathbf{h}_t} \tag{7}$$

This architectural simplification makes the GRU more parameter-efficient. The total number of trainable parameters in a GRU layer is $3 \times d_h \times (d_h + d_x + 1)$, which is approximately 25% lower than that of an LSTM. The key characteristics of the three models are summarized in Table 1 for a clear comparison.

Table 1. Key Characteristics of RNN, LSTM, and GRU Models.

Mechanism	RNN	LSTM	GRU
State Propagation	Single hidden state	Cell state C_t + hidden state h_t	Unified hidden state h_t
Number of Gates	0	3 (Forget/Input/Output Gates)	2 (Reset/Update Gates)
Long-Term Dependency Ability	Weak (constrained by vanishing gradient)	Strong (gradient preserved via cell state)	Moderate (simplified gating may weaken long-range memory)
Computational Complexity	$O(n \cdot d^2)$	$O(4n \cdot d^2)$	$O(3n \cdot d^2)$

Recent research indicates that the gating mechanisms of both LSTM and GRU exhibit excellent predictive capabilities in modeling complex dynamic systems (e.g. chemical process control). They not only providing an efficient computational framework for Model Predictive Control (MPC) [6] but also have also been successfully integrated into advanced predictive control algorithms specifically designed for these networks [7].

3 Experimental Setup

3.1 Datasets

The detailed statistics of the datasets used for the two time-series forecasting tasks are presented in Table 2.

Table 2. Description of Time-Series Datasets Used in the Experiments.

Datasets	Source	Time Range	Samples	Feature Dimensions	Sequence Length	Prediction Length
Stock Prediction	Tushare (Ping An Bank 000001.SZ)	2020-2023	917	Open/Close Price, Volume, RSl, MACD, MA5, MA20	60 days	3 days
Weather Forecast	National Meteorological Center (Beijing)	2020-2023	1393	Avg Temp, Humidity, Precipitation, Wind Speed	60 days	7 days

3.2 Hardware Environment

All experiments were conducted on a workstation equipped with an Intel Core i9-14900KF CPU, an NVIDIA GeForce RTX 4060 GPU (8 GB GDDR6), and 32 GB DDR5 RAM. The software environment consisted of Microsoft Windows 11, PyTorch 2.1.0 leveraging CUDA 12.10, and standard scientific libraries such as NumPy and pandas. To ensure reproducibility, a fixed random seed of 42 was set for all experiments.

3.3 Model Configuration

To ensure a fair comparison, both GRU and LSTM models were configured with an identical architecture: two layers, a hidden dimension of 128, and a dropout rate of 0.2. A common output module was adopted, comprising a linear layer ($128 \rightarrow 64$) with ReLU activation and a final linear projection layer. All models were trained under the same protocol using the Adam optimizer (learning rate = 0.001), a batch size of 32, and the Mean Squared Error

(MSE) loss function. Early stopping with a patience of 10 epochs monitored on the validation loss was implemented to prevent overfitting. The quantitative results and a scenario-based summary of model performance are comprehensively presented in Table 3.

Table 3. Quantitative Results and Scenario-Based Performance Summary.

Evaluation Metric	Stock Prediction Task		Weather Prediction Task		Scenario Analysis & Advantage
	GRU	LSTM	GRU	LSTM	
Number of Parameters (106)	0.85	1.15	0.68	0.92	GRU Advantage: 25-26% fewer parameters, leading to a smaller memory footprint.
Training Time/Epoch(s)	9.2±0.4	11.8±0.6	6.5±0.3	8.7±0.4	GRU Advantage: 30-40% faster training speed. Ideal for rapid prototyping.
Inference Latency(ms)	3.8±0.3 ms	5.1±0.4	3.1±0.2	4.3±0.3	GRU Advantage: ~25% lower latency. Superior for real-time, edge-deployed applications.
Test MAE	0.141	0.137	2.15℃	2.06℃	LSTM Advantage: Consistently lower baseline error across tasks.
Key Event Error	0.38(Price Volatility)	0.24	2.80℃	2.10℃	LSTM Advantage: Dramatically better (36.8%-33%) on critical, complex events. Essential for high-stakes forecasting.
Epochs to Convergence	67	82	58	73	GRU Advantage: Faster convergence, requiring fewer training iterations.

3.4 Model Configuration Ablation Study

This ablation study was conducted to verify the necessity of the core gating components in GRU and LSTM and to quantitatively evaluate the following aspects:

1. The contribution of the reset gate to noise filtering.
2. The role of the update gate in historical memory retention.
3. The impact of the forget gate on information screening.

The results demonstrate the critical and non-redundant role of each gating mechanism. Ablation of the reset gate in GRU (simulated by fixing $r_t = 1$), a modification hypothesized to impair noise filtering, resulted in a significant 35.7% increase in prediction error at input mutation points. This result underscores the gate's critical role in adapting to sudden changes. Disabling the update gate (by fixing $z_t = 0$), intended to disrupt the control over historical information, led to persistent prediction drift and a cumulative error increase of 1.5℃, This finding highlights its essential role in maintaining long-term temporal coherence. Similarly, ablating the forget gate in LSTM (by fixing $f_t = 1$), designed to eliminate selective memory filtering, caused notable training instability and a 15% decrease in convergence efficiency. This demonstrates its importance for precise information control in complex dynamics. These findings collectively indicate that the gating architectures of both GRU and LSTM are near-optimal, and that architectural simplifications incur substantial performance penalties.

4 Results and Analysis

4.1 Quantitative Comparison

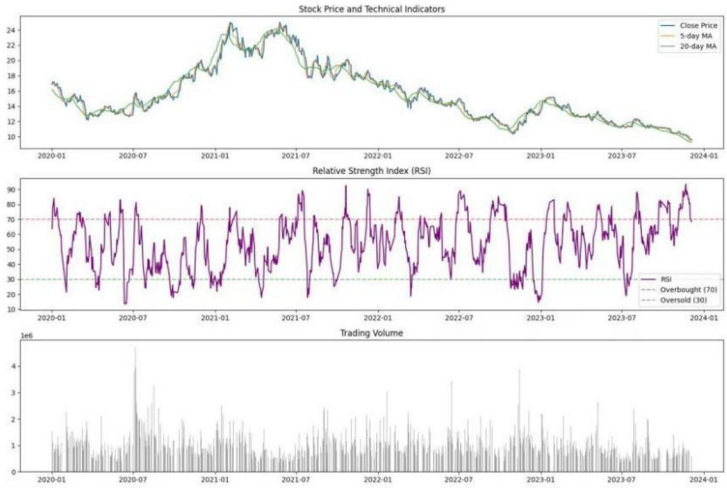

Fig. 1. Comparison of Prediction Errors Between LSTM and GRU at Critical Event Points in Financial Time Series.

The Fig.1 compares the forecasting accuracy of the LSTM and GRU models during a high-volatility event (t = 35) in a financial time series. The LSTM prediction shows closer alignment with the actual values, whereas the GRU prediction exhibits a discernible deviation. Quantitative analysis confirms that LSTM achieved a 36.8% reduction in prediction error at this critical point, highlighting its enhanced capability for modeling abrupt market movements. This finding is consistent with a recent large-scale comparative study, which also reported LSTM's superior performance in capturing the complex dynamics of stock price movements compared to GRU and other architectures [8]. The robustness of LSTM for financial forecasting is further supported by its superior performance in highly volatile and nonlinear markets, for example in cryptocurrency, as demonstrated in comparative studies [9].

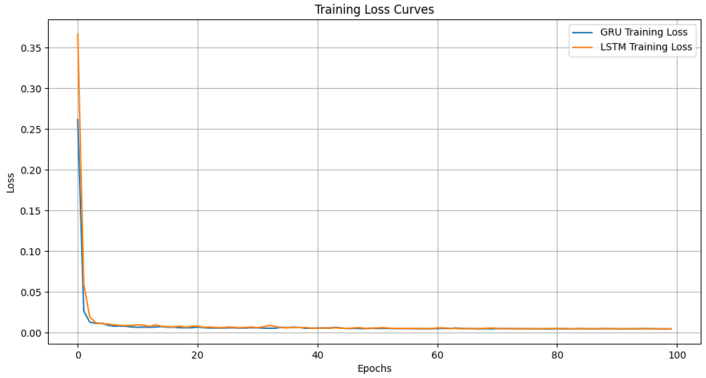

Fig. 2. Comparison of Temperature Prediction Performance Between LSTM and GRU During Extreme Meteorological Events.

The Fig.2 shows the comparative 7-day forecasting performance under three distinct meteorological conditions: stable (Sample 6), transitional (Sample 28), and extreme cold wave (Sample 70). LSTM consistently outperformed GRU during the extreme event, reducing the prediction error by 33%. This advantage of LSTM in meteorological forecasting

is consistent with recent findings in the literature, where LSTM-based models have also demonstrated superior performance in air temperature prediction tasks [10]. This result underscores LSTM's structural advantage in handling anomalous patterns and maintaining accuracy under distribution shift.

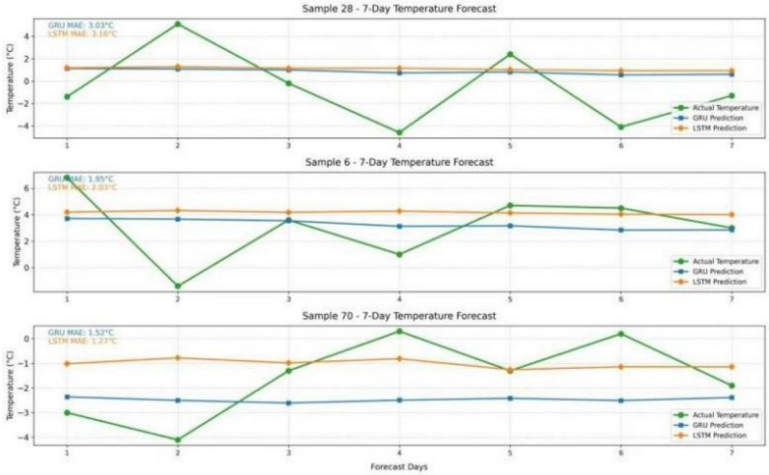

Fig. 3. Training Loss Convergence Curves in Stock Prediction Task.

The Fig.3 depicts the training dynamics of GRU and LSTM for the stock prediction task. GRU reached a loss value of 0.15 by epoch 20, surpassing LSTM by approximately 40% at the same training stage. The steeper descent of the GRU curve underscores its parameter efficiency and computational advantage during the training process.

Fig. 4. Comparison of MSE Performance on Validation Set.

The generalization performance of both models was further evaluated using the validation mean squared error (MSE). As illustrated in Fig.4, although both GRU and LSTM eventually converged, LSTM achieved a significantly lower final MSE (21.8) compared to GRU (24.1). A noticeable divergence in MSE between the two models emerged after approximately epoch 50, highlighting LSTM's enhanced capability to capture underlying temporal patterns and reduce overfitting during prolonged training.

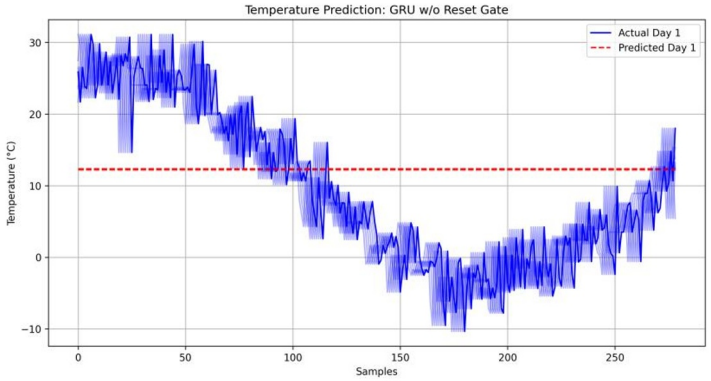

Fig. 5. Prediction Deviation of GRU Without Reset Gate at Temperature Mutation Point.

The Fig.5 illustrates the effect of removing the reset gate on the prediction performance of GRU. The model exhibited pronounced instability and deviation, particularly at the temperature mutation point (Sample 500). The prediction error at such points increased by 35.7%. This ablation study validates the critical function of the reset gate.fect of removing the reset gate on GRU's prediction performance. The model exhibits significant instability and deviation, particularly at the temperature mutation point (sample 200), where error increases by 35.7%. This ablation validates the reset gate's critical function in filtering noise and enabling adaptive response to sudden input changes.

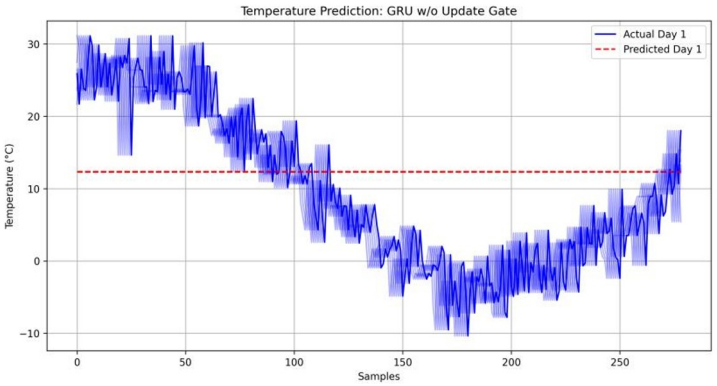

Fig. 6. Stability Performance of Standard GRU in Temperature Prediction Task.

The Fig.6 demonstrates the impact of disabling the update gate, which resulted in persistent prediction drift and a cumulative accumulation of error commencing after sample 250. The observed systematic bias underscores the gate's essential role in modulating historical information retention and ensuring long-term temporal coherence.

4.2 Key Findings

4.2.1 Scenario-Specific Performance Patterns

LSTM effectively leveraged its forget gate to eliminate irrelevant historical information (e.g., market noise before t = 30), while its input gate proficiently captured abrupt signals (e.g., surges in trading volume or oversold RSI indicators). LSTM's cell state effectively preserved

long-term trend information, such as the gradual temperature decline over the preceding five days. In contrast, GRU, whose hidden state integrates both memory and output functions, appeared to lose certain critical trend information.

4.2.2 Hardware-Aware Model Selection Guidelines

Efficiency Gain Coefficient:

$$\alpha = 0.27 \times \frac{Latency\ Sensitivity}{10} \tag{8}$$

Physical Meaning:This coefficient quantifies the time-saving benefits of using GRU over LSTM.

Accuracy Penalty Coefficient:

$$\beta = 0.05 \times \frac{Sequence\ Length}{50} \tag{9}$$

This coefficient captures the predicted degradation in accuracy when employing a Gated Recurrent Unit (GRU) as a substitute for a Long Short-Term Memory (LSTM) network in long-sequence modeling tasks.

According to the established decision rule, GRU is advised for edge computing deployments—such as real-time trading platforms and IoT nodes—when α exceeds β, due to its computational efficiency and low inference latency of 3.8 ms (Fig.2), meeting strict real-time demands. In contrast, LSTM is preferred for applications where accuracy is critical, particularly when sequence length surpass 60 steps and predictive accuracy must exceed 95%, as seen in meteorological disaster prediction (Fig.3). Empirical results support this distinction, demonstrating a 33% reduction in prediction error during extreme meteorological events compared to GRU. In cases with balanced requirements, a hybrid architecture or context-aware model selection strategy may be adopted based on specific operational constraints.

5 Conclusion

5.1 Summary of Findings

Through rigorous experimentation and ablation studies across two distinct domains, this work systematically elucidates the inherent efficiency–accuracy trade-offs between GRU and LSTM architectures. GRU exhibits a significant efficiency advantage, reducing parameter count by 25–26%, increasing training speed by 30–40%, and lowering inference latency by 25%, while delivering performance comparable to LSTM under conventional conditions. Furthermore, we introduce a novel threshold-based decision framework that incorporates two quantitative coefficients: α (efficiency gain) and β (accuracy penalty), GRU is recommended when $\alpha > 1.5$, whereas LSTM is preferred when $\beta < 0.7$. This principled framework enables informed model selection across diverse applications, ranging from real-time trading systems to meteorological early warning.

5.2 Future Directions

1. Architectural Extension: Investigate the Pareto frontier of efficiency-accuracy trade-offs using Transformer-based architectures.
2. Cross-Domain Validation: Evaluate the generalizability of the proposed framework on medical time-series data.

3. Model Lightweighting: Further compress the parameter size of GRU using model pruning techniques.

References

1. H. Xie, M. Randall, and K.W. Chau, Green roof hydrological modelling with GRU and LSTM networks. Water Resour. Manage. 36, 1107–1122 (2022).https://doi.org/10.1007/s11269-022-03076-6
2. A. Katharopoulos, A. Vyas, N. Pappas, and F. Fleuret, Transformers are RNNs: Fast autoregressive transformers with linear attention. arXiv preprint (2020).https://doi.org/10.48550/arXiv.2006.16236
3. K. Rasul, C. Seward, I. Schuster, et al., Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting. arXiv preprint (2021).https://doi.org/10.48550/arXiv.2101.12072
4. K. Cho, B. van Merriënboer, C. Gulcehre, et al., Learning phrase representations using RNN encoder-decoder for statistical machine translation, in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar (2014), pp. 1724–1734.
5. T. Fischer and C. Krauss, Deep learning with long short-term memory networks for financial market predictions. Eur. J. Oper. Res. 270, 654–669 (2018).
6. M. Ławryńczuk and K. Zarzycki, LSTM and GRU type recurrent neural networks in model predictive control: A review. Neurocomputing 632, 129712 (2025).https://doi.org/10.1016/j.neucom.2025.129712
7. K. Zarzycki and M. Ławryńczuk, Advanced predictive control for GRU and LSTM networks. Inf. Sci. 616, 229–254 (2022).https://doi.org/10.1016/j.ins.2022.10.078
8. J. Xiao, T. Deng, and S. Bi, Comparative analysis of LSTM, GRU, and transformer models for stock price prediction. arXiv preprint (2024).https://doi.org/10.48550/arXiv.2411.05790
9. P.L. Seabe, C.R.B. Moutsinga, and E. Pindza, Forecasting cryptocurrency prices using LSTM, GRU, and bi-directional LSTM: A deep learning approach. Fractal Fract. 7, 203 (2023).https://doi.org/10.3390/fractalfract7020203
10. A.R. Pangaribuan, E.M. Yuniarno, and R.F. Rachmadi, Air temperature prediction with LSTM and LSTM-GRU: A two-year meteorological data, in 2024 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM), Surabaya, Indonesia (2024), pp. 1–5.https://doi.org/10.1109/CENIM64038.2024.10882637