Proceedings of the CONF-MLA 2026 Symposium: Explainable Computing, Modeling & Data Science in Complex Systems
DOI: 10.54254/2755-2721/2026.GF37837

Temporal Convolutional Networks for Stock Movement Prediction: Normalization, Parameter Sensitivity, and Robustness Evaluation

Yongyuan Zhang

WeBank Institute of Financial Technology, Shenzhen University, Shenzhen, China
zhangyongyuan555@gmail.com

Abstract. Daily stock-movement prediction is difficult because its signal-to-noise ratio is low, market regimes change, and neural results vary with initialization. This study develops an auditable price-only classification pipeline under the fixed StockNet temporal protocol. Target returns of at least 0.55% are labeled upward, returns below -0.50% are labeled downward, and intermediate targets are excluded. Volume is log-transformed, and all five channels are standardized with training-period statistics only. Majority, multilayer perceptron, long short-term memory, and temporal convolutional network (TCN) models are compared on identical samples. Validation Macro-F1 guides a 3×3 sensitivity analysis of history length and kernel width; only configurations whose input is no shorter than the theoretical receptive field are eligible for selection. The locked TCN uses a 20-day window and kernel width 3. Across 20,325 training, 2,555 validation, and 3,720 test samples, five TCN seeds achieve Accuracy 0.5283 ± 0.0092, Macro-F1 0.5163 ± 0.0171, Matthews correlation coefficient 0.0678 ± 0.0162, and ROC-AUC 0.5356 ± 0.0068. Every run has accuracy above 0.50 and positive correlation, but long short-term memory is slightly stronger on three metrics. The evidence supports limited, repeatable out-of-sample discrimination rather than comprehensive TCN superiority.

Keywords: stock movement prediction, financial time series, temporal convolutional network, channel-wise standardization, robustness evaluation

1. Introduction

Predicting the next trading day's direction from historical prices is a weak-signal classification problem. Daily returns jointly reflect macroeconomic news, firm information, liquidity, and investor behavior, so their conditional distribution changes across market regimes. A credible experiment must therefore use a strict temporal holdout, prevent preprocessing leakage, and distinguish genuine two-class discrimination from apparent accuracy caused by class imbalance.

StockNet defines asymmetric thresholds and fixed training, validation, and test dates for stock-direction prediction [1]. Its price branch reports 54.96% accuracy and an MCC of 0.0165, illustrating that price-only performance is typically only modestly above chance. Accuracy alone is insufficient: a constant majority-class predictor can exceed 50% while its Macro-F1 and MCC

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/2026.GF37837

reveal no balanced discrimination. This study consequently reports Accuracy, Macro-F1, MCC, and ROC-AUC together.

Two design choices are central. First, price variables and volume have different scales; all transformations must be fitted on training data only. Second, TCN window length, kernel width, and dilation jointly determine the receptive field. A receptive field longer than the input makes many positions depend on padding and weakens the intended temporal interpretation. The experiment therefore (i) reproduces the StockNet labeling and temporal split, (ii) verifies channel-wise training-only standardization, (iii) compares Majority, MLP, LSTM, and TCN under common training rules, and (iv) selects the TCN window and kernel on validation data before reporting five-seed test results.

2. Related work

2.1. Price sequences and stock-direction classification

Financial direction prediction is constrained by nonstationarity and low signal-to-noise ratios. Fischer and Krauss show that LSTM can learn useful representations in a large stock universe, but the economic result depends on the evaluation period and transaction assumptions [2]. StockNet excludes target returns in $[-0.5\%, 0.55\%)$, reducing random fluctuations near zero at the class boundary [1]. Its published TechnicalAnalyst result is retained as a task-scale reference rather than treated as a reproduced baseline because its representation and objective differ from those used here.

2.2. Deep time-series classification

LSTM gates mitigate vanishing gradients and recursively summarize an input history [3]. TCNs instead combine causal and dilated convolutions. Bai et al. find that generic convolutional sequence models can compete with recurrent alternatives and stress the importance of an adequate receptive field [4]. Standardized comparisons across datasets also show that conclusions are sensitive to preprocessing and evaluation design [5, 6]. For financial data, adjacent dates must not be randomly distributed across training and test sets. The present models therefore share samples, transformations, objectives, and temporal splits.

2.3. Evaluation of weak signals

Macro-F1 weights the two class-specific F1 scores equally. MCC uses all four confusion-matrix cells and remains informative under imbalance; values near zero indicate little association between predictions and labels [7]. ROC-AUC assesses probability ranking independently of a fixed threshold. Model selection is restricted to validation data, and multiple seeds quantify initialization and mini-batch uncertainty rather than provide a pool from which to select a favorable run.

3. Methodology

3.1. Task definition and temporal windows

Let $r_{s,t}$ be the target return of stock s on day t. The StockNet label is

DOI: 10.54254/2755-2721/2026.GF37837

$$y_{s,t} = \begin{cases} 1, & r_{s,t} \geq 0.0055, \\ 0, & r_{s,t} < -0.005, \\ \varnothing, & -0.005 \leq r_{s,t} < 0.0055, \end{cases} \tag{1}$$

where $\varnothing$ denotes an excluded neutral target. Neutral days may remain inside another observation's history. For each valid target, the preceding L days form

$$\boldsymbol{X}_{s,t} = \left[\boldsymbol{x}_{s,t-L}, \ldots, \boldsymbol{x}_{s,t-1}\right]^{\top} \in \mathbb{R}^{L \times 5}. \tag{2}$$

The history-end date is required to precede the target date. Candidate windows are 10, 15, and 20 trading days.

3.2. Features and training-only standardization

The daily feature vector is

$$\boldsymbol{x}_{s,t} = \left(o_{s,t}, h_{s,t}, l_{s,t}, r^{c}_{s,t}, \log\left(1 + v_{s,t}\right)\right)^{\top}, \tag{3}$$

where o, h, l are preprocessed open, high, and low variables, r^{c} is the historical close return, and v is volume. The log transform reduces volume skew. For channel j,

$$\widetilde{x}_{s,t,j} = \frac{x_{s,t,j} - \mu_j^{\mathrm{tr}}}{\sigma_j^{\mathrm{tr}}}, \qquad j = 1, \ldots, 5, \tag{4}$$

using means and standard deviations fitted only to training-history rows. Validation and test windows reuse these parameters. No channel has zero variance; standardized training means are below 3.0×10^{-16} in absolute value and standard deviations equal 1 to numerical precision.

3.3. Baselines and TCN architecture

Majority predicts the training-set majority class. The MLP flattens the $L \times 5$ input and uses hidden widths 128 and 64. The LSTM has one 64-dimensional recurrent layer and a $64 \to 32 \to 2$ head. The TCN contains two residual blocks with 32 and 64 channels, dilations 1 and 2, two causal convolutions per block, ReLU, and dropout. MLP, LSTM, and TCN contain 21,314, 20,322, and 26,626 parameters, respectively.

A causal convolution is

$$z_q^{(\ell)} = \textstyle\sum_{a=0}^{K-1} w_a^{(\ell)} x_{q-d_\ell a}^{(\ell-1)}, \tag{5}$$

and the theoretical receptive field is

$$R = 1 + 2(K-1)(1+2). \tag{6}$$

Kernel widths 2, 3, and 5 therefore give $R = 7, 13, 25$. Only candidates satisfying $L \geq R$ are eligible for final selection. After the second block, the terminal causal state is used instead of temporal averaging:

DOI: 10.54254/2755-2721/2026.GF37837

$$\boldsymbol{h}_{s,t}^{\mathrm{TCN}} = \boldsymbol{H}_{:,L}, \qquad \boldsymbol{H} \in \mathbb{R}^{64\times L}. \tag{7}$$

This state represents the latest complete history and avoids equally weighting early positions dominated by left padding.

3.4. Training and metrics

The objective is class-balanced weighted cross-entropy,

$$\mathscr{L} = -\sum_{i=1}^{N}\sum_{k=0}^{1} \omega_k y_{i,k} \log p_{i,k}, \qquad \omega_k = \frac{N}{2N_k}. \tag{8}$$

AdamW [8] uses learning rate 10^{-3}, batch size 512, dropout 0.2, and at most 30 epochs. Training stops after six epochs without validation Macro-F1 improvement and restores the best checkpoint. Sensitivity analysis uses seed 42; final models use seeds 11, 22, 33, 44, and 55. MCC is

$$\mathrm{MCC} = \frac{TP\cdot TN - FP\cdot FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}. \tag{9}$$

Saved logits and labels are used to recompute all metrics independently; discrepancies are below 10^{-10}.

4. Data and experimental design

4.1. Data and temporal split

The public StockNet price table[1] contains 108,501 rows, 88 stocks, and 1,257 dates from 2012-09-05 to 2017-09-01. Formal samples are restricted to 2014-01-01 through 2015-12-31. Target-date intervals are 2014-01-01–2015-07-31 for training, 2015-08-01–2015-09-30 for validation, and 2015-10-01–2015-12-31 for testing. A 20-day window yields the class counts in Table 1 after excluding 16,802 neutral targets.

Table 1. Fixed StockNet temporal splits and class composition

Split	Target-date range	Down	Up	Total
Training	2014-01-01 to 2015-07-31	10,014	10,311	20,325
Validation	2015-08-01 to 2015-09-30	1,416	1,139	2,555
Test	2015-10-01 to 2015-12-31	1,812	1,908	3,720

Figure 1 shows that validation has a larger downward share, whereas testing has a slight upward majority. This shift motivates balanced metrics.

[1] Repository: https://github.com/yumoxu/stocknet-dataset; Parquet mirror: https://huggingface.co/datasets/nomnomshark41/stocknet-dataset.

DOI: 10.54254/2755-2721/2026.GF37837

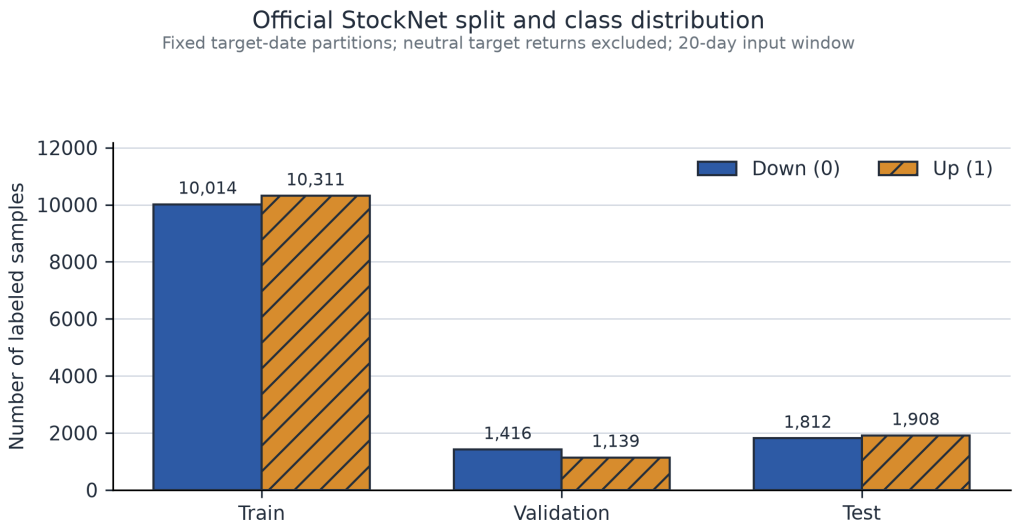

Figure 1. Fixed StockNet temporal splits and binary-class sample distribution

4.2. Validation-first sensitivity design

Nine TCN configurations combine $L \in \{10, 15, 20\}$ with $K \in \{2, 3, 5\}$. Each is trained with seed 42 and ranked by validation Macro-F1 after enforcing $L \geq R$; the test set is not evaluated during selection. Table 2 shows that the 10-day, width-3 result is ineligible because $R = 13 > L$. Among valid candidates, the 20-day, width-3 configuration has the highest Macro-F1, 0.5506, and is locked for final testing.

Table 2. Validation sensitivity of TCN history window and kernel width

Window L	Kernel K	Field R	Valid	Accuracy	Macro-F1
10	2	7	Yes	0.5432	0.5332
10	3	13	No	0.5562	0.5540
10	5	25	No	0.5440	0.5438
15	2	7	Yes	0.5503	0.5307
15	3	13	Yes	0.5495	0.5493
15	5	25	No	0.5272	0.5227
20	2	7	Yes	0.5205	0.5204
20	3	13	Yes	0.5507	0.5506
20	5	25	No	0.5170	0.5102

DOI: 10.54254/2755-2721/2026.GF37837

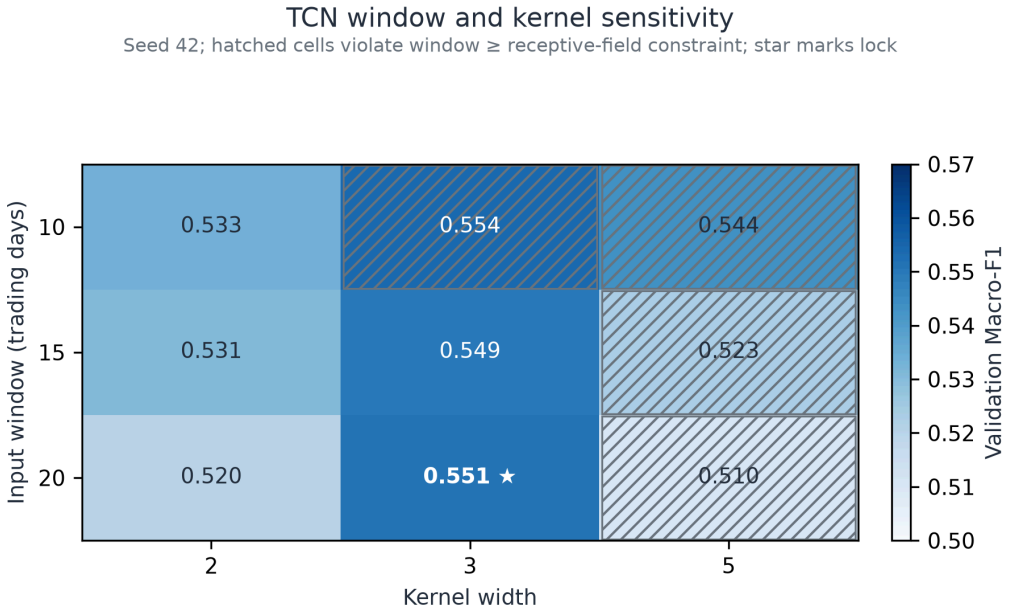

Figure 2. Validation Macro-F1 over TCN input window and kernel width

5. Experimental results

5.1. Baseline comparison

Table 3 shows that Majority reaches 0.5129 accuracy but zero MCC, confirming that accuracy above 50% can arise without two-class discrimination. Neural models have positive mean MCC and ROC-AUC above 0.53. LSTM is slightly stronger than TCN in Accuracy, MCC, and ROC-AUC, while MLP has the highest Macro-F1. The results therefore support competitive but not superior TCN performance.

Table 3. Test results for StockNet price-only baselines

Model	Accuracy	Macro-F1	MCC	ROC-AUC
Majority	0.5129	0.3390	0.0000	0.5000
MLP	0.5249 ± 0.0052	0.5238 ± 0.0058	0.0529 ± 0.0097	0.5373 ± 0.0030
LSTM	0.5316 ± 0.0063	0.5158 ± 0.0154	0.0787 ± 0.0114	0.5464 ± 0.0016
TCN	0.5283 ± 0.0092	0.5163 ± 0.0171	0.0678 ± 0.0162	0.5356 ± 0.0068
TechnicalAnalyst [1]	0.5496	–	0.0165	–

The absolute gaps among learned models are modest. LSTM exceeds TCN by 0.0033 in Accuracy, 0.0109 in MCC, and 0.0108 in ROC-AUC, whereas TCN is 0.0005 higher in Macro-F1. These differences are comparable with the reported seed dispersion, so the ranking depends on both the selected metric and the initialization. The TechnicalAnalyst result further illustrates why Accuracy should not be read alone: its higher Accuracy is paired with a much smaller MCC than those of the neural baselines.

DOI: 10.54254/2755-2721/2026.GF37837

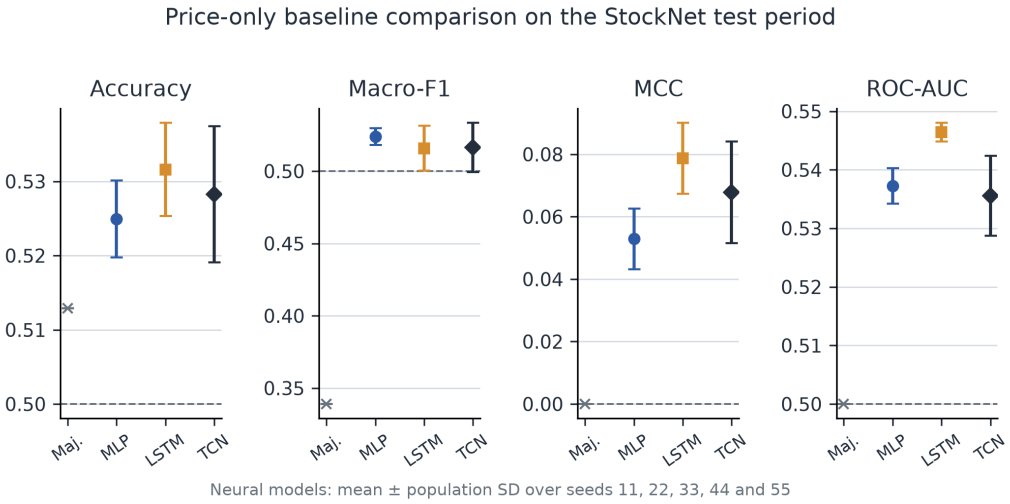

Figure 3. Five-seed means and standard deviations of price-only baselines

5.2. Seed-level robustness and representative diagnostics

TCN Accuracy ranges from 0.5169 to 0.5417, MCC from 0.0507 to 0.0933, and ROC-AUC from 0.5261 to 0.5468. All five runs exceed chance on these directional checks, although Macro-F1 variability remains material (Table 4). The best single run is not substituted for the mean.

Table 4. Seed-level results for the locked TCN configuration

Seed	Best epoch	Accuracy	Macro-F1	MCC	ROC-AUC
11	4	0.5169	0.4888	0.0537	0.5261
22	15	0.5191	0.5057	0.0507	0.5320
33	6	0.5417	0.5353	0.0933	0.5468
44	8	0.5331	0.5213	0.0796	0.5357
55	13	0.5306	0.5306	0.0618	0.5371
Mean	–	0.5283	0.5163	0.0678	0.5356
Population SD	–	0.0092	0.0171	0.0162	0.0068

Dispersion is visible but not dominated by a single failed run. Seed 33 is strongest on all four reported metrics, while seeds 11 and 22 occupy the lower end of the distribution. Across the five runs, the observed ranges are 0.0248 for Accuracy, 0.0465 for Macro-F1, 0.0426 for MCC, and 0.0207 for ROC-AUC. Reporting the mean and population standard deviation therefore gives a more defensible summary than selecting the best seed or presenting one initialization as typical.

DOI: 10.54254/2755-2721/2026.GF37837

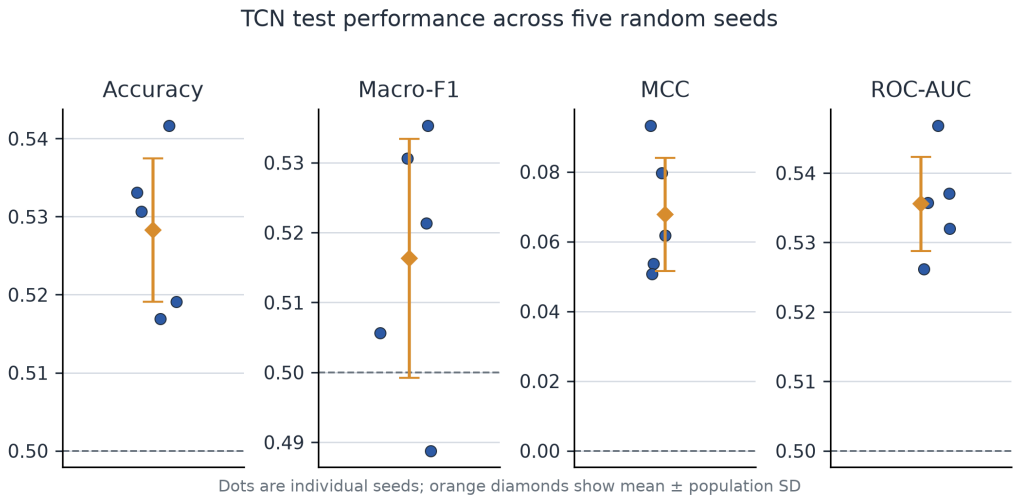

Figure 4. Test-metric distribution of the locked TCN across five seeds

Seed 55 is used for diagnostics because its MCC is closest to the five-seed mean. Its confusion matrix is $\begin{bmatrix} 980 & 832 \\ 914 & 994 \end{bmatrix}$, corresponding to recalls of approximately 54.1% for Down and 52.1% for Up. Both classes are predicted without collapse, and neither class alone explains the positive MCC. Validation loss fluctuates despite declining training loss, while validation Macro-F1 selects epoch 13; Figure 5 therefore also shows why the final epoch is not automatically the preferred checkpoint.

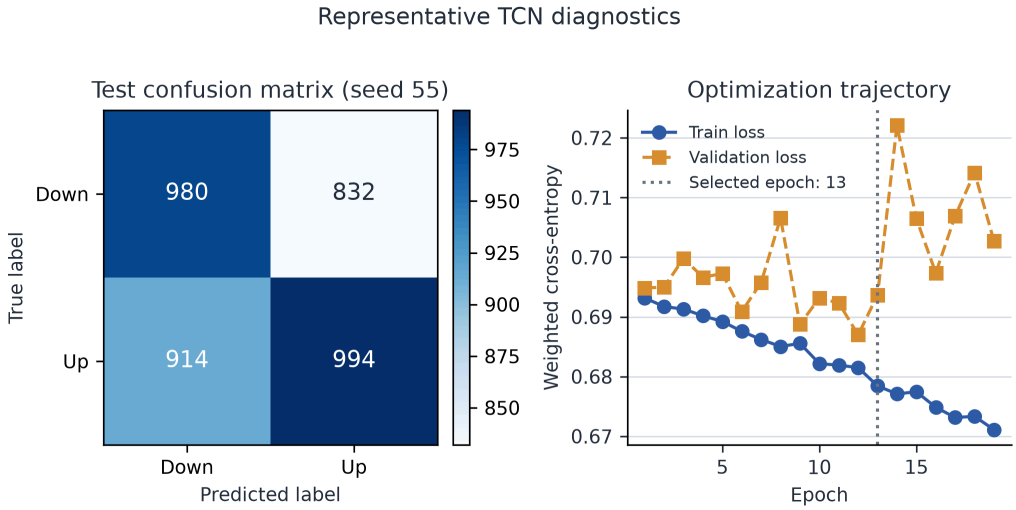

Figure 5. Confusion matrix and training trajectory for representative TCN seed 55

6. Discussion

The mean TCN accuracy of 52.83% is consistent with a weak price-only signal. Positive MCC and ROC-AUC values across all seeds indicate repeatable statistical association, but their magnitude remains small. A three-month test period cannot establish economic value, and statistical discrimination does not directly specify a profitable trading rule. Such a claim would require probability calibration, an explicit decision threshold, rolling retraining, executable position rules,

DOI: 10.54254/2755-2721/2026.GF37837

transaction costs, market impact, turnover limits, and evaluation over several independent market regimes.

Sensitivity results also show no monotonic benefit from longer windows or wider kernels. Kernel width 5 produces $R = 25$, longer than every candidate input, and offers no stable validation improvement. The locked 20-day, width-3 model is a validation-specific choice, not a universal optimum. LSTM may benefit from recursively absorbing all 20 observations, whereas the selected TCN's theoretical field covers 13 recent positions.

Training-only channel standardization is equally important. It improves optimization while preventing future mean or volatility information from entering the model. Window-wise normalization would answer a different question because it can remove level and trend information. The reported results should therefore be interpreted as properties of the stated preprocessing pipeline rather than of the TCN architecture in isolation. Reproducible comparisons require the same split dates, neutral-label rule, window construction, scaling provenance, checkpoint criterion, and seed policy.

Several limitations constrain generalization. The experiment uses one equity universe, one historical interval, and a single three-month test block. Removing neutral targets changes both the economic question and the effective class distribution, while overlapping windows may produce highly related observations within each chronological split. Five seeds reveal material optimization variability but are insufficient for precise uncertainty intervals or formal claims of model dominance. The baseline comparison is consequently descriptive: it supports a stable, weak signal and rejects a broad claim of TCN superiority.

Stronger evidence would separate optimization uncertainty from temporal uncertainty. A rolling-origin design could evaluate multiple nonoverlapping test blocks while retaining the same preprocessing and model-selection rules. Baselines should be trained on identical windows and random seeds, and comparisons should use paired predictions rather than independent summary scores. Block bootstrap intervals grouped by date or stock-date cluster would better respect dependence than treating overlapping samples as independent. Class-specific recall, calibration, threshold sensitivity, and turnover should accompany aggregate classification metrics before any economic interpretation is attempted.

Future work should first repeat the same pipeline across nonoverlapping periods, additional markets, and multiple test horizons, with confidence intervals constructed from both temporal blocks and random seeds. Cross-stock information is a promising extension: CMIN demonstrates the value of causality-enhanced inter-stock relations [9]. Copula models could separate marginals from dependence under Sklar's theorem [10], compare Gaussian, Student- t, Clayton, and Gumbel families [11, 12], and allow time-varying dependence through conditional-correlation ideas [13]. High-frequency extensions require event-time labels, realistic execution assumptions, and order-book features rather than the daily protocol; DeepLOB provides an established benchmark [14].

7. Conclusion

This study presents a reproducible StockNet price-only evaluation of TCNs. Labels and temporal splits follow the fixed protocol, volume is log-transformed, and every channel is standardized using training data only. Validation analysis locks a 20-day input and kernel width 3 under an explicit receptive-field constraint. Across five seeds, TCN obtains Accuracy 0.5283, Macro-F1 0.5163, MCC 0.0678, and ROC-AUC 0.5356. All MCC values are positive, but LSTM is slightly stronger on three metrics and the absolute differences are small relative to seed variability. The appropriate conclusion is therefore limited, repeatable discrimination rather than general TCN superiority or demonstrated

DOI: 10.54254/2755-2721/2026.GF37837

trading profitability. Robust financial classification should disclose temporal protocols, preprocessing provenance, baseline comparisons, checkpoint selection, model-selection rules, and random-seed dispersion. These controls make modest predictive results easier to reproduce and harder to overstate.

References

[1] Xu, Y., & Cohen, S. B. (2018). Stock movement prediction from tweets and historical prices. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 1970–1979). Association for Computational Linguistics. https://doi.org/10.18653/v1/P18-1183

[2] Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research, 270*(2), 654–669. https://doi.org/10.1016/j.ejor.2017.11.054

[3] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

[4] Bai, S., Kolter, J. Z., & Koltun, V. (2018). An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv:1803.01271. https://doi.org/10.48550/arXiv.1803.01271

[5] Ismail Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., & Muller, P.-A. (2019). Deep learning for time series classification: A review. *Data Mining and Knowledge Discovery, 33*(4), 917–963. https://doi.org/10.1007/s10618-019-00619-1

[6] Bagnall, A., Lines, J., Bostrom, A., Large, J., & Keogh, E. (2017). The great time series classification bake off: A review and experimental evaluation of recent algorithmic advances. *Data Mining and Knowledge Discovery, 31*(3), 606–660. https://doi.org/10.1007/s10618-016-0483-9.

[7] Matthews, B. W. (1975). Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica et Biophysica Acta, 405*(2), 442–451. https://doi.org/10.1016/0005-2795(75)90109-9.

[8] Loshchilov, I., & Hutter, F. (2019). Decoupled weight decay regularization. In *7th International Conference on Learning Representations*. OpenReview.net.

[9] Luo, D., Liao, W., Li, S., Cheng, X., & Yan, R. (2023). Causality-guided multi-memory interaction network for multivariate stock price movement prediction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 12164–12176). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.acl-long.679.

[10] Sklar, A. (1959). Fonctions de répartition à dimensions et leurs marges. *Publications de l'Institut de Statistique de l'Université de Paris, 8*, 229–231.

[11] Nelsen, R. B. (2006). *An introduction to copulas* (2nd ed.). Springer. doi:10.1007/0-387-28678-0.

[12] Patton, A. J. (2006). Modelling asymmetric exchange rate dependence. *International Economic Review, 47*(2), 527–556. doi:10.1111/j.1468-2354.2006.00387.x.

[13] Engle, R. F. (2002). Dynamic conditional correlation: A simple class of multivariate generalized autoregressive conditional heteroskedasticity models. *Journal of Business & Economic Statistics, 20*(3), 339–350. doi:10.1198/073500102288618487.

[14] Zhang, Z., Zohren, S., & Roberts, S. (2019). DeepLOB: Deep convolutional neural networks for limit order books. *IEEE Transactions on Signal Processing, 67*(11), 3001–3012. doi:10.1109/TSP.2019.2907260.