Cureus Journal of Computer Science
https://doi.org/10.7759/s44389-026-00383-6

Original Article

Stacking a BiLSTM-Attention Network With a Pretrained Time-Series Foundation Model and Gradient-Boosted Residual Correction for Short-Term Electricity Load Forecasting: A Comparative Evaluation on the Tetouan City Dataset

Khushvir Singh [1, ✉], Anurag Tewari [1]

1. *School of Computing and AI, Lovely Professional University, Jalandhar, IND*

Received: September 03, 2026 | Review began: September 07, 2026 | Review ended: September 29, 2026
| Published: November 20, 2026

© **Copyright** 2026
This is an open access article distributed under the terms of the Creative Commons Attribution License CC-BY 4.0., which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Abstract

This study investigates whether adding a pretrained time-series foundation model (Chronos-Bolt) as a second forecaster alongside a domain-specific Bidirectional Long Short-Term Memory (BiLSTM)-attention network, combined through a learned gradient-boosted correction layer, improves short-term electricity load forecasting compared with a recently published hybrid LSTM-XGBoost baseline using the Tetouan City dataset. A BiLSTM with additive attention was trained on 24-hour rolling windows of weather, calendar, and load-lag features for Zone 2 of the Tetouan City dataset (52,416 ten-minute readings from 2017). Chronos-Bolt-Small was applied in a zero-shot setting, forecasting from raw load history alone without fine-tuning. A Light Gradient-Boosting Machine model combined both forecasts with the engineered features to learn a residual correction on top of the stronger anchor forecast, with early stopping based on a held-out validation slice. Two early design choices reduced accuracy before the final version was reached: applying principal component analysis to the exogenous features and training the meta-model to predict load directly rather than a correction. Both issues were diagnosed and addressed. Separately, a validation-checkpointing problem that distorted the deep-learning ablation ranking was corrected by evaluating the models six times per training epoch instead of once. The final pipeline achieved a root mean square error (RMSE) of 391.68 kWh, a mean absolute percentage error (MAPE) of 1.21%, and an R^2 of 0.9949 on a chronological 80/20 Zone 2 split. This represents a 43% lower RMSE and approximately 50% lower MAPE than the figures reported for the published baseline (which was not re-run here; see Limitations), and an 18% improvement over Chronos-Bolt used alone. A Diebold-Mariano test against an internally trained plain LSTM indicated a statistically significant difference (Diebold-Mariano test = 67.07, $p < 10^{-6}$). After the validation fix, the ablation study showed improvements in RMSE and R^2 at every stage, with MAPE following the same overall trend apart from a small increase at the BiLSTM step. When run without the foundation-model signal, the pipeline did not consistently match the published Zone 2 figures (Zone 2 without Chronos: 955.93 kWh; Zone 3: 678.06 kWh; Zone 1: 1,433.77 kWh), indicating that most of the improvement over the published baseline comes from the foundation-model forecast. SHapley Additive exPlanations showed that the Chronos forecast was the dominant input to the correction layer, with short-term load volatility and time-of-day features contributing smaller but consistent signals. The combination of a pretrained, covariate-free foundation model and a domain-specific deep network through a residual correction layer achieved lower errors than those reported for a published hybrid baseline on this dataset, and the foundation model proved to be the strongest individual forecaster tested. The comparison with the external baseline is indicative rather than a fully controlled experiment, and the results are limited to a single dataset and forecast horizon.

Categories: Machine Learning (ML), Data Visualization, Deep Learning

How to cite this article:
Singh K, Tewari A (November 20, 2026) Stacking a BiLSTM-Attention Network With a Pretrained Time-Series Foundation Model and Gradient-Boosted Residual Correction for Short-Term Electricity Load Forecasting: A Comparative Evaluation on the Tetouan City Dataset. Cureus J Comput Sci 3 : es44389-026-00383-6. DOI https://doi.org/10.7759/s44389-026-00383-6

Keywords: time series foundation models, chronos, stacked generalization, lightgbm, explainable ai, short-term load forecasting, time series

Introduction

Grid operators rely on short-term load forecasts to balance supply, plan demand response, and schedule distribution assets. Forecasting methods for this task have shifted from classical statistical models to deep sequence networks, and more recently to hybrid models that pair a sequence model with a gradient-boosted correction stage. On the Tetouan City dataset, which records 10-minute power readings from three distribution zones in northern Morocco throughout 2017, along with temperature, humidity, wind speed, and solar diffuse-flow measurements, a hybrid Long Short-Term Memory (LSTM)-XGBoost model recently reported a Root Mean Square Error (RMSE) of 687.21 kWh, a Mean Absolute Percentage Error (MAPE) of 2.41%, and an R^2 of 0.984 on Zone 2 [1], combining temporal sequence modeling with residual correction driven by weather and calendar covariates. This raises a direct question: Can a pretrained time-series model, with no knowledge of this dataset and no access to any covariates, add information that a purpose-built, weather-aware network cannot supply on its own, once the two are combined through a learned stacking layer rather than one replacing the other?

Time-series foundation models such as Chronos [2] are trained on large, varied collections of real and synthetic time series and can forecast an unfamiliar series from raw history alone, without fine-tuning or covariates. This provides a different kind of information from that provided by a Bidirectional LSTM (BiLSTM) network trained end-to-end on this dataset's weather and calendar features: the foundation model carries broad sequential pattern knowledge learned elsewhere, while the domain-specific network learns relationships specific to this dataset. Whether stacking the two together outperforms either one alone or the published hybrid baseline does not appear to have been tested on this benchmark before this study.

This paper differs from a typical hybrid-model study in three ways. First, it stacks a pretrained foundation model alongside a domain-specific deep network instead of choosing one over the other. Second, it reports two design choices that initially reduced performance - a feature-compression step and an early stacking formulation that underperformed one of its own inputs - rather than fixing them quietly and reporting only the final result, since both failures are informative for anyone attempting something similar. Third, it identifies what actually drives the final result using an ablation study and SHAP (SHapley Additive exPlanations) attribution, rather than reporting a single accuracy number without explanation.

The Tetouan dataset has supported a range of forecasting studies, beginning with a comparison of feedforward networks, random forests, decision trees, and support vector regression [3], and continuing through more recent hybrid and ensemble methods, including the LSTM-XGBoost baseline used here for comparison [1]. The modeling choices in this study build on LSTM networks [4] and additive attention in the Bahdanau style [5], with LightGBM [6] handling the gradient-boosted stage. The stacking layer follows the logic of stacked generalization [7], model behavior is examined using SHAP [8], and forecast comparisons use the Diebold-Mariano (DM) procedure [9].

The broader modeling landscape extends beyond the methods used directly in this pipeline. On the tree-ensemble side, gradient-boosted trees, as implemented in XGBoost [10], and bagged decision trees in the Random Forest algorithm [11] remain common baselines in load-forecasting research, including the hybrid model used for comparison in this study. Sequence modeling has moved along a parallel path away from standard recurrence: the gated recurrent unit [12] offered a lighter alternative to the LSTM cell, and the Transformer architecture [13] replaced recurrence with self-attention entirely, forming the basis of most current time-series foundation models. Chronos sits alongside other pretrained forecasters built on the same attention backbone, including the decoder-only TimesFM [14] and the interpretable, covariate-aware Temporal Fusion Transformer [15]. Benchmarking Chronos against these alternatives as the base forecaster for the correction layer falls outside the scope of this study and is discussed again under Future Work.

How to cite this article:
Singh K, Tewari A (November 20, 2026) Stacking a BiLSTM-Attention Network With a Pretrained Time-Series Foundation Model and Gradient-Boosted Residual Correction for Short-Term Electricity Load Forecasting: A Comparative Evaluation on the Tetouan City Dataset. Cureus J Comput Sci 3 : es44389-026-00383-6. DOI https://doi.org/10.7759/s44389-026-00383-6

Materials And Methods

Dataset

All experiments use the UCI Power Consumption of Tetouan City dataset [3], which contains 52,416 ten-minute observations from January 1 through December 30, 2017, across three distribution zones, along with five weather measurements (temperature, humidity, wind speed, general diffuse flow, and diffuse flow). Zone 2 is the primary zone, used for the headline comparison with the published baseline [1]; Zones 1 and 3 are used afterward to check whether the pipeline generalizes. No rows were dropped, and no values were missing in the source data. Figure *1* summarizes the overall pipeline described in this section, from data preparation through evaluation.

How to cite this article:
Singh K, Tewari A (November 20, 2026) Stacking a BiLSTM-Attention Network With a Pretrained Time-Series Foundation Model and Gradient-Boosted Residual Correction for Short-Term Electricity Load Forecasting: A Comparative Evaluation on the Tetouan City Dataset. Cureus J Comput Sci 3 : es44389-026-00383-6. DOI https://doi.org/10.7759/s44389-026-00383-6

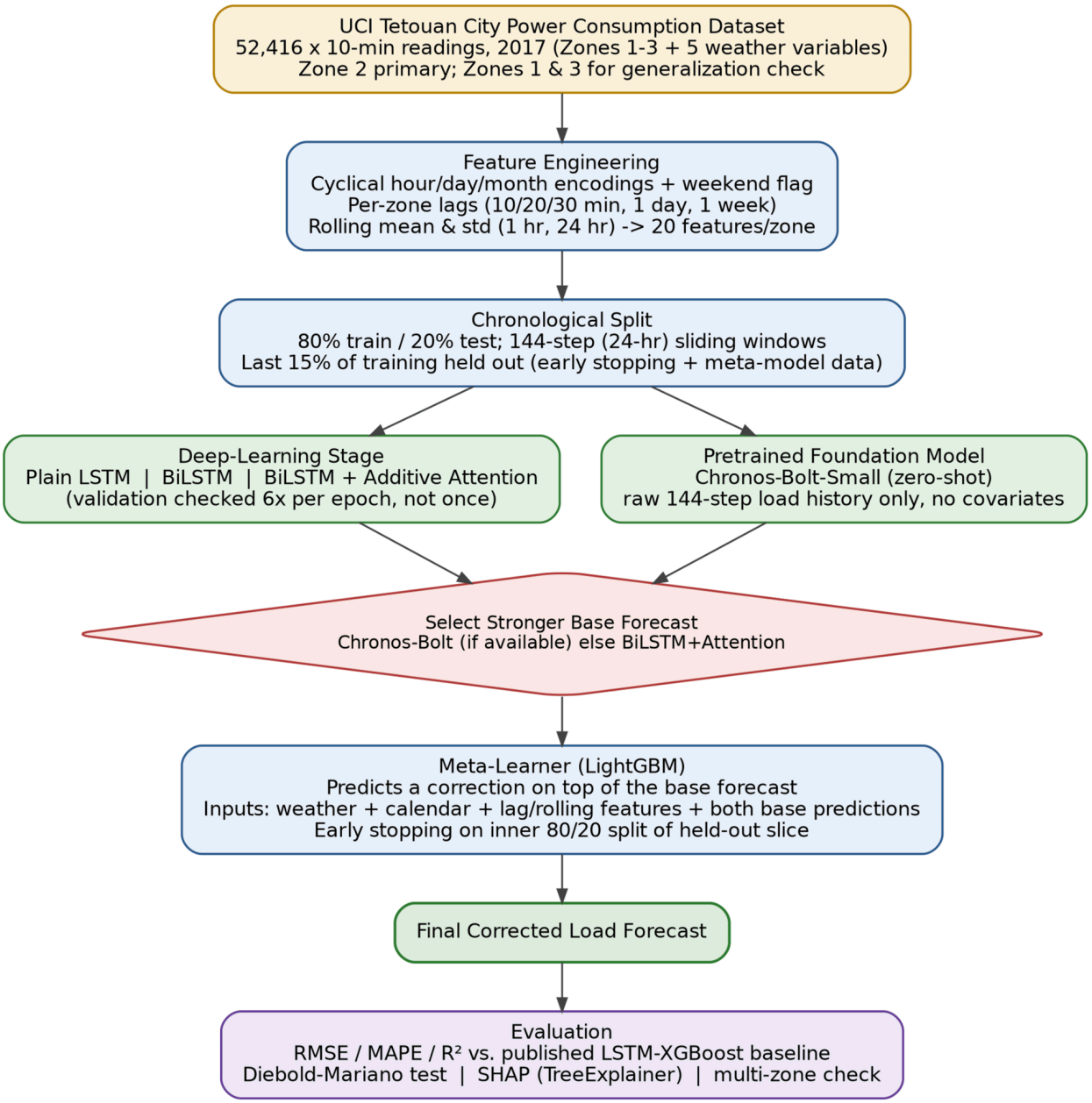


FIGURE 1: Overview of the forecasting pipeline: data, feature engineering, the deep-learning and Chronos-Bolt branches, the stronger-base-forecast selection rule, the LightGBM correction layer, and evaluation

BiLSTM, Bidirectional Long Short-Term Memory; LightGBM, Light Gradient Boosting Machine; LSTM, Long Short-Term Memory; MAPE, Mean Absolute Percentage Error; RMSE, Root Mean Square Error; SHAP, SHapley Additive exPlanations

Published baseline

The comparison target is the hybrid LSTM-XGBoost model reported by Sadik et al. [1], with published Zone 2 figures of 687.21 kWh RMSE, 2.41% MAPE, and 0.984 R^2. That model was not rebuilt here; the comparison uses their published numbers as a fixed reference point rather than as the output of an identical experiment re-run on the same split. The implications of this distinction for interpreting the results are addressed under Limitations and Scope.

How to cite this article:
Singh K, Tewari A (November 20, 2026) Stacking a BiLSTM-Attention Network With a Pretrained Time-Series Foundation Model and Gradient-Boosted Residual Correction for Short-Term Electricity Load Forecasting: A Comparative Evaluation on the Tetouan City Dataset. Cureus J Comput Sci 3 : es44389-026-00383-6. DOI https://doi.org/10.7759/s44389-026-00383-6

Feature engineering

Hour-of-day, day-of-week, and month were each encoded as sine/cosine pairs to preserve their cyclical structure, and a binary weekend flag was added. For each zone, eight descriptors of that zone's own load history were constructed: the values 10, 20, and 30 minutes earlier (lag1 through lag3); the value at the same time on the previous day (lag144) and the same time the previous week (lag1008); the rolling mean and standard deviation of the past hour (6 steps); and the rolling mean of the past 24 hours (144 steps). Each of these features was computed with a one-step shift applied before the rolling window, ensuring that no information at or after the prediction time leaked into a feature intended to describe the past. Dropping the resulting warm-up period (1,008 rows, or one week, determined by the longest lag feature) left 51,408 rows with 20 features per zone for Zone 2 (5 weather, 7 calendar, and 8 own-zone lag and rolling features).

Train, validation, and test split

The data were split chronologically: 80% for training (41,126 rows) and 20% for testing (10,282 rows), with the scalers fitted only on the training portion. Sliding 144-step (24-hour) windows were then constructed from the scaled data, giving 40,982 training sequences and 10,282 test sequences. A further chronological split within the training sequences set aside the last 15% (6,147 rows) as a held-out slice, leaving 34,835 rows for model fitting. This held-out slice served two purposes: early stopping for the deep-learning models and providing out-of-sample predictions used to train the LightGBM meta-model described below.

Deep-learning stage, and a bug in how it was validated

Three sequence models were trained, each mapping a 144-step window of features to a single next-step load value. The first was a plain, one-directional LSTM (64 units, followed by 32), included as an ablation reference point. The second was BiLSTM with the same widths that used the hidden state from the final timestep without an attention mechanism. The third, the model proposed here, adds an additive attention layer [5] over all 144 timesteps of the BiLSTM's output, followed by dropout and a single dense output unit.

The first attempt to train these three models used ordinary once-per-epoch validation and produced an unexpected result: all three models reached their lowest validation loss after a single pass through the training data and got steadily worse in every epoch after that. Early stopping's restore-best-weights behavior ended up selecting whichever checkpoint happened to be closest to the true optimum somewhere within that first pass, providing a coarse measurement given that a full pass covers roughly 270 batches and validation was checked only once at the end of each pass. One consequence was an unexpected ablation ranking: the attention model scored worse than the same network without attention, simply because of where its checkpoint landed rather than any actual difference between the architectures. The fix was to check validation loss six times per pass instead of once, dividing each pass into 45-batch checkpoints (270 total batches per pass), with an explicitly repeating dataset to ensure that training continued correctly across the finer checkpoints. Early-stopping and learning-rate-reduction patience were also scaled accordingly (12 and 6 of the finer checkpoints, roughly two passes and one pass, respectively). The total number of gradient updates allowed remained unchanged; only the frequency of checking changed. This fix was applied identically to all three deep-learning models and to the multi-zone validation loop described later, and it produced the ablation ordering reported in Results.

Pretrained foundation model

Chronos-Bolt-Small [2] was loaded in a zero-shot setting through the Hugging Face chronos-forecasting package, with no fine-tuning and no input beyond each window's own 144-step raw load history. It never sees weather, calendar features, or anything derived from them. It produced a one-step-ahead point forecast for every validation and test window, with the windows processed in batches of 256. Because it requires no training of its own, using it in the multi-zone loop or expanding the meta-learner's training data in future work adds inference time but requires no additional model training.

How to cite this article:
Singh K, Tewari A (November 20, 2026) Stacking a BiLSTM-Attention Network With a Pretrained Time-Series Foundation Model and Gradient-Boosted Residual Correction for Short-Term Electricity Load Forecasting: A Comparative Evaluation on the Tetouan City Dataset. Cureus J Comput Sci 3 : es44389-026-00383-6. DOI https://doi.org/10.7759/s44389-026-00383-6

Meta-learner: correcting the stronger forecast, and a second thing that went wrong first

The first version of the stacking layer trained a LightGBM regressor to predict the load value directly from all available inputs: weather, calendar, lag and rolling features, and both the BiLSTM's and Chronos's predictions. This produced an ensemble with an RMSE of 557.49 kWh on Zone 2, which was worse than Chronos-Bolt's own unaided forecast of 477.89 kWh. An ensemble performing worse than one of its own inputs is a clear signal that reconstructing the full target from scratch, using a training set of only around 6,000 rows, was harder than necessary and that the model was more likely fitting noise than capturing any real pattern in the gap.

The version used in the final pipeline instead trains LightGBM to predict a correction on top of whichever base forecast is stronger - Chronos when available and the BiLSTM otherwise - using the same feature set, along with both base predictions as inputs, and early stopping (50-round patience) on an inner 80/20 split of the held-out slice. Predicting a small correction rather than the full target simplifies the task: when there is no real signal to find, the correction shrinks toward zero instead of pulling the forecast away from an already reasonable anchor. This design substantially reduces the risk of the earlier failure but does not eliminate it, since the correction is still fitted on a finite sample and could, in principle, land on the wrong side of zero on unseen data. On the held-out slice, early stopping settled on 358 boosting rounds for the full meta-model (743 for the no-Chronos ablation version discussed below), well short of overfitting a training set of that size.

Statistical analysis

A DM test [9] compared squared-error loss between the proposed pipeline and an internal plain-LSTM baseline on the same test points to determine whether the two forecasts had equal predictive accuracy. This test used the internal baseline rather than the published external baseline [1] because point-by-point predictions from the published model were not available. This limitation is discussed further under Limitations and Scope.

Explainability

SHAP [8] (TreeExplainer) was applied to the final LightGBM meta-model using a random sample of 1,000 test rows. Because the meta-model predicts a correction rather than the final load value, the resulting attributions describe each feature's role in adjusting the anchor forecast rather than its role in the overall compound prediction, which matters when interpreting the ranking in Results.

Software, implementation, and AI-assisted development

The pipeline was implemented in Python using TensorFlow/Keras for the sequence models, the chronos-forecasting package for the foundation model, LightGBM for the meta-model, and scikit-learn, SHAP, NumPy, pandas, and Matplotlib for supporting analysis. The implementation was run in Google Colab on a T4 GPU.

A large language model (Claude, Anthropic) was used throughout code development, debugging, and analysis, as well as for paraphrasing and drafting assistance. All code was executed by the author rather than the model, and all analytical decisions, interpretation of results, and final text were reviewed and approved by the author.

Results And Discussion

Headline comparison against the published baseline

Table *1* and Figure *2* report Zone-2 test performance at each stage of the pipeline, alongside the published baseline for reference.

Model	RMSE (kWh)	MAPE (%)	R^2
Baseline LSTM (plain)	2,824.17	9.05	0.7370

BiLSTM + Attention (DL only)	2,382.10	8.44	0.8129
Chronos-Bolt zero-shot alone	477.89	1.47	0.9925
BiLSTM + LightGBM stack (no Chronos)	955.93	3.10	0.9699
Proposed: BiLSTM + Chronos + LightGBM	391.68	1.21	0.9949
Published baseline (Sadik et al. [1])	687.21	2.41	0.984

TABLE 1: Zone-2 test performance by pipeline stage, compared against the published baseline

BiLSTM, Bidirectional Long Short-Term Memory; DL, Deep Learning; LightGBM, Light Gradient Boosting Machine; LSTM, Long Short-Term Memory; MAPE, Mean Absolute Percentage Error; RMSE, Root Mean Square Error

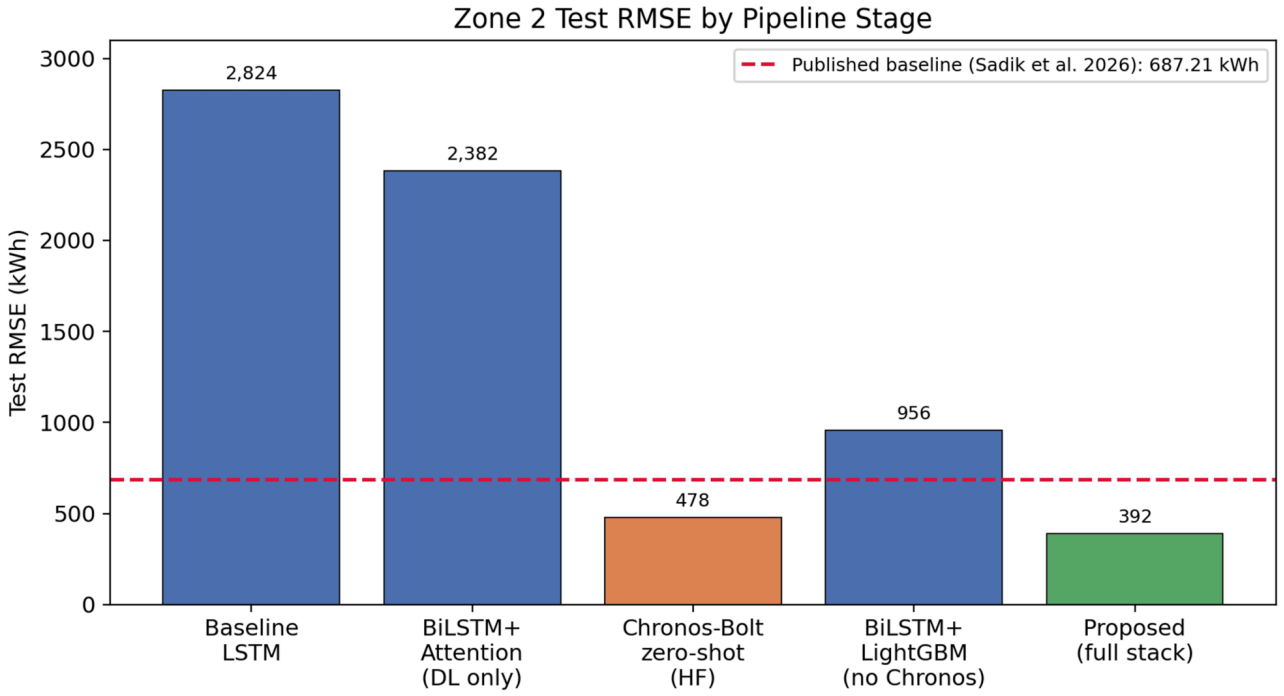


FIGURE 2: Zone-2 test RMSE by pipeline stage, with the published baseline shown as a reference line

BiLSTM, Bidirectional Long Short-Term Memory; LightGBM, Light Gradient Boosting Machine; LSTM, Long Short-Term Memory; RMSE, Root Mean Square Error

Compared with the published figures, the full pipeline's RMSE was 43.0% lower and its MAPE roughly 50% lower, with a slightly higher R^2 as well (0.9949 versus 0.984). It also outperformed Chronos-Bolt used alone by 18.0% on RMSE, providing the clearest indication that the residual-on-anchor redesign resolved the earlier problem of the ensemble

underperforming one of its own components. Chronos-Bolt alone (477.89 kWh) already falls below the published figure, while the stack without Chronos (955.93 kWh) does not. Most of the margin over the published baseline is therefore attributable to the foundation-model forecast rather than to the BiLSTM-attention design.

Ablation study

Table *2* and Figure *3* break down the contribution of each pipeline component, using the corrected, finer-grained validation checkpointing described above.

Stage	RMSE (kWh)	MAPE (%)	R^2	Δ RMSE vs. prev.
1. LSTM (plain)	2,824.17	9.05	0.7370	—
2. BiLSTM (no attention)	2,712.69	9.13	0.7573	−3.9%
3. BiLSTM + Attention	2,382.10	8.44	0.8129	−12.2%
4. + LightGBM stack (no Chronos)	955.93	3.10	0.9699	−59.9%
5. + Chronos (full proposed)	391.68	1.21	0.9949	−59.0%

TABLE 2: Ablation study: incremental contribution of each pipeline component on Zone 2, after the validation-checkpointing fix

BiLSTM, Bidirectional Long Short-Term Memory; LightGBM, Light Gradient Boosting Machine; LSTM, Long Short-Term Memory; MAPE, Mean Absolute Percentage Error; RMSE, Root Mean Square Error

How to cite this article:
Singh K, Tewari A (November 20, 2026) Stacking a BiLSTM-Attention Network With a Pretrained Time-Series Foundation Model and Gradient-Boosted Residual Correction for Short-Term Electricity Load Forecasting: A Comparative Evaluation on the Tetouan City Dataset. Cureus J Comput Sci 3 : es44389-026-00383-6. DOI https://doi.org/10.7759/s44389-026-00383-6

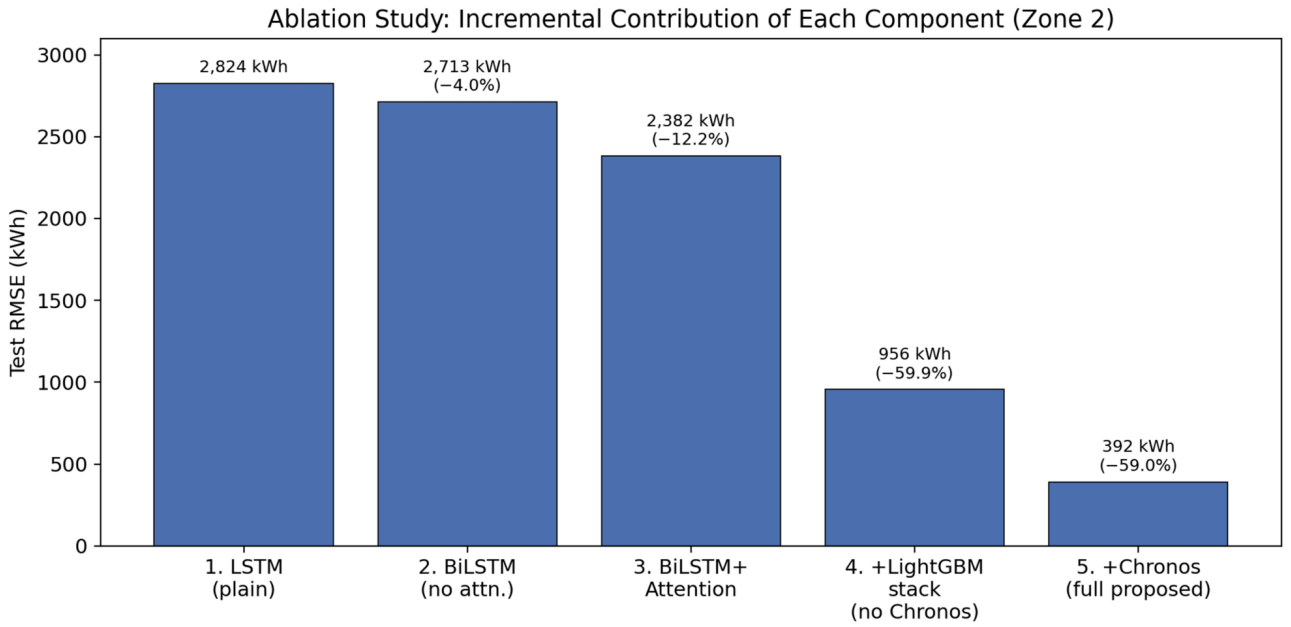


FIGURE 3: Ablation study - each stage's incremental RMSE reduction on Zone 2, after the validation-checkpointing fix

BiLSTM, Bidirectional Long Short-Term Memory; LightGBM, Light Gradient Boosting Machine; LSTM, Long Short-Term Memory; RMSE, Root Mean Square Error

RMSE and R^2 improve at every stage; MAPE follows the same trend except for a small increase at the BiLSTM step (row 2), before falling sharply from row 3 onward. Attention itself clearly helps once added (row 3 versus row 2), across all three metrics. This is the opposite of what an earlier run showed under the coarser once-per-pass validation check, where attention appeared to hurt performance. The largest single improvement comes from adding the LightGBM stacking stage; Chronos then adds a further, smaller but still meaningful reduction on top of an already strong stacked model.

Statistical significance

Comparing the proposed pipeline against the internal plain-LSTM baseline using a DM test on the 10,282 paired forecast errors from the Zone-2 test set gave DM = 67.0724, $p < 10^{-6}$ (n = 10,282), rejecting the hypothesis that the two forecasts are equally accurate. As noted in Materials and Methods, this test uses the internal LSTM baseline rather than the published external one, since paired point predictions from the latter were not available.

Multi-zone generalization

Table *3* and Figure *4* apply the same pipeline to Zones 1 and 3, leaving Chronos out of the meta-learner's inputs to keep the validation loop's runtime manageable (see Limitations and Scope), alongside the Zone-2 result already reported.

Zone	RMSE (kWh)	MAPE (%)	R^2
Zone 1	1,433.77	3.69	0.9457
Zone 2 (primary)	391.68	1.21	0.9949
Zone 3	678.06	3.88	0.9580

TABLE 3: Multi-zone generalization of the proposed pipeline. Comparisons to the published Zone-2 baseline for Zones 1 and 3 are informal, contextual comparisons only - the baseline was reported for Zone 2 alone, and the Zones 1/3 runs use a lighter meta-learner configuration without Chronos

MAPE, Mean Absolute Percentage Error; RMSE, Root Mean Square Error

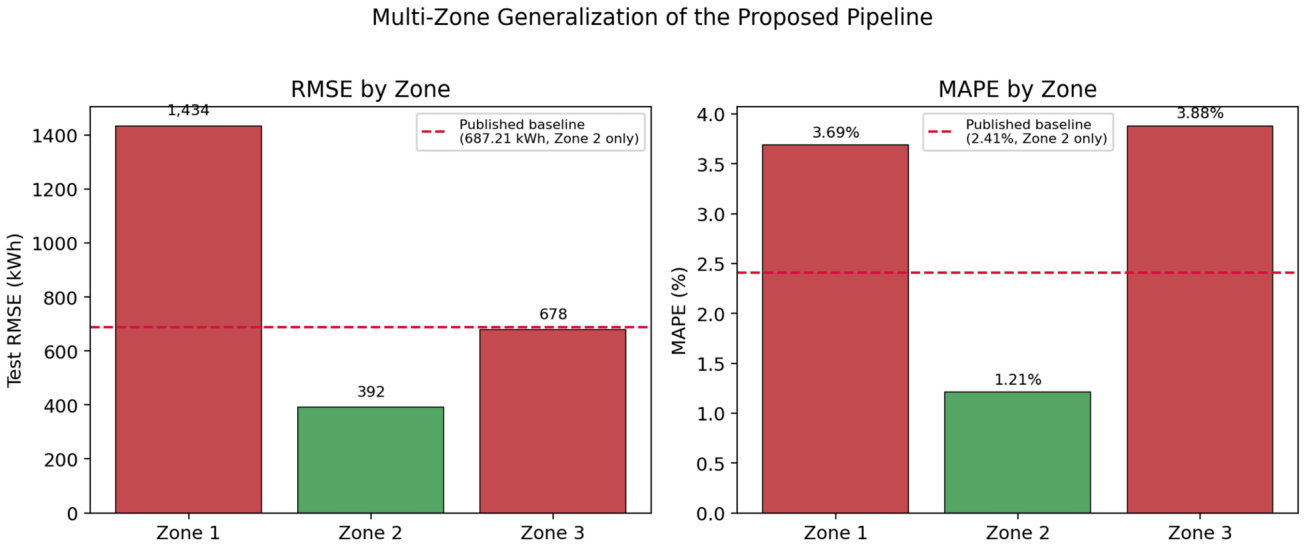


FIGURE 4: Multi-zone results for the proposed pipeline (Chronos omitted from Zones 1 and 3; see Limitations and Scope). Dashed lines show the published Zone-2 figures for context only

MAPE, Mean Absolute Percentage Error; RMSE, Root Mean Square Error

Zone 3 reaches an RMSE (678.06 kWh) close to the published Zone-2 figure (687.21 kWh), while Zone 1 and both zones' MAPE are higher. Because the baseline was reported for Zone 2 only, and these runs omit Chronos and use a lighter meta-learner, this experiment shows that the pipeline runs on other zones. It does not show that it surpasses the baseline there, and it cannot separate zone difficulty from the reduced configuration.

The Chronos forecast accounts for roughly three times the attribution of the next-ranked feature, consistent with how well it performs on its own (Table *1*) and with the fact that a 10-minute-ahead horizon is dominated by very recent persistence, exactly the regime a strong general-purpose sequence model is built to handle. The next tier of features by importance - the past hour's volatility, the two shortest lags after lag1, and hour-of-day - are not simply echoing Chronos: their attributions suggest the correction layer is drawing on weather-independent local volatility and time-of-day structure that a covariate-blind, zero-shot model cannot see. The BiLSTM's own prediction ranks tenth of fifteen, a modest but non-negligible contribution, consistent with the earlier finding that the deep-learning stage alone (row 3 of the ablation table) already captures real signal before either Chronos or the meta-learner is introduced.

Discussion

A zero-shot model with no access to weather or calendar information outperformed a purpose-built, covariate-aware deep learning model working alone (477.89 kWh for Chronos versus 2,382.10 kWh for BiLSTM+Attention). One plausible explanation is that, at a 10-minute horizon, most of what is predictable about load is short-term persistence and local seasonal pattern - the kind of structure a broadly pretrained sequence model can capture well from raw history alone. Weather and calendar information may matter more at longer horizons, where persistence alone would not be sufficient, though this study did not test other horizons, and this remains an interpretation rather than a direct finding. Under this

How to cite this article:
Singh K, Tewari A (November 20, 2026) Stacking a BiLSTM-Attention Network With a Pretrained Time-Series Foundation Model and Gradient-Boosted Residual Correction for Short-Term Electricity Load Forecasting: A Comparative Evaluation on the Tetouan City Dataset. Cureus J Comput Sci 3 : es44389-026-00383-6. DOI https://doi.org/10.7759/s44389-026-00383-6

reading, the deep-learning and gradient-boosting stages function less as the primary forecaster and more as a correction layer that recovers whatever covariate-dependent signal Chronos cannot see by design. The SHAP ranking in Figure *5* is consistent with this interpretation.

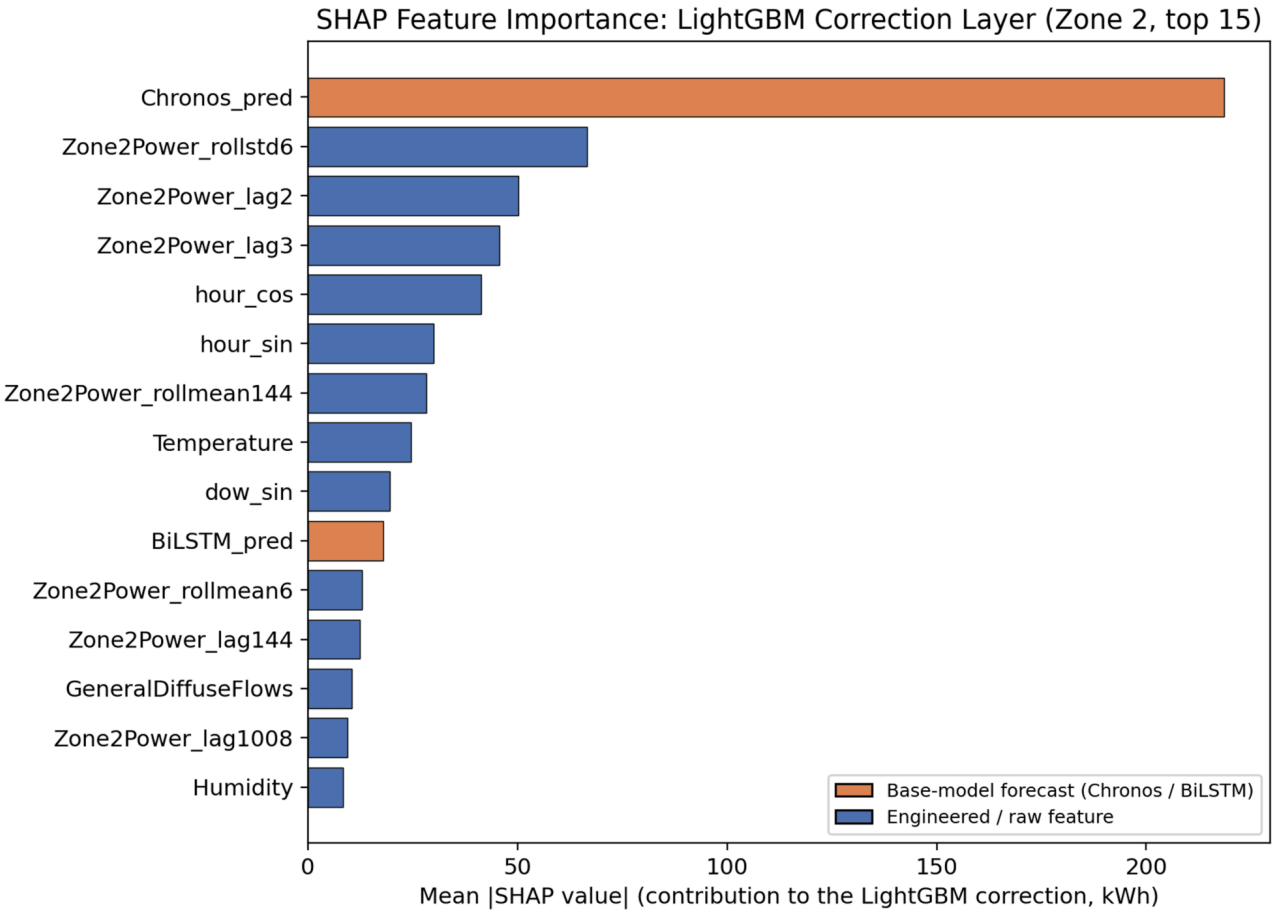


FIGURE 5: Top 15 features by mean absolute SHAP value for the LightGBM correction layer (Zone 2, 1,000 random test rows)

BiLSTM, Bidirectional Long Short-Term Memory; LightGBM, Light Gradient Boosting Machine; SHAP, SHapley Additive exPlanations

The two failed designs reported here - principal component analysis reducing the usefulness of the feature set, and the first stacking attempt underperforming its own best input - point to the same underlying issue from different directions: with a meta-training set in the low thousands of rows, it is easy to make the model's task harder than necessary, whether by compressing away a useful feature or asking it to reconstruct the full target from scratch. Both fixes worked by simplifying what the meta-learner had to learn, not by adding complexity.

Limitations and scope

The comparison against the published baseline [1] is indicative rather than a controlled experiment: their exact train/test split, preprocessing, and hyperparameters were not reproduced here, so the reported improvement should be read as this pipeline outperforming the previously published number on this split of this dataset, not as a paired comparison under identical conditions. A stricter test would rebuild the baseline exactly and evaluate both models on the same folds.

The evaluation relies on one dataset, one 10-minute horizon, and a single chronological split, rather than the repeated cross-validation common in tabular benchmarks. Time-series data does not lend itself to the same repeated-fold approach, but this means the headline numbers come from a single split rather than a distribution across many, and the only significance test available (DM) compares against the internal baseline rather than the external one. The held-out

slice is used both for early stopping of the deep-learning models and for training the meta-learner, which may make the stacking stage slightly optimistic. The published baseline's forecast horizon and target definition were also not verified to match this study's.

The multi-zone check omitted Chronos and used a lighter LightGBM configuration to keep runtime reasonable on free-tier compute, so the higher error observed in Zones 1 and 3 cannot be attributed cleanly to genuine zone difficulty versus that reduced configuration without re-running them under the full Zone-2 setup.

Only the smallest Chronos-Bolt model was tested; larger variants, or the newer Chronos-2 family with built-in covariate support [2], might shift the balance between the foundation model and the deep-learning stage and were not evaluated here. The residual-on-anchor design, even after the fix, reduces rather than eliminates the risk of the meta-learner underperforming its anchor on unseen data, since it is still fit on a single finite validation slice. A purged k-fold out-of-fold scheme would provide more training data at roughly k times the deep-learning training cost and was not attempted here for that reason.

This work was also built across several development rounds rather than a single continuous run: code was written, executed elsewhere, and revised based on the results returned each time, and both negative results described above surfaced through that process. It represents a different kind of verification path than one person independently reproducing every number in a single sitting.

Future work

Several directions follow from the limitations above: rebuilding the published baseline's exact protocol for a properly controlled comparison; testing longer forecast horizons, where covariate information may contribute more than it does at 10 minutes; evaluating larger Chronos models and the covariate-aware Chronos-2 family; replacing the single held-out validation slice with purged k-fold out-of-fold stacking to give the meta-learner more data to work with; running the full Zone-2 configuration, including Chronos, on Zones 1 and 3 to separate genuine zone difficulty from configuration differences; and testing whether the same architecture holds up on other load-forecasting datasets with different noise and seasonality characteristics.

Conclusions

On the Tetouan City Zone-2 forecasting benchmark, combining a BiLSTM-attention network with a pretrained time-series foundation model (Chronos-Bolt, used zero-shot) through a gradient-boosted correction layer achieved 43% lower RMSE and about 50% lower MAPE than the figures reported for a recently published hybrid LSTM-XGBoost baseline. The improvement over an internally trained plain-LSTM baseline was confirmed by a DM test (paired point predictions from the published baseline were not available for this test); applied to Zones 1 and 3 without Chronos, the pipeline did not consistently beat the published Zone-2 figures, so the gain over the published baseline was not shown to generalize beyond Zone 2. The foundation model, despite having no fine-tuning and no covariates, proved to be the strongest individual forecaster in the pipeline, a result that likely reflects how persistence-dominated a 10-minute horizon is rather than a general claim that foundation models outperform domain-specific deep learning. Two design choices that initially reduced performance - a principal component analysis step that diluted a useful feature, and a meta-learner asked to reconstruct the target from scratch instead of correcting it - are reported alongside the final numbers, on the view that the failure and how it was diagnosed are as useful to a reader building something similar as the result itself.

Additional Information

Author Contributions

All authors have reviewed the final version to be published and agreed to be accountable for all aspects of the work.

Concept and design: Khushvir Singh

How to cite this article:
Singh K, Tewari A (November 20, 2026) Stacking a BiLSTM-Attention Network With a Pretrained Time-Series Foundation Model and Gradient-Boosted Residual Correction for Short-Term Electricity Load Forecasting: A Comparative Evaluation on the Tetouan City Dataset. Cureus J Comput Sci 3 : es44389-026-00383-6. DOI https://doi.org/10.7759/s44389-026-00383-6

Acquisition, analysis, or interpretation of data: Khushvir Singh, Anurag Tewari

Drafting of the manuscript: Khushvir Singh

Critical review of the manuscript for important intellectual content: Khushvir Singh, Anurag Tewari

Supervision: Anurag Tewari

Disclosures

Human subjects: All authors have confirmed that this study did not involve human participants or tissue.

Animal subjects: All authors have confirmed that this study did not involve animal subjects or tissue.

Conflicts of interest: In compliance with the ICMJE uniform disclosure form, all authors declare the following:

- **Payment/services info:** All authors have declared that no financial support was received from any organization for the submitted work.
- **Financial relationships:** All authors have declared that they have no financial relationships at present or within the previous three years with any organizations that might have an interest in the submitted work.
- **Other relationships:** All authors have declared that there are no other relationships or activities that could appear to have influenced the submitted work.

Data Availability Statements

The data (and/or code) supporting this study are openly available in UCI Machine Learning Repository, GitHub, at doi.org/10.24432/C5B034 (Dataset 89), github.com/khushvir11/Short-Term-Load-Forecasting.

References

1. Sadik Y, Nejmi A, Oumiguil L, Baite M: Forecasting energy consumption using a hybrid LSTM-XGBoost model. Engineering Proceedings. 2026, 144:4. 10.3390/engproc2026144004
2. Ansari AF, Stella L, Turkmen C, et al.: Chronos: learning the language of time series [PREPRINT]. arXiv. 2024, 10.48550/arXiv.2403.07815
3. Salam A, El Hibaoui A: Comparison of machine learning algorithms for the power consumption prediction: case study of Tetouan city. 2018 6th International Renewable and Sustainable Energy Conference (IRSEC), Rabat, Morocco. 2018, 1-5. 10.1109/IRSEC.2018.8703007
4. Hochreiter S, Schmidhuber J: Long short-term memory. Neural Computation. 1997, 9:1735-1780. 10.1162/neco.1997.9.8.1735
5. Bahdanau D, Cho K, Bengio Y: Neural machine translation by jointly learning to align and translate [PREPRINT]. arXiv. 2016, 10.48550/arXiv.1409.0473
6. Ke G, Meng Q, Finley T, et al.: LightGBM: a highly efficient gradient boosting decision tree. NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems. 2017, 3149-3157.
7. Wolpert DH: Stacked generalization. Neural Networks. 1992, 5:241-259. 10.1016/s0893-6080(05)80023-1
8. Lundberg SM, Lee SI: A unified approach to interpreting model predictions. NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems. 2017, 4768-4777.
9. Diebold FX, Mariano RS: Comparing predictive accuracy. Journal of Business & Economic Statistics. 1995, 13:253-263. 10.1080/07350015.1995.10524599
10. Chen T, Guestrin C: XGBoost: a scalable tree boosting system. KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016, 785-794. 10.1145/2939672.2939785
11. Breiman L: Random forests. Machine Learning. 2001, 45:5-32. 10.1023/a:1010933404324

How to cite this article:
Singh K, Tewari A (November 20, 2026) Stacking a BiLSTM-Attention Network With a Pretrained Time-Series Foundation Model and Gradient-Boosted Residual Correction for Short-Term Electricity Load Forecasting: A Comparative Evaluation on the Tetouan City Dataset. Cureus J Comput Sci 3 : es44389-026-00383-6. DOI https://doi.org/10.7759/s44389-026-00383-6

12. Cho K, van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y: Learning phrase representations using RNN encoder-decoder for statistical machine translation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2014, 1724-1734. 10.3115/v1/D14-1179

13. Vaswani A, Shazeer N, Parmar N, et al.: Attention is all you need. NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems. 2017, 6000-6010.

14. Das A, Kong W, Sen R, Zhou Y: A decoder-only foundation model for time-series forecasting. Proceedings of the 41st International Conference on Machine Learning. 2024, 235:10148-10167.

15. Lim B, Arık SÖ, Loeff N, Pfister T: Temporal fusion transformers for interpretable multi-horizon time series forecasting. International Journal of Forecasting. 2021, 37:1748-1764. 10.1016/j.ijforecast.2021.03.012

How to cite this article:
Singh K, Tewari A (November 20, 2026) Stacking a BiLSTM-Attention Network With a Pretrained Time-Series Foundation Model and Gradient-Boosted Residual Correction for Short-Term Electricity Load Forecasting: A Comparative Evaluation on the Tetouan City Dataset. Cureus J Comput Sci 3 : es44389-026-00383-6. DOI https://doi.org/10.7759/s44389-026-00383-6