

Article

Nonlinear Prediction with Data Augmentation and Regularization: A Ensembled LSTM–XGBoost Model

Devine Grace D. Funcion^{1,*}, Marvee Cheska B. Natividad²

¹ College of Computing Studies, Information Communication Technology, Isabela State University – Cuayan Campus, 3305, Philippines; e-mail: devinegrace.funcion_cyn@isu.edu.ph (D. G. D. Funcion), marveecheska.b.natividad@isu.edu.ph (M. C. B. Natividad).

* Correspondence Author

The authors received no financial support for the research, authorship, and/or publication of this article.

Abstract: Nonlinear prediction faces its main challenge in the form of overfitting, which leads to inaccurate predictions. The problem becomes evident when researchers attempt to apply advanced deep learning systems to small agricultural data collections. Researchers use knowledge discovery in databases (KDD) to evaluate regularization methods with ensemble techniques, but have not sufficiently explored how structured data augmentation interacts with MaxNorm regularization. The research explores how a sliding-window transformation, together with bootstrap augmentation methods, works when ElasticNet, Bayesian, and MaxNorm regularization techniques are combined into an LSTM-XGBoost prediction system to predict Tikog grass demand. The research showed that data augmentation techniques helped reduce model overfitting, thereby improving performance on prediction tasks. Among the regularization strategies applied to LSTM, MaxNorm achieved the largest reduction in error, with testing MSE decreasing from 0.060472 to 0.002090 after augmentation. A comparative evaluation further shows that LSTM-XGBoost achieved the highest overall performance ($R^2 = 0.997806$), while deep learning models showed greater sensitivity to augmentation and regularization strategies. These findings highlight that structured time-series augmentation combined with norm-based regularization enhances generalization capability, particularly for high-capacity sequence models trained on limited agricultural data.

Keywords: Bootstrapping; Data augmentation; Forecasting; Sliding window; Time-series.

Copyright: © 2026 by the authors. This is an open-access article under the CC-BY-SA license.



1. Introduction

Machine learning has become increasingly prevalent across many industries, such as business, education, and agriculture, due to the ever-growing volume of data generated every day. Predictive analytics forms the foundation of modern decision-support systems; therefore, developing accurate and reliable prediction models is essential for enabling informed and meaningful decisions across various domains. Although predictive analytics has been widely applied in areas such as healthcare and finance, where it supports early disease detection, treatment optimization, and financial forecasting [1], [2], its effectiveness remains highly dependent on data availability and model generalization capability, particularly in domain-specific applications.

A significant challenge in machine learning is overfitting, especially when working with limited datasets. This

phenomenon occurs when a model learns noise rather than underlying patterns, resulting in poor generalization to unseen data [3], [4]. To address this issue, ensemble learning has been widely adopted, as it combines multiple models to improve predictive performance and robustness [5]. Techniques such as boosting, bagging, and stacking have demonstrated superior performance compared to single-model approaches in many applications [6], [7]. However, ensemble methods alone are often insufficient when data scarcity limits the diversity and representativeness of training samples.

This challenge becomes more pronounced in agricultural time-series prediction, where datasets are typically small, noisy, and highly nonlinear. Accurately capturing temporal dependencies while maintaining generalization remains difficult under such constraints. Existing studies have primarily addressed this problem through isolated

approaches, such as data augmentation, regularization, or ensemble learning, without systematically investigating their combined effects. As a result, the interaction between structured data augmentation techniques and regularization strategies in Ensembled learning frameworks remains insufficiently explored.

To address this research gap, this study proposes a Ensembled LSTM-XGBoost framework that integrates sliding-window transformation, bootstrap-based data augmentation, and multiple regularization strategies within a unified predictive system. The proposed approach leverages the temporal modeling capability of Long Short-Term Memory (LSTM) networks and the nonlinear optimization strength of XGBoost to enhance predictive accuracy and generalization. By systematically combining augmentation and regularization mechanisms, this study aims to provide a more robust solution for nonlinear prediction in small agricultural datasets, specifically for forecasting Tikog grass demand.

The main contributions of this study are threefold. First, it introduces a structured integration of sliding-window and bootstrap augmentation tailored for limited time-series data. Second, it evaluates the comparative effectiveness of multiple regularization strategies, including ElasticNet, Bayesian-inspired methods, and MaxNorm, within a deep learning framework. Third, it demonstrates the effectiveness of a Ensembled LSTM-XGBoost model in improving generalization performance under data-constrained conditions.

2. Literature Review

Long Short-Term Memory (LSTM) networks have been developed specifically to overcome the limitations of standard Recurrent Neural Networks (RNNs) when handling sequential data, thereby maintaining both short- and long-term dependencies [8]. LSTMs accomplish this by using a special memory cell and three gating mechanisms (input gate, forget gate, and output gate) that will allow them to filter the data coming through in time and keep only the most important information for an extended length of time while still being able to take the input back out when needed [9], [10]. Because of their ability to store information for longer periods, LSTMs have proven highly effective for tasks such as predicting nonlinear data, understanding human language, and recognizing spoken words. In addition, LSTMs provide a robust method for addressing limitations in temporal dependencies within sequences by using conditional threshold structures that define how previous inputs are remembered and processed. This means they will be able to identify and retain only the meaningful features within the state cell over longer periods, while simultaneously removing any meaningless information processed [11].

XGBoost (Extreme Gradient Boosting) is a framework for building an ensemble of decision trees via gradient

boosting, where trees are built sequentially (one after another). Each tree is built off the "residuals" of the previous tree(s) (i.e., minimizing the residuals of the prior trees to increase the accuracy of the prediction). XGBoost utilizes the "gradient descent" method to find the best-fit gradient at each step to minimize the residuals from the prior tree; this leads to better accuracy in predictions. In addition, XGBoost uses "regularization" techniques to limit tree complexity and reduce overfitting, making it very useful for modeling complex, nonlinear relationships in structured datasets [12]. In 2011, Chen Tianqi and Carlos introduced the XGBoost model, an ensemble boosting technique that uses gradient descent to minimize the loss function [13]. XGBoost has emerged as a promising model for nonlinear system modeling. These models provide more robust and accurate predictions than single learning algorithms, using bagging and boosting ensemble learning methods [14]. The XGBoost algorithm can model complex problems with high precision and high speed [15]. Hence, this research paper leverages the LSTM-XGBoost algorithm to predict Tikog grass. What are the basic materials used to make native products such as bags, mats, and slippers from Basey, Samar, Philippines? With LSTM's strength in capturing long-term dependencies and XGBoost's performance in nonlinear settings, it is the best method for predicting Tikog grass. LSTM-XGBoost achieves high long-term soil moisture prediction accuracy, with R^2 values of 98.67%, making it a practical tool for real-time irrigation scheduling and enhancing water use efficiency in precision agriculture [16]. Additionally, the cold chain loading environment for agricultural items is predicted using LSTM-XGBoost values of 2.5343 (MAE), 5.1906 (RMAE), and 0.8971 (R^2). These statistics show the model's ability to accurately predict temperature trends for a future time period in the cold chain loading process, which encourages energy efficiency and environmental sustainability while also reducing the risk of agricultural product spoilage [17]. Additionally, LSTM-XGBoost shows increased accuracy and resilience against overfitting [18].

Training complex models with non-linear data is challenging because data are often limited and unbalanced. In many classification problems, class imbalance is common; rare but important events are vastly outnumbered by typical patterns [19]. Moreover, large training samples are difficult to obtain due to data and resource constraints [20]. Data scarcity and lack of diversity provide limited examples from which to learn, resulting in the model memorizing its training set rather than learning generalized time-series dynamics. Therefore, models may have overfitted their training set and will not perform well with new data. The acquisition of large amounts of data is often an ongoing challenge for many time-series recognition applications; however, an increased dataset will enhance the model's generalization capability and overall

performance [21], and susceptible to bias when dealing with unexpected or new contexts.

In addition to the difficulties associated with sparsity and imbalance in a database when predicting using a time series, these limitations are even more stringent when the performance of previous or traditional methods is considered. Regularization provides the fundamental motivation for a model to learn and be generalizable by constraining its complexity. The hard problem of overfitting (common in high-dimensional data sets) is best handled with the regularization technique [22]. Regularization and all forms of structural risk minimization improve a model's generalization performance by making an estimator more biased and therefore less variable. But there are no guarantees of equal weighting of bias in the estimator across all classes in the database [23]. Regularization techniques to reduce overfitting introduce uncertainty, as excessive regularization can also lead to underfitting. Therefore, finding the right balance between these extremes depends on knowing which classifier is best for the problem, and since this knowledge does not exist, no single technique is guaranteed to work for all cases. Regularization techniques must therefore be applied with careful consideration and evaluated through appropriate experimentation to assess the capacity of the resulting model [24].

Hence, with the limitations of the regularization, this study will utilize data augmentation to prevent overfitting during model training. By using data augmentation (DA) techniques, which can create synthetic versions of the original training samples [20], [25], mimicking their style and characteristics to help replace or supplement the original samples. One way to address the shortage of labeled training data is to use data augmentation with deep learning, which creates additional synthetic samples to increase the size of the sample dataset [26]. Two of the most prominent data augmentation techniques are bootstrapping and sliding windows.

One way to generate additional training sets is to use bootstrapping to create samples that are representative of the original data. This bootstrapping process allows the creation of many different sample sets, which can be used to increase the training set size and improve model performance [25]. An example of this scenario is when we have a monthly dataset of electricity bills, but only a very small subset is available for training a model to predict future bills. We can use bootstrapping to address this. When using this technique, sample the original dataset several times, and in each sample, take certain observations from the original dataset while still leaving some observations out of each sample. In essence, all samples are the same size as the original dataset, but they have different combinations of the original data points. Training on many bootstrapped datasets yields a more stable model, reduces overfitting, and increases confidence in estimates of its performance, even when the original dataset is limited in

size. The distribution of sample statistics from simulated samples is used as the sampling distribution, allowing for precise confidence intervals and hypothesis testing [27]. While bootstrap methods do not require knowledge of population parameters such as the mean or standard deviation, they may suffer from limited diversity and low precision in synthetic data generation. Post-processing requires two main steps: outlier elimination and data filtering to produce better synthetic data. The process of combining independent bootstrap data sets yields more accurate, more diverse results [27].

Organizations across multiple sectors find value in sliding window techniques because these methods enable them to develop predictive models that work across various application areas, including neural network-based forecasting of tool surface temperatures. Multiple network types achieve good performance when using sliding windows to predict tool surface temperature, demonstrating the method's effectiveness [28]. In industrial maintenance, the selection of an optimal Sliding Window can have a substantial effect on machine learning models for failure prediction, which reinforces the importance of selecting an appropriate prediction window [29]. Additionally, the Sliding Window method has been employed to predict reservoir water levels and tested with different window sizes to improve the accuracy of artificial neural networks [30].

In the Sliding Window method, data is transformed into input-output pairs by sliding a fixed-length window over the dataset, with the data points within the window used as inputs and the data point immediately following the window as the output. The sliding window size is a hyperparameter that should be adjusted based on the nature of the data. The sliding window technique is an advantageous method of increasing the amount of training data available for small nonlinear datasets. The process of dividing original nonlinear data into overlapping segments produces expanded datasets that retain all essential temporal patterns, including trends and seasonal variations. The extra data enables models to identify general patterns, leading to better predictions with reduced risk of overfitting. The sliding window technique serves as an effective method for users to identify better nonlinear prediction models based on their selection confidence [31].

Regularization approaches have been widely used to mitigate overfitting. The impact of simultaneously using both augmented data and regularization remains poorly understood, particularly for smaller agricultural datasets that may lack sufficient data. Existing studies predominantly investigate regularization or ensemble modeling in isolation, with limited empirical evaluation of bootstrapping and sliding-window augmentation techniques integrated with regularization strategies. Furthermore, there is a lack of domain-specific evidence assessing the robustness of such approaches in agricultural prediction applica-

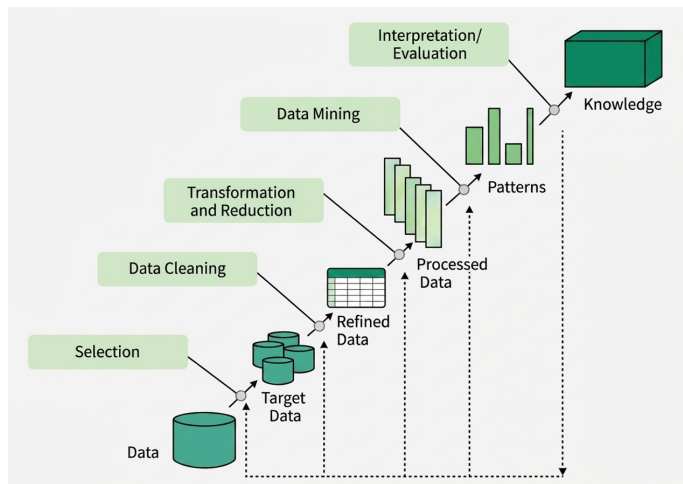


Figure 1. KDD process.

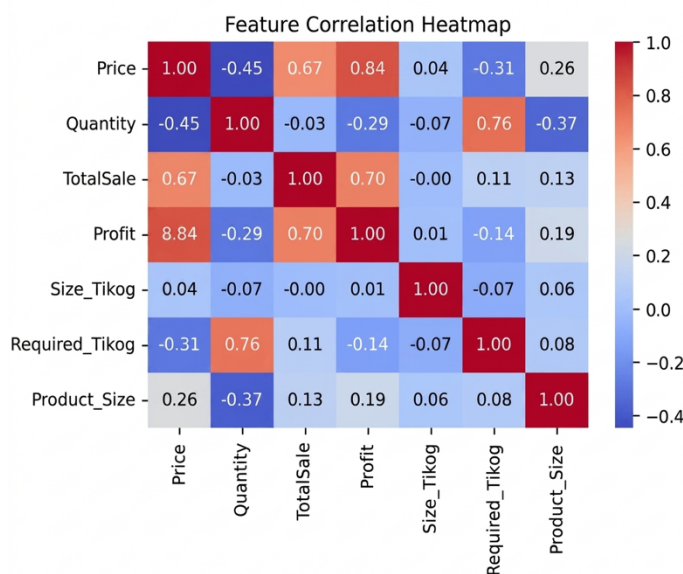


Figure 2. Feature Correlation Heatmap.

tions, including underrepresented contexts such as Tikog grass demand. Consequently, the optimal augmentation + regularization configuration for improving generalization and reducing overfitting in ensembled LSTM–XGBoost models remains unclear. Therefore, this study aims to systematically assess and compare the effectiveness of data augmentation methods and regularization techniques in reducing overfitting and improving the generalization capabilities of an LSTM-XGBoost predictive model for Tikog grass forecasting.

3. Method

The proposed method establishes a structured machine learning system that operates through specific stages to generate better predictive results and improve model generalization. As illustrated in Fig. 1, the system integrates data preprocessing, data augmentation, regularization, and an ensemble-based prediction framework.

To enhance generalization and limit overfitting, especially for relatively small and nonlinear datasets, data augmentation is applied to increase dataset size and diversity.

This is achieved through sliding-window transformation and bootstrap resampling, which generate synthetic samples based on the statistical characteristics of the original data. In addition, regularization techniques are used to control model complexity. One commonly used method is MaxNorm regularization, which constrains the maximum norm of model weights.

3.1. Research Stages

The method follows the Knowledge Discovery in Databases (KDD) framework, consisting of five main stages: data selection, data cleaning, transformation, data mining, and interpretation/evaluation, as shown in Fig. 1.

3.2. Data Selection

The dataset consists of monthly sales reports from 2017 to 2024, comprising 862 records with attributes such as product name, price, quantity, total sales, product size, and Required_Tikog. Data augmentation was applied to the training data, producing 588 samples using the sliding window method and 2940 samples using bootstrapping.

A correlation heatmap (Fig. 2) was used to guide feature selection. The results show a strong positive correlation between Profit and Price (0.84), indicating that pricing significantly influences profitability. Required_Tikog also shows a strong correlation with Quantity (0.76), suggesting that production volume strongly affects raw material requirements. These findings help identify the most relevant features for model input.

3.3. Data Cleaning

Data cleaning involves handling missing values and ensuring data consistency. Missing values introduced during lag feature generation are addressed using backward filling to maintain temporal continuity. This step ensures that the dataset remains complete and suitable for time-series modeling.

3.4. Transformation

Data preprocessing generates calendar-based predictors, including year, month, day, weekday, and a weekend indicator, to capture temporal patterns. Lag features of Profit and Quantity (up to three previous periods) are also created to model short-term dependencies.

The dataset is then transformed into a supervised learning structure using a sliding window approach. For each time index t , the input tensor is defined as:

$$\mathbf{X}_t = [\mathbf{x}_{t-5}, \mathbf{x}_{t-4}, \mathbf{x}_{t-3}, \mathbf{x}_{t-2}, \mathbf{x}_{t-1}], \quad (1)$$

where $\mathbf{x}_t \in \mathbb{R}^{15}$ represents the feature vector, and the target is $y_t = \text{Required_Tikog}_t$. This results in an input tensor $\mathbf{X} \in \mathbb{R}^{2995 \times 5 \times 15}$. The dataset is split chronologically into 80% training and 20% testing sets without shuffling to preserve temporal dependencies.

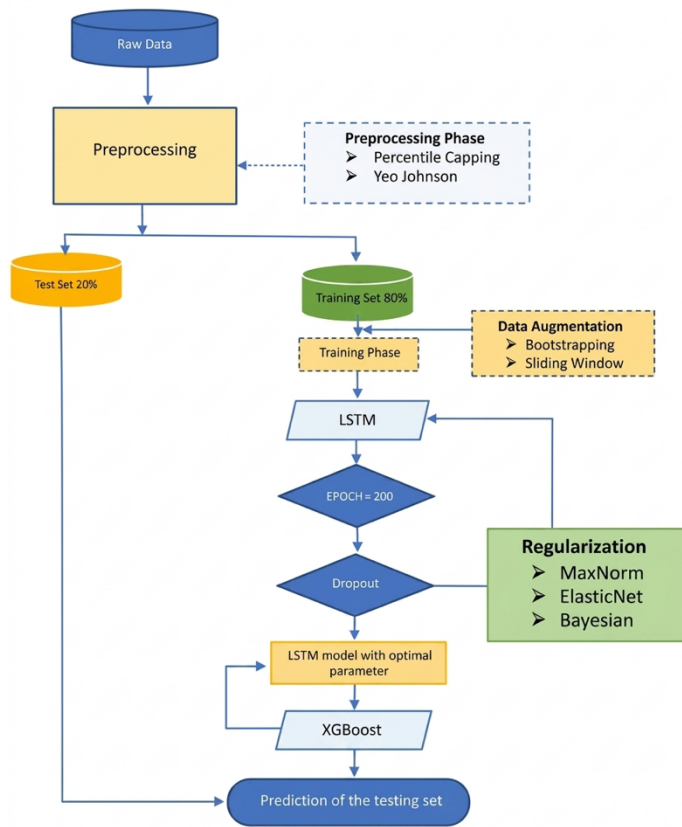


Figure 3. Proposed LSTM_XGBoost Framework with Data Augmentation and Regularization.

Table 1. Hyperparameter Configuration of LSTM–XGBoost Model.

Model	Hyperparameter	Value
LSTM	Number of layers	2
	Units per layer	128, 64
	Activation function	ReLU
	Dense layer	16 units (tanh)
	Optimizer	Adam
	Learning rate	0.0001–0.0003
	Loss function	Huber
	Epochs	200
	Batch size	64
	Dropout	0.3
XGBoost	Regularization	ElasticNet, Bayesian, MaxNorm
	Number of estimators	300
	Learning rate	0.01
	Max depth	3
	Subsample	0.6
	Colsample_bytree	0.6

Feature scaling is performed using z-score normalization based only on training data:

$$\tilde{x} = \frac{x - \mu_{train}}{\sigma_{train}}, \tilde{y} = \frac{y - \mu_{y,train}}{\sigma_{y,train}}, \quad (2)$$

This prevents data leakage from the test set into the training process.

To improve generalization, data augmentation is applied only to the training set using bootstrap resampling with optional Gaussian noise:

$$\mathbf{X}_{aug} = \mathbf{X}_{sampled} + \epsilon, \epsilon \sim \mathcal{N}(0, 0.01^2). \quad (3)$$

The sliding window provides temporal structure, bootstrapping increases sample size, and noise injection introduces controlled variation.

3.5. Data Mining

The data mining phase employs a stacked Long Short-Term Memory (LSTM) network to capture nonlinear temporal dependencies. The architecture consists of two LSTM layers (128 and 64 units), followed by dense layers for regression output. Hyperparameters such as learning rate, regularization strength, and network configuration were tuned empirically based on validation performance. The key hyperparameters of the LSTM and XGBoost models are summarized in Table 1.

Standalone LSTM models were first evaluated using different regularization techniques, including ElasticNet, Bayesian-inspired regularization, and MaxNorm. In this context, Bayesian regularization refers to a probabilistic approach for constraining model parameters.

Based on experimental results (Tables 2 and 3), MaxNorm achieved the best performance and was selected as the primary regularization method. It is applied as a weight constraint to improve training stability and reduce overfitting.

The overall modeling pipeline is illustrated in Fig. 3. The process starts from raw data preprocessing, followed by chronological splitting into training (80%) and testing (20%) sets. Data augmentation techniques, including sliding window and bootstrapping, are applied exclusively to the training set. Regularization methods are incorporated during the LSTM training phase to improve generalization and stability.

A Ensembled LSTM–XGBoost model was then developed. The penultimate dense layer of the LSTM produces latent representations, which are used as input features for XGBoost:

$$\hat{y}_t = \sum_{k=1}^K f_k(\mathbf{z}_t), f_k \in \mathcal{F}, \quad (4)$$

where \mathcal{F} denotes the space of regression trees. This Ensembled approach combines LSTM’s sequential learning capability with XGBoost’s optimization strength.

For benchmarking, additional models were implemented, including standalone XGBoost, ARIMA, Support Vector Regression (RBF kernel), and a naïve persistence model defined as $\hat{y}_t = y_{t-1}$.

3.6. Interpretation and Evaluation

The evaluation phase assesses model performance using multiple metrics:

- 1) **MSE (Mean Squared Error):** Measures the average squared difference between predicted and actual values. A lower MSE indicates a more accurate model. MSE is computed using the formula [32].

$$MSE = \frac{1}{n} \sum (A_t - F_t)^2 \tag{5}$$

- 2) **MAE (Mean Absolute Error):** Measures the average absolute differences between predicted and actual values. MAE provides insight into the model’s accuracy in real terms. MAE is computed [32].

$$MAE = \frac{1}{n} \sum_{i=1}^n |A_t - F_t| \tag{6}$$

- 3) **RMSE (Root Mean Squared Error):** Provides the square root of MSE, making the metric easier to interpret in the same units as the data. RMSE is computed [33].

$$RSME = \sqrt{\frac{1}{n} \sum_{i=1}^n (A_t - F_t)^2} \tag{7}$$

- 4) **R² (Coefficient of Determination):** This indicates how well the model explains the variance in the data. A higher R² indicates that the model is better at predicting the target variable. Mathematical formula for R² [34].

$$R^2 = 1 - \frac{\sum_{i=1}^n (A_t - F_t)^2}{\sum_{i=1}^n (A_t - \bar{A})^2} \tag{8}$$

where A_t represents actual values, F_t predicted values, and \bar{A} the mean of observed values.

These metrics are used to compare the proposed LSTM–XGBoost model with baseline models. Visual comparisons between predicted and actual values are also provided to illustrate model performance and the impact of regularization and ensemble learning on Required_Tikog prediction accuracy.

3. Result and Discussion

This study examines the effects of data augmentation (DA) and MaxNorm regularization on the performance of an LSTM–XGBoost ensemble model. Model performance is evaluated using Mean Squared Error (MSE), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE)

[35]. Lower values of these metrics indicate better predictive performance. MSE is sensitive to large errors and tends to penalize outliers more heavily [36], while RMSE expresses prediction error in the same scale as the original data, making it easier to interpret [37]. MAE provides the average magnitude of errors without emphasizing large deviations [37].

A comparison of various regularization techniques has been conducted through a pre-experimental analysis, with ElasticNet, Bayesian, and MaxNorm Regularization methods tested on the LSTM architecture to identify the most appropriate regularization approach. Based on the results of this pre-experiment, MaxNorm was found to outperform the others; therefore, it will be included in the final Ensembled LSTM–XGBoost Model and in an ablation analysis. As shown in Tables 2 and 3, data augmentation (DA) generally improves the performance of standalone LSTM models across all regularization configurations, resulting in consistently lower error values compared to non-augmented models. The MaxNorm configuration achieved the best results, with training MSE of 0.001236 and testing MSE reduced from 0.060472 to 0.002090 after augmentation. These findings suggest that augmented data can enhance model generalization in LSTM-based architectures.

Furthermore, the unexpected behavior of DA, which can sometimes increase uncertainty in estimates, suggests a need for careful consideration of how these techniques interact within complex architectures [38]. These results suggest that integrating sliding window transformation with bootstrap resampling can serve as an effective augmentation strategy for limited time-series datasets. While sliding window transformation and bootstrap techniques are individually well established in time-series preprocessing and statistical resampling [39], their structured integration as a unified augmentation pipeline for deep sequence models remains relatively underexplored. A similar study by Demir et al. (2021) indicates that these augmentation methods can greatly enhance the model’s ability to correctly predict electricity prices, as evidenced by average MAEs reduced by 2.23%, 2.73%, and 2.97% for the three benchmark models, respectively [40].

Moreover, the sliding window method for data augmentation provides an effective answer to the problems that emerge when working with limited nonlinear data. The quality of synthetic data raises doubts about its ability to perform well across diverse applications, as machine learning tasks often face data imbalance. The generative models can produce high-quality synthetic data, but they may fail to perform effectively when the generated samples do not adequately preserve the underlying data distribution [41]. The evaluation process for synthetic data requires utility measures to determine whether artificial samples enhance model learning or introduce additional data problems, such as noise and bias [42].

Table 2. Effects of data augmentation on training data.

NO AUGMENTATION (Training Set)			
REGULARIZATION	ElasticNet	Bayesian	MaxNorm
MSE	0.009991	0.010270	0.008908
MAE	0.077638	0.078367	0.074448
RMSE	0.099954	0.101340	0.094381
WITH AUGMENTATION (Training Set)			
REGULARIZATION	ElasticNet	Bayesian	MaxNorm
MSE	0.003985	0.008295	0.001236
MAE	0.042725	0.069213	0.017567
RMSE	0.063128	0.091077	0.035151

Table 3. Effects of data augmentation on testing data.

NO AUGMENTATION (Test Set)			
REGULARIZATION	ElasticNet	Bayesian	MaxNorm
MSE	0.072682	0.081415	0.060472
MAE	0.210144	0.231574	0.199156
RMSE	0.269595	0.285334	0.245911
WITH AUGMENTATION (Test Set)			
REGULARIZATION	ElasticNet	Bayesian	MaxNorm
MSE	0.007291	0.011129	0.002090
MAE	0.053549	0.079776	0.023650
RMSE	0.085385	0.105495	0.045780

Table 4. Ablation Study of the Ensembled LSTM–XGBoost Model.

	DA	Max-Norm	MSE	MAE	RMSE	R²
Ensembled without DA and MaxNorm	No	No	0.693183 ± 0.000499	0.664326 ± 0.006664	0.832576 ± 0.000300	0.174054 ± 0.000594
Ensembled with DA only	Yes	No	0.705695 ± 0.002373	0.655163 ± 0.014653	0.840056 ± 0.001412	0.159145 ± 0.002828
Ensembled with MaxNorm only	No	Yes	0.687966 ± 0.004687	0.662330 ± 0.000916	0.829435 ± 0.002826	0.180270 ± 0.005585
Ensembled with DA + MaxNorm	Yes	Yes	0.691785 ± 0.031195	0.652859 ± 0.017336	0.831631 ± 0.018756	0.175718 ± 0.037170

Research shows mixed results on the benefits of untargeted augmentation, yet current data suggest that augmentation success depends on dataset dimensions, feature organization, and regularization configuration. Models trained without augmentation are exposed to a limited data distribution, increasing susceptibility to variance and noise. The limited volume of training data makes it difficult for deep learning model architectures to adjust their multiple hyperparameters, as there is insufficient data to find effective parameter values, leading to model overfitting. The solution of gathering additional training data to combat model overfitting encounters two main obstacles stemming from financial and time constraints in actual deployment scenarios. DA is an effective solution to the problem of overfitting. Users can create multiple versions of their training data using DA, which preserves the original meaning of all data elements [43].

The ablation study of the Ensembled LSTM–XGBoost model is evaluated in Table 4, which shows the effects of

data augmentation and MaxNorm regularization, both individually and together. The Ensembled baseline model has an RMSE of 0.832576 and serves as a basis for evaluating the impact of each component. Results reveal that when MaxNorm is applied independently, the RMSE of 0.829435 is reduced, yielding the highest R² value, indicating that constraining model weights promotes stable learning and improves generalization by reducing overfitting. In contrast, the data augmentation mechanism applied independently does not improve performance, with the RMSE increasing to 0.840056. The results imply that data augmentation, when applied independently, has led to variability that does not correlate well with the Ensembled architecture.

Data augmentation has been cited in the literature to improve model performance, especially when using a limited dataset [39]. This study indicates that the effect of data augmentation varies across models. When MaxNorm and data augmentation are combined, the Ensembled model

Table 5. Final Model Performance Comparison on the Test Set.

Metric	LSTM	XGBoost	Ensembled	ARIMA	SVM	Naive
MSE	0.005169	0.009213	0.002212	1.073942	0.028286	2.092630
MAE	0.052455	0.022787	0.006206	0.918528	0.134859	1.109928
RMSE	0.071899	0.095985	0.047035	1.036312	0.168186	1.446593
R ²	0.995065	0.990861	0.997806	-0.025200	0.972997	-0.997660
WAPE	5.755547	2.776513	0.756130	100.784300	14.797270	121.785400

produces the lowest MAE (0.652859); however, it does not produce a lower RMSE than MaxNorm alone. In general, the findings indicate that MaxNorm consistently improves the generalization performance of Ensembled models, while data augmentation effects vary across models.

Table 5 demonstrates that the Ensembled LSTM–XGBoost solution achieved the best overall accuracy and generalization performance as indicated by the lowest RMSE (0.047035) and highest R² (0.997806) values possible, demonstrating its ability to model temporal dependencies and non-linear relationships in the training data set. Additionally, although the component model of XGBoost achieved comparatively low MAE and WAPE, the Ensembled model produced a balanced set of performance metrics. Thus, combining LSTM-based sequence feature extraction with XGBoost-based prediction capabilities increased predictive accuracy. Conversely, traditional ARIMA, SVM, and Naïve models produced significantly higher error rates when predicting complex time-series data. The Naïve model, in particular, failed to capture any structural patterns in the underlying data, resulting in poor predictive accuracy.

Results also support the effectiveness of combining data augmentation with regularizing techniques, such as MaxNorm, to enhance generalization performance by reducing overfitting and stabilizing the learning process, consistent with the bias–variance trade-off theory [44]. The weight constraints of MaxNorm examples are consistent with prior research findings on controlling complexity and reducing the variance of model outputs; therefore, the MaxNorm method reduced model complexity, thereby improving predictive quality [45].

By applying a sliding-window transformation in conjunction with the bootstrap method to augment the training dataset and increase diversity, researchers have demonstrated in previous studies that this process prevents overfitting and enhances the robustness of models developed from limited datasets [46]. The data augmentation methods depend on the type of model; there is no improvement in predictive quality across all models and configurations, as shown in the ablation data in Table 3.

The proposed Ensembled approach has provided better predictive accuracy than separate models, as shown by the results overall. Additionally, it was found that, with the right combination of structured data augmentation and proper regularization, performance on limited data sets can be improved using a practical method.

The Ensembled LSTM-XGBoost model performs extremely well when sequence-level feature extraction is combined with Ensemble Prediction. Data Augmentation provides a means to increase the Variety of data available for developing the Model; however, the effects depend on the type of Model used to predict the Outcome Variable. On the other hand, the generalization of the Model is substantially More Improved when using Regularization (specifically, MaxNorm). By combining Slides and Bootstrap Resampling, the Ensembled LSTM-XGBoost Model Makes Better Use of Minimal Time-Series Data, helping to achieve a Better Balance between Bias and Variance.

4. Conclusion

The ensemble LSTM-XGBoost framework developed in this research achieves better predictive performance than both individual models and conventional baseline systems for the Required_Tikog demand forecasting. The system achieves better performance by combining sequential deep representation learning with gradient-boosted tree optimization, which identifies both time-based patterns and complex feature relationships. The research established that structured data augmentation methods, which combine sliding-window transformations with bootstrapped resampling, help models learn better by increasing the diversity of their training data. The combination of MaxNorm regularization with L2 penalties and dropout helps control weight sizes and produce stable gradients, thereby preventing model overfitting during training. The data show that temporal sequence modeling, combined with augmentation techniques and norm-based regularization methods, yields a powerful system that improves forecast accuracy for production demand.

Despite the strong predictive performance of the proposed ensembled LSTM–XGBoost framework, several limitations must be acknowledged.

- 1) The modeling system depends mainly on internal transaction and production data but fails to include external factors, which include weather patterns, climate changes, economic trends, supply chain interruptions, and customer purchasing behavior. The production of Tikog naturally depends on both environmental conditions and market dynamics, so the model becomes less dependable when major changes occur in the system or when data shows non-typical patterns.

- 2) The dataset consists of 2940 observations derived from a single operational context. The researchers used data augmentation methods, including sliding-window transformations and bootstrapped sampling with Gaussian perturbations, to enhance their model's ability to generalize, but their synthetic data failed to reproduce the actual diversity observed in natural data distributions. The research needs to verify the existence of its model through validation of external validity and the ability to produce results across different regional settings.
- 3) The sliding window mechanism uses a fixed short-term temporal dependency structure, which assumes that recent historical data holds the most important information for making predictions. The current configuration fails to capture the full extent of long-term relationships between production delays and seasonal patterns that extend beyond the set time limits.
- 4) The model received MaxNorm regularization together with L2 penalties, dropout, and batch normalization to prevent overfitting but hyperparameter selection took place through a restricted search area. The system's lack of effective optimization methods prevents researchers from finding better architectural solutions, thereby limiting the potential for improved results.
- 5) The proposed ensembled system needs extra computational power because its learning process operates through two separate stages, which include deep feature extraction and gradient boosting analysis. The system takes longer to run, which creates difficulties when used for instant deployment or in production environments with limited computing resources. The study evaluates performance using deterministic point-prediction metrics, including MSE, RMSE, MAE, and WAPE, but it does not include probabilistic prediction or uncertainty quantification in its analysis. The model does not generate predictive confidence intervals, which restrict its usefulness for making production plans that require risk assessment and inventory management decisions.

5. Declarations

5.1. Author Contributions

Devine Grace D. Funcion: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data Curation, Writing - Original Draft, Writing - Review & Editing, Visualization; **Marvee Cheska B. Natividad:** Supervision – Dissertation Adviser.

5.2. Institutional Review Board Statement

Not applicable.

5.3. Informed Consent Statement

Not applicable.

5.4. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

5.5. Acknowledgment

Not applicable.

5.6. Conflicts of Interest

The authors declare no conflict of interest.

6. References

- [1] D. Srivastava, H. Pandey, & A. K. Agarwal, "Complex predictive analysis for health care: a comprehensive review.," *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 1, pp. 521-531, 2023. <https://doi.org/10.11591/eei.v12i1.4373>.
- [2] N. Rahul, "Optimizing Claims Reserves and Payments with AI: Predictive Models for Financial Accuracy.," *International Journal of Emerging Trends in Computer Science and Information Technology*, vol. 1, no. 3, pp. 46-55., 2020. <https://doi.org/10.63282/3050-9246.IJETCSIT-V1I3P106>.

- [3] J. M. Kernbach, & V. E. Staartjes, "Foundations of machine learning-based clinical prediction modeling: Part II—Generalization and overfitting.," *Machine Learning in Clinical Neuroscience: Foundations and Applications*, pp. 15-21., 2021. https://doi.org/10.1007/978-3-030-85292-4_3.
- [4] O. A. Montesinos López, A Montesinos López, & J. Crossa, "Overfitting, model tuning, and evaluation of prediction performance.," In *Multivariate statistical machine learning methods for genomic prediction*, pp. 109-139, 2022. https://doi.org/10.1007/978-3-030-89010-0_4.
- [5] I. D., Mienye, & Y. Sun, "A survey of ensemble learning: Concepts, algorithms, applications, and prospects.," *IEEE Access*, vol. 10, pp. 99129-99149, 2022. <https://doi.org/10.1109/ACCESS.2022.3207287>.
- [6] E. Esenogho, I. D. Mienye, T. G. Swart, K., Aruleba, & G. Obaido, "A neural network ensemble with feature engineering for improved credit card fraud detection.," *IEEE Access*, 10, pp. 16400-16407, 2022. <https://doi.org/10.1109/ACCESS.2022.3148298>.
- [7] K. A. Nguyen, W. Chen, B. S. Lin, & U. Seeboonruang, "Comparison of ensemble machine learning methods for soil erosion pin measurements.," *ISPRS International Journal of Geo-Information*, vol. 10, no.1, art. No. 42., 2021. <https://doi.org/10.3390/ijgi10010042>.
- [8] I. D. Mienye, T. G. Swart, & G. Obaido, "Recurrent neural networks: A comprehensive review of architectures, variants, and applications.," *Information*, vol. 15, no. 9, art. No. 517., 2024. <https://doi.org/10.3390/info15090517>.
- [9] M. Krichen, & A. Mihoub, "Long short-term memory networks: A comprehensive survey.," *AI*, vol. 6, no. 9, art. No. 215, 2025. <https://doi.org/10.3390/ai6090215>.
- [10] T. Kandadi, & G. Shankarlingam, "Drawbacks of LSTM Algorithm: A Case Study.," Available at SSRN, p. 5080605., 2025. <https://doi.org/10.2139/ssrn.5080605>.
- [11] T. Liu, J. Bao, J. Wang, & Y. Zhang, "A Ensembled CNN–LSTM algorithm for online defect recognition of CO2 welding.," *Sensors*, vol. 18, no. 12, art. No. 4369, 2018. <https://doi.org/10.3390/s18124369>.
- [12] M. Noorunnahar, A. H. Chowdhury, & F. A. Mila, "A tree based eXtreme Gradient Boosting (XGBoost) machine learning model to forecast the annual rice production in Bangladesh.," *PLoS one*, vol. 18, no. 3, p. e0283452, 2023. <https://doi.org/10.1371/journal.pone.0283452>.
- [13] W. Li, Y. Yin, X. Quan, & H. Zhang, "Gene expression value prediction based on XGBoost algorithm.," *Frontiers in genetics*, vol. 10, pp. 1077, 2019. <https://doi.org/10.3389/fgene.2019.01077>.
- [14] G. Gelete, "Ensembled extreme gradient boosting and nonlinear ensemble models for suspended sediment load prediction in an agricultural catchment.," *Water Resources Management*, vol. 37 no. 14, pp. 5759-5787, 2023. <https://doi.org/10.1007/s11269-023-03629-3>.
- [15] J. Cao, J. Gao, H. Nikafshan Rad, A. S. Mohammed, M. Hasanipanah, & J. Zhou, "A novel systematic and evolved approach based on XGBoost-firefly algorithm to predict Young's modulus and unconfined compressive strength of rock.," *Engineering with computers*, vol. 38, pp. 3829-3845, 2022. <https://doi.org/10.1007/s00366-020-01241-2>.
- [16] D. F. Kandamali, E. Porter, W. M. Porter, A. McLemore, D. O. Kiobia, A. P. Tavandashti, & G. C. Rains, "Ensembled LSTM Method for Multistep Soil Moisture Prediction Using Historical Soil Moisture and Weather Data.," *AgriEngineering*, vol. 7, no. 8, art. No. 260, 2025. <https://doi.org/10.3390/agriengineering7080260>.
- [17] Z. Luo, W. Liu, J. Wu, H. Aiqing, & J. Guo, "Prediction of cold chain loading environment for agricultural products based on K-medoids-LSTM-XGBoost ensemble model.," *PeerJ Computer Science*, vol. 10, p. e2510., 2024. <https://doi.org/10.7717/peerj-cs.2510>.
- [18] F. Dakheel, & M. Çevik, "Optimizing Smart Grid Load Forecasting via a Ensembled Long Short-Term Memory-XGBoost Framework: Enhancing Accuracy, Robustness, and Energy Management.," *Energies*, vol. 18, no. 11, art. No. 2842, 2025. <https://doi.org/10.3390/en18112842>.
- [19] Q. Wen, L. Sun, F. Yang, X. Song, J. Gao, X. Wang, & H. Xu, "Time series data augmentation for deep learning: A survey.," in *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pp. 4653-4660, 2020. <https://doi.org/10.24963/ijcai.2021/631>.
- [20] Y. Zhou, F. Dong, Y. Liu, Z. Li, J. Du, & L. Zhang, "Forecasting emerging technologies using data augmentation and deep learning.," *Scientometrics*, vol. 123, no.1, pp. 1-29, 2020. <https://doi.org/10.1007/s11192-020-03351-6>.
- [21] B. K. Iwana, & S. Uchida, "An empirical survey of data augmentation for time series classification with neural networks.," *Plos one*, vol. 16, no. 7, p. e0254841, 2021. <https://doi.org/10.1371/journal.pone.0254841>.
- [22] T. Kotsilieris, I. Anagnostopoulos, & I. E. Livieris, "Special Issue: Regularization Techniques for Machine Learning and Their Applications.," *Electronics*, vol. 11, no. 4, art. No. 521, 2022. <https://doi.org/10.3390/electronics11040521>.

- [23] R. Balestriero, L. Bottou, & Y. LeCun, "The effects of regularization and data augmentation are class dependent.," in *Proceedings of the 36th International Conference on Neural Information Processing Systems*, art. No. 2745, pp. 37878-37891, 2022. <https://dl.acm.org/doi/abs/10.5555/3600270.3603015>.
- [24] R. Moradi, R. Berangi, B. Minaei, "A survey of regularization strategies for deep models.," *Artificial Intelligence Review*, vol. 53, no. 6, pp. 3947-3986, 2020. <https://doi.org/10.1007/s10462-019-09784-7>.
- [25] A. A., Semenoglou, E. Spiliotis, & V. Assimakopoulos, "Data augmentation for univariate time series forecasting with neural networks.," *Pattern Recognition*, vol. 134, art. No. 109132, 2023. <https://doi.org/10.1016/j.patcog.2022.109132>.
- [26] A. Debnath, G. Waghmare, H. Wadhwa, S. Asthana, & A. Arora, "Exploring generative data augmentation in multivariate time series forecasting: opportunities and challenges," *Solar-Energy*, 2021. https://kdd-milets.github.io/milets2021/papers/MiLeTS2021_paper_7.pdf.
- [27] D. Caragea, "Unlocking the potential of bootstrapping: A journey towards balanced and reliable synthetic data A framework for evaluating Bootstrap in the context of synthetic data generation," Master's thesis, 2024. <https://studenttheses.uu.nl/handle/20.500.12932/45795>.
- [28] V. V. Pozevalkin, & A. N. Polyakov, "A model for predicting the temperature of a machine tool structure by a neural network using the sliding window method.," in *IOP Conference Series: Materials Science and Engineering*, 2021. <https://doi.org/10.1088/1757-899X/1061/1/012035>.
- [29] J. Leukel, J. González, & M. Riekert, "Machine learning-based failure prediction in industrial maintenance: improving performance by sliding window selection.," *International Journal of Quality & Reliability Management*, vol. 40, no.6, pp. 1449-1462, 2023. <https://doi.org/10.1108/IJQRM-12-2021-0439>.
- [30] D. Kartini, F. Abadi, & T. H. Saragih, "Prediksi Tinggi Permukaan Air Waduk Menggunakan Artificial Neural Network Berbasis Sliding Window.," *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, vol. 5, no.1, pp. 39-44, 2021. <https://doi.org/10.29207/resti.v5i1.2602>.
- [31] W. Jiang, L. Ling, D. Zhang, R. Lin, & L. Zeng, "A time series forecasting model selection framework using CNN and data augmentation for small sample data.," *Neural processing letters*, vol. 55, no. 5, pp. 5783-5810, 2023. <https://doi.org/10.1007/s11063-022-11113-z>.
- [32] S. Fitriansyah, Samijan, & D. Kartika, "Medical Product Sales Forecasting for Business Optimization Using Double Exponential Smoothing," *Methods in Science and Technology Studies*, vol. 1, no. 1, pp. 16-23, 2025. <https://doi.org/10.64539/msts.v1i1.2025.24>.
- [33] T. Bibi, H. Zhou, S. Akbar, and L. Awasthi "Ensembled Machine Learning Framework for Joint Prediction of Window Mean and Bit Error Rate in SC-LDPC Decoding.," *Scientific Journal of Engineering Research*, vol. 2, no. 1, pp. 33-49, 2026. <https://doi.org/10.64539/sjer.v2i1.2026.364>.
- [34] M. Migunani, A. Setiawan, and I. Sembiring "Optimizing automated machine learning for ensemble performance and overfitting mitigation," *Aptisi Transactions on Technopreneurship (ATT)*, vol. 7, no. 3, pp. 808-822, 2025. <https://doi.org/10.34306/att.v7i3.763>.
- [35] A. V. Tatachar, "Comparative assessment of regression models based on model evaluation metrics.," *International Research Journal of Engineering and Technology (IRJET)*, vol. 8, no. 9, pp. 2395-0056, 2021. <https://www.irjet.net/archives/V8/i9/IRJET-V8I9127.pdf>.
- [36] D. Chicco, M. J. Warrens, & G. Jurman, "The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation.," *Peerj computer science*, vol. 7, p. e623, 2021. <https://doi.org/10.7717/peerj-cs.623>.
- [37] J. Kaliappan, K. Srinivasan, S. Mian Qaisar, K. Sundararajan, C.-Y. Chang, Suganthan, "Performance evaluation of regression models for the prediction of the COVID-19 reproduction rate.," *Frontiers in Public Health*, vol. 9, 2021. <https://doi.org/10.3389/fpubh.2021.729795>.
- [38] K. H. Huang, P. Orbanz, and M. Austern, "Gaussian and Non-Gaussian Universality of Data Augmentation.," *arXiv preprint arXiv:2202.09134*, 2022. <https://doi.org/10.48550/arXiv.2202.09134>.
- [39] S. Volkova, "An overview on data augmentation for machine learning.," in *International Scientific and Practical Conference Digital and Information Technologies in Economics and Management*, vol. 942, pp 143–154, 2023. https://doi.org/10.1007/978-3-031-55349-3_12.
- [40] S. Demir, K. Mincev, K. Kok, & N. G. Paterakis, "Data augmentation for time series regression: Applying transformations, autoencoders and adversarial networks to electricity price forecasting," *Applied Energy*, vol. 304, art. No. 117695, 2021. <https://doi.org/10.1016/j.apenergy.2021.117695>.
- [41] F. George, M. Khalid, A. Khan, K. Aslansefat, A. R. Rivera, "Evaluating the Effectiveness of Diffusion Models in Cross-Domain Data Augmentation," 2025. <https://doi.org/10.2139/ssrn.5087057>.

- [42] E. Espinosa and A. Figueira, " On the Quality of Synthetic Generated Tabular Data. *Mathematics*," vol. 11, no. 15, art. No. 3278, 2023. <https://doi.org/10.3390/math11153278>.
- [43] S. H. & J. J. C. Oh, "Time-series data augmentation based on interpolation.," *Procedia Computer Science*, vol 175, pp. 64-71, 2020. <https://doi.org/10.1016/j.procs.2020.07.012>.
- [44] R. Dwivedi, C. Singh, B. Yu, & M. J. Wainwright, " Revisiting minimum description length complexity in overparameterized models.," *arXiv preprint arXiv:2006.10189*, 2020. <https://doi.org/10.48550/arXiv.2006.10189>.
- [45] Y. Y. C. Y. J. S. & Y. M. Z. Yang, "Rethinking bias-variance trade-off for generalization of neural networks.," in *International Conference on Machine Learning*, vol. 119, pp. 10767-10777, 2020. <https://proceedings.mlr.press/v119/yang20j/yang20j.pdf>.
- [46] Z. Ju, Y. Chen, Y. Qiang, X. Chen, C. Ju, & J. Yang, " A systematic review of data augmentation methods for intelligent fault diagnosis of rotating machinery under limited data conditions," *Measurement Science and Technology*, vol. 35, no. 12, art. No. 122004, 2024. <https://doi.org/10.1088/1361-6501/ad7a97>.