JOIN (Jurnal Online Informatika)
p-ISSN: 2528-1682, e-ISSN: 2527-9165
Volume 8 Number 1 | June 2023: 107-114
DOI: 10.15575/join.v8i1.1037

Multi-Step Vector Output Prediction of Time Series Using EMA LSTM

Mohammad Diqi[1], Ahmad Sahal[2], Farida Nur Aini[3]
[1] Department of Informatics, Universitas Respati Yogyakarta, Indonesia
[2,3] Department of Information Technology, Universitas Respati Yogyakarta, Indonesia

Article Info

Article history:

Received Sep 28, 2023
Revised May 31, 2023
Accepted June 01, 2023

Keywords:

Deep learning
EMA
LSTM
Multi-step vector output
Time series prediction

ABSTRACT

This research paper proposes a novel method, Exponential Moving Average Long Short-Term Memory (EMA LSTM), for multi-step vector output prediction of time series data using deep learning. The method combines the LSTM with the exponential moving average (EMA) technique to reduce noise in the data and improve the accuracy of prediction. The research compares the performance of EMA LSTM to other commonly used deep learning models, including LSTM, GRU, RNN, and CNN, and evaluates the results using statistical tests. The dataset used in this study contains daily stock market prices for several years, with inputs of 60, 90, and 120 previous days, and predictions for the next 20 and 30 days. The results show that the EMA LSTM method outperforms other models in terms of accuracy, with lower RMSE and MAPE values. This study has important implications for real-world applications, such as stock market forecasting and climate prediction, and highlights the importance of careful preprocessing of the data to improve the performance of deep learning models.

Corresponding Author:

Mohammad Diqi
Department of Informatics, Universitas Respati Yogyakarta
Jl. Laksda Adisucipto KM 6.3 Depok Sleman - Yogyakarta 55281, Indonesia
Email: diqi@respati.ac.id

1. INTRODUCTION

Time series prediction is an important problem in many fields, and deep learning has emerged as a promising approach for improving prediction accuracy. However, the best deep learning models and strategies for multi-step vector output prediction of time series remain unclear. Several studies have investigated this problem, but the findings have been mixed. For the specified time series problems, Chandra et al. [1] discovered that the bidirectional and encoder-decoder LSTM network gives the best performance in accuracy. Yunpeng et al. [2] found that LSTM RNN outperforms traditional models. An and Anh [3] found that the DirREC strategy is better than all other strategies for multi-step ahead forecasting using neural network.

Other studies have focused on the common deep learning architectures used for time series prediction, finding that artificial neural networks (ANNs), long short-term memory (LSTM) models [4], and temporal-convolutional neural networks (TCNNs) [5] are among the most common methods. Lara-Benítez et al. [6] found that the most accurate forecasts are obtained with LSTMs, while CNNs are more efficient. Torres el al. [7] found that feed forward networks, recurrent neural networks (RNNs), and convolutional neural networks (CNNs) are the most common deep learning architectures used for time series prediction [8].

Moreover, feature engineering and data preprocessing have been found to improve the accuracy of time series prediction models. Shen et al. [9] found that the SeriesNet model, which uses deep learning, can learn features of time series data in different interval lengths and results in higher predictive accuracy compared to models using fixed time intervals. It has been discovered by Li et al. [10] that LSTM is more sensitive to feature fusion than SVM, and that the feature's position in the feature sequence can have a noticeable impact on the outcomes of the forecast.

Incorporating external data and domain knowledge is another important consideration when building time series prediction models. DTr-CNN framework [11] and SARIMA [12] are promising approaches for this, with the former being a transfer learning framework with deep architectures and the latter being a statistical method that can outperform other machine learning algorithms.

Additionally, some studies have explored ways of incorporating uncertainty and variability into time series prediction models. Using a general concept of uncertainty, Rußwurm et al. [13] and Li et al. [14] discovered that the distribution of multi-step ahead forecasts can be approximated by Monte Carlo simulation, respectively.

In conclusion, while deep learning models hold promise for time series forecasting, there is still much work to be done to determine the best models and strategies for multi-step vector output prediction [1]. The literature provides a wealth of knowledge and suggestions for improving time series prediction accuracy, and future research should continue to explore these approaches and assess their effectiveness in real-world applications.

Some studies have also explored the use of other machine learning techniques in combination with deep learning to further improve accuracy. Zhang et al. [15] found that their proposed model is superior to other baseline models for time series forecasting tasks. Data retrieval via inferring time-dependent correlation structures was demonstrated by Rosato et al. [16] using their suggested model. When testing their suggested model on several common time series datasets, Shen et al. [9] discovered that it has improved forecasting accuracy and stability.

Despite the promise of deep learning, it is important to note that different models may be best suited for different types of data. For example, Xuan et al. [17] provided detailed results achieved by different indexes, with accuracy, precision, recall, and F1 score being important metrics to consider when evaluating performance. Time series forecasting also takes uncertainty assessment into account. For instance, uncertainties may arise from the presence of noisy data, the difficulty of selecting an appropriate model, challenges in accurately estimating model parameters, unpredictability of future events, complexity and variability of human behavior, or the difficulty of modeling complex data trends such as seasonality or long-term trends. A novel end-to-end Bayesian deep model for time series prediction with uncertainty estimation was suggested by Zhu and Laptev [18].

Though deep learning models have proven useful for predicting time series, it is unclear which models and methods are best for predicting multi-step vector outputs. Terzi et al. [19] proposed a method for building multi-step-ahead prediction models for linear systems using past input-output data. McElroy and Mc Cracken [20] developed the theory of multi-step-ahead forecasting for vector time series that exhibit temporal nonstationarity and co-integration. Bao et al. [21] proposed a novel approach for multi-step-ahead time series prediction using multiple-output support vector regression (M-SVR) with multiple-input multiple-output (MIMO) prediction strategy. There is still a wealth of limitations dealing with uncertainty and variability, and machine learning techniques to boost accuracy available in the published literature. In order to identify the optimal method for multi-stage vector output prediction of time series using deep learning, more study is required.

To address the problem of multi-step vector output prediction of time series using deep learning, we propose a novel detection method called EMA LSTM. The EMA LSTM method is an extension of the LSTM network that incorporates an exponential moving average (EMA) smoothing algorithm to improve the accuracy of multi-step vector output prediction. The EMA LSTM method was designed to address the limitations of existing methods such as LSTM, GRU, RNN, and CNN by improving the stability and robustness of the model, particularly for noisy and non-stationary data.

Our proposed EMA LSTM method offers several contributions. First, it is capable of modeling complex temporal patterns in time series data, making it particularly useful for real-world applications such as financial forecasting and weather prediction. Second, it is robust to noisy and non-stationary data, which are common in many real-world scenarios. Third, it offers a stable and reliable method for multi-step vector output prediction, which is essential for accurate and trustworthy forecasting.

As for the rest of the document, it is structured as follows. Our experimental setup and details on the dataset we used are provided in the Methods part, where we also outline the EMA LSTM approach. We show and analyze experimental results acquired using the proposed method and compare them to the results of the other deep learning models in the Result and Discussion section. In this section, we draw conclusions about the findings and discuss the potential applications of the proposed method. Finally, we summarize the paper's contributions and point the way for future study in the final part.

2. METHOD

2.1. Mathematical Concepts

Exponential Moving Average (EMA): EMA is a smoothing algorithm used to reduce the noise and variation in time series data. It gives more weight to recent data points while discounting older data points. The EMA at time t, denoted as $EMA(t)$, is computed in (1) as follows:

$$EMA(t) = \alpha * X(t) + (1 - \alpha) * EMA(t - 1) \tag{1}$$

where $X(t)$ is the input value at time t, $EMA(t - 1)$ is the EMA at the previous time step, and α is the smoothing factor. α is a value between 0 and 1 and determines the weight given to the current value compared to previous values. There are some different approaches to determine α for calculating the EMA. Mirkamali [22] proposed the maxima nomination sampling (MNS) method and Khan et al. [23] proposed an EWMA control chart for monitoring exponential distributed quality characteristics and derived the upper and lower control limits using the mean and variance of EWMA statistics. You et al. [24] examined the EWMA chart with estimated process parameters and suggested optimal design procedures for minimizing the out-of-control median run length and expected median run length of the EWMA chart with estimated process parameters.

Long Short-Term Memory (LSTM): LSTM is a subclass of recurrent neural network (RNN) that can represent causal relationships over extended periods of time. In it, barriers regulate the entry and exit of data into and out of memory cells, allowing for long-term data storage. LSTM model parameters are as shown in (2).

$$i_t = \sigma(W_i * [h_{t-1}, x_t] + b_i)$$

$$f_t = \sigma\left(W_f * [h_{t-1}, x_t] + b_f\right)$$

$$o_t = \sigma(W_o * [h_{t-1}, x_t] + b_o)$$

$$c_t = f_t * c_{t-1} + i_t * tanh(W_c * [h_{t-1}, x_t] + b_c)$$

$$h_t = o_t * tanh(c_t) \tag{2}$$

where i_t, f_t, and o_t are the input, forget, and output gates, respectively, while c_t and h_t are the cell state and hidden state, respectively. W_i, W_f, W_o, and W_c are the weight matrices, b_i, b_f, b_o, and b_c are the bias terms, and x_t is the input at time t.

EMA LSTM: EMA LSTM is an extension of the LSTM model that incorporates the EMA smoothing algorithm to improve the accuracy of multi-step vector output prediction of time series. The primary difference between standard LSTM and EMA-LSTM lies in how the output is smoothed in EMA-LSTM. In standard LSTM, the model predicts the output directly based on the hidden state of the LSTM cells. However, in EMA-LSTM, the output is smoothed using the EMA algorithm. The EMA takes into account previous predictions and applies a weighting scheme to give more importance to recent predictions, resulting in a smoother output. The EMA LSTM equations are as in (3):

$$i_t = \sigma(W_i * [h_{t-1}, EMA(x_t)] + b_i)$$

$$f_t = \sigma\left(W_f * [h_{t-1}, EMA(x_t)] + b_f\right)$$

$$o_t = \sigma(W_o * [h_{t-1}, EMA(x_t)] + b_o)$$

$$c_t = f_t * c_{t-1} + i_t * tanh(W_c * [h_{t-1}, EMA(x_t)] + b_c)$$

$$h_t = o_t * tanh(c_t) \tag{3}$$

where $EMA(x_t)$ is the EMA of the input at time t.

2.2. EMA LSTM Method

Specifically, this study employs an EMA LSTM technique that makes use of a two-layer architecture consisting of 128 LSTM units. We determine 128 LSTM units through a process of trial and error, where different numbers of units were tested to find a balance between model complexity and generalization capability. To avoid overfitting, an additional dropout layer is linked to the input layer. To train the model, we use stochastic gradient descent (SGD) with a 0.001-rate of learning and a 32-batch size. After some trial and error with the validation group, we settled on a value of 0.3 for the EMA smoothing factor.

2.3. Data Preprocessing

To conduct our research, we obtained the daily stock market prices of Microsoft Corporation (MSFT) between the period of February 1, 2013, and February 17, 2023, from Yahoo Finance in a CSV format. As a preliminary step, we used MinMax scaler to transform the data so that it ranged from 0 to 1. Finally, we applied an 80-10-10 split to the data to create separate training, validation, and test groups. This data split was considered to strike a balance between having enough data for training and validation, while also having sufficient data to reliably test the model's performance. Models were trained on the training set, validated on the validation set to fine-tune hyperparameters and avoid overfitting, and tested on a separate dataset.

2.4. Comparison Methods

We compared the results of the LSTM, Gated Recurrent Unit (GRU), RNN, and Convolutional Neural Network models, as well as the proposed EMA LSTM technique, when applied to time series prediction (CNN). These models were chosen because they are widely employed deep learning frameworks for time series prediction.

When it comes to time series prediction, LSTM is a common RNN architecture that has been shown to be effective. The GRU model is a simplified version of the LSTM that can be trained in less time. The output from the prior time step is fed back into a recurrent neural network (RNN), a simple but effective neural network architecture for modeling time series data. As a type of neural network design, CNN is typically employed for image recognition tasks, but it can also be used with time series data by modeling it as a one-dimensional image.

2.5. Training and Evaluation

The performance of the models was then measured against the test group. Root-mean-squared-error, mean-absolute-error, mean-percentage error, and R-squared (R2) were some of the performance measures we used to assess the models. The root-mean-squared error (RMSE) and the mean absolute percentage error (MAPE) both assess how far off the mark your predictions were. The absolute error between the forecast and the observed value is measured by the mean absolute error (MAE), while the percentage of the dependent variable's variance explained by the independent variable is quantified by the coefficient of determination (R2). These measurements were used to evaluate the efficiency of the proposed EMA LSTM technique and to make comparisons between the various models' outputs.

3. RESULT AND DISCUSSION

3.1. Performance of EMA LSTM Method

On the test set, we used inputs from 60, 90, and 120 days in the past to make predictions for the next 20 and 30 days, and we compared the results to an existing LSTM technique. Table 1 summarizes the findings.

Table 1: Performance of EMA LSTM Method on Test Set

Model	Input/Output	RMSE	MAPE	MAE	R2
EMA LSTM	60/20	0.032	2.85%	0.021	0.95
EMA LSTM	60/30	0.038	3.21%	0.026	0.92

EMA LSTM	90/20	0.030	2.66%	0.019	0.96
EMA LSTM	90/30	0.035	2.98%	0.023	0.93
EMA LSTM	120/20	0.028	2.49%	0.017	0.97
EMA LSTM	120/30	0.033	2.82%	0.021	0.94

The EMA LSTM model exhibits notable performance across various input/output configurations, displaying diminished prediction errors and substantial predictive capability. An augmentation of the input length generally results in enhanced performance, implying that augmenting the model with more historical information augments its capacity for predictive accuracy. Conversely, a marginal rise in performance metrics is observed when extending the output length, indicating a slight decrease in accuracy when forecasting over an extended time horizon. The range of RMSE values, spanning from 0.028 to 0.038, substantiates the model's propensity for generating relatively low prediction errors across the diverse configurations examined. Similarly, the MAPE values, varying between 2.49% and 3.21%, indicate that the average prediction error of the model approximates the actual values within the range of 2.49% to 3.21%. Additionally, the range of MAE values from 0.017 to 0.026 signifies that the average prediction error of the model deviates by merely 0.017 to 0.026 units from the actual values. Furthermore, the R2 values ranging from 0.92 to 0.97 affirm the model's capability to explain 92% to 97% of the variance in the output, attesting to its elevated predictive power. Collectively, these findings corroborate the effectiveness of the EMA LSTM model in accurately predicting the target variable within the context of the provided time series data. The model's strong performance, with limited prediction errors and considerable predictive power, coupled with the observed impact of varying input and output lengths, underscores its potential as a valuable tool for time series prediction tasks.

3.2. Comparison to other methods

Using the same input and output parameters, we compared the effectiveness of the proposed EMA LSTM technique to that of other deep learning models, such as LSTM, GRU, RNN, and CNN.

Table 2: Comparison of EMA LSTM to LSTM

Model	Input/Output	RMSE	MAPE	MAE	R2
LSTM	60/20	0.036	3.02%	0.024	0.91
LSTM	60/30	0.041	3.34%	0.028	0.88
LSTM	90/20	0.032	2.77%	0.021	0.94
LSTM	90/30	0.037	3.10%	0.025	0.90
LSTM	120/20	0.031	2.70%	0.019	0.95
LSTM	120/30	0.035	2.97%	0.023	0.92

Table 3: Comparison of EMA LSTM to GRU

Model	Input/Output	RMSE	MAPE	MAE	R2
GRU	60/20	0.043	3.57%	0.029	0.86
GRU	60/30	0.048	3.91%	0.032	0.83
GRU	90/20	0.039	3.28%	0.026	0.89
GRU	90/30	0.044	3.61%	0.030	0.86
GRU	120/20	0.038	3.16%	0.025	0.90
GRU	120/30	0.042	3.48%	0.028	0.87

Table 4: Comparison of EMA LSTM to RNN

Model	Input/Output	RMSE	MAPE	MAE	R2
RNN	60/20	0.050	4.11%	0.034	0.81
RNN	60/30	0.056	4.64%	0.038	0.78
RNN	90/20	0.045	3.72%	0.029	0.85
RNN	90/30	0.050	4.16%	0.034	0.81
RNN	120/20	0.043	3.57%	0.027	0.87
RNN	120/30	0.048	3.94%	0.032	0.84

Table 5: Comparison of EMA LSTM to CNN

Model	Input/Output	RMSE	MAPE	MAE	R2
CNN	60/20	0.047	3.91%	0.032	0.83
CNN	60/30	0.054	4.43%	0.037	0.79
CNN	90/20	0.042	3.46%	0.028	0.86
CNN	90/30	0.048	3.92%	0.032	0.83
CNN	120/20	0.040	3.28%	0.022	0.89
CNN	120/30	0.045	3.71%	0.025	0.85

We can see from Table 2 that the LSTM model is the closest competitor to the EMA LSTM method, but it still performs worse in terms of all performance metrics. The GRU model performs worse than LSTM, RNN, and CNN. The RNN and CNN models perform worse than LSTM and EMA LSTM in terms of all performance metrics.

Across all input/output configurations, the EMA-LSTM consistently achieves lower values for RMSE, MAPE, and MAE compared to the LSTM model. This signifies that the EMA-LSTM method generates predictions that are closer to the actual values, resulting in lower prediction errors. For instance, when considering the 60/20 input/output configuration, the EMA-LSTM achieves an RMSE of 0.032, while the LSTM model achieves 0.036. Similarly, the MAPE and MAE values for the EMA-LSTM are consistently lower, indicating better accuracy.

Furthermore, the R2 values, which measure the model's ability to explain the variance in the output, are consistently higher for the EMA-LSTM compared to the LSTM model. Higher R2 values signify that the EMA-LSTM captures a larger proportion of the variability in the target variable. For instance, in the 120/20 input/output configuration, the EMA-LSTM achieves an R2 value of 0.97, while the LSTM model achieves 0.95.

These findings highlight that the EMA-LSTM method exhibits notable superiority over the LSTM model in multi-step vector output prediction of time series data. By incorporating the EMA smoothing algorithm, the EMA-LSTM enhances the accuracy and stability of predictions. This suggests that the EMA-LSTM approach holds promise as an effective technique for time series forecasting tasks, providing more reliable and accurate predictions compared to the standard LSTM method.

3.3. *Statistical Tests*

To ensure the accuracy of the findings, we compared the EMA LSTM strategy to the other techniques using a two-tailed paired t-test at a 95% confidence level. In every input/output scenario tested, the EMA LSTM technique was found to have a p-value less than 0.05 and thus to be statistically superior to all other methods. This further demonstrates the efficacy of the EMA LSTM technique over competing approaches to multi-step vector output prediction of time series data.

3.4. Discussion

An unexpected finding in this study was the importance of the EMA technique in enhancing the performance of the LSTM model. These findings are consistent with previous research [25], which has suggested that incorporating the EMA technique with empirical mode decomposition or the LSTM-ARIMA algorithm can improve prediction performance in time series data. However, more research is needed to determine which of these hybrid methods is most effective. A limitation of this study is the relatively small dataset, which may limit the generalizability of the results.

The main finding of this study is that the EMA LSTM method outperforms other commonly used deep learning models in multi-step vector output prediction of time series data. These findings are consistent with previous research suggesting that LSTM models are more accurate than traditional machine learning models [12], [26]–[28]. This study has important implications for real-world applications that require accurate multi-step vector output prediction of time series data, such as stock market forecasting and climate prediction.

The strength of the approach is the use of a novel method, the EMA LSTM, which combines the EMA technique with the LSTM model to improve the accuracy of multi-step vector output prediction of time series data. These findings are consistent with previous research [28]–[31], which has suggested that various methods can be used to improve the accuracy of time series prediction models. However, a limitation of this study is the relatively small dataset, which may limit the generalizability of the results. Furthermore, the study only compared the proposed method to other deep learning models and did not include other traditional machine learning models or statistical models.

4. CONCLUSION

This research concludes that the EMA LSTM method is superior to other widely used deep learning models when applied to the problem of multi-step vector output prediction of time series data. The EMA method is incorporated into the LSTM approach used in the EMA LSTM method to decrease noise in the data and increase precision. The significance of data preprocessing to boost the efficiency of deep learning models is also highlighted.

The limitations of the study include the relatively small dataset used, which could limit the generalizability of the results. Furthermore, the study only compared the proposed method to other deep learning models and did not include other traditional machine learning models or statistical models.

In order to further verify the efficacy of the EMA LSTM technique, future study could make use of larger datasets. Further, the proposed technique could be evaluated in comparison to more conventional machine learning models or statistical models. In conclusion, the EMA LSTM approach could be modified to include additional data and domain expertise, thereby enhancing the precision of time series prediction models in practical settings.

ACKNOWLEDGEMENTS

This research was conducted in the Department of Informatics, Faculty of Science and Technology, Universitas Respati Yogyakarta, Indonesia.

REFERENCES

[1] R. Chandra, S. Goyal, and R. Gupta, "Evaluation of Deep Learning Models for Multi-Step Ahead Time Series Prediction," *IEEE Access*, vol. 9, pp. 83105–83123, 2021, doi: 10.1109/access.2021.3085085.

[2] L. Yunpeng, H. Di, B. Junpeng, and Q. Yong, "Multi-step Ahead Time Series Forecasting for Different Data Patterns Based on LSTM Recurrent Neural Network," in *2017 14th Web Information Systems and Applications Conference (WISA)*, IEEE, Nov. 2017. doi: 10.1109/wisa.2017.25.

[3] N. H. An and D. T. Anh, "Comparison of Strategies for Multi-step-Ahead Prediction of Time Series Using Neural Network," in *2015 International Conference on Advanced Computing and Applications (ACOMP)*, IEEE, Nov. 2015. doi: 10.1109/acomp.2015.24.

[4] M. Diqi, "StockTM: Accurate Stock Price Prediction Model Using LSTM," *ijicom*, vol. 4, no. 1, p. 1, Oct. 2022, doi: 10.35842/ijicom.v4i1.50.

[5] D. M. Ahmed, M. M. Hassan, and R. J. Mstafa, "A Review on Deep Sequential Models for Forecasting Time Series Data," *Applied Computational Intelligence and Soft Computing*, vol. 2022, pp. 1–19, Jun. 2022, doi: 10.1155/2022/6596397.

[6] P. Lara-Ben\' itez, M. Carranza-Garc\' ia, and J. e C. Riquelme, "An Experimental Review on Deep Learning Architectures for Time Series Forecasting," *International Journal of Neural Systems*, vol. 31, no. 03, p. 2130001, Feb. 2021, doi: 10.1142/s0129065721300011.

[7] J. e F. Torres, D. Hadjout, A. Sebaa, F. Mart\' inez-\' Alvarez, and A. Troncoso, "Deep Learning for Time Series Forecasting: A Survey," *Big Data*, vol. 9, no. 1, pp. 3–21, Feb. 2021, doi: 10.1089/big.2020.0159.

[8] M. Diqi, S. H. Mulyani, and R. Pradila, "DeepCov: Effective Prediction Model of COVID-19 Using CNN Algorithm," *SN Computer Science*, vol. 4, no. 4, p. 396, May 2023, doi: 10.1007/s42979-023-01834-w.

[9] Z. Shen, Y. Zhang, J. Lu, J. Xu, and G. Xiao, "A novel time series forecasting model with deep learning," *Neurocomputing*, vol. 396, pp. 302–313, Jul. 2020, doi: 10.1016/j.neucom.2018.12.084.

[10] L. Li, Y. Ou, Y. Wu, Q. Li, and D. Chen, "Research on feature engineering for time series data mining," in *2018 International Conference on Network Infrastructure and Digital Content (IC-NIDC)*, IEEE, Aug. 2018. doi: 10.1109/icnidc.2018.8525561.

[11] R. Ye and Q. Dai, "Implementing transfer learning across different datasets for time series forecasting," *Pattern Recognition*, vol. 109, p. 107617, Jan. 2021, doi: 10.1016/j.patcog.2020.107617.

[12] A. R. S. Parmezan, V. M. A. Souza, and G. E. A. P. A. Batista, "Evaluation of statistical and machine learning models for time series prediction: Identifying the state-of-the-art and the best conditions for the use of each model," *Information Sciences*, vol. 484, pp. 302–337, May 2019, doi: 10.1016/j.ins.2019.01.076.

[13] M. Russwurm, M. Ali, X. X. Zhu, Y. Gal, and M. Korner, "Model and Data Uncertainty for Satellite Time Series Forecasting with Deep Recurrent Models," in *IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium*, IEEE, Sep. 2020. doi: 10.1109/igarss39084.2020.9323890.

[14] L. Li, J. Yan, X. Yang, and Y. Jin, "Learning Interpretable Deep State Space Model for Probabilistic Time Series Forecasting," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, International Joint Conferences on Artificial Intelligence Organization, Aug. 2019. doi: 10.24963/ijcai.2019/402.

[15] S. Zhang, Y. Chen, W. Zhang, and R. Feng, "A novel ensemble deep learning model with dynamic error correction and multi-objective ensemble pruning for time series forecasting," *Information Sciences*, vol. 544, pp. 427–445, Jan. 2021, doi: 10.1016/j.ins.2020.08.053.

[16] A. Rosato, F. Succetti, R. Araneo, A. Andreotti, M. Mitolo, and M. Panella, "A Combined Deep Learning Approach for Time Series Prediction in Energy Environments," in *2020 IEEE/IAS 56th Industrial and Commercial Power Systems Technical Conference (I&CPS)*, IEEE, Jun. 2020. doi: 10.1109/icps48389.2020.9176818.

[17] A. Xuan, M. Yin, Y. Li, X. Chen, and Z. Ma, "A comprehensive evaluation of statistical, machine learning and deep learning models for time series prediction," in *2022 7th International Conference on Data Science and Machine Learning Applications (CDMA)*, IEEE, Mar. 2022. doi: 10.1109/cdma54072.2022.00014.

[18] L. Zhu and N. Laptev, "Deep and Confident Prediction for Time Series at Uber," in *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*, IEEE, Nov. 2017. doi: 10.1109/icdmw.2017.19.

[19] E. Terzi, L. Fagiano, M. Farina, and R. Scattolini, "Learning multi-step prediction models for receding horizon control," in *2018 European Control Conference (ECC)*, 2018, pp. 1335–1340. doi: 10.23919/ECC.2018.8550494.

[20] T. McElroy and M. W. McCracken, "Multistep ahead forecasting of vector time series," *Econometric Reviews*, vol. 36, no. 5, pp. 495–513, 2017, doi: 10.1080/07474938.2014.977088.
[21] Y. Bao, T. Xiong, and Z. Hu, "Multi-step-ahead time series prediction using multiple-output support vector regression," *Neurocomputing*, vol. 129, pp. 482–493, 2014, doi: https://doi.org/10.1016/j.neucom.2013.09.010.
[22] S. J. Mirkamali, "An improved exponentially weighted moving average chart for monitoring proportions using maxima nomination sampling," *Journal of Statistical Computation and Simulation*, vol. 91, no. 2, pp. 282–299, Sep. 2020, doi: 10.1080/00949655.2020.1813294.
[23] N. Khan, M. Aslam, and C.-H. Jun, "A EWMA Control Chart for Exponential Distributed Quality Based on Moving Average Statistics," *Quality and Reliability Engineering International*, vol. 32, no. 3, pp. 1179–1190, Jul. 2015, doi: 10.1002/qre.1825.
[24] H. W. You, M. B. C. Khoo, P. Castagliola, and L. Qu, "Optimal exponentially weighted moving average charts with estimated parameters based on median run length and expected median run length," *International Journal of Production Research*, vol. 54, no. 17, pp. 5073–5094, Feb. 2016, doi: 10.1080/00207543.2016.1145820.
[25] \" Umit Çavuş B\" uy\" ukşahin and Ş. Ertekin, "Improving forecasting accuracy of time series data using a new ARIMA-ANN hybrid method and empirical mode decomposition," *Neurocomputing*, vol. 361, pp. 151–163, Oct. 2019, doi: 10.1016/j.neucom.2019.05.099.
[26] M. Sarkar and A. De Bruyn, "LSTM Response Models for Direct Marketing Analytics: Replacing Feature Engineering with Deep Learning," *Journal of Interactive Marketing*, vol. 53, pp. 80–95, Feb. 2021, doi: 10.1016/j.intmar.2020.07.002.
[27] M. H. Adalat, R. Azmi, and J. Bagherinejad, "An Enhanced LSTM Method to Improve the Accuracy of the Business Process Prediction," *Journal of Industrial Management Perspective*, vol. 10, no. 3, pp. 71–97, Oct. 2020, doi: 10.29252/jimp.10.3.71.
[28] L. Chen, Y. Chi, Y. Guan, and J. Fan, "A Hybrid Attention-Based EMD-LSTM Model for Financial Time Series Prediction," in *2019 2nd International Conference on Artificial Intelligence and Big Data (ICAIBD)*, IEEE, May 2019. doi: 10.1109/icaibd.2019.8837038.
[29] C. Sun, Y. Nong, Z. Chen, D. Liang, Y. Lu, and Y. Qin, "The CEEMD-LSTM-ARIMA Model and Its Application in Time Series Prediction," *Journal of Physics: Conference Series*, vol. 2179, no. 1, p. 012012, Jan. 2022, doi: 10.1088/1742-6596/2179/1/012012.
[30] J. Li, B. Yang, H. Li, Y. Wang, C. Qi, and Y. Liu, "DTDR–ALSTM: Extracting dynamic time-delays to reconstruct multivariate data for improving attention-based LSTM industrial time series prediction models," *Knowledge-Based Systems*, vol. 211, 2021, doi: 10.1016/j.knosys.2020.106508.
[31] Q. Gu, Q. Dai, H. Yu, and R. Ye, "Integrating Multi-Source Transfer Learning, Active Learning and Metric Learning paradigms for Time Series Prediction," *Applied Soft Computing*, vol. 109, 2021, doi: 10.1016/j.asoc.2021.107583.