11-1-2022

LSTM-SDM: An integrated framework of LSTM implementation for sequential data modeling[Formula presented]

Hum Nath Bhandari
Roger Williams University, hbhandari@rwu.edu

Binod Rimal
Florida Atlantic University

Nawa Raj Pokhrel
Xavier University of Louisiana

Ramchandra Rimal
Middle Tennessee State University

Keshab R. Dahal
Truman State University

Recommended Citation

Bhandari, H., Rimal, B., Pokhrel, N., Rimal, R., & Dahal, K. (2022). LSTM-SDM: An integrated framework of LSTM implementation for sequential data modeling[Formula presented]. *Software Impacts, 14*
https://doi.org/10.1016/j.simpa.2022.100396

Software Impacts 14 (2022) 100396

Contents lists available at ScienceDirect

Software Impacts

journal homepage: www.journals.elsevier.com/software-impacts

Original software publication

LSTM-SDM: An integrated framework of LSTM implementation for sequential data modeling

Hum Nath Bhandari [a], Binod Rimal [b,*], Nawa Raj Pokhrel [c], Ramchandra Rimal [d], Keshab R. Dahal [e]

[a] *Department of Mathematics, Roger Williams University, Bristol, RI, USA*
[b] *Department of Mathematical Sciences, Florida Atlantic University, Boca Raton, FL, USA*
[c] *Department of Physics and Computer Science, Xavier University of Louisiana, New Orleans, LA, USA*
[d] *Department of Mathematical Sciences, Middle Tennessee State University, Murfreesboro, TN, USA*
[e] *Department of Statistics, Truman State University, Kirksville, MO, USA*

ARTICLE INFO

Keywords:
LSTM
Prediction
Deep learning
Sequential data modeling
Time series
Data science

ABSTRACT

LSTM-SDM is a python-based integrated computational framework built on the top of Tensorflow/Keras and written in the Jupyter notebook. It provides several object-oriented functionalities for implementing single layer and multilayer LSTM models for sequential data modeling and time series forecasting. Multiple subroutines are blended to create a conducive user-friendly environment that facilitates data exploration and visualization, normalization and input preparation, hyperparameter tuning, performance evaluations, visualization of results, and statistical analysis. We utilized the LSTM-SDM framework in predicting the stock market index and observed impressive results. The framework can be generalized to solve several other real-world time series problems.

Code metadata

Current code version	*Version 1.0*
Permanent link to code/repository used for this code version	https://github.com/SoftwareImpacts/SIMPAC-2022-113
Permanent link to Reproducible Capsule	https://codeocean.com/capsule/3511724/tree/v1
Legal Code License	*MIT License*
Code versioning system used	*Google Colab*
Software code languages, tools, and services used	*Python, Tensorflow, Keras*
Compilation requirements, operating environments & dependencies	Code can be executed in Anaconda or Google Colab
If available Link to developer documentation/manual	*N/A*
Support email for questions	ml.researchgroup2020@gmail.com

1. Introduction

In recent years, the rapid development of deep learning and neural network architectures has transformed many industries such as business, finance, health, security, software, and manufacturing into an automated forecasting process [1–3]. The growth of technology and digitization has significantly transformed numerous industries utilizing machine learning for their prediction process. Many data-driven or algorithm-based solutions, from naive to complex, are applied in various scientific fields [4–10]. These developed algorithms lack a unified and dedicated framework for forecasting sequential data irrespective of any domain. For instance, data collection, data wrangling, data

The code (and data) in this article has been certified as Reproducible by Code Ocean: (https://codeocean.com/). More information on the Reproducibility Badge Initiative is available at https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals.

* Corresponding author.
E-mail addresses: hbhandari@rwu.edu (H.N. Bhandari), brimal2014@fau.edu (B. Rimal), npokhrel@xula.edu (N.R. Pokhrel), ramchandra.rimal@mtsu.edu (R. Rimal), kdahal@truman.edu (K.R. Dahal).

https://doi.org/10.1016/j.simpa.2022.100396
Received 26 June 2022; Received in revised form 18 July 2022; Accepted 22 July 2022

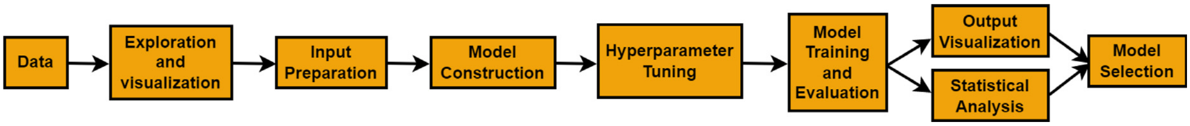

Fig. 1. Schematic diagram of LSTM-SDM framework.

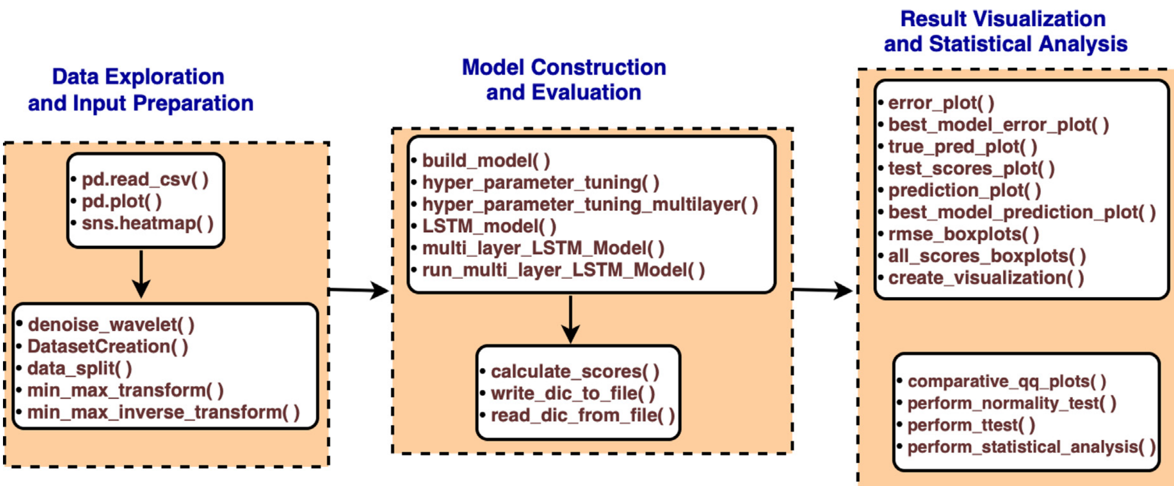

Fig. 2. Workflow analysis of LSTM-SDM functionalities.

preprocessing, algorithmic implementation, and statistical validation are performed without integrating them into a holistic framework. So there is a need to provide a comprehensive toolbox of functions enabling all the required jobs in a unified whole. This study introduces Long Short-Term Memory-Sequential Data Modeling (LSTM-SDM), an integrated deep learning framework that uses single and multilayer LSTM architecture and is designed for the best possible prediction of sequential and time series data.

The automated process of LSTM-SDM is outlined via the schematic diagram in Fig. 1. Furthermore, the sub-components of the process include (a) extraction or collection of essential features required for the prediction of the intended discipline, (b) exploring the data numerically and visually to get its initial insight, (c) preparing the data in such a way that it is compatible with the model architecture, (d) implementation of single and multilayer LSTM cooperative deep learning architecture, and (e) selection of the final optimal model architecture using hyperparameter tuning and statistical validation. The LSTM-SDM framework has been used in recently published research articles to predict the closing price of the stock market indices [11–13].

2. Functionalities

The overall functionality of LSTM-SDM is illustrated in Fig. 2 which can be summarized into three categories: (a) data exploration and input preparation, (b) model construction and evaluation, and (c) result visualization and statistical analysis.

In the first stage, the data is accessed and the trends of response variables are observed through the time series plots. The *sns.heatmap*() outputs the linear correlation between the input features. After observing the overall trend and the degree of variability, the variables can be denoised if necessary using the *denoise_wavelet*(). Since the sequence prediction requires the input sequence and the corresponding output based on the time-step or look-back period, *DataCreation*() is constructed which returns (input, output) pairs based on the given time-step. The *data_split*() helps to partition the original data into train-validation (or train-test) subsets. The resulting subsets are then normalized using *min_max_transformation*(). Later in the process, the model prediction is inverse-transformed into the original scale using *min_max_inverse_transform*().

The second part of the framework consists of the functionalities which are designed to create the LSTM models for a given number of neurons, layers, and other input hyperparameters such as learning rate, batch size, optimization method, etc. The *LSTM_model*() and *multi_layer_LSTM_Model*() utilize the Tensorflow and Keras APIs to build desired LSTM models by implementing the subroutine, called *build_model*(). User can utilize validation data and perform multiple experiments to find the best hyperparameters for the single and multilayer LSTM models using *hyper_parameter_tuning*() and *hyper_parameter_tuning_multilayer*() functions. The tuning process is thorough, automated, and data-driven. In addition, *run_multi_layer_LSTM_Model*() simplifies and automates the executions of the numerous single and multilayer LSTM models taking data, hidden layers, hyperparameters, time step, test split, epochs, number of replicates as inputs and provides the complete performance scores on both train and test data as the output dictionary. The crucial inputs such as the layers, time step, number of features, optimizer, batch size, and learning rate can be easily customized in these routines.

Finally, model selection is an essential component in the computational project. Output results are visualized and statistical analysis is performed. Several functions are available in the result visualization and statistical analysis category which help the users for rich visualization of the results and perform the statistical tests. Reading the output dictionary and creating the prediction plots, error plots, and plots of the performance scores can be obtained easily using a single driver routine, called *create_visualization*(). It combines the multiple subroutines listed in Fig. 2. In addition, the *perform_statistical_analysis*() utilizes the subroutines designed for conducting the normality test and two-sample t-tests. This statistical analysis is helpful to validate the conclusions drawn from other empirical experiments.

3. Impact on related fields

LSTM-SDM is the integrated cooperative platform that supports the sequential data modeling practitioners by predicting the future values with LSTM architecture. It uses data-driven predictive modeling

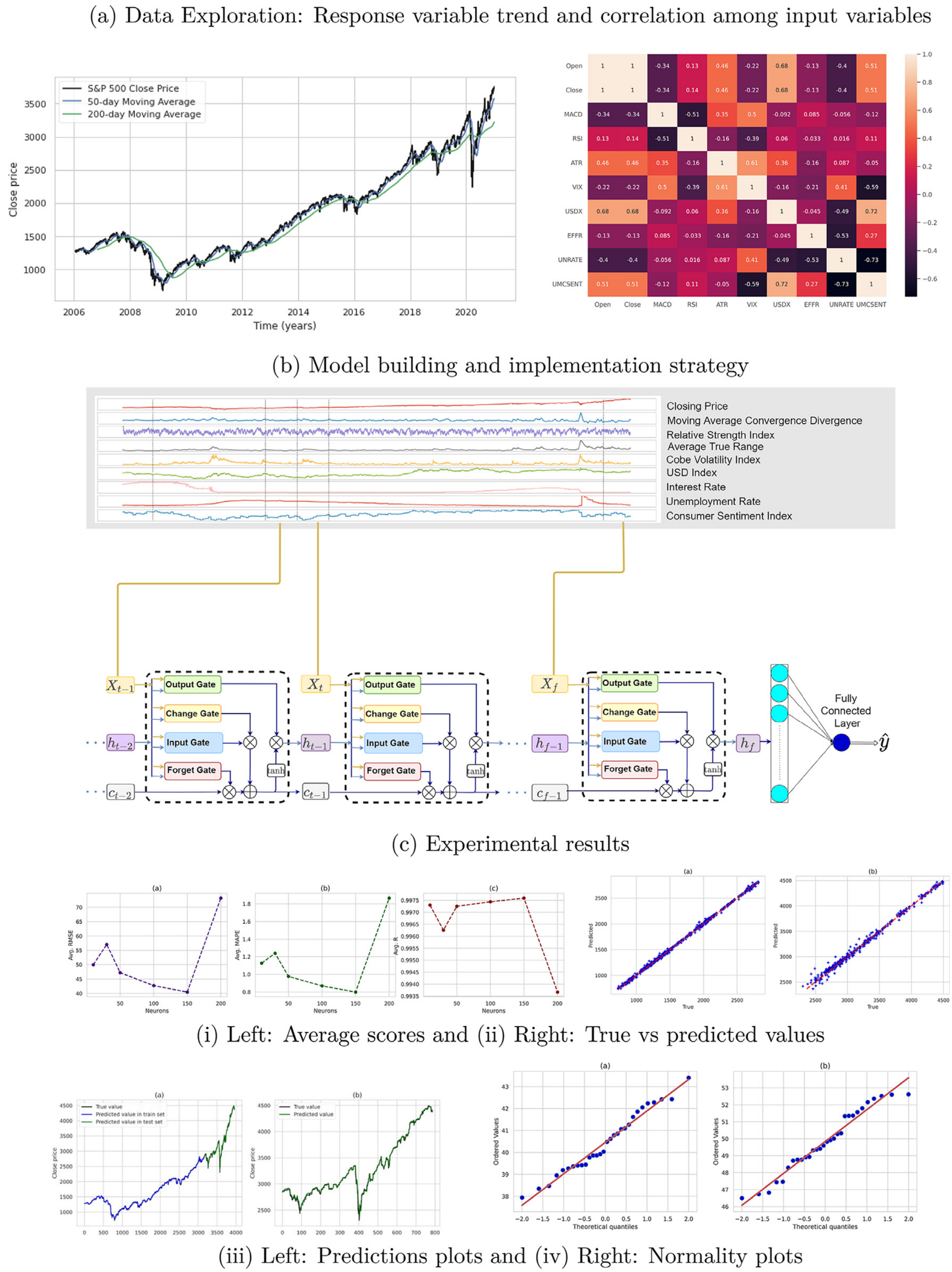

Fig. 3. Implementation of the proposed LSTM-SDM framework for S&P 500 index price prediction [12].

techniques to gain insight from the time series data and provides the projection of the future trend with a minimal time and effort. It also connects with the scientific and research community to solve the sequence prediction problems. The interaction with the platform recommends knowledge of working mechanism of deep neural networks architecture and programming skills. However, one can use the LSTM-SDM with a basic level of familiarity with the model architecture. The start-to-end process from data import to model selection is systematically organized and encapsulated in the framework. Our consolidated object-orientated unified methodology ensures the consistency and quality of the outcome by utilizing standard assessment metrics, high-quality prediction graphs, and multiple optimization techniques. The intermediate results of the prediction are stored in separate files to get access in the future. The developed framework is built using a modular engineering approach and can be generalized regardless of any domain if the data exhibits a time series pattern. Although the current framework implements LSTM architecture, it can be extended to other deep learning architectures. Companies and researchers dealing with sequential data may use the package to analyze the data and make informed decision.

4. Limitations and further improvements

LSTM-SDM provides a user-friendly computing framework for supervised modeling of sequential data. It can be improved by integrating additional state-of-art deep learning models such as Gated Recurrent Unit (GRU), Convolutional Neural Network (CNN), and Restricted Boltzmann Machines (RBM) with the added functionality to compare the several model performance. Another improvement could be incorporating a sub-routine that could perform web scraping to extract textual data such as news articles, customer feedback, and financial reports. These improvements are the motivations for our future work.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

The authors would like to thank the Association of Nepalese Mathematicians in America for creating a collaborative research opportunity that resulted in this work.

Appendix. Illustrative examples

We recently implemented the LSTM-SDM framework in financial market index prediction [12], a popular application of sequential data modeling. In the study, the goal was to predict the next-day closing price of the S&P 500 index by using the multivariate input sequences. Both single layer and multilayer LSTM models were developed using the chosen input variables, and their performances were compared using standard assessment metrics —Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and Correlation Coefficient (R). The Fig. 3 provides the overall strategy of LSTM-SDM implementation and the obtained experimental results. The process of data exploration of response variable (i.e. closing price) and correlation among input variables are presented in Fig. 3(a) where as LSTM-SDM model building and implementation strategy can be found in Fig. 3(b). Similarly, Fig. 3(c) provides some key results obtained from experiments. For more detailed information, readers are encouraged to visit the full paper [12].

References

[1] Yang Lu, Artificial intelligence: a survey on evolution, models, applications and future trends, J. Manag. Anal. 6 (1) (2019) 1–29.

[2] Oludare Isaac Abiodun, Aman Jantan, Abiodun Esther Omolara, Kemi Victoria Dada, Nachaat AbdElatif Mohamed, Humaira Arshad, State-of-the-art in artificial neural network applications: A survey, Heliyon 4 (11) (2018) e00938.

[3] Giang Nguyen, Stefan Dlugolinsky, Martin Bobák, Viet Tran, Alvaro Lopez Garcia, Ignacio Heredia, Peter Malík, Ladislav Hluchỳ, Machine learning and deep learning frameworks and libraries for large-scale data mining: a survey, Artif. Intell. Rev. 52 (1) (2019) 77–124.

[4] Alaa Sagheer, Mostafa Kotb, Time series forecasting of petroleum production using deep LSTM recurrent networks, Neurocomputing 323 (2019) 203–213.

[5] Ashutosh Kumar Dubey, Abhishek Kumar, Vicente García-Díaz, Arpit Kumar Sharma, Kishan Kanhaiya, Study and analysis of SARIMA and LSTM in forecasting time series data, Sustain. Energy Technol. Assess. 47 (2021) 101474.

[6] Samit Bhanja, Abhishek Das, Deep neural network for multivariate time-series forecasting, in: Proceedings of International Conference on Frontiers in Computing and Systems, Springer, 2021, pp. 267–277.

[7] Steefan Contractor, Moninya Roughan, Efficacy of feedforward and lstm neural networks at predicting and gap filling coastal ocean timeseries: Oxygen, nutrients, and temperature, Front. Mar. Sci. (2021).

[8] Srinidhi Hiriyannaiah, G.M. Siddesh, M.H.M Kiran, K.G. Srinivasa, A comparative study and analysis of LSTM deep neural networks for heartbeats classification, Health Technol. 11 (3) (2021) 663–671.

[9] Blake A. Richards, Timothy P. Lillicrap, Philippe Beaudoin, Yoshua Bengio, Rafal Bogacz, Amelia Christensen, Claudia Clopath, Rui Ponte Costa, Archy de Berker, Surya Ganguli, et al., A deep learning framework for neuroscience, Nature Neurosci. 22 (11) (2019) 1761–1770.

[10] Zhaoran Wang, Pearse A. Keane, Michael Chiang, Carol Y. Cheung, Tien Yin Wong, Daniel Shu Wei Ting, Artificial intelligence and deep learning in ophthalmology, Artif. Intell. Med. (2020) 1–34.

[11] Hum Nath Bhandari, Binod Rimal, Nawa Raj Pokhrel, Ramchandra Rimal, Keshab R. Dahal, Rajendra K.C. Khatri, Predicting stock market index using LSTM, 2022, http://dx.doi.org/10.24433/CO.5976425.v1, https://codeocean.com/capsule/3511724/tree/v1.

[12] Hum Nath Bhandari, Binod Rimal, Nawa Raj Pokhrel, Ramchandra Rimal, Keshab R. Dahal, Rajendra K.C. Khatri, Predicting stock market index using LSTM, Mach. Learn. Appl. (2022) 100320.

[13] Nawa Raj Pokhrel, Keshab Dahal, Ramchandra Rimal, Hum Nath Bhandari, Rajendra K.C. Khatri, Binod Rimal, William Edward Hahn, Predicting nepse index price using deep learning models, Mach. Learn. Appl. (2022).