

Forecasting Zhejiang Province's GDP Using a CNN-LSTM Model

Yingliang Wan^{1,*}, Hong Tao² and Li Ma³

¹School of Data Science, Xianda College of Economics & Humanities, Shanghai International Studies University, Shanghai, China

²School of Business, Xianda College of Economics & Humanities, Shanghai International Studies University, Shanghai, China

³School of Economics and Management, Shanxi Jinzhong Institute of Technology, Shanxi, China

*Corresponding author email: 2311160@xdsisu.edu.cn

Abstract: Zhejiang province has experienced notable economic growth in recent years. Despite this, achieving sustainable high-quality economic development presents complex challenges and uncertainties. This study employs advanced neural network methodologies, including Convolutional Neural Networks (CNN), Long Short-Term Memory networks (LSTM), and an integrated CNN-LSTM model, to predict Zhejiang's economic trajectory. Our empirical analysis demonstrates the proficiency of neural networks in delivering reasonably precise economic forecasts, despite inherent prediction residuals. A comparative assessment indicates that the composite CNN-LSTM model surpasses the individual CNN and LSTM models in accuracy, providing a more reliable forecasting instrument for Zhejiang's high-quality economic progression.

Keywords: CNN; LSTM; CNN-LSTM model; Zhejiang province; GDP.

1. Introduction

In the realm of contemporary regional economic research, the intricate nature of economic datasets presents a formidable challenge [1]. Traditional analytical methodologies—such as multiple linear regression, grey correlation models, and time-series analysis—are occasionally beneficial in elucidating economic phenomena but frequently fall short in capturing the subtle complexities inherent in these datasets, subsequently engendering issues related to predictive precision. To surmount this challenge with greater efficacy, scholars have progressively embraced nonlinear dynamical approaches, which have demonstrated substantial promise in revealing hidden interrelations within the data.

Artificial neural networks, as robust nonlinear forecasting instruments, proffer distinctive advantages in deciphering the complexities of regional economic data. Exploiting the structural and algorithmic prowess of neural networks, researchers are enabled to depict a more accurate reflection of economic trends [2, 3]. The integration of this innovative methodology offers a novel outlook in the field of regional economic analysis and furnishes policymakers with more reliable data support, thus positioning itself to play an instrumental role in the strategic formulation of future economic planning and policy.

2. CNN Neural Network

CNNs are often structured with an initial convolutional hidden layer that is tasked with the processing of one-dimensional sequences. In scenarios where the input sequences are particularly extensive, it may be advocated to incorporate an additional convolutional layer to augment the network's depth and capacity for feature extraction. Subsequent to the convolutional operations, a pooling layer is conventionally employed to distill the most salient features from the outputs of the convolutional layers. The architecture proceeds with fully connected layers, which serve to

synthesize the attributes delineated by the convolutional components into a higher-order representation of the input data. To facilitate the transition between the convolutional layers and the fully connected layers, a flattening layer is strategically inserted to transmute the multi-dimensional feature maps into a one-dimensional array, thus preparing the data for the final classification or regression tasks.

2.1. Data Preparation

The data employed in the present study encompasses the GDP of Zhejiang Province, China, for the period extending from 1979 to the current year, as obtained from the Zhejiang Provincial Bureau of Statistics.

This research undertakes a longitudinal analysis of Zhejiang's GDP data spanning 44 years, from 1979 to 2022. For the purposes of model construction, data from the initial 42 years were utilized, whereas data from the subsequent two years were earmarked for model validation and the projection of GDP for the forthcoming years 2023 and 2024. The dataset was divided into a series of input/output configurations, termed "samples," in which triads of temporal data points were used as inputs, and a singular temporal data point was used as the output.

The objective was to forecast the output for the next temporal data point. This division was operationalized through the implementation of a function, `split_sequence()`, specifically devised to partition the univariate data sequence into a multitude of samples, each encompassing a designated number of temporal data points for input and one for output.

2.2. Model Establishment

The central tenet of the proposed model is the delineation of the input shape parameters, which encapsulate the model's anticipatory framework for each data sample in relation to the temporal steps and feature dimensions.

In the context of a univariate data sequence, the feature dimension is inherently singular, thus set to unity. The temporal dimensionality is ascertained through the

parameterization within the `split_sequence()` function during the partitioning phase of the dataset preparation. The expected input shape for individual data samples is articulated within the input shape parameter of the model's inaugural hidden layer. Confronted with an array of samples, the anticipated dimensionality of the input data is structured as a three-dimensional array, encapsulating [samples, temporal steps, feature dimensions]. Nonetheless, the resultant training dataset, X , as procured from the `split_sequence()` function, manifests as a two-dimensional array, [samples, temporal steps], compelling a necessary transformation to introduce an additional feature dimension, thereby aligning with the CNN model's input specifications.

Subsequent to the data preprocessing, a convolutional layer is instantiated, succeeded by a max-pooling layer and a densely connected layer, with the intent to interpret and process the input features. The culmination of this architecture is an output layer engineered to prognosticate a singular numerical value. The model's fitting procedure is executed employing the stochastic gradient descent optimization algorithm, with the objective function being the minimization of mean squared error.

Upon the completion of the model training, the fitted model stands ready for the undertaking of predictive tasks. For instance, deploying input GDP data ranging from the years 2018 to 2020 to extrapolate the subsequent value in the series mandates that the CNN model be presented with a three-dimensional input array, formatted as [samples, temporal steps, feature dimensions]. As a prerequisite for prediction execution, individual input data samples necessitate reshaping to conform to this three-dimensional schema.

The efficacy of the model's temporal forecasting capabilities, predicated on the CNN architecture, is quantitatively appraised through the metric of relative error. The computation of this metric is governed by the formula:

$$\text{Relative Error} = (\text{Prediction} - \text{True Value}) / \text{True Value} \times 100\%$$

The quantitative forecasts engendered by the CNN model with respect to the temporal series, alongside the computed relative errors as dictated by the aforementioned formula, are exhibited in Table 1.

Table 1. Time series prediction table based on CNN model

Temporal Step	Year	Prediction	True Value	Relative Error
1	2021	69292.47	74040.8	-6.41%
2	2022	80151.22	77715.4	3.13%
3	2023	83742.99		
4	2024	91261.14		

3. LSTM Neural Network

The LSTM neural network represents an advanced variant of recurrent neural networks (RNNs), characterized by its utilization of gated units to regulate the information flow through input, forget, and output gates. This intricate mechanism enables the LSTM to selectively preserve or exclude historical information, thus facilitating the dynamic update of cell states in light of newly acquired data. Figure 1 delineates the foundational architecture of the LSTM model[4].

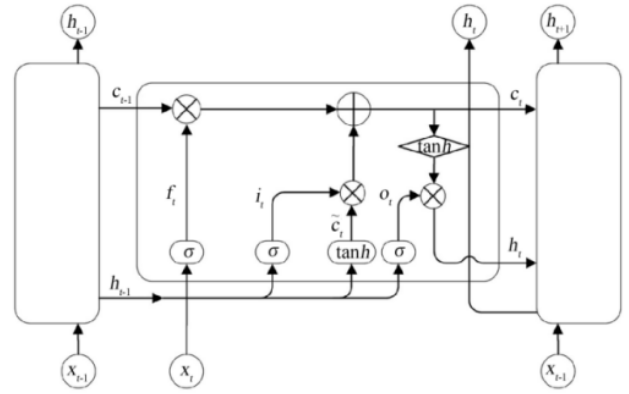


Figure 1. The structure of LSTM

3.1. Data Preparation

The LSTM model is designed to learn a mapping function that transforms a sequence of past observations into future predictions. To this end, it is imperative to restructure the observed data sequence into a format conducive to the LSTM's learning process. In the context of a univariate sequence—such as the GDP measurements from Zhejiang Province—the data are partitioned into numerous input/output configurations, herein termed "samples." This partitioning involves using three sequential time steps as inputs to forecast the subsequent time step as the output. Following this data restructuring, the formulation of the neural network model and its subsequent training can proceed[5].

3.2. Model Establishment

In addressing the challenge of overfitting within time series data, dropout layers are strategically interspersed among the network layers. These layers operate by intermittently obscuring a fraction of the neurons during each training iteration, thereby compelling the network to train within this partially occluded state. With each new iteration, a distinct subset of neurons is concealed, and this iterative process persists until the completion of the training phase. This method effectively attenuates the network's weight magnitudes, thereby bolstering the model's resilience to the disruption of specific neuronal connections and augmenting its capacity for nonlinear forecasting. The calculated relative errors, pursuant to the relative error formula, are tabulated in Table 2.

Table 2. Time series prediction table based on LSTM model

Temporal Step	Year	Prediction	True Value	Relative Error
1	2021	77094.66	74040.8	4.12%
2	2022	79239.64	77715.4	1.96%
3	2023	84626.36		

4. CNN-LSTM Neural Network

The LSTM model is recognized for its proficiency in processing time series data, particularly in capturing temporal dependencies and enhancing data utilization efficiency. Nonetheless, the projection of GDP demands a comprehensive analysis that accounts for both temporal dynamics and the intricate interplay among diverse data features. Complementarily, the CNN model exhibits formidable capabilities in feature extraction, substantially

reducing the quantity and dimensionality of feature parameters, thus mitigating computational demands. Consequently, this research advocates for the amalgamation of LSTM and CNN models to augment the precision of GDP trend forecasts. Initially, CNN and LSTM models are constructed autonomously, and their performance is evaluated using a designated loss function. The ensuing step involves the amalgamation of the models' loss measures in a weighted schema to concurrently minimize total loss and preserve predictive accuracy. Through methodical experimentation, it has been ascertained that assigning weights of 0.1 and 0.9 to the CNN and LSTM models' predictions, respectively, fosters the hybrid model's superlative performance.

Upon the determination of these optimal weights, the predictions from both models are synthesized via a weighted summation, culminating in a synergistic effect. The white

noise test's P-value, derived from extant data, provides the confidence metric for the CNN-LSTM model's predictive output. The forecasted values for the hybrid model are subsequently derived through Equation (1):

$$\hat{R}_t^{CNN-LSTM} = (1 - \alpha)(\alpha \times \hat{R}_t^{CNN} + \beta \times \hat{R}_t^{LSTM}) + R_{t-1}. \quad (1)$$

Here, \hat{R}_t^{CNN} represents the prediction from the CNN branch at time t , \hat{R}_t^{LSTM} represents the prediction from the LSTM branch at time t , and $\hat{R}_t^{CNN-LSTM}$ is the weighted sum of the predictions from both branches. Additionally, $\alpha \in (0,1)$ represents the white noise test P-value, and R_{t-1} denotes the actual total GDP of Zhejiang Province at time $t - 1$. The overall structure of the CNN-LSTM model is illustrated in Figure 2:

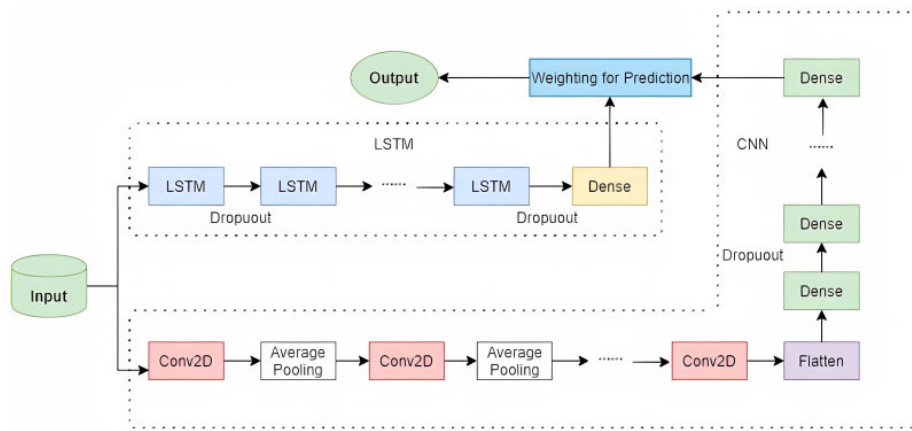


Figure 2: The structure of CNN-LSTM model

The relative errors computed according to the relative error formula are presented in Table 3.

Table 3. Time series prediction table based on CNN-LSTM model

Temporal Step	Year	Prediction	True Value	Relative Error
1	2021	75613.85	74040.8	2.12%
2	2022	78073.85	77715.4	0.46%
3	2023	82993.36		

5. Conclusion

Based on the comprehensive analysis of the previous tables, the prediction results for 2021 based on the CNN-LSTM model (75613.85) are closest to the true value (74040.8) with the smallest absolute relative error (2.12%). In contrast, the results based on the CNN model (69292.47) are farthest from the true value with the largest absolute relative error (-6.41%). In 2022, the CNN-LSTM model's prediction results (78073.85) are closest to the true value (77715.4) with the smallest absolute relative error (0.46%), while the CNN model's results (80151.22) deviate the most from the true

value with the largest absolute relative error (3.13%). Therefore, the CNN-LSTM model demonstrates the best forecasting performance. However, there are still errors between the predicted and actual values for 2021 and 2022. Hence, the CNN-LSTM model is suitable for short-term forecasting and should be used with caution.

References

- [1] Shukla S, Bisht K, Tiwari K, et al. Comparative Study of the Global Data Economy[M]//Data Economy in the Digital Age. Singapore: Springer Nature Singapore, 2023: 63-86.
- [2] Kurani A, Doshi P, Vakharia A, et al. A comprehensive comparative study of artificial neural network (ANN) and support vector machines (SVM) on stock forecasting[J]. Annals of Data Science, 2023, 10(1): 183-208.
- [3] Németh K, Hadházi D. GDP nowcasting with artificial neural networks: How much does long-term memory matter?[J]. arXiv preprint arXiv:2304.05805, 2023.
- [4] He H, Chen L, Wang S. Flight short-term booking demand forecasting based on a long short-term memory network[J]. Computers & Industrial Engineering, 2023, 186: 109707.
- [5] Albeladi K, Zafar B, Mueen A. Time Series Forecasting using LSTM and ARIMA[J]. International Journal of Advanced Computer Science and Applications, 2023, 14(1).