# Applying Convolutional LSTM Network to Predict El Niño Events: Transfer Learning from The Data of Dynamical Model and Observation

Bin Mu, Shaoyang Ma and Shijin Yuan
*School of Software Engineering*
*Tongji University*
*Shanghai, China*
binmu@tongji.edu.cn

Hui Xu
*The State Key Laboratory of Numerical Modeling for Atmospheric Sciences and Geophysical Fluid Dynamics*
*Beijing, China*
xuh@lasg.iap.ac.cn

*Abstract*—**Neural network as a statistical method is widely used for weather forecasting. But for the prediction of El Niño grid data, the data record is short. In this paper, we use deep learning to handle spatiotemporal information and transfer learning to transfer knowledge from dynamical model (Zebiak–Cane model) data to the prediction of realistic El Niño. A ConvLSTM (Convolutional Long Short-Term Memory Network) architecture is constructed to predict the grid data of sea surface temperature and thermocline depth at lead times from 3 to 12 months. Cross-validation is used to evaluate the predictions. The entire data record from 1980 to 2018 is divided into 10 groups and used for training and validation. The experiment results show that transfer learning have a positive impact on the El Niño prediction, especially for the strong Eastern-Pacific type. Compared with the predictions of the Zebiak-Cane model, it can be inferred that the role of the model data in transfer learning is greater than the observation data.**

***Keywords-component; Deep learning; transfer learning; convolutional LSTM; dynamical models; El Niño prediction***

## I. INTRODUCTION

El Niño, the warming phase of the El Niño–Southern Oscillation (ENSO) cycle, is associated with a band of warmer than average ocean water temperatures that periodically develops in equatorial Pacific. It occurs approximately every 4 years and affects the environment and social economy of much of the tropics and worldwide. It has been increasingly recognized that different types of El Niño exist [1]. The two types are the Central-Pacific (CP) type that has sea surface temperature (SST) anomalies near the Date Line and the Eastern-Pacific (EP) type that has anomalies occurred over the Pacific coast of South America. The EP type has been considered as the conventional type of El Niño, and the CP type has been occurring more frequently in recent decades.

Significant progress has been made in ENSO theories and predictions in the last few decades, especially through a whole suite of models with different degrees of complexity. These models can be roughly divided into dynamical models and statistical models [2]. Both dynamical models and statistical models produce useful forecasts for lead times up to 6 months [3]. Dynamical models are based on theory. They use dynamic equations to describe climate phenomena and then solve the equations through computers. Statistical models are based on data. They use some learning methods to solve the non-convex optimization problem for minimizing the errors between predicted and actual values in the historical data.

Due to the development of hardware and technology, deep learning is widely used and has achieved remarkable results in image recognition and speech recognition [4]. Deep learning is very good at discovering intricate interactions in high-dimensional data. There were some successful works in wind speed prediction [5], precipitation prediction [6], and sea surface temperature prediction [7]. But for interannual climate events such as ENSO, data records are not enough. Deep learning requires enormous data to perform complex tasks such as grid sea surface temperature anomalies (SSTA) prediction. Dynamical models containing specific physical mechanisms are good sources of data. And we can get almost infinite data by running the dynamical models. Transfer learning can use the data generated by the dynamics model to improve the predictions of realistic ENSO events. It is the improvement of learning in the target domain with insufficient data through the transfer of knowledge from a related/source domain with sufficient data [8].

A deep learning method based on the data from the Zebiak-Cane (hereafter referred to as ZC) model and observation is introduced for El Niño grid data prediction. A Convolutional Long Short-Term Memory Network (ConvLSTM) model (hereafter referred to as CLM) is used to process the spatiotemporal information. Through the results of the experiment, we try to analyze how transfer learning helps El Niño predictions and which data are more important.

The rest of the paper is organized as follows: Section 2 briefly describes the data used, and the data preprocessing. In Section 3, the CLM and transfer learning are presented. The data are applied in the CLM in Section 4, which discusses the skill of the models to predict El Niño. Finally, the paper concludes with conclusions and discussions in Section 5.

This study is supported by the Fundamental Research Funds for the Central Universities of China (Grant 22120190207).

978-1-7281-6313-0/20/$31.00©2020 IEEE



Authorized licensed use limited to: VIT University- Chennai Campus. Downloaded on January 23,2025 at 15:39:35 UTC from IEEE Xplore. Restrictions apply.

## II. DATA

### A. The Observation Data

Generally, an El Niño is defined as the 3-month running mean of SSTA over the Niño 3.4 region (170ºW - 120ºW and 5ºS - 5ºN, Fig. 1), known as Ocean Niño Index, is exceeding 0.5 for at least 5 consecutive months. Besides SSTA, previous studies on the ZC model also used thermocline depth anomaly (THA) [9]. These two variables are related to the recharge oscillator theory of ENSO [10]. In this study, both the predictors (the input of the CLM) and the predictands (the output of the CLM) are SSTA and THA. 465 observation monthly data are got from January 1980 to September 2018.

To be consistent with the ZC model data, the observation data are processed into the format of the ZC model by linear interpolation. The region of the ZC model is shown in Fig. 1. Attention is focused on the area for SST physics (129.375ºE-84.375ºW and 19ºN-19ºS) with a resolution of 5.625º longitude by 2º latitude, which forms a 20×27 grid. To obtain the monthly anomalies of the observation data, we subtract the climatological monthly means which are calculated based on the entire record of the data set.

### B. The ZC Data

The ZC model first demonstrates the possibility of ENSO prediction by forecasting the 1986/87 El Niño in real-time [11]. Two sets of model integrations are built to generate model data. The first is a free run with no external forcing aside from an initial kick (a 4-month westerly wind anomaly over the western equatorial Pacific) as used by Zebiak and Cane [11]. Then the first 60 months are not considered, to discard the effect of the initial conditions. The second is a forced run, in which the wind stress data from Florida State University wind analysis are used as pointed out by Chen et al. [12]. Finally, we get a ZC simulation data set with 12000 months and a ZC prediction data set with 458 months (Jan 1964 to Feb 2002). The simulation data set is subsequently used to train the model. The prediction data set is used for comparison only and contains predictions for different lead times. The ZC data are originally in the form of anomalies. Both of the ZC data sets appear to be simple SSTA patterns and a regular ENSO period, as described by Lian et al. [13].

### C. Data Normalization

Before the data are fed into the model, we need to normalize it. Data normalization transforms the solution space, making the solution easier to find in current technologies and devices. Min-Max and Z-score are very common for data normalization. But in this study, we choose the method that is specific to SSTA and THA as used by Yu et al. [9]: SSTA divided by 2ºC, and THA divided by 50m.

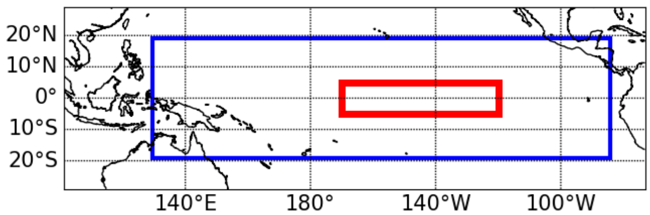


Figure 1. The SST physics area (blue rectangle), and the Niño 3.4 area (red rectangle).

## III. METHOD

### A. CLM Architecture

Fully connected neural networks can theoretically map any non-linear relation if there are enough neurons. When dealing with complex problems, more layers and neurons are needed to improve the model's capabilities. This makes the model difficult to train. To reduce the parameters in the neurons, convolutional neural network (CNN) shares parameters on the spatial scale, while Long Short-Term Memory Network (LSTM) shares them on the time scale [14], [15]. Shared parameters allow more layers, so CNN and LSTM can better handle spatial and temporal information separately. This is why they are called deep learning. Shi et al. [16] proposed ConvLSTM combining CNN and LSTM by changing the parameters into tensor form and using convolution operations instead of the original vector operations. ConvLSTM has shown its powerful ability to capture the spatiotemporal information.

Fig. 2 shows the main architecture of the constructed CLM, which has two inputs and one output. The main body of CLM is an Encoder-Decoder composed of ConvLSTM layers, similar to [16]. SSTA and THA data are input to the Main Input as two color channels. The output of the CLM is also a 20×27×2 tensor consisting of them. The numbers 0-11 representing January to December are entered into Extra Input to provide seasonal cycle information, like [17]. The embedding layer and the concatenate layer are also used in CLM [18]. The rectangles in Fig. 2 represent the model layers and the rounded rectangles represent the input or the output tensors. The number before @ indicates the number of neurons or convolutional kernels. The number after @ indicates the size of the convolutional kernel.

### B. Spatiotemporal Sequence Prediction

From a higher-level perspective, ENSO prediction is intrinsically a spatiotemporal sequence forecasting problem in

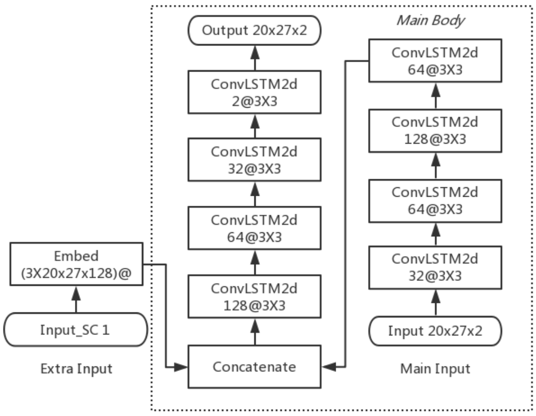


Figure 2. CLM architecture.



Authorized licensed use limited to: VIT University- Chennai Campus. Downloaded on January 23,2025 at 15:39:35 UTC from IEEE Xplore. Restrictions apply.

which both the input and output are spatiotemporal sequences. In this work, the length of the input sequence is 3, as used in [19]. And the length of the output sequence is 1. For multi-step prediction, there are two strategies: Iterated Multi-step (IMS) and Direct Multi-step (DMS) [20]. Generally, the prediction of DMS is more accurate. However, when the lead time is long, DMS may cause the model to fail to train. So we use IMS, which is also called rolling mechanism [21].

The IMS strategy trains a one-step-ahead forecasting model and iteratively feeds the generated samples to the model to get the multi-step-ahead prediction. The CLM m is applied to predict ENSO:

$$S_t = X_{t-2}, X_{t-1}, X_t. \quad (1)$$

$$Y_{t+1} = m(S_t). \quad (2)$$

$$S_{t+1} = X_{t-1}, X_t, Y_{t+1}. \quad (3)$$

Here $X_t$ is the single sample at time t. $S_t$ is the input sequence. $Y_{t+1}$ is the prediction output for time t+1. We build the input and output pairs to supervise learning in this way. The CLM fits a nonlinear process with a lead time of one month.

### *C. Transfer Learning*

When the source and target domains are related, transfer learning works by extracting common features. In the transfer learning process of this study, there are two data set to train the CLM: sufficient model data and insufficient observation data. And two corresponding training stages, pre-training and fine-tuning. Here, we refer to the hypothesis of Erhan et al. [22] to explain the transfer learning mechanism from the perspective of solving non-convex optimization problems.

When the architecture of the model is determined, the solution space is determined. When the training data set is input to the model, the range of the solution is constrained by the loss function. A schematic diagram illustrating the cost function surface is shown in Fig. 3. The point *0* is the starting condition. If we directly use observation data to train the CLM, the solution will fall into the local minima *1* and cannot reach the global minima *3*. We first pre-train the CLM using the ZC model data and could get a relatively good solution *2* because the ZC model data are simpler and more sufficient. Pre-training is to provide a better starting condition for later fine-tuning. This is called *Optimization*. Then the pre-trained CLM is fine-tuned with the observation data. The solution is restricted to a relatively small volume of solution space (purple dotted circle) that is delineated by the boundary of the local basin of attraction of the fine-tuning cost function. Pre-training determines the position of the basins of attraction and has a greater impact on the final solution. This is called *Regularization*.

## IV. EXPERIMENTS AND ANALYSE

We use 10-fold cross-validation to evaluate the CLM on observation data. First, the data is collated into ($[X_{t-2}, X_{t-1}, X_t]$, $X_{t+1}$) pairs for single-step prediction. Then these data pairs are divided into 10 groups without shuffle. There are 46 or 47 pairs of data in each group, consistent with the 4-year cycle of ENSO. Table I shows the types, intensities, and groups of El Niño events during this period. The classification of El Niño refers to [13]. EP and CP represent the type of El Niño. W, M, and S respectively represent that the intensity of El Niño is weak, moderate, and strong. Although the verification groups are not divided according to El Niño events, we classify each group according to the types and intensities of the events it contains. Groups 1, 2, 3, 5, and 10 are EP-S. Groups 4, 6, 7, and 8 are CP or EP-W. Group 9 has no event.

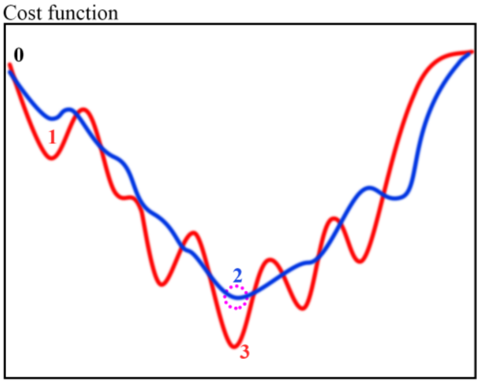


Figure 3. A schematic diagram illustrating the cost function surfaces for sufficient model data (blue) and insufficient observation data (red).

The loss function of the CLM model is Mean Square Error (MSE). Single-step predictions use Root Mean Square Error (RMSE) for evaluation. And for multi-step predictions, the correlation r evaluates the time series of predicted Nino 3.4 SSTA (three-month-moving-averaged):

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i'-Y_i)^2. \quad (4)$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i'-Y_i)^2}. \quad (5)$$

$$r = Cov(X,Y)/X_{std}Y_{std}. \quad (6)$$

In the following experiments, we first discuss the predictions of the ZC model. Then, we use cross-validation methods to verify the performance of the CLM under different validation sets. The python library—Keras is used to build our CLM [23].

### *A. The Predictions of The ZC Model*

We get the predictions of the ZC model with the wind stress forced for the period from January 1980 to February 2002 and some details are different from Chen et al. [12]. To compare with the results of the cross-validation, the predictions of the ZC model are also divided into the same groups. The predictions of group 6 are not complete. Table II shows the correlations of the ZC model. *Average* represents the average of the correlations of the 6 groups. In probabilistic space, most of the arithmetic operations have no meaning. *Average* can



Authorized licensed use limited to: VIT University- Chennai Campus. Downloaded on January 23,2025 at 15:39:35 UTC from IEEE Xplore. Restrictions apply.

TABLE I. THE TYPES AND GROUPS OF EL NIÑO EVENT

| El Niño event | 1982/83 | 86/87/88 | 1991/92 | 1994/95 | 1997/98 | 2002/03 | 2004/05 | 2006/07 | 2009/10 | 2015/16 |
|---|---|---|---|---|---|---|---|---|---|---|
| Type | EP-S | EP-MS | EP-S | CP-M | EP-S | CP-M | CP-W | EP-W | CP-M | EP-S |
| Validation group | 1 | 2, 3 | 3, 4 | 4 | 5 | 6, 7 | 7 | 7, 8 | 8 | 10 |

TABLE II. THE TORRELATION OF ZC PREDICTIONS

| | Lead Time | Validation Set 1 | 2 | 3 | 4 | 5 | 6 | Average | Whole |
|---|---|---|---|---|---|---|---|---|---|
| ZC-Predictions | 0 | 0.7740 | 0.9324 | 0.4058 | -0.3095 | 0.6704 | -0.3712 | 0.3503 | 0.6046 |
| | 3 | 0.8364 | 0.9485 | 0.2984 | -0.1273 | 0.6008 | -0.5356 | 0.3368 | 0.5859 |
| | 6 | 0.8740 | 0.9151 | 0.6428 | 0.2177 | 0.6807 | -0.4789 | 0.4752 | 0.6419 |
| | 9 | 0.9010 | 0.9253 | 0.6843 | 0.1650 | 0.6925 | -0.4687 | 0.4832 | 0.6562 |
| | 12 | 0.8875 | 0.8986 | 0.6181 | 0.2034 | 0.4793 | -0.3134 | 0.4623 | 0.6050 |

only be used as a reference to judge the relative distance. Strong EP El Niño events have occurred in Groups 1, 2, and 5. The predictions of those groups are good. Group 3 has no complete event, and the prediction is average. Moderate CP events occur in Groups 4 and 6, and the predictions are poor. The prediction results show that the ZC model is good at predicting strong EP events.

### B. *Cross-Validation of Transfer Learning*

We use the ZC simulation data to train the CLM to get the CLM-FREE. Then using the observation data to fine-tune the CLM-FREE to obtain the CLM-TRANS as the transfer learning results. We fine-tune all the layers of the CLM-FREE and get the best performance. This may be because the CLM is symmetrical, and the layers near the input and the layers near the output both need to be tuned. The CLM-HIS is trained with observation data directly. The models represent by the CLM-1 means that the validation set is Group 1 and the training sets are the rest.

First, we compare the difference between the CLM-HIS and the CLM-TRANS during the training phase. Fig. 4 shows the training loss of the CLM-10. There are three common measures by which transfer learning could improve machine learning [8]: higher initial performance, higher final performance, and faster learning speed. The CLM-TRANS have obvious first two advantages. Table III shows the RMSE losses during the cross-validation training phase, which are one-month lead time predictions. The RMSE of the CLM with different validation sets is roughly the same and the RMSE of the CLM-TRANS is always better than the CLM-HIS. From the perspective of one-step prediction, transfer learning has obvious benefits.

Then we compare the results of the multi-step prediction using cross-validation. This is to measure the generalization of the CLM. The correlations of the Nino 3.4 SSTA observed and predicted by CLM is calculated in Table IV. The correlations of the CLM with different validation sets are very different. We analyze the results of cross-validation from two points. (1) The relative changes in forecasting skills before and after transfer learning are mentioned. The prediction correlations of the CLM-3, 5, 8, and 10 are completely better after transfer learning. And the prediction of the CLM-1, 2, and 6 are some better. Whereas the predictions of the CLM-4, 7, and 9 are worse. Overall, the *Average* for all the CLM-TRANSs improved. Counting all the prediction results, 24 results are improved after transfer learning and 16 results become worse. This also shows that transfer learning can improve the prediction of ENSO. (2) For all EP-S groups (1, 2, 3, 5, and 10), the predictions are better after transfer learning. For the remaining groups except Group 6, the prediction of some models became worse after transfer learning (4, 7, and 9). The CLM-8 prediction is improved after transfer learning, but it is still poor. The ZC model is good at simulating and predicting the strong EP El Niño, but bad at the CP type. This shows that the pre-training using ZC model data is more important for the final result of transfer learning.

*Optimization* explains why transfer learning can improve prediction results, and *Regularization* explains why pre-training is more important for transfer learning.

## V. CONCLUSIONS AND DISCUSSION

A successful attempt is made in this paper to use deep learning methods to predict the grid data of El Niño. Transfer learning is used to deal with insufficient observation data. Here, we construct a ConvLSTM architecture, especially with the addition of the Extra Input. The CLM use the knowledge learned from the ZC data to predict the realistic El Niño.

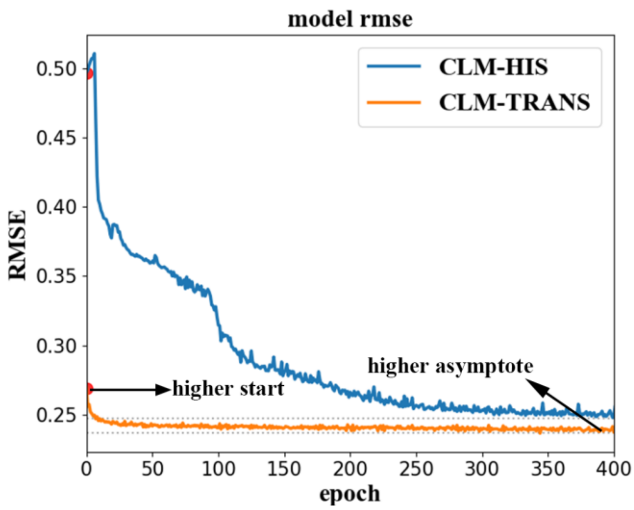


Figure 4. The RMSE of validation during CLM-HIS (blue) and CLM-TRANS (orange) training.



Authorized licensed use limited to: VIT University- Chennai Campus. Downloaded on January 23,2025 at 15:39:35 UTC from IEEE Xplore. Restrictions apply.

TABLE III. THE ONE-MONTH LEAD RMSE LOSS DURING THE CROSS-VALIDATION TRAINING PHASE

| Model | Validation Set | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| CLM-HIS | 0.2902 | 0.2514 | 0.2432 | 0.2578 | 0.2804 | 0.2432 | 0.2386 | 0.2464 | 0.2454 | 0.2472 |
| CLM-TRANS | **0.2730** | **0.2402** | **0.2311** | **0.2408** | **0.2574** | **0.2276** | **0.2228** | **0.2334** | **0.2298** | **0.2366** |

TABLE IV. THE CORRELATION OF MULTI-STEP PREDICTION DURING THE CROSS-VALIDATION EVALUATING PHASE

| Model | Lead Time | Validation Set | | | | | | | | | | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| CLM-HIS | 3 | **0.9161** | **0.8850** | 0.8822 | 0.6101 | 0.9239 | 0.9502 | 0.8263 | 0.8851 | **0.7994** | 0.9337 | 0.8612 |
| | 6 | **0.7534** | **0.6379** | 0.8475 | **0.3142** | 0.7826 | **0.9135** | **0.7419** | 0.6993 | **0.5926** | 0.6659 | **0.6949** |
| | 9 | 0.5527 | 0.5438 | 0.7743 | **0.2009** | 0.6016 | **0.8508** | **0.5674** | 0.0371 | **0.4256** | 0.3344 | 0.4889 |
| | 12 | 0.5895 | 0.7307 | 0.7014 | **-0.2908** | 0.4696 | 0.6240 | **0.5597** | -0.3603 | **0.3034** | 0.2403 | 0.3567 |
| CLM-TRANS | 3 | 0.8796 | 0.8537 | **0.8977** | **0.6909** | **0.9661** | **0.9543** | **0.8842** | **0.8898** | 0.7894 | **0.9451** | **0.8751** |
| | 6 | 0.6679 | 0.5091 | **0.8748** | 0.2870 | **0.9230** | 0.8947 | 0.4782 | **0.7354** | 0.4667 | **0.8493** | 0.6686 |
| | 9 | **0.6478** | **0.5708** | **0.8557** | -0.0705 | **0.8890** | 0.8383 | 0.0457 | **0.3645** | 0.2702 | **0.7623** | **0.5174** |
| | 12 | **0.6453** | **0.7820** | **0.7154** | -0.4270 | **0.9044** | **0.6761** | 0.1005 | **-0.2634** | 0.1736 | **0.5383** | **0.3845** |

Cross-validation is used to evaluate the CLM. The results show that transfer learning can improve the prediction skills of El Niño, and the pre-training using dynamical model data is more important for transfer learning.

For future work, we will investigate the following two aspects: (1) A numerical model that can better simulate and predict the CP El Niño events will be used to generate model data; (2) Using a more appropriate loss function (e.g., Structural similarity) to train the model.

## ACKNOWLEDGMENT

Thanks to Dr. Stephen E. Zebiak for the code of the Zebiak-Cane model. Thanks to IRI/LDEO Climate Data Library for providing observation data and some analysis tools.

## REFERENCES


[1] H. Y. Kao and J. Y. Yu, "Contrasting Eastern-Pacific and Central-Pacific types of ENSO," J. Clim., vol. 22, no. 3, pp. 615–632, 2009.

[2] D. Chen and M. A. Cane, "El Niño prediction and predictability," J. Comput. Phys., 2008.

[3] S. Kirtman, BP and Shukla, J and Balmaseda, M and Graham, N and Penland, C and Xue, Y and Zebiak, "Current status of ENSO forecast skill: A report to the Climate Variability and Predictability (CLIVAR) Working Group on Seasonal to Interannual Prediction," WCRP Informal Rep, vol. 23, pp. 01--31, 2001.

[4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, May 2015.

[5] Q. Hu, R. Zhang, and Y. Zhou, "Transfer learning for short-term wind speed prediction with deep neural networks," Renew. Energy, vol. 85, pp. 83–95, 2016.

[6] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W. Wong, and W. Woo, "Convolutional LSTM Network: A machine learning approach for precipitation nowcasting," pp. 1–9, 2015.

[7] Q. Zhang, H. Wang, J. Dong, G. Zhong, and X. Sun, "Prediction of sea surface temperature using long short-term memory," IEEE Geosci. Remote Sens. Lett., vol. 14, no. 10, pp. 1745–1749, 2017.

[8] L. Torrey and J. Shavlik, "Transfer learning," in Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques, IGI Global, 2010, pp. 242–264.

[9] Y. Yu, W. Duan, H. Xu, and M. Mu, "Dynamics of nonlinear error growth and season-dependent predictability of El Niño events in the Zebiak-Cane model," Q. J. R. Meteorol. Soc., vol. 135, no. 645, pp. 2146--2160, 2009.

[10] C. Wang and J. Picaut, "Understanding ENSO physics—a review," Geophys. Monogr. Ser., vol. 147, pp. 21–48, 2004.

[11] S. E. Zebiak and M. A. Cane, "A model El Ni&ntilde–Southern Oscillation," Mon. Weather Rev., vol. 115, no. 10, pp. 2262–2278, 1987.

[12] D. Chen, S. E. Zebiak, A. J. Busalacchi, and M. A. Cane, "An improved procedure for EI Nino forecasting: implications for predictability," Science (80-. )., vol. 269, no. 5231, pp. 1699–1702, Sep. 1995.

[13] T. Lian, D. Chen, Y. Tang, and Q. Wu, "Effects of westerly wind bursts on El Niño: A new perspective," Geophysical Research Letters, vol. 41, no. 10. pp. 3522–3527, 2014.

[14] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proc. IEEE, 1998.

[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Comput., 1997.

[16] X. SHI, Z. Chen, H. Wang, D.-Y. Yeung, W. Wong, and W. WOO, "Convolutional LSTM network: a machine learning approach for precipitation nowcasting," in Advances in Neural Information Processing Systems 28, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 802–810.

[17] P. D. Nooteboom, Q. Y. Feng, C. López, E. Hernández-García, and H. A. Dijkstra, "Using network theory and machine learning to predict El Niño," Earth Syst. Dyn., vol. 9, no. 3, pp. 969–983, Jul. 2018.

[18] Y. Gal and Z. Ghahramani, "A theoretically grounded application of dropout in recurrent neural networks," in Advances in neural information processing systems, 2016, pp. 1019–1027.

[19] Y. G. Ham, J. H. Kim, and J. J. Luo, "Deep learning for multi-year ENSO forecasts," Nature, vol. 573, no. 7775, pp. 568–572, 2019.

[20] X. Shi and D.-Y. Yeung, "Machine learning for spatiotemporal sequence forecasting: A Survey," arXiv Prepr. arXiv1808.06865, no. 1, pp. 1–20, 2018.

[21] B. Mu, C. Peng, S. Yuan, and L. Chen, "ENSO forecasting over multiple time horizons using ConvLSTM network and rolling mechanism," Proc. Int. Jt. Conf. Neural Networks, vol. 2019-July, no. July, pp. 1–8, 2019.

[22] D. Erhan, Y. Bengio, A. Courville, P.-A. Manzagol, P. Vincent, and S. Bengio, "Why does unsupervised pre-training help deep learning?," J. Mach. Learn. Res., vol. 9, pp. 201–208, 2010.

[23] F. Chollet and others, "Keras." 2015.




Authorized licensed use limited to: VIT University- Chennai Campus. Downloaded on January 23,2025 at 15:39:35 UTC from IEEE Xplore. Restrictions apply.