Predicting storage in the Lake Chad Basin with LSTM *

Noé Carème Fouotsa Manfouo[1], Linke Potgieter[1], and Johanna Hannelie Nel[1]

Stellenbosch University, Stellenbosch, South Africa

Abstract. The number of applications of machine learning methods used for storage forecasting is increasingly showing their ability to outperform process driven methods, which are generally data intensive, with some specific types of data which are difficult to find. The specific structure of long-short term memory, which presents the ability of learning the long-term patterns is a competitive alternative that produce state of the art results in terms of time series forecasting. This paper investigates their application for storage forecasting in the case of the Lake Chad Basin, by comparing a 12, 24 and 36 window size univariate and multivariate LSTM. The results obtained indicate that the multivariate approach, with 36 window size produces the best predictions, in terms of sensitivity to ouliers, with a RMSE of 5.79 m^3, and an overall performances, with a MAPE of 10.73%. A comparison of RMSE, with the existing literature indicate that LSTM may present better abilities to predict streamflows.

Keywords: Forecasting · LSTM · Lake Chad Basin · Storage.

1 Introduction

Lake reservoirs are important freshwater suppliers for irrigation, livestock, and fishing [20, 11]. If provided with commendable accuracy, lake storage predictions can be very useful for water resources management, such as water allocation [14]. This explains the increasing interest on investigating new methods for storage forecasting. The existing literature on lake storage forecasting is organised into two categories, namely (1) space based approaches; and (2) the statistical based approaches. The space based approach distinguishes between direct and indirect calculation methods [8]. Direct methods are used when dealing with reservoirs of a small dimension, but become difficult to implement when considering reservoirs of a lake size with irregular shapes of both underwater and water surface. It results mostly in inaccurate water storage estimation. Indirect methods, namely mid area and prismoidal methods are suitable alternatives [7, 9]. They are based on the subdivision of underwater and surface shapes into well-known geometric figures, such that the overall reservoir volume can be expressed as the sum of surface layers. These approaches are effective and accurate, but lack efficiency

* Graduate School of Economics and Management at Stellenbosch University.

in terms of workload and costs of labour and equipment to collect the necessary data [15]. Besides, these methods are only suitable for short-term estimation of reservoir storage. Statistical methods are able to handle storage prediction for a longer period of time.

Statistical approaches for lake storage forecasting use historical data to mimic the fluctuating behaviors of lake inflows [14]. This family of approaches include multiple linear regression (MLR) [6, 11], stochastic modeling, autoregressive moving average (ARMA) [13], chaos techniques [23, 21, 12], genetic programming (GEP) [11], adaptive-neuro-fuzzy inference system (ANFIS) [6, 13], Nonlinear Local Prediction (NLP) technique, and artificial neural networks (ANNs) [4, 11, 10, 13].

The literature has experienced an increasing number of studies which applies ANNs for storage forecasting, and their comparison with other approaches have shown their superiority over other statistical approaches (see Table 1). Nevertheless, the investigation of other promising architectures of neural networks, especially the Long Short-Term Memory neural network (LSTM) remains very limited. The aim of this study is therefore to implement and deploy a LSTM neural network, for three years ahead storage forecasting in the Lake Chad basin. This implementation considers two scenario, namely (1) the case where only lag variables derived from the storage time series data are considered in the forecasting process; (2) the case where the lag storage variables are combined with other atmospheric variables, to predict future storage in the lake Chad basin. The steps involved in the process include: (1) Data description; (2) Selection of input variables; (3) Formalisation of the LSTM model; (4) Results and discussion; (5) Conclusion.

Case study	RNN	ANFIS	ARMA	AR	-
Lake Egidir	0.27	0.36	0.31	0.51	-
	NLP	ANN	SARIMA	GEP	MLR
Lake Trafford	0.13	0.11	0.11	0.12	0.12
	ANN	ANFIS	ARMA	GEP	-
Lake Iznik	0.073	0.089	0.5	0.07	-

Table 1. Comparison of statistical methods for storage forecasting, based on the RMSE

1.1 Data

Two data sets are used for the Lake Chad storage forecasting, namely (1) 51 years of daily storage [2], recorded from 1 956 to 2 007; (2) 56 years of daily atmospheric data collected [24] from 1 961 to 2 007, containing 31 variables. A summary of these datasets is presented in Table 2.

The variables in the dataset are not all of identical unit. A transformation of all variables into dimensionless values, which range between 0 and 1, helps to

have a dataset with variables of same unit. This makes it much easier to derive the predictor-predictants relationship. The scaling formula is given by

$$x'_{it} = \frac{x_{it} - \min x_i}{\max x_i - \min x_i}, \tag{1}$$

where x'_i is the scaled value of the variable i, at time t; x_{it} is the actual value of the variable i at time t; $\max x_i$ and $\min x_i$ are respectively the maximum and minimum values of a variable i.

Variables Code	**Description**	**Units**
Storage	Daily storage in the Lake Chad basin	m^3
dswr	Direct shortwave radiation	*
lftx	Surface lifted index	*
mslp	Mean sea level pressure	*
p___f	Geostrophic airflow velocity near the surface	*
p___u	Zonal velocity component near the surface	*
p___v	Meridional velocity component near the surface	*
p___z	Vorticity near the surface	*
p_th	Wind direction near the surface	^{o}N
p_zh	Divergence near the surface	*
p5_f	Geostrophic airflow velocity at 500 hPa	*
p5_u	Zonal velocity component at 500 hPa	*
p5_v	Meridional velocity component at 500 hPa	*
p5_z	Vorticity at 500 hPa	*
p5th	Wind direction at 500 hPa	^{o}N
p5zh	Divergence at 500 hPa	*
p8_f	Geostrophic airflow velocity at 850 hPa	*
p8_u	Zonal velocity component at 850 hPa	*
p8_v	Meridional velocity component at 850 hPa	*
p8_z	Vorticity at 850 hPa	*
p8th	Wind direction at 850 hPa	^{o}N
p8zh	Divergence at 850 hPa	*
p500	500 hPa geopotential height	*
p850	850 hPa geopotential height	*
pottmp	Potential temperature	*
pr_wtr	Precipitable water	*
prec	Precipitation total	mm
r500	Relative humidity at 500 hPa height	*
r850	Relative humidity at 850 hPa height	*
rhum	Near surface relative humidity	*
shum	Near surface specific humidity	*
temp	Near surface air temperature	^{o}C

Table 2. List of atmospheric variables collected from the Global Climate Simulation model. * denotes a dimensionless Z-score with mean zero and standard deviation one [17].

2 Selection of input variables

Storage forecasting in the literature is generally performed with its lag variables, streamflows, precipitations, evaporation and infiltration as input variables [14]. The Lake Chad Basin presents incomplete dataset on streamflows, and does not have historical data on evaporation and infiltration. Therefore, atmospheric data as presented in §1.1 are used as canditate predictors. The input selection criteria used to select the suitable predictors for storage forecasting is the correlation coefficient. The high correlations are coloured in dark red or dark blue, for positive or negative correlations respectively. These correlation variables help to remove two types of variables from the dataset, namely the redundant predictors and the atmospheric variables which present little correlation with historical storage. Two predictors are redundants if they both present a correlation coefficient of close to one or minus one. If that is the case, they are likely to have similar explanatory power on the dependent variable. One of them is removed from the dataset, as keeping both may lead to overfitting, also considered as multicollinearity [5].

In this case study, an independent variable (j) is considered as presenting little correlation with the dependent variable (i) if the correlation coefficient ($\rho_{ij} < 0.3$).

Given the two criteria mentioned, 14 out of the 31 atmospheric variables given in Table 2 are combined as an input variables vector x_t at a specific time t, such that $x_t = \left[x_{1t}, x_{2t}, \ldots, x_{14t}\right]^T$. The terms of the input vector x_t are given in Table 3. The variables selected are used to design the 12, 24, and 36 windows size models, in both the univariate and multivariate to produce an output sequence consisting of the predicted monthly storage values, denoted by S_t.

The data of the variables selected are divided into training ($1961 - 2000$), validation ($2001 - 2004$) and testing ($2005 - 2007$) sets.The training and validation sets are used to calibrate the parameters of the storage prediction model, and the test set is used as unseen data, to evaluate the ability of the model to perform well in the real world.

3 LSTM model

LSTM networks are composed of an input layer, hidden layers, and an output layer. The hidden layers consist of memory cells. The structure of an LSTM memory cell is shown in Figure 1. Each of the cells has three gates that maintain and adjust its cell state, c_t, namely the input gate (i_t), the forget gate(f_t), and the output gate (o_t). At every time step t, each of the three gates is presented with the input data $\mathbf{x}_t$ defined in Table 3 including the corresponding lag variables denoted by x_{t-j}, depending on the window size of the LSTM, as well as the output of the memory cells at the previous time step ($h_t - 1$). The selection of input data from $\mathbf{x}_t$ to be added to the cell state is performed at the input gate. The identification of elements of the cell state that are not relevant is performed at the forget gate. The output gate specifies the information from the cell state

Variables	Code	Input variable	Description	Units
Storage	S_t		Daily storage in the Lake Chad basin	km^3
dswr	x_{1t}		Direct shortwave radiation	*
p___v	x_{2t}		Meridional velocity component near the surface	*
p_zh	x_{3t}		Divergence near the surface	*
p8_v	x_{4t}		Meridional velocity component at 850 hPa	*
p8zh	x_{5t}		Divergence at 850 hPa	*
p500	x_{6t}		500 hPa geopotential height	*
p850	x_{7t}		850 hPa geopotential height	*
pr_wtr	x_{8t}		Precipitable water	*
prec	x_{9t}		Precipitation total	mm
r500	x_{10t}		Relative humidity at 500 hPa height	*
r850	x_{11t}		Relative humidity at 850 hPa height	*
rhum	x_{12t}		Near surface relative humidity	*
mslp	x_{13t}		Mean sea level pressure	*
temp	x_{14t}		Near surface air temperature	^oC

Table 3. List of the selected atmospheric data to be used for further analysis in the storage prediction model. The units denoted by * are dimensionless Z-score with mean zero and standard deviation one [17]

that is used as output. The mathematical operations that define the output of the LSTM at time t, h_t, is given by

$$\mathbf{i}_t = \sigma\big(\mathbf{W}_{i,x}\mathbf{x}_t + \mathbf{W}_{i,h}\mathbf{h}_{t-1} + b_i\big) \tag{2}$$

$$\mathbf{f}_t = \sigma\big(\mathbf{W}_{f,x}\mathbf{x}_t + \mathbf{W}_{f,h}\mathbf{h}_{t-1} + b_f\big) \tag{3}$$

$$\mathbf{o}_t = \sigma\big(\mathbf{W}_{o,x}\mathbf{x}_t + \mathbf{W}_{o,h}\mathbf{h}_{t-1} + b_o\big) \tag{4}$$

$$\tilde{\mathbf{c}}_t = tanh\big(\mathbf{W}_{\tilde{c},x}\mathbf{x}_t + \mathbf{W}_{\tilde{c},h}\mathbf{h}_{t-1} + b_{\tilde{c}}\big) \tag{5}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \tilde{\mathbf{c}}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t \tag{6}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot tanh\big(\mathbf{c}_t\big) \tag{7}$$

where $\mathbf{W}_{i,x}$, $\mathbf{W}_{i,h}$,$\mathbf{W}_{f,x}$,$\mathbf{W}_{f,h}$, $\mathbf{W}_{o,x}$, $\mathbf{W}_{o,h}$ and $\mathbf{W}_{\tilde{c},x}$, $\mathbf{W}_{\tilde{c},h}$ are weight matrices, and b_i, b_f, b_o and $b_{\tilde{c}}$ are bias vectors. Furthermore, σ denote the Sigmoid activation function, and $\odot$ denotes the Hadamard product. Finally, the output h_t is used to the estimate storage output S_t at time t by

$$\hat{S}_t = \mathbf{W}_s\mathbf{h}_t \tag{8}$$

where $\mathbf{W}_s$ is a projection matrix to reduce the dimension of $\mathbf{h}_t$. The weights matrices are updated through backpropagation, in an optimisation process, through stochastic gradient descent, such that only the weights that minimise the error function are used in the final model.

The value of the error function (E_k) at the kth iteration ($k \in K$), is given by the difference between the observed storage (S_t) and its predicted/estimated value ($\hat{S}_t$). This difference is squared in order to penalise the outliers in the

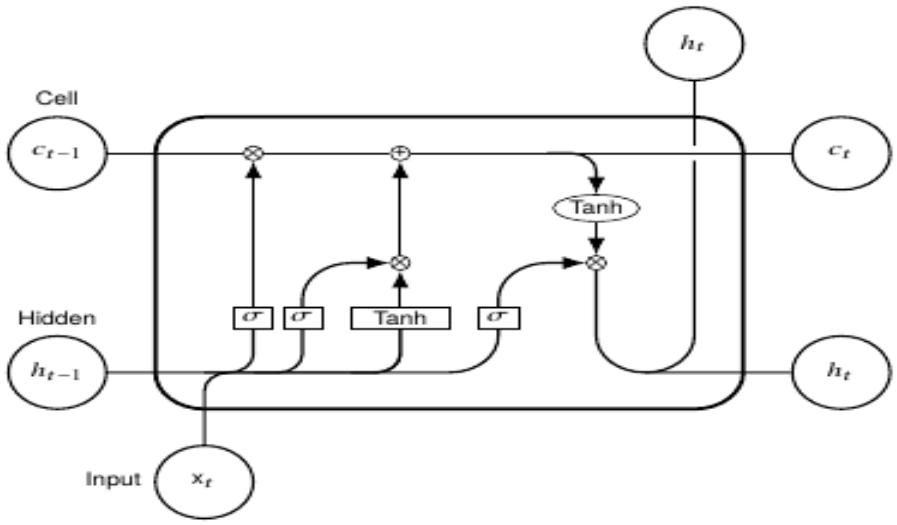

Fig. 1. Memory block for LSTM artificial neural network(Author).

process of weight calibration, such that

$$E_k = \frac{1}{2}\sum_{t=0}^{T}\left(S_t - \hat{S}_t\right)^2 \; k \in K. \tag{9}$$

4 Results

The implementation of the LSTM model is done using pandas, numpy and keras libraries in Python3.6, by following the steps presented in Algorithm 2.

Data preparation and handling is entirely conducted in Python 3.6 [19], relying on the packages numpy [22] and pandas [16]. Our LSTM neural networks are developed with keras [3] on top of Google TensorFlow, a powerful library for large-scale machine learning on heterogenous systems [1]. We make use of scikit learn [18], the implementation of keras in Python. Performance evaluation,namely RMSE and MAPE are computed in Python 3.6, with numpy package. The LSTM network is trained on GPU clusters of Google colaboratory.

4.1 Hyperparameters selection

The architectures obtained from the trial and error search approach, and used for both the univariate and the multivariate models are given in Table 4.

For the univariate LSTM, the 12 and 24 window size options required identical configurations to obtain reasonable accuracy. The 36 window size option, on the other hand, required 20 000 epochs, to obtain reasonable accuracy, with the rest of the configuration the same as with the other two window sizes.

Algorithm 1 LSTM implementation steps in Python for storage forecasting in the Lake Chad Basin

Import Pandas,numpy,keras,sklearn,matplotlib.
Input: Selected Predictors x_t;
for all variables in x_t **do**:
Normalise data to scale $[0;1]$
Split Data into training, validation and test subsets
Use the training and validation sets to define the LSTM model
Compile and fit the LSTM model
Tune parameters with Stochastic Gradient Descent (SGD) until $\Delta E_k \to 0$
Save the model's parameters
Use the saved model to perform the predictions on the test set
Perform 30 simulations and store the output
Compute average predictions of the simulated outputs
Save average simulations
Compute RMSE and MAPE =0

Fig. 2. Algorithm LSTM artificial neural network.

Univariate LSTM							
Window size	Nodes	Epochs	Lr	Hidden Layer	Optimizer	Activation	Dropout
12	8	10000	0.001	1	adam	relu	0.2
24	8	10000	0.001	1	adam	relu	0.2
36	8	20000	0.001	1	adam	relu	0.2
Multivariate LSTM							
Window size	Nodes	Epochs	Lr	Hidden Layer	Optimizer	Activation	Dropout
12	50	2500	0.001	2	adam	tanh	0.2
24	50	2500	0.001	2	adam	tanh	0.2
36	50	2500	0.001	2	adam	tanh	0.2

Table 4. Configuration of LSTM architectures for storage forecast.

For the multivariate LSTM, the 12, 24 and 36 window size options required identical configurations to obtain reasonable accuracy, with 50 nodes, 2 500 epochs, a learning rate (Lr) of 0.001, 2 hidden layers, a hyperbolic tangent optimizer and a dropout of 20 percent.

Multivariate LSTM presents a more complex architecture, with a higher number of nodes and hidden layers, as well as different activation functions. Nevertheless, the smaller number of epochs in the multivariate architecture suggests less computing time, as opposed to the univariate approach. This shows that additional features in the network helps the algorithm to easily identify the inherent patterns between the dependent and independent variables.

The suitability of selected hyperparameters is proven on the loss function plot (see Figure 3 and 4), by both the absence of overfitting or underfitting between the training set and the validation set, as well as an asymptotic variation of the average loss function to zero, as the number of epochs reaches the selected threshold.

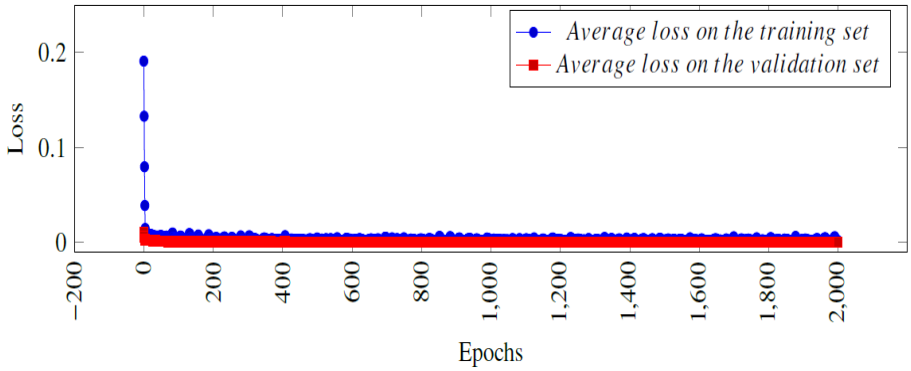

Fig. 3. Plots of loss function values for the training and validation of the univariate storage prediction model with LSTM.

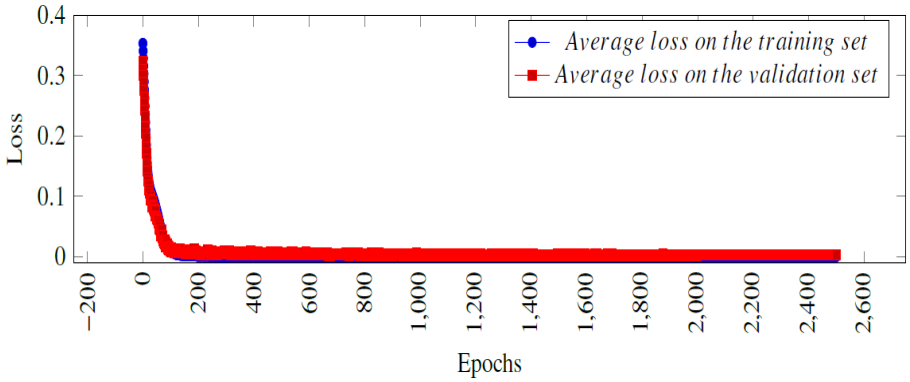

Fig. 4. Plots of loss function values for the training and validation of the multivariate storage prediction model with LSTM.

4.2 Evaluation and comparison of model performances

An ensemble method is used to determine both the stability and the performances of the storage prediction models. The average performance metrics of 30 simulations performed with the best univariate and multivariate LSTM, are given in Table 5.

Results indicate that the multivariate LSTM with window size 36 produce the smallest RMSE (5.79) and MAPE % (10.73) on the test set, as compared to the best performing univariate approach, which show a RMSE of 6.3m^3, and a MAPE of 37.15% on the test set. The multivariate LSTM with window size 36 is subsequently used for storage forecasting in the net water supply estimation model. Output from this model may be used to forecast lake storage for three years ahead, with an estimated accuracy of 89.27%. The graphical representation

Univariate LSTM						
Window size	Train		Validation		Test	
	RMSE	MAPE	RMSE	MAPE	RMSE	MAPE
12	9.4	28.57	18.66	65.2	40.09	65.19
24	9.28	30.05	19.35	58.75	6.79	39.85
36	8.27	36.66	12	39.58	6.3	37.15
Multivariate LSTM						
Window size	Train		Validation		Test	
	RMSE	MAPE	RMSE	MAPE	RMSE	MAPE
12	18.4	16.5	16.14	13.99	17.99	16.28
24	1.73	4.49	1.04	1.77	14.38	16.07
36	12.99	9.2	4.87	7.45	5.79	10.73

Table 5. LSTM accuracy for three years storage forecast.

of the model output, provided in Figure 5 and Figure 6 are visual evidences of the lather interpretations.

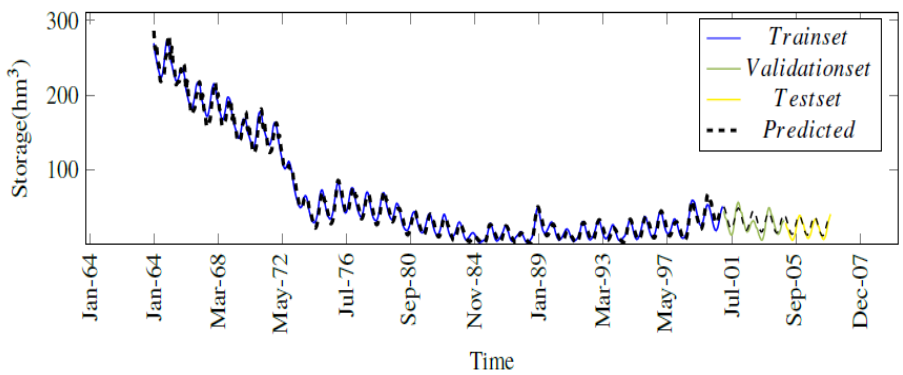

Fig. 5. Univariate LSTM model outputs for three years ahead storage forecast.

5 Conclusion

The aim of this paper was to compare the univariate and multivariate Long Short-Trem Memory artificial neural networks (LSTM) for storage forecasting in the Lake Chad basin. The data of 51 years lenght, were used for the purpose. For both the univariate and multivariate approaches three window size models were investigated, namely 12, 24 and 36. The 36 months window size model was found to be the most accurate, with and average RMSE of 6.3 m^3 and an average MAPE of 37.15% for the univariate approach, and a mean RMSE of 5.79 m^3 as well as an average MAPE of 10.73% for the multivariate approach. A comparison

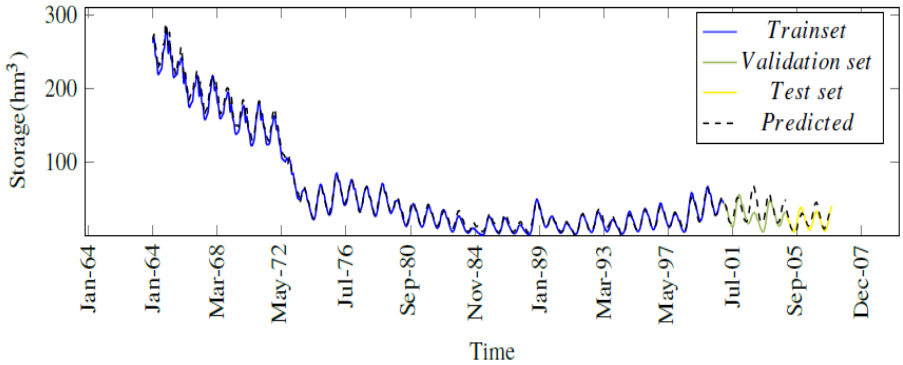

Fig. 6. Multivariate LSTM model outputs for three years ahead storage forecast.

Reservoir names	RMSE (m^3)	Forecasting method
Lake Buchanan	19.04	RAPID
Canyon Lake	12.19	RAPID
Livingston Lake	58.24	RAPID
Egirdir Lake	16.45	ANFIS
Egirdir Lake	11.48	RNN
Egirdir Lake	16.96	AR
Egirdir Lake	16.72	ARMA
Lake Trafford	28.97	Lyapunov exponent
Lake Chad	5.79	Multivariate LSTM

Table 6. Sample of RMSE of storage forecasting models [6, 14, 11].

of the prediction accuracy between the univariate and the multivariate LSTM for three years storage forecasting indicates an outperformance of the multivariate approach. A further comparison of the 36 window size multivariate LSTM accuracy with a sample of previous studies identified in the literature (see Table 6) also demonstrate a better RMSE for the multivariate LSTM proposed in this paper. Hence, multivariate LSTM, with atmospheric predictors, may present a promising approach to obtain state of the art accuracy, in terms of long-term storage predictions. This also show that large scale atmospheric variables, which are easier to obtain, may stand as suitable predictors for storage forecasting.

References

1. Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M, K.M., Levenberg, J., Monga, R., Moore, S., G, D., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., Zheng, X.: Tensorflow: A system for large-scale machine learning **16**, 265–283 (2016)
2. Bader, J.C., Lemoalle, J., Leblanc, M.: Modèle hydrologique du lac tchad. Hydrological Sciences Journal–Journal des Sciences hydrologiques **56**(3), 411–425 (2011)

3. Chollet, F.: Deep Learning mit Python und Keras: Das Praxis-Handbuch vom Entwickler der Keras-Bibliothek. MITP-Verlags GmbH & Co. KG (2018)
4. Faber, B.A., Stedinger, J.R.: Ssdp reservoir models using ensemble streamflow prediction (esp) forecasts. In: Bridging the Gap: Meeting the World's Water and Environmental Resources Challenges. pp. 1–10 (2001)
5. Graham, M.H.: Confronting multicollinearity in ecological multiple regression. Ecology **84**(11), 2809–2815 (2003)
6. Güldal, V., Tongal, H.: Comparison of recurrent neural network, adaptive neuro-fuzzy inference system and stochastic models in eğirdir lake level forecasting. Water resources management **24**(1), 105–128 (2010)
7. Harris, A.R., Mason, I.M.: Lake area measurement using avhrr a case study. International Journal of Remote Sensing **10**(4-5), 885–895 (1989)
8. Hayashi, M., der Kamp, G.V.: Simple equations to represent the volume–area–depth relations of shallow wetlands in small topographic depressions. Journal of hydrology **237**(1), 74–85 (2000)
9. Kalvit, S.K., Kulkarni, S.N.: Remote sensing for monitoring sedimentation in lakes in maharashtra. Lake 2010: Wetlands, Biodiversity and Climate Change (2010)
10. Khan, M.S., Coulibaly, P.: Application of support vector machine in lake water level prediction. Journal of Hydrologic Engineering **11**(3), 199–205 (2006)
11. Khatibi, R., Ghorbani, M., Naghipour, L., Jothiprakash, V., Fathima, T., Fazelifard, M.: Inter-comparison of time series models of lake levels predicted by several modeling strategies. Journal of Hydrology **511**, 530–545 (2014)
12. Khatibi, R., Ghorbani, M.A., Aalami, M.T., Kocak, K., Makarynskyy, O., Makarynska, D., Aalinezhad, M.: Dynamics of hourly sea level at hillarys boat harbour, western australia: a chaos theory perspective. Ocean Dynamics **61**(11), 1797–1807 (2011)
13. Kisi, O., Shiri, J., Nikoofar, B.: Forecasting daily lake levels using artificial intelligence approaches. Computers & Geosciences **41**, 169–180 (2012)
14. Lin, P., Yang, Z.L., Cai, X., David, C.H.: Development and evaluation of a physically-based lake level model for water resource management: A case study for lake buchanan, texas. Journal of Hydrology: Regional Studies **4**, 661–674 (2015)
15. Lu, S., Ouyang, N., Wu, B., Wei, Y., Tesemma, Z.: Lake water volume calculation with time series remote-sensing images. International journal of remote sensing **34**(22), 7962–7973 (2013)
16. McKinney, W.: Data structures for statistical computing in python **445**, 51–56 (2010)
17. NCEP: Statistical downscaling model, 2018, [online], [cited september 25th, 2018] (2017), https://sdsm.org.uk/Definitions.pdf
18. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al.: Scikit-learn: Machine learning in python. Journal of machine learning research **12**(Oct), 2825–2830 (2011)
19. Rossum, V G & Drake Jr, F.L.: Python tutorial, vol. 3-5-1. Centrum Voor Wiskunde en Informatica Amsterdam, Netherlands (1995)
20. Sawunyama, T., Senzanje, A., Mhizha, A.: Estimation of small reservoir storage capacities in limpopo river basin using geographical information systems (gis) and remotely sensed surface areas: Case of mzingwane catchment. Physics and Chemistry of the Earth, Parts A/B/C **31**(15), 935–943 (2006)
21. Solomatine, D., Rojas, C., Velickov, S., Wust, H.: Chaos theory in predicting surge water levels in the north sea. In: Proc. 4th Int. Conference on Hydroinformatics, Cedar-Rapids (2000)

22. Van Der Walt, S., Colbert, S.C., Varoquaux, G.: The numpy array: a structure for efficient numerical computation. Computing in Science & Engineering **13**(2), 22 (2011)
23. Vittori, G.: On the chaotic structure of tide elevation in the lagoon of venice. In: Coastal Engineering 1992, pp. 1826–1839 (1993)
24. Wilby, R.L.: Statistical downscaling model, 2018, [online], [cited December 04th, 2018] (2018), https://sdsm.org.uk/data.html