Acea Smart Water Analytics

ECE 9309B/9039B Machine Learning: From Theory to Applications Final Project Report

Xinze Li
Department of ECE
Western University
London, Canada
xli2966@uwo.ca

Peizhi Yu
Department of ECE
Western University
London, Canada
pyu83@uwo.ca

Qiaomei Han
Department of ECE
Western University
London, Canada
qhan42@uwo.ca

***Abstract*—Water is a precious natural resource as it is limited and essential for a vast range of species on earth. With the goal of helping the Acea group preserve water, this project predicts water availability using machine learning and deep learning methods. Bilancino Lake and Arno River datasets are selected for the water analysis. After preprocessing, the time-dependent data have been obtained. The main models, Multi-layer Perceptron (MLP) neural network, Support Vector Regression (SVR) and Long and Short-Term Memory (LSTM) models are applied. Then the hyperparameter tuning processes are operated for the better model. Experiments have been implemented on these tuned models, where the lake level, flow rate and hydrometry features are selected to predict. Results show that the LSTM model performs better than the others, while still consists some challenges for the future.**

***Index Terms*—Water Analytics, MLP, LSTM, SVR, Prediction**

I. Introduction

Due to economic and technological development, the population soared in recent decades. Along with the increasing population, is the increasing demand for water. At the same time, some areas have suffered from floods and droughts, resulting in poor water resources management. On the other hand, due to the impact of climate change on hydrological processes such as precipitation, evaporation, humidity, *etc*., it has had a huge impact on the water system and caused huge changes in water resources [1]. The demand for water, the growing climate and hydrological gaps have jointly pushed decision-makers and managers of water resources to find strategies for effective water management [2].

This project aims to help Acea Group preserve precious waterbodies. As it is easy to imagine, a water supply company struggles with the need to forecast the water level in a waterbody (water spring, lake, river, or aquifer) to handle daily consumption. It is critical to understand total availability in order to preserve water. During fall and winter waterbodies are refilled, but during spring and summer they start to drain. To help preserve the health of these waterbodies, it is important to predict water availability, in terms of the depth to groundwater, flow rate, hydrometry, and *etc*. Features vary based on different types of waterbodies. For this project, the hydrometry of the Arno River as well as the lake level and flow rate of the Bilancino Lake are predicted using MLP, SVR and LSTM.

II. Related Work

For preserving water resources, this project aims to analyze the the availability of the waterbodies by forecasting the water level in a waterbody (water spring, lake, river, or aquifer) to handle daily consumption. As the availability is related to some time-dependent indexes, we will focus on the time-series analysis algorithms.

A. Multi-layer Perceptron (MLP) neural network

Multi-layer Perceptron neural network is artificial neural network. It is consists of multiple layers of neurons or perceptrons. The first mathematical model for neural network was introduced in 1943 by Warren McCulloch and Walter Pitts [3]. Later, the perceptron algorithm was created in 1958 at the Cornell Aeronautical Laboratory by Frank Rosenblatt [4].

B. Support Vector Machine(SVM)

The Support Vector Machine was invented in 1963 by Vladimir and Alexey. And the kernel trick of the maximum-margin hyperplanes was created in 1992 [5].

The kernel is the most important part for the Support Vector Machine that support SVM and SVR could classified and regress non-linear data. For some data samples which cannot be classified in finite-dimensional vector space, the kernel could map them to high-dimensional feature space to classification or regression. In other words, The non-linear SVM is the combination of kernel and linear SVM. However, the high-dimensional feature space may infinite-dimensional cause that the calculation of $\phi(x_\mathrm{i})\ \phi(x_\mathrm{j})$ is difficulty. Therefore, The new kernel function $k(x_\mathrm{i}, x_\mathrm{j}) = \phi(x_\mathrm{i})\ \phi(x_\mathrm{j})$ will be set for avoid the scalar calculations.

C. Long and Short-Term Memory (LSTM)

The LSTM network is proposed by Hochreiter and Schmidhuber [6], where the storage blocks replace the traditionally hidden layer neurons. This block prevents the disappearance and explosion of gradients during long-term training. Therefore, LSTM enables to process long-term sequences. Many LSTM-based deep learning models have been successfully applied to predict time-series data. However, achieving reliable and accurate prediction in complex, nonlinear, and time-varying conditions remains a challenge.

III. Dataset

The dataset is on Acea Smart Water Analytics retrieved from Kaggle, https://www.kaggle.com/c/acea-water-prediction/data [7]. There are nine different sub-datasets, completely independent and not linked to each other. Each sub-dataset can represent a different kind of waterbody. As each waterbody is different from the other, the related features as well are different from each other. To simplify, we will consider the information of two sub-datasets and make predictions based on their different features, which is the Arno River and the Bilancino lake.

The availability of Arno River is evaluated by checking the hydrometric level of the river at the section of Nave di Rosano. In this sub-dataset, there are 16 features, which are the rainfalls in 14 locations, the temperature and the hydrometry. Bilancino lake is an artificial lake. During the winter months, the lake is filled up and then, during the summer months, the water of the lake is poured into the another waterbodes. In this sub-dataset, there are 8 features, which are the rainfalls in 6 locations, the lake level and the flow rate.

Therefore, the hydrometry of the Arno River as well as the lake level and flow rate of the Bilancino Lake will be predicted using Multilayer Perceptron neural network, Long short-term Memory, and SVR in this project. The detailed descriptions will be explained in the following sections.

As noticed, some features like rainfall and temperature, which are present in each dataset, don't go alongside the date. Indeed, both rainfall and temperature affect features like level, flow and hydrometry some time after it fell down. As we don't know how many days/weeks/months later rainfall affects these features, this is another aspect to keep into consideration when analyzing the dataset.

IV. Methodology

A. Exploratory Data Analysis

There are a total of 6603 entires and 9 columns in the Lake Bilancino dataset, among which 6 columns are features and 2 columns are targets. For the River Arno dataset, there are 8217 entires and 17 columns in total. 15 columns are features and the last column is the target. Both dataset contain rows with NaN values, and this will be handled in the next step. The following plots display the analysis results on Lake Bilancino and River Arno.

B. Data Preprocessing

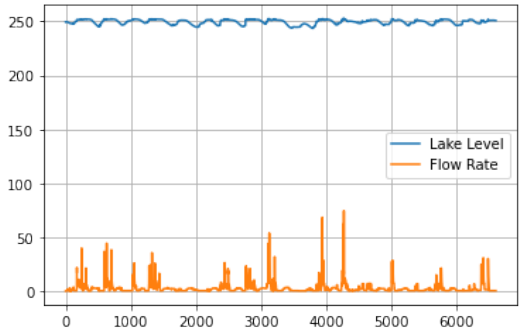
In this project, NaN values are handled by deleting the entire row, and the data remains in chronological order.

Next, we split the data into training and testing sets.

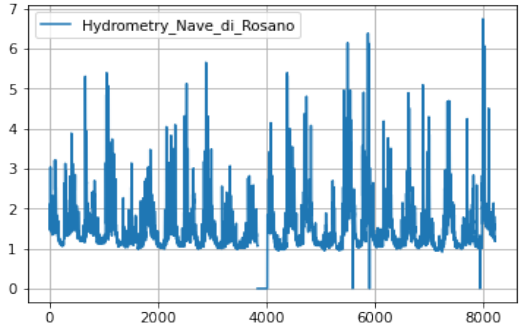
Furthermore, normalization has been performed on the training and the dataset, respectively, where every feature is normalized between 0 and 1.

C. Models

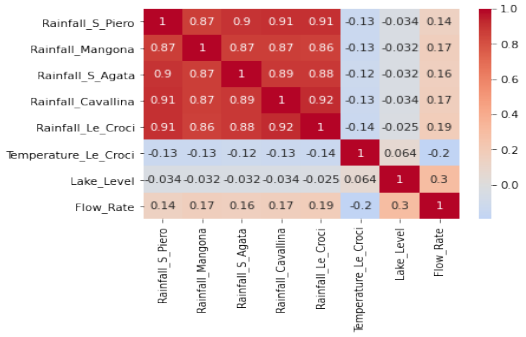
1) Multi-layer Perceptron (MLP) neural network: Multi-layer Perceptron neural network belongs to feedforward neural



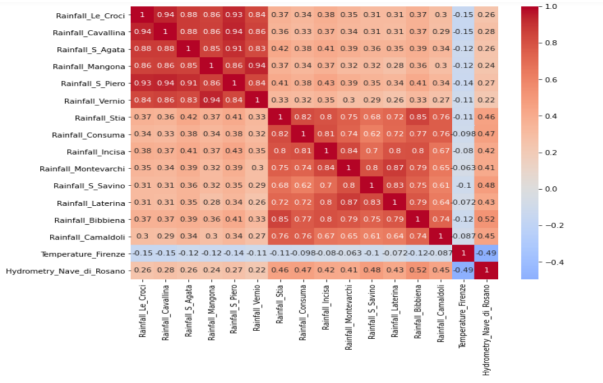
(a) Lake Level and Flow Rate (Lake Bilancino)



(b) Hydrometry (River Arno)



(c) Correlation Heatmap (Lake Bilancino)



(d) Correlation Heatmap (River Arno)

Fig. 1. EDA on Bilancino Lake and River Arno data

networks. MLP consists of multiple layers of neurons including an input layer, one or more hidden layers and an output layer. The weighted sum of neurons in each layer is past to the neurons in the next layer. Then it goes through their corresponding activation functions, which map the input to the output of this neuron. The architecture and settings for the MLP used is as displayed in Table I and Table II.

TABLE I
ARCHITECTURE OF MLP NEURON NETWORK (LAKE BILANCINO)

Layers	Dense	neuron = 16	activation = ReLu
	Dense	neuron = 16	activation = ReLu
	Dense	neuron = 16	activation = ReLu
	Output Dense	neuron = 2	activation = Linear
Loss function		MSE	
Batch size		16	
Epoch		100	
Optimizer		Adam	
Early Stopping		5	

TABLE II
ARCHITECTURE OF MLP NEURON NETWORK (RIVER ARNO)

Layers	Dense	neuron = 8	activation = ReLu
	Output Dense	neuron = 1	activation = Linear
Loss function		MSE	
Batch size		16	
Epoch		100	
Optimizer		Adam	
Early Stopping		5	

2) Support Vector Regression(SVR): A SVR model has a configuration that is somewhat similar to a three-layer ANN with a hidden layer with the same number and feature as the support vectors. To put it another way, the number of hidden nodes is the same as the number of support vectors. It's important to remember that in SVR, the whole model architecture is adaptively generated directly, while in ANN, only the weights are automatically generated. That is, while the adaptive Support Vector Regression(SVR) algorithm determines the number of support vectors, the number of hidden layers and hidden nodes for each layer must be artificially estimated in advance for ANN [8].

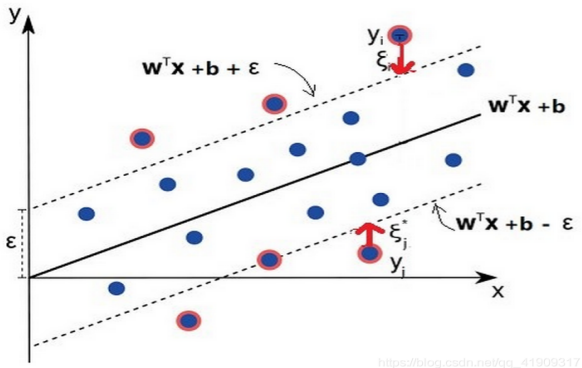


Fig. 2. The example of Support Vector Regression [9]

Compare with linear regression, the Support Vector Regression is a Slack regression model. Linear regression using linear function to fit the sample in the vector space. The model takes the comprehensive distance from the actual positions of all samples to the linear function as the loss and obtains the parameters of the linear function by minimizing the loss. For linear regression, as long as a sample does not fall exactly on the linear function as the model, the loss will be calculated.Oppositely, although Support Vector Machine also using the linear function as the model functions, the principle of the loss calculations are different. In addition, the objective function and optimization algorithm are also different.

SVR creates an "interval band" on both sides of the linear function. For all samples that fall into the interval band that will not calculated the loss. Oppositely, the samples outside the interval band will included in the loss function. Then optimize the model through the minimizing width of the interval band and the total loss. According to Fig. 2, the red circled samples will be calculate the loss. Therefore, the function of the model is a linear function

$$f(x) = wx + b \tag{1}$$

Then, the border line of the interval band is that

$$wx + b + \epsilon wx + b - \epsilon \tag{2}$$

And the slack variable E is the difference between the projection of the samples around the border line and the samples. The equations of the slack variable are

$$\begin{cases} E_\mathrm{i} = y_\mathrm{i} - (f(x_\mathrm{i}) + \epsilon), & if\ y_\mathrm{i} > f(x_\mathrm{i}) + \epsilon \\ E_\mathrm{i} = 0, & Otherwise \end{cases} \tag{3}$$

$$\begin{cases} E_\mathrm{i}^* = (f(x_\mathrm{i}) - \epsilon) - y_\mathrm{i}, & if\ y_\mathrm{i} < f(x_\mathrm{i}) - \epsilon \\ E_\mathrm{i}^* = 0, & Otherwise \end{cases} \tag{4}$$

For any sample x_i, the E_i and E_i^* is 0 when the sample is into or onto the border line of the interval band. Otherwise, the $E_\mathrm{i} > 0$ and $E_\mathrm{i}^* = 0$ when the sample above the interval band. Oppositely, the $E_\mathrm{i} = 0$ and $E_\mathrm{i}^* > 0$ when the sample below the interval band [9].

The kernel is the most important for Support Vector Regression that will map the input x to a high-dimensional feature space through the mapping function $\phi(x)$, then do the regression in the feature space.

3) Long and Short-Term Memory (LSTM): In the LSTM network structure, the LSTM memory unit is located in the center, the input is known data, and the output is the predicted result. There are three gates in the memory unit, which can selectively add and filter the information passing through the structure. The three types of "gates" include: forget gates, input gates and output gates. The forget gate is used to control the historical information stored in the hidden layer node and at the previous moment. The input gate is used to control the input of the hidden layer node at the current moment. The output gate is used to control the output of the hidden layer node at the current moment.

In the LSTM network, the input values need to be reshaped into 3-D as samples. As described above, we need to split the training and testing sets. We chose the 0.2 as the test size for the sub-dataset Bilancine Lake and the Arno River for testing, and the left samplings for training. The input values have been reshaped into 3-D as samples, where the time step is set 10 and the input dim is set 8 and 16, respectively. In this way, the preprocessed data can be fed to LSTM to make predictions through historical and current data.

According to the information flow in the memory unit structure, the status update and output of the memory unit are:

$$i_{\mathrm{t}} = \sigma\left(W^{(i)}X_{\mathrm{t}} + U^{(i)}S_{\mathrm{t}-1}\right) \tag{5}$$

$$f_{\mathrm{t}} = \sigma\left(W^{(f)}X_{\mathrm{t}} + U^{(f)}S_{\mathrm{t}-1}\right) \tag{6}$$

$$o_{\mathrm{t}} = \sigma\left(W^{(o)}X_{\mathrm{t}} + U^{(o)}S_{\mathrm{t}-1}\right) \tag{7}$$

$$\tilde{S}_t = \tanh\left(W^{(\mathrm{c})}X_{\mathrm{t}} + U^{(c)}S_{\mathrm{t}-1}\right) \tag{8}$$

$$S_{\mathrm{t}} = f_{\mathrm{t}} \circ S_{\mathrm{t}-1} + i_{\mathrm{t}} \circ \tilde{S}_t \tag{9}$$

$$O_{\mathrm{t}} = o_{\mathrm{t}} \circ \tanh\left(S_{\mathrm{t}}\right) \tag{10}$$

where 'o' represents the Hadamard product of the matrix; i_{t},f_{t} and o_{t} are the outputs of the input gate, forget gate and output gate, respectively; $\tilde{S}_t$ is the new state of the memory unit, while S_{t} is the final state of the memory unit; O_{t} is the final output of the memory unit; $W^{(i)}$, $W^{(f)}$, $W^{(o)}$ and $W^{(\mathrm{c})}$ respectively represent the input gate, forget gate, output gate and the weight matrices of the previous state; $U^{(i)}$, $U^{(f)}$, $U^{(o)}$ and $U^{(c)}$ mean the input gate, forget gate, output gate and the deviation matrices of the previous state. Besides, $\sigma\left(\right)$ represents the sigmoid activation function, which is $\sigma\left(x\right) = \frac{1}{1+e^{-x}}$; and $\tanh\left(\right)$ indicates the tanh activation function, $\tanh\left(x\right) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$.

The network architecture for these two datasets are shown in Table IV and Table V, some parameters are tuned by the hyperparameter-tuning processes, which will be discussed the next sub-section.

D. Hyperparameter Tuning

1) MLP: The hyperparameters were tuned using GridSearchCV in sklearn. Choices for optimizer are ['adam', 'SGD', 'rmsprop']. Batch size includes [64, 128], and the number of epochs include [200, 300, 400].

2) SVR: For hyperraramter tuning of the Support Vector Machine in sklearn. The tuned parameters of Support Vector Regression(SVR) has shown in Table III.

Each parameter has a different effect on the model. The main tuned parameters are penalty, kernel, degree, and gamma. Parameter C is the Penalty of slack variable, which larger C causes the penalty for misclassification increase that means the accuracy of the training set will be higher. The Kernel is the most important part for Support Vector Regression. As mentioned in section IV, the kernel will map the input variables to the feature space that will help the classification and regression for the support vector machine. The degree related to the dimension. The gamma is the parameter for rbf kernel that means the influence of a single training example reaches, low value means far and high value means close, usually be 1/n'features. In addition, the parameter Epsilon is related to the desired accuracy of the model. And Scale and Shrinking are about the hit rate of the kernel cache. Then, Cache size is about the running time of the program.

TABLE III
ARCHITECTURE OF SVR FOR THE DATASET

Name	Parameter
C	2.25
Kernel	rbf
Degree	3
Gamma	0.1
Epsilon	0.1
Shrinking	True
Cache Size	200
Verbose	False

TABLE IV
ARCHITECTURE OF FOR THE ARNO RIVER DATASET

Layers	LSTM	neuron = 128	activation = ReLu
	Dense	neuron = 100	activation = ReLu
	Dense	neuron = 50	activation = ReLu
	Dense	neuron = 30	activation = ReLu
Loss function	MSE		
Batch size	128		
Time step	10		
Epoch	50		
Optimizer	Adam		

3) LSTM: We need to tune hyper-parameters for the LSTM model through the grid research approach. The epoch number is tuned from [50, 100, 200], Batch size is from [64, 128], and Neurons in the first layer is from [32, 64, 128]. The activation function is from ['relu','sigmod','tanh'], and the optimizer is from ['Adam','SGD','RMSprop'].

The tuned structure of the LSTM have already been shown in Table IV and Table V.

V. RESULTS AND ANALYSIS

A. Metrics

After providing the appropriate methods, it is also critical to evaluate this kind of time-series problem. Therefore, accuracy measurements should be discussed. As our targeting problem can be classified into the regression field, the involved values include the predicted values and the ground truth, which can be represented by $\hat{\varphi}_i$ and φ_i for clarify.

TABLE V
ARCHITECTURE OF FOR THE BILANCINO LAKE DATASET

Layers	LSTM	neuron = 32	activation = ReLu
	Dense	neuron = 30	activation = ReLu
Loss function	MSE		
Batch size	64		
Time step	10		
Epoch	50		
Optimizer	RMSprop		

The Mean absolute error (MAE) and the root mean square error (RMSE) can be used to evaluate the performance of the forecasting models. The definition are given by:

$$\mathrm{MAE} = |\hat{\varphi}_\mathrm{i} - \varphi_i| \tag{11}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{\mathrm{n}} \sum_{i=1}^{n} (\hat{\varphi}_i - \varphi_i)^2} \tag{12}$$

Where $\hat{\varphi}_i$ is the forecast data, while φ_i is the ground truth.

B. Benchmarking

The MLP, LSTM, and SVR models in this project will be compared with each other, and the optimal one will be selected for the prediction purpose.

This subsection will be supplemented after all the models having been trained.

C. Results

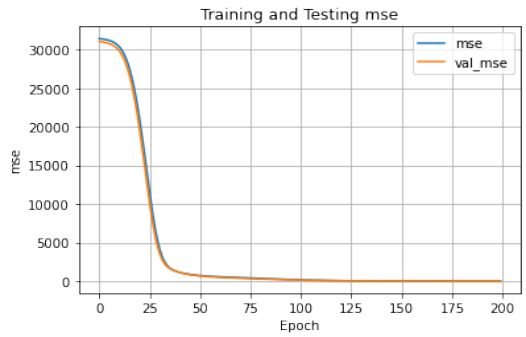
1) MLP: As it is displayed in Fig. 3, the training and testing MSE decreases significantly as the number of epoch increases. The best MSE value for Lake Bilancino is around 13.33 and the best validation MSE is around 12.54. For River Arno, the best MSE is around 0.14 and the best validation MSE is around 0.11. It is worth noting in the lake level and flow rate predictions plot that sudden increases or decreases are generally not predicted accurately. Events causing such abrupt changes can sometimes be totally random and therefore, they can be difficult to be predicted.

2) SVR: Fig. 4 shown the experimental results, the (a) of Fig. 4 is the comparison between the prediction of hydrometry and the true data of hydrometry. Similarly, the (b), (c),(d) of Fig. 4 are the comparison between the prediction of lake level, flow rate and the true data of them. In addition, the mean square error of each data has shown on the program. The mse of hydrometry is around 0.11. Moreover, the mse of lake level and flow rate are around 4.41 and 19.62. The reason for the mean square error has a huge difference is that the data of each feature has a quantitative difference. For example, after dropping the NAN data of hydrometry, there are more than 4500 out of 7000 have been dropped.

3) LSTM: Fig. 5 shows the virtualization of the experimental results, indicating the loss function values based on the training and the testing data during the training process. The X-axis is the epoch number, and the Y-axis shows the loss function values. Sometimes, the loss function of training data continuously decreases; however, the testing data does not. Table VI indicates the best MAE of the LSTM model, which are selected for the predictions.

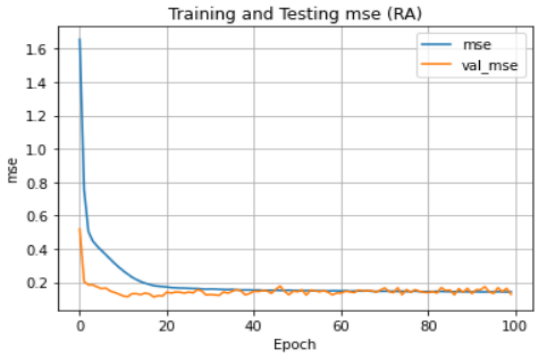
To avoid overfitting, the early stopping mechanism is adopted, where the patience is set as 10. As the epoch number increases, if the test error is found to rise on the validation set in the continuous 10 epochs, the training has to stop. Then the values after stopping can be used as the final parameters of the network.

As known, there are three types of layers in the LSTM network, including the input layer, the hidden layer, and the

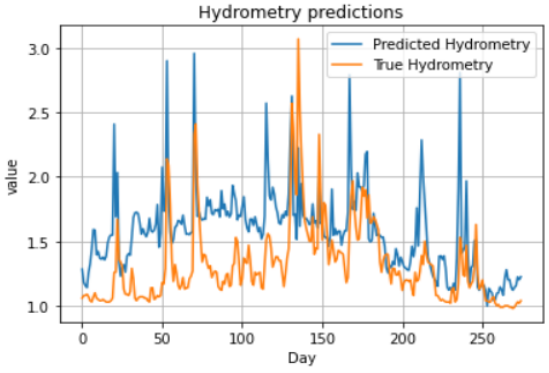


(a) MLP Training and Testing MSE (Lake Bilancino)

(b) Lake Level and Flow Rate predictions (Lake Bilancino)

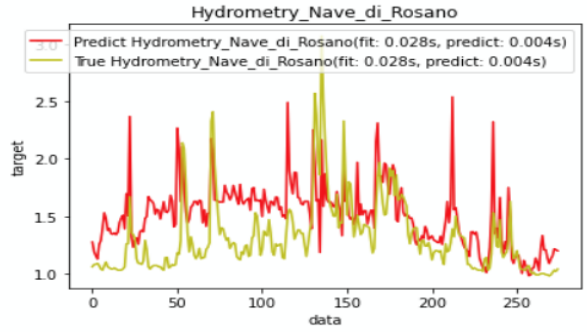


(c) MLP Training and Testing MSE (River Arno)



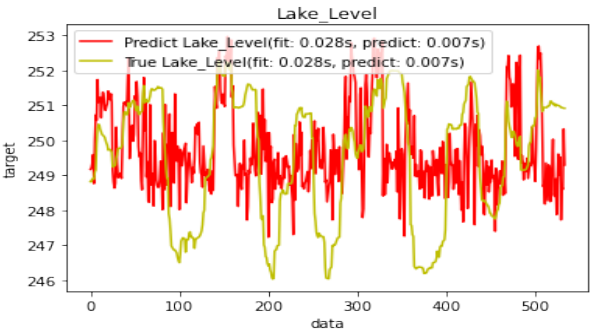
(d) Hydrometry prediction (River Arno)

Fig. 3. MLP results on Lake Bilancino and River Arno

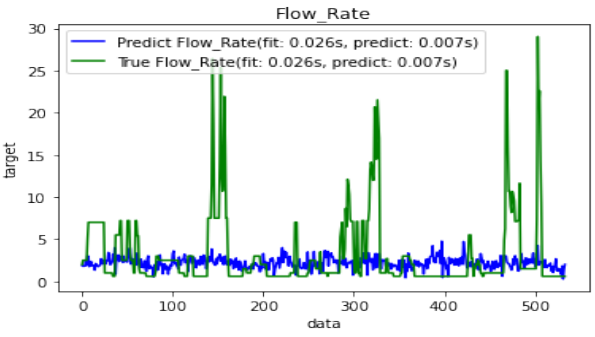


(MSE): 0.11191260065636119

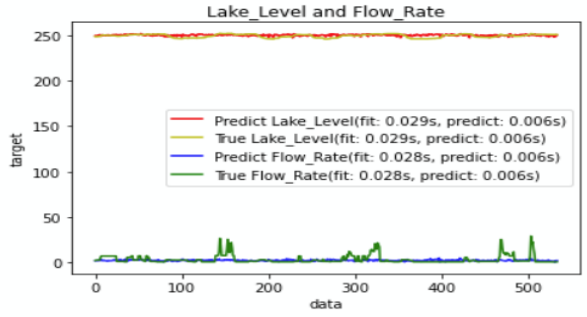
(a) hydrometry of the Arno River



(b) lake level of the Bilancino Lake



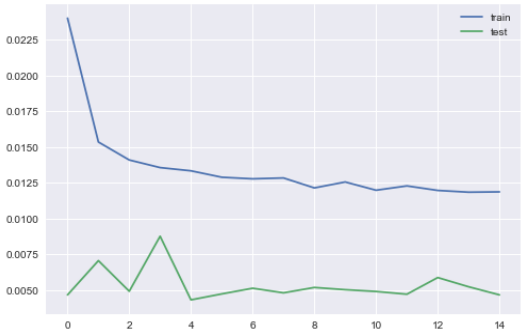
(c) flow rate of the Bilancino Lake



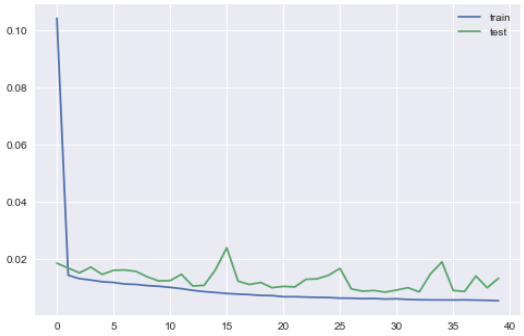
Lake_Level(MSE): 4.413611843134026
Flow_Rate(MSE): 19.625813052110637

(d) Overview of the Bilancino Lake

Fig. 4. Loss function values based on the training and the testing data.



(a) hydrometry of the Arno River



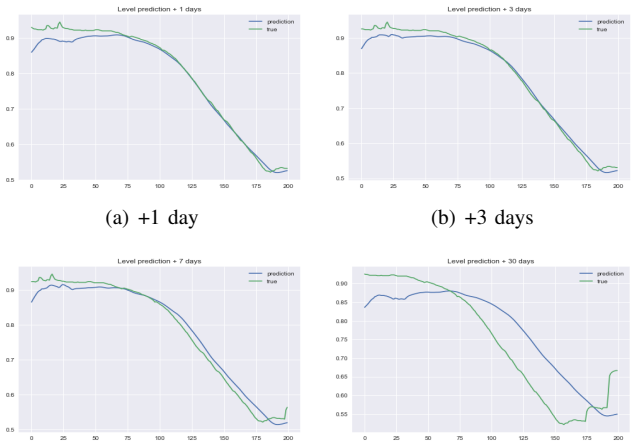
(b) lake level of the Bilancino Lake

(c) flow rate of the Bilancino Lake

Fig. 5. Loss function values based on the training and the testing data.

output layer. The input to the LSTM network is the historical data. The output is the predicted data for a period of time.The number of memory cells is determined by the time step. The hidden layer number in Table IV and Table V is one, but it can be easily extended to several layers according to the situation.

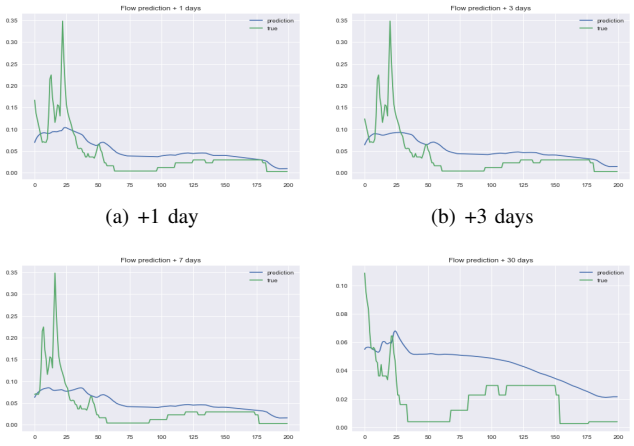
Fig. 6, Fig. 7 and Fig. 8 show the lake level, flow rate and hydrometry prediction results from taspects of in 1, 3, 7 and 30 days. The blue lines show the predicted data, while the green lines show the ground truth. The top eight graphs are for the Bilancino Lake dataset, and the bottom four are the prediction for the Arno River dataset. Table VI also reveals the RMSE value of the predicted results. It can be observed that the short-term prediction commonly performs better than the long-term results after applying the LSTM models.



(a) +1 day (b) +3 days

(c) +7 days (d) +30 days

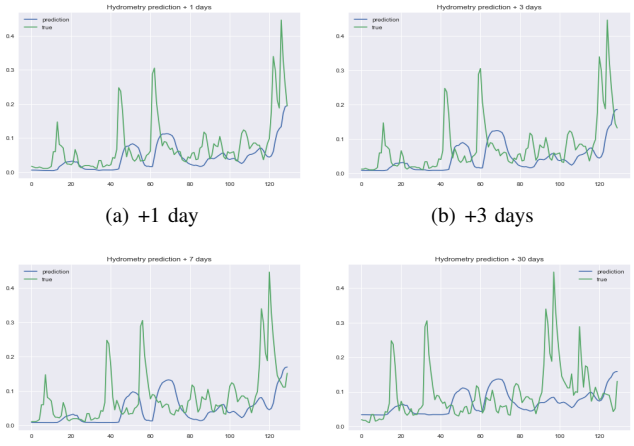
Fig. 6. Lake level of the Bilancino Lake prediction.



(a) +1 day (b) +3 days

(c) +7 days (d) +30 days

Fig. 7. Flow rate of the Bilancino Lake prediction.



(a) +1 day (b) +3 days

(c) +7 days (d) +30 days

Fig. 8. Hydrometry of the Arno River prediction.

TABLE VI
EVALUATION RESULTS THROUGH THE LSTM MODELS

	Best MAE	Prediction time interval	RMSE
Hydrometry of Arno River	0.0520	1 day	0.0616
		3 days	0.0689
		7 days	0.0758
		30 days	0.0728
Lake level of Bilancino Lake	0.1498	1 day	0.0415
		3 days	0.0508
		7 days	0.0795
		30 days	0.1609
Flow rate of Bilancino Lake	0.1524	1 day	0.0200
		3 days	0.0313
		7 days	0.0383
		30 days	0.0455

VI. CONCLUSION

After comparisons, LSTM model preforms better than the MLP and SVR models, from the MSE acpect. Moreover, the LSTM model is more suitable for the short-term prediction than the long-term. However, there are also problems in this project: On the one hand, there exists large number of null values in the datasets, which need proper methods for progressing. On the other hand, sudden spikes can be hard to predict. Therefore, these problems can be developed to solve.

REFERENCES

[1] J. L. Schnase, D. Q. Duffy, G. S. Tamkin, D. Nadeau, J. H. Thompson, C. M. Grieg, M. A. McInerney, and W. P. Webster, "Merra analytic services: Meeting the big data challenges of climate science through cloud-enabled climate analytics-as-a-service," *Computers, Environment and Urban Systems*, vol. 61, pp. 198–211, 2017.

[2] R. Chalh, Z. Bakkoury, D. Ouazar, and M. D. Hasnaoui, "Big data open platform for water resources management," in *2015 International Conference on Cloud Technologies and Applications (CloudTech)*, 2015, pp. 1–8.

[3] W. P. Warren S. McCulloch, "A logical calculus of the ideas immanent in nervous activity," *The bulletin of mathematical biophysics*, p. 115–133, 1943.

[4] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain," *Psychological Review*, pp. 65–386, 1958.

[5] J. L. Schnase, D. Q. Duffy, G. S. Tamkin, D. Nadeau, J. H. Thompson, C. M. Grieg, M. A. McInerney, and W. P. Webster, "A training algorithm for optimal margin classifiers," *Proceedings of the fifth annual workshop on Computational learning theory*, vol. 92, p. 144, 1992.

[6] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[7] Kaggle, "Acea smart water analytics [online]," *Available: https://www.kaggle.com/c/acea-water-prediction/data*, [Accessed: 22-Mar-2021].

[8] G. I. Ruas, T. A. Bragatto, M. V. Lamar, A. R. Aoki, and S. M. de Rocco, "Electrical energy demand prediction using artificial neural networks and support vector regression," in *2008 3rd International Symposium on Communications, Control and Signal Processing*. IEEE, 2008, pp. 1431–1435.

[9] WNotSyer, "Svr model notebook [online]," *Available: https://blog.csdn.net/qq*$_4$1909317/*article*/*details*/88542892, [*Accessed* : 17 − *Apr* − 2021].

VII. APPENDIX

All three members contributed in the process of this project. Each member is responsible for a model. Exploratory data analysis and data preprocessing are the results of group discussion. Xinze Li worked on building and tuning MLP neural

networks, Peizhi Yu worked on SVR model, and Qiaomei Han worked on LSTM model.

About the codes, the url of the repository is https://bitbucket.org/YoMikey/finalprojectcode/src/master/.