Transactions on Machine Learning and Artificial Intelligence - Vol. 10, No. 5
Publication Date: October, 25, 2022
DOI:10.14738/tmlai.105.13049.
Karimanzira, D., Ritzau, L., & Emde, K. (2022) Catchment Area Multi-Streamflow Multiple Hours Ahead Forecast Based on Deep Learning. Transactions on Machine Learning and Artificial Intelligence, 10(5). 15-29.

Catchment Area Multi-Streamflow Multiple Hours Ahead Forecast Based on Deep Learning

Divas Karimanzira
Fraunhofer Institute of Optronics
System Technologies and Image Exploitation (IOSB)
Am Vogelherd 90, 98693 Ilmenau, Germany

Linda Ritzau
Fraunhofer Institute of Optronics
System Technologies and Image Exploitation (IOSB)
Am Vogelherd 90, 98693 Ilmenau, Germany

Katharina Emde
School 1Fraunhofer Institute of Optronic
System Technologies and Image Exploitation (IOSB)
Am Vogelherd 90, 98693 Karlsruhe, Germany

ABSTRACT

Modeling of rainfall-runoff is very critical for flood prediction studies in decision making for disaster management. Deep learning methods have proven to be very useful in hydrological prediction. To increase their acceptance in the hydrological community, they must be physic-informed and show some interpretability. They are several ways this can be achieved e.g. by learning from a fully-trained hydrological model which assumes the availability of the hydrological model or to use physic-informed data. In this work we developed a Graph Attention Network (GAT) with learnable Adjacency Matrix coupled with a Bi-directional Gated Temporal Convolutional Neural Network (2DGAT-BiLSTM). Physic-informed data with spatial information from Digital Elevation Model and geographical data is used to train it. Besides, precipitation, evapotranspiration and discharge, the model utilizes the catchment area characteristic information, such as instantaneous slope, soil type, drainage area etc. The method is compared to two different current developments in deep learning structures for streamflow prediction, which also utilize all the spatial and temporal information in an integrated way. One, namely Graph Neural Rainfall-Runoff Models (GNRRM) uses timeseries prediction on each node and a Graph Neural Network (GNN) to route the information to the target node and another one called STA-LSTM is based on Spatial and temporal Attention Mechanism and Long Short Term Memory (LSTM) for prediction. The different methods were compared in their performance in predicting the flow at several points of a pilot catchment area. With an average prediction NSE and KGE of 0.995 and 0.981, respectively for 2DGAT-BiLSTM, it could be shown that graph attention mechanism and learning the adjacency matrix for spatial information can boost the model performance and robustness, and bring interpretability and with the inclusion of domain knowledge the acceptance of the models.

Keywords: Deep learning; Streamflow prediction; Graph Attention Neural Networks, Sequential model. Rainfall-Runoff Modelling.

INTRODUCTION

Among all extreme events such as droughts, fires, etc., flood is the most common and cause a lot of deaths according to the UN [1,2]. Therefore, rainfall-runoff studies are very important in several respects. Thy can be used to simulate flow peaking, changes in the water quantity and quality and helps in decision on water resources management. Due to the nature of hydrological processes, interconnectivity between several influencing variables, nonlinearity, etc, rainfall-runoff modelling has many challenges. Traditionally, physically-based or numerical hydrological models are used for flood prediction, e.g. in [3], fully dynamic, diffusive, and kinematic waves models are used to describe the surface water flow and produces very good results in flow and water level prediction. Although the performance of the physically-based models in terms of prediction are excellent, these models require various types of hydrological and geomorphological observations. Their setup and operation are very time-expensive, which makes short-term prediction required, not feasible [4]. Furthermore, developing a physically-based model requires in-depth domain knowledge and expertise on hydrological parameters and model region, which is very challenging. Physically-based models postulates inserting maltitudes of different types of hydrological data for learning, but in fact, it is known that there are too many influencing factors of floods which are difficult to fully simulate them. Therefore, Cea et al,[5] and others, suggest to use as less variables as possible to increase the transferability of the models. For example Cea et al, [5], pointed out the correlation between rainfall and runoff.

In recent years, data-driven and deep learning models are becoming the point of interest in hydrological studies [6]. Mostly, LSTM (Long Short-Term Memory), TCN (Temporal Convolutional Networks) are used as rainfall-runoff modelling is taken as a timeseries problem [7,8,9,10].

If geospatial information such as topography and other watershed characteristics considering the upstream-downstream relationships are integrated into deep learning, a combination of Graph Neural Network (GNN) or Convolutional Neural Networks and LSTMs are used as in [11],[12]. In this case, CNNs are used to extract and encode spatial information at each time step which is then fed into an LSTM module for the time-series modeling and prediction. Due to the nature of hydrological data that it is non-euclidean, using CNNs might not be physically correct. The CNNs have convolution kernels with fixed size filter which assumes Euclidean data and takes all the local neighbors of each node into consideration. In hydrology the fact that streams or hillslopes are regionally close to one another does not mean that they are hydrologically related. A more physically appealing solution is to use GNNs which capture the upstream-downstream dependency graphically [13].

There are two different current developments in deep learning structures for streamflow prediction. One from the authors [14] Graph Neural Rainfall-Runoff Models (GNRRM) assumes raster information and uses timeseries prediction models on each grid cell and Graph Neural Networks to aggregate grid cells with similar distance from the target node to compute its streamflow. Through the GNN it is non-euclidean as it utilizes spatial information. The author showed that for the sequence models in the nodes to model temporal behavior, BiGTCN

performed best. Its drawback is the time consumption required during training for to the vast number of sequence models per grid cell even though they are aggregated. The sequence node models are aggregated based on its distance to the outlet node using d8 flow algorithm. Thereby, a simple hierarchical graph representation of the original di-graph is created based on the direct distance. And the node aggregation operations can be the sum for the area, and the weighted average for precipitation intensity.

Another method from [15] is based on Spatial and Temporal Attention Mechanism and LSTM (STA-LSTM) for streamflow prediction. Even though it uses attention mechanism to represent the spatial and temporal relationships, it looks only on the individual target node and use rainfall only as a feature. the attention mechanism is inspired by the human selective visual attention mechanism. It uses limit attention to choose more critical information from numerous input features [16].

To overcome some of the drawbacks of these two current methods, we introduce a model based on 2D Graph Attention Networks (2DGAT) and a BiLSTM. Adjacency matrix as in [15] is created based on the direct distance between stations is not enough for representing the spatial relationships between the nodes. There are too many unknown influences in hydrology. Therefore, the GAT architecture is extended by equipping it with a learnable adjacency matrix as well as incorporating a new attention mechanism with the aim of obtaining attention scores per co-variable. The output of the GAT based model is combined with the bidirectional LSTM (BiLSTM) layer to enable it to exploit both the spatial and temporal characteristics of the multivariate multidimensional meteorological and streamflow data. Through the 2D GAT, spatial information from DEM such as slope, time of concentration and geographic data such as soil type, landcover etc. can be utilized. Furthermore, the obtained attention scores per feature increase the interpretability of the model, which is difficult in deep learning. The three models will be compared based on the Nash Sutcliffe Efficiency (NSE) [17] and Kling-Gupta Efficiency (KGE,)[18] for their prediction performance for streamflow values in multiple flow stations for several hours ahead ranging from 1 up to 6 hours.. Data from a pilot region in Germany will be used for these studies. The region had some extreme flood events recently.

The rest of the paper is organized as follows. Our method and the other deep learning methods used for comparison will be described in Section 2. Section 3 will give an overview of the numerical experiments conducted. The results will be discussed in Section 4 and the paper will be closed with some conclusions and future work.

METHOD

To understand our approach and motivation which will be described later in this section we first look at a catchment area and its attributes. Figure 1 shows an overview of an example catchment area with rainfall and flow gauging stations. It is apparent that every area covered by a flow station has its own topological and geographical conditions.

URL: http://dx.doi.org/10.14738/tmlai.105.13049

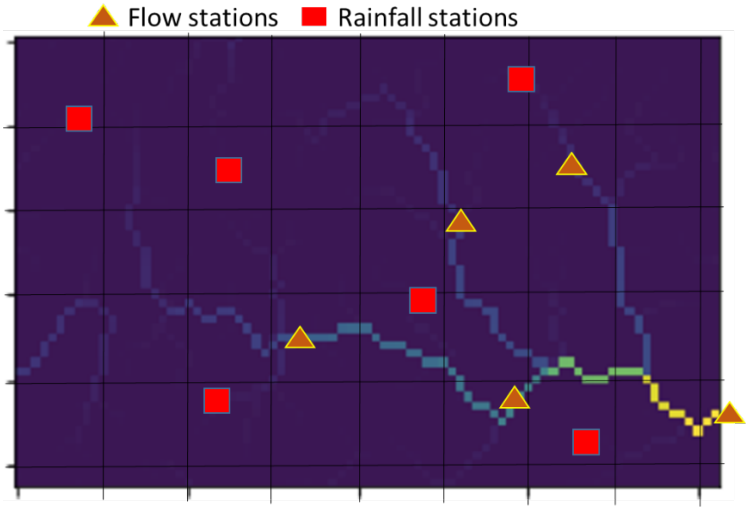


Figure 1: Overview of an example catchment area with rainfall and flow stations

Precipitation is the main driver of flooding and this in different areas in the upstream will affects lower areas in the downstream hydrological stations according to the river network and delay caused by the topological and geographical conditions. How the rivers converge and the interlinks or relationships between the hydrological stations, the precipitation in various areas resembles a complex graphical structure, which can easily be analyzed using Graphical Neural Network (GNN). Node adjacency is complex in such a system, because direct distance or even hydraulic distance is not enough to describe it. There are so many other influences to describe the adjacency.

Firstly, the method requires physic-inspired data. Therefore, it is divided into two parts. The first part deals with the representation of the catchment area as a graph to include all relevant meteorological, hydrologic and geographic features in the nodes and the second part focuses on the deep learning architecture for streamflow prediction based on the graph, dynamic, and geographic information.

Graph representation of a catchment area and its attributes

Information from a digital elevation map (DEM) such flow direction and rainfall data of a catchment area can be used to construct a dependency graph for the catchment area. If meteorological data is available in form of grid cells e.g. from radar, these grid cell can be taken as graph nodes aggregated to the resolution of the DEM. If rainfall and discharge/water level are measured by gauges the position of the gauges can be considered as the graph nodes. Next, the adjacency of the nodes (i.e., graph edges) is determined by using d4 or d8 algorithms to get physically meaningful neighborhood. d4 is a kernel method that routes flow to one of the four cardinal directions based on the steepest descent, and d8 considers eight surrounding directions [19]. At the end, and besides dynamic variables such as rainfall, evapotranspiration, node characteristic information such as drainage area, land use, soil type can be used as the node features. Xiang et al, [8] showed that these characteristic features are effective in deep learning based modeling to represent and induce different behavior in different watersheds in regional models. If data is available on one or more of these features it is recommended to use them in the graph networks.

Deep learning architecture

In the two baseline methods which will be described later, it is assumed that the adjacency matrix is known from the direct distances between the stations, but this is just an approximation because the adjacency is influenced by several other factors which are unknown. Therefore, it is necessary to design a graph structure with a learnable adjacency matrix to overcome this drawback and to show the importance of the different variables in the system in a transparent manner. We propose an architecture as shown in Figure 2. It is composed of a of Graph Attention Networks (GAT) layer for spatial encoding leading to a BiLSTM layer which captures the temporal dependencies of the meteological, hydrological and geographical variables at different timesteps. Unlike the classical GAT with each node having a one dimensional feature vector as described in [20,25], the GAT incorporate 2-Dimensional node features. This has the aim to learn the spatial dependencies of various nodes of a digraph and how the features of a node may affect features of other nodes. In addition, the GAT layer has a learnable adjacency matrix which helps the system to learn the connection between hydrological stations in an end-to-end fashion.

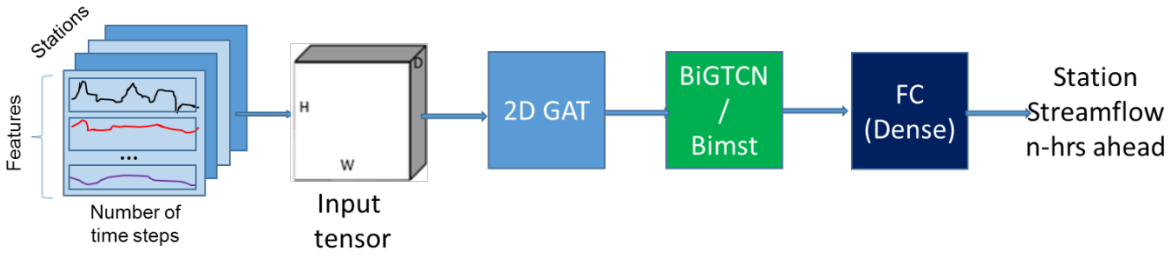


Figure 2: Overview of the proposed GAT-BiGTCN model.

For a GAT with N number of nodes, it is assumed that the set of node features $h = \{h_1, h_2, \ldots, h_N\}$, with $h_i \in R^d$, where d is the number of features in each node. A shared weight matrix is first used to linearly transform the feature values of each node to a selected number of features $\acute{d}$. Following this operation, attention scores of every node with respect to the others are calculated using a self-attention mechanism. Mathematically, the attention score α_i for a given pair of nodes i and j is calculated using equation 1.

$$\alpha_{i,j} = \frac{exp\left(LeakyReLU\left(a^T[Wh_i||Wh_j]\right)\right)}{\sum_{k\epsilon N} exp\left(LeakyReLU\left(a^T[Wh_i||Wh_j]\right)\right)} \quad (1)$$

$W \in R^{d' \times d}$ denotes the shared weight matrix which linearly transforms features of each node and the self-attention mechanism is modeled by a single layer feed-forward neural network parameterized by $a \in R^{2d'}$. To account for the nonlinearity, a LeakyRELU is applied with negative slope followed by a softmax to normalize the attention score. In the classical GAT as described in [20], the softmax is applied with respect to all nodes within the neighborhood of the j-th node which is pre-determined by the adjacency matrix of the graph. However, in our case, adjacency matrix is learned during the training phase without imposing any constraint on the connectivity between nodes of the graph. Therefore, all nodes are connected and the strength between the nodes are learnt through the elements of the adjacency matrix. The linearly transformed features of nodes are multiplied with the attention scores and then the sum is taken with respect to neighboring vertices.

URL: http://dx.doi.org/10.14738/tmlai.105.13049

In 2D GAT which operates on 2-dimensional features, the attention scores are calculated for every meteorological or hydrological variable. So the representation for a node in this case will be as follows:

$$\hat{h}_i = \sigma\left(\sum_{j=N} W h_j diag\left(\alpha_{ij}^1, \dots, \alpha_{ij}^d\right)\right) \quad (2)$$

where, α_{ij}^p are the attention scores between i^{th} and j^{th} nodes for the p^{th} meteorological or hydrological variable. K-independent attention mechanisms are applied and their results are concatenated to obtain the final representation of the nodes.

As described the 2D GAT extracts spatial features from the input variables. To be able to further capture the temporal features dependency and how different feature influence each other at different timesteps, the 2D GAT is coupled with a BiLSTM timeseries model. Therefore, the output of the GAT system is reshaped and fed into the BiLSTM. For the regression part a Fully Connected Network is added at the output for final prediction of the several hours ahead. Any state-of-the art sequence model like LSTM will do the job [21,22], but we chose the BiLSTM because according to the studies in Zhongrun und Demir, the BiLSTM and from our experience performs best in terms of quality and time in several catchments for streamflow prediction.

Baseline methods

Graph Neural Rainfall-Runoff Models (GNRRM-TS)

The first baseline method is based on the model GNRRM-TS in [14] (Figure 3). It considers the catchment area as a k-hop graph where the k-value is determined by the maximum distance to the outlet in the catchment area. The different nodes are aggregated according to their hops (i.e. the distance from the outlet) in the graph using the d4 or d8 algorithm. Every feature is aggregated using appropriate operation, for example the area is summed up, weighted average is used for precipitation intensity, and mode operator is applied to aggregate the soil type. For every node in the graph a multivariate state-of-the art sequence model such as LSTM or Gated Temporal Convolution Network (GTCN) is designed for capturing the temporal behavior of precipitation, discharge or evapotranspiration. It is obvious that the aggregation reduces the number of sequence models required at some performance loss. Finally, for the extraction of the spatial information a second sequence model is applied. We refer to Zhongrun und Demir,[14] for further details.

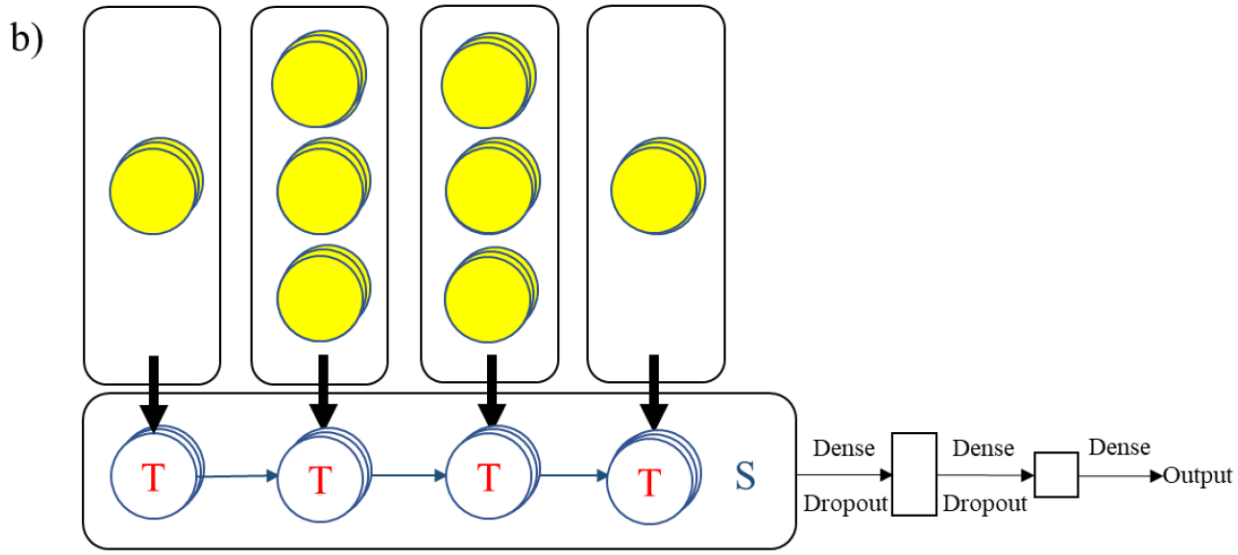


Figure 3: Structure of the GNRRM-TS model (source: [14])

Spatio-temporal attention and LSTM (STA-LSTM)

The second baseline is described in detail in [15] and is not graph based, but makes full use of spatio-temporal information directly. In this method, the LSTM is modified to include spatial and temporal attention mechanism as shown in Figure 4. It takes the input variables e.g. precipitation, discharge, evapotranspiration and use them to predict the future catchment area streamflow n-timesteps. In the first part of the model, the input features are dynamically weighed in every time step as in Equation 3 using the weight vector $\alpha_k = SA(x_k) = [\alpha_1^k, \alpha_2^k, \dots, \alpha_m^k]_{1\times m}^T$. Afterward, the hidden layer state of each step of LSTM is fully utilized to dispatch temporal attention weights to the hidden layer state of each time step. The two attention modules helps to dynamically adjust the attention weights as well as improve the performance of LSTM cell. The input to the spatial attention module is a 2D spatio-temporal feature matrix $X \in R^{m\times k}$ which can be divided into $m \times k$ -dimension vectors $x_k = [f_1^k, f_2^k, \dots, f_m^k]_{m\times 1}$. Hereby, the number of features in each step is m and the number of time steps are k. The hydrology features usually include rainfall and flow of multiple stations.

$$x'_k = \alpha_k \cdot x_k = [\alpha_1^k f_1^k, \alpha_2^k f_2^k, \dots, \alpha_m^k f_m^k]_{1\times m}^T \quad (3)$$

Before the feature inputs are multiplied by the weights α using the Hadamard product (elementwise multiplication), they are activated by sigmoid function and then normalized using a softmax function to ensure the limited additivity of weights.

The results of the spatial attention module flows into a LSTM cell for the temporal attention. The hidden layer state output sequence $H = [h_1, h_2, \dots, h_k]_{k\times s}$ is obtained successively. s is the number of state features. After ReLU activation and normalization with a softmax function the attention weight vector $\beta = TA(H) = [\beta_1, \beta_2, \dots, \beta_k]_{1\times k}$ is multiplied the hidden feature vector as in Equation 6 to get the attention-weighted state features h_{att} and finally the outputs p are generated directly without further activation function as in Equation 5.

$$h_{att} = \beta \times H = \sum_{i=1}^{k} \beta_i h_i, h_{att} \in R^{1\times s} \quad (4)$$

$$p = O(h_{att}), p \in R^{1\times n} \quad (5)$$

URL: http://dx.doi.org/10.14738/tmlai.105.13049

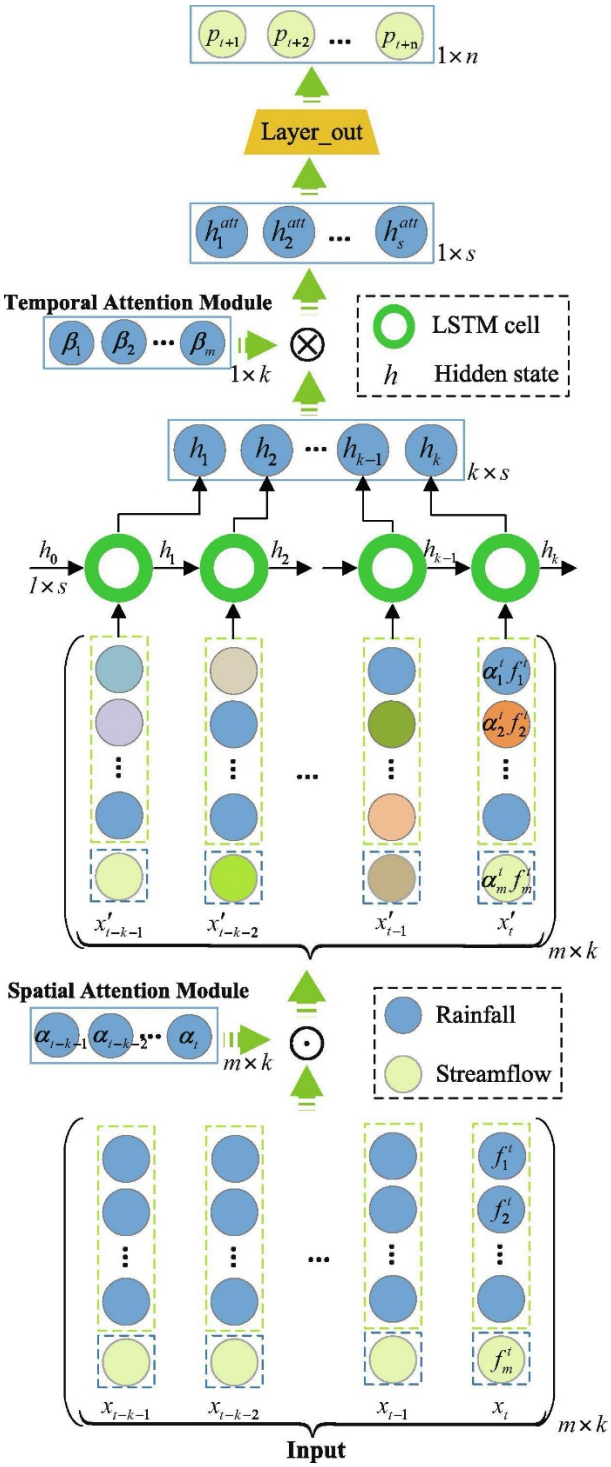


Figure 4: Spatio-Temporal Attention LSTM model (source: [15]).

EXPERIMENTS AND ANALYSIS

Data from a model region in Germany was used for the studies to compare the different physic-inspired methods for streamflow prediction. For fair comparison, the datasets (training, validation and test) are identical for all of the models. Figure 5 shows exemplary an extraction of the rainfall and streamflow for a station. The data included 15 minutes observations of precipitation, evapotranspiration and discharge from different stations in the area. Further,

some constant characteristic features of the sites were also available. This included information on drainage area, time of concentration, slope calculated from a DEM and soil information which was encoded into different types (loam, impervious surface, etc.). As the data falls in different magnitudes, they were normalized using the min-max operator before the training phase. The observed data was divided into training, validation and testing sample. Observations from January 2003 to July 2011 was used for training, Data from July 2011 to Juni 2016 was used for validation and the rest of the data from Juli 2016 to February 2022 was used for testing. In addition some early recordings of extreme events from 1804 and 1910 were added to the training set. The models use past seventy two hours information on rainfall and streamflow to predict the streamflow of the next hours. As a goal we set to predict a minimum of 6 hours ahead as may flash floods happens in the that time after heavy rainfall. It has been found in several experiments and also in [14] that the knowledge about the past three days of a catchment are critical for future predictions.

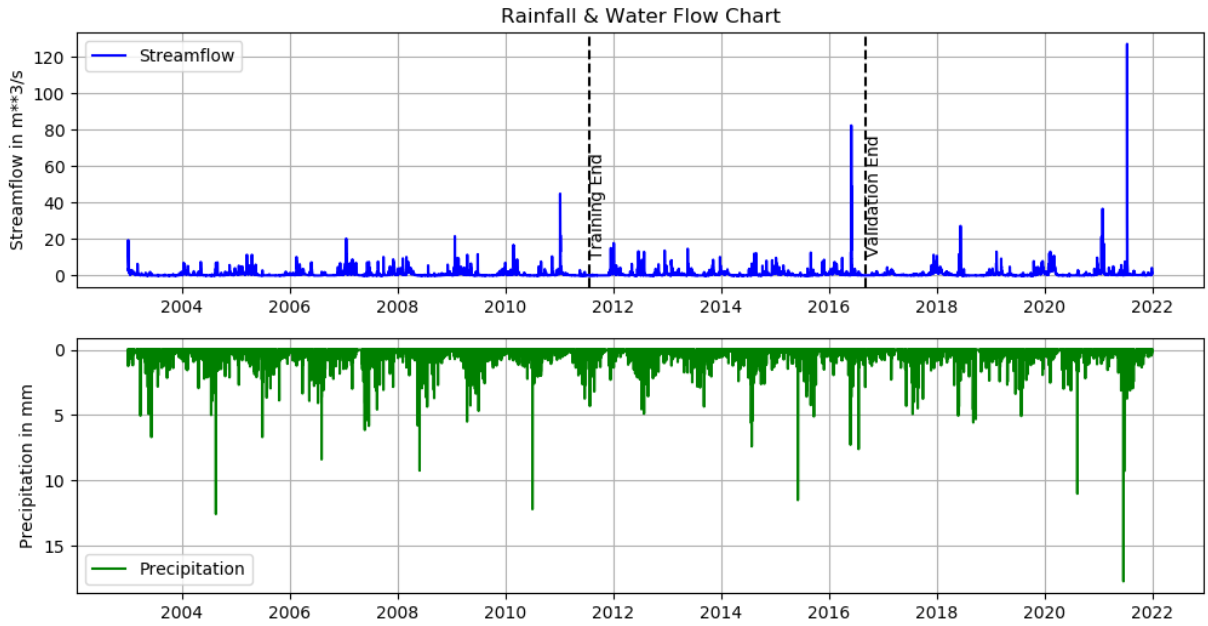


Figure 5: Rainfall and streamflow chart for the region divided into training, validation and test sets

As required by our 2D-GAT-BiLSTM method, the inputs were reshaped into a tensor consisting of 3 dimensions: features, timesteps and flow stations. For the experiment, flow measuring stations with fifteen features were taken into consideration for comparison. Therefore, each training sample has $15 \times 72 \times 3$ shape, corresponding to the $features \times timesteps \times stations$. As in the other models seventy two past timesteps were taken for the sampling. The model was trained using RMSprop optimizer with learning rate of 0.0003 with early stopping at 20 epochs.

For the GNRRM-TS several sequence models (LSTM, GTCN) and their bidirectional forms were used in the graph and tested for their performance in [14]. The LSTM was finally chosen as the sequence model due to the following reasons. In bidirectional form, all models have the ability to capture more information coming from both directions, unfortunately, this is associated with more parameters for learning and the training takes more time and memory. This has also been

URL: http://dx.doi.org/10.14738/tmlai.105.13049

confirmed in many other studies such as in [23],[24]. The same applies to the GTCN and the number of parameter increase with increasing length which is required to capture long-term information necessary in streamflow prediction. The advantage of the LSTM are apparent in that the model parameters will not be affected by the sequence length, even though also the computing time increase when the sequence length increases. The accuracy of all the models differs in the case study marginal. Therefore, a LSTM model with a 5-layers with 32 neurons in each layer has shown the best performance with higher computational efficiency.

For the STA-LSTM model, the input size was set to $k \times m$, where k is the number of timesteps and m is the number of features, which is fifteen. Experiments were also conducted to find the best performing STA-LSTM structure in terms of the activation functions in the spatial and temporal attention modules. The results showed that applying sigmoid activation in the spatial module and relu in the temporal module produced the best robustness and accuracy of the model. The model was trained for 60 epochs using the Adam optimizer to ensure convergence and efficiency. The weight decay in the optimizer was set to 0.0004. The initial learning rate was 0.05.

RESULTS AND DISCUSSIONS

Two performance metrics are used to evaluate the similarity between observed and simulated discharge of the prediction models for the different timesteps ahead. The first one is the traditional commonly used in hydrological models NSE, because it normalises model performance into an interpretable scale as expressed in Equation 3.

$$NSE = 1 - \frac{\sum_{t=1}^{T} \hat{Q}(t) - Q(t)^2}{\sum_{t=1}^{T} Q(t) - \bar{Q}(t)^2}, \quad (3)$$

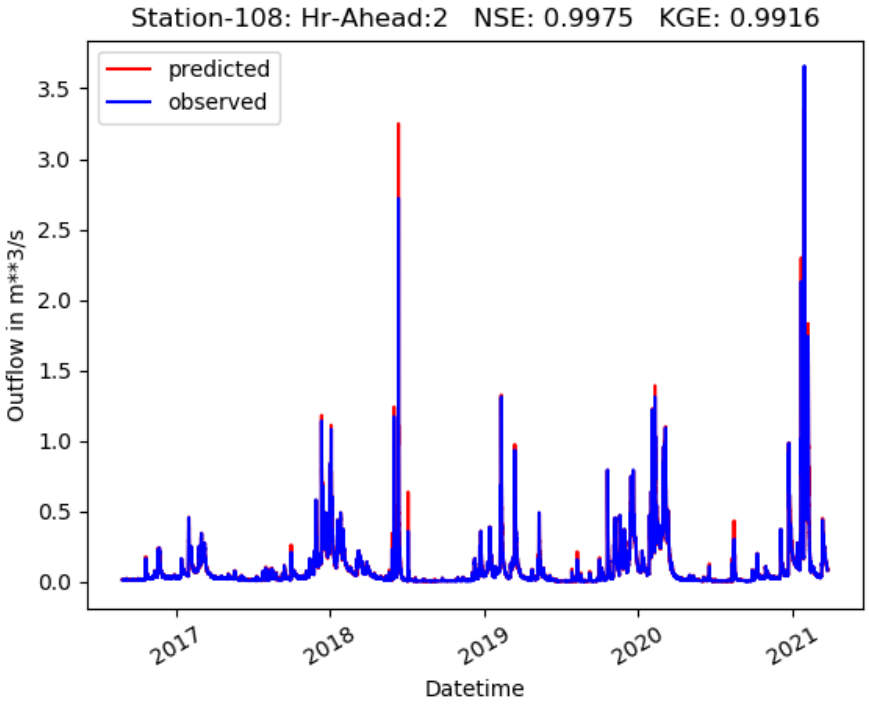
where T is the total number of time steps, $\hat{Q}(t)$ the simulated discharge at time t, Q (t) the observed discharge at time t,and $\bar{Q}$(t) is the mean observed discharge. NSE = 1 indicates perfect correspondence between simulations and observations; NSE= 0 indicates that the model simulations have the same explanatory power as the mean of the observations.

The second metric is the KGE. This metric addresses several shortcomings of the NSE and is increasingly being used for model calibration and evaluation for hydrological models.

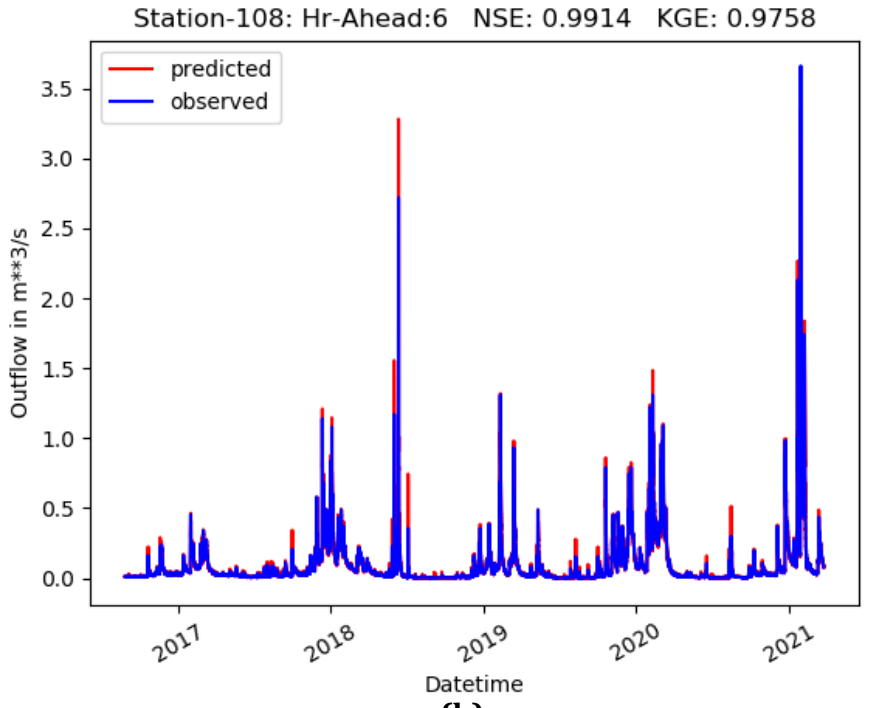
$$KGE = 1 - \sqrt{(r - 1)^2 + \left(\frac{\hat{\sigma}}{\sigma} - 1\right)^2 + \left(\frac{\hat{\mu}}{\mu} - 1\right)^2}, \quad (4)$$

where r is the linear correlation between observations and simulations, $\alpha = \frac{\hat{\sigma}}{\sigma}$ is a measure of the flow variability error, and β=$\frac{\hat{\sigma}}{\mu}$ is a bias term. σ is the standard deviation in observations, $\hat{\sigma}$ the standard deviation in simulations, $\hat{\mu}$ is the simulation mean, and μ the observation mean. Like NSE, KGE = 1 indicates perfect agreement between simulations and observations. Analogous to NSE = 0, certain authors state that KGE < 0 indicates that the mean of observations provides better estimates than simulations [26], although 5 others state that this interpretation should not be attached to KGE = 0 [27],[28]. Various authors use positive KGE values as indicative of 'good' performance whereas negative KGE values are considered 'bad', without explicitly indicating that they treat KGE = 0 as their threshold between 'good' and 'bad' performance.

Firstly, the results of the developed method will be shown and then compared to the baselines. For illustration the prediction of 2 and 6 hours ahead for station 108 are shown in Figure 6 a-c. It can be seen that the method produces a very good result with an NSE for 6 hours ahead of 0.822 and a KGEof 0.7434 for this station. As shown in the Figure 7, the NSE and KGE depreciate slightly for each hour ahead.



(a)



(b)

Figure 6: Prediction results for Station 108 for the a) 2 hrs ahead, b) 6 hrs ahead

URL: http://dx.doi.org/10.14738/tmlai.105.13049

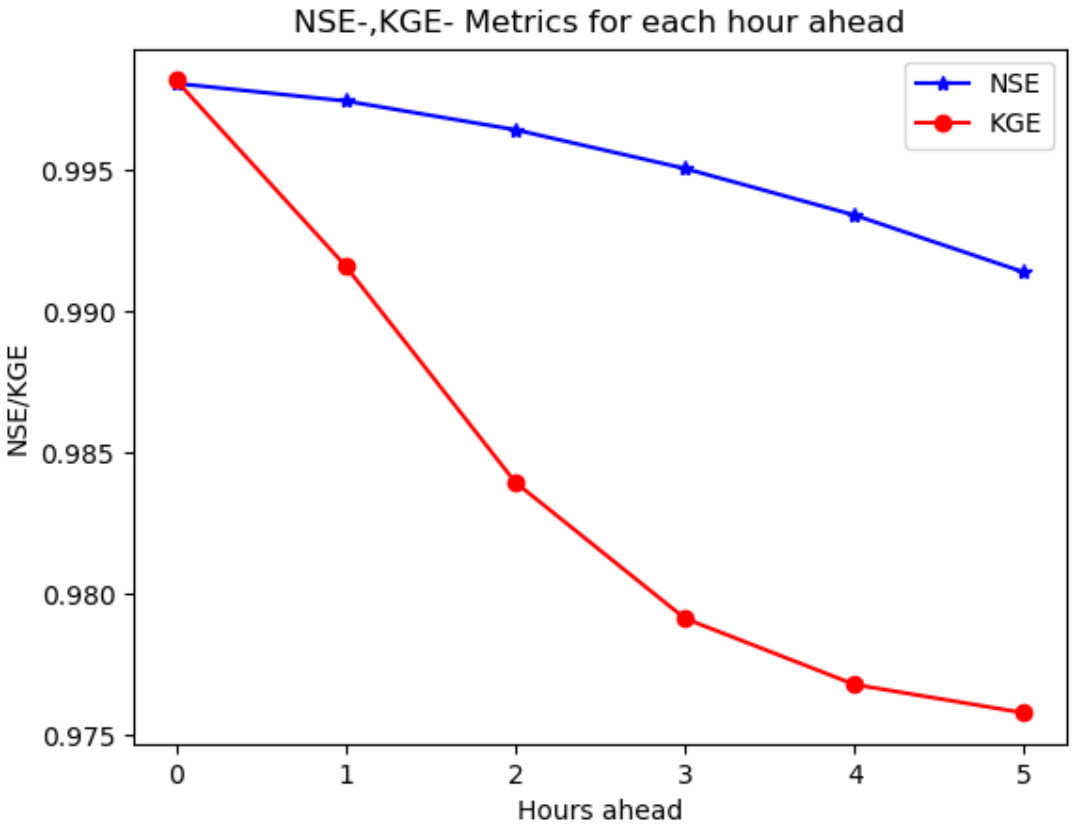


Figure 7: depreciation of the NSE and KGE for each hour ahead for station 108

The results of the streamflow prediction by the three models in comparison are shown in Table 1. It can be seen that all the model types produce acceptable results with an average prediction NSE and KGE of 0.946 and 0.939, respectively for GNRRM, an average prediction NSE and KGE of 0.995 and 0.981, respectively for 2DGAT-BiLSTM and an average prediction NSE and KGE of 0.816 and 0.747, respectively for STA-LSTM. It is evident that all the methods benefit from the simultaneous consideration of spatial and temporal features. Regarding the training time, the GNRRM-TS required 45 min on GPU for 100 epochs, GAT-LSTM 22min for 100 epochs and STA-LSTM 32min for 100 epochs on CPU.

Table 1: Model performance in comparison according to the median NSE and KGE for the catchment area

	1hr ahead		**3hrs ahead**		**6hrs ahead**		**6hrs average**		Training time
Models	NSE	KGE	NSE	KGE	NSE	KGE	NSE	KGE	
GNRRM-TS	0.968	0.962	0.957	0.936	0.935	0.914	0.946	0.939	45 min GPU
2DGAT-BiLSTM	**0.998**	**0.998**	**0.996**	**0.983**	**0.993**	**0.976**	**0.995**	**0.981**	**22 min CPU**
STA-LSTM	0.817	0.757	0.815	0.748	0.812	0.743	0.816	0.747	32 min CPU

These results show that the proposed 2D GAT with BiGTCN or BiLSTM model produces the best result for each station. The proposed 2D GAT-BiLSTM model utilizes all the spatial and temporal information of the data which is enabled by the combination of GAT and BiLSTM layers. Compared to GNRRM with a fixed Adjacency Matrix based on the direct distance between nodes shows that the spatial relationships between the nodes are complex and need learnable Adjacency matrix. Furthermore, GNRRM do not have an implicit attention mechanism and the temporal information is possibly utilized more efficiently with the LSTM layer in 2D GAT-BiLSTM than the one dimensional filters present in Temporal Convolutional Networks in GNRRM-TS which use GTCN. Moreover, GNRRM depends heavily on the method to find the direction of flow. The attention based model, whether GAT or Spatial and temporal attention mechanism perform better and train faster. The problem with the STA-LSTM method even though it has an attention mechanism is that it is not based on a graph structure to really imitate the flow directions.

The experiments and their results show that the 2D GAT-BiLSTM developed in this study is an effective model structure for multi-steamflow modeling under consideration of meteorological, hydrological and geographical information of the catchment area.

As an extra result, the 2DGAT-LSTM gives the attention weights between the different features in the nodes. For the three stations attention weights are represented in Figure 6. The colormaps shows how much streamflow of a target station is affected by features from other stations. For example, the streamflow from station No. 505 has significant effect on the streamflow of station No. 108 as shown in figure 6a.

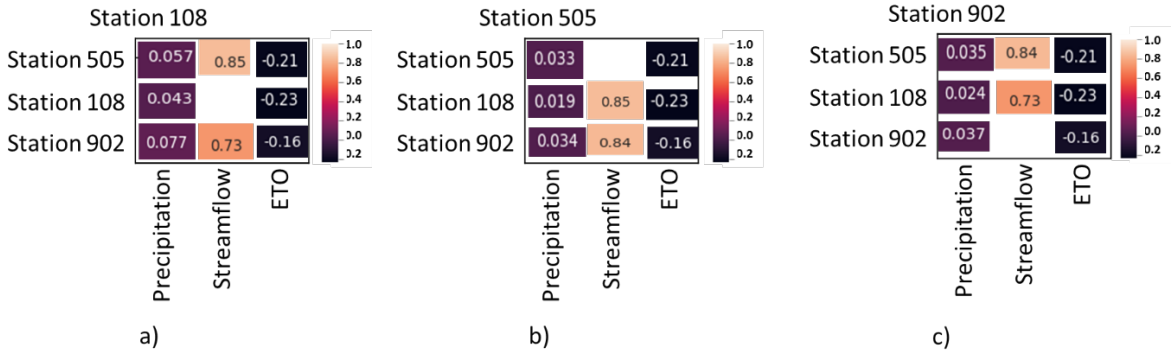


Figure 6: The attention weights of stations per meterological/hydrological variables

CONCLUSION

A method based on 2D Graph Attention Neural Networks and BiGTCN/BiLSTM was presented. The method was compared for its performance to two current developments in deep learning models for streamflow prediction one based on GNN and the other one based on Spatial-Temporal Attention mechanism. It outperforms the comparison models in terms of NSE and KGE due its way of dynamically adjusting the learnable adjacency matrix for creating reliable representations of the spatial relationships between the features of the different station. With the BiGTCN/BiLSTM for modelling the temporal behavior, it has more tunable parameters and can look further back in the timeseries. Analysis of the produced attention weights brings interpretability in the model. As an outlook, it is important to find out whether, or not the results will be confirmed in other region for streamflow prediction.

URL: http://dx.doi.org/10.14738/tmlai.105.13049

References

Löwe, R.; Urich, C.; Domingo, N.D.F.S.; Mark, O.; Deletic, A.; Arnbjerg-Nielsen, K. Assessment of Urban Pluvial Flood Risk and Efficiency of Adaptation Options Through Simulations—A New Generation of Urban Planning Tools. J. Hydrol. **2017**, 550, 355–367

Yildirim, E., & Demir, I. (2021). An Integrated Flood Risk Assessment and Mitigation Framework: A Case Study for Middle Cedar River Basin, Iowa, US. International Journal of Disaster Risk Reduction, 56, 102113.

Xia, X.; Liang, Q.; Ming, X.; Hou, J. An efficient and stable hydrodynamic model with novel source term discretization schemes for overland flow and flood simulations. Water Resour. Res. **2017**, 53, 3730–3759

Kim, B.; Sanders, B.F.; Famiglietti, J.S.; Guinot, V. Urban flood modeling with porous shallow-water equations: A case study of model errors in the presence of anisotropic porosity. J. Hydrol. **2015**, 523, 680–692.

Cea, L.; Garrido, M.; Puertas, J. Experimental validation of two-dimensional depth-averaged models for forecasting rainfall-runoff from precipitation data in urban areas. J. Hydrol. 2010, 382, 88–102.

Sit, M., Demiray, B. Z., Xiang, Z., Ewing, G. J., Sermet, Y., & Demir, I. (2020). A comprehensive review of deep learning applications in hydrology and water resources. Water Science and Technology, 82(12), 2635-2670.

Kratzert, F., Klotz, D., Hochreiter, S., & Nearing, G. S. (2021). A note on leveraging synergy in multiple meteorological data sets with deep learning for rainfall–runoff modeling.Hydrology and Earth System Sciences, 25(5), 2685-2703

Xiang, Z., Demir, I., Mantilla, R., & Krajewski, W. F. (2021). A Regional Semi-Distributed Streamflow Model Using Deep Learning.EarthArxiv.https://doi.org/10.31223/X5GW3V

Xu, Y., Hu, C., Wu, Q., Li, Z., Jian, S., & Chen, Y. (2021). Application of temporal convolutional network for flood forecasting.Hydrology Research.

Lin, K., Sheng, S., Zhou, Y., Liu, F., Li, Z., Chen, H., ... & Guo, S. (2020). The exploration of a Temporal Convolutional Network combined with Encoder-Decoder framework for runoff forecasting.Hydrology Research, 51(5), 1136-1149.

Van, S. P., Le, H. M., Thanh, D. V., Dang, T. D., Loc, H. H., & Anh, D. T. (2020). Deep learning convolutional neural network in rainfall–runoff modelling. Journal of Hydroinformatics, 22(3), 541-561.

Zhao, Q., Zhu, Y., Shu, K., Wan, D., Yu, Y., Zhou, X., & Liu, H. (2020). Joint Spatial and Temporal Modeling for Hydrological Prediction.IEEE Access,8, 78492-78503.

Zhou, J., Cui, G., Zhang, Z., Yang, C., Liu, Z., & Sun, M. (2018). Graph neural networks: A review of methods and applications. arXiv preprint arXiv: 1812.08434.

Zhongrun & Demir, Ibrahim. (2022). Fully distributed rainfall-runoff modeling using spatial-temporal graph neural network. 10.31223/X57P74.

Yukai Ding, Yuelong Zhu, Jun Feng, et al. Interpretable spatio-temporal attention LSTM model for flood forecasting. Neurocomputing 2020(403): 348-359.

S. Song, C. Lan, J. Xing, W. Zeng, J. Liu, An end-to-end spatio-temporal attention model for human action recognition from skeleton data, in: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4–9, 2017, San Francisco, California, USA, 2017, pp. 4263–4270. URL: http://aaai.org/ocs/index.php/AAAI/AAAI17

Nash, J. E. and Sutcliffe, J. V.: River flow forecasting through conceptual models part I — A discussion of principles, J. Hydrol., 10(3), 282–290, doi:10.1016/0022-1694(70)90255-6, 1970.

Gupta, H. V., Kling, H., Yilmaz, K. K. and Martinez, G. F.: Decomposition of the mean squared error and NSE performance criteria: Implications for improving hydrological modelling, J. Hydrol., 377(1-2), 80–91, doi:10.1016/j.jhydrol.2009.08.003, 2009.

Shelef, E., & Hilley, G. E. (2013). Impact of flow routing on catchment area calculations, slope estimates, and numerical simulations of landscape development.Journal of Geophysical Research: Earth Surface,118(4), 2105-2123.

P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, andY. Bengio, “Graph attention networks,” 2017.

Fang, K., Shen, C., Kifer, D., & Yang, X. (2017). Prolongation of SMAP to spatiotemporally seamless coverage of continental US using a deep learning neural network.Geophysical Research Letters,44(21), 11-030.

Kong, X., Xing, W., Wei, X., Bao, P., Zhang, J., & Lu, W. (2020). STGAT: Spatial-Temporal Graph Attention Networks for Traffic Flow Forecasting.IEEE Access,8, 134363-134372.

Kang H, Yang S, Huang J, Oh J. Time series prediction of wastewater flow rate by bidirectional LSTM deep learning. International Journal of Control, Automation and Systems. 2020 Dec;18(12):3023-30.

Lees, T., Buechel, M., Anderson, B., Slater, L., Reece, S., Coxon, G., & Dadson, S. J. (2021). Benchmarking Data-Driven Rainfall-Runoff Models in Great Britain: A comparison of LSTM-based models with four lumped conceptual models. Hydrology and Earth System Sciences Discussions, 1-41.

T. Stańczyk and S. Mehrkanoon, “Deep graph convolutional networks for wind speed prediction,”Accepted for publication in Proc. of the 29th European Symposium on Artificial Neural Networks (ESANN), Oct. 2021, Bruges, Belgium, 2021.

Castaneda-Gonzalez, M., Poulin, A., Romero-Lopez, R., Arsenault, R., Chaumont, D., Paquin, D. and Brissette, F.: Impacts of Regional Climate Model Spatial Resolution on Summer Flood Simulation, in HIC 2018. 13th International Conference on Hydroinformatics, vol. 3, pp. 372–362., 2018.

Gelati, E., Decharme, B., Calvet, J. C., Minvielle, M., Polcher, J., Fairbairn, D. and Weedon, G. P.: Hydrological assessment of atmospheric forcing uncertainty in the Euro-Mediterranean area using a land surface model, Hydrol. Earth Syst. Sci., 22(4), 15 2091–2115, doi:10.5194/hess-22-2091-2018, 2018.

Mosier, T. M., Hill, D. F. and Sharp, K. V.: How much cryosphere model complexity is just right? Exploration using the conceptual cryosphere hydrology framework, Cryosph., 10(5), 2147–2171, doi:10.5194/tc-10-2147-2016, 2016.

Kling, H., Fuchs, M. and Paulin, M.: Runoff conditions in the upper Danube basin under an ensemble of climate changescenarios, J. Hydrol., 424-425, 264–277, doi:10.1016/j.jhydrol.2012.01.011, 2012.

URL: http://dx.doi.org/10.14738/tmlai.105.13049