Convolutional Neural Network (CNN) for Streamflow Projection

Shiheng Duan[1] Paul Ullrich[1] Lele Shu[1] David Hall[2]

[1]University of California, Davis [2]Nvidia

Model Design

Motivation, Data and Area of Interested

Streamflow prediction and projection can be viewed as a time series problem. Long-Short-Term Memory (LSTM) neural networks have been a popular choice for this problem. However, inspired by Bai et al. (2018), we investigate the use CNN for streamflow projection. Catchment Attributes for Large-Sample Studies (CAMELs) provides hydrology data and forcing data for 671 basins over CONUS. There are 40 basins in California, and we selected 20 of them without missing values. We used NLDAS as our forcing data.

Model Inputs and Output

The input variables to our model are precipitation, solar radiation and temperature. The output is streamflow. The time window is set to 365 days. Thus, our model can be expressed as:

$$Q_{365} = f(P_1, P_2, \ldots, P_{365}; T_1, T_2, \ldots, T_{365}; S_1, S_2, \ldots, S_{365}),$$

where Q stands for streamflow, P for precipitation, T for temperature, and S for solar radiation.

Figure: Map for basins and stations in California.

Model Architecture

We used 1-D temporal CNN (TCNN) for streamflow data. In our model, there are three TCNN blocks. In each block, there are two CNN layers and one residual connection, which helps stack deeper layers. In each CNN layer, a dilated CNN and causal padding method are used to keep physical causation and increase reception fields, respectively.

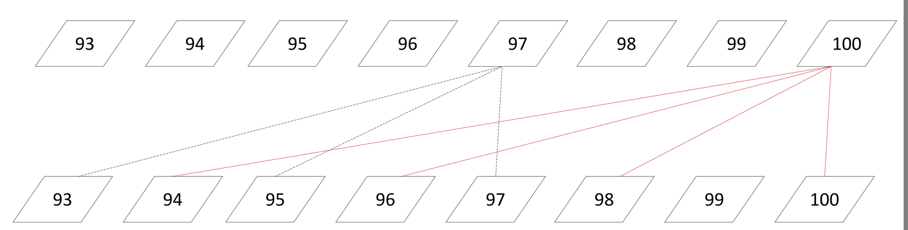

Figure: illustration of dilated causal padding method.

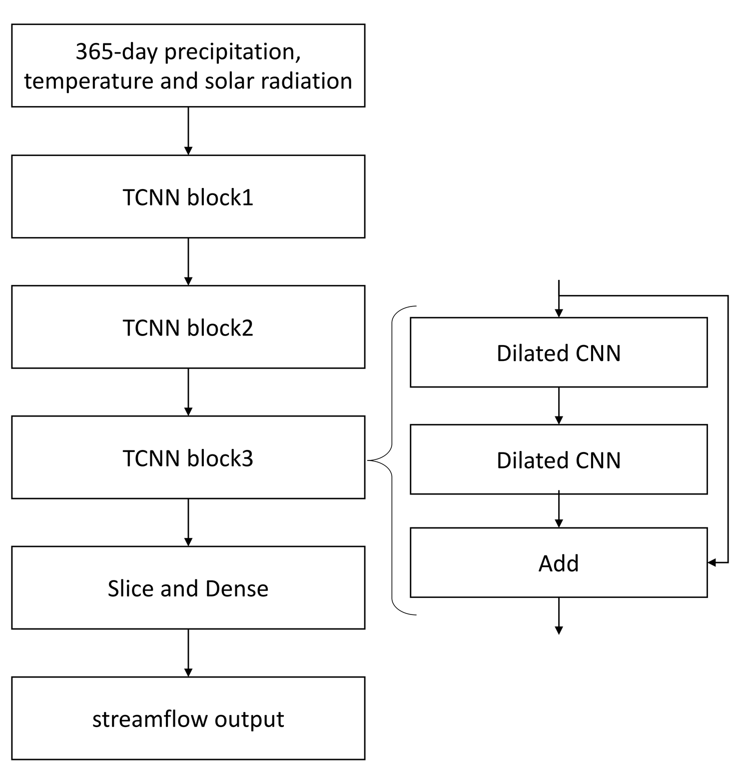

Figure: The data flow chart through the CNN.

With such dilated causal padding structure, the information will concentrate in the last few neurons after the three TCNN blocks. To keep simplicity and avoid overfitting, we placed a slice layer in the end of the TCNN blocks to only retain the last 20 neurons.

Model Training Setup

The model is trained separately for each station. There are 9636 training samples and 2419 testing samples. When tuning hyperparameters, 25% of the training samples are used as validation set. To make our model simple and general, the hyperparameters are the same for all basins. The loss function is set to 1-NSE, where NSE is Nash-Sutcliffe coefficient. The training time with a single RTX 2080Ti is around 44 seconds.

Prediction

Ensemble Run

Since the neural networks use gradient based method to optimize the loss function, the initial weights will affect the model performance. In order to obtain a general idea for how our model performs for a random initial state, we run the model 10 times to get a distribution of NSE values.

The model with the maximum NSE value for each station is then used for our projection test.

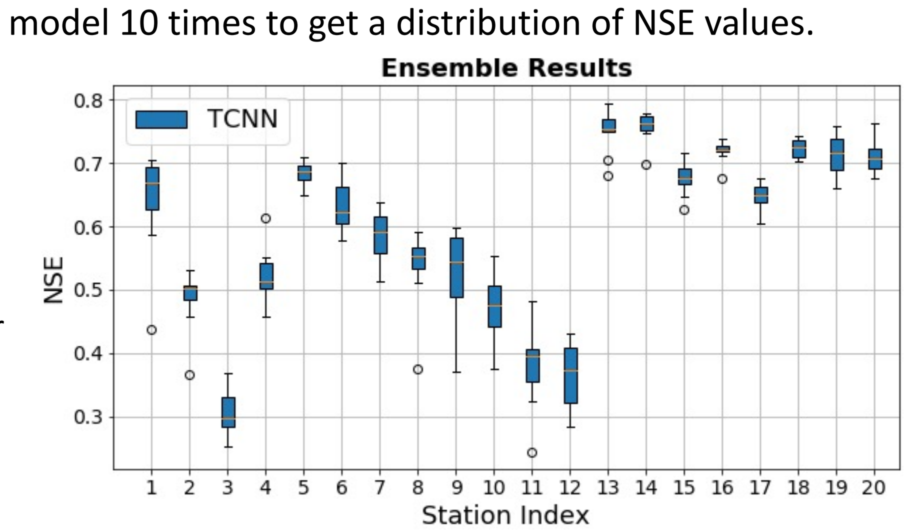

Figure: Ensemble results for different stations. (right)

Time Window Size

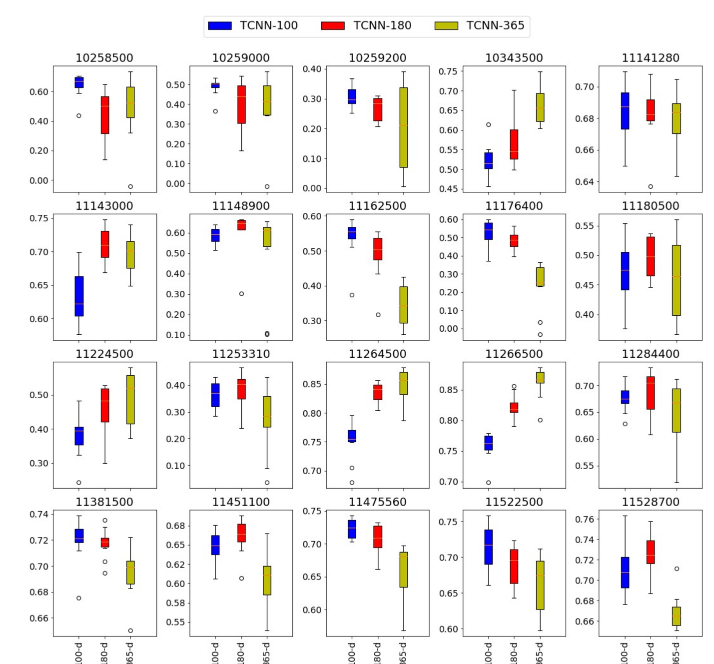

Figure: Ensemble results for different time window sizes (100, 180, and 365 days).

The input time window size is an important hyperparameter which should be explained using physical relationships. In our case, the response time for precipitation, groundwater and snowpack ranges from several hours to months.

Here we tested 100-day, 180-day and 365-day as time window size. Stations where NSE is monotonically increasing with window size are in mountainous area. Stations where NSE is monotonically decreasing with window size are in northern California and close to costal area. This suggests that snow dynamics requires a longer window size, and that window size needs to be reflective of the groundwater dynamics in a basin.

Predictability of Different Variables

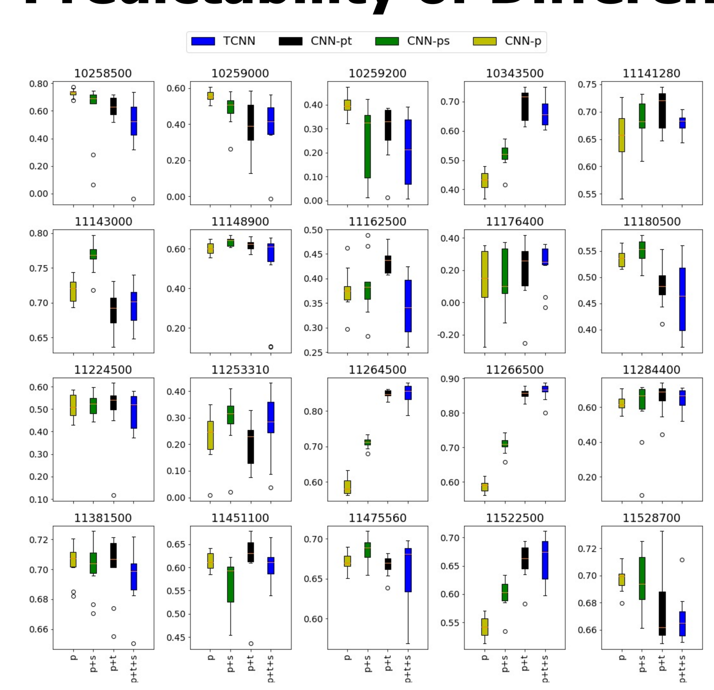

Figure: Ensemble results for models using a different subset of input variables – precipitation (p), solar radiation (s), and temperature (t).

We also tested the predictability of different variables under 365-day time window. Here we focus on temperature and solar radiation.

Temperature boosts predictability in Stations 10343500, 11162500, 11264500, 11266500 and 11522500. These stations are in inland mountain areas except 11162500 and 11522500, which are in mountains not far from coastline.

Solar radiation provides a boost to predictability in Stations 11143000 and 11180500. These two stations represent coastal basins. 11143000 is near Big Sur in Central Coastal California and 11180500 is in the southern San Francisco Bay.

Projection

Projection Data

Localized Constructed Analogs (LOCA) dataset provides all the necessary input variables for our model. It is a statistical downscaled climate dataset with 6-km resolution. There are four major climate models in LOCA dataset, which are CanESM2, MIROC5, HadGEM2-ES and CNRM-CM5. Before feeding LOCA data into our model, we scaled LOCA data based on historical records to make LOCA consistent with NLDAS forcing data. The projection period is from 2070-01-01 to 2099-12-31.

Double Precipitation Test

The model was tested with a two times precipitation scenario to determine if our model can avoid unphysical results even when the climatology is extrapolated. Double precipitation is an unrealistic extreme scenario, even in light of climate change. Results are shown here from one station.

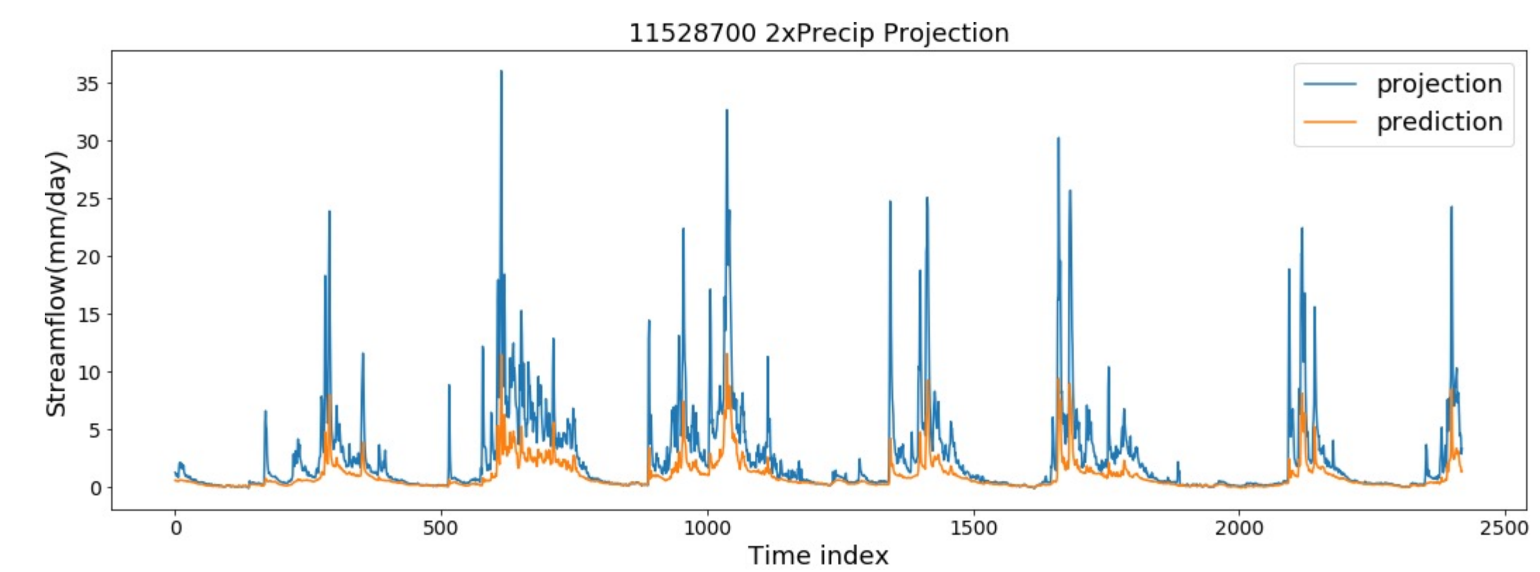

Figure: Hydrograph for the double precipitation projection experiment.

Projection with CanESM2

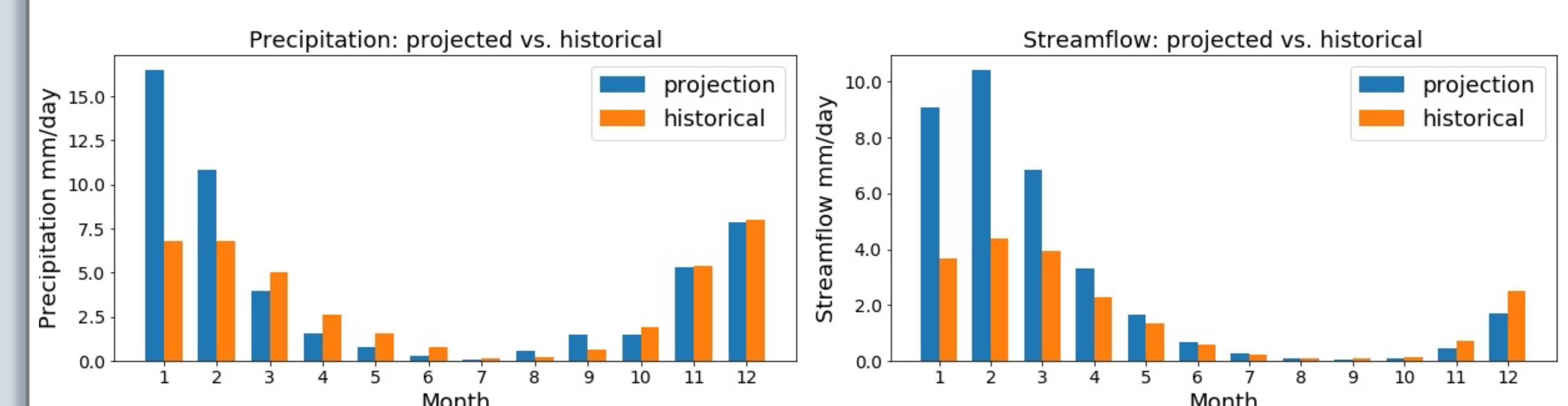

Figure: Monthly mean precipitation and streamflow from CanESM2 for station 11528700.

These figures depict projected precipitation and streamflow versus the historical monthly mean for station 11528700, using LOCA data downscaled from CanESM2. Together with the flow duration curve, in terms of climatology the maximum and minimum of streamflow will not change too much under the RCP 8.5 scenario. However, overall flow rate will increase and, if these results are examined in conjunction with a flood model, we observe that the probability of flooding will also increase.

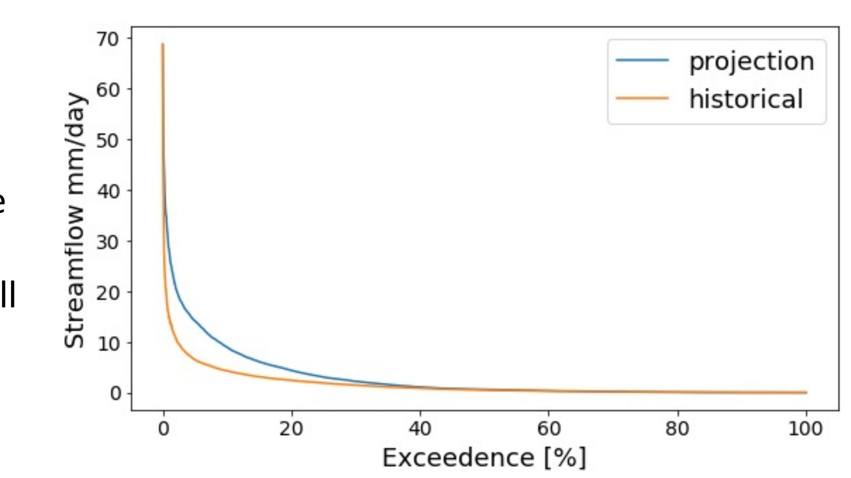

Figure: Flow duration curve comparison.

References

- Newman, A. J., et al. "A large-sample watershed-scale hydrometeorological dataset for the contiguous USA." UCAR/NCAR, doi 10 (2014): D6MW2F4D.
- Bai, Shaojie, J. Zico Kolter, and Vladlen Koltun. "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling." arXiv preprint arXiv:1803.01271 (2018).
- Photograph by Nick Ares, distributed under a CC BY-SA 2.0 license.

Acknowledgements

Support comes from the U.S. Department of Energy, Office of Science, Office of Biological and Environmental Research program under Award Number DE-SC0016605 "A framework for improving analysis and modeling of Earth system and intersectoral dynamics at regional scales."

Contact

shiduan@ucdavis.edu