Data-driven approaches to rainfall nowcasting for application in hydrological modelling

Rim Mhedhbi [a], **Marina G. Erechtchoukova** [a]

[a] *School of Information Technology, York University, Toronto, Canada (rimouch1@yorku.ca)*

Abstract: Flash floods are amongst the most complex and destructive phenomena. An abundance of process-based and data-driven models was proposed to serve as decision support tools for flood management authorities. While various observed hydrological and meteorological characteristics were usually used as an input for flash flood modelling, it was also found that integrating rainfall forecasts could considerably enhance the models' predictive ability. This study focuses on finding reliable and efficient data-driven rainfall nowcasting models (0-2h lead time). These models could then be integrated into a short-term flash flood prediction framework to investigate the framework performance including the effect of the precipitation nowcasts on the reliability of the modelling results. It is important to note that only data from rain gauges located on the same watershed are used to predict future precipitation. Rainfall data obtained from two rain gauges installed in the Spring Creek watershed, Ontario, Canada were used in this study. The investigated watershed is highly urbanized and prone to flash floods. Investigated data spanned four years from 2013 to 2016. We tackled this data-driven modelling problem from two perspectives: (1) an algorithmic and (2) a data-centric. From the algorithmic perspective, a comparative study of three data-driven models was performed. These models included the status quo persistence model, the statistical AutoRegressive Integrated Moving Average (ARIMA) model and the deep learning Long Short-Term Memory (LSTM) model. These models were applied to each time series to predict rainfall in the respective rain gauge location (univariate modelling). Following the data-centric approach, data from both sensors were combined into one dataset to predict rainfall in each sensor location (multivariate modelling). Lagged rainfall values from the sensor at the target location and the adjacent sensor were fed into an LSTM model to predict rainfall at the target location. Models were created for each investigated year for lead times ranging from 15 minutes to 60 minutes (corresponding to the time scale of the investigated rainfall events). Data for each year were chronologically split into training and testing with a 70%:30% split ratio. Root Mean Square Error (RMSE) and Maximum Residual Error (MRE) were used as evaluation metrics. Obtained results showed that overall, according to the estimated RMSE, LSTM demonstrated a better performance for all years except the year 2015. Figure 1 depicts models' performance for 2013 at the Hart Lake location using single sensor data. Further analysis revealed that the year 2015 had major hydrological pattern difference between the training and testing sets. MRE did not indicate major variations between the years; it was found that all the models performed approximately at the same level as the persistence model. The models failed to predict extreme values accurately. The data-centric approach, however, showed different results. According to the RMSE and MRE metrics, LSTM models trained using data from both sensors demonstrated major improvement on data from years 2014 and 2015 for both target areas. Evaluation of the model performance on data from years 2013 and 2016 gave inconsistent results. Further investigation showed that the improvement in the model predictive ability coincided with the sensors' location and the dominating wind direction in the modeled years. In general, combining data from multiple sensors when used with the LSTM model showed promising results. Further extension of input variables including meteorological data collected on the investigated watershed will be the next step of the presented study.

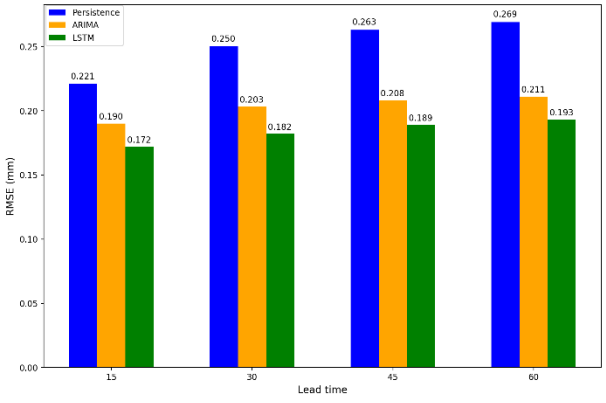

Figure 1. Models performance in terms of RMSE on Heart Lake location 2013 data.

Keywords: *Rainfall nowcasting, data-driven modelling, LSTM, ARIMA, precipitation*

1. INTRODUCTION

This study stems from a framework for short-term prediction of hydrological events using the machine learning (ML) approach. The framework requires only data generated by hydrological monitoring networks and are readily available almost in real time. Foundational work conducted by Erechtchoukova et al. (2016) proved the effectiveness of this framework as a warning tool for flash flood events. The proposed framework generates a trained model, which takes past observed rainfall and stage measurements as input to predict the occurrence of flood events. In practice, flood management teams usually receive a meteorological forecast including rainfall nowcasting from Environmental Agencies and submit the forecast as a part of input data to a process-based model predicting hydrological conditions on a watershed. Therefore, modification of the framework to include rainfall predictions into a set of independent variables is strongly desirable.

This idea finds its support in the research community. Tao et al. (2015) reported that introducing Ensemble Precipitation Forecast products in a hydrological model remarkably enhanced water discharge prediction. Song et al. (2019) showed that, with the use of future rainfall values, regression models successfully predicted flood events. Brendel et al. (2020) proved the effectiveness of integrating quantitative precipitation forecasts in an urban flooding hydrology model. In a similar study, Ko et al. (2020) worked on enhancing short-term intensive rainfall forecasts to improve a hydrologic model's predictive ability.

This study investigates data-driven regression models for rainfall forecasting using data available on a watershed. The goal is to find a suitable model to generate short-term precipitation predictions in an efficient way. These predictions will be integrated into the short-term flash flood prediction framework to investigate the extended framework performance including the effect of the precipitation nowcasts on the reliability of the modelling results. It is important to note that only data from rain gauges located on the same watershed are used to predict future precipitation.

2. METHODOLOGY

2.1. Background

Modern approaches to rainfall nowcasting heavily rely on available data sources and mathematical and computational techniques. These approaches include Numerical Weather Prediction (NWP) using simulation models describing physical atmospheric and oceanic processes, data assimilation which combines and adjusts the results of simulations with newly obtained observation data, radar nowcasting modelling which is based on relatively simple extrapolation of radar data. Given the large volumes of accumulated data and the growing availability of open-source data analytics software, data-driven approaches attract the attention of researches and practitioners.

A plethora of data-driven prediction models was proposed for rainfall nowcasting in the literature. Statistical and machine learning models represent two main subcategories. AutoRegressive Integrated Moving Average (ARIMA) models are considered among the most widely used statistical model. With the growing popularity of machine learning, a large number of learners was proposed. These learners range from relatively simple linear regression algorithms to more sophisticated ones such as Artificial Neural Networks (ANNs) (RanjanNayak et al. (2013)).

Depending on the characteristics and the scale of the modeled rainfall phenomenon, different models were found suitable. Toth et al. (2000) compared the performance of several data-driven models. The investigated models included AutoRegressive Moving Average (ARMA), ARIMA, ANN and, K-Nearest Neighbor models (KNN). The rainfall prediction lead time spanned from one to six hours. The authors reported that ANN delivered the best predictive performance and improved rainfall-runoff modelling. Nasseri et al. (2008) integrated ANN with the multi-sensor data from rain gauges for short-term rainfall prediction. Genetic Algorithm (GA) was used to select informative subset of rain gauges. The prediction lead time ranged from one to 150 minutes. The authors applied sensitivity analysis to determine the best lag times for input variables and the best surrounding stations. It was found that ANN coupled with GA for network optimization consistently outperformed the ANN. Using a simpler machine learning model, Nikam and Gupta (2014) trained a Support Vector Machine (SVM) based model for very short-term intensive rainfall forecasting with lead times varying from five to 20 minutes. A rainfall event was considered intensive if the intensity exceeded 50 mm/h. The authors found that their model gave satisfactory results when different models were trained for different rainfall ranges. However, it was also found that the proposed model underestimated peak values.

Recently, a lot of attention was shifted toward the application of deep learning algorithms in meteorological modelling. Particularly, Long Short-Term Memory (LSTM) models demonstrated superior performance in many studies. Shi et al. (2015) applied a Convolutional LSTM (ConvLSTM) sequence-to-sequence model for rainfall prediction based on radar maps. The forecasting window spanned up to six hours. The proposed model was compared with the Fully Connected-LSTM (FC-LSTM) model and a radar extrapolation technique called Rover. It was found that ConvLSTM significantly outperformed both models thanks to its ability to capture spatio-temporal variations. Based on the LSTM model architecture, Sato et al. (2018), also proposed a model for rainfall forecasting called PredNet. The model takes as input maps of rainfall amounts and outputs frames for the next 10 timesteps with five minutes resolution. The proposed model was compared with the Gated Recurrent Unit (GRU) network model for both classification and regression tasks and was found to outperform the compared model. It was also reported that the model performed better at the classification task than the regression task.

Overall, a wide variety of data-driven models was proposed for rainfall modelling. However, modelling for rainfall nowcasting was not investigated extensively. More specifically, very short-term rainfall prediction based on limited input variables (e.g., only rainfall gauges measurements) was not fully addressed.

2.2. Data-driven rainfall prediction models

The aim of this study was to find a data-driven regression model that has a satisfactory predictive ability to fit the short-term flood prediction framework. Due to data limitations, this model takes only ground-based rainfall measurements. To tackle this modelling problem, we considered two approaches. The first is an algorithm-centric approach wherein we investigated the performance of several data-driven models using single sensor data. The second approach is data-centric. Based on the latter approach we combined data coming from neighboring sensors to predict the rainfall at the target location.

In the context of the algorithm-centric approach, we investigated the performance of the persistence, AutoRegressive Integrated Moving Average (ARIMA), and LSTM models.

The persistence model is a trivial model. It assumes that atmospheric conditions remain constant. The model predicts rainfall at the time (t + lead time) using rainfall value at time t. It is also called the naïve model. This model is used as a baseline for the comparative analysis.

The ARIMA model is a popular statistical time series forecasting model. It consists of two components. The first component is a linear combination of observed values (deterministic). The second component is the sum of random errors (stochastic). Future values of the time series variable Y_t are predicted according to the formula presented in equation (1):

$$Y_t = \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + .. + \beta_p Y_{t-p} + \epsilon_t + \varphi_1 \epsilon_{t-1} + \varphi_2 \epsilon_{t-2} + .. + \varphi_q \epsilon_{t-q,} \quad (1)$$

where ϵ_t is the random error at time t, β_i $(i = 1 .. p)$ and φ_i $(i = 1 .. q)$ represent the model parameters; p is the autoregressive component order and q is the moving average component order. It is important to note that depending on the degree of differencing d, the variable Y_t would be differenced d times before being modeled using equation (1).

LSTM is a variant of the Recurrent Neural Network (RNN) architecture. The latter is a type of ANN architecture that models sequential data. Unlike traditional ANN, RNN models the dependencies between input data by adding links between inter-layer nodes. The RNN model was found to deliver good performance for time series modelling. Nonetheless, due to its recurrent architecture and when the number of layers increases, this architecture is prone to the vanishing gradient problem. That is, changes in later layers cannot be reflected in the preceding layers. This renders the RNN architecture unable of capturing long-term dependencies. LSTM was thus proposed to tackle this issue (Pascanu et al. (2012), Jozefowicz et al. (2015)). LSTM architecture is mainly characterized by its memory cells. A memory cell helps to conserve information from one LSTM block to another which gives the model the ability to capture and represent long-term dependencies. LSTM networks are therefore capable of modelling both short-term and long-term dependencies, which makes them an attractive tool for time series modelling (Lipton et. al (2015)).

The second approach we considered for rainfall modelling focused on the data used rather than the modelling algorithm. For each prediction location, lagged rainfall values from the target and neighboring locations were used as input. To determine the lag time of adjacent gauges, we conducted an analysis of the time difference between the beginning of rainfall events at the adjacent gauges and the target gauge. Based on the distribution of time differences, a suitable number of lags was used for each gauge.

2.3. Evaluation metrics

Various evaluation metrics are used for rainfall modelling. In this study, the Maximum Residual Error (MRE) and the Root Mean Squared Error (RMSE) were selected to assess the compared models' performance. MRE was considered because it reveals the maximum error committed while RMSE penalizes large errors. These properties are of particular importance since we intend to integrate the forecasted values in the flash flood prediction framework. Equations (2) and (3) represent the formulae for MRE and RMSE respectively:

$$MRE = MAX_i(|\hat{Y}_i - Y_i|), \quad (2)$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{Y}_i - Y_i)^2}, \quad (3)$$

where Y_i represent the observed value and $\hat{Y}_i$ is the predicted value by the model.

3. DATASET AND ITS SOURCES

Computational experiments were conducted on data collected by the Toronto and Region Conservation Authority (TRCA) monitoring network. The studied area covers the Spring Creek watershed and is situated in the southern part of the Peel Region, Ontario, Canada. The rainfall time series are recorded by rain gauges installed on the studied area. The rain gauges were placed in Heart Lake (HL) and Mississauga Works Yard (M) locations (Figure 2). Observed values span the warm period (April-December) of four years from 2013 to 2016. It is also important to mention that rain gauges generate data with five minutes temporal resolution. To have a better understanding of the considered dataset, Table 1 presents summary statistics of the measured rainfall amounts at the HL observation site during the considered years. Table 1 shows that the studied years have distinct hydrological characteristics. With an accumulated rainfall amount of 780.2 mm during the warm period, the year 2013 was considered wet. The remaining years were considered dry.

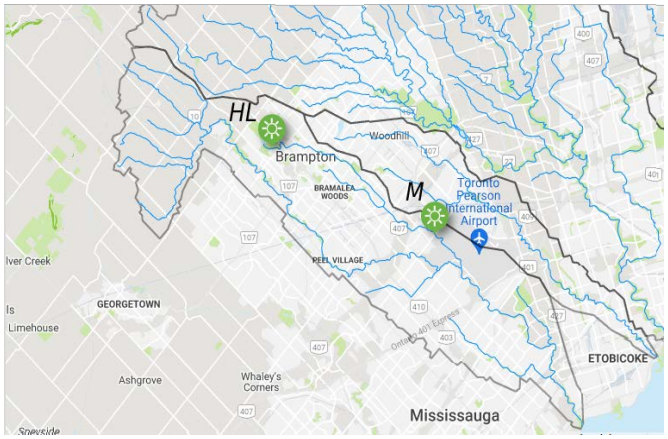

Figure 2. Rain sensors installed at Spring Creek watershed (TRCA real-time gauging).

4. DATA PREPROCESSING AND EXPERIMENTAL SETTINGS

Before the modelling phase, the collected data needs to go through data pre-processing which includes data cleansing and transformation. Missing and erroneous values were first identified. Data analysis revealed that such values corresponded to a dry period. The faulty records were therefore imputed with zero values. Further, the used time series were characterized by a fine granularity (five minutes). Thus, upscaling was performed to transform the temporal resolution from five minutes to 15 minutes.

Table 1. Summary statistics of rainfall at Hart Lake observation site

Year	Accumulated rainfall (mm)	Maximum observed rainfall (mm) *
2013	780.2	17.6
2014	529.2	13.2
2015	527.8	12.6
2016	289.4	11.2

* Maximum observed rainfall values with 15 min granularity.

The experiments were conducted using Python programming language version 3.7. This software tool offers a wide variety of packages for time series processing and modelling, it is characterized by its ease of use and significant community support. Overall, three sets of experiments were performed. These sets correspond to the application of the persistence, ARIMA, LSTM and multi-sensor approach to each of the studied years. Models for lead times ranging from 15 min to 60 min were created. Further, data corresponding

to each year were chronologically split to train and test sets. 70% of the data were used for models' training and 30% for models' error estimation.

The persistence model was directly applied to the test set, as it does not involve any model training. To create the ARIMA estimator, a modelling procedure was adopted. First, data were checked for stationarity. Auto Correlation Function (ACF) and Partial Auto Correlation Function (PACF) plots were produced and examined to get insights about the maximum values of the moving average and the autoregressive parameters, q and p respectively. The models were then created for different combinations of q and p while taking into account their maximum values. The best performing model was selected according to the Akaike information criterion, which is a statistical measure of the goodness of fit of an estimator. Selected model performance is assessed on the holdout test sample. The Python package 'statsmodels' was used to automate this procedure.

To develop LSTM models several steps were performed. The time series values were first normalized using the MinMax scaler. Obtained data were then transformed into a format of input-output variables, wherein the input variables represented lagged measurements of rainfall prior and up to time t and the output variable represented a precipitation magnitude at the prediction time (t + number of prediction time steps). To determine the suitable number of the lagged measurements, insights from ACF and PACF plots were taken into account. A trial-and-error procedure was then applied to pick the number of steps appropriate for each dataset. Furthermore, the ADAM optimizer was used for model training. The architecture consisted of a number of stacked LSTM hidden layers and a fully connected dense layer. Various hyperparameters were considered. We ranged the number of layers and the number of units in each layer. We also considered several values for the learning rate, batch size and the number of epochs. RELU, SIGMOIND and TANH functions were also tested out for the transfer function. A randomized hyperparameters search was applied for the hyperparameter tuning process. This is a combination of hyperparameters selected randomly from a set of all possible combinations. LSTM models were created based on TensorFlow 2.5 deep learning package and Keras 2.5 library. The latter uses TensorFlow as an engine and offers abstraction to facilitate modelling process. Automated hyperparameter tuning was performed using Keras tuner framework for each developed model.

It is important to add that all algorithms were first compared using data coming from a single sensor, i.e. rainfall at each location was predicted based on historic data from the same location. In the data-centric approach, multi-sensor data were fed into the LSTM model. That is data from both HL and M were used to predict rainfall at HL and M locations.

5. RESULTS AND DISCUSSION

Figure 3 presents the predictive ability of the investigated models in terms of RMSE on HL location using solely HL data. Overall, the persistence model delivered the worst performance. The remaining compared models gave inconsistent results for different modelled years. While in 2013, the LSTM model significantly outperformed the other models with a decrease in RMSE reaching roughly 30%, in the other years results were different. In 2014 and 2016, the LSTM model delivered comparable results to those of the ARIMA model. However, the LSTM model had the largest error on data for the year 2015. Further analysis of the hydrological characteristics of the precipitation in 2015 showed salient discrepancies in rainfall distributions between the training and the testing sets periods. The model was thus unable to learn all the hydrological patterns due to the non-representativeness of

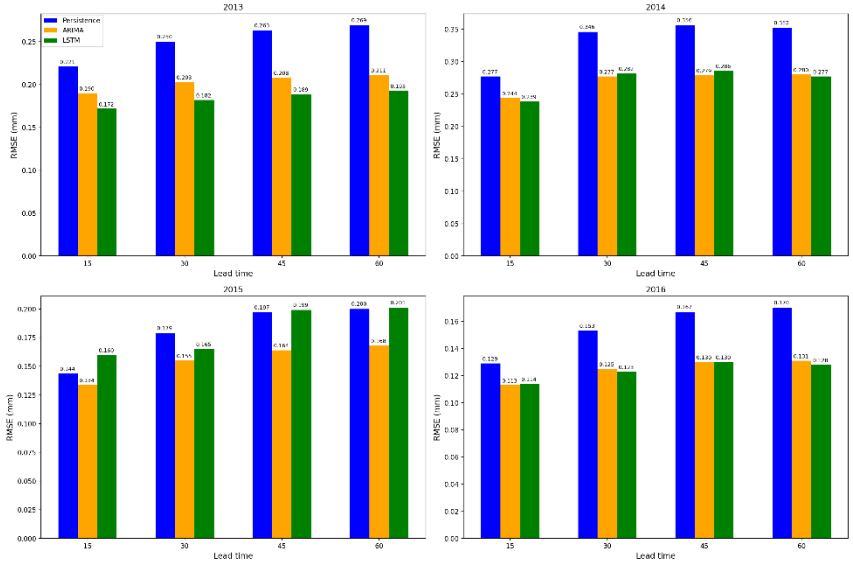

Figure 3. Persistence, ARIMA, and LSTM models' performance in terms of RMSE on the rainfall prediction task at HL location using data from HL sensor.

the training set. Overall, even though the LSTM model yielded superior performance in some years, its superiority cannot be generalized and largely depends on the patterns in hydrological conditions within a single year. Therefore, a greater focus should be placed on data representativeness rather than model sophistication.

In addition to the RMSE, models were compared using MRE evaluation metric. It was found that all the models performed comparably similar to the naïve model (around 9 mm for 2013, 12 mm for 2014, 4 mm for 2015 and 2016). Error analysis showed that the models erroneously predict extreme peak values that occurred at the beginning or end of the rainfall events.

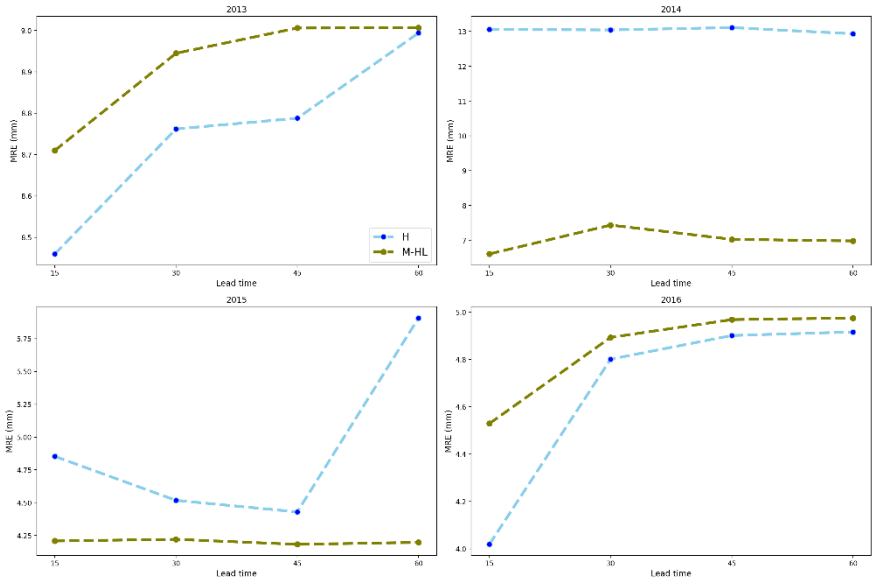

Figure 4. LSTM rainfall prediction model performance at HL location using data solely from HL sensor and a combination of both datasets from HL and M

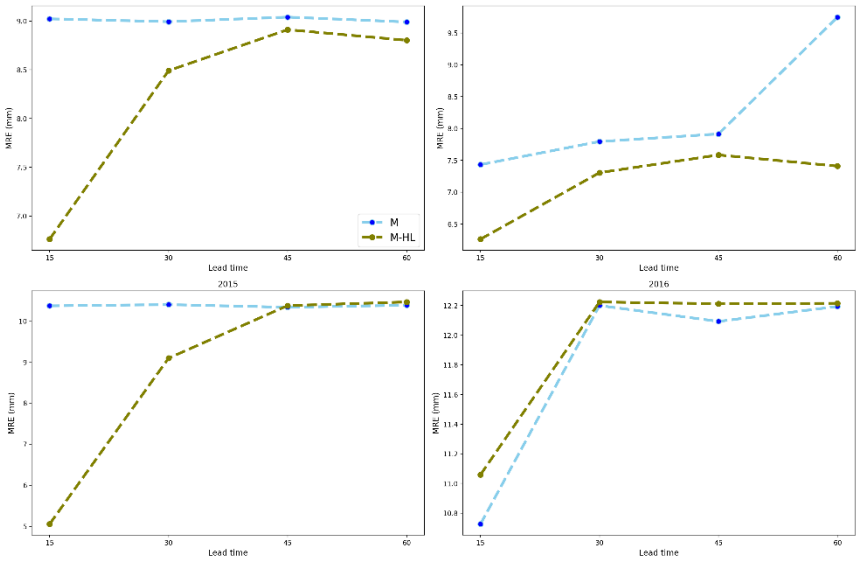

Figure 5. LSTM rainfall prediction model performance at M location using data solely from M sensor and a combination of both datasets from HL and M

Figures 4 and 5 present the MRE obtained from the multi-sensors (or data-centric) approach models. Figure 4 compares the rainfall prediction error at the HL location calculated taking as input only data from HL rain gauge with the ones calculated based on data from both M and HL observation sites. Figure 5 presents the results for rainfall prediction at the M location. Contracting the results obtained from the first set of experiments for algorithm-centric approach, and depending on the investigated year and location, the trained model showed major improvement in MRE estimated on the testing sets. Overall, the results demonstrated that combining results from both stations enhanced the models' predictive ability in both locations for 2014 and 2015. Nevertheless, while integrating 2013 data from both gauges significantly increased the performance for location M, the model performance dropped for HL location. Combining 2016 data from both gauges failed to enhance the performance as well. This fact can be explained by dominating meteorological conditions on the watershed varying from year to year. Archived data obtained from the Environment Canada website showed that records for 2013 mostly represent winds blowing from north to south. Given that the HL is located in the northern part of the investigated watershed and M gauge is installed on the southern part, the improved prediction at the location M while data from both gauges are utilized can be easily justified.

6. CONCLUSION

In general, we found that for rainfall nowcasting using univariate data (data obtained from a single sensor), the LSTM model outperformed the ARIMA and naïve models with the exception of the year 2015, where major discrepancies in the rainfall distribution were observed. Nonetheless, all the models were unable to accurately predict extreme values and performed similarly to the naïve model. Computational experiments showed that integration of data from several sensors brought major improvement to models' performance depending on the hydrological characteristics of the year and the location of sensors. More specifically, using lagged values from sensors that coincided with the dominating direction of the air flow enhanced models' predictive ability considerably.

Obtained results revealed that the data-centric approach (combining multiple sensors) along with the LSTM model gave promising results. Thus, more focus should be placed on data representativeness and richness. On the next step of the study, the uncertainty analysis of predicted rainfall magnitudes will be performed. Future work will be based on the application of the LSTM model to datasets from multiple sensors which selection will be done according to meteorological data analysis.

ACKNOWLEDGMENTS

Data used in the study was provided by Toronto and Region Conservation Authority. The authors are thankful to TRCA and, especially, J. Duncan for providing data and necessary clarifications. The authors are grateful to editors and anonymous reviewers for their thoughtful suggestions and helpful comments on the manuscript improvement.

REFERENCES

Brendel, C.E., Dymond, R.L., Aguilar, M.F., 2020. Integration of quantitative precipitation forecasts with real-time hydrology and hydraulics modeling towards probabilistic forecasting of urban flooding. Environmental Modelling & Software 134, 104864. doi:10.1016/j.envsoft.2020.104864.

Erechtchoukova, M.G., Khaiter, P.A., Saffarpour, S., 2016. Short-term predictions of hydrological events on an urbanized watershed using supervised classification. Water Resources Management 30, 4329–4343. doi:10.1007/s11269-016-1423-6.

Józefowicz, R., Zaremba, W., Sutskever, I., 2015. An empirical exploration of recurrent network architectures, in: ICML.

Ko, C.M., Jeong, Y.Y., Lee, Y.M., Kim, B.S., 2020. The development of a quantitative precipitation forecast correction technique based on machine learning for hydrological applications. Atmosphere 11, 111. doi:10.3390/atmos11010111.

Lipton, Z.C., Berkowitz, J., Elkan, C., 2015. A critical review of recurrent neural networks for sequence learning arXiv:1506.00019.

Nasseri, M., Asghari, K., Abedini, M., 2008. Optimized scenario for rainfall forecasting using genetic algorithm coupled with artificial neural network. Expert Systems with Applications 35, 1415–1421. doi:10.1016/j.eswa.2007.08.033.

Nikam, V., Gupta, K., 2014. SVM-based model for short-term rainfall forecasts at a local scale in the Mumbai urban area, india. Journal of Hydrologic Engineering 19, 1048–1052. doi:10.1061/(asce)he. 1943-5584.0000875.

Pascanu, R., Mikolov, T., Bengio, Y., 2012. On the difficulty of training recurrent neural networks arXiv:1211.5063.

Ranjan Nayak, D., Mahapatra, A., Mishra, P., 2013. A survey on rainfall prediction using artificial neural network. International Journal of Computer Applications 72, 32–40. doi:10.5120/12580-9217.

Sato, R., Kashima, H., Yamamoto, T., 2018. Short-term precipitation prediction with skip-connected PredNet, in: Artificial Neural Networks and Machine Learning – ICANN 2018. Springer International Publishing, pp. 373–382. doi:10.1007/978-3-030-01424-7_37.

Shi, X., Chen, Z., Wang, H., Yeung, D.Y., kin Wong, W., chun Woo, W., 2015. Convolutional lstm network: A machine learning approach for precipitation nowcasting arXiv:1506.04214.

Song, T., Ding, W., Wu, J., Liu, H., Zhou, H., Chu, J., 2019. Flash flood forecasting based on long short-term memory networks. Water 12, 109. doi:10.3390/w12010109.

Tao, P., Tie-Yuan, S., Zhi-Yuan, Y., Jun-Chao, W., 2015. Application of quantitative precipitation forecasting and precipitation ensemble prediction for hydrological forecasting. Proceedings of the International Association of Hydrological Sciences 368, 96–101. doi:10. 5194/piahs-368-96-2015.

Toth, E., Brath, A., Montanari, A., 2000. Comparison of short-term rainfall prediction models for real-time flood forecasting. Journal of Hydrology 239, 132–147. doi:10.1016/s0022-1694(00)00344-9.