24th IMEKO TC4 International Symposium
22nd International Workshop on ADC and DAC Modelling and Testing
IMEKO TC-4 2020
Palermo, Italy, September 14-16, 2020

Real-time flood prediction using Recurrent Neural Networks and Random Forest

Petar Sekulić, Paola Regina, Luana Spadafina, Giuseppe Dentamaro, Alessandro Porcelli, Cristiano Bove, Slavko Kovačević, Mirko Kalezić

Research Group at Omnitech S.r.l.

***Abstract* – Floods are one of the most destructive natural disasters as they cause severe material damage and often the loss of human life. Predicting a flood is a challenging task and recent progress in this field was brought by machine learning (ML) models. This paper aims to discover a dependency between spatial and temporal data patterns about the ground and weather that will subsequently lead to the floods. Taking into account features deriving from remote sensing images and weather stations, the proposed algorithm aims to predict whether the flood will happen or not by using Recurrent Neural Networks and Random Forests methods. The algorithm could be considered as a starting point of the research related to prediction of other natural disasters, such as landslides, heavy rainfall, droughts, weather forecasting, etc. The future direction of research aims at improving the accuracy of the algorithm by employing a broader and more structured data set built from validated sources.**

I. INTRODUCTION

Flood forecasting is a process in which a timely estimation of flooding, its magnitude and duration is performed. The aim of this paper is to predict the flood as early as possible, with the intention of alerting the populace in a timely manner and thus attempting to reduce both casualties and material damage. As a result, the probability of a flood occurring at a given point was obtained, with the setting of a certain threshold, which produced a binary output. The study area included in the research consisted of Apulia and Liguria regions (Italy), for which data from weather stations, satellite images, DEM and land use maps were considered within a 15 years timespan. Aforementioned data were processed using artificial neural networks (ANNs) algorithms. Current state-of-the-art algorithms in natural disaster prediction are artificial neural networks (ANNs) which brings hope that additional breakthrough can be made not only in the accuracy of flood forecasting but in the speed of calculations too. The main reason why ANNs were used in this paper is to cope with the chaotic and non-linear nature of the dynamic features in the system, such as the one related to weather. Besides, considering the use of a larger dataset as future research, ANNs would be a reliable tool as they can cope with data scalability [1]. Finally, Recurrent Neural Networks (RNNs) [2], in particular, were considered a natural choice as they create a temporal sequence of input data which allows the algorithm to observe gradual escalation of rainfall. There are several variables whose participation leads to flooding and they were splitted into two distinctive groups, namely dynamic and static data, as features like rainfall or wind speed vary in real-time, providing information about current weather, while some other, typically spatial ground data like vicinity to rivers or hills, or water absorption, are considered as fixed and can serve to reveal location vulnerabilities. Given the infrequency of flood events, the dataset was characterized by skewed classes, thus days without flooding were dominant. The proposed research combined the RNN with Random Forest (RF) for classification [3]. As an ensemble of decision trees models, Random Forest handles outliers and non-linear features. Moreover, it exhibits low bias and moderate variance model [4].

II. RELATED WORKS

In the last decade, machine learning models for flood prediction have become more frequently used [5]. Based on the duration of predictions, state-of-the-art models could be divided into short and long-term. Both of these groups can be later divided into single and hybrid methods: the former use only one algorithm to achieve their goal, while the latter combine several algorithms into one model. In [6], fully-connected neural networks, along with the use of GIS were used for flood modeling, being their causative factors known. Data layers with different thematic corresponding to causative and intensifying flood factors were considered. Seven features, namely, rainfall, geology, slope, elevation, soil, flow accumulation and land use were chosen as inputs to the neural network that had just two hidden layers and only one output. Hidden layers had only few neurons but they were enough to generate flood maps. In [7], a flood hazard risk assessment model was presented, based on random forest decision trees. The model took as risk indices maximum three-day precipitation, runoff depth, typhoon frequency, digital elevation model (DEM), and topographic wetness index and, secondarily, normalized difference vegetation index (NDVI), stream power index, soil texture, distance to the river, slope, and land use pattern. The research presented

in [8] compared empirical methods to estimate the instantaneous peak flow (IPF) - a parameter taken into account in the design of solutions for the flood risk management - based on maximum mean daily flow (MMDF), artificial neural networks (ANN), and adaptive neuro-fuzzy inference system (ANFIS). In [9] Season-multilayer perceptron (SAS-MP) and hybrid wavelet-season-multilayer perceptron (W-SAS-MP) were combined to form a system aimed at predicting heavy rainfall that will subsequently lead to flood. This system enhanced prediction accuracy and extend prediction lead time of daily rainfall up to 5 days.

III. DATASET

There are probably hundreds of possible ground features that could be used by a model, but not all of them are relevant for determining the flooding risk of index for a given location. As previously mentioned, the study area comprised Apulia and Liguria regions in a timespan ranging from 2005 to 2019. Weather data were provided by Arpa Puglia[1] and Regione Liguria[2] were employed. Arpa Puglia source offers 30 to 40 stations, while there were 208 stations for Liguria. In order to cope with the variability of the number of stations and some time differences between measurements through time, an interpolation was performed, considering the mean time difference, which is 5 hours. Considering only the stations closer to the flood events, and characterized by the by the lowest presence of missing data, only 10 stations for Apulia and 4 for Liguria were selected. The research also took into account the Digital Elevation Model (DEM) made accessible by regional authorities for both Apulia[3] and Liguria[4]. For the Normalized Difference Vegetation Index (NDVI) and Land Use Pattern maps, the data provided by the Copernicus European Program[5] were employed. Finally, soil texture data were provided by SoilGrids[6]. Main static data features employed in the current research are Digital elevation model (DEM), Stream Power Index (SPI), Slope (SL, degrees), Soil texture (ST) and Distance to the river (DR, m), while features that are of dynamical type are pressure (mbar), temperature (celsius), wind speed (meters per second), wind direction (degrees), humidity (percentage) and precipitation (mm). Before launching the algorithm, a preprocessing phase was performed on the data. In particular, both static and dynamic data underwent a standard score normalization technique.

IV. THE PROPOSED SOLUTION

In the proposed study, only the dynamical data passed through RNN which consisted of two LSTM cells [10]. From the available features in one particular moment, vector was formed with the size of $F \times 1$ where F represents number of dynamic features, 6 in this case. Time frame for each single event was 3 days. There were 3×24 time steps where 24 represents number of hours in a day as 1 hour is time resolution. Input size was $batch\_size \times F \times 72$. Each sample passed through both LSTM cells 72 times. The last state of this architecture was the output of the RNN. This process is shown in Fig. 1, output has $batch\_size \times 64$ size where $batch\_size$ is 128.

A. Architecture

In the proposed approach, as shown in Fig. 2, the output of the RNN went through two dense layers. First dense layer dimensions were 64×32 and it used ReLU activation function. Second dense layer dimensions were 32×1 and it used Sigmoid activation function. Result was a value in $[0, 1]$ range which represented probability of flood happening. Before adding Random Forest, initial network architecture was trained. The first dense layer was used as a feature extractor as the output weights are input to Random Forest algorithm. This feature extractor extracted 32 features for Random Forest. Extracted features were concatenated with static data which consisted of 5 features, forming the vector of size 37 that served as an input for Random Forest algorithm. Number of trees was 100 and the maximum depth of the three was 2. In order to test the effectiveness of the dynamic architecture of the proposed approach, the classification results were also compared with static data. In particular, a mean F1 score of 0.448 and a mean AUC value of 0.482 were reported for the RF classifier with the static data as input.

B. Training and cross-validation

Neural networks were trained in batches. The dataset consisted of only 160 flood events vs. 7294 no-flood events, which resulted in a severe displacement. Stochastic gradient descent optimization wouldn't work without some adjustment and the process of training would be deformed. The proposed approach balanced the available dataset so the algorithm used an equal number of flood and no-flood events, by randomly selecting 160 no-flood events from the whole sample. Loss function was a binary cross-entropy function, sometimes called logarithmic loss, which penalized mistakes in order to lead the model to learn to differentiate between two possible states. Dataset was splitted on training, test and validation sets so that the training sets contained 80 entries, while test and validation sets contained 40 of both flood and no-flood samples each. Both training and cross-validation were performed in 100 experiments in which samples were randomly chosen for each of the set.

[1] http://dati.arpa.puglia.it/
[2] https://regione.liguria.it/
[3] http://www.sit.puglia.it
[4] https://geoportal.regione.liguria.it/
[5] https://www.copernicus.eu/
[6] https://www.soilgrids.org/

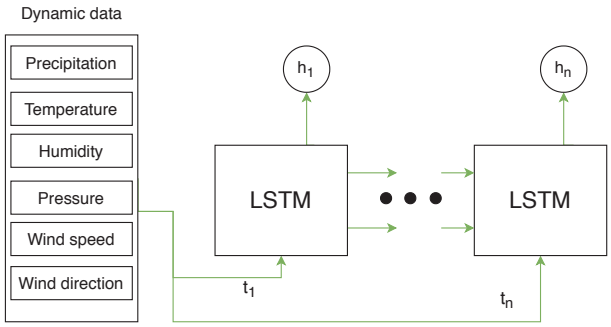

Fig. 1. RNN architecture that process dynamic data.

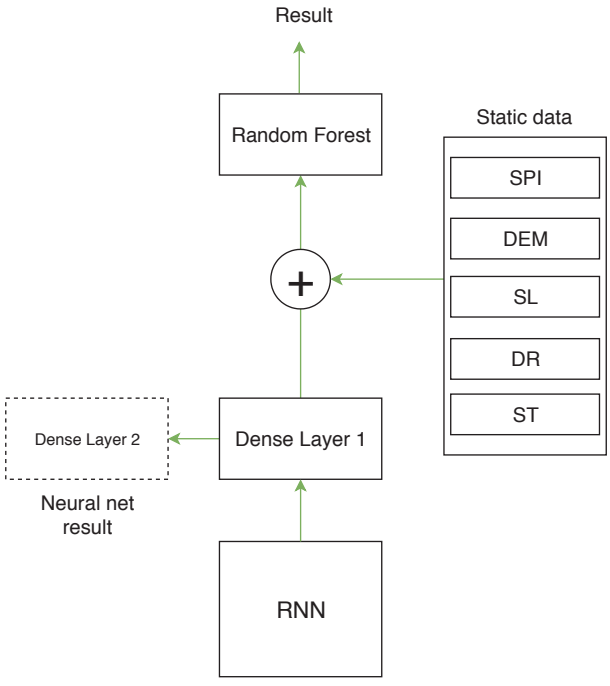

Fig. 2. Proposed architecture, concatenating is represented with + sign and is done after the initial training.

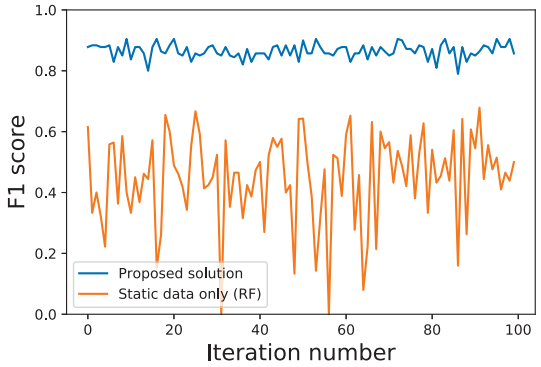

Fig. 3. F1 scores for each of experiments.

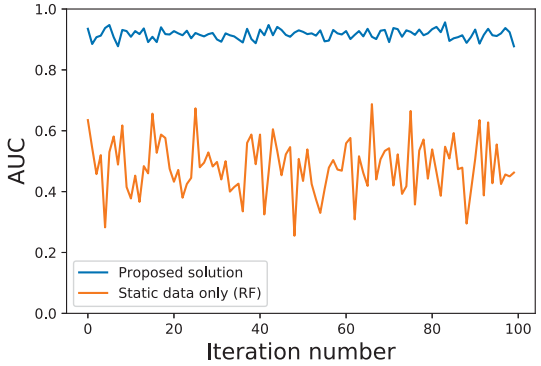

Fig. 4. AUC values for each of experiments.

V. RESULTS

Confusion matrix for the best experiment is shown in the Fig. 5. Best result shows 0.25 FPR and 0.95 TPR. F1 scores for all of the experiments are shown on the Fig. 3, while AUC values are shown on the Fig. 4. To evaluate the results, mean F1 score and mean AUC were calculated. The proposed model exhibits a mean F1 score of 0.864 while the mean AUC value is 0.916. Introducing RNN and dynamic data improves the performances reached by the first experiment, which employed only static data.

VI. MODEL COMPLEXITY

In order to test if other linear architectures could achieve similar results, we employed a simple linear logistic regression classifier (logit). Moreover, the experiment was repeated for five times in order to ensure statistical generalization. Similarly to the proposed architecture, the output of recurrent neural networks was concatenated with static data and the new dataset was provided as input to the Logistic Regression Classifier. In the Fig. 6 reports the F1 score for each iteration and each round; in the Fig. 7 shows the AUC values for the same experiments. The average value of the F1 score and AUC are 0.827 and 0.88, respec-

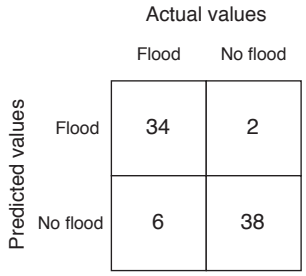

Fig. 5. Confusion matrix is represented by the best from 100 experiments.

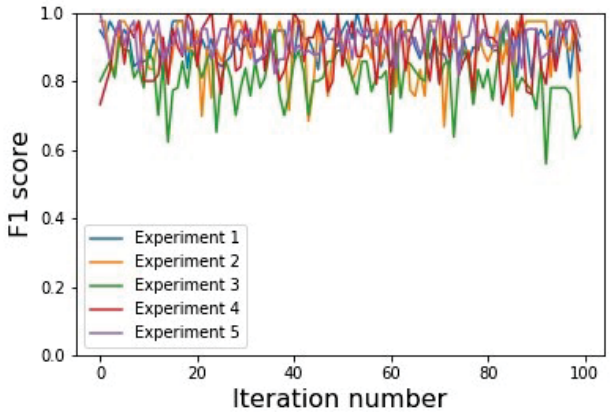

Fig. 6. F1 score for each of experiments.

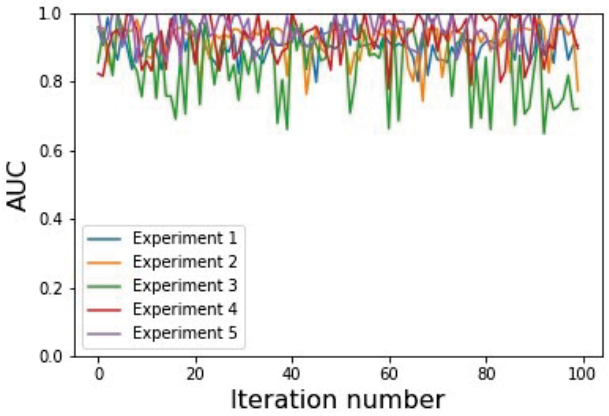

Fig. 7. AUC values for each of experiments.

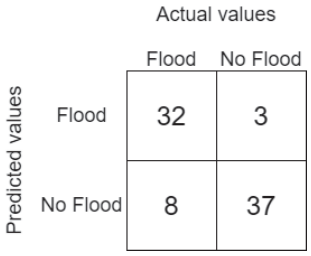

Fig. 8. Confusion matrix of the best experiment is represented.

tively. Finally, in Fig. 8 the confusion matrix of the best experiment is represented. It is worth to consider that the logistic regression algorithm slightly worsens the results compared to the previous performance.

VII. CONCLUSIONS AND FUTURE WORKS

The proposed solution aimed at performing a flood prediction combining RNN, used as a feature selector in the non-linear space, with Random Forest algorithm. The results of the research could be useful for the calculation of flood risk index for particularly exposed areas. Besides, since governments are responsible for providing reliable flood maps, this kind of solution can be employed as a valuable basis for generating flood maps and creating early warning flood systems. Still, the study presented has a significant false negative rate, on which research will have to focus on minimizing. Moreover, in order to get available data related to a given event, the proposed approach selected the nearest stations. This approach is prone to errors, as air distance can be misleading for area characterized by changes in terrain and sea level. As a future development, the employ of stations could be replaced by segmented maps produced by geologists and other experts in the field, in which each segment would represent a distinct area of the map with unique features. Events would then be geo-referenced to segments. Finally, a further improvement could be accomplished by using a larger sample, for what concerns both time frame and location considered.

Acknowledgment.

The present study was developed and granted in the framework of the project: "SeVaRA" (European Community, Minister of the Economic Development, Apulia Region, BURP n. 1883 of the 24/10/2018, Id:2NQR592).

REFERENCES

[1] AW Jayawardena and Feizhou Lai. Analysis and prediction of chaos in rainfall and stream flow time series. *Journal of hydrology*, 153(1-4):23–52, 1994.

[2] Michael I. Jordan. Attractor dynamics and parallelism in a connectionist sequential machine. In *Proceedings of the Eighth Annual Conference of the Cognitive Science Society*, pages 531–546. Hillsdale, NJ: Erlbaum, 1986.

[3] Leo Breiman. *Machine Learning*, 45(1):5–32, 2001.

[4] Longjun Dong, Xibing Li, and Gongnan Xie. Nonlinear methodologies for identifying seismic event and nuclear explosion using random forest, support vec-

tor machine, and naive bayes classification. In *Abstract and Applied Analysis*, volume 2014. Hindawi, 2014.

[5] Amir Mosavi, Pinar Ozturk, and Kwok wing Chau. Flood prediction using machine learning models: Literature review. *Water*, 10(11):1536, October 2018.

[6] Masoud Bakhtyari Kia, Saied Pirasteh, Biswajeet Pradhan, Ahmad Rodzi Mahmud, Wan Nor Azmin Sulaiman, and Abbas Moradi. An artificial neural network model for flood simulation using GIS: Johor river basin, malaysia. *Environmental Earth Sciences*, 67(1):251–264, December 2011.

[7] Zhaoli Wang, Chengguang Lai, Xiaohong Chen, Bing Yang, Shiwei Zhao, and Xiaoyan Bai. Flood hazard risk assessment model based on random forest. *Journal of Hydrology*, 527:1130–1141, August 2015.

[8] Patricia Jimeno-Sáez, Javier Senent-Aparicio, Julio Pérez-Sánchez, David Pulido-Velazquez, and José Cecilia. Estimation of instantaneous peak flow using machine-learning models and empirical formula in peninsular spain. *Water*, 9(5):347, May 2017.

[9] Abdusselam Altunkaynak and Tewodros Assefa Nigussie. Prediction of daily rainfall by a hybrid wavelet-season-neuro technique. *Journal of Hydrology*, 529:287–301, October 2015.

[10] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, November 1997.