Deep Learning for Urban Flood Prediction: An LSTM Model Integrating Satellite Reanalysis and Historical Weather Data in Curitiba

Lucas Iuri dos Santos[1], Luiz Gomes-Jr[1]

[1]UTFPR - Curitiba-PR - Brazil

lcjunior@utfpr.edu.br

Abstract. *Urban floods pose increasing threats, causing significant human and economic losses. This research focuses on improving flood prevention using deep learning techniques together with satellite reanalysis data. We present the results of the implementation of an LSTM Neural Network with satellite reanalysis data to predict floods in the city of Curitiba, Brazil. To assess the benefits of the proposed model, we compare the results with datasets from previous related models. The results of each trained model are analyzed based on Receiver Operating Characteristic (ROC) curves. The results show the positive impact of the use of satellite data with respect to the area under the curve (AUC) metric.*

1. Introduction

Brazil has been facing a surge in extreme climate events in the past few years. In 2024 there were 1690 disasters registered, with an economic loss of over U$ 6.4 billion (Maciel 2025). The annual average of registered flood disasters has doubled since 2020 and the main cause has been pointed to be the lack of investment in natural disaster prevention (Maciel 2025).

One possible way to minimize the impacts of these events is by the use of advanced technologies. Promising results have been obtained by the use of machine and deep learning techniques around the world (Al-Rawas et al. 2024). One such attempt is the ICARUS project for extreme event response in the city of Curitiba, Brazil. (Fernandez and Splendore 2021) propose a system to automatically identify hydrometeorological risks and their impacts on communication systems to provide rerouting options. (Noboa et al. 2024) uses feature engineering on climate, terrain and hydrological data to forecast flood events using Random Forest Classifiers.

This work is part of the ICARUS project and follows the line of flood forecasting by training an LSTM Recurrent Neural Network model with data from previous works and adding satellite data. The main goal here is to compare the results obtained from the LSTM architecture trained on the data from previous work alone and also with the addition of satellite data for daily flood occurrences for each region of interest.

The remainder of the paper is organized as follows: Section 2 presents some fundamental concepts and related work, Section 3 explains the methodology followed in the development of the model, Section 4 presents the results and Section 5 concludes the paper and discusses its limitations and future work.

2. Fundamentals and related works

2.1. Flood categories definitions

According to the United States Geological Survey (USGS), floods can be categorized in two types. *River flooding* occurs when the water level slowly rises in a large area due to excessive runoff from longer-lasting rainstorms or from snow melting. *Flash floods* are characterized by rapid rise in water height of a stream or a normally-dry channel.

The event of interest for this work is flash floods. In this work, the term “flood” refers specifically to “flash floods”. Brazilian authorities adopt a broader variety of events, but there is one category that fits the definition of flash floods (SEDEC 2024, in Portuguese), which are the events recorded in the available data.

2.2. Satellite data

There is a plethora of data available from satellites. One kind is the reanalysis data. The term reanalysis means the gathering of historical data and its processing through modern numerical weather prediction models to amplify the temporal and spatial coverage. One source of such data, used in this work, is the European Centre for Medium-Range Weather Forecasts (ECMF) Reanalysis v5 (ERA5) (Hersbach et al. 2020).

ERA5 maintains several datasets with particular characteristics. One of them is ERA-Land that provides hourly reanalysis data since 1950 for the entire Earth. One characteristic is the spatial resolution, that provides data in the form of a grid, in this case spaced in about 11 km, although it may suffer some distortions as it gets further from the equator. The Earth surface is divided in a grid of squares, where each region contains the average values of the variables for the entire square.

2.3. Recurrent Neural Networks and Long Short-term Memory

One of the main deep learning architectures used to handle sequences of information is the Recurrent Neural Networks (RNNs). Their difference to the regular neural networks is that its processing is made in steps, and in each step the output of the previous step, also called hidden state in this context, is fed as input to the current step. This way, previous information can influence the output.

The architecture of choice for this work, Long Short-term Memory (LSTM), is a type of RNN. Proposed by (Hochreiter 1997), an LSTM network is composed by layers of LSTM cells. An LSTM cell is represented in Figure 1 and its difference from the standard type is the addition of a forget gate, which combines the previous hidden state of the cell, represented by the function $h(t-1)$, with the new information from the new data input $x(t)$ and the previous cell state $c(t-1)$, to select what is important to retain. The function $h(t)$ can be interpreted as the short-memory of the cell because it is the output of the previous step of processing. The cell state $c(t)$ can be interpreted as the long-term memory of the cell because it is constantly being updated by the input and forget gates, accumulating information from earlier steps. The input gate (a combination of the inputs $x(t)$ and $h(t-1)$) selects what data should be added to the cell state $c(t)$. The output gate is responsible for deciding which parts of the cell state, previous hidden state and inputs should be combined to the cell output and hidden state. The LSTM architecture keeps relevant information for much longer sequences than regular RNNs.

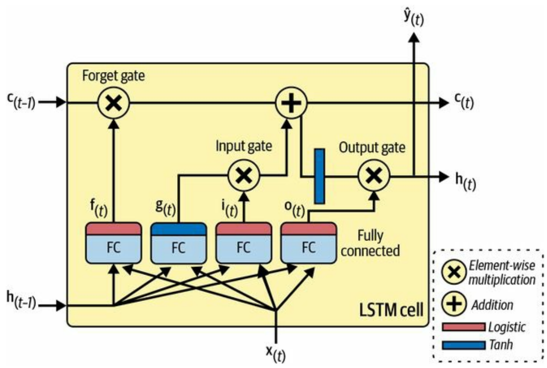


Figure 1. LSTM cell. Source: (Géron 2022)

The LSTM architecture was chosen because of the results achieved in similar scenarios of disaster early-warning prediction with climate and satellite data (Al-Rawas et al. 2024). In the study, LSTM was placed among the best strategies compared to other machine learning algorithms, although the number of studies that adopted it were low because of the variety of other architectures.

2.4. Related works

A number of related works were considered in the choices made for the model and data in this work. Similar to the objectives here, the work of (Fang et al. 2021) uses an LSTM architecture with satellite data to predict hydrological disasters in Shangyou, China. The work of (Le et al. 2019) uses LSTM networks to predict floods in Da River, Vietnam, but it uses weather data only. Another important factor is the geographic context, and considering this, the work of (de Sousa Araújo et al. 2022) uses LSTM networks to predict extreme rainfalls in southeastern Brazil, close to the region of Curitiba. Also in the Brazilian scenario, (Batalini de Macedo et al. 2024) compares the results of LSTM networks to physical models for flood prediction in São Paulo, Brazil and Narmada, India.

The *Integrated Crisis Awareness and Resource Utilization for Smartcities* (ICARUS) is a system to prevent disasters and mitigate its impacts in urban scenarios. (Fernandez and Splendore 2021) initially proposed a system to identify natural disasters through outlier detection techniques and simulation of the network infrastructure in order to keep connectivity during critical events. The next step for ICARUS was proposed by (Noboa et al. 2024) and started the prediction of flood events. It used Random Forest classifiers on data from rain gauges, altimetry, and information of river and lakes in the city of Curitiba. The main goal was to use feature engineering to identify the most important features to predict floods. This paper builds upon their results, motivated by their discoveries, and employing a modern deep learning architecture to develop a complementary forecasting model.

3. Methodology

3.1. Data acquisition and processing

The data utilized included the same climate data from (Noboa et al. 2024), namely rain gauge obtained from (CEMADEN 2024) and altimetry and hydrological characteristics obtained from (IPPUC 2022). The data are aggregated as hexagonal $600m^2$ geographic regions for the period of January 1st, 2015 to December 31st, 2017. The flood records were obtained from the city's Civil Defense and cover a larger period from January 1st, 2009 to December 31st, 2018 containing the date and location of the event.

The ICARUS project utilizes novel satellite data obtained from the Google Earth API to predict floods. This data includes precipitation, surface runoff, humidity (two soil layers), and surface pressure, all sourced from ERA5 and corresponding to periods of flood records. While ERA5 provides hourly data, the project's objective is daily flood prediction, so the data was aggregated: precipitation by sum and other variables by mean. A key limitation is the 11 km spatial resolution of the satellite data (Hersbach et al. 2020). This resolution is too coarse for the original granularity of flood prediction, leading to repeated values across hexagonal grid cells. To address this, the city of Curitiba was divided into nine regions based on these repeated values, as seen in Figure 2. Although this regionalization is generally satisfactory, some misclassified hexagons remain. After processing and aggregation, each row of the dataset contained the date, the region ID and the values for each feature.

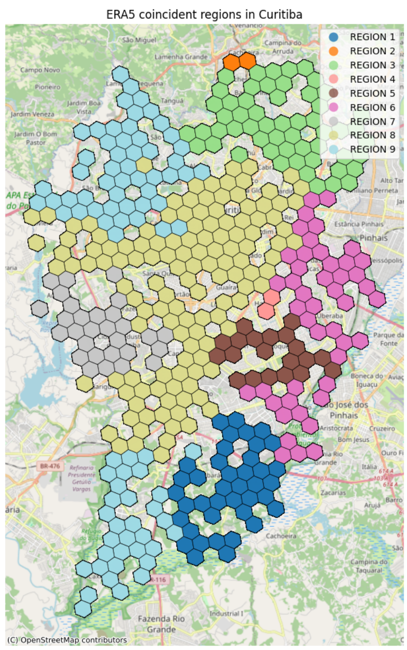


Figure 2. Map of Curitiba split by regions.

3.2. Modeling

The network architecture contains three layers of LSTM, each followed by a batch normalization layer to improve training speed and stability, followed by a dense layer with a sigmoid activation function to predict the flood probability. The structure, with its inputs and output shapes, can be visualized in Table 1. Notice that the input shape considers a look-back window of 30 days and 5 features for this example.

The data was split in train and test following the time series nature of the data (the last 20% of the period covered was used for testing). The flood records are heavily imbalanced, so it was necessary to calculate the weights according to the Equation 1 for flood events and absences to be used in training. By assigning a higher weight to the

Layer	Input shape	Output shape
LSTM (1)	(None, 30, 5)	(None, 30, 8)
Batch Normalization (1)	(None, 30, 8)	(None, 30, 8)
LSTM (2)	(None, 30, 8)	(None, 30, 8)
Batch Normalization (2)	(None, 30, 8)	(None, 30, 8)
LSTM (3)	(None, 30, 8)	(None, 30, 8)
Batch Normalization (3)	(None, 30, 8)	(None, 30, 8)
Dense	(None, 8)	(None, 1)

Table 1. LSTM structure

minority class in training, the loss function is modified, and the network is more penalized for misclassifying objects of the underrepresented class.

$$w_{class} = \frac{1}{n_{class}} * \frac{n_{total}}{SF} \tag{1}$$

In Equation 1, w_{class} is the weight of the class, n_{class} is the number of records of the class, n_{total} is the total number of records, and SF is the scale factor. In essence, this equation adjusts the weights according to the inverse of the frequency of the classes, and the scale factor is for controlling the magnitude of the weights. The scale factor in this case is 2, the number of classes.

3.3. Training and evaluation

For the training, the batch size is 32 and the number of epochs is 50. The learning rate is 0.05 and the λ parameter for L2 regularization is 0.03. These values were chosen because of preliminary experiments. The model has two callbacks, both very common in similar scenarios, one for adjusting the learning rate to half of its value if the loss does not improve for one epoch. The other callback implements its early stopping, which stops training if the precision of the model does not improve for 30 epochs, restoring the weights for the best epoch. Because of limitations in schedule and computing power associated with this work, the hyperparameters were set in a trial-and-error effort with no separate optimization step.

Since the objective is to predict flood occurrences, it is a binary classifier. Because of the nature of the classification, the results will be evaluated according to their Receiver Operating Characteristic (ROC) curve and their associated area under the curve (AUC).

This metric is more appropriate than the more common metrics like precision and recall because it associates all probability classification thresholds to the rate of false positives and true positives. Because floods are critical events, authorities may want to adjust the threshold to get a higher true positive rate, and the ROC curve can support this decision while offering a good comparison of trained models through the AUC.

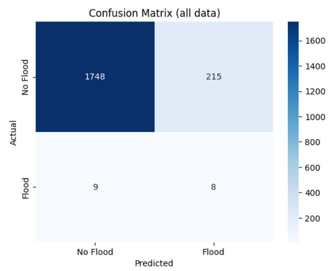
4. Results

The AUC results obtained for the same combinations of features of (Noboa et al. 2024) are summarized in Table 2. Note that these results were obtained using the same data for training and testing, which covered the period of January 1st, 2015 to December 31st, 2017.

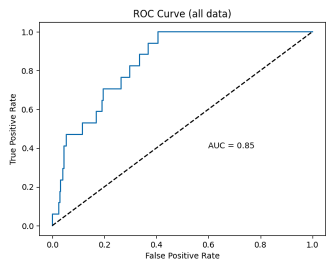
Features	Without satellite data	With satellite data
Precipitation	0.80	0.82
Altimetry	0.47	0.78
Hydrological	0.50	0.84
Precipitation + altimetry	0.80	0.83
Precipitation + altimetry + hydrological	0.81	**0.85**
Satellite only	-	0.84

Table 2. Summary of results (AUC)

The best results were obtained using all features available. Its weighted average *f-score* is 0.92, and the confusion matrix and the ROC curve can be seen on Figure 3. The ROC curve shows a steep increase in accuracy as the level of accepted false positives is increased. This suggests that the model is applicable to the extreme events scenario, where some false positives are acceptable since the cost of false negatives are too high.



(a) Confusion matrix.



(b) ROC curve and AUC.

Figure 3. Results for the model with all features.

5. Conclusion

The main objective of this work was to propose an improved model for flood forecasting in the city of Curitiba. The direct comparison with previous models from the ICARUS project is not possible because of the different nature of the prediction here, given the area limitations of the satellite data. However, it can be clearly seen that adding satellite data positively impacted the predictions for all the sets of features.

Despite the satisfactory results, this work had some limitations. The most clear is the spatial resolution of the ERA5 data, which imposed the limitation to predictions in only 9 regions. Another major limitation is the lack of hyperparameter tuning. IAs for the dataset, the flood records are recorded manually based on call center contacts. The manual registration is error-prone and may diverge from the real scenarios. Another limitation is the short period covered by the combination of data, which resulted in few flood occurrences, probably interfering with the training of the model.

Given the results and limitations, there are some directions that can be followed in future work. For instance, different satellite data with better spatial resolution can be used to improve the area precision of the predictions, or provide more features from satellites to improve the performance of the model. Other recent advances in deep learning architectures, like *transformers*, can also be tested. Finally, this work could be replicated for other regions or disasters of a different nature.

References

[Al-Rawas et al. 2024] Al-Rawas, G., Nikoo, M. R., Al-Wardy, M., and Etri, T. (2024). A critical review of emerging technologies for flash flood prediction: examining artificial intelligence, machine learning, internet of things, cloud computing, and robotics techniques. *Water*, 16(14):2069.

[Batalini de Macedo et al. 2024] Batalini de Macedo, M., Mangukiya, N. K., Fava, M. C., Sharma, A., Fray da Silva, R., Agarwal, A., Razzolini, M. T., Mendiondo, E. M., Goel, N. K., Kurian, M., et al. (2024). Performance analysis of physically-based (hec-ras, caddies) and ai-based (lstm) flood models for two case studies. *Proceedings of IAHS*, 386:41–46.

[CEMADEN 2024] CEMADEN (2024). Mapa Interativo — mapainterativo.cemaden.gov.br. `https://mapainterativo.cemaden.gov.br/#`. Accessed: 2024-12-16.

[de Sousa Araújo et al. 2022] de Sousa Araújo, A., Silva, A. R., and Zárate, L. E. (2022). Extreme precipitation prediction based on neural network model–a case study for southeastern brazil. *Journal of Hydrology*, 606:127454.

[Fang et al. 2021] Fang, Z., Wang, Y., Peng, L., and Hong, H. (2021). Predicting flood susceptibility using lstm neural networks. *Journal of Hydrology*, 594:125734.

[Fernandez and Splendore 2021] Fernandez, H. G. and Splendore, P. R. (2021). Sistema de Identificação Automática de Riscos Hidrometeorológicos com Retroalimentação e Reestruturação Autônoma da Infraestrutura de Comunicação.

[Géron 2022] Géron, A. (2022). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. " O'Reilly Media, Inc.".

[Hersbach et al. 2020] Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horányi, A., Muñoz-Sabater, J., Nicolas, J., Peubey, C., Radu, R., Schepers, D., et al. (2020). The era5 global reanalysis. *Quarterly journal of the royal meteorological society*, 146(730):1999–2049.

[Hochreiter 1997] Hochreiter, S. (1997). Long short-term memory. *Neural Computation MIT-Press*.

[IPPUC 2022] IPPUC (2022). Registros alagamentos. Accessed: 2022-11-07.

[Le et al. 2019] Le, X.-H., Ho, H. V., Lee, G., and Jung, S. (2019). Application of long short-term memory (lstm) neural network for flood forecasting. *Water*, 11(7):1387.

[Maciel 2025] Maciel, E. (2025). Brazil faces huge surge in climate disasters amid poor prevention funding. `https://www.developmentaid.org/news-stream/post/193539/brazil-faces-surge-in-climate-disasters`. Accessed: 2025-06-07.

[Noboa et al. 2024] Noboa, C. S., Pigatto, D., Buffon, E. M., and Gomes-Jr, L. (2024). Data analytics for a changing climate: Feature engineering for the forecast of hydrometeorological events. In *Simpósio Brasileiro de Banco de Dados (SBBD)*, pages 715–721. SBC.

[SEDEC 2024] SEDEC (2024). Classificação e Codificação Brasileira de Desastres (COBRADE). `https://www.gov.br/mdr/pt-br/centrais-de-conteudo/publicacoes/protecao-e-defesa-civil-sedec/DOCU_cobrade2.pdf`. Accessed: 2025-06-07.

[USGS] USGS. What are the two main types of floods? `https://www.usgs.gov/faqs/what-are-two-types-floods`. Accessed: 2025-06-07.