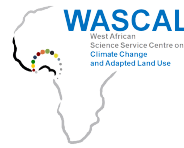


UNIVERSITE JOSEPH KI-ZERBO

\_\_\_\_\_  
ECOLE DOCTORALE INFORMATIQUE  
ET CHANGEMENTS CLIMATIQUES



BURKINA FASO

*Unité-Progress-Justice*

**MASTER RESEARCH PROGRAM**

**SPECIALITY: INFORMATICS FOR CLIMATE CHANGE (ICC)**

**MASTER THESIS**

**Subject:**

**Predicting discharge in catchment outlet using deep learning  
methods: case study of Niamey in the Ansongo-Niamey basin**

by

**Julien Yise Peniel Adoukpe**

**Major Supervisor:**

Prof. Eric Adechina Alamou

**Co-Supervisor:**

Dr. Belko Abdoul Aziz Diallo

**Jury Members:**

**Internship supervisor:**

Dr. Abdou Ali

**Academic year: 2020 - 2021**

## Summary

Summary . . . . .	i
Dedication . . . . .	ii
Acknowledgments . . . . .	iii
Abstract . . . . .	iv
Acronyms and Abbreviations . . . . .	v
List of Tables . . . . .	vi
List of Figures . . . . .	vii
<b>1 Introduction</b>	<b>1</b>
1.1 Context . . . . .	2
1.2 Justification of Study . . . . .	3
1.3 Literature review . . . . .	5
1.4 Objectives . . . . .	21
1.5 Host Structure . . . . .	22
<b>2 Methodology</b>	<b>24</b>
2.1 Study Area . . . . .	25
2.2 Data . . . . .	35
2.3 Tools . . . . .	37
2.4 Methods . . . . .	38
<b>3 Results and Discussion</b>	<b>48</b>
3.1 Results . . . . .	49
3.2 Discussion . . . . .	55
<b>4 Conclusion and Perspectives</b>	<b>56</b>
<b>Bibliography</b>	<b>59</b>
<b>Appendix</b>	<b>I</b>
<b>Table of Contents</b>	<b>V</b>

## **Dedication**

I dedicate this work to my beloved mother, Victoire Adoukpe.

## Acknowledgments

The student wishes to express his sincerest gratitude and warm appreciation to the following persons and organizations who had to assist in the making of this valuable piece of work.

- West African Science Service on Climate Change and Adapted Land Use (WASCAL) for providing this wonderful framework of capacity building.
- German Ministry of Education and Research (BMBF) for funding this program.
- Doctorate School of Informatics and Climate Change (ED-ICC) for explicitly organizing all related activities of the master's program in Informatics and Climate Change.
  - Prof. Tanga Pierre Zoungrana, the Director of ED-ICC,
  - Dr. Ousmane Coulibaly, the Deputy Director of ED-ICC,
  - and to ED-ICC staff members
- Prof. Alamou Eric, my major supervisor, for supporting my work with his extensive knowledge of hydrology
- Dr. Belko Abdoul Aziz Diallo, my co-supervisor, for showing interest since the early stages of this work
- Dr. Abdou Ali, my internship supervisor and the head of the information and research department at the AGRHYMET Regional Center. He showed great interest in my work and provided me with all the required support during my internship. A special thanks for the Dr. Ali's team at AGRHYMET for supporting this research. This team is made of the following people: Bernard Minoungou, Dr. Charlene Gaba, Dr. Gadedjisso-Tossou Agossou, Henri Songoti, Mandela Hounnibo, Mohamed Hamatan, Valentin Iogo, ...
- Didier Zinsou, the Director of the Niger Basin Authority (NBA) Observatory, for he welcoming me with open arms. He also introduced me to Bachir Tanimoun, NBA's expert hydrologist.
- Prof. Adamou Rabani, the Director of WASCAL Niger, for being so welcoming and finding accommodation at Higher Institute of Mining, Industry and Geology (EMIG) during my period of internship.
- Thanks to my colleagues and friend with whom I had a wonderful experience during the 2 years of the program.
- Last but not least, Prof. Julien Adoukpe, my dad for showing me support, advice and love.

## Abstract

Hydrological models are one of the key challenges in hydrology. Their goal is to understand, predict and manage water resources. Most of the hydrological models so far were physically based and conceptual models. But in the past two decades, fully data-driven (empirical) models started to emerge with the breakthroughs of novel deep learning methods in runoff prediction. The breakthrough was mostly favored by the large volume, variety and velocity of water-related data. Long Short-Term Memory networks (LSTMs), particularly achieved the outstanding milestone of outperforming classic hydrological models in less than a decade. Moreover, they have the potential to change the way hydrological modeling is performed. In this study, we used precipitation, maximum and minimal temperature at the Ansongo-Niamey basin mixed with the discharge at Ansongo and Kandadji to predict the discharge at Niamey using LSTM neural networks. After data preprocessing and hyperparameter optimization, the LSTM model performed well with a  $R^2$  of 0.897, a NSE of 0.852 and a RMSE of 229.158. This performance matches those of well-known physically based models used to simulate Niamey's discharge and therefore demonstrates the efficiency of deep learning methods in a West African context.

Keywords: Ansongo-Niamey basin; hydrological modeling; Long Short-Term Memory (LSTM); Artificial Neural Networks.

## Résumé

Les modèles hydrologiques constituent l'un des principaux défis de l'hydrologie. Leur objectif est de comprendre, de prévoir et de gérer les ressources en eau. Jusqu'à présent, la plupart des modèles hydrologiques étaient des modèles à base physique et conceptuels. Mais au cours des deux dernières décennies, des modèles entièrement basés sur des données (empiriques) ont commencé à émerger avec les percées des nouvelles méthodes d'apprentissage profond dans la prédiction du ruissellement. Les réseaux de neurones disposant de mémoire à long et à court terme (LSTM), en particulier, ont réussi à surpasser les modèles hydrologiques classiques en moins d'une décennie. En outre, ils ont le potentiel de changer la façon dont la modélisation hydrologique est effectuée. Dans cette étude, nous avons utilisé les précipitations, la température maximale et minimale du bassin Ansongo-Niamey associées avec le débit à Ansongo et Kandadji pour prédire le débit à Niamey en utilisant les réseaux neurones LSTM. Après le prétraitement de données et l'optimisation des hyperparamètres, le modèle LSTM a bien fonctionné avec un  $R^2$  de 0,897, un NSE de 0,852 et un RMSE de 229,158. Cette performance rivalise avec celle d'autres modèles à base physique utilisés pour simuler le débit de Niamey. Ceci démontre donc l'efficacité des méthodes d'apprentissage profond dans un contexte ouest africain.

Mots-clés : bassin Ansongo-Niamey ; modélisation hydrologique ; LSTM ; réseau de neurones artificiels.

## Acronyms and Abbreviations

AI	Artificial Intelligence
ANN	Artificial Neural Networks
ARC	AGRHYMET Regional Center
BGS	British Geological Survey
BMBF	German Ministry of Education and Research
CGIAR-CSI	CGIAR Consortium for Spatial Information
CILSS	Permanent Interstate Committee for Drought Control in the Sahel
DCW	Digital Chart of the World
DL	Deep Learning
ED-ICC	Doctorate School of Informatics and Climate Change
EMIG	Higher Institute of Mining, Industry and Geology
ESA	European Space Agency
FEWS	Flood Early Warning System
FFNN	Feed Forward Neural Network
GADM	Database of Global Administrative Areas
GRU	Gated Recurrent Unit
ML	Machine Learning
NSE	Nash-Sutcliffe Efficiency
NBA	Niger Basin Authority
NN	Neural Networks
LSTM	Long Short-Term Memory Neural Networks
RMSE	Root Mean Square Error
RNN	Recurrent Neural Networks
USGS	United States Geological Survey
WASCAL	West African Science Service on Climate Change and Adapted Land Use

## **List of Tables**

1.1	Classic vs machine learning program in a chess game . . . . .	7
1.2	Characteristics and classification of hydrological model . . . . .	17
1.3	Performance of LSTM for river flow prediction . . . . .	20
2.1	Data used for objectives . . . . .	35
2.2	Additional data used to describe study area . . . . .	36
2.3	Optimized hyperparameters using Scikit-Optimize . . . . .	42
4.1	Descriptions of aquifer type categories . . . . .	II
4.2	Descriptions of aquifer productivity categories . . . . .	II
4.3	Hyperparameters values tuned with Scikit-Optimize . . . . .	IV

## List of Figures

1.1	Chess playing machine: the Turk . . . . .	5
1.2	Key moments in the AI journey . . . . .	6
1.3	Classic vs machine learning programs . . . . .	7
1.4	Subsets of artificial intelligence . . . . .	7
1.5	Structure of a biological neuron . . . . .	8
1.6	Structure of an artificial neuron . . . . .	9
1.7	Simple perceptron . . . . .	9
1.8	Examples of linear separability . . . . .	11
1.9	Boolean functions and linear separability . . . . .	11
1.10	XOR logic implementation with feed forward network . . . . .	12
1.11	Sigmoid activation function . . . . .	13
1.12	Hyperbolic tangent activation function . . . . .	13
1.13	Computational graph of loss function . . . . .	14
1.14	Structure of a vanilla recurrent neural network . . . . .	15
1.15	Gradient flow in vanilla RNN cells . . . . .	16
1.16	Evolution of neural networks for river forecasting in publications and citation from 1992 to 2010 . . . . .	18
2.1	Map of Ansongo-Niamey basin . . . . .	27
2.2	Administrative map of Ansongo-Niamey basin . . . . .	28
2.3	Hydrographs of Ansongo, Kandadji and Niamey for June 1991 to Mai 1992 . . . . .	29
2.4	Hydrology of Ansongo-Niamey basin . . . . .	30
2.5	Monthly average of the temporal distribution of precipitation, maximum temperature and minimum temperature in the Ansongo-Niamey basin from June 1981 to December 2010 . . . . .	31
2.6	Average of the spatial distribution of precipitation, maximum temperature and minimum temperature in the Ansongo-Niamey basin from June 1981 to December 2010 . . . . .	32
2.7	Geology of Ansongo-Niamey basin . . . . .	33
2.8	Hydrogeology of Ansongo-Niamey basin . . . . .	34
2.9	Summary of the main methodology . . . . .	39
2.10	Process of catchment delineation in QGIS . . . . .	40
2.11	Hyperparameter optimization process . . . . .	42
2.12	Structure of a long short-term memory neural network cell . . . . .	43
2.13	Gradient flow in LSTM cells . . . . .	44
2.14	Different subdivision of the dataset . . . . .	45
3.1	Convergence plot . . . . .	49
3.2	Objective plot . . . . .	50
3.3	Evaluation plot . . . . .	51
3.4	LSTM model summary . . . . .	52
3.5	Evaluation of the LSTM model . . . . .	52



*LIST OF FIGURES*

---

3.6	Predictions of the LSTM model from June 2006 to December 2010 . . . . .	53
3.7	Predictions of the LSTM model with type of flood . . . . .	54
4.1	Missing data at Ansongo . . . . .	III
4.2	Missing data at Kandadji . . . . .	III

# **Chapter 1**

## **Introduction**

## 1.1 Context

“In recent decades, changes in climate have caused impacts on natural and human systems on all continents and across the oceans. In many regions, changing precipitation or melting snow and ice are altering hydrological systems, affecting water resources in terms of quantity and quality.” [1]. In the Niger Basin (particularly in the Middle Niger Basin), extensive catastrophic flooding has increased drastically in the last two decades [2].

In Niamey, the capital of Niger, the latest flood of August 23 2020 had the highest river discharge ever recorded since the installation of its hydrological station with a river discharge of 2.716 m<sup>3</sup>/s (corresponding height of 6.40 m). That series of floods in Niger affected approximately 15,981 people, destroyed 868 houses washed away 77 animals and 7 destroyed granaries [3].

However, various structural (levees, reservoirs, diversion channels, and spillways) and non-structural (flood early warning systems, land-use planning and zoning, rainwater harvesting, flood insurance schemes, and awareness campaigns) flood mitigation measures are implemented to minimize flood disasters. Both types of measures can reduce casualties and economic losses, and therefore are critical for international disaster risk reduction efforts.

Global focus on disaster risk reduction has significantly increased since 2000 when the millennium declaration called on the global community to say intensify our collective efforts to reduce the number and effects of natural and man-made disasters [4]. Also, in 2015, 10 of the 17 sustainable development goals are related to disaster risk reduction. 25 targets are also related to disaster risk reduction with the main ones being: targets 11.5 and 11.b that aim at reducing the number of deaths, people affected, and economic losses caused by disasters by 2030; target 13.1 that aims to strengthen the integration between disaster and climate resilience to protect broader development paths and target 9.1 that focuses on the development of disaster-resilient infrastructure.

Given the frequency and the extent of flooding that the city of Niamey has been experiencing over the years, finding means to achieve those Millennium Declaration’s targets will significantly alleviate the suffering of the population. Flood Early Warning System (FEWS) plays a major role in global DRR efforts.

FEWS is defined as “an integrated system of hazard monitoring, forecasting and prediction, disaster risk assessment, communication, and preparedness activities systems and processes that enable individuals, communities, governments, businesses, and others to take timely actions to reduce disaster risks in advance of hazardous events” [5].

Hydrological models are the main components involved in forecasting and prediction systems of FEWS. Many hydrological models have been deployed to predict the discharge at Niamey. All of those models happened to be physically based hydrological models such as HYPE, ISBA–TRIP, SWAT and HGS [6–9]. AI based model are fully data driven models capable to overcome deficiencies of physically based hydrological models [10]. This research would explore the potential of a new generation of empirical model.

## 1.2 Justification of Study

Artificial Intelligence (AI) is perceived to be the technology of the future because of its profound impact on industries, and livelihoods by providing new ways of solving problems [11]. AI, also called machine intelligence, refers to the stimulation of human intelligence in machines programmed to think like humans and mimic their actions. Among the technologies of AI are machine learning, deep learning, natural language processing, and computer vision.

Deep Learning (DL) is a subset of machine learning in AI that uses layers of representation of data for problem-solving. DL has networks capable of unsupervised learning, data processing, and pattern recognition for use in decision-making. It could be used for diverse applications such as cancer diagnosis, precision medicine, self-driving cars, speech recognition and in our case predictive forecasting [12].

“Deep learning neural networks are able to automatically learn arbitrary complex mappings from inputs to outputs and support multiple inputs and outputs. These are powerful features that offer a lot of promise for time series forecasting, particularly on problems with complex-nonlinear dependencies, multivalent inputs, and multi-step forecasting.” [13]

In hydrology, DL is now well known for predicting and estimating floods, monitoring drought accurately, analyzing atmospheric imaging, estimating tropical cyclones and their precursors, and the prediction of hydrological processes [14]. Long Short-Term Memory (LSTM) is an innovative DL method for the prediction of time series flooding and of high accuracy as well as reducing the cost of CPU which increases the sustainability of this method significantly [15]. LSTMs are a special type of Recurrent Neural Networks (RNNs) with an internal memory that can learn and store long-term dependencies of the input-output relationship.

A DL-based FEWS method is being suggested to use in this work. DL mainly is reported to be essential for achieving higher accuracy, robustness, efficiency, computation cost, and overall model performance [10]. During recent years, AI methods (machine learning and DL) have indeed become very popular among the research communities for modeling and predicting hydrological processes, climate change, and earth systems [14]. DL is seen as the answers to the new challenges in water science [10].

- The interdisciplinarity and human dynamics challenge: water scientists’ societal missions now mandate us to account for not only the movement and transformations of the water cycle, but also how it reciprocates and interacts with other subsystems such as biogeochemistry, climate, vegetation dynamics, biotic and abiotic soil environments, and human dynamics including socioeconomics
- The data deluge and the data discoverability challenge: new sources of information should be bound to enrich our understanding and empower our modeling capabilities. However, data discoverability is an issue. Extracting comprehensible yet hidden information from the unprecedented volume of data or even devising approaches to extract such information can be daunting tasks. Conventional information retrieval approaches often require extensive human labor and domain expertise because they often

require case-by-case calibration and adjustment. More importantly, without advanced techniques, we may not even realize what kind of hidden and abstract information can be extracted nor recognize the limits of accuracy of such extraction, which results in the under-utilization of available data. These practical aspects greatly limit their use in providing inputs and constraints to models.

- Unrecognized/un-represented linkages: models could produce errors if important processes and relationships are not included, due to either knowledge or capability limitations. In some cases, we have a crude idea of what processes/variables might be important and have some data at hand, but implementing the corresponding model in an integrated modeling system might be too time-consuming, yet the relationships between these variables are complex and do not manifest themselves when they are simply plotted against each other. Some data-driven mechanisms to explore potential sophisticated linkages between variables would enable rapid hypothesis testing.
- The model scaling and equifinality challenges: these are long-standing modeling issues that have recently been shunned rather than addressed. It has long been known that process-based models' results can be scale-dependent. Equifinality (competing models and parameter sets that can generate similar outcomes) is another long-standing issue.
- The need for regionalized parameters and models: with traditional data mining methods, data-driven hydrologic studies are often forced to regionalize or bin their datasets and create specialized models that fit those particular regions or bins. This need also applies to the calibration of conceptual hydrologic models. Typical reasons for regionalization include a lack of knowledge of the underlying process or influential latent variable. A result of regionalization is that in each region the available data points are reduced and thus have less statistical power to condition the models.

Out of the DL methods to predict floods, the LSTM networks stand out of the bunch. LSTM is a long- and short-memory artificial neural network constituted of a recurrent neural network (RNN) structure that learns directly from time-series data and is proposed by Hochreiter and Schmidhuber in 1997 [16] (See also literature review). Since the neural network contains a time memory unit, it is suitable for processing and predicting the interval and delay events in the time series [17].

Thus, this study will focus on the component forecasting and prediction of river flow in FEWS by using LSTM NN. The study will be done on the Ansongo-Niamey basin with Ansongo (Mali) and Niamey (Niger) being respectively the upstream and downstream of the Niger river.

This research would focus on the following questions:

- Why using DL methods for hydrological modeling in the first place?
- Why the choice of LSTM Neural Networks (NN) among all DL methods?
- How to apply a LSTM NN to predict river in catchment outlet, particularly in a West African setting?

## 1.3 Literature review

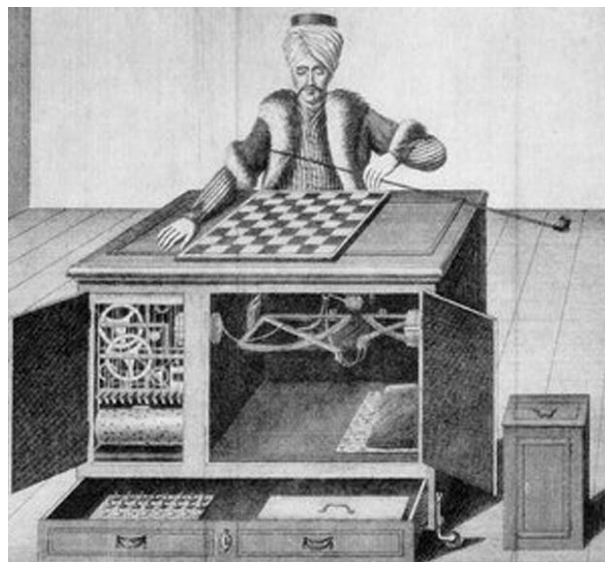
The literature review of this work would focus on explaining why LSTM networks are one of the best DL methods for predicting time series. This review would start with a short history of AI. Then, the core concepts of the major evolutionary stages of Artificial Neural Networks (ANN) would be overlooked and their limitations would also be highlighted. At the end of this review, the prowess of LSTM NN in hydrological modeling would be presented.

### 1.3.1 Short history on AI

“The beginnings of artificial intelligence are traced to philosophy, fiction, and imagination. Early inventions in electronics, engineering, and many other disciplines have influenced AI.” [18]

Philosophers describe intelligent machines as literary devices that could help us understand mankind. Science fiction writers enhanced this idea by extending the potential of intelligent non-humans and by giving the unique and distinct characteristics of humans.

During the 18th and 19th centuries, one of the most remarkable uses of AI started to develop: the chess-playing machine. Chess was widely used in the early stages of AI for establishing inferences and representation mechanisms. This even gave birth to the Turk, also known as the Mechanical Turk or Automaton Chess Player that appeared to be able to play a strong game of chess against a human opponent. Unfortunately, **the Turk** was a mechanical illusion where a human chess master was hidden inside the machine to operate it [19].



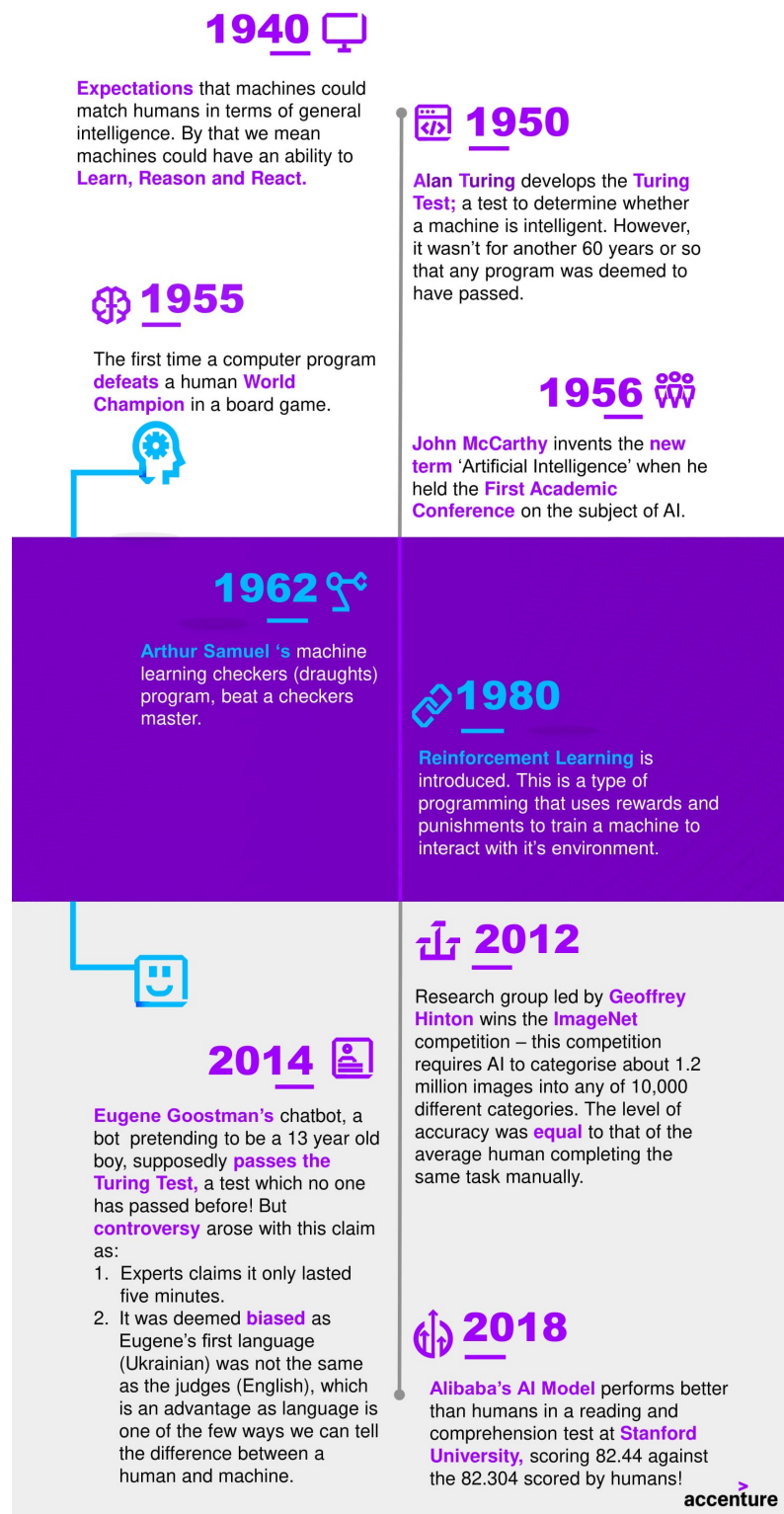
**Figure 1.1:** Chess playing machine: the Turk

Source: From an engraving by Freiherr Joseph Friedrich zu Racknitz (1789)

In the early 19th century, as a result of inventions in electronics, computers were born and their awesome calculating power gave a huge boost to the AI domain! Soon, computational devices and programming languages were developed to build experimental tests of ideas about what intelligence is.

For instance, it was one century later (in 1955) that the chess-playing machines lived up to their expectations when the **Logic Theorist**, the first AI program was created. By the end of the 20th century (1997), **Deep Blue**, a chess-playing computer developed by IBM, was the first computer to win both a chess game and a chess match against a reigning world champion under regular time controls. See in the figure 1.2 for other major milestones accomplished by AI.

Although AI did see significant growth throughout the 20th century, AI between 1974 and 1980 faced disappointment after failing to deliver on its hype. This period is referred to as “**AI winter**” and occurred mainly because computers were just not powerful enough yet to store the information needed to communicate intelligently [21]



**Figure 1.2:** Key moments in the AI journey

Source: FutureLearn [20]

## 1.3.2 Artificial Intelligence - Machine Learning - Deep Learning

Before we move on, let's clarify some basic concepts around AI and how AI systems perform tasks. In fact, for a computer system to perform a task that usually requires human intelligence, AI systems must be equipped with knowledge and experience in two ways.

- You can program each instruction so that the machines solve the task step by step this is comparable to a cooking recipe or assembly instructions alternatively.
- You can use programs that learn from data themselves this enables them to detect relevant information draw conclusions or make predictions this is known as Machine Learning (ML).

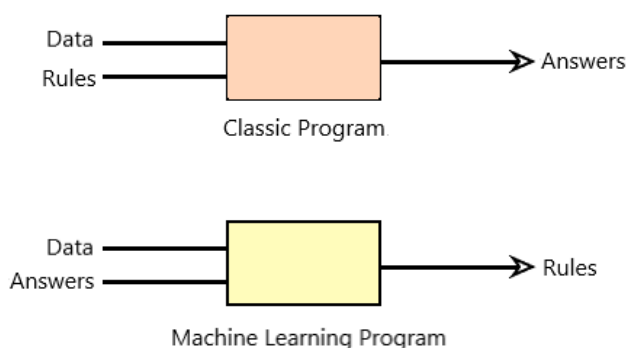
Let us come back to the recurrent example of the chess game. If you want a computer system to play chess, you could either write up codes of the behavior of each piece and their interactions with each other or use ML algorithms to play chess by experiences. Table 1.1 shows the pros and cons of each method.

**Table 1.1:** Classic vs machine learning program in a chess game

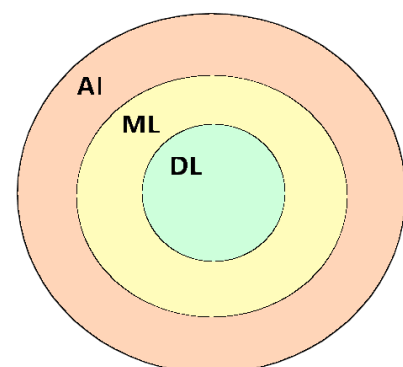
Method	Pros	Cons
Classic program	start out by wining	only as good as programmer
Machine learning	improve after each game played	start out by losing

In general, classic programs could be seen as a combination of input data and a set of rules to get answers while ML programs would rather combine input data with their corresponding answers to get a set of rules. See figure 1.3 right bellow.

Deep Learning is a subset of Machine Learning based on Artificial Neural Networks (ANN) that uses a layered representation of data. Figure 1.4 shows the idea that we should have when talking about AI and its subsets.



**Figure 1.3:** Classic vs machine learning programs



**Figure 1.4:** Subsets of artificial intelligence



The main reasons for choosing DL methods over ML methods is because [22]:

- DL **outperforms** traditional ML methods by a huge margin in several domains, especially in the fields of computer vision, speech recognition, natural language processing and in our case time series,
- with DL, more and more complex features can be learned as the layers in the DL NN increase. Due to this automatic feature-learning property, DL **reduces the feature-engineering time**, which is a time-consuming activity in traditional ML approaches and
- DL works best for **unstructured data** (images, text, speech, sensor data, ...).

Meanwhile, high performance comes at a high cost:

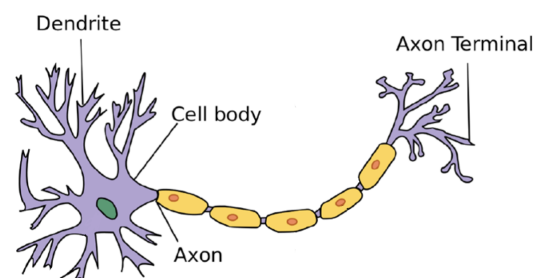
- DL ANNs generally tend to have a lot of parameters, and for such implementations, there should be a **sufficiently large volume of data** to train. If there are not enough data, DL approaches will not work well since the model will suffer from overfitting.
- The complex features learned by the DL network are often **hard to interpret**.
- DL networks require a **lot of computational power** to train because of the large number of weights in the model as well as the data volume.

### 1.3.3 Inspiration behind Artificial Neural Networks

The goal of ANNs are to mimic human behavior, therefore, ANNs are directly inspired by biological learning systems.

In a **biological neuron**:

- the **dendrites** receive input signals from hundreds of neighboring neurons,
- the **cell body** receives the previously collected signals and sums them up and
- the **axon** receives the summed up signal and releases a new signal if the summed up signal is greater than a specific threshold. The new signal will then be passed into the neighboring dendrites of other neurons.

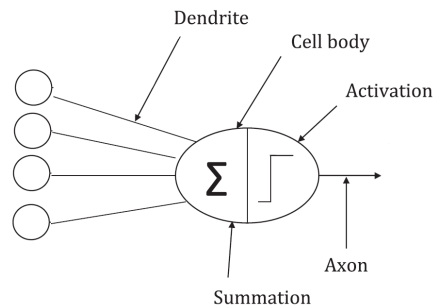


**Figure 1.5:** Structure of a biological neuron

Source: Santanu Pattanayak [22]

In an **artificial neuron**:

- the **input connections** receive input signals from other neighboring neurons,
- the **neuron cell** sums up the input information and releases a new signal after the signal passes through an activation function and
- the **output connection** receives and transfers the output signals to the next layer of artificial neurons.



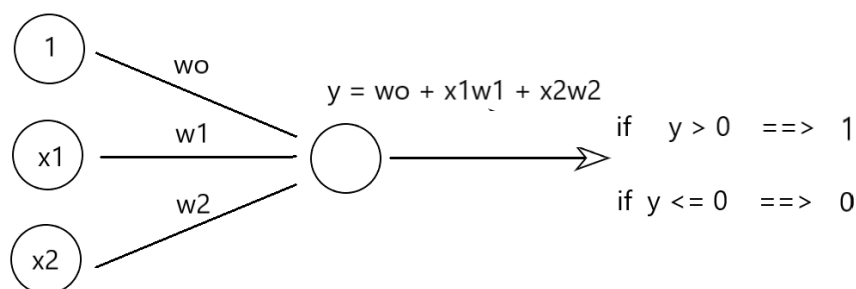
**Figure 1.6:** Structure of an artificial neuron

Even though artificial neurons do not operate exactly as biological neurons, they do have basic similarities.

### 1.3.4 Perceptron

#### Concept

Perceptrons are the most basic form of ANN. To understand ANNs, knowing the mechanisms of perceptrons is the best way to start. A perceptron is a linear binary classifier that uses a hyperplane to separate the output data into 2 classes. Its input layers are directly linked to its output layer. Figure 1.7 shows a perceptron made up of 2 inputs and one binary output.



**Figure 1.7:** Simple perceptron

$$y = f(w, x) = (w_1 \ w_2) \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + w_0 \quad (1.1)$$

Equation 1.1 is the matrix form of the equation expressed in the figure 1.7.

The output layer  $y$  is the weighted sum of the inputs plus a bias, see equation 1.2.  $y$  fires the value of 1 if  $y$  is greater than 0 and fires 0 if  $y$  is smaller or equal to 0.

$$y = \begin{cases} 1, & \text{if } \sum_{i=1}^n w_i x_i + b > 0 \\ 0, & \text{otherwise} \end{cases} \quad (1.2)$$

Equation 1.2 is a general function of a perceptron.

The weights  $w_1$  and  $w_2$  and the bias  $w_0$  are parameters of the perceptron. Another parameter is the learning rate  $\eta$  of the perceptron that controls the speed at which the perceptron learns. Be careful to not make it too big or too small, the learning rate should be around  $10^{-1}$  and  $10^{-6}$  depending on the ANN structure.

The weights are randomly initialized and the learning rate is set. The equation 1.2 becomes an equation of a line in the form  $y = ax + b$  which represents the equation of the hyperplane with  $x$  and  $y$  being the inputs  $x_1$  and  $x_2$ . The error  $e$  of the classification algorithm is calculated by the difference between the desired output  $d$  and the model output  $y$  (equation 1.3). The weights are updated at each iteration through gradient descent with the new weights being  $w'_i$  (equation 1.4).

$$e = \sum_{i=1}^n d_i - y_i \quad (1.3)$$

$$w'_i = w_i + \eta * e * x_i \quad (1.4)$$

This series of instructions is known as backpropagation. The goal of the backpropagation training algorithm is to modify the weights of a neural network to minimize the error of the network outputs compared to some expected output in response to corresponding inputs. It is a supervised learning algorithm that allows the network to be corrected concerning the specific errors made [23].

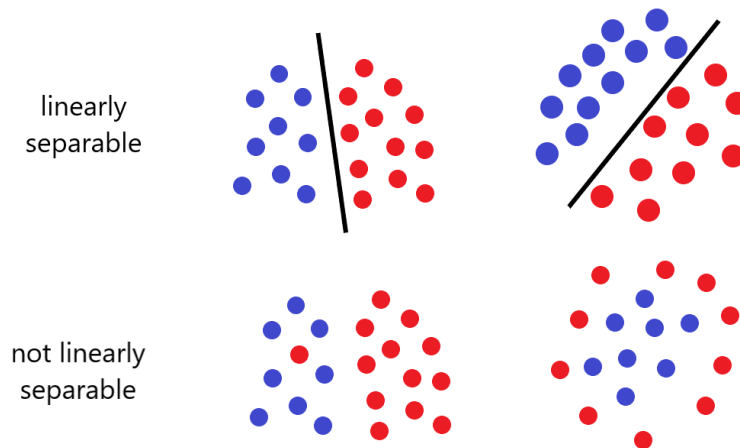
The general algorithm of backpropagation is as follows:

1. Present a training input pattern and propagate it through the network to get an output.
2. Compare the predicted outputs to the expected outputs and calculate the error.
3. Calculate the derivatives of the error with respect to the network weights.
4. Adjust the weights to minimize the error.
5. Repeat.

The goal of the perceptron is to adjust the weights and bias to obtain the minimum classification error. All ANNs have this basic principle in common.

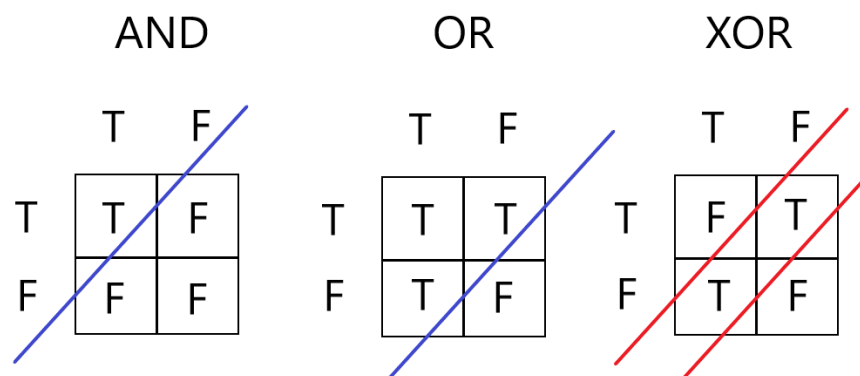
## Limits

The limit of perceptrons is that they can only solve linearly separable problems while in real-world data is full of non-linearity [24].



**Figure 1.8:** Examples of linear separability

Another simple example is that a perceptron could only solve AND and OR logic of the Boolean function but not the XOR logic. Meanwhile, XOR is simply the addition of the opposite result of AND plus OR:  $XOR = !AND + OR$



**Figure 1.9:** Boolean functions and linear separability

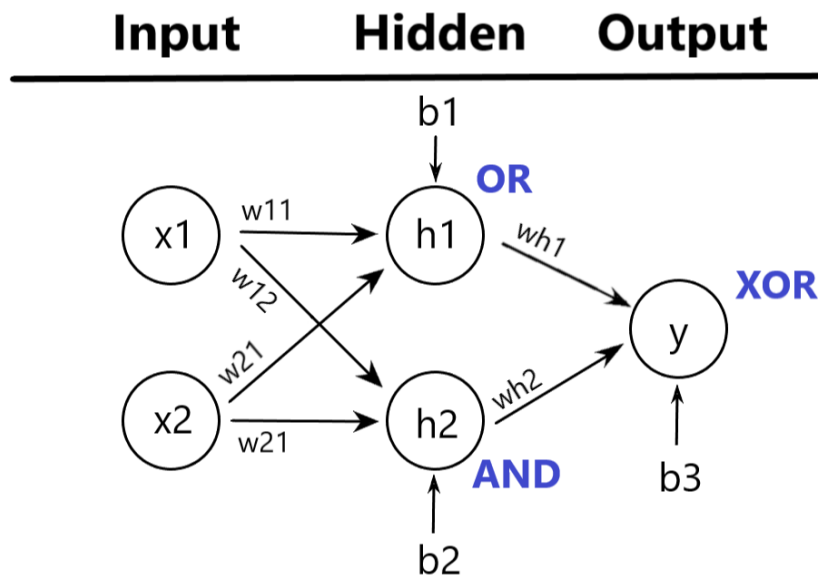
This is what introduced the combination of perceptrons capable of solving nonlinear problems. That solution was the feed forward neural network.

### 1.3.5 Feed Forward Neural Network

#### Concept

A Feed Forward Neural Network (FFNN) is a class of NN that possesses multiple layers of computational units. Connected in a feed-forward way, each neuron of one layer is fully connected to the neurons of the subsequent layer. The FFNN differs from its predecessor due to its hidden layers that interconnect the input and the output layers.

To implement the XOR logic of the Boolean function, the two perceptrons of the hidden layer would each be performing one logic. One would be performing the OR logic while the second one would be performing the AND logic. In this NN, the nonlinearity is solved by adding more neurons to the system.



**Figure 1.10:** XOR logic implementation with feed forward network

This FFNN is a 2 layer NN or a 1 hidden layer NN. The term vanilla is referred to a ANN with one hidden layer. we could also call it a vanilla FFNN.

$$f^{IH}(w^{IH}, x) = \sigma \left( \begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} \right) \quad (1.5)$$

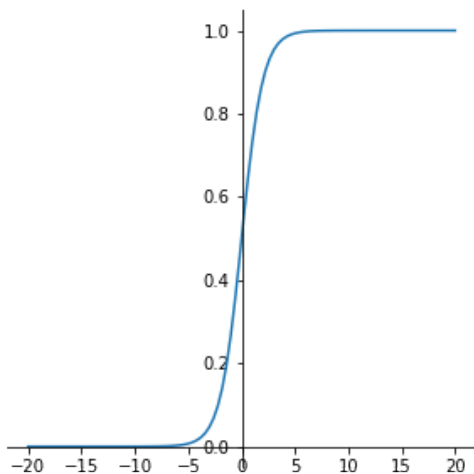
$$f^{HO}(w^{HO}, h) = \sigma \left( \begin{pmatrix} w_{h1} & w_{h2} \end{pmatrix} \begin{pmatrix} x_{h1} \\ x_{h2} \end{pmatrix} + b_3 \right) \quad (1.6)$$

$$\begin{aligned} H &= \sigma * (w_{ij}^{IH} * I + B^H) \\ 0 &= \sigma * (w_{ij}^{HO} * H + B^O) \end{aligned} \quad (1.7)$$

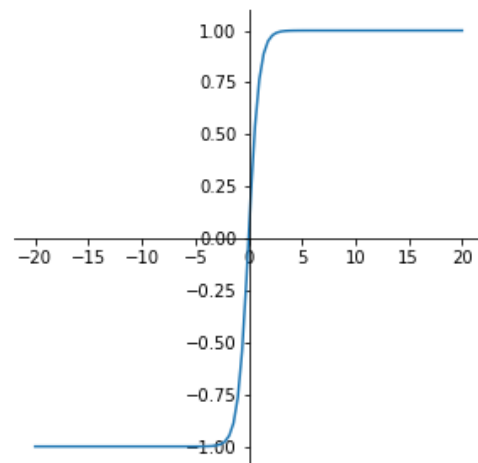
The equations 1.5 and 1.6 are the matrix equations of the figure 1.10. 1.5 represents the output of the hidden layer and 1.6 represents the output of the output layer. The equations 1.5 are general equations of a vanilla FFNN with 1 hidden layer.

Due to the added complexity to the network of neurons, there would be additional mechanisms to make it sustainable and robust. The principle mechanisms we would care about are the activation functions and the loss function.

In the equations above, you notice a sigma ( $\sigma$ ) term added to the function.  $\sigma$  is a sigmoid function, it acts as an activation function. An activation function in a NN defines how the weighted sum of the input is transformed into an output from a node or nodes in a layer of the network, they decide whether the neuron should be activated or not. They could also be called squashing functions. For the following demonstrations, it would be useful to have 2 types of activation functions in mind: the sigmoid function  $\sigma$  (logistic function) and the hyperbolic tangent function  $\tanh$ .



**Figure 1.11:** Sigmoid activation function



**Figure 1.12:** Hyperbolic tangent activation function

The sigmoid function takes any real value as input and outputs the value in the range of 0 and 1. The hyperbolic tangent function is very similar to the sigmoid function, the only difference is that it outputs the value in the range of -1 and 1.

The following are the equations of the activation functions.

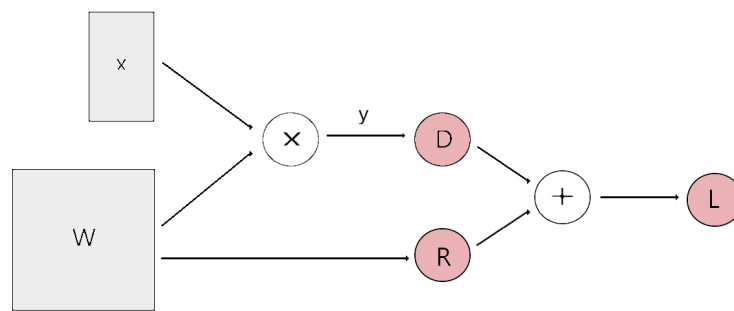
$$\begin{aligned}\sigma(x) &= \frac{1}{1 + e^{-x}} \\ \tanh(x) &= \frac{e^x - e^{-x}}{e^x + e^{-x}}\end{aligned}\tag{1.8}$$

Now that we have clarified the activation function we can now go into some detail about the loss function.

If we consider a function that's the product of two matrices, with one matrix being the weight and the second matrix being the input. The equation of this matrix would be:

$$y = f(W, x) = Wx \quad (1.9)$$

In deep learning, computational graphs are one of the best ways to express and evaluate mathematical expressions [25]. A computational graph is defined as a directed graph where the nodes correspond to mathematical operations. The figure below shows the computational graph of the loss function.



**Figure 1.13:** Computational graph of loss function

A loss function tells how good our model is. In the perceptron subsection, we only considered the data loss term calculating the loss function. There is a regularization term that is added to the data loss. The regularization is controlled by the regularization strength  $\lambda$  by considering the following equation:

$$L(W, x) = D_i(W, x) + \lambda R(W) \quad (1.10)$$

There could be many combinations of  $W$  to get a minimum loss, the regularization would select the weight matrix with the “simplest” structure. Regularization is a technique for preventing over-fitting by penalizing a model for having large weights. This helps the model to work better on test data. The greater  $\lambda$  is, the greater the penalty would be.

## Limits

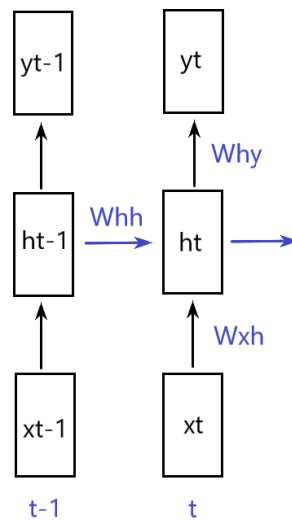
Although FFNNs are quite remarkable and are used for many applications, they are not suitable for **time dependent** variables. FFNNs have “lower” memory to retain information over timesteps compared to other ANNs such as LSTMs [26]. The main cause of this is that the hidden units of the ANN are not interconnected to each other.

### 1.3.6 Recurrent Neural Networks

#### Concept

Recurrent Neural Networks (RNNs) are seen as the go-to solution when it comes to sequence data such as time series. The difference between RNNs and FFNNs is that RNNs have interconnected hidden units which boost their performance.

The following figure shows an example of a vanilla RNN.



**Figure 1.14:** Structure of a vanilla recurrent neural network

In this figure, there is a weight  $W_{hh}$  connecting the previous hidden state  $h_{t-1}$  to the new hidden state  $h_t$ . There also weights  $W_{xh}$  connecting the input unit to the hidden unit and  $W_{hy}$  connecting the hidden unit to the output unit. The equations below are the mathematical expression figure 1.14.

$$\begin{aligned} h_t &= f_w(h_{t-1}, x_t) \\ &= \tanh \left( (w_{hh} \quad w_{hx}) \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix} \right) \end{aligned} \quad (1.11)$$

$$h_t = \tanh \left( W \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix} \right)$$

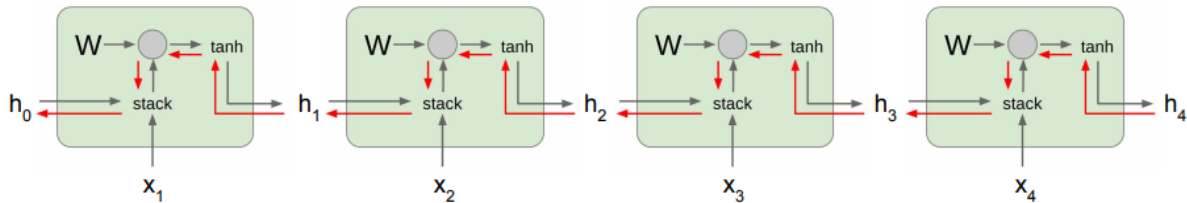
$$y_t = W_{hy} h_t \quad (1.12)$$

From the equation 1.11, we could notice that the weights  $W_{xh}$  and  $W_{hh}$  are stacked up into a big matrix.



## Limits

The drawback of RNNs is that during backpropagation from  $h_t$  to  $h_{t-1}$ , the gradient is multiplied by the weight matrix (its transpose). This is repeated over each timesteps 1.15. The back-propagated error signals tend to either grow (exploding gradients) or shrink (vanishing gradients) with every timestep. Standard RNNs cannot bridge more than 10 timesteps [24].



**Figure 1.15:** Gradient flow in vanilla RNN cells

Source: Fei-Fei et al. [25]

In some works [27, 28], people go through hacks to avoid his problems and have more effective RNNs:

- for the exploding gradients, they clip the norm of the exploded gradients and
- for the vanishing gradients, they use a regularization term that forces the error signal not to vanish as it travels back in time.

Meanwhile, we would not use these methods to improve the performance of the DL model. We would rather move to the most promising and effective RNN structure for time series prediction.

## 1.3.7 Long Short-Term Memory Neural Networks

LSTMs are certainly not perfect RNN, but they stand out by their ability to learn from time series by properly managing their gradient flow. This reason justifies the choice of LSTMs among all other ANNs (see the concept behind the LSTM model in the methods). During our research, we also encountered the Gated Recurrent Unit (GRU) which is an upgrade of the LSTM. Hence, the GRU performs slightly better than LSTM. However, for this work, we would rather consider LSTMs because it is better documented.

### Applications of LSTM NN in streamflow prediction

A hydrologic model simulates a flux, flow, or change of water storage with time within one or more components of the natural hydrologic cycle. Before getting to the applications of LSTMs in hydrology, let's overview the existing hydrological (rainfall-runoff) models and group them by category.

“Rainfall-runoff models are classified based on model input, parameters and the extent of physical principles” [29]. The table below summarizes the 3 main categories of hydrological models and some of their characteristics. For Gayathro et al., ANN models are empirical models with a low explanatory depth of hydrological systems. Meanwhile, their high predictive power makes them useful to hydrologists. This is why during the last three decades, the use of ANN in rainfall-runoff and stream flow modeling has increased significantly.

**Table 1.2:** Characteristics and classification of hydrological model

Empirical model	Conceptual model	Physically based model
Data based or metric or black box model	Parametric or grey box model	Mechanistic or white box model
Involve mathematical equations Derive value from available time series	Based on modeling of reservoirs Include semi empirical equations with a physical basis	Based on spatial distribution Evaluation of parameters describing physical characteristics
Little consideration of features and processes of system	Parameters are derived from field data and calibration	Require data about initial state of model and morphology of catchment
High predictive power Low explanatory depth	Simple and can be easily implemented in computer code	Complex model Require human expertise and computation capability
ANN, Unit hydrograph	HBV model, TOPMODEL	SHE or MIKESHE model, SWAT

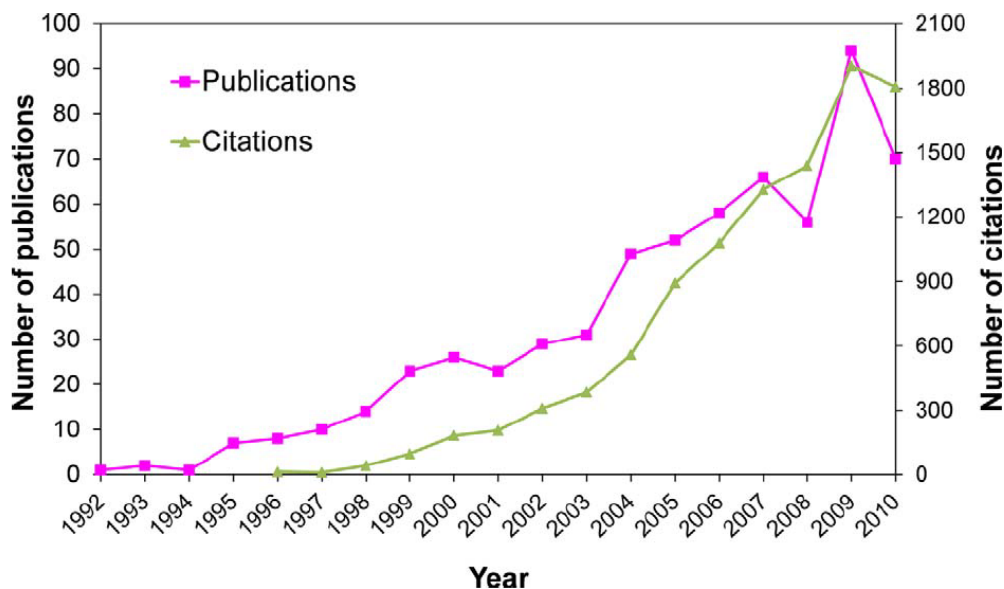
Source: Gayathri K. Devii et al. [29]

In 2012, Abrahart et al. reviewed NN river forecasting publications and citations [30]. The results of these publications are in the figure below. This assessment of broad trends in the popularity of NN from 1992 to 2010 was done by identifying existing publications of relevance to this field from a publication database (See figure 1.16).

LSTM NN among all ANN has gained popularity in the hydrology scientific community over the recent years as well. They are now seen as the go-to DL technique when it comes to predicting hydrological time series. Here are some work interesting works that used LSTMs to predict river flow.

Youchuan Hu et al. [31] forecasted the stream-flow downstream of the Tunxi catchment (China). They used data from 11 precipitation stations in the catchment and the streamflow volume data at the Tunxi station from 1981 to 2003 (23 years) with a 6-hour timestep. They also compared LSTM to other machine learning models (Support Vector Regression SVR and FFNN) and found out that the LSTM model provides more stable and more accurate predictions.

Hamidreza Ghasemi et al. [32] predicted the streamflow outlet of the Brazos river basin (Texas, USA). They used as daily input data: precipitation, 6 climate variability indices and streamflow from 2007 to 2010. The results presented were the prominent capacity of deep learning (LSTM) for an accurate prediction of discharges, and of course, an efficient alternative to the conventional hydrological models. The LSTM also outperformed the CaMa-Flood (Catchment-based physically explicit Macro-scale Floodplain) hydrological model.



**Figure 1.16:** Evolution of neural networks for river forecasting in publications and citation from 1992 to 2010

Source: Abrahart et al. [30]

Bibhuti Sahoo et al. [33] used LSTM NN to forecast low-flow hydrological time series in the Mahanadi river basin (India). They are referred to as “low-flow”  $Q_{75}$  percentile of discharge in the stream. The only input data used in the LSTM NN model was the monthly discharge data at Basantapur (outlet of catchment) from June 1971 to May 2010.

Many other articles prove the efficiency of LSTM NN in streamflow prediction but from my point of view, Frederik Kratzert’s research is one of the most inspiring of them all. Not only did he produced fabulous articles in the application of LSTM in rainfall-runoff modeling but he also shared all the codes used for his work in this GitHub repository. In 2018, Frederik et al. [34] used LSTM NN to predict discharge of 241 catchments of the United States using daily 5 meteorological parameters (precipitation, shortwave downward radiation, maximum and minimum temperature and humidity) and discharge at catchments outlet from 1980 to 2015. Within three experiments, the researchers explored possible applications of LSTMs and demonstrated that LSTMs can simulate the runoff with competitive performance compared to the SAC-SMA + Snow-17 model (Snow-17 model coupled with the Sacramento Soil Moisture Accounting Model). In the first experiment they looked at classical single basin modeling, in a second experiment they trained one regional model for all basins, and in a third experiment, they

showed that using a pre-trained model helps to increase the model performance in single basins. Additionally, they showed an illustrative example of why traditional RNNs should be avoided in favor of LSTMs. This study underlined that the LSTM models also performed slightly better than compared to the well-established SAC-SMA + Snow-17 model. Moreover, better performance of the LSTM models could have been achieved by exhaustive (catchment-wise) hyperparameter search. Another exciting result of this work was that pre-trained knowledge of LSTM models can be transferred into different catchments, which might be a possible approach for reducing the data demand and/or regionalization applications, as well as for prediction in ungauged basins or basins with few observations. In 2019, Frederik et al. [35] explored the former perspective in the article “towards learning universal, regional, and local hydrological behaviors via machine learning applied to large-sample datasets”. This time around, they used 531 basins using daily meteorological time series data (cumulative precipitation, minimum air temperature, maximum air temperature, average short-wave radiation and vapor pressure) and 27 static catchment characteristics were used as input features. They were able to significantly improve performance compared to a set of several different hydrological benchmark models (SAC-SMA + Snow-17 model and National Water Model NWM). The proposed approach not only significantly outperforms hydrological models that were calibrated regionally but also achieves better performance than hydrological models that were calibrated for each basin individually. Furthermore, they propose an adaption to the standard LSTM architecture, which we call an Entity-Aware-LSTM (EA-LSTM), that allows for learning catchment similarities as a feature layer in a deep learning model. They showed that these learned catchment similarities correspond well to what we would expect from prior hydrological understanding.

**Table 1.3:** Performance of LSTM for river flow prediction

Authors and year	Nash–Sutcliffe efficiency
Youchuan Hu et al., 2020	0.97 LSTM 0.96 FFNN 0.92 SVR
Hamidreza Ghasemi et al., 2019	0.89 LSTM -1.13 CaMa-Flood
Bibhuti Sahoo et al., 2019	0.88 LSTM 0.84 RNN 0.7 Naïve
Frederiket al., 2018	Mean over all catchments 0.63 LSTM 0.58 SAC-SMA + Snow-17
Frederiket al., 2019	Mean over all catchments 0.69 LSTM 0.74 EA-LSTM 0.64 SAC-SMA + Snow-17 0.58 NWM

These works clearly show the potential of LSTM in predicting streamflow. LSTM models perform as well as, if not better than, physically-based hydrological models. In addition, LSTM models could be capable of associating physical characteristics of catchments for model regionalization.

## 1.4 Objectives

### 1.4.1 Hypothesis

- **H1:** The learning rate, the number of LSTM units, the number of epochs and the batch size of the LSTM model are the model's main hyperparameters.
- **H2:** Precipitation, maximum temperature, minimum temperature and discharge at Ansongo and Kandadji are sufficient inputs of the LSTM model for the effective simulation of Niamey's discharge.
- **H3:** LSTM model match and could even outperform classical hydrological models at predicting historical river flow at Niamey.


### 1.4.2 Main Objective

Simulate efficiently the Niger river discharge at the city of Niamey using the deep learning method: long short-term memory networks.

### 1.4.3 Specific Objectives

- **SO1:** Select the best hyperparameters for the LSTM model
- **SO2:** Calibrate, validate and test the LSTM model
- **SO3:** Discuss the LSTM model's performance by comparing it to previous works on river flow simulation at Niamey

## 1.5 Host Structure

<u>Acronym:</u>	ARC	
<u>Full Name:</u>	AGRHYMET Regional Center	
<u>Website:</u>	<a href="http://agrhymet.cilss.int/">http://agrhymet.cilss.int/</a>	
<u>Established:</u>	1974	
<u>Location:</u>	Niamey, Niger	
<u>Areas of expertise:</u>	Agricultural development, rural development and natural resources management	
<u>Mandate:</u>	To inform and train on food security, the combat against desertification and water control in the Sahel and West Africa	

### Short Description:

Created in 1974, the AGRHYMET Regional Center is a specialized institution of the Permanent Interstate Committee for Drought Control in the Sahel (CILSS), which includes thirteen countries: Benin, Burkina Faso, Cape Verde, Chad, Côte d'Ivoire, Gambia, Guinea, Guinea-Bissau, Mali, Mauritania, Niger, Senegal and Togo. It is an interstate public institution with legal personality and financial autonomy. It has an international status with headquarters in Niamey, Niger.

ARC is a tool with a regional vocation, specialized in sciences and techniques applicable to the sectors of agricultural development, rural space planning and natural resources management.

### Main Objectives:

- Contribute to food security and increase agricultural production in CILSS and Economic Community of West African States (ECOWAS) member countries.
- Help improve the management of natural resources in the Sahel and West Africa region by providing information and training to development actors and their partners in the fields of agro-ecology in the broadest sense (agro-climatology, hydrology, plant protection, ...).

### Area of Expertise:

- Train executives from the Sahel countries and elsewhere.
- Monitor agrometeorological and hydrological at the regional level.
- Monitor agricultural statistics and crop.
- Regional data banks.
- Manage and disseminate information on the monitoring of natural resources in the Sahel.
- Stores documentation on: agrometeorology, plant protection, environmental monitoring, desertification, natural resources management, ...

- Maintain meteorological instruments and electronic equipment,
- Strengthen interstate cooperation through the exchange of methodology and technology.

Departments:

The AGRHYMET Regional Center has three departments: Information and Research Department, Training and Research Department and Technical Support Department.

The Information and Research Department is a technical department is mandated for the collection, analysis and management of biophysical and socio-economic data on the one hand, and the transfer of methodological tools, production and dissemination of information on food security, markets, water control, natural resources management and the environment on the other hand. The head of this department is Dr. Abdou Ali.

The Training and Research Department is mandated to strengthen the capacities of CILSS member countries in areas related to food security, water control, natural resource management and the fight against desertification through basic training (diploma courses) and continuing education. The head of this department is Prof. Sanoussi Atta.

The Technical Support Department is mandated to support and assist the other entities of the CRA in the field of computer science, methods development, applications, databases, computer and infrastructure maintenance, network management, telecommunications and promotion of the field. The head of this department is Henri Songoti.

Reference: See article on AGRHYMET [36]



## **Chapter 2**

### **Methodology**

This chapter is made up of 4 sections: study area, data, tools and methods. First of all, there will be a presentation of the Ansongo-Niamey basin and its main features. Secondly, the data used in this research would be mentioned. After that, all the tools: programming language, interfaces, environments and libraries and other software used would be pointed out. This chapter would end with a detailed description of the steps followed to reach the objectives of this thesis.

## 2.1 Study Area

A good presentation of the study area is fundamental to all work related to environmental sciences and this thesis is no exception. Thus, this section is fully dedicated to understanding the most important environmental processes (human activities, hydrology, climatology, geology and hydrogeology) in the Ansongo-Niamey basin.

### 2.1.1 Ansongo-Niamey Basin

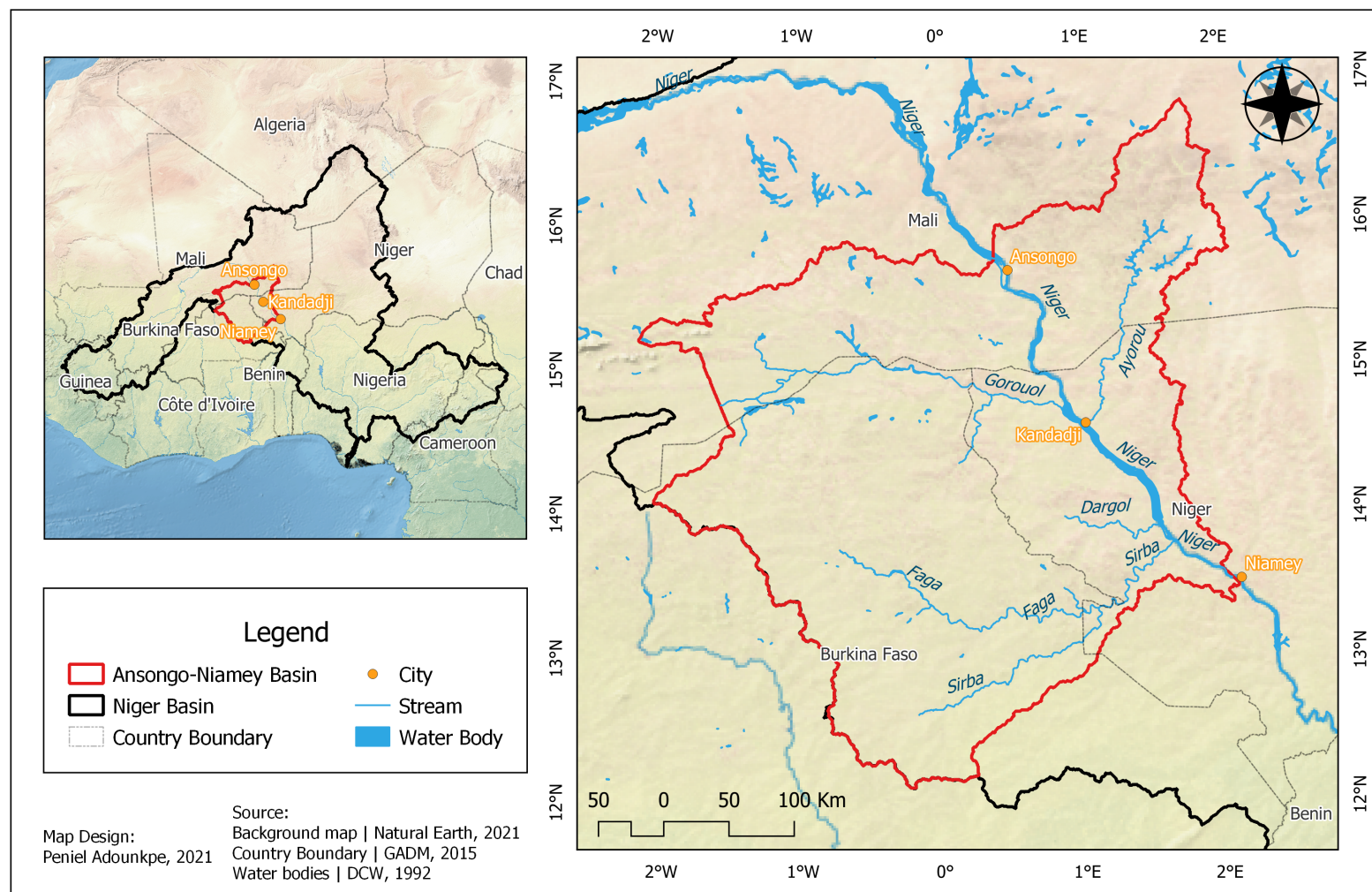
The Ansongo-Niamey basin is a catchment located in the middle region of the Niger basin between the cities of Ansongo (Mali) and Niamey (Niger). This basin was chosen as study area because it would be challenging to collect data in the upstream of the Niger basin: the upper Niger basin. Regarding that, Ansongo being the downstream of the upper delta was considered as the upstream of our study area and Niamey the outlet of the basin.

The Niger basin was named after the Niger river, the most significant river of West Africa (3rd of Africa after the Congo and the Nil rivers). The Niger basin has an area of around 2 111 473 km<sup>2</sup> which is spread out over 10 countries (Algeria, Benin, Burkina Faso, Cameroun, Chad, Côte d'Ivoire, Guinea, Mali, Niger and Nigeria). The Ansongo-Niamey basin is 126 161 km<sup>2</sup> wide, approximately 17% of the Niger basin's area. The upstream of the Ansongo-Niamey basin is Ansongo, a rural commune and small town in the Gao Region of eastern Mali, and its downstream is Niamey, the capital and largest city of Niger. The study area is located between the latitudes of 2°W and 2°E on one hand and the longitudes of 12°N and 17°N on the other.

### 2.1.2 Administration and Human Activities

The Niger basin countries have developed a framework for transboundary basin management. The Niamey Act was signed by the 9 countries of the basin (Algeria excluded) in 1963, followed by the creation of the Niger River Commission in 1964. In 1980, a new inter-governmental convention led to the creation of the Niger Basin Authority (NBA) which replaced the previous commission. The objectives of the NBA are “to promote cooperation between the member countries and to ensure integrated development of the basin in all the fields by the development of its resources in particular on the levels of energy, hydraulics, agriculture, livestock breeding, fishing, pisciculture, forestry, transport and communication and industry”. The Niger River has always been used for fishing and for extracting building materials.

The populations of the basin have also historically used the river for agriculture, in particular by exploiting its hydrological cycles and by cultivating rice and sorghum at its banks after the floods. Trade within the basin also developed thanks to the river, which allowed the transport of agricultural products, gold and many more (copper, fabrics, manufactured goods, salt, ...). Nowadays, activities around the basin have developed, in particular irrigation and rain-fed agriculture, but also mining. The hydraulic development of the river although still limited, meanwhile, it tends to increase by allowing a balanced and sustainable development between countries for irrigation and hydroelectric production [37]. The Ansongo-Niamey basin is shared between Burkina Faso, Mali and Niger respectively at the proportions of 47%, 31% and 22%. The basin comprises 5 regions in Burkina Faso (Centre-Est, Centre-Nord, Est, Plateau-Central and Sahel), 3 regions in Mali (Gao, Mopti and Timbuktu) and 1 department (Tillabery) and 1 district (Niamey) in Niger.



**Figure 2.1:** Map of Ansongo-Niamey basin

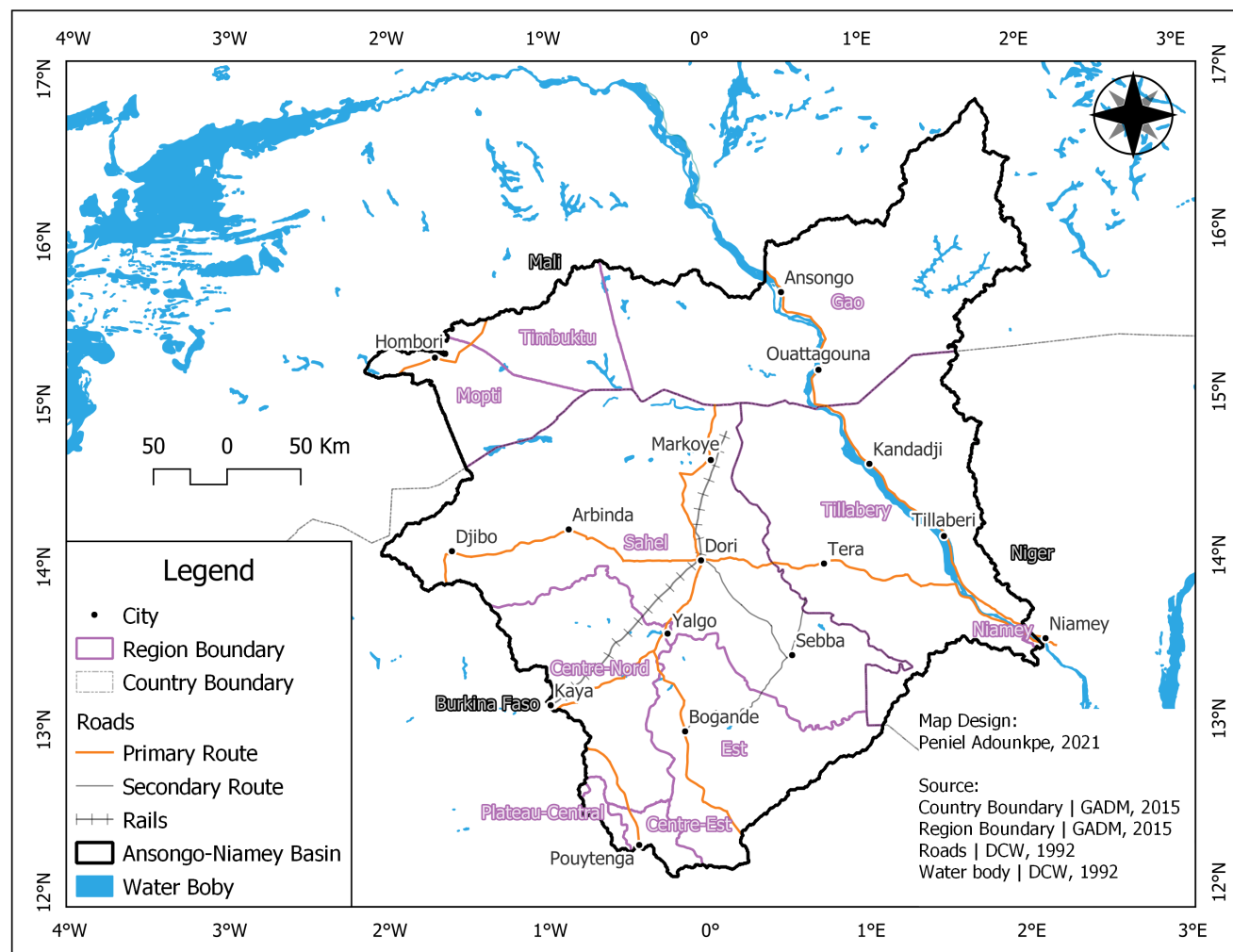


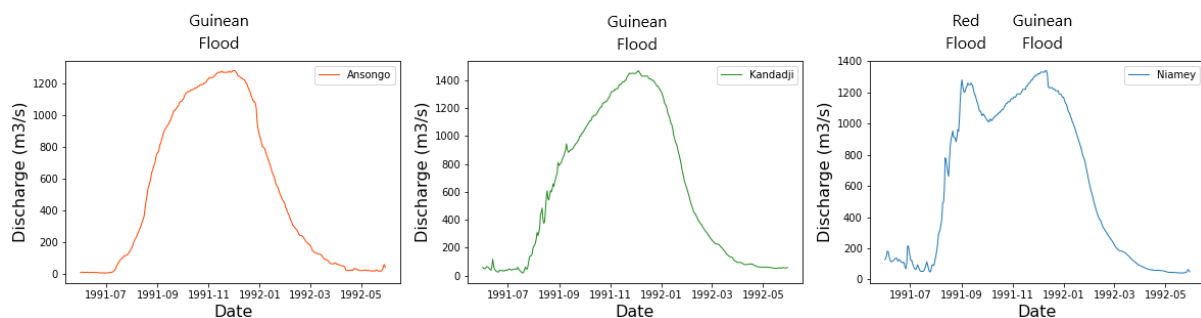
Figure 2.2: Administrative map of Ansongo-Niamey basin

### 2.1.3 Hydrology

The Niger River has its source in Guinea and flows in a west-east direction through Mali, Niger and Benin and outlets in Nigeria. The portion of the Niger river within the study area measures 380 km and has for main tributaries the Gorouol, Dargol and the Sirba rivers at its right bank and Ayorou river at its left bank. From the upstream to the downstream of the Ansongo-Niamey basin we have the following hydrology:

- The **Garouol** river originates from Hombori and Djibo and travels 413 km to join the Niger river near Kandadji.
- The **Ayorou** river originates from Tera and travels 213 km to join the Niger river at Kandadji.
- The **Dargol** river originates from Tera and travels 210 km to join the Niger river at Tillaberi.
- The **Sirba** river, the most important branch of the Niger river in our area of study, originates from Pouytenga and travels 419 km to join the Niger river near Niamey. The Faga river also flows into the Sirba.

The Ansongo-Niger basin runoff regime is affected by different types of reoccurring floods, which result from the geographic locations and characteristics of their main source areas. The first one is the Guinean Flood, which originates from the headwaters of the Niger in the Guinean highlands during the rainy season between July and November. The flood originating in the Guinean highlands experiences its peak usually around October. From here, the Niger flows into a vast wetland that covers an area and delays there for approximately three months. This upper flood arrives in the Ansongo around January, although rainfall in the Sahelian region falls at the same time as in the Guinean highlands. The second one is the annual peak during the rainy season (July to November) in the Ansongo-Niamey basin called the “Red Flood” or “Sahelian Flood” [2]. Figure 2.3 illustrates well the basin’s runoff regime by distinguishing red flood from the Guinean flood.



**Figure 2.3:** Hydrographs of Ansongo, Kandadji and Niamey for June 1991 to Mai 1992

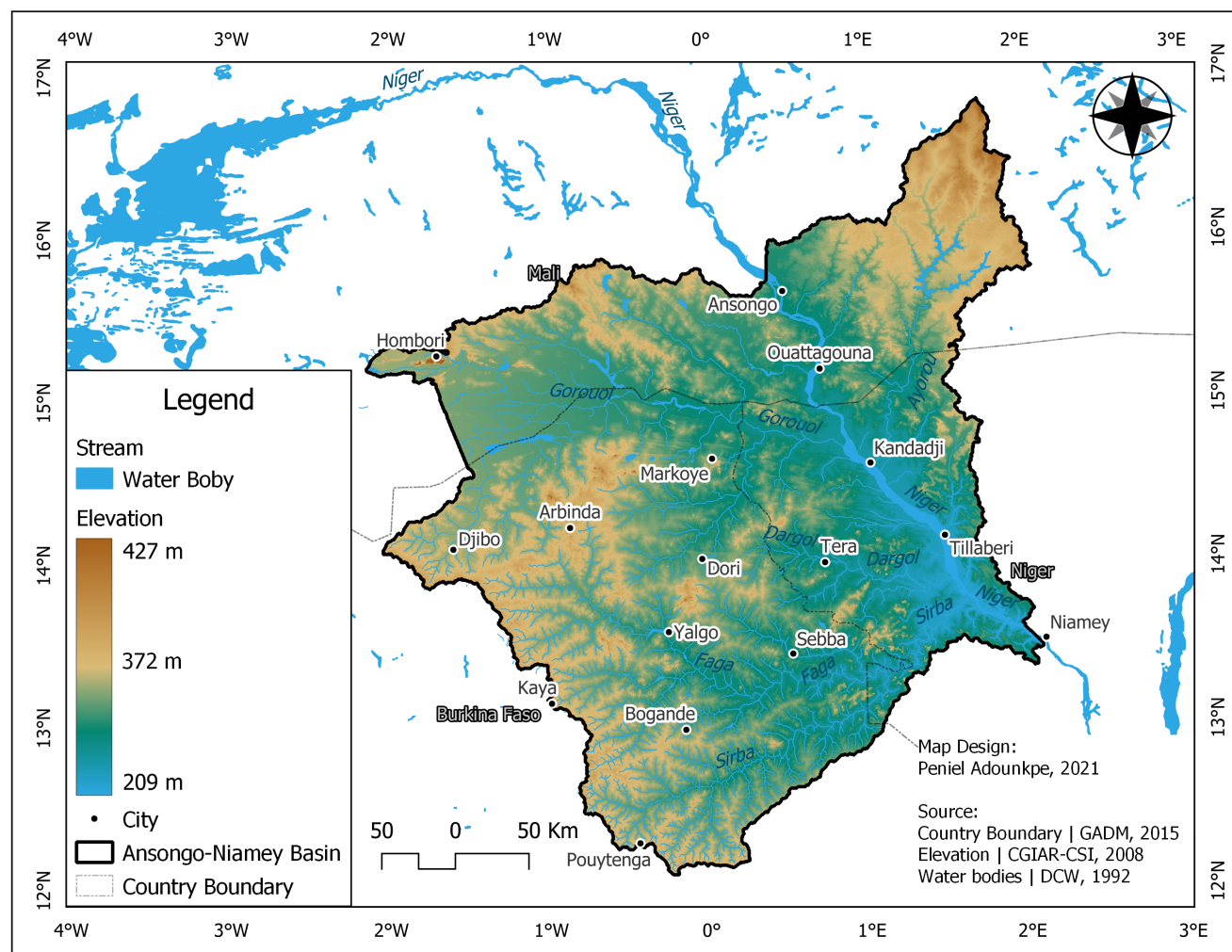
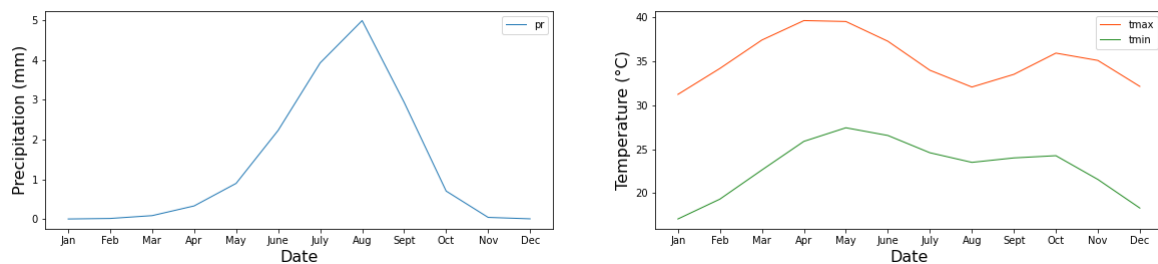


Figure 2.4: Hydrology of Ansongo-Niamey basin

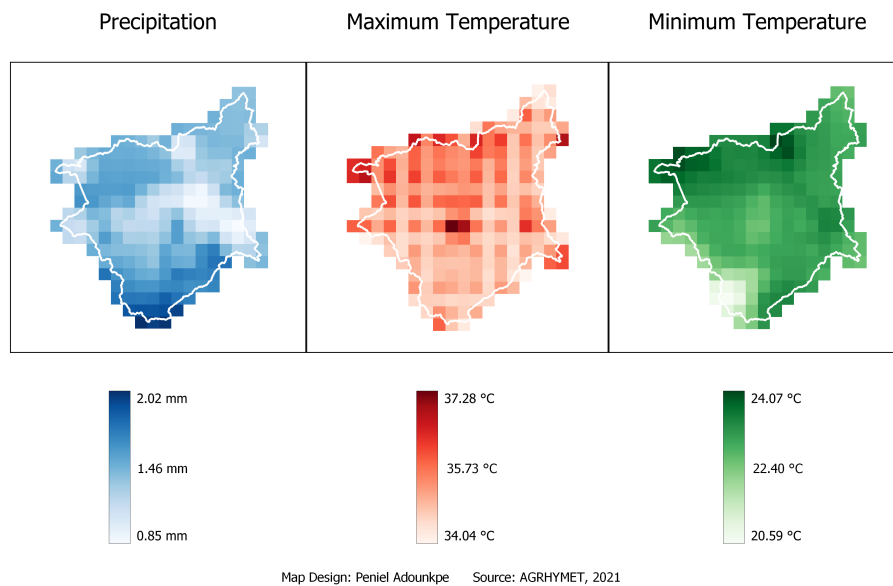
### 2.1.4 Climatology

The climate of the study area is governed by the West African monsoon system. From the temporal distribution in figure 2.5, we can say that the Ansongo-Niamey has a unimodal rainfall regime with the rain starting in May, reaching its peak (around 5 mm) in July and ending in October. The minimum temperature differs from the maximum temperature by around 12 °C. We could spot an obvious relation between the precipitation and the temperature. The temperature rises from January to May and declines as soon as the first rains occur in the basin. The temperature is at its lowest in July whereas the precipitation is at its highest. After July, the temperature rises a bit more before declining again, this time around because of harmattan. The spatial distribution of the precipitation and temperature in the study area shows also the same deduction. The southern parts of the basin have more precipitation and less temperature compared to the northern parts of the Ansongo-Niamey basin. Thus, we can say that the precipitation is inversely correlated to the temperature in the Ansongo-Niamey basin (figure 2.6).



**Figure 2.5:** Monthly average of the temporal distribution of precipitation, maximum temperature and minimum temperature in the Ansongo-Niamey basin from June 1981 to December 2010



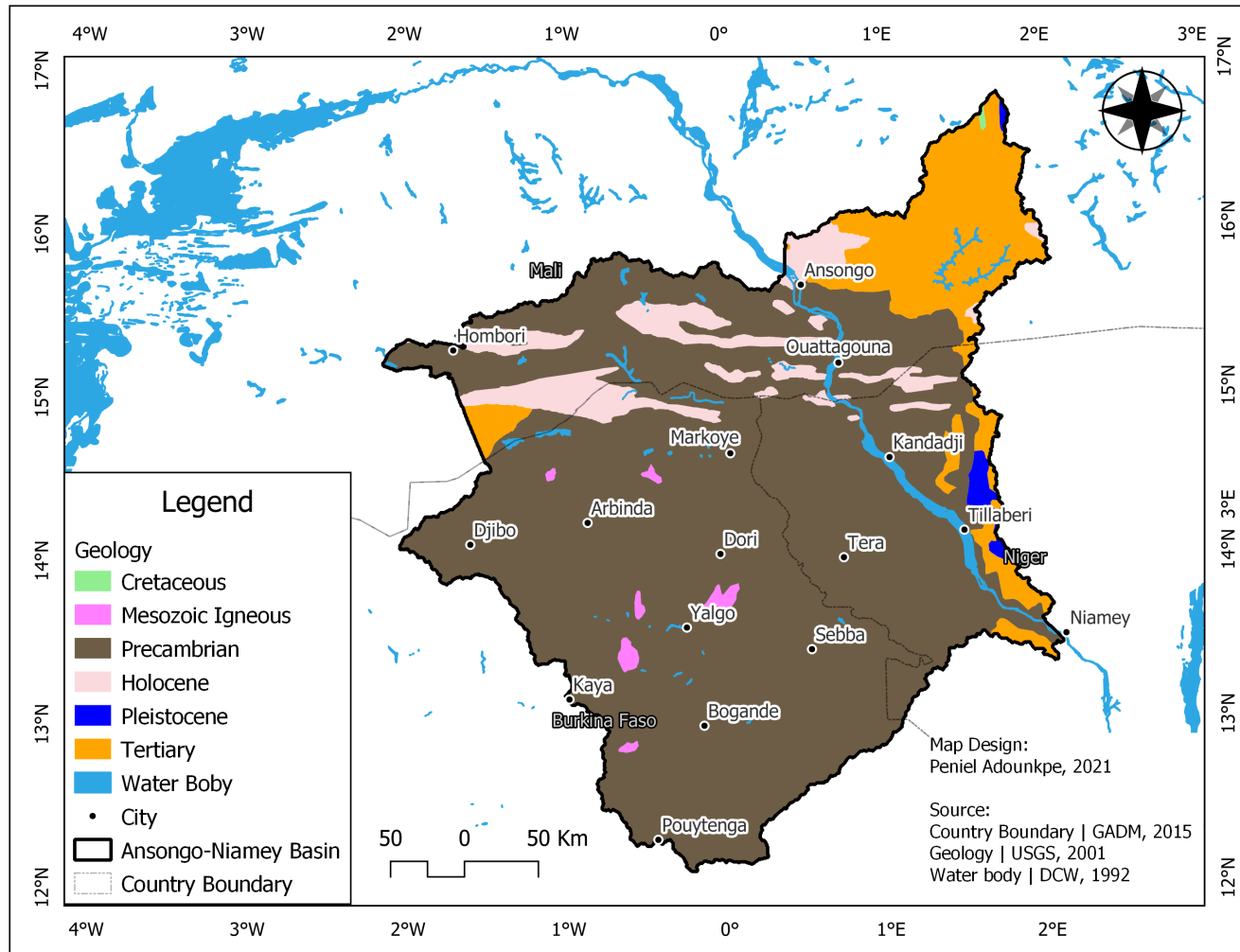


**Figure 2.6:** Average of the spatial distribution of precipitation, maximum temperature and minimum temperature in the Ansongo-Niamey basin from June 1981 to December 2010

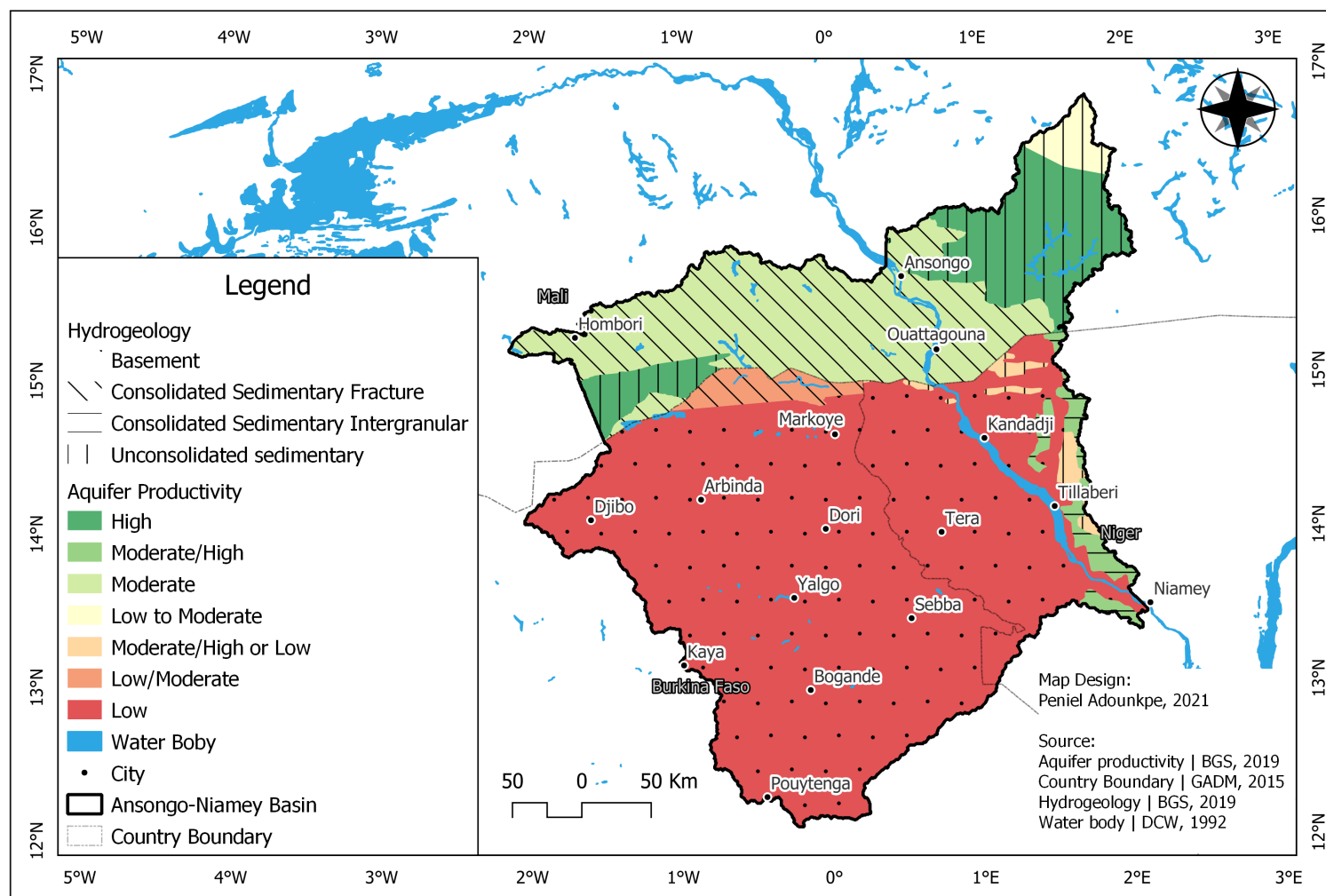
### 2.1.5 Geology and Hydrogeology

The geology shown in the figure 2.7 describes significant geological units at a national scale. The geological descriptions include either the geological age or lithological description. In this map, the Precambrian unit is the predominant category with nearly 4/5 of the basin's area. After there is the Tertiary unit, predominantly located in the northeast, with 12% of the catchment's surface. the 3rd geological unit is the Holocene, scattered across the north, with 7% of the catchment's surface.

The hydrogeological map (figure 2.8) refers to the aquifer type and is defined in terms of the hydrogeological environment, or the geological characteristics that largely control the nature of groundwater flow and storage in the aquifer. Aquifer productivity is defined based on a synthesis of borehole yield data, which is the most widely available data for aquifers. For more information, description of aquifer type and range in yield, go to the appendix 4 of the document. Only the northern part of the Ansongo-Niamey basin has acceptable aquifer potential. The other 64% found in the south has low aquifer productivity. This must be due to the upper Niger basin's delta that is located upstream of the study area.



**Figure 2.7:** Geology of Ansongo-Niamey basin



**Figure 2.8:** Hydrogeology of Ansongo-Niamey basin

## 2.2 Data

There are two categories of data used in other to carry out this thesis: data need to reach the objectives of this research (table 2.1) and the data need to describe the study area (table 2.2).

**Table 2.1:** Data used for objectives

Description	Format	Source, Date
Meteorological data: daily precipitation, minimum temperature and maximum temperature from 1981 to 2015	NetCDF (nc)	AGRHYMET, 2021
Hydrological data (1): daily river discharge at the station of Ansongo, Kandadji and Niamey from June 1981 to December 2010	Open Office spreadsheet (xlsx)	AGRHYMET, 2021
Hydrological data (2): daily river discharge at the station of Ansongo, Kandadji and Niamey from 1981 to 2010	Open Office spreadsheet (xlsx)	NBA, 2021
Performance of hydrological model: efficiency criteria for hydrological model assessment	Text	Journal articles, Publication date
Topographic data: digital elevation model (SRTM 90m database) of West Africa	Tag Image File Format (tif)	CGIAR-CSI, 2008

Note: AGRHYMET provided the methodology used to get the meteorological data. These data (precipitation, minimum temperature and maximum temperature) were obtained by: bias correction of the satellite-estimated weather variable and the actual fusion of the bias-corrected data with the in situ observations. The quantile mapping method was used for the bias correction of the satellite data. This technique is one of the most widely used statistical methods of bias correction. To get more information about this method, please check the work of Ines & Hansen, 2006; Piani et al., 2010a; Gudmundsson et al., 2012; Ajaaj et al., 2015 on quantile mapping.

The additional data collected was used to understand the study area. They were to generate thematic maps of the Ansongo-Niamey basin and other information related to the study area's description. These data are open source and free of use for scientific research.

**Table 2.2:** Additional data used to describe study area

Data	Format	Source, Date
Country boundary	Shapefile (shp)	GADM, 2015
Background map	Tag Image File Format (tif)	Natural Earth, 2021
Water bodies, roads and railroads	Shapefile (shp)	DCW, 1992
Geology	Shapefile (shp)	USGS, 2001
Hydrogeology and aquifer productivity	Shapefile (shp)	BGS, 2019

## 2.3 Tools

For this research, open-source tools prevailed. Python is the interpreted high-level general-purpose programming language that was used. Jupyter Notebook is the open document format based on JSON for interactive computing that was used as a programming interface. Anaconda is a distribution of the Python and R programming languages for scientific computing, that aims to simplify package management and deployment. Two environments were built in Anaconda for this work **GIS** for the processing of spatial data and **tf** for processing of the xlsx files and machine learning operations.

The following libraries (sorted by the frequency of use) were imported in Python:

- Numpy is the fundamental package needed for scientific computing with Python. It was used to manage multidimensional arrays.
- Pandas is the powerful Python data analysis toolkit. It was used to manage data structures.
- TensorFlow is an end-to-end open-source platform for machine learning. It was used as deep learning library in Python
- Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. It was used to create graphs.
- Scikit-Optimize is a simple and efficient library to minimize (very) expensive and noisy black-box functions. It was used to optimize the deep learning model's hyperparameters.
- Scikit-learn is a Python module for machine learning built on top of SciPy. It was used to normalize the data.
- Openpyxl a Python library to read/write Excel 2010 files. It was used to read xlsx files.
- Xarray is a Python package that works with labeled multi-dimensional arrays. It was used to manage netCDF files.
- Glob a Python module to find all the pathnames matching a specified pattern according to the rules used by the Unix shell. It was used while concatenating all netCDF files in folders.

All the details about the versions of the modules and environments in the link below:  
<https://anaconda.org/Pyaj/environments>

QGIS is a free and open-source cross-platform desktop geographic information system application that supports viewing, editing, and analysis of geospatial data. QGIS was used to make the maps in the thesis and to delineate the Ansongo-Niamey basin. Paint 3D, a raster graphics and 3D modeling application, was used to make and comment on some of the figures of this document. And last but not least, Overleaf is the collaborative cloud-based LaTeX editor used to write the report of this thesis.

## **2.4 Methods**

To complete the first 2 objectives set for this research, 3 main steps were followed:

- data preprocessing during which the raw data from the data providers (AGRHYMET and NBA) would be transformed in a suitable format for deployment into the LSTM model,
- hyperparameter optimization where the best set of hyperparameters were selected (SO1) and
- LSTM model was calibrated, validated and tested (SO2).

The figure below shows the summary of the methodology used in this research. The text in blue represent the modules and those in green represent the main methods used in this research.

The subsections below will go into all the necessary details of the process carried out in such a manner to make this work as reproducible as possible.

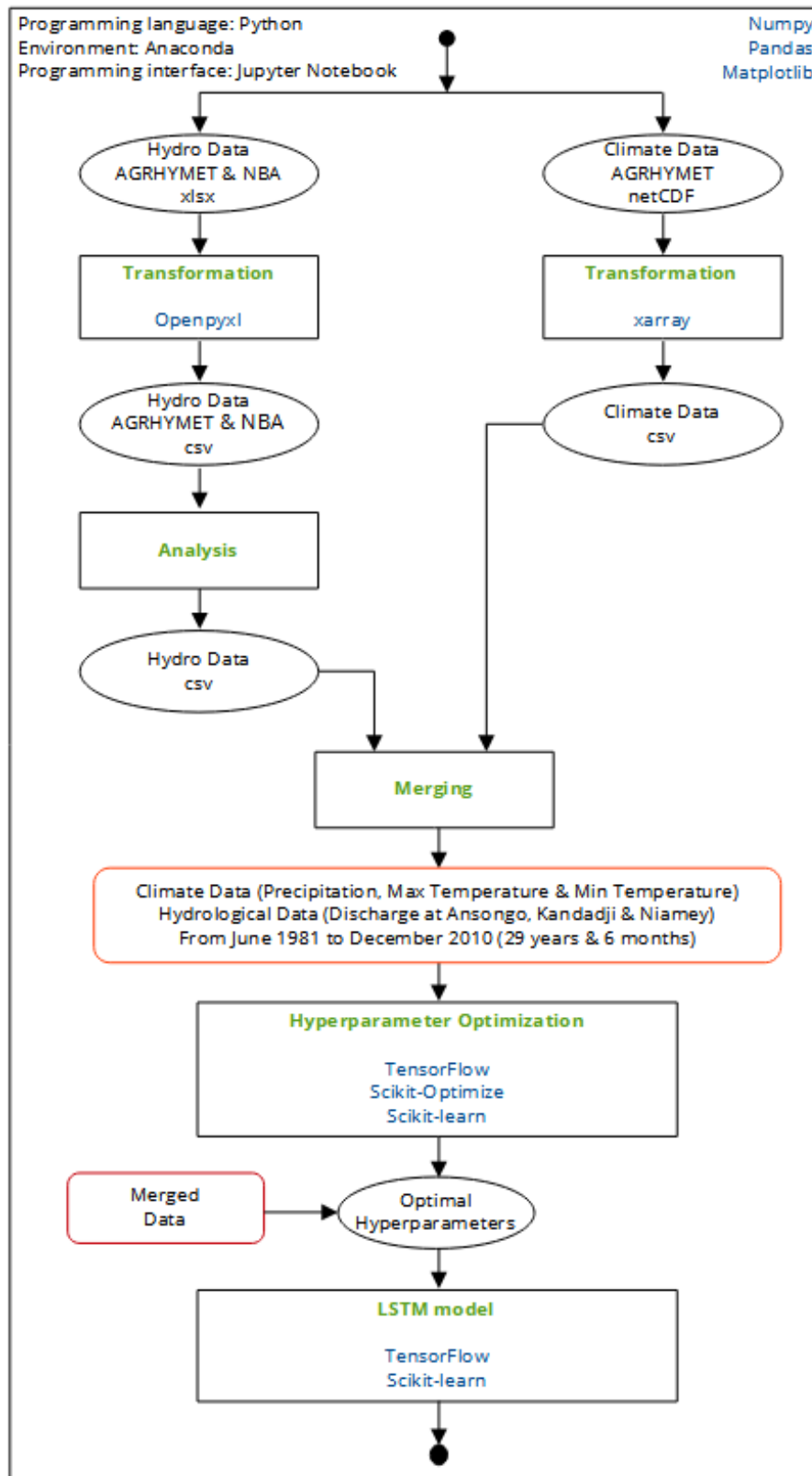


Figure 2.9: Summary of the main methodology

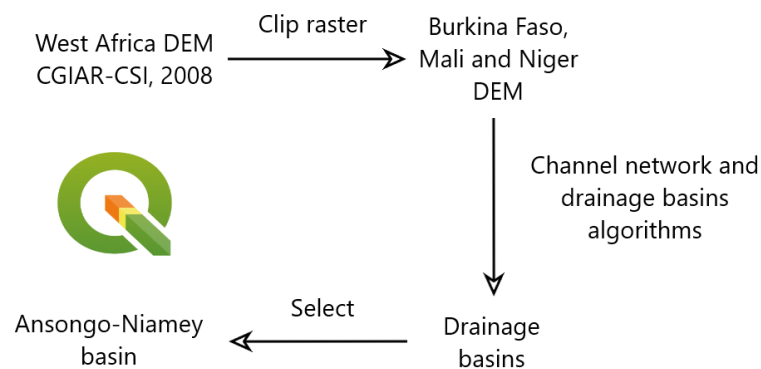


### 2.4.1 Data Preprocessing

The data processing was done into 5 steps:

1. Delineation of the Ansongo-Niamey basin,
2. Transformation of the climate data (precipitation, maximum and minimal temperature) from netCDF to csv files,
3. Transformation of the hydrological data (river discharge at Ansongo, Kandadji and Niamey) from xls/xlsx files to csv files,
4. Analysis of the hydrological data by comparing their different sources data and filling the eventual missing data inside the time series and
5. Merging the climate and the hydrological data to have a final csv ready to be used in the LSTM model.

The first step of catchment delineation was the only one applied using QGIS. The topographic data (DEM) of West Africa of CGIAR-CSI was used for this purpose. First of all the DEM was clipped into a smaller region (Burkina Faso, Mali and Niger). The channel network and drainage basins algorithm, found in the QGIS toolbox after installation of the SAGA GIS in QGIS, was used to get drainage basins of the clipped DEM with a threshold of 2. The Ansongo-Niamey basin was selected among the catchments generated. All of the other steps of data processing were done in the Python Jupyter Notebook.



**Figure 2.10:** Process of catchment delineation in QGIS

The second step was to transform the climate data (precipitation, maximum and minimum temperature) from netCDF to csv format. This was the only operation executed in the GIS environment of anaconda. The precipitation, maximum temperature and minimum temperature were concatenated by date using glob and xarray into one netCDF file. The concatenated files were grouped by date then averaged by their spatial distribution in the Ansongo-Niamey basin. This file was finally converted to pandas then to csv.

The remaining steps were executed in the **tf** environment. The third step was to transform the hydrological data (discharge at Ansongo, Kandadji and Niamey) from **xlsx** to **csv** format. The **xlsx** discharge data from AGRHYMET and NBA were loaded using **pandas** and **openpyxl**. The data was rearranged and uniformed by simple operations such as renaming columns, dropping blank columns and index manipulation of dataframe.

The fourth step was analyzing the hydrological data received from AGRHYMET and NBA. During this step, the data from AGRHYMET was chosen over the NBA source. Meanwhile, the NBA source was used to fill the missing data of AGRHYMET at the station of Kandadji from 2006 to 2010. For the remaining missing data, simple interpolation techniques were used such as linear interpolation and polynomial interpolation. The polynomial interpolation method was only used to fill the period from June 1997 to August 1998 at the Ansongo station. This method of interpolation was proposed for this period because the distribution had experienced minimums.

The last step of the preprocessing was to merge the climate and the hydrological data using the date column. The final output used is a dataframe of 10806 rows and 7 columns with the number of rows representing the number of days of observation and the number of columns representing the features or input variables of our model.

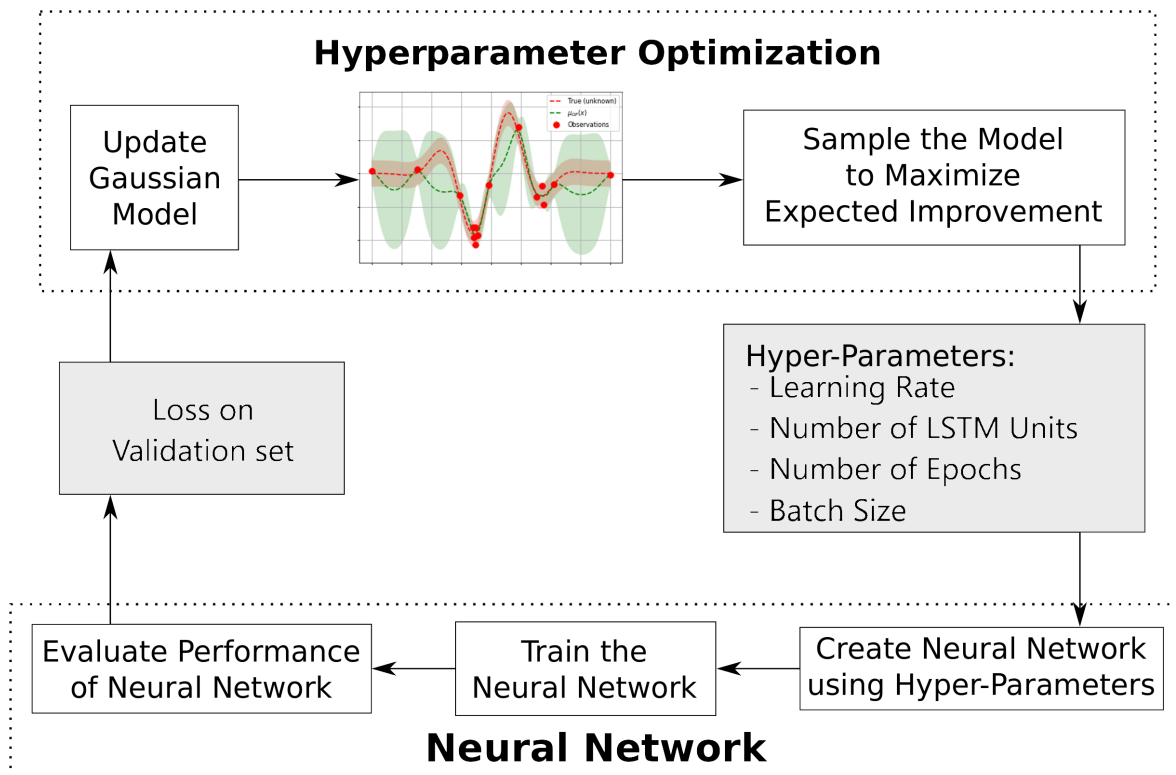
## 2.4.2 Hyperparameter Optimization

Hyperparameter optimization refers to performing a search to discover the set of specific model configuration arguments that result in the best performance of the model on a specific dataset.

There are many ways to perform hyperparameter optimization, although modern methods, such as Bayesian optimization, are fast and effective. The Scikit-Optimize (**skop**) library is an open-source Python library that provides an implementation of Bayesian Optimization that can be used to tune the hyperparameters of machine learning models from the scikit-learn Python library. Bayesian optimization is an efficient optimization algorithm based on Bayes Theorem capable of finding the minimum or maximum of arbitrary cost functions [13].

In our case, the goal was to minimize the cost function of the LSTM model. The selected LSTM NN parameters to optimize were the learning rate, the number of LSTM units, the dimension of epochs and the batch size. The idea is that we have a true fitness function (the performance of a NN on some dataset) and we want to find the minimum. To find out how the NN performs with a given set of hyperparameters, we need to train the network. A model called the Gaussain Process is built. The search starts with the initial values of the hyperparameters. At the early optimization phase, the search is explored more and in the later phase, the search is focused on the best regions found. At the end of the process, we would get a set of hyperparameters corresponding to the minimal validation loss.

To resume, Scikit-Optimize provides a drop-in replacement for grid search, by utilizing Bayesian optimization where a predictive model referred to as “surrogate” is used to model the search space and utilized to arrive at a good parameter values combination as soon as possible [38].



**Figure 2.11:** Hyperparameter optimization process

The following table shows the hyperparameters, their search dimension, their search range and their initial values.

**Table 2.3:** Optimized hyperparameters using Scikit-Optimize

Hyperparameter	Dimension	Range	Initial Value
Learning rate	Real	0.001 to 0.1	0.01
Number of LSTM units	Integer	1 to 500	100
Number of Epochs	Integer	1 to 500	100
Batch size	Integer	1 to 500	100

At the end of this subsection, there would be an interpretation of the optimization process. To help understand how the optimization process is proceeding, we would plot the location and order of the points at which the loss is evaluated with the objective plot (plots the partial dependence of the objective, as represented by the surrogate model, for each dimension and as pairs of the input dimensions) and the evaluation plot (helps with visualizing the location and order in which samples are evaluated)

### 2.4.3 LSTM Model

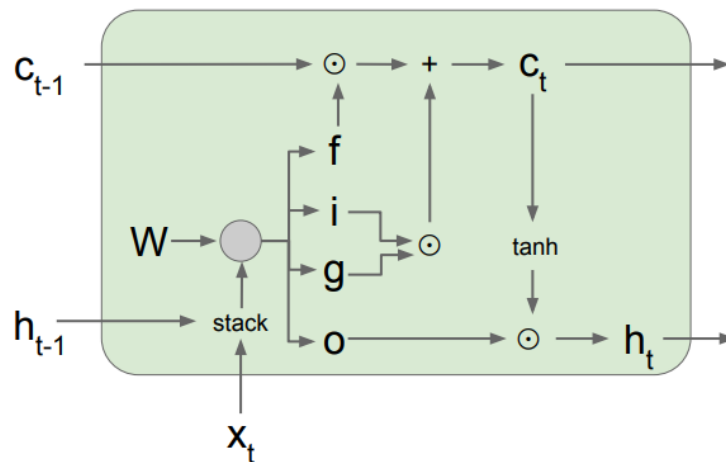
#### Concept

LSTMs are famous for their ability to learn and store long-term dependencies of the input-output relationship. They are capable of learning directly from time series (can learn to bridge minimal time lags above 1000 discrete-time steps) [39]. It was designed with a more complex architecture to have better gradient flow properties to solve the vanishing and exploding gradient problems. Here are the equations expressing the LSTM's structure.

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} W \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix} \quad (2.1)$$

$$\begin{aligned} c_t &= f \odot c_{t-1} + i \odot g \\ h_t &= o \odot \tanh c_t \end{aligned} \quad (2.2)$$

LSTM maintains 2 hidden states at every timestep: the hidden state and the cell state. In addition to the classical hidden state, there is a new state called the cell state that is responsible for the long-term memory of the network.



**Figure 2.12:** Structure of a long short-term memory neural network cell

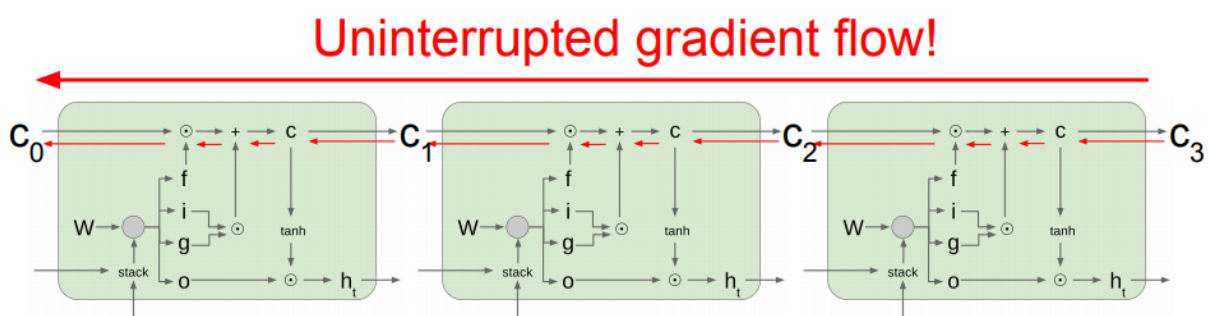
Source: Fei-Fei et al. [25]

There is also a new set of gates, each with a specific task:

- forget gate  $f$  - whether to erase cell
- input gate  $i$  - whether to write to cell

- gate gate  $g$  - how much to write to cell
- output gate  $o$  - how much to reveal

During the backpropagation from  $c_t$  to  $c_{t-1}$  of the LSTM, there is only element-wise multiplication by  $f$  coupled with an addition of the product of  $i$  and  $g$ . Knowing that the add gate acts as a gradient distributor and the multiplication gate acts as a gradient switcher, the backpropagation is way easier to compute and does not involve a matrix multiplication by the weight  $W$ . When the LSTM is put into the network, the cell states act up as a gradient superhighway that lets gradients pass uninterruptedly from the last cell state to the first cell state.



**Figure 2.13:** Gradient flow in LSTM cells

Source: Fei-Fei et al. [25]

This explains why LSTM networks are the go to RNN when it comes to time series prediction.

After preprocessing the data and found the optimal hyperparameters, here are the detailed methods used for setting up, training, validating and testing the LSTM model.

The data used in this model is the preprocessed data containing:

- 7 columns: date, precipitation, maximum temperature, minimum temperature, discharge at Ansongo, discharge at Kandadji and discharge at Niamey and
- 10806 rows representing the number of days from June 1981 to December 2010.

### Neural Network Data Preprocessing

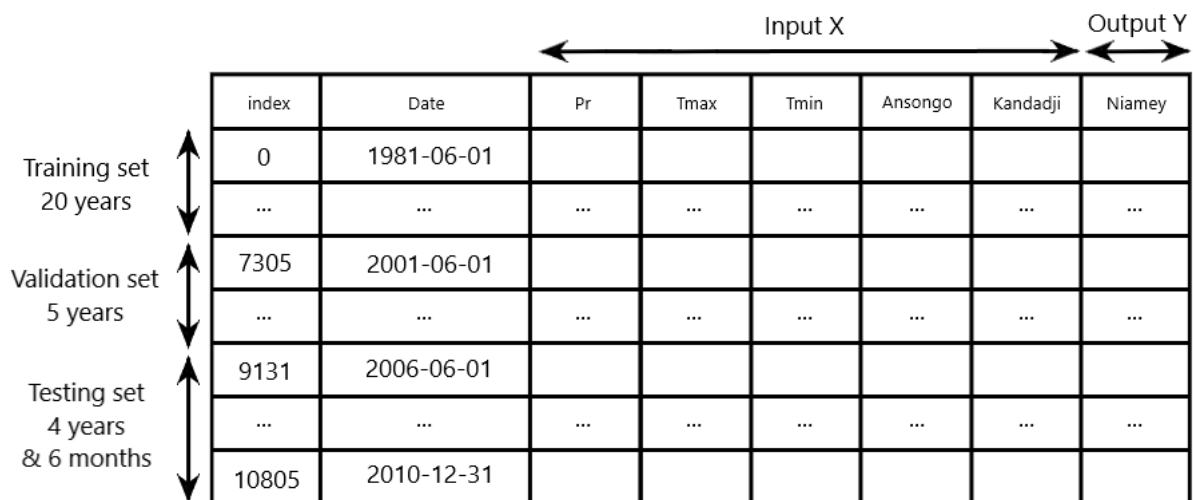
Before introducing the data into the LSTM model, there are a few transformations to apply.

1. Data normalization,
2. Time series to supervised learning series transformation and
3. Data splitting

The first operation is to normalize the features of the dataset. When a network is fit on unscaled data that has a range of values large inputs can slow down the learning and convergence of your network and in some cases prevent the network from effectively learning your problem. To solve this issue, it is highly recommended to normalize the dataset. Normalization is a rescaling of the data from the original range so that all values are within the range of 0 and 1.

The second operation is to transform the time series into a supervised learning series. Time series forecasting problems must be re-framed as supervised learning problems. From a sequence to pairs of input and output sequences. A time series is a sequence of numbers that are ordered by a time index. This can be thought of as a list or column of ordered values. A supervised learning problem is comprised of input patterns ( $X$ ) and output patterns ( $y$ ), such that an algorithm can learn how to predict the output patterns from the input patterns [13]. To do this, we used the “series\_to\_supervised” function of Jason Brownlee and assumed that the discharge of Ansongo and Kandadji both takes one timestep (a day) to get to Niamey.

The third operation is to split the data into training, validation and training sets. The model is initially fit in a training set, which is a set of examples used to fit the parameters of the model. Successively, the fitted model is used to predict the responses for the observations in a second dataset called the validation dataset. The validation dataset provides an unbiased evaluation of a model fit on the training dataset. The test dataset is a dataset used to provide an unbiased evaluation of a final model fit on the training dataset. The dataset was also split into inputs and outputs. The figure below illustrates the different divisions of the dataset.



**Figure 2.14:** Different subdivision of the dataset

To finish this transformation, the dataset was reshaped in a 3D format with the final shape of each subdivision of the dataset being: (number of samples, time steps, features).

### Setting up the LSTM model

Google's guide on deep learning with Keras using the TensorFlow was of great help with setting up and training the LSTM model [40].

A sequential model was used for this work. A Sequential model is appropriate for a plain stack of layers where each layer has exactly one input tensor and one output tensor. As agreed, an LSTM layer was added to the model and the number of its internal units was set using the value obtained after hyperparameter optimization. A dense layer of 1 unit was added to the LSTM units to sum up the result of the LSTM units and to set it up as output data.

Now that the model is defined we can compile it and specify its optimizer, learning rate and loss function. The process of minimizing any mathematical expression is called optimization. Optimizers are algorithms or methods used to change the attributes of the neural network such as weights, biases and learning rate to reduce the losses. The optimizer used here is the Adaptive Moment Estimation (Adam) optimizer. Adam optimization is a stochastic gradient descent method that is based on the adaptive estimation of first-order and second-order moments. According to Kingma the method is "computationally efficient, has little memory requirement, invariant to diagonal rescaling of gradients, and is well suited for problems that are large in terms of data/parameters" [41]. The learning rate obtained from the hyperparameter optimization process is also set here. The last step of compiling our LSTM model would be to set its loss function. Since the present work would use a large number of samples, we would not set a regularization term because the chances of overfitting the data are low. As for the data loss function, the mean square error would be chosen because larger mistakes are attributed to bigger penalties due to the squaring inside the function.

$$D_i(W, x) = \sum_{i=1}^n (d_i - y_i)^2 \quad (2.3)$$

In this equation,  $n$  represents the number of samples,  $D_i$  is the data loss,  $d_i$  is the targeted value of the output and  $y_i$  is the output value of the model.

### Training and evaluation of LSTM model

For the training phase of our LSTM, the number of epochs and the batch size obtained after hyperparameter optimization are set in the model to fit. The evaluation of the model is done using the test dataset. We evaluated our LSTM model by analyzing the curve of the loss function over the number of epochs. To evaluate the performance of forecasting models in fields related to hydrology, coefficient of determination ( $R^2$ ), Nash-Sutcliffe Efficiency (NSE) and Root Mean Square Error (RMSE) are often used as statistical methods to compare predicted values with observed values. The  $R^2$  is the proportion of the variance in the dependent variable that is predictable from the independent variables. The NSE measures the ability to predict variables different from the mean and gives the proportion of the initial variance accounted for by the model. The RMSE is frequently used to evaluate how closely the predicted values match the

observed values, based on the relative range of the data.

$$R^2 = \frac{\left[ \sum_{i=1}^n (y_i - \bar{y}_i)(\hat{y}_i - \bar{\hat{y}}_i) \right]^2}{\sum_{i=1}^n (y_i - \bar{y}_i)^2 \sum_{i=1}^n (\hat{y}_i - \bar{\hat{y}}_i)^2} \quad (2.4)$$

$$NSE = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y}_i)^2} \quad (2.5)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (2.6)$$

In the equations,  $y_i$  represents the observed discharge at a time  $t$ ,  $\hat{y}_i$  represents the simulated discharge at a time  $t$ ,  $\bar{y}_i$  is the mean of the observed discharge,  $\bar{\hat{y}}_i$  is the mean of the simulated discharge and  $n$  is the number of observations. The  $R^2$  values ranges from 0 to 1, with values close to 1 indicating good agreement, and typically values greater than 0.5 are considered acceptable in hydraulic simulations. Similarly to the  $R^2$ , the NSE values range from  $-\infty$  to 1 with the perfect model having the value 1. The RMSE values range from 0 to  $\infty$  with 0 implying the model has a perfect fit.

All notebooks used for this research can be accessed via the link below:  
<https://drive.google.com/drive/folders/10V4-a45OShwCrQOPK8jj9c2nF8ZJYicd>



## **Chapter 3**

### **Results and Discussion**

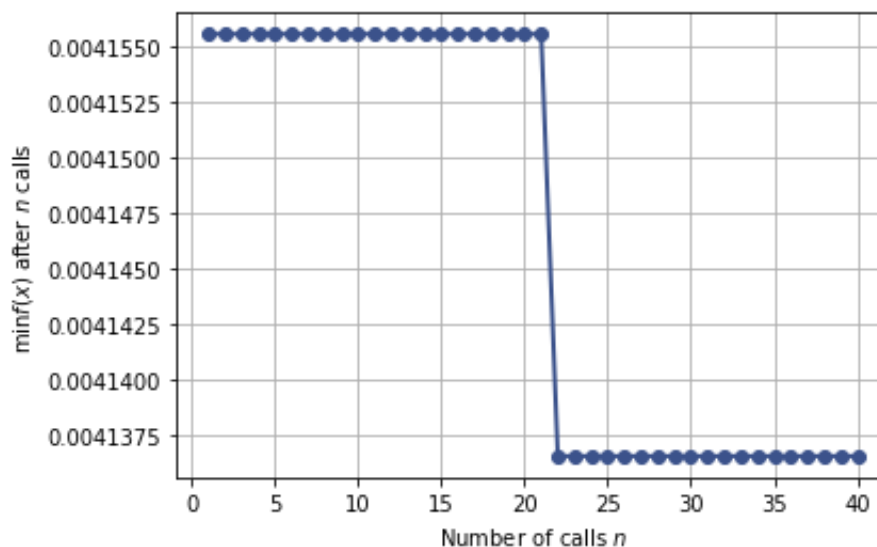
## 3.1 Results

This section would present the main findings of our work. The optimal hyperparameters of the model and its evaluation would be presented in the results. The discussion section would compare the DL model to physical models deployed at Niamey in literature. This discussion would be an attempt to complete the third research objective (SO3).

### 3.1.1 Hyperparameter Optimization

The hyperparameter optimization process went through 40 iterations. The hyperparameters and their associated loss could be found in the appendix table 4.3. The optimized hyperparameters values found by the Bayesian optimizer are  $1.5616 \cdot 10^{-2}$  for the learning rate, 30 for the number of LSTM units, 500 for the number of epochs and 128 for the batch size. The lowest validation loss was  $4.136 \cdot 10^{-3}$ .

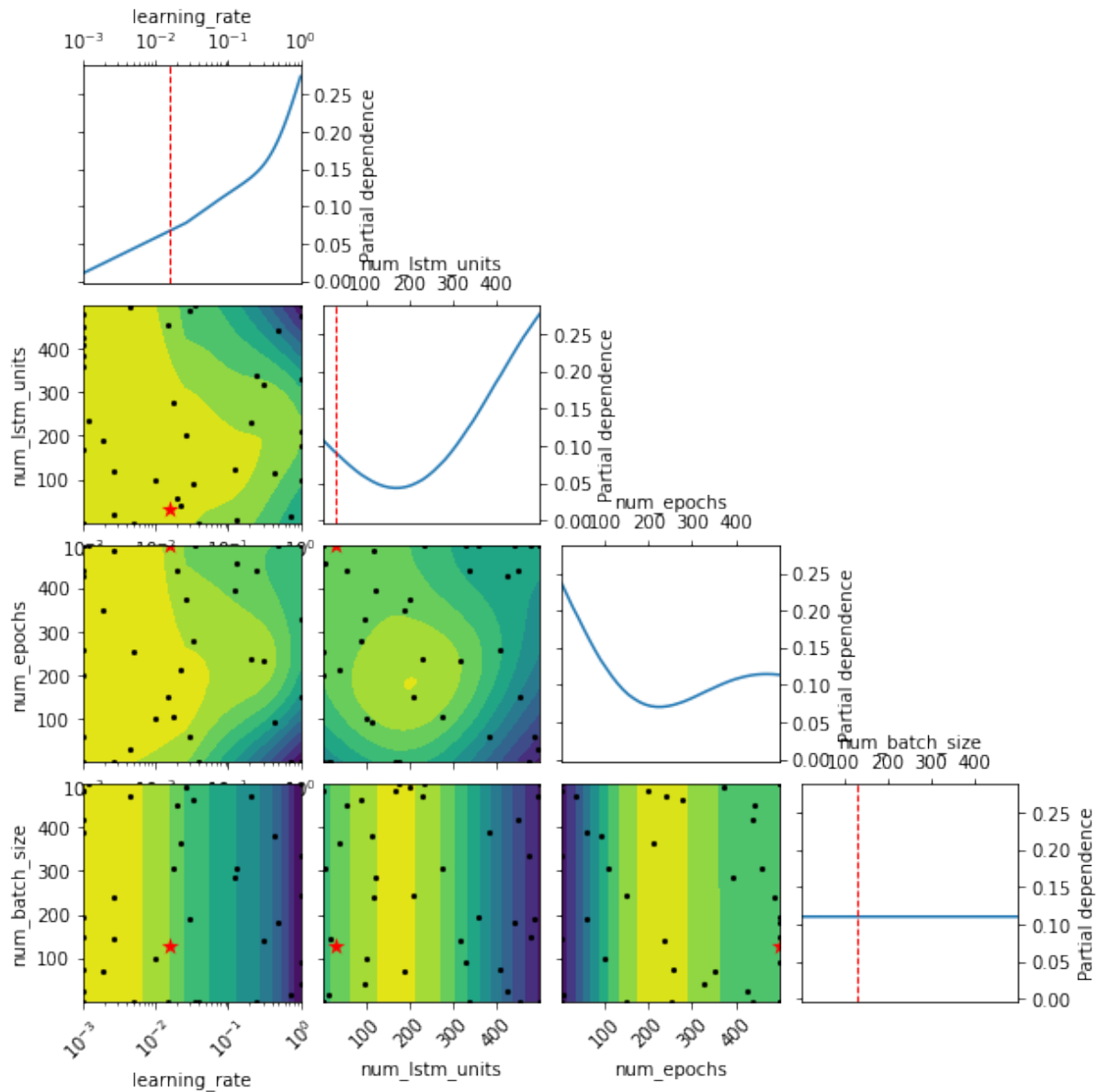
The progress of the hyper-parameter optimization can be easily plotted in the convergence plot. The best fitness value found is plotted on the y-axis.



**Figure 3.1:** Convergence plot

The convergence plot shows that the loss function of the LSTM model is already pretty low from the beginning of the process. At the 22nd search, the loss function drops by around  $2 \cdot 10^{-5}$ .

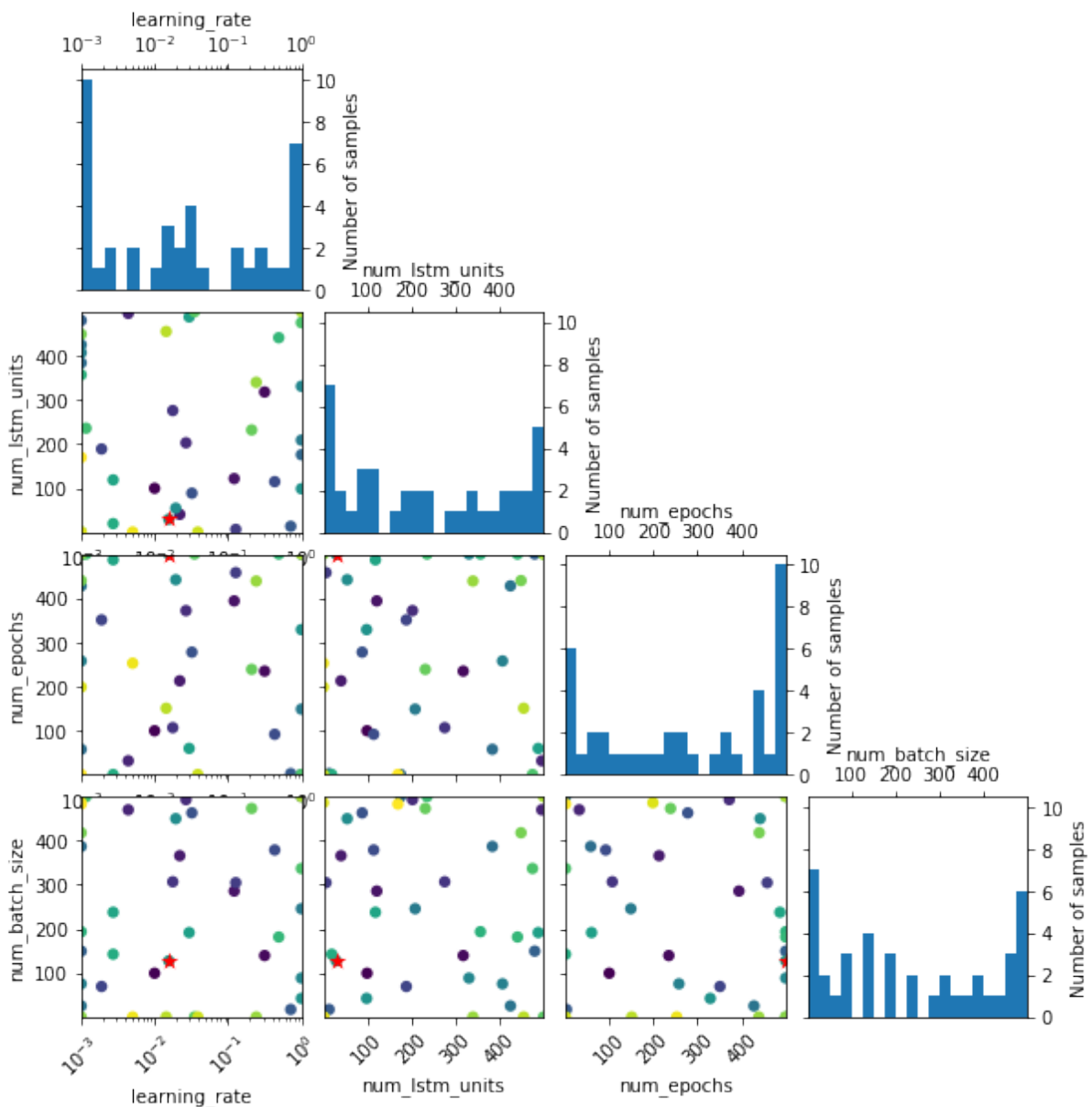
The following figure shows a matrix plot of all combinations of dimensions (hyperparameters). The verticals show the influence of a single dimension on fitness. This is a so-called “Partial Dependence plot” for that dimension. It shows how the approximated fitness value changes with different values in that dimension. The plots below the diagonal show the Partial Dependence for 3 dimensions. This shows how the approximated fitness value changes when we are varying 3 dimensions simultaneously.



**Figure 3.2:** Objective plot

The yellow regions show areas where the loss on the validation set is the lowest as contrary to the darker regions. The star in the plot represents the location where the optimal hyperparameters were found. From the curves shown in the figure above, the loss improves when the LSTM model’s learning rate is lower (the model learns better at smaller time steps), its number of LSTM units is not more than 350 and its number of epochs is bigger than 100 epochs. The batch size does not seem to affect a lot the loss function. The partial dimension plot of the batch size is flat which indicates that the performance might change with the batch size but there is then another choice of the number of epochs, the number of LSTM units and the learning rate in the layer that compensate for that.

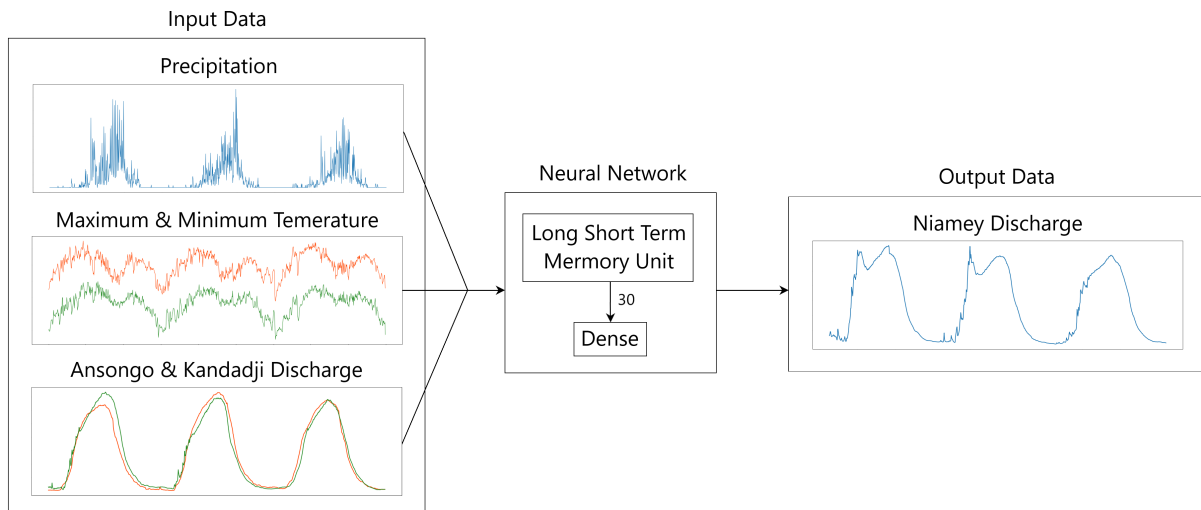
We also displayed another type of matrix plot: the evaluation plot. Here, the verticals show histograms of the sample distributions for each of the hyperparameters during the Bayesian optimization. The plots below the diagonal show the location of samples in the search space and the color-coding shows the order in which the samples were taken. The early samples are darker while the late samples are yellow. As expected, the earlier samples were more random and the later samples were more in clusters because a certain pattern started to form, the search was more structured. The hyperparameters were mostly sampled at the limits of the search range. If we had done more iterations, the results would have been even better and the clustering pattern would have been easier to identify.



**Figure 3.3:** Evaluation plot

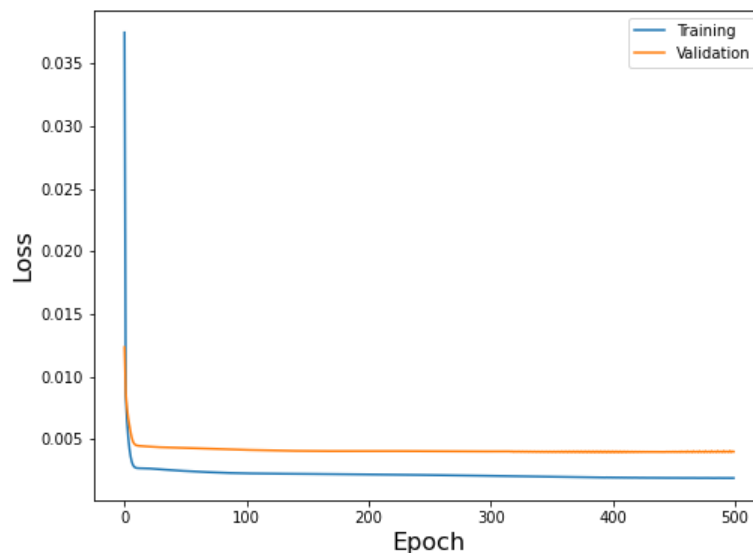
### 3.1.2 LSTM Model

Here is a quick summary of our LSTM model. As said in the methodology, its inputs are the precipitation and the maximum and minimum temperature on the Ansongo-Niamey basin and the discharge at Ansongo and Kandadji. 20 years of this input are used to train to model and 5 years of this input are used to validate the model. An additional 4 years and 6 months are used to test the model. The model is composed of 30 LSTM units and 1 dense layer.



**Figure 3.4:** LSTM model summary

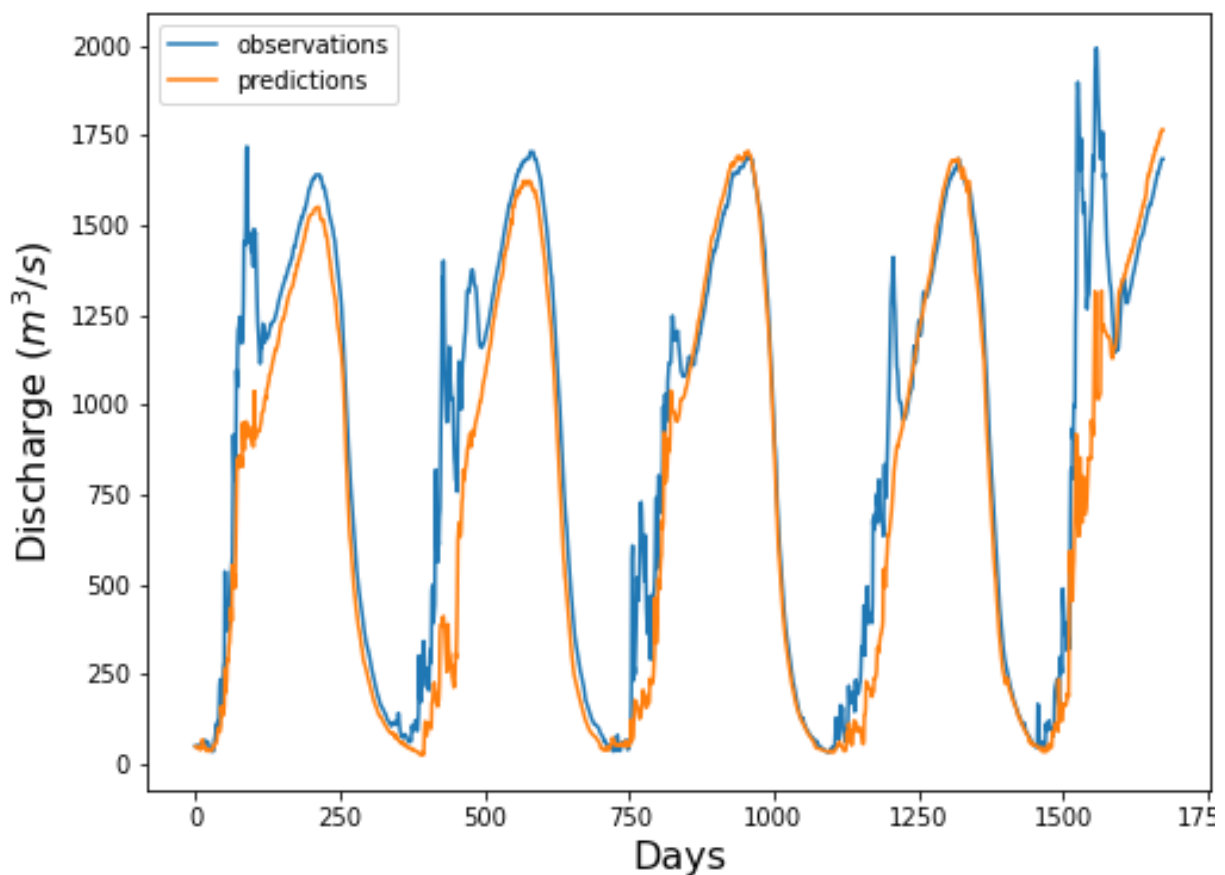
The next figure shows a plot of the loss function over the number of epochs during the training and the validation phase.



**Figure 3.5:** Evaluation of the LSTM model

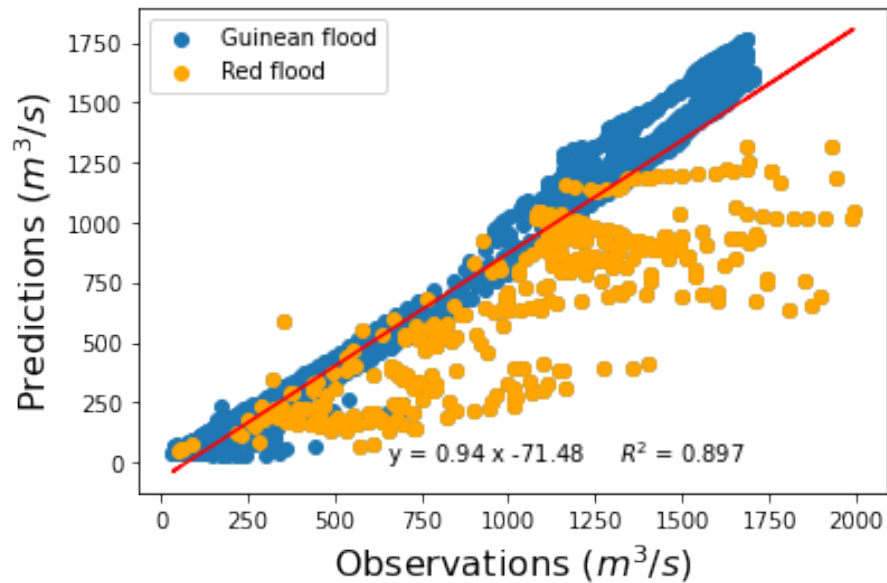
The trend of the loss curve shows that the model converged reasonably quickly and both train and test performance remained equivalent. The performance and convergence behavior of the model suggest that mean squared error is a good match for a neural network learning this problem. The validation curve being above the the training curve might suggest that the model is overfitting but the very small difference between the losses means that the model does not have significant overfitting issues. Moreover, the training/validation plot implies that the model presents low bias and low variance which attests that the model has a right balance and is neither overfitting nor underfitting.

The next step would be to evaluate our LSTM model over the test set of data. The figure below presents the observed discharge vs the predicted discharge.



**Figure 3.6:** Predictions of the LSTM model from June 2006 to December 2010

When presented with the LSTM model with a new set of data (figure above), we notice that the Guinean flood is well simulated while the red flood is not well simulated. The peaks of the red flood are underestimated by the deep learning model. This may be explained by the fact that maybe the right set of climate variables were not included in the model because the red flood originates from the climatic conditions within the Ansongo-Niamey basin. While using this model, the best thing to do would be to overestimate the first peaks show by the LSTM model. The next scatter plot below illustrates this better.



**Figure 3.7:** Predictions of the LSTM model with type of flood

The Guinean flood is slightly overestimated while the red flood is considerably underestimated. The slope of the comparison between the observed values and the predicted ones is 0.94 and its intercept is -71.48.

The  $R^2$ , NSE and RMSE calculated were respectively 0.897, 0.852 and 229.158. This confirms the fact that the LSTM model is overall good at simulating discharge in general and particularly at the outlet of the Ansongo-Niamey basin.

## 3.2 Discussion

Even though there have been some rare articles were found on the use of ANN in West Africa, there was no published work found on using ANN to predict streamflow in Niamey. Therefore, the discussion would be centered on identifying physical-based hydrological models studied in Niamey, their input data and their performance. Next, there would be an assessment of our LSTM model compared to other hydrological models. At the end of this section, we would suggest improvements to be done to this model based on the research hypothesis.

Jafet Andersson et al. (2017) studied the peak flow statistics and forecasting in the Niger River basin [6]. The goal of that study was to estimate peak river flow statistics (the discharge magnitude with a 100-year return period) across the Niger River basin. To achieve this goal, the HYPE (HYdrological Predictions for the Environment) model was set up and adapted to the Niger River basin. The model was based on openly available data on topography, land use, soil, lakes, reservoirs and daily air temperature and precipitation from the WFDEI dataset. The Niger-HYPE model had a good performance with a NSE of 0.72 at Niamey over the validation phase from 1995 to 2010.

Claire Casse et al. (2015) studied the impact of rainfall forcing on the flooding of the Niger River in Niamey [7]. They used the ISBA–TRIP (Interaction between Soil Biosphere and Atmosphere – Total Runoff Integrating Pathway) model on the Ansongo-Niamey basin. This model used 8 different climate variables with a 3 hourly and  $0.5^{\circ} \times 0.5^{\circ}$  resolution and daily discharge at Ansongo and Niamey as main input data. The climate variables were the solar radiation, the atmospheric radiation, the liquid precipitation, the solid precipitation, the air temperature at 2m altitude, the horizontal wind speed at 10m altitude, the specific humidity of the air at 2m altitude and the air pressure at the surface. This model performed very well with a NSE of 0.93 over the period from 2003 to 2012.

Other models were calibrated and validated in West Africa for simulating discharge at Niamey such as the SWAT (Soil and Water Assessment Tool) model [8] and the HGS (HydroGeoSphere) model [9]. The SWAT model had a bad performance at Niamey while the HGS simulated better the Niger river's discharge at Niamey. Only the HGS peaks had the most errors.

From all the information above, we can say that physically based models deployed in Niamey had a good performance and are capable of addressing Niger's discharge at Niamey. Our LSTM model also performs equally as those models with way fewer input data. The execution of the deep learning model required low computational resources (was run on a CPU in less than 5 minutes). Only the hyperparameter optimization process was time-consuming because of the number of epochs set as a hyperparameter. The LSTM networks are indeed promising in the field of hydrology.



## **Chapter 4**

# **Conclusion and Perspectives**

#### 4. CONCLUSION AND PERSPECTIVES

---

This research can be considered as a first step in the application of artificial intelligence in hydrological modeling in West Africa. It investigated the potential of using Long Short-Term Memory networks (LSTMs) for simulating discharge from climate and hydrological observations. LSTMs are special type of recurrent neural networks with an internal memory that can learn and store long-term dependencies of the input-output relationship. This work proposed to find the optimal hyperparameters for our LSTM model and to deploy his model to predict discharge at the outlet of the Ansongo-Niamey basin. This work confirms that the LSTM model trained, validated and tested achieved high accuracy and efficiency and low computational cost. The main results of this research are that:

- The optimal hyperparameters of the LSTM found were of:
  - $1.5616 \cdot 10^{-2}$  for the learning rate,
  - 30 for the number of LSTM units,
  - 500 for the number of epochs and
  - 128 for the batch size.
- The LSTM model had “very good” a performance with a  $R^2$  of 0.897, a NSE of 0.852 and a RMSE of 229.158
- The model slightly overestimates the Guinean flood and the red flood of the Ansongo-Niamey basin is considerably underestimated.

The following is the verification of the hypothesis of this research:

- **H1:** The assumption that the learning rate, the number of LSTM units, the number of epochs and the batch size of the LSTM model were the model’s main hyperparameters was verified. It might be interesting to maintain the number of epochs and the batch size and to replace them with other hyperparameters in the optimization phase.
- **H2:** Precipitation, maximum temperature, minimum temperature and discharge at Ansongo and Kandadji were sufficient inputs of the LSTM model for the effective simulation of Niamey’s discharge. Meanwhile, the red flood could have been better simulated if additional climate variables were added to the model.
- **H3:** LSTM model matched and even outperformed classical hydrological models at predicting historical river flow at Niamey.

#### 4. CONCLUSION AND PERSPECTIVES

---

Deep learning methods presents undeniable advantages in hydrology. In other to exploit neural networks to a full extent in hydrology, important perspectives are needed to be pursued to further the present work:

- see how the weights, bias, gates and cells of the LSTM model behaved in other to understand what information was learned and if possible link them to observation data
- add climate variables to the precipitation and the maximum and minimum temperature to get a better simulation of the red flood.
- use the Gated Recurrent Unit neural networks instead for LTMSs to see if there would be an improvement

Those improvements would make the DL model seem less of a black box and would increase trust in data-driven approaches in general and in DL methods in particular. The next step after that would be to regionalize a DL model over West Africa to simulate ungauged catchments. This regionalization could integrate static catchment attributes to the LSTM model and thereby enhance the explainability of the model. Artificial Neural Networks (ANNs) generally perform better in the industry when they are coupled with other models. Associating 2 ANN or 1 ANN with a physical-based model could be also an exciting prospect.

Deep learning is now a common and generally accepted technology, its use in hydrology help to achieve hydrological information, efficient, and correct choices meanwhile all this could also be improved.

This research would suggest to the scientific community (WASCAL, NBA, AGRHYMET and African researchers) the potential of using Long Short-Term Memory for simulating surface runoff. In the case of implementation of an AI technical component of flood early warning systems in Niger, much more reduction of economic losses and mitigation of the number of injuries or deaths from a disaster would be done. Accurate and precise information that allows individuals and communities to protect their lives and property would be provided on time because early warning information empowers people to take action before a disaster.

# **Bibliography**

- [1] Christopher B Field and Vicente R Barros. *Climate change 2014—Impacts, adaptation and vulnerability: Regional aspects*. Cambridge University Press, 2014.
- [2] Valentin Aich, Bakary Kone, Fred F Hattermann, and Eva N Paton. Time series analysis of floods across the niger river basin. *Water*, 8(4):165, 2016.
- [3] International Federation of Red Cross and Red Crescent Societies. Niger: Floods - emergency plan of action (EPoA) - DREF operation n° MDRNE024 - niger.
- [4] Duminda Perera, Ousmane Seidou, Jetal Agnihotri, Mohamed Rasmy, Vladimir Smakhtin, Paulin Coulibaly, and Hamid Mehmood. Flood early warning systems: A review of benefits, challenges and prospects. *United Nations Univ. Inst. Water, Environ. Heal.*, 2019.
- [5] Reid Basher. Global early warning systems for natural hazards: systematic and people-centred. *Philosophical transactions of the royal society a: mathematical, physical and engineering sciences*, 364(1845):2167–2182, 2006.
- [6] Jafet CM Andersson, Abdou Ali, Berit Arheimer, David Gustafsson, and Bernard Minoungou. Providing peak river flow statistics and forecasting in the niger river basin. *Physics and Chemistry of the Earth, Parts A/B/C*, 100:3–12, 2017.
- [7] Claire Casse. *Impact du forçage pluviométrique sur les inondations du fleuve Niger a Niamey. Etude a partir de données satellitaires et in-situ*. PhD thesis, Université de Toulouse, Université Toulouse III-Paul Sabatier, 2015.
- [8] Thomas Pomeon, Bernd Diekkruger, Anne Springer, Jurgen Kusche, and Annette Eicker. Multi-objective validation of swat for sparsely-gauged west african river basins—a remote sensing approach. *Water*, 10(4):451, 2018.
- [9] Boubacar Abdou Boko, Moussa Konate, Nicaise Yalo, Steven J Berg, Andre R Erler, Pibgnina Bazie, Hyoun-Tae Hwang, Ousmane Seidou, Albachir Seydou Niandou, Keith Schimmel, et al. High-resolution, integrated hydrological modeling of climate change impacts on a semi-arid urban watershed in niamey, niger. *Water*, 12(2):364, 2020.
- [10] Chaopeng Shen. A transdisciplinary review of deep learning research and its relevance for water resources scientists. *Water Resources Research*, 54(11):8558–8593, 2018.
- [11] Spyros Makridakis. The forthcoming artificial intelligence (ai) revolution: Its impact on society and firms. *Futures*, 90:46–60, 2017.
- [12] Ajay Shrestha and Ausif Mahmood. Review of deep learning algorithms and architectures. *IEEE Access*, 7:53040–53065, 2019.
- [13] J. Brownlee. *Deep Learning for Time Series Forecasting: Predict the Future with MLPs, CNNs and LSTMs in Python*. Machine Learning Mastery, 2018.

- [14] Sina Ardabili, Amir Mosavi, Majid Dehghani, and Annamária R Várkonyi-Kóczy. Deep learning and machine learning in hydrological processes climate change and earth systems a systematic review. In *International Conference on Global Research and Education*, pages 52–62. Springer, 2019.
- [15] Z. Wang, D. Xiao, F. Fang, R. Govindan, C. C. Pain, and Y. Guo. Model identification of reduced order fluid dynamics systems using deep learning. *International Journal for Numerical Methods in Fluids*, 86(4):255–268, 2018.
- [16] Muhammed Sit, Bekir Z Demiray, Zhongrun Xiang, Gregory J Ewing, Yusuf Sermet, and Ibrahim Demir. A comprehensive review of deep learning applications in hydrology and water resources. *Water Science and Technology*, 82(12):2635–2670, 2020.
- [17] Jinbo Qin, Ji Liang, Tao Chen, Xiaohui Lei, and Aiqing Kang. Simulating and predicting of hydrological time series based on tensorflow deep learning. *Polish Journal of Environmental Studies*, 28(2), 2019.
- [18] Bruce G Buchanan. A (very) brief history of artificial intelligence. *Ai Magazine*, 26(4):53–53, 2005.
- [19] Tom Standage. *The Turk: The life and times of the famous eighteenth-century chess-playing machine*. Walker & Company, 2002.
- [20] FutureLearn. The journey of AI - digital skills: Artificial intelligence - accenture.
- [21] Christine Chien. History of artificial intelligence.
- [22] Santanu Pattanayak, Pattanayak, and Suresh John. *Pro deep learning with tensorflow*. Springer, 2017.
- [23] Jason Brownlee. A gentle introduction to backpropagation through time.
- [24] Ralf C Staudemeyer and Eric Rothstein Morris. Understanding lstm—a tutorial into long short-term memory recurrent neural networks. *arXiv preprint arXiv:1909.09586*, 2019.
- [25] CS231n: Convolutional neural networks for visual recognition online course video lectures by stanford.
- [26] Ramazan Gencay and Tung Liu. Nonlinear modelling and prediction with feedforward and recurrent networks. *Physica D: Nonlinear Phenomena*, 108(1-2):119–134, 1997.
- [27] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994.
- [28] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In Sanjoy Dasgupta and David McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, Proceedings of Machine Learning Research, pages 1310–1318, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR.

- [29] Gayathri K Devia, B Pa Ganasri, and G Sa Dwarakish. A review on hydrological models. *Aquatic Procedia*, 4:1001–1007, 2015.
- [30] Robert J Abrahart, François Anctil, Paulin Coulibaly, Christian W Dawson, Nick J Mount, Linda M See, Asaad Y Shamseldin, Dimitri P Solomatine, Elena Toth, and Robert L Wilby. Two decades of anarchy? emerging themes and outstanding challenges for neural network river forecasting. *Progress in Physical Geography*, 36(4):480–513, 2012.
- [31] Youchuan Hu, Le Yan, Tingting Hang, and Jun Feng. Stream-flow forecasting of small rivers based on lstm. *arXiv preprint arXiv:2001.05681*, 2020.
- [32] Hamidreza Ghasemi Damavandi, Reepal Shah, Dimitrios Stampoulis, Yuhang Wei, Dragan Boscovic, and John Sabo. Accurate prediction of streamflow using long short-term memory network: a case study in the brazos river basin in texas. *International Journal of Environmental Science and Development*, 10(10):294–300, 2019.
- [33] Bibhuti Bhusan Sahoo, Ramakar Jha, Anshuman Singh, and Deepak Kumar. Long short-term memory (lstm) recurrent neural network for low-flow hydrological time series forecasting. *Acta Geophysica*, 67(5):1471–1481, 2019.
- [34] Frederik Kratzert, Daniel Klotz, Claire Brenner, Karsten Schulz, and Mathew Herrnegger. Rainfall–runoff modelling using long short-term memory (lstm) networks. *Hydrology and Earth System Sciences*, 22(11):6005–6022, 2018.
- [35] Frederik Kratzert, Daniel Klotz, Guy Shalev, Günter Klambauer, Sepp Hochreiter, and Grey Nearing. Towards learning universal, regional, and local hydrological behaviors via machine learning applied to large-sample datasets. *Hydrology and Earth System Sciences*, 23(12):5089–5110, 2019.
- [36] Seydou B. Traore, Abdou Ali, Seydou H. Tinni, Mamadou Samake, Issa Garba, Issoufou Maigari, Agali Alhassane, Abdallah Samba, Maty Ba Diao, Sanoussi Atta, Pape Oumar Dieye, Hassan B. Nacro, and Kouamé G.M. Bouafou. Agrhymet: A drought monitoring and capacity building center in the west africa region. *Weather and Climate Extremes*, 3:22–30, 2014. High Level Meeting on National Drought Policy.
- [37] Claire Casse. *Impact du forage pluviométrique sur les inondations du fleuve Niger a Niamey. Etude a partir de donnees satellitaires et in-situ*. PhD thesis, Université de Toulouse, Université Toulouse III-Paul Sabatier, 2015.
- [38] Scikit optimize project. Scikit-optimize: sequential model-based optimization in python — scikit-optimize 0.8.1 documentation.
- [39] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
- [40] Google. Guide | TensorFlow core.
- [41] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.

# Appendix



**Table 4.1:** Descriptions of aquifer type categories

Category	Definition	Attribute Field Code
Unconsolidated Sedimentary	Unconsolidated sedimentary aquifers with dominantly intergranular flow	U
Consolidated Sedimentary Fracture	Aquifers with dominantly fracture flow	CSF
Consolidated Sedimentary Intergranular	Aquifers with dominantly intergranular flow	CSI
Basement	Crystalline basement aquifers with typical weathered/fractured aquifer properties	B

Source: British Geological survey (2019)

**Table 4.2:** Descriptions of aquifer productivity categories

Category	Approximate range in yield (L/s)	Attribute Field Code
High	5 – 20	H
Moderate	2 – 5	M
Low to Moderate	0.5 – 2	LM
Low	0.1 – 0.5	L
Very Low	< 0.1	VL

Source: British Geological survey (2019)

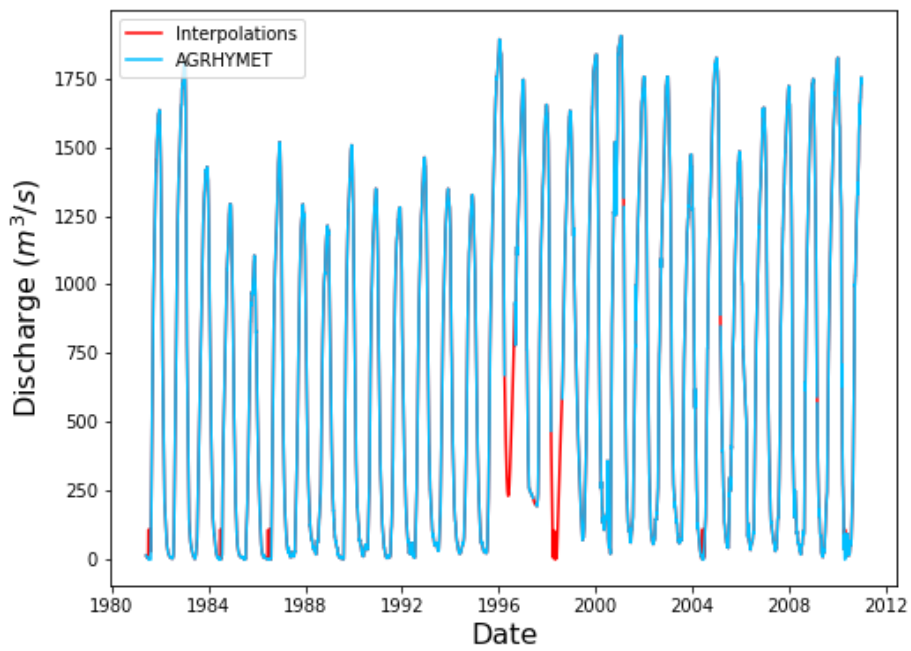


Figure 4.1: Missing data at Ansongo

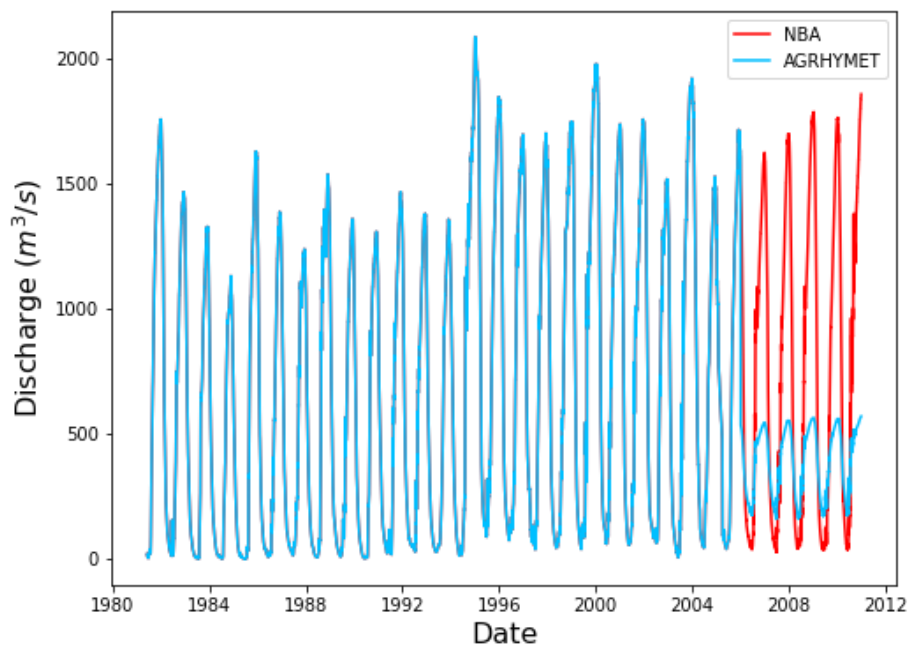


Figure 4.2: Missing data at Kandadji

**Table 4.3:** Hyperparameters values tuned with Scikit-Optimize

Loss	Learning rate	Units	Epochs	Batch size
0.004136530216783285	0.01561601307656165	30	500	128
0.004155533853918314	0.01	100	100	100
0.004326516762375832	0.4334862687432915	115	92	379
0.004370750393718481	0.02958197314641175	488	60	192
0.004542979411780834	0.1277294961686797	7	458	305
0.004700458608567715	0.0018950585350399308	189	351	70
0.0047556255012750626	0.001	407	258	76
0.004877073690295219	0.001	1	500	1
0.004932855721563101	0.001	425	428	26
0.004972529131919146	1.0	500	500	500
0.005517391487956047	0.4919967434016415	441	500	182
0.006089142058044672	0.032242457725426804	89	278	463
0.006123912986367941	0.017697684634001537	276	107	307
0.006264571100473404	0.0011624505344010765	236	500	500
0.0064185503870248795	0.02201991410428172	41	213	366
0.006625488866120577	0.001	480	500	150
0.006798347923904657	0.001	449	440	418
0.007092648185789585	0.12159090639428773	122	394	286
0.007519216276705265	0.001	1	199	486
0.00759211415424943	0.001	357	500	194
0.007792661432176828	0.0027274729876038523	119	487	238
0.00807427242398262	0.001	384	58	387
0.008278263732790947	0.31576237166334437	318	235	140
0.00930437259376049	0.21031953265427586	232	239	473
0.009309734217822552	0.01944153922922999	55	442	450
0.009427351877093315	0.02678837355906454	203	372	493
0.012887410819530487	0.004376482956726247	496	32	470
0.014559315517544746	1.0	209	149	246
0.0433134064078331	0.0027248552600231493	20	1	143
0.05107715725898743	0.005029621261430646	1	253	1
0.08007945865392685	0.01438861733940544	455	151	1
0.08410190790891647	0.001	170	1	483
0.10518210381269455	1.0	331	500	89
0.17476697266101837	0.7165448743509659	14	3	18
0.181575745344162	1.0	176	1	500
0.22631193697452545	0.24275089163011399	340	439	1
0.32057997584342957	1.0	99	329	43
0.3514423668384552	0.0350179548277067	500	500	1
0.4032553434371948	0.038801277679289814	1	1	1
1.712756872177124	1.0	475	1	337

# **Table of Contents**

## TABLE OF CONTENTS

---

Summary . . . . .	i
Dedication . . . . .	ii
Acknowledgments . . . . .	iii
Abstract . . . . .	iv
Acronyms and Abbreviations . . . . .	v
List of Tables . . . . .	vi
List of Figures . . . . .	vii
<b>1 Introduction</b>	<b>1</b>
1.1 Context . . . . .	2
1.2 Justification of Study . . . . .	3
1.3 Literature review . . . . .	5
1.3.1 Short history on AI . . . . .	5
1.3.2 Artificial Intelligence - Machine Learning - Deep Learning . . . . .	7
1.3.3 Inspiration behind Artificial Neural Networks . . . . .	8
1.3.4 Perceptron . . . . .	9
Concept . . . . .	9
Limits . . . . .	11
1.3.5 Feed Forward Neural Network . . . . .	12
Concept . . . . .	12
Limits . . . . .	14
1.3.6 Recurrent Neural Networks . . . . .	15
Concept . . . . .	15
Limits . . . . .	16
1.3.7 Long Short-Term Memory Neural Networks . . . . .	16
Applications of LSTM NN in streamflow prediction . . . . .	16
1.4 Objectives . . . . .	21
1.4.1 Hypothesis . . . . .	21
1.4.2 Main Objective . . . . .	21
1.4.3 Specific Objectives . . . . .	21
1.5 Host Structure . . . . .	22
<b>2 Methodology</b>	<b>24</b>
2.1 Study Area . . . . .	25
2.1.1 Ansongo-Niamey Basin . . . . .	25
2.1.2 Administration and Human Activities . . . . .	25
2.1.3 Hydrology . . . . .	29
2.1.4 Climatology . . . . .	31
2.1.5 Geology and Hydrogeology . . . . .	32
2.2 Data . . . . .	35
2.3 Tools . . . . .	37
2.4 Methods . . . . .	38
2.4.1 Data Preprocessing . . . . .	40
2.4.2 Hyperparameter Optimization . . . . .	41

*TABLE OF CONTENTS*

---

2.4.3	LSTM Model . . . . .	43
	Concept . . . . .	43
	Neural Network Data Preprocessing . . . . .	44
	Setting up the LSTM model . . . . .	46
	Training and evaluation of LSTM model . . . . .	46
<b>3</b>	<b>Results and Discussion</b>	<b>48</b>
3.1	Results . . . . .	49
	3.1.1 Hyperparameter Optimization . . . . .	49
	3.1.2 LSTM Model . . . . .	52
3.2	Discussion . . . . .	55
<b>4</b>	<b>Conclusion and Perspectives</b>	<b>56</b>
	<b>Bibliography</b>	<b>59</b>
	<b>Appendix</b>	<b>I</b>
	<b>Table of Contents</b>	<b>V</b>