Hydroinformatics and Applications of Artificial Intelligence and Machine Learning in Water-Related Problems

Corzo Perez, Gerald A.; Solomatine, Dimitri P.

DOI
10.1002/9781119639268.ch1

Publication date
2024

Document Version
Final published version

Published in
Advanced Hydroinformatics

Citation (APA)
Corzo Perez, G. A., & Solomatine, D. P. (2024). Hydroinformatics and Applications of Artificial Intelligence and Machine Learning in Water-Related Problems. In G. A. Corzo Perez, & D. P. Solomatine (Eds.), *Advanced Hydroinformatics: Machine Learning and Optimization for Water Resources* (pp. 1-38). (Special Publications; No. 78). AGU Publications. https://doi.org/10.1002/9781119639268.ch1

1

HYDROINFORMATICS AND APPLICATIONS OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING IN WATER-RELATED PROBLEMS

Gerald A. Corzo Perez[1] **and Dimitri P. Solomatine**[1,2,3]

[1] *IHE Delfit Institute for Water Education, Delfit, The Netherlands*
[2] *Water Resources Section, Delfit University of Technology, Delft, The Netherlands*
[3] *Water Problem Institute of the Russian A 'ademy of S 'iences, Moscow, Russia*

In recent years, there has been a surge of interest in machine learning (ML) and artificial intelligence (AI) due to the effectiveness of deep learning algorithms and the increasing availability of large data sets. This chapter provides a brief overview of the applications of AI and ML techniques in hydroinformatics, a field that deals with advanced information technology, data analytics, and modeling for aquatic environment management. Data-driven models are becoming more common in water management as they can reveal hidden patterns in data and offer improved accuracy in certain situations. This chapter highlights the importance of spatiotemporal data analysis, pattern recognition, and optimization approaches in water resources management under uncertainty. It does not offer a comprehensive review of all methods but rather focuses on selected ML techniques widely used in water-related problems. Additionally, the

Advanced Hydroinformatics: Ma hine Learning and Optimization for Water Resour 'es, Special Publication 78
Fir t Edition. Edited by Gerald A. Corzo Perez and Dimitri P. olomantine.

DOI: 10.1002/9781119639268. h1

chapter discusses the challenges associated with using ML models, such as black-box criticisms, and the potential of hybrid models that combine the strengths of ML and physically based process models for more robust solutions in hydroinformatics.

1.1. Introduction

Hydroinformatics deals with advanced information technology, data analytics, modeling artificial intelligence (AI), and optimization applied to problems of aquatic environment for the purpose of informing management. Many of these technologies have become standard tools that support water management decisions around the world. However, the technologies are developing further, new ones are emerging, and this allows for applying them to more complex and interesting problems. One can find multiple examples when environmental and hydrological problems have been dealt with not only by employing physically based (process) models, but also advanced data analysis tools and machine learning models have been used. Using AI techniques in geosciences has a long history. Hydroinformatics formulated by Abbott (1991) 30 yr ago, has been defined as a union of computational hydraulics (CH) and AI (so that $\mathrm{HI} = \mathrm{CH} \cup \mathrm{AI}$), and during the last three decades we have been witnessing a much wider use of AI, with a large number of successful practical applications. The first stage of such development ha been covered, for example, in the edited volume *Practical hydroinformatics: Computational intelligence and technological developments in water applications* (Abrahart et al., 2008), and in dozens of other books and hundreds of research papers covering these new developments.

Currently, we see a new wave of interest in machine learning (ML) and AI, which is partly explained by the demonstrable effectiveness of the new generation of deep learning algorithms and availability of large data sets (see, e.g., Nearing et al., 2021), and this brings new possibilities for hydroinformatics research and practice. With an increasing amount of data collected about the environment, physically based models are more and more complemented and sometimes even replaced by data-driven models. Lacking the ability of physically based models to explain the physics of underlying processes, data-driven models are however able to discover the hidden patterns in data and often can be more accurate, and play an important supporting role, in water management. Pattern recognition (e.g., automatic identification of flooded areas on satellite images) has been one of the main tasks solved by machine learning and lately has been given an additional push by the development and use of deep learning an important class of machine learning algorithms, and of AI in general. Data analytics plays an important role in water resources when data are multidimensional and spatial and time dimensions have to be dealt with in a coordinated fashion. In relation to water resources both dimensions were always important, but recently the need to handle huge amounts of remote sensing data ("big data") has become more pronounced. These developments have motivated new research efforts in the context of predicting hydrological extremes and call for te ting novel approaches of spatiotemporal data analysi and machine learning. Due to much easier acce s to supercomputing facilities, there are increased possibilities to study the models uncertainty (typically using Monte Carlo frameworks), and machine learning can also play a role in building predictive models of such uncertainties. An issue in water resources management is optimal planning and operation under uncertainties, and this is where the role of AI-driven approaches is also becoming more important. Classical optimization approaches (gradient-based nonlinear optimization) typically cannot help much, since such optimization is model based, and objective functions (and their gradient) cannot be analytically expres ed. Optimization approaches developed under the framework of computational intelligence (various types of randomized search e.g. evolutionary approaches) have been the focus of hydroinformatics for three decades, but the new problems and the increased data availability lead to the necessity of testing new approaches and their critical analy is.

This chapter aims at presenting a brief overview of AI- and ML-related building processes and methods widely used for water-related problems, in the context of the chapters presented in this volume. AI is a concept that covers a wide area of science and technology, however. quite often it is used interchangeably with ML, which is in fact a narrower notion. One may find in literature quite a large number of AI- and ML-related subareas: big data data mining, pattern recognition (PR), natural language processing (NLP) neural networks deep learning, and so on. We will not go into a discussion about terminology and differences in AI and ML· for the purpose of this chapter and the issues covered in the book, it would be right to use a somewhat narrower term, that is, machine learning.

ML techniques have been widely used in water resources during the last decades however. at the same time one may observe also inadequate use of ML-related modeling procedures, unjustified selection of algorithms, and even lack of understanding of why a model provide good or poor performance in mathematical and statistical sense. There is al o well-known criticism of ML and statistical techniques by practitioners who are used to employing physically based (process) models· they are pointing out that a water resources problem interpretation i hidden in the so-called black box of a ML model. There is indeed a challenge of posing the problem in the right way: how domain knowledge can drive election building, and tuning a ML model. Lack of data and its uncertainty also makes it difficult for practitioners to feel confident about ML models.

On the other hand, the strength of ML is in its ability to represent the relationships between inputs and outputs, provided enough data are available. Although the relatively recent advances in deep learning have opened the door to the new ways of using spatiotemporal data and at the same time motivating new algorithm development from spatial patterns and in general, all types of computer vision algorithm , not all problems can be tackled by ML. Input and output relations can be so complex that ML techniques may not be able to find the hidden patterns, and in such cases hybrid models combining power of ML and process models (so-called physics-aware AI· see, e.g., Jiang et al., 2020) would be needed. Such hybrid approaches are given now increased attention in hydroinformatics.

This chapter is not intended to provide a comprehensive review of methods (which are covered in hundreds of books and in the referred literature herein), but rather focuses on some important elements of ML model building, and presents ba ics of several selected ML techniques quite widely used in solving water-related problems, allowing for feeling the flavor ' of ML.

1.2. Key Principles of ML/Hydroinformatics

1.2.1. AI and ML Definitions

There is a large number of evolving definitions of AI, and this can be explained by its permanent evolution and shifts in priorities and the advances in the used mathematical instruments. Many literature sources point out that for the first time the term *AI* was used in 1956 at the Dartmouth Conference, were John McCarthy, Alan Turing, and other founding fathers of AI, help to coin the term *artificial intelligence*. One of the definitions reads: "AI is the field devoted to building artificial animals (or at least artificial creatures that, in suitable context *appear* to be animals) and for many, artificial persons (or at least artificial reatures that, in suitable contexts, *appear* to be per ons)" (*Stanford Ency lopedia of Philosophy*, 2018). On the other hand, Wikipedia define it as the ' intelligence demonstrated by machines unlike the natural intelligence displayed by humans and animals' (Artificial Intelligence, 2022). Yet another definition (sometimes referred to as being given by IBM) state that “AI leverages computers and machines to mimic the problem-solving and decision-making capabilitie of the human mind. All the e definitions differ in details, but are very similar in the main idea: a machine (a programmed computer) i suppo ed to imitate some behavior of a living creature.

An old debate regarding whether humans will be replaced by machine has been reinitiated in variou public media in the view of the latest developments in AI, especially generative AI, a implemented for example in platforms like ChatGPT. Indeed AI has evolved into different types related to an extent to which it may take over ome of humans’ activities. The first ideas of what could be achieved are purely rea tive, which is highly related to the beginnings of computer science where AI does not have any memory which basically mean no initial data base or information of processes. This concept can be applied to solving narrow specialized tasks. For example, a forecast is performed based only on the current ituation, limited historical samples and known variables. Further development can lead to building up memory, by collecting previous experience and more complex and voluminous data and continue adding it to the memory. Such AI systems have enough memory or experience to support humans in performing various task but their ability is still limited and they are still seen as a helping hand. For example it can provide adaptive forecasts depending on the context such as previous performance, climatic conditions, type of a river basin and other . An even higher level of AI can be explained as a theory of mind (Premack & Woodruff, 1978) where AI can understand thoughts and em tions and interact ocially. Thi type of concept needs an integration of many component of AI development of more sophisticated mathemati al app ratus so uch developments are still at a rudimentary level. At the top level, it i po ible to con ider how these systems can become aware of life and even become elf-aware. This concept link to the idea that AI machine an create new knowledge and at the same time, build internal ystem concepts that link intelligence sentience, and consciousness.

Advances of AI have been numerous and applied in various areas. We should admit however, that in water resources, only a few of such developments have been used and these relate to application of pecific machine learning techniques.

Figure 1.1 presents a schematization of some of the key techniques of ML, with references to decades when these methods tarted to develop. Due to a wide application of ANN this ar hitecture is presented in more detail. One of the relatively new development is natural language processing (NLP)· it uses deep learning (DL) to train model that help

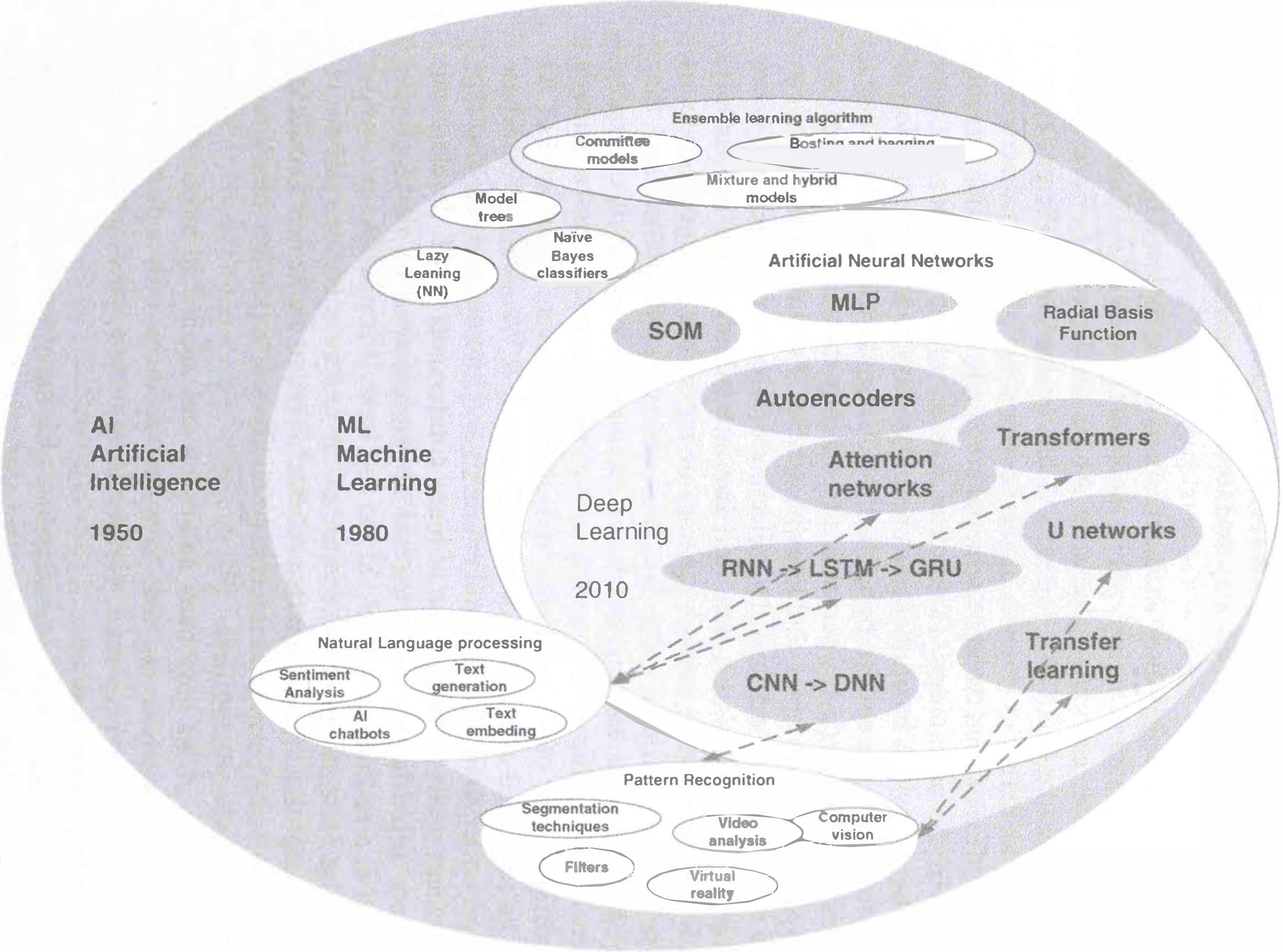

Figure 1.1 Evolution of AI topics: From artificial intelligence to machine learning and deep learning in hydroinformatics.

interpret text and reinforcement learning concept to use DL. Pattern recognition uses convolutions and DL, which develop pattern recognition to extract features. Finally metaheuristics provide the basis for new models of DL. The following are some of the concepts used in this chapter:

1. ML (machine learning). Mathematical models that aim to represent groups and/or input-output relationships from data
2. NLP (natural language processing). The use of language elements, in general, text encoded into numbers and its analysis, mainly from the transformation of text and processing it to solve, replicate semantics and understand them
3. Pattern recognition. ML can be characterized as a subarea that explores how data and their attributes (variables or features) can be detected. Many ML algorithms do implicitly detect patterns and therefore these areas are interrelated. Computer vision is an important area of their application. It is worth noting that a number of important pattern recognition mathematical apparatus and algorithms are not explicitly positioned in the machine learning realm for example, procedures of denoising and filtering, segmentation of images, 3D virtual reality patterns, vector fields flow, but they for sure contribute to solving the pattern recognition problems.

1.2.2. Machine Learning (ML)

There are various ways of contextualizing ML. From the perspective of computer science, the concept of ML can be seen as aiming at changing the programming paradigm (Fig. 1.2). Aim here is to develop a computer program that will not require significant analysis to understand how to create an algorithm to obtain certain responses· instead, a ML algorithm, theoretically can learn from inputs and responses (outputs).

In many applications, however, ML is not seen as a tool to generate computer programs but instead is expected to help in building input-output models by learning from data, in other words data-driven models (see Fig. 1.6). Their use is quite varied and most of the time is justified by the idea that a system might be very complex and we may not observe all the internal states of a modeled system or process (e.g. in hydrological modeling this may be soil moisture). This implies that if we have a complex system, with only a limited understanding of the driving variables of a natural process (or any process in general) and we can measure the consequences of events (i.e., outputs resulting from particular

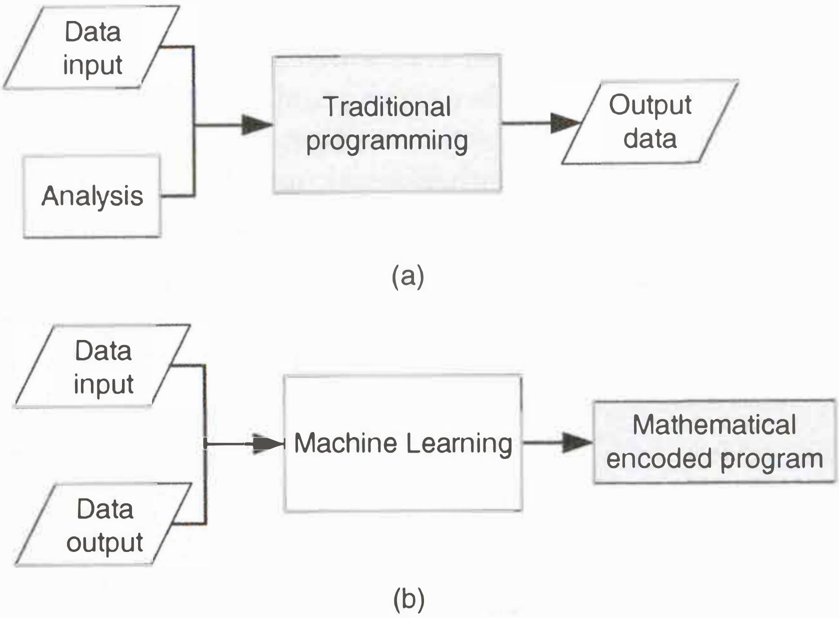

Figure 1.2 Differences between traditional programming and ML, as seen in computer science: (a) Computer science algorithm development; (b) Machine Learning algorithm development.

inputs), then with this information, it is possible to generate ML models (Fig. 1.3).

In most cases, the ML engines (e.g., artificial neural networks) work with numerical (real-valued) data, so are, in fact, nonlinear regression models. If data are nonnumerical (e.g., classes, images, or words), they have to be first transformed (encoded) into numerical form, and then processed.

In relation to Earth sciences, wide adoption of ML has not been fast, to say the least, since many scientists were pointing out that there is no clear justification for using these algorithms. Their reasoning was that the models, as descriptors of reality, should be based on scientific understanding of processes (e.g., physics), and not on a statistical encapsulation of data sets. Water resources are not an exception in this sense, and early applications of ML have been criticized, as they end up reproducing natural problems that do not need to be reproduced abstractly with ML, which was arguably resulting often in building a blind representation of a well-known problem. However, during the last two to three decades, there have been many examples of successful applications of ML reported and implemented in decision support systems. It has been shown that ML methods are often more accurate than the traditional hydrologic models in forecasting (see, e.g., Nearing et al., 2021; Arsenault et al., 2023). ML also helps to replace complex slow-running physically based models: a ML model is

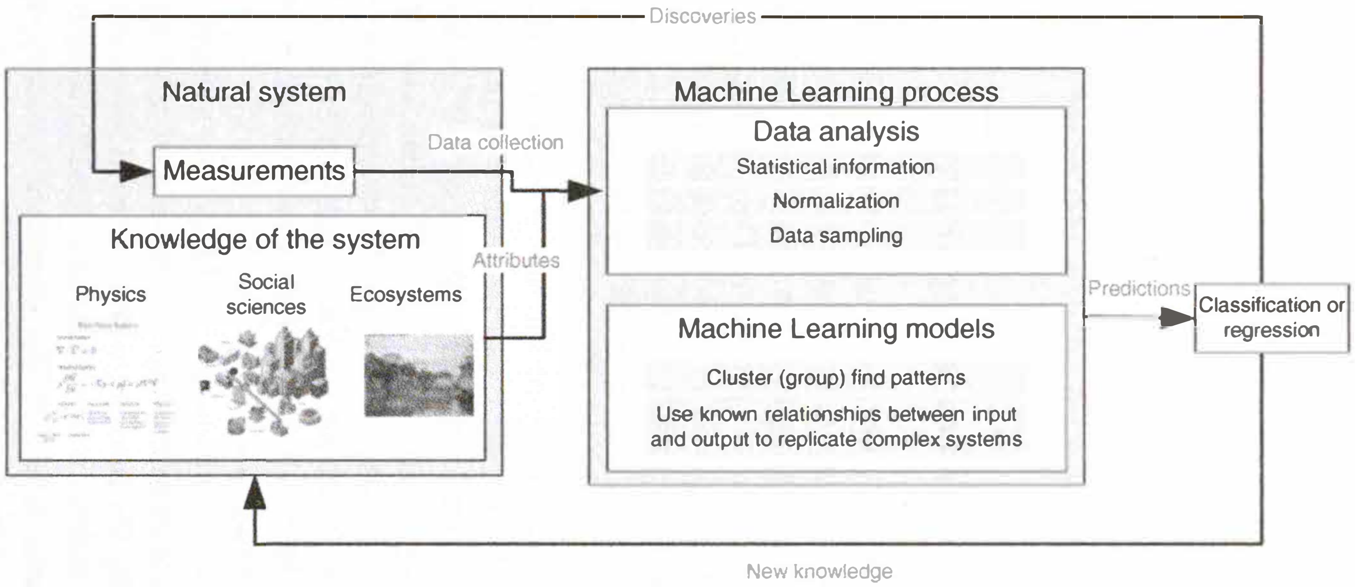

Figure 1.3 Encapsulation of natural systems and processes in ML models, with feedbacks.

trained on data generated by a process model and such fast metamodel (surrogate) would replace a much slower process model in operational systems and therefore be used in real-time forecasting to provide warnings in an efficient manner. ML-based pattern recognition algorithms can also help to automate the detection of critical scenarios, combining variable that might not be easily related physically and capturing nontrivial relationships and patterns implicitly present in data, reproducing thus complex phenomena. Therefore, ML has become a powerful analytical and predictive tool.

Aside from ML there are other areas in AI worth attention, that is natural language processing and metaheuristics, and they are also considered due to their potential for water resources management.

1.2.3. Natural Language Processing (NLP)

The Internet has allowed us to arrange access to billions of documents, images and audio and video material in very different areas of human activities. It would be interesting to understand if and how we can use these data for solving water resources problems. Data on the Internet are often not structured and linked to a variety of sources, from web ites of organizations, to social media, news, blogs, videos, and more. In many cases useful data are presented as text. The idea of text mining is not new; however, the large amount of data available on the Internet, in the form of text format, has generated a boom in developing intelligent tools, referred to as natural language processing (NLP). NLP can be defined as the ability of a computer program to understand human language as it is spoken and written, that is natural language (Sun et al., 2022). It starts with the idea of processing text and develops ways to interpret and reproduce it. The ways of understanding how we write have been formalized in tools for sentiment analysis of text, generation of text, correction of text text extraction, and concept of artificial assistants.

NLP converts letters, words and phrases into numerical representation. This is done sometimes in simple terms, like numbering each word in a phrase and repeating the number when the word repeats itself. However the results of this numerical representation need to follow the basics of the language (Khurana et al., 2023). Therefore, typically the process of interpreting the language focuses on five steps:

1. Lexical (morphological) analysis: In essence it i breaking text into paragraphs, phrases and words. Furthermore it is possible to understand at the level of individual words the morphemes as the smallest units of a word. Last, lexical analysis identifies the

morphemes and allows us to characterize the word and understand its meaning knowing its root form. The final objective of this step is to help to identify words which are normally referred to as tokens since the original word in fact possesses some information and, for programming, it is a sequence of characters which represent a unit of information.

2. Syntax analysis: it allows for checking the grammar and with this the way words are arranged in a sentence. As a consequence this order allows us to find how words should be normally arranged. Using this information it is possible to build relationships between them. Knowing this, it is possible to assess the parts of a sentence (POS) and tag this information based on the structure found.
3. Semantic analysis: This step aims at finding the meaning of the statement, how the phrase reads literally. This understanding provides the basis for rejecting syntactically valid but illogical statements.
4. Discourse integration: The context in which a phrase is used can be very important so this step aims at establishing links between the different sentences especially the immediately preceding one.
5. Pragmatic analysis this concept uses a set of rules that describe cooperative dialogs as in social content. What can be found in social media and common interactions can become a rule and with this we can comprehend the way the communication takes place.

NLP sentiment analysis in marketing for example has developed tools that allow us to provide basic knowledge extraction or information provider from how people feel about a product, and with this has opened numerous possibilities in decision making and understanding of people's behavior. In water resources several exploratory projects have been carried out at the IHE Delft Institute for Water Education aiming at understanding how people write about their water bodies in social media and newspapers (see Fig. 1.4). The concept presented here is based on media information on the Web on how people express (share) feelings in the form of text in the media. This concept applies to society in general and not only to citizens per se which will include how policy makers and news also influence the media and therefore citizens. This cycle has two parts since people tend to be susceptible to changing their feelings toward other people based on media information. With this the media is at the same time the engine that moves people but it also records all actions on citizens publications.

Application of NLP technologies resulted in development of AI chatbots. Conventional chatbots have evolved since 1966 (Weizenbaum

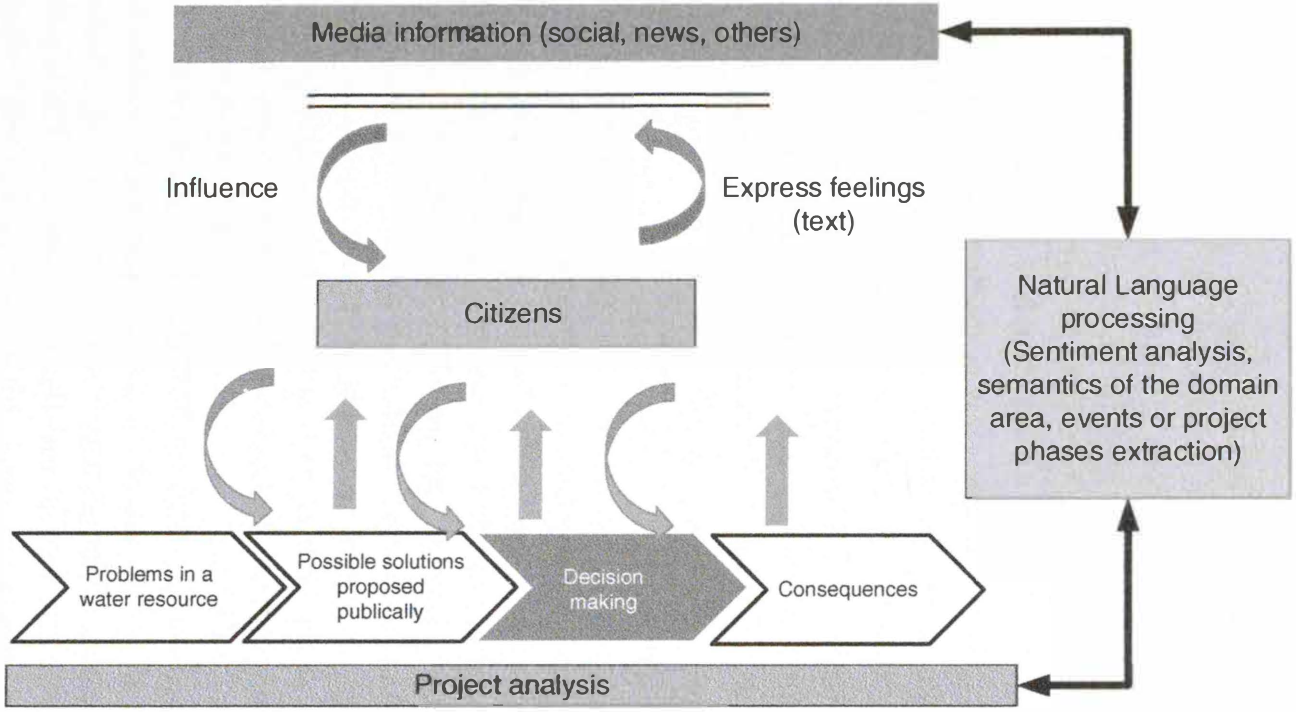
Media information (social, news, others)
Influence
Express feelings (text)
Citizens
Natural Language processing (Sentiment analysis, semantics of the domain area, events or project phases extraction)
Problems in a water resource
Possible solutions proposed publically
Decision making
Consequences
Project analysis

Figure 1.4 Interactions of digital information sources and citizens, and their relation to decision making in a project using NLP.

1966) when the first concept of a chat algorithm was conceived as chatterbot program ELIZA and was defined as a program designed to interact with people by simulating human conversation. Since then, there have been plenty of tools and publications using sequential, functional, database and object-oriented programming concepts. Most of these conventional programming techniques were too costly in development in terms of time and are not flexible enough to adapt to new data and new concepts. However, in the last 5 to 7 yrs technology using AI to improve the way we create chatbots has advanced considerably, and chatbots have been made widely available by their inclusion into standard smartphone software. Such chatbots serve as front-end engines to interpret human queries and then access knowledge bases to generate answers. These knowledge bases can encapsulate vast amounts of data stored on Internet servers worldwide, and they can be seen as ML systems trained on these data. Examples are Google Assistant (Google), Alexa (Amazon), and SIRI (Apple). In 2022–2023, the new noticeable implementations have been released: ChatGPT (OpenAI, a subsidiary of Microsoft), Bard (Google), and Ernie (Baidu).

1.2.4. Pattern (Image) Recognition

The area of image analysis and interpretation has been a focus in computer science and ML for several decades, and it is often referred to as pattern recognition. In this context, a pattern is an image that needs to be identified, interpreted, and classified. For example, recent research in atmospheric and hydrologic sciences has focused on how patterns in spatial data can be identified, likened to typical hydrometeorological events, and later used to predicts such events.

In water resources, pattern recognition techniques have been applied mainly to remote sensing (spatial data), finding flood patterns after extreme rainfall, identifying river networks and catchment land-use cover, learning from patterns of remote sensing bands. For example, in remote sensing, numerous pixels in a crop area are surveyed on site and then mapped to image bands. Having each sample as a training element to learn what the image band combinations are can be interpreted as this is quite important for obtaining the right spatial information about land cover and agricultural practices. As an example, recent works on determining the spatial extents of drought anomalies and the spatiotemporal patterns of rainfall that lead to floods can be mentioned (Corzo et al., 2018; Diaz et al., 2020; Khoshnazar et al., 2021; Varouchakis et al., 2021).

1.3. Model Building and Input Variable Selection in Machine Learning for Water-related Problems

ML procedures are described in detail in many book (e.g., Haykin, 1999). In relation to the use of ML in water-related problems most papers and books present similar modeling frameworks (Fig. 1.5). For example, Elshorbagy et al. (2010) compiled information about methodologies used in various studies, presenting a general-purpose framework, which was tested on several relevant water-related cases. Similar frameworks can be found also in more recent publications (e.g., Potgieter & Dahlberg, 2022).

A large number of procedures for designing data-driven models (learning systems) have been proposed in different areas. Some of these procedures have been generalized in such a way that they can be applied to other modeling approaches (Mitchell, 2007· Pyle, 1999; Abrahart et al., 2008). Corzo Perez (2009) presented the concept of characterizing the way models can include physical-based knowledge. A typical procedure of model building, presented, for example, by Haykin (1999) and many other authors, is as follows:

1. Explore the problem and solution spaces, and state the problem.
2. What is the expected result and how will it be used?
3. Select the input and output variables (features).
4. Specify the appropriate modeling methods and choose the tools (software and algorithms).
5. Prepare and survey the data. Partition the date into the training, verification and test subsets.
6. Build (train) the model, using training and validation subsets.
7. Test (validate) the model, using the test subset.
8. Apply the model and evaluate the results.

In reality, the process of modeling is not linear but continuous with feedback loops. For example, a lack of particular data may lead to a change in the modeling method selected. For these processes, there is a sort of checklist, or golden rules, widely accepted in the ML community, helping a modeler in the process of model building.

1. Clearly define the problem that the model will help to solve.
2. Specify the expected solution for the problem.
3. Evaluate if the delivered solution will be accepted and used in practice.
4. Learn the problem, collect the domain knowledge, analyze and understand it.

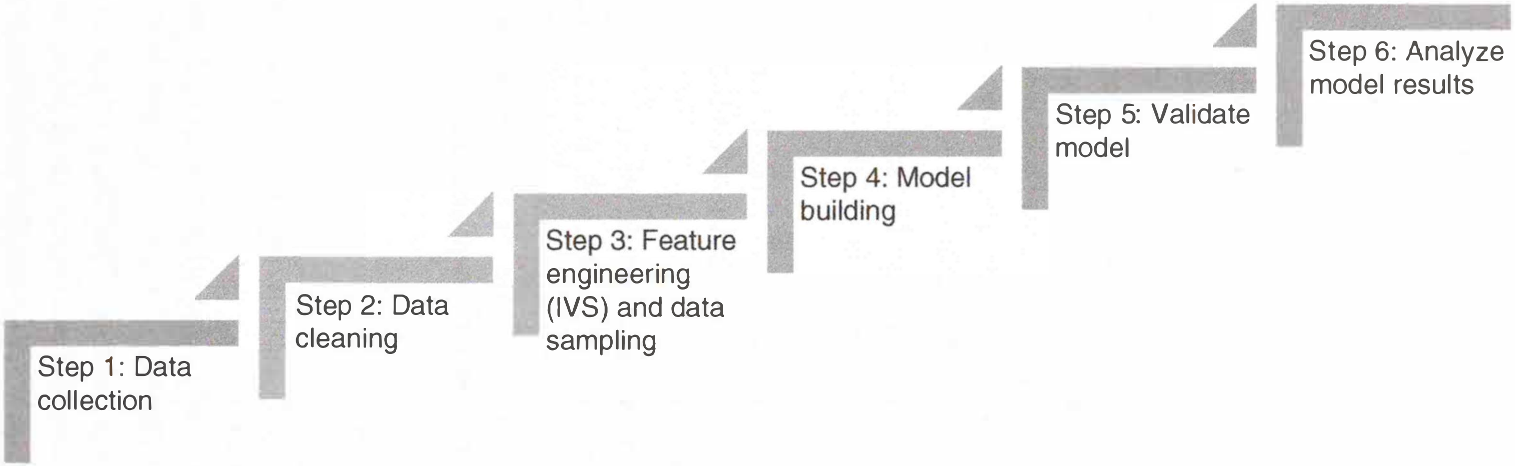

Figure 1.5 A typical sequence of steps in Machine Learning Modeling Framework.

5. Let the problem drive the modeling effort including selection of techniques, data preparation, and so on. Take the best tool for the job not just a job you can do with the available tool.
6. Clearly define assumptions and con traints, and discus them with domain knowledge experts.
7. Refine the model iteratively (try different things until the model seems as good as it is going to get).
8. Make the model as simple as possible but no simpler. This rule is formulated sometimes in different way , such as KISS for example (keep it sufficiently simple). Another formulation i the minimum description length principle which states that the best model is the smalle t (including the information to pecify both the form of the model and the values of the parameter). More generally this idea is widely known as the Occam's razor principle formulated by William of Occam in 1320 in the following form: have all the unneeded philosophy off the explanation.
9. Define instabilities in the model and its sensitivity (critical areas where small changes in inputs lead to large changes in output or even to model crash).
10. Define uncertainties in the model (critical areas and ranges in the data set where the model produces low-confidence predictions). Try to reduce such uncertaintie
11. Draw conclusions from model application.

The model training process (Fig. 1.6) is in fact solving an optimization problem: minimization of the difference between the mea ured and predicted results through the update of the model parameter erve as decision variables in this optimization problem.

1.3.1. Data Partitioning

In ML data are typically partitioned into three subsets which ideally should be statistically similar.

1. Training data: Data used to calculate the model error with the aim of its minimization by updating the model parameters.
2. Validation (or cross-validation) data: Data required for the intermediate evaluation of the model performance at various step of training. These data are used either to tune the model structure or to aid the early stopping of training to prevent overfitting. A common practice is also to use the so-called n-cross-validation procedure (n is often set

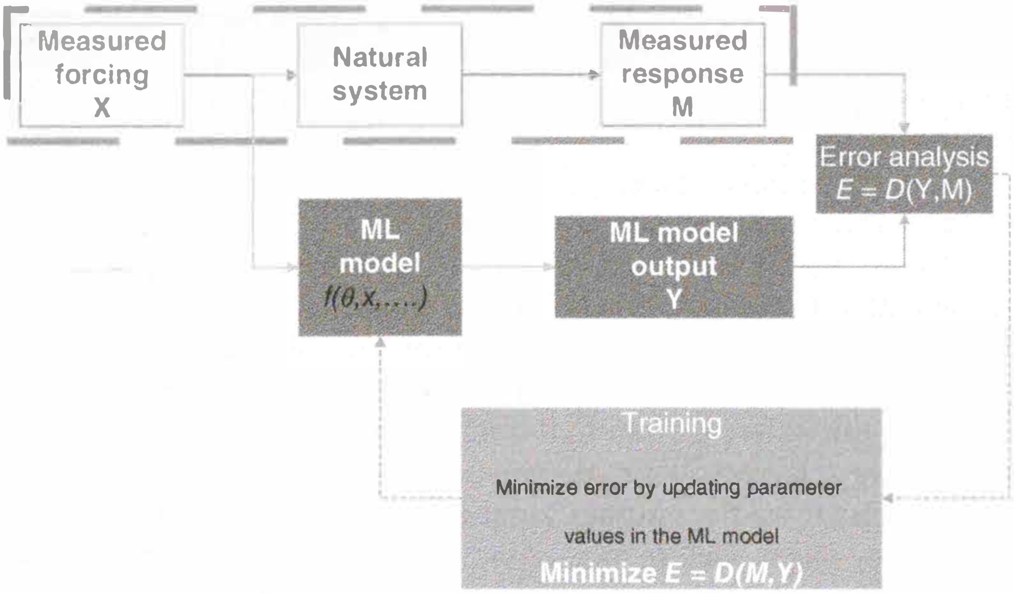

Figure 1.6 Training a Machine Learning model as an iterative optimization process. X is a vector of input variables (forcing); Θ is the parameters vector (decision variables); D is the error metrics to be minimized, the function measuring the difference between the predicted (Y) and the measured values (M).

to 10). In 10-fold cross-validation, 10 models are trained on 90% of the training set and validated on the remaining 10%. A similar performance of all models shows that the modeling process can be seen as successful, and any model can be taken as the final model, or the final model could be an ensemble of all these n trained models.

3. Testing (or verification) data: Data used for the final performance assessment of a model before its use. It requires a subset that has not been used during model training.

1.3.2. Input Variable Selection (IVS) (Feature Engineering)

It is important to ensure that a ML model would use the data and variables relevant for the objective of data analysis or modeling. Selecting the relevant variables (features) is referred to as feature engineering, or input variables selection (IVS). Input variables have to have relatedness to the output, and the simplest method here would be to use correlation analysis, or average mutual information (AMI). The correlation analysis reflects only linear relationships; therefore, when processes are highly nonlinear, AMI would be a better choice. A detailed presentation of IVS procedures can be found in Guyon and Elisseeff (2003), and, in relation to water modeling, in Bowden et al. (2005a, 2005b) and Galelli and Castelletti (2013).

One of the important steps in ML is data transformation, preprocessing and postprocessing of data (Pyle, 1999). For some of the methods, such as multilayer perceptron artificial neural networks (MLP ANNs), data transformation (normalization) is almost a must. Other methods benefit from this because it typically leads to improved performance.

In ML, the selection of input variables is of great importance and this process is quite different from that normally used in process-based modeling. Having a large volume of measurements and even knowing what input variables drive the output in physical terms, doe not guarantee the high accuracy of a model. In most cases, it is necessary to identify (1) when and (2) to what extent these inputs contribute to the model output. Since information about the modeled phenomena is not always available in ML, most methods rely on statistics and information theory to determine the appropriate input variables for data-driven models.

Normally, the input selection process starts with all the knowledge (data) about the process that will be modeled, and then the selection space is narrowed based on a more detailed analysis. In contrast to hydrological process-based models, ML allows for the inclusion of any variable (or its combination) even of those that do not necessarily force the phenomenon (discharge) directly. For example, rainfall runoff models may use past discharges to forecast current or future ones· however, they are not the actual triggers of flood situations (in relation to hydrological modeling see, e.g., discussion in Moreido et al., 2021).

To consider past information about the modeled phenomenon, often a model has to be fed with previous (lagged) values of an input variable, perhaps aggregated. In the context of hydrological foreca ting, these are typically precipitation and discharges optimally lagged. In this context lag is defined as the number of time steps by which a time series is shifted from its current value, itself (when autocorrelated), or relative to the corresponding time values of another time series (when cross-correlated). Table 1.1 presents an example of building an input data matrix with the lagged precipitation (current time is assumed to be 01/01/2015).

Understanding of how much delayed inputs are correlated with output is used to make a comparative graph of autocorrelation or cross-correlated variables (Fig. 1.7). In principle, all lags can be important to represent different types of responses due to the memory of the system. However, every

Table 1.1 Example of the input matrix generation for a rainfall-runoff model (Ourthe river basin).

Dates	Precipitation (mm)	Lag prec 1 day	Lag prec 2 day	Lag prec 3 day	Lag prec 4 day	Discharge (m^3/s)
01/01/2015	4.37					4.37
02/01/2015	11.72	4.37				11.72
03/01/2015	33.57	11.72	4.37			33.57
04/01/2015	32.20	33.57	11.72	4.37		32.20
05/01/2015	**8.20**	**32.20**	**33.57**	**11.72**	**4.37**	8.20
06/01/2015	0.56	8.20	32.20	33.57	11.72	0.56
07/01/2015	26.41	0.56	8.20	32.20	33.57	26.41
08/01/2015	28.84	26.41	0.56	8.20	32.20	28.84
09/01/2015	0.23	28.84	26.41	0.56	8.20	0.23
10/01/2015	0.77	0.23	28.84	26.41	0.56	0.77
11/01/2015	10.79	0.77	0.23	28.84	26.41	10.79
12/01/2015	1.27	10.79	0.77	0.23	28.84	1.27
13/01/2015	32.60	1.27	10.79	0.77	0.23	32.60
14/01/2015	37.05	32.60	1.27	10.79	0.77	37.05
15/01/2015	1.15	37.05	32.60	1.27	10.79	1.15
16/01/2015	0.16	1.15	37.05	32.60	1.27	0.16
		0.16	1.15	37.05	32.60	
			0.16	1.15	37.05	
				0.16	1.15	
					0.16	

new variable represents an increase in the degrees of freedom in the problem and an unnecessary increase in a model complexity, and possibly, may lead to overfitting.

In this problem of rainfall-runoff, as stated before, the size of the basin plays an important role in what is to be expected from the complexity of the model. Although average precipitation is considered, it is often not enough to include only one or two lagged variables in the model. Precipitation events taking place close to the discharge measurement point would lead to an increase in discharge with a lag, which could be much smaller than the average one, and events far from this point would have a larger lag. This leads to another type of problem, which is spatiotemporal in nature.

The correlation coefficient is commonly used to determine mathematical linear relations between two samples of random variables or time series. In the case of building a rainfall runoff model, the variables are lagged precipitation and discharge. The lag time, leading to a high correlation, would be the one to adopt as the most probable candidate for the final model.

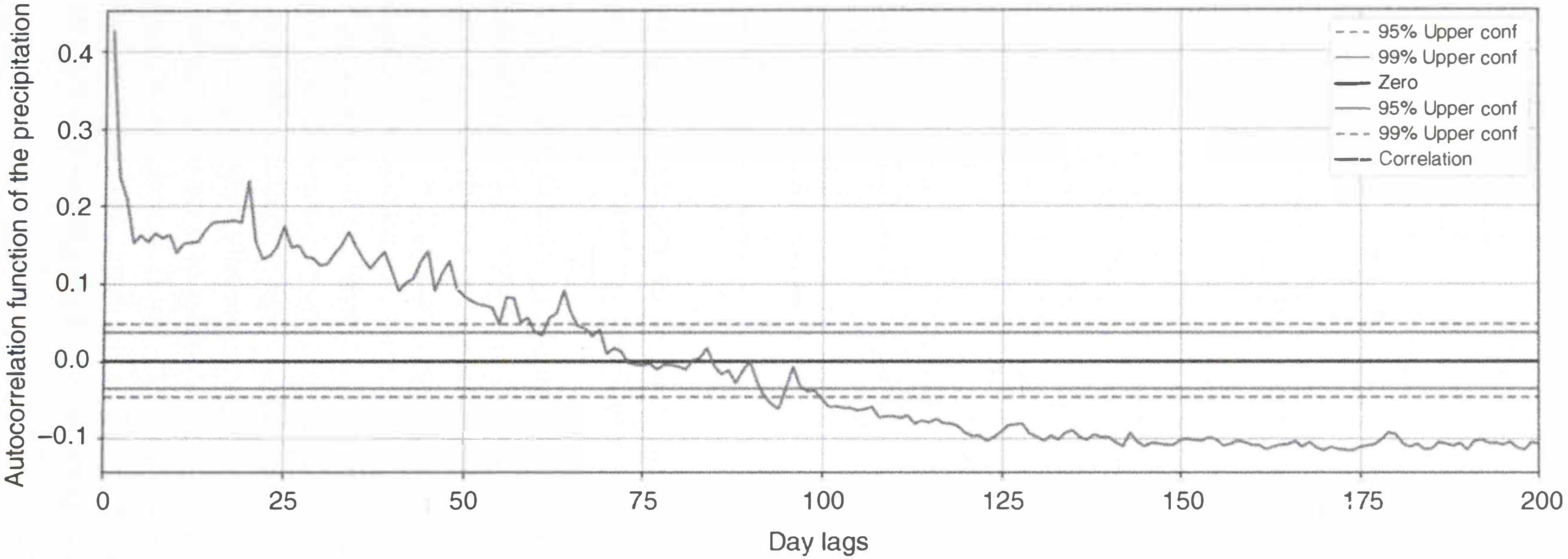

Figure 1.7 Autocorrelation for the discharge at Ourthe River basin (tributary of the Meuse).

Sin e the correlation coefficient can be misleading if not enough data are u ed or there could be events with a different correlation structure it could be u eful to employ the two other analy is techniques: 1) to analyze the variation of the correlation coefficient with different sizes of data sets and (2) to perform the correlation analy is eparately for certain events, for example with the discharge in a particular range.

It should be noted that a number of ML models (e.g. recurrent neural networks; see below) are ba ed on an architecture where the whole time series (or its part) representing an input variable i fed into the model and the choice of the previou (lagged) values and their weighting is done via an optimization process during the model training. In uch cases IVS is in choosing the right variable (rather than specific lagged values), without paying much attention to how they should be lagged in a model ince this is done automatically.

1.3.3. Optimization

The third step shown in Figure 1.5 is the modeling proce per se which in prin iple is called learning and i an optimization process to choose the model parameter and the structure to fit the model output to the measured (target) values. Numerous single-objective and multiobjective optimization te hnique are available for such purpo es and they are typically very pecific to each of the techniques. It is not pos ible to state that one technique will always provide a better solution than another. This can be partly explained by the existence of random components in many algorithm

1.3.4. Model Evaluation (Testing)

The final step in the training proce i the e aluation of the model s performance. After the model is optimized and a u h probably ele ted after some validation iteration are done the error of the model on te t data is used to as ess the resulting model performanc

It is worth mentioning that in proce -ba ed modeling in situation of poor data availability, data partitioning into training, validation and test subsets is not always carried out o proper model testing may be difficult to arrange. However in machine learning exerci es the procedure of data partitioning are normally followed.

The error measure (metrics) used in the training (learning) proce of the ML model (typically based on the mean squared error) should not be seen as the only way of model performance estimation. Commonly vi ual comparison, additional tests on extreme phenomena (crash tests), and a benchmark or reference model are used to assess the quality of the model as well. It is difficult to choose only one performance measure for a good model, so model training can be posed also as a problem of multiobjective optimization; on multiobjective machine learning, see, for example, Corzo and Solomatine (2006)· and on multitask learning, a good introductory overview is given by Ruder (2017). In such cases, the modeling results should preferably be a matter of further discussions with stakeholders about the selection of a ingle model out of a Pareto-optimal set of models generated as a result of multiobjective optimization (training).

1.3.5. Uncertainties in the Process of Building a Model

Modeling leads to a generation of new knowledge and new insights into the modeled processes, and typically involves cycles of progressive refinement and improvement, and even reformulations of the initial modeling goals. Some of possible questions raised during modeling experiments and the related cycles, are shown in Figure 1.8. Each of these questions is associated with an uncertain variable in the process. Uncertainty in data inevitably propagates to models' outputs. Changes in data partitioning also lead to different model parameterizations. Changes in the input variables set can lead to different model outputs and hence varying performance. Many ML models require random initializations of parameters, so that multiple runs may lead to different results. Finally the idea of having one objective function in the optimization process, where the real problem can normally have multiple objectives, makes the model biased to a certain modeling objective. For example, a ML model built for high flow might not be optimal for low flows and vice versa. Since there is no perfect measure for performance, there is uncertainty associated with the choice of what is defined as optimal in the modeling process.

In ML theory and practice all these problems have been studied, are quite well understood and adequate techniques have been developed. For example, *n*-fold cross-validation is used to deal with uncertainties in data partitioning. Uncertainty is reduced if instead of using a single model multiple (ensemble) models are built, based on different parameter (weights) randomizations, and then aggregated using (dynamic) weighting. The problem of multiregimes of the modeled system and seasonality is solved by modular or committee systems (Corzo & Solomatine, 2007b, 2007a· Kayastha et al., 2013). For problems related to the IVS, feature engineering offers a number of techniques (see Guyon & Elisseeff 2003; Bowden et al., 2005a, 2005b; Galleli & Castelletti, 2013). Robustness

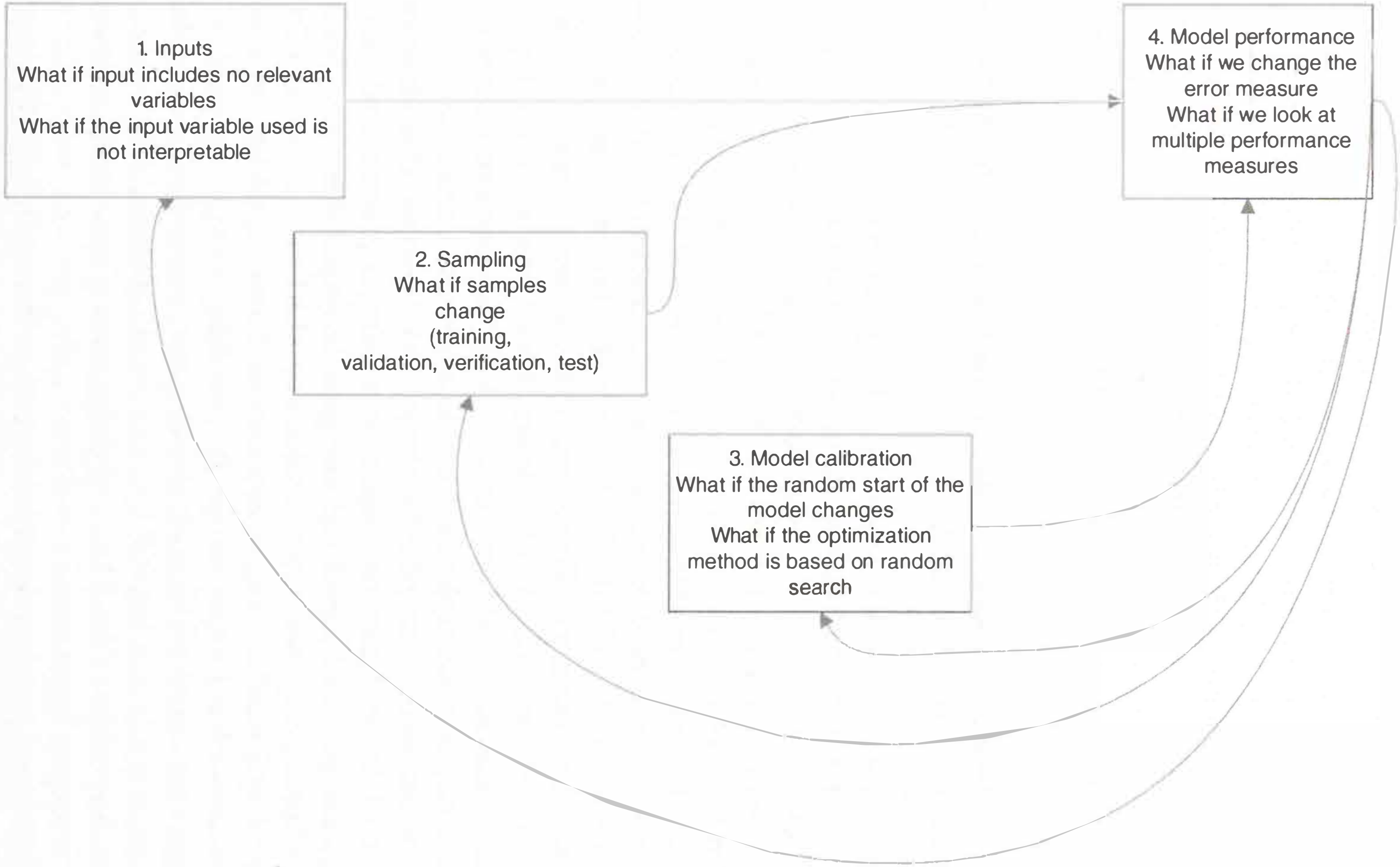
1. Inputs
What if input includes no relevant variables
What if the input variable used is not interpretable
2. Sampling
What if samples change
(training,
validation, verification, test)
3. Model calibration
What if the random start of the model changes
What if the optimization method is based on random search
4. Model performance
What if we change the error measure
What if we look at multiple performance measures

Figure 1.8 Possible cycles in ML model building.

of models can be increased also by encoding inputs and generating new features aggregating variables and thus reducing the input space dimension.

Uncertainty of the final model can be effectively studied u ing Monte Carlo technique . More, uncertainty of model outputs when working with new data in the future can be estimated by using methods like MLUE (Shrestha et al., 2009), which encapsulates the results of Monte arlo analysis of parametric uncertainty in a ML model, and UNEEC (Solomatine & Shrestha, 2009) and its extension (Wani et al., 2017), where residual uncertainty is encapsulated in a ML model, and its probability den ity function can be forecasted for the new inputs and model runs.

1.4. Advanced Techniques in Machine Learning for Water Resources Applications

A wide range of machine learning (ML) models has been extensively used in water resource applications. Among the reported re earch, a ignificant focus has been placed on five to seven popular technique in luding linear regression multilayer perceptron (MLP) ANN, radial-basis function (RBF) ANN, support vector machines, M5 model and regre sion tree random forests, and, more recently, LSTM (deep learning) network .

Machine learning plays a crucial role in various water re ource task such as hydrological modeling flood forecasting, drought prediction and water resource management. Depending on the nature of the output, the e application areas can be broadly classified into either clas ification or regression (numerical prediction) problems. In this chapter, we not only explore the popular ML techniques employed in water resources but also exp!icitly focus on the fascinating realm of deep learning. Deep learning characterized by architectures ba ed on recurrent multilayered neural networks, has garnered significant attention in recent years. Specifically, we delve into Long Short-Term Memory (LSTM) networks, Gated Recurrent Units (GRUs) and convolutional networks, which are widely used in water-related applications. By providing detailed explanations and insights into these techniques, this chapter aims to equip researcher and practitioners in the field of hydroinformatics with a comprehen ive understanding of the various ML and deep learning approache employed in water resources applications.

Classification problems involve predicting a discrete (nominal) output such as whether a flood event will occur or not. In classification, ML models are trained to output a class. For this purpose, regre ion model (like ANN) can be used as well: the predicted numerical value is encoded as a clas index. For example, in the case of flood foreca ting a ML numerical prediction model can be trained to predict the discharge values, but the final result will be pre ented as a binary class: "no flood ' (if discharge is below a predefined threshold) and ' flood' (otherwise).

There are al o other ways of encoding classes: data are transformed and the problem is formulated as having *N* binary outputs where *N* is the number of classe . For example in ca e of two possible outcome (like no flood and flood) output vectors (real-valued) would be either (0, 1) for flood conditions or (1 0) for no-flood conditions. A regression ML model is trained to reproduce real-valued vector . If. for the new inputs thi model gives an output vector close to (0 1) for example (0.11 0.85), then it i interpreted as flood, and if it is close to (1 0) for example (0.15 0.91) it is interpreted as no-flood.

However. in water resources a mu h more widely spread type of problem is a regression (numerical prediction) problem, which aims to predict a continuous output such a the di harge of a river or the water level in a re ervoir. For example, in the case of water resource management a ML model is employed to predict the re ervoir inflows water demand and water availability, and then to use the predictions to optimize the water allocati n.

This section investigates the use of machine learning techniques in water resources, emphasizing their relevance and efficacy. It begins by looking into artificial neural networks (ANNs) with a focus on the popular multilayer perceptron (MLP) ANN. MLP ANN are popular in water re ources re earch due to their apacity to handle complicated interactions and big data sets. The section gives a review of MLP ANN architecture and training procedures, emphasizing its applicability to water resource modeling. Following that, the subject hift to regres ion and model tree as well as bagging and boosting trategies for developing en emble model . These methods provide modular solution as well as tati tical tool for increasing model tability, accuracy and generalization. In regre ion and model trees, the combination of decision trees with linear model allows for flexible modeling of nonlinear connections while bagging and boosting approaches improve performance by combining numerous model . The se tion also discu ses the recent spike in intere t in deep learning, particularly in relation to water resources. Deep learning a represented by recurrent neural networks, exhibits exceptional ability in managing time series data and capturing complicated patterns. Long short-term memory (LSTM) networks, in particular. have hown considerable potential in hydrological modeling by overcoming long-term dependency concerns. The section emphasizes the significance of u ing deep learning methods into water management and forecasting. Overall this in-depth examination of machine learning approaches in water re ources highlights their potential and u efulne s in furthering re earch and applications in the field.

1.4.1. Artificial Neural Networks (ANNs)

The most widely used ANN is a multilayer perceptron (MLP). An MLP ANN consists of multiple layers of interconnected node (neuron · Fig. 1.9) that process input vector to produce a vector output (often a single value, however). The node in each layer first linearly combine inputs and then apply a nonlinear function (sigmoid or hyperbolic tangent) to transform the input further and pass the result to the next layer. whereas the final layer produces the network' output. MLP are typically trained by backpropagation, a method that progres ively adjusts the weight between nodes to minimize the difference between the network's output and the desired output; it can be seen as a specialized version of nonlinear gradient-based optimization. MLP may have several intermediate (hidden) layers, but the most widely used architecture u es only one. On details see, for example Haykin (1999) and many other books and Internet resources.

Overall MLP ANNs have proven to handle complex nonlinear relationships and large amounts of data, and can make predictions with high accuracy. There are hundreds of succe sful applications of ANN in water resources (see, e.g., Abrahart et al., 2012) and a MLP ANN is often seen as the first choice of a ML model.

1.4.2. Regression and Model Trees

Following a modular approach to modeling, a data-driven model should compri e several submodels. The training set is split into ubsets corresponding to a particular subprocess to be modeled, and then each module (Fig. 1.1) is trained on these nonintersecting subsets (actually, these subsets can be inter ecting leading to ome versions of en emble models, but this option is not considered here). When a new input vector i presented, it is first classified into one of the regions (corresponding to the subsets) for which the modules were trained, and then only one module is run to generate the prediction. A class of such methods employing consecutive progressive splits is typically referred to as *trees*. Examples are decision trees, regression trees (Breiman, 1984), which use zero-order models, that is, constants, in leaves; multiadaptive regre sion plines (MARS) (Friedman, 1991)· and M5 model trees (Quinlan 1992) which use linear

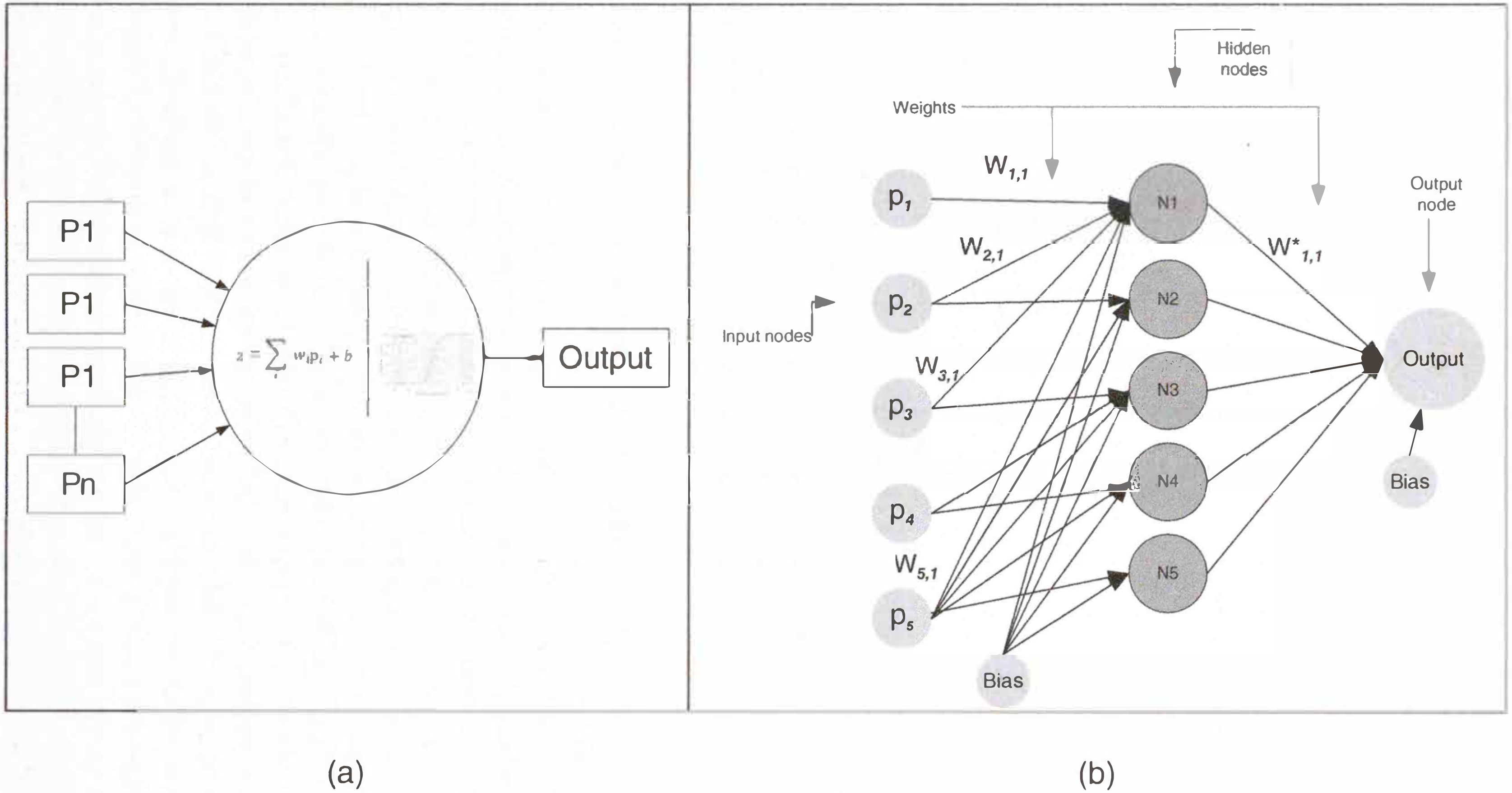

Figure 1.9 A typical neuron and a single-output multilayer perceptron (MLP ANN): (a) A neuron in an MLP ANN; (b) multilayer perceptron diagram.

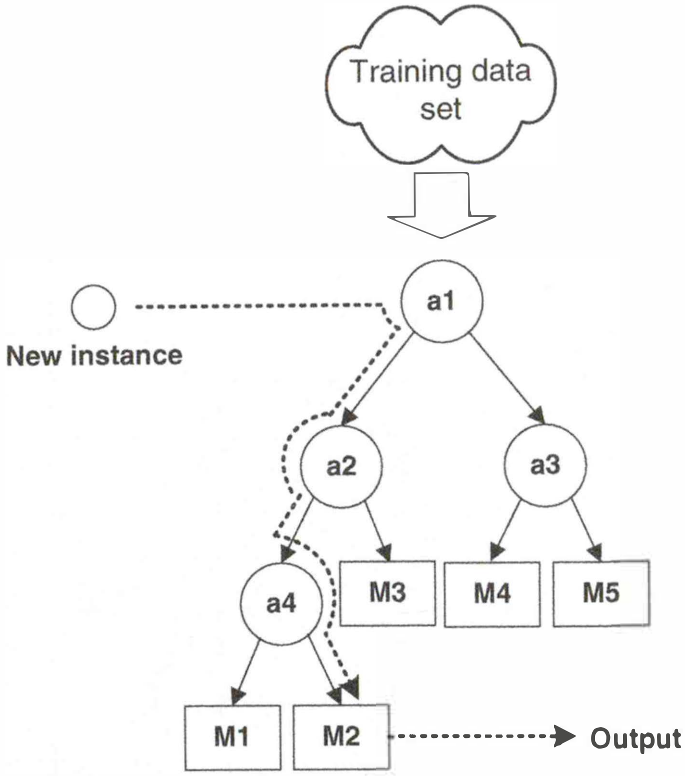

Figure 1.10 Training M5 model trees and their operation on a new unseen instance: a1–a3 are data partitioning rules; M1–M5 are multiple linear regression models.

regression models in leaves (Fig. 1.10, adopted from Bhattacharya & Solomatine, 2005).

Since for each data instance (input vector) only one local model is used for prediction, there is a problem with compatibility at the boundary between the regions for which the modules are responsible. For the two neighboring input vectors, the predicted outputs could be distinctive. A solution could be to update the local models to make them compatible at the boundaries, as is done in the M5 model tree algorithm through smoothing. Wang and Witten (1996) presented the M5 algorithm based on the original M5 algorithm but were able to deal with enumerated attributes, treat missing values, and use a different splitting termination condition. Several advantages of using the model tree are that it is a non-black-box model, understandable, easy to use and to learn, fast in training, robust when dealing with missing data, able to handle a large number of features, and able to tackle tasks with very high dimensionality. It has been shown that M5 model trees have accuracy similar or exceeding that of MLP ANNs (on water-related applications see, e.g., Bhattacharya & Solomatine, 2005; Solomatine & Xue, 2004).

In ummary, M5 model trees are a type of machine learning algorithm that combines the trengths of regression trees and linear models allowing for more flexibility in modeling nonlinear relation hips.

1.4.3. Bagging and Boosting Techniques for Building Committee Models

Bagging

Two or more models who e output are combined form a committee model. Statistical technique widely u ed in preparing data for building such models are bagging and boosting. The e methods are used to generate sub amples of data, which are used to train several models and then combine them in a committee.

The term *bagging* comes from bootstrap aggregating (bagging) which is an algorithm used to improve classification and regres ion models in terms of tability and classification accuracy. It also reduces variance and helps avoid overfitting. Bagging can be een a a special case of the model averaging approach. Bagging was proposed by Leo Breiman (1984).

Given a tandard training set D of size n bagging generate m new training sets Di of size $n' \leq n$, by sampling examples from D uniformly and with replacement. By sampling with replacement it is likely that some example will be repeated in each Di. If $n' = n$ then for large n the set Di is expected to have 63.2% of the unique examples of D with the re t being duplicates. This kind of sample is known a a boot trap sample. The m models are built using the above m bootstrap samples and combined by averaging the output (for regre sion) or voting (for cla ification). Sin e the method averages several predi tors, it i not useful for improving linear model .

Random Forest

A very popular algorithm based on bagging that u e tree-like structuring of data is the random forest algorithm (Breiman 2001). It is an ensemble method that combines multiple decision tree (in the original version regression trees) to make predictions. The basic idea behind this method is to build a large number of decision (regre sion) trees using random subsets of the data and features and then combine the prediction of these trees to make a final prediction.

The random forest algorithm works by fir t selecting a random subset of the data called a bootstrap ample to use as the training et for each decision tree. Then for each tree, a random subset of the features is selected to use as the split variables at each node. The algorithm then grows a decision tree using this random subset of data and features. This process is repeated many times to build a large number of decision trees, each with its own random subset of data and features. Finally, the predictions of all the trees (constituting an ensemble) are combined to make a final prediction.

One of the advantage of random forest is that it can reduce overfitting by averaging out the predictions of many decision trees. The algorithm also has the ability to handle large numbers of input variables and nonlinear relationships between the input variables and the target variable, which is a limitation of linear regression models.

Boosting

Boosting is an approach leading also to a committee (ensemble) of ML models, but it is a sequential model, where each subsequent model is dependent on the outcome of the previous. Boosting assigns weak learners to a weighted subset of the original data set. Weak learners have little predictive ability and perform just marginally better than random guessing. Subsets that were previously misclassified are given more weight and hence the probability to be selected for the subsequent learner. As a result, the ensemble has good generalizing ability. The two widely used versions of boosting are gradient boosting (Friedman, 2001) and adaptive boosting, AdaBoost (on its realization for regression problems, AdaBoost.RT, see Shrestha & Solomatine, 2006). A popular implementation of the former is in XGBoost (extreme gradient boosting) a C++ library with APIs for several languages (XGBoost, 2023), which is extensively employed in various types of applications.

1.4.4. Deep Learning

In relation to water resources, lately, a lot of attention has been given to the so-called deep learning, a general term given to architectures based on recurrent multilayered neural networks. Especially popular are long short-term memory (LSTM), gated recurrent units (GRUs· a version of LSTM), and convolutional networks. Deep learning (DL) is commonly referred to models with large number of layers and complex nodes with memory, for example, able to handle time series ‘ deep in time” or process images. DL has become a fashionable term, and currently is widely used, sometimes even (incorrectly) replacing a more general term ML.

1.4.5. Recurrent Neural Networks (RNNs) and LSTM

RNN

RNNs are designed to automatically learn and understand the mutual relationship between inputs which are provided in sequential order. RNNs are often applied to NLP and peech recognition, since they have sequential structure fit to process long time series (Fig. 1.11). In a typical RNN the output become a variable of state (*h*) used a input into the ame network for the following time step in the sequence.

Long Short-Term Memory (LSTM) Networks

Traditional RNNs face difficulties in training model with long equences, e pecially vanishing or exploding gradient problem. Thi problem was addressed in many studies, and one of the most popular techniques nowadays i the long- hort-term-memory (LSTM) network which was introduced by Hochreiter and Schmidhuber (1997). This and later development allowed for much wider adoption of deep learning in time series forecasting.

Learning of su h long-term dependencies between inputs and outputs is vital in hydrological modeling because the lag time between hydrological processes may vary from days to years. For instance the long lag time can be seen in the cases of glaciers now, and groundwater processes (Kratzert et al., 2019). Due to developments of new DL algorithm in the la t everal years, there has been clear revival of interest in using ML in water-related problems. The e have demonstrated high accuracy for example in flood forecasting (e.g., Kratzert et al. 2019· Kao et al. 2020; Arsenault et al., 2023) and drought forecasting (Brust et al. 2021· Dikshit & Pradhan, 2021).

Kratzert et al. (2019) investigated the use of LSTM in rainfall runoff hydrology using Catchment Attributes and Meteorology for Large-Sample Studie (CAMELS) data sets for numerou catchment . The authors also studied the extent to which a single LSTM model could be regionalized, that is, applied to other catchment . As a re ult, the LSTM predicted di charge with good accuracy compared with the reference model. The final conclusion claims that by using LSTM the problem that is generally related to ML models of being black box models could be overcome, and explanation is in the following. In LSTM the cell tate s behavior in respon e to the hydrological trend and patterns can be physically interpreted, for in tance, by analyzing the melting state of the cell tate in comparison with the change in temperature. Con equently, with low temperatures, there is a

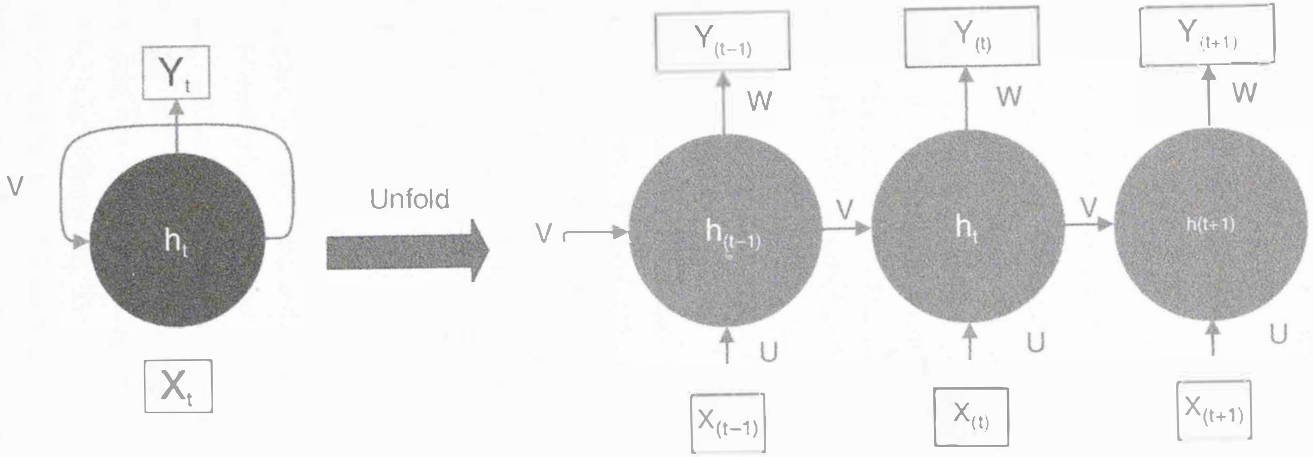

Figure 1.11 Recurrent neural network. Inputs from a time series (X) are fed into processing nodes, producing output (Y) at each time step. The state or memory variable h_t is equal to $Y_{(t)}$ in the basic algorithm.

certain increase in the cell state and when the temperature exceeds certain degree , the cell state starts to decrease (this imitates the snow accumulation and melting processes).

Examples of recent applications of DL in hydrometeorological problems which are characterized by very large data et can be mentioned also. Schultz et al. (2021) evaluated numerical weather prediction models, whi h are quite important for analyzing all types of water re ources problems. In this work, it is highlighted that DL is not yet widely used within the numerical weather prediction (NWP) context mainly due to the lack of both the interpretability of the neural network and the physical constraints. They mention also that many of the early experiments u ing imple neural network did not capture the complexity of weather proces es. Although these limitations are important, there are three main rea on for considering DL for weather prediction problems. Fir t large amounts of data are currently available. Second new architectures of the neural networks have been developed that can capture well the time dependencie (RNNs) as well as patial distributions and imagery such as the convolutional neural network (CNN). Third the computational power capacity needed for training such networks (graphical and specialized proces or available on the loud) has become e ily available. All this bring the latest advanced ML techniques at the disposal of modeler and forecasters in water resources, leading thus to optimal water management.

For further information on the use of AI and ML in Earth y tem science the reader is directed for example to the recorded lectures at the Summer School "Artificial Intelligence for Earth System Science (AI4ESS 2020) where some of the method and impres ive applications are pre ented.

1.5. Future Directions and Challenges

Use of modeling and forecasting tools is indispensable for effective water management. Traditionally such models belonged to the clas of the process-ba ed (physically based) model , typically based on numerical solution of differential equations of water motion. During the la t three decades advances in machine learning techniques, increased computing power and data availability have led to changes in this land cape. Hydroinformatics i seen as the major area where machine learning uch algorithms have been tested and applied to a large variety of water-related problems, and have proven to be tools effectively complementing, and sometimes replacing physically based models. An improved physical representation does not necessarily guarantee an increased accuracy and utility of traditional models, so alternative ways are sought. This chapter presented an overview of the main approach in ML, relevant for this volume, in most cases with references to the authors' own experience, and to the sources for further reading.

There is currently a serious change in the attitude toward the data-driven models within the practicing modelers community and the current discussions in the hydrological literature is an indicator of this (see, e.g., Nearing et al., 2021; Beven, 2020). ML not only provides techniques for modeling, but also enhances ability of citizens to get access and better understand the modeling results via the natural language processing (NLP) and intelligent chatbots.

We see the future advances of modeling technologies in more effective combination of various types of models and their hybridization (e.g., in the line of physics-aware AI see Jiang et al. 2020) and coevolution (as outlined by Razavi et al., 2022). Out of the experience of the authors, we see the following as the main future directions:

1. Integration of AI and ML into operational water management: Integrating AI and ML models into real-time decision support systems for efficient and adaptive water management is one potential approach. This entails creating models that can adapt to changing environmental circumstances in real time and improve water allocation and distribution.

2. Improved data availability and quality: ML models require access to high-quality diversified data. Future work should concentrate on improving data-gathering methods, strengthening data-sharing channels, and creating approaches for dealing with missing or incomplete data. This will aid in the development of more accurate and robust ML models.

3. Uncertainty quantification and risk assessment: Addressing uncertainty and mea uring risks associated with machine learning predictions are major problems. Future research should concentrate on creating approaches for quantifying uncertainties in ML-based predictions, as well as incorporating risk assessment frameworks for improved decision making under uncertainty.

4. Explainability and interpretability of ML models: ML models particularly deep learning models, are often considered black boxes due to their complex structures. Enhancing the interpretability and explainability of ML models in water resource applications is an important

direction. This will enable stakeholders to understand the reasoning behind model predictions and build trust in their use.

5. Integration of domain knowledge and expert systems (hybrid models): Integration of domain knowledge and expert systems can boost ML models. To increase model accuracy and interpretability future initiatives include constructing hybrid models that integrate ML methods with physics-based models or expert knowledge.
6. Ethical and responsible AI: As AI and ML play an increasing role in water resources management, addressing ethical considerations, data privacy and bias becomes crucial. Future efforts should focus on developing guidelines standards, and frameworks for the ethical and responsible use of AI in water-related applications.
7. Bridging the research-practice divide: It is critical to translate research results into practical applications and to develop cooperation among scholars, practitioners, and policymakers. Future directions should emphasize knowledge transfer, capacity building, and effective communication to ensure that ML techniques are effectively applied in real-world water management scenarios.

Finally, the chapter presented a thorough assessment of the future directions and difficulties in the field of hydroinformatics and machine learning for water-related problems. As we navigate the ever-changing world of water resource management, it is evident that embedding AI and machine learning technologies into operational ystem provides a viable path toward better water allocation and distribution. However, in order to fully exploit these technologies' promise, data availability and quality must be enhanced uncertainties measured, and dangers analyzed. Furthermore, the explainability and interpretability of ML models domain knowledge integration, ethical issues, and effective knowledge transfer are critical topics that require study. We can support sustainable water management practices and assure the appropriate and successful use of AI and ML in tackling the ever-growing complexities of water-related concern by embracing these difficulties and investigating the recommended future approaches.

References

Abbott, M. B. (1991). *Hydroinformatics: Information technology and the aquatic environment*. Avebury Technical.

Abrahart, R. J. Anctil F., Coulibaly, P., Dawson C. W. Mount, N. J. See L. M. et al. (2012). Two decades of anarchy? Emerging themes and out tanding challenge for neural network river forecasting. *Progress in Physical Geography, 36*(4) 480–513. http ://doi.org/ 10.1177/0309133312444943

Abrahart, R. J., See, L. M., & Solomatine, D. P. (Eds.) (2008). *Practical hydroinformatics: Computational intelligence and technological developments in water applications* (Vol. 68). Springer Science & Business Media.

AI4ESS (2020). *Artificial Intelligence for Earth System Science Summer School* 2020. https://www2.cisl.ucar.edu/events/summer-school/ai4ess/2020

Arsenault, R., Martel, J.-L., Brunet, F., Brissette, F., & Mai, J. (2023). Continuous streamflow prediction in ungauged basins: Long short-term memory neural networks clearly outperform traditional hydrological models. *Hydrology and Earth System Sciences, 27*(1), 139–157.

Artificial intelligence (2022, December 2). In Wikipedia. https://en.wikipedia.org/wiki/Artificial_intelligence

Beven, K. (2020). Deep learning, hydrological processes and the uniqueness of place. *Hydrological Processes, 34,* 3608–3613.

Bhattacharya, B., & Solomatine, D. P. (2005). Neural networks and M5 model tree in modeling water level-discharge relationship. *Neurocomputing, 63,* 381–396.

Bowden, G. J., Dandy, G. C., & Maier, H. R. (2005a). Input determination for neural network models in water resources applications. Part 1, Background and methodology. *Journal of Hydrology, 301,* 75–92.

Bowden, G. J., Dandy, G. C., & Maier, H. R. (2005b). Input determination for neural network models in water resources applications. Part 2, Case study: Forecasting salinity in a river. *Journal of Hydrology, 301,* 93–107.

Breiman, L. (1984). *Classification and regression trees.* Routledge. https://doi.org/10.1201/9781315139470

Breiman, L. (2001). Random forests. *Machine Learning, 45,* 5–32.

Brust, C., Kimball, J. S., Maneta, M. P., Jencso, K., & Reichle, R. H. (2021). Droughtcast: A machine learning forecast of the United States drought monitor. *Frontiers in Big Data, 4.* https://doi.org/10.3389/fdata.2021.773478

Corzo, G., & Solomatine, D. (2007a). Baseflow separation techniques for modular artificial neural network modelling in flow forecasting. *Hydrological Science Journal, 52*(3), 491–507.

Corzo, G., & Solomatine, D. (2007b). Knowledge-based modularization and global optimization of artificial neural network models in hydrological forecasting. *Neural Networks, 20*(4), 528–536.

Corzo, G. A., & Solomatine, D. P. (2006). *Optimization of base flow separation algorithm for modular data-driven hydrologic models.* Proceedings of the 7th International Conference on Hydroinformatics (HIC 2006). Nice, France.

Corzo, G. A., Diaz Mercado, V., & Laverde Barajas, M. (2018). Spatiotemporal Hydrological Analysis. *International Journal of Hydrology, 2*(1), 25–26.

Corzo Perez, G. A. (2009). *Hybrid models for hydrological forecasting: Integration of data-driven and conceptual modeling techniques.* Taylor & Francis, CRC Press.

Diaz, V., Corzo Perez, G. A., van Lanen, H. A. J., Solomatine, D., & Varouchakis, E. A. (2020). An approach to characterize spatio-temporal drought dynamics. *Advances in Water Resources.*

Dikshit, A., & Pradhan, B. (2021). Explainable AI in drought forecasting. *Machine Learning With Applications, 6,* 100192.

El horbagy, A. Corzo, G. Sriniva ulu, S. & Solomatine D. P. (2010). E p rimental investigation of the predictive capabiliti of data-driven modeling te hnique in hydrology. Part 1 Concept and methodology. *Hydrolog and Earth System Sciences*.

Friedman, J. H. (1991). Multivariat adaptiv regre ion pline . *Th Annal. of Stati. ti s*, *19*(1), 1–67. http://www.j tor.org/ tabl /2241837

Friedman, J. H. (2001). Greedy function approximation: A gradient boo ting machine. *The Annals of Statistics, 29*(5), 1189–1232.

Galelli S., & Ca telletti A. (2013). Tree-based iterative input variable lection for hydrological modeling. *Water Resources Re earch. 49* (7) 4295–4310.

Guyon, I., & Eli eeff, A. (2003), An introduction to variable and feature election. *Journal of Machine Learning Research, 3* 1157–1182.

Haykin, S. (1999). *Neural net orks and learning machines*. Pear on / Prentice Hall.

Ho hreiter S., & Schmidhuber, J. (1997). Long hort-term memory. *Neural Computation, 9*(8) 1735–1780.

Jiang, S., Zh ng Y., & Solomatine D. (2020). Improving AI y tem awarene of geoscience knowledge: Symbiotic integration of phy ical approache and deep learning. *Geophy ical Research Letter* , *47*(13) e2020GL088229.

Kao, I-F., Zhou, Y., Chang, L-C. & Chang F-J. (2020). Exploring a long short-term memory ba ed encoder-decoder framework for multi- tep-ahead flood forecasting. *Journal of Hydrolog* , *5 3* 1246 1.

Kaya tha, N. Ye J. Fenicia, F. Kuzmin V., & Solomatin D. P. (2013). Fuzzy committees of pecialized rainfall-runoff model : Further enhancement and test . *Hydrology and Earth System Sci n* , *17*, 4441–4451.

Kho hnazar A. Corzo Perez G. A., & Diaz V. (2021). Spatiotemporal drought ri k a se ment con idering re ilience and heterog neou vulnerability factors: Lempa transboundary river ba in in the central American dry corridor. *Journal of Marine cience and Engineering* 9(4).

Khurana D. Koli A., Khatter K. et al. (2023). Natural language proc ing: State of the art current trend and challenge . *Multimedia Tools and Appli ation* , *82*, 3713–3744.

Kratzert, F. Klotz D., Shalev, G., Klambauer, G. Hochreiter. S. & earing G. (2019). Benchmarking a catchment-aware long hort-term memory network (LSTM) for large- cale hydrological modeling. *Hydrology and Earth Sy tem Sci n es Discu s, 2019* 1–32.

Mitchell T. M. (2007). *Machine learning* (Vol. 1). New York: McGraw-Hill.

Moreido V., Gart man, B., Solomatine D. P., & Suchilina Z. (2021). How well can machine learning models perform without hydrologi t ? Application of rational feature election to improve hydrological for a ting. *Water, 13*(12) 1696.

Nearing G. S., Kratzer F., Samp on, A. K. P li ier, C. S., Klotz D. Frame J. M., et al. (2021). What role doe hydrological cience play in the ag f ma hine learning?. *Water Resources Research,* 7 e2020WR02809.

Potgi ter, T. & Dahlberg, J. (2022). *Automat d machine l arning on AWS: Fast-track the development of our production-r ad machine learning appli ations the AWS way*. PACkt Publishing

Premack, D. & Woodruff, G. (1978). Doe the chimpanzee have a theory of mind? *Behavioral and Brain Science* *1*(4) 515–526.

Pyle D. (1999). *Data preparation for data mining.* MorganKaufmann, USA.

Quinlan J. R. (1992). Learning with continuou cla es. *Proceeding Au tralian Joint Conference on Artificial Int lligence* (pp. 343–348). Singapore: World Scientific.

Razavi, S., Hannah, D. M. Elshorbagy A., Kumar S., Mar hall, L., Solomatine D. P., et al. (2022). Co volution of machine learning and proce -ba ed modeling to revolutionize Earth and environmental cience : A per pective. *Hydrological Processes, 36* e14596. http ://doi.org/10.1002/hyp.14596

Ruder, S. (2017). *An overview of multi-task learning in deep neural net orks.* http :// www.ruder.io/multi-ta k

Schultz M. G. Betancourt, C. Gong, B. Kleinert, F., Langguth M. Leufen L. H. et al. (2021). Can deep learning beat numerical weather prediction? *Philo ophical Tran actions of the Royal Society A, 379* (2194).

Shrestha D. L. & Solomatine D. P. (2006). Experiment with AdaBoost.RT an improved boo ting cheme for regre ion. *Neural omputation, 17,* 1678–1710.

Shre tha D. L. Kaya tha, N., & Solomatine D. P. (2009). A no el approach to parameter uncertainty analy i of hydrological model u ing neural network . *Hydrology and Earth System Sciences, 13,* 1235–1248.

Solomatine D. P. & Shre tha, D. L. (2009). A no el method to e timate model uncertainty using machine learning techniques. *Water Re ource Research, 45,* W00B11.

Solomatine, D. P., & Xue, Y. (2004). M5 model tree compared to neural network : Application to flood foreca ting in the upper reach of the Huai River in China. *Journal of Hydrologic Engineering, 9*(6) 491–501.

Stanford Encyclopedia of Philosophy (2018). Artificial intelligence. http ://plato .stanford.edu/entries/artificial-intelligence/

Sun T. X. Liu, X. Y., Qiu X. P., & Huang X. J. (2022). Paradigm hift in natural language processing. *Machine Intelligence Research, 19*(3), 169–183.

Varouchaki , E. A., Hri topulo D. T. Karatzas, G. P. Perez, G. A. C., & Diaz V. (2021). Spatiotemporal geo tatistical analysi of precipitation combining ground and atellite observations. *Hydrology Research, 52*(3).

Wang, Y., & Witten I. H. (1996). Induction of model tree for predicting continuous classes. In Artificial intelligence, Wikipedia (2023, March 30). https://en .wikipedia.org/wiki/Artificial_intelligence

Wani, O., Beckers, J., Weert , A. H., & Solomatine, D. P. (2017). Re idual uncertainty estimation using in tance-ba ed learning with applications to hydrologic forecasting *Hydrology and Earth System S iences, 21*, 4021–4036.

Weizenbaum J. (1966). ELIZA: A computer program for the tudy of natural language communication between man and machine. *Communications of the ACM, 9*(1), 36–45.

XGBoost (2023). https://github.com/dmlc/xgboo t