

Article

Tuning ANN Hyperparameters for Forecasting Drinking Water Demand

Andrea Menapace ^{*}, Ariele Zanfei  and Maurizio Righetti 

Faculty of Science and Technology, Free University of Bozen-Bolzano, Universitätsplatz 5, 39100 Bolzano, Italy; ariele.zanfei@natec.unibz.it (A.Z.); maurizio.righetti@unibz.it (M.R.)

* Correspondence: andrea.menapace@unbz.it

Abstract: The evolution of smart water grids leads to new Big Data challenges boosting the development and application of Machine Learning techniques to support efficient and sustainable drinking water management. These powerful techniques rely on hyperparameters making the models' tuning a tricky and crucial task. We hence propose an insightful analysis of the tuning of Artificial Neural Networks for drinking water demand forecasting. This study focuses on layers and nodes' hyperparameters fitting of different Neural Network architectures through a grid search method by varying dataset, prediction horizon and set of inputs. In particular, the architectures involved are the Feed Forward Neural Network, the Long Short Term Memory, the Simple Recurrent Neural Network and the Gated Recurrent Unit, while the prediction interval ranges from 1 h to 1 week. To avoid the problem of the Neural Networks tuning stochasticity, we propose the selection of the median model among several repetitions for each hyperparameter's configurations. The proposed iterative tuning procedure highlights the change of the required number of layers and nodes depending on Neural Network architectures, prediction horizon and dataset. Significant trends and considerations are pointed out to support Neural Network application in drinking water prediction.



Citation: Menapace, A.; Zanfei, A.; Righetti, M. Tuning ANN Hyperparameters for Forecasting Drinking Water Demand. *Appl. Sci.* **2021**, *11*, 4290. <https://doi.org/10.3390/app11094290>

Academic Editor: Francesco Granata

Received: 7 April 2021
Accepted: 4 May 2021
Published: 10 May 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: artificial neural networks; drinking water consumption; machine learning; grid search hyperparameters; short-term forecasting; tuning analysis; water smart grids; water supply systems

1. Introduction

Nowadays the forecasting of water demand is a fundamental task for the proper management of Water Distribution Systems (WDS). The new era of Big Data has made an enormous amount of information available, leading to the development of more precise prediction tools in several fields. At the same time, the increase of water demands from humans and agriculture purposes with the negative effects of climate change on water availability requires sustainable and efficient management of water and energy resources [1–3]. The scientific community focused on developing new and advanced methodologies to forecast the water consumption in a WDS [4]. In recent years, a plethora of methodologies have been developed for short-term forecasting, e.g., [5,6]. Early works tackled this topic by using conventional and statistical approaches [7]. For instance, these forecasting techniques include linear regression and time series analysis [8] or the Seasonal AutoRegressive Integrated Moving Average (SARIMA) class of methods [9,10].

With the growth of computational power from the beginning of this century, many studies started to explore methods that belong to the machine learning field. It was noticed that models based on machine learning have the potential to improve the forecast procedure. It is worth noting that the forecasting of urban water demand is a demanding task due to the stochastic and complex nature of the demand itself [11]. In this frame, machine learning methods appeared a suitable and promising option. In fact, many studies [12,13] showed how machine learning methods outperform the more classical approaches. Thereafter, the scientific community started to work at the development of new methodologies even more powerful to deal with the complexity of the different water demand time series.

For example, Guo et al. [6] proposed a methodology based on a Gated Recurrent Unit (GRU) to predict the water consumption until a time resolution of 15 minutes of two districts in China. The authors made also a comparison with a classic Artificial Neural Network (ANN) model and with a SARIMA one, proving that the GRU model was able to outperform the other 2 in the proposed test cases. Furthermore, Ambrosio et al. [14] proposed a machine learning committee model to combine different techniques to achieve a more robust and performing model. The authors analysed different combinations of techniques as the multilayer perceptron, the support vector machines, the random forest and many others, to show that the resulting committee model was able to outperform single models. Another significant example is given by Xenochristou and Kapelan [15]. They proposed an ensemble model to predict the consumption of the Southwest of England demonstrating that the performance of the ensemble model overcomes those of single models for both accuracy and reliability. Moreover, they compared different bias correction techniques aimed at the improvement of the model performances.

Hence, machine learning can be considered the current state-of-the-art technique in the field of the short-term forecasting of water demand. However, a large part of these methods relies on hyperparameters to achieve high-level performances [16]. For these reasons, it is fundamental to tune properly the hyperparameters of ANN models. Several strategies were developed to find the optimal settings for their machine learning models, which include grid search, random search, Bayesian optimization, gradient-based optimization and evolutionary optimization. One of the widely used technique is the grid search, which consists of a deep evaluation through a specified space of hyperparameters. This method can be computationally slow in case of wide space to explore. However, this method ensures a complete study of the hyperparameters combinations of ANN models due to its excellent explainability. An alternative method is the random search [17], which selects some sets of parameters randomly making the tuning computationally less intensive. Then, many other optimization approaches have been proposed by the scientific community, ranging from Bayesian optimization [18,19] to evolutionary optimization [20,21]. For instance, Perea et al. [22] proposed to couple a dynamic artificial neural network with a genetic algorithm. This strategy allowed the authors to find the optimal hyperparameters for their model, whose aim was to forecast daily irrigation water demand.

The reason for this study lies in the current challenge of designing a reliable machine learning model able to manage the complex task of selecting the best Artificial Neural Networks structure in the field of drinking water. We intend to advance in the field of ANN tuning for water demand forecasting by providing significant outcomes obtained from the procedure presented afterwards based on an iterative grid search analysis. Indeed, this procedure investigates the tuning of nodes and layers, which constitute the structure of deep ANN, for the short term forecasting models of different prediction horizons. A suitable ANN tuning guarantees a reliable water demand prediction which is a crucial information for water distribution operators [12]. Short term water demand prediction provides hourly values from 1 h up to 1 week into the future able to support different WDS activities. Several practical implications follow: the optimal management relies on the accuracy of the tuned models and their outcomes, including tanks levels management, valves regulation, pumping operations scheduling, pressure management, anomaly detection, leakages identification and maintenance activities planning [23–27].

Specifically, the focus of this study relies on insightful analysis of the deep ANN tuning through a iterative grid search, which is the most widely used method. Although this method is not the most performing one in term of accuracy and computational intensity, it ensures the best explainability which is a crucial aspect for the proposed tuning analysis. Thus, the grid search allows for the investigation of the results of different models by varying the number of layers and the number of nodes for different combinations of dataset, set of inputs, forecasting horizon and also for four ANN architectures. The latter involved are the Feed Forward Neural Network (FFNN), the Long Short-Term Memory (LSTM), the Simple Recurrent Neural Network (SRNN) and the Gated Recurrent Unit (GRU), which

are state-of-the-art algorithms widely used in engineering problems. Moreover, 24 different combinations of layers and neurons, which range respectively between 1–3 and 16–96, have been investigated. Regarding the dataset, 2 types of water demand time series have been used, one synthetic and the other real. Besides, this study has been carried out on 4 forecasting horizons, which are 1 h, 6 h, 1 day and 1 week. Eventually, two sets of inputs for each forecasting horizons, meaning different numbers of lags, have been selected. For the sake of clarity, the proposed analysis involves the ANN tuning based on univariate time series of water consumption modelling, which means that the only included data are the time series itself and other related calendar effects.

The reliability of the ANN tuning results of each combination has been ensured by repeated tuning simulations followed by the selection of median of the models chosen on the basis of Mean Absolute Percentage Error (MAPE). An in-depth analysis is presented on the variability of the ANN tuning results of different architectures and prediction horizons for supporting the selection of the number of tuning iterations. This selection has been done looking at the correct compromise between computational intensity and accuracy of models selection. Finally, the results of the proposed tuning investigation on the multiple combinations of hyperparameters are presented highlighting the identified trends and comparing the 4 ANN architectures. The required number of layers and nodes of the models depends on the Neural Network technique, the prediction horizon, the set of inputs and also the dataset. Significant considerations are pointed out to support Neural Network application in drinking water prediction. Despite that, the outcomes carried out in this study are strictly related to hyperparameters tuning of water demand forecasting due to the data driven nature of ANN forecasting modelling and are not directly portable to other types of data. Indeed, the uniqueness of this study in the field of water engineering does not allow a comparative analysis with other works. Nevertheless, a few analyses on the tuning process of ANN have been proposed in different areas, e.g., spatial data in ecological modelling [28], hyperparameters in AlphaZero-like self-play algorithm [29] and energy smart grids [30], resulting in useless comparison due to the specificity of data driven models.

The remainder of the paper is structured as follows. Section 2 introduces the data concerning synthetic and real case study and the analysis methodology, specifying each step from procedure structure to evaluation metrics. Section 3 illustrates the implications of ANN stochasticity in tuning (Section 3.1) and then presents the main results of the several analysed combinations of hyperparameters (Section 3.2). Final remarks in Section 4 conclude the article.

2. Methods and Materials

2.1. Materials

Two case studies are proposed to carry out the tuning analysis. The first regards an artificially generated time series following the procedure proposed by Menapace et al. [31], hereafter called *ts1*, and the second one is a real WDS in Trentino (Italy), hereafter called *ts2*. The complete time series are reported in Figure 1.

An insight of the inner structure of the proposed consumption time series is given in Figure 2.

Both the time series are characterised by an important variability in their behaviour. In particular, the synthetic time series (*ts1*), which is depicted in Figures 1a and 2a, shows an important daily but also yearly seasonality. *ts1* has been generated to supply water to approximately 5000 users and random noises have been added to the water request to enhance its variability and producing a realistic final water consumption. For additional details about the procedure, see [31]. Concerning the real consumption time series (*ts2*) reported in Figures 1b and 2b, it shows a seasonal component with an enhanced daily variability. In addition, an anomaly occurs between 2017 and 2018 making the learning of the ANN models more complex.

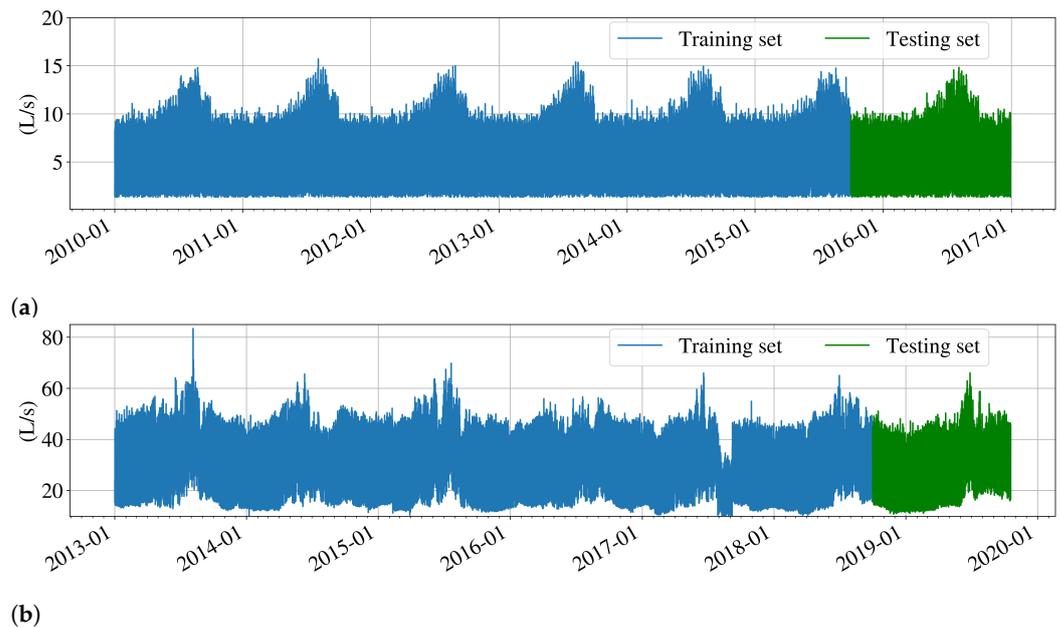


Figure 1. Full length of the time series of water consumption adopted in this study. (a) depicts the synthetic time series (*ts1*), whereas (b) represents a real consumption of a WDS in Trentino (*ts2*).

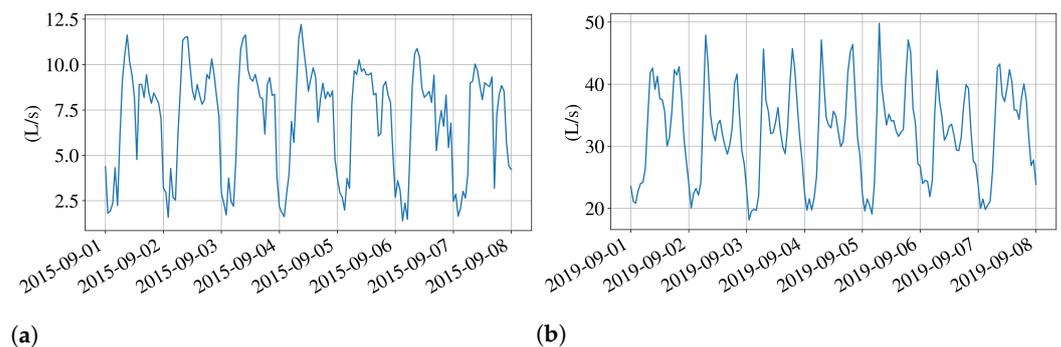


Figure 2. A weekly view of the two time series. (a) and (b) show the intra-daily behaviour of *ts1* and *ts2*, respectively.

2.2. Methods

2.2.1. Methodology Structures

This section presents a description of the methodology used for the proposed investigation. It is worth noting that the training of a neural network model is an optimisation procedure aiming to build up the more performing model to accomplish a defined task. Due to the nature of this optimisation, the model resulting from the training procedure is slightly different even adopting the same set of parameters. This is due to the fact that ANN algorithms use randomness during learning to help finding the best model and avoiding the local optima. For this reason, it is proposed to perform an iterative investigation to address this stochastic component. Figure 3 summarises the proposed investigation methodology.

Figure 3 shows the proposed iterative grid search procedure, whose aim is to assess the hyperparameters tuning in the context of water demand forecasting. The procedure, which is repeated N times, consists of the problem definition, the Neural Network selection, the combination of variables testing and finally the evaluation of each model. The grid search has carried out for each problem, which is defined for a selected dataset with forecasting horizon ranges among 1, 6, 24, 168 h. For the sake of clarity, the horizon means the number of next hourly demand values that the model has to predict. Hereafter, the ANN architecture to analyse can be chosen from FFNN, LSTM, GRU and SRNN. Several combinations of variables have been tested including nodes and layers, which constitute

ANN structure and also the set of inputs. These latter comprise both 2 different set of past observation but also a set of other indicators that are simple variables that give useful information about the time of the prediction. These are the day type, which indicates whether it is a working day or not, the actual month, whose aim is to give information about the seasonality and the actual hour which is a number indicating the hour target of the prediction and suggests to the model the daily behaviour of the water demands. A summary of the selected parameters is proposed in Table 1.

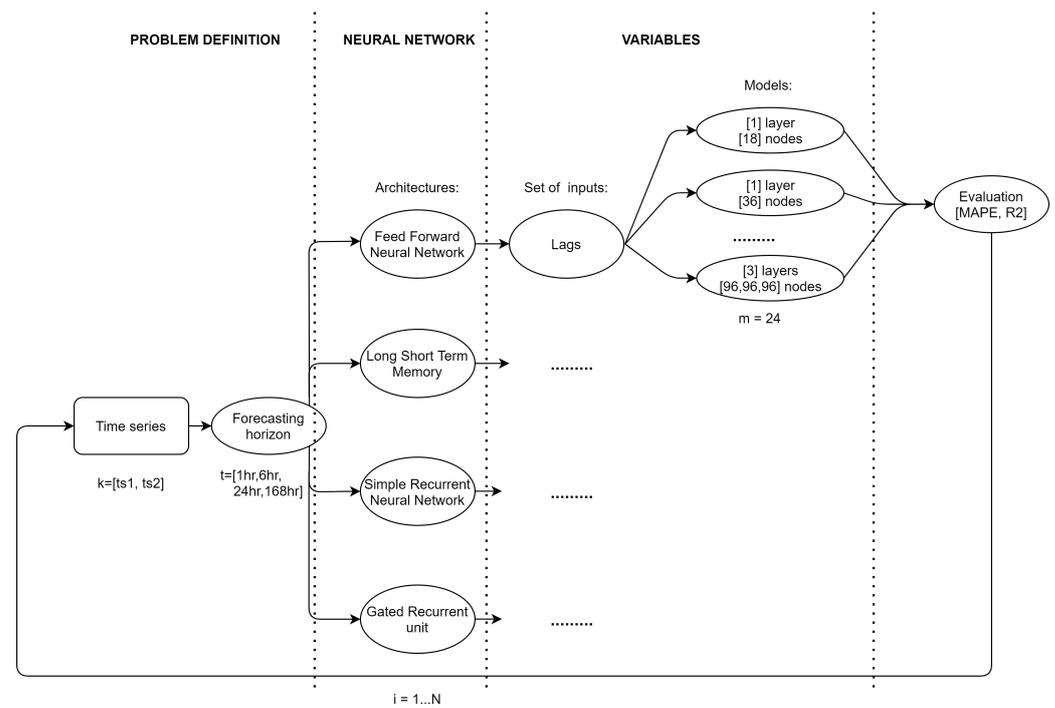


Figure 3. Flowchart of the proposed procedure for the analysis of ANN nodes and layers tuning based on an iterative grid search suitable for water demand prediction.

Then, the 24 combinations of layers and nodes selected for this investigation are iteratively tested to evaluate their performances. The nodes and layers combinations are reported in Table 2.

Table 1. Selected parameters for each forecasting horizon: batch size, past observations inputs and other inputs; q_t represents the first consumption hourly value to be predicted.

Horizon	Batch	Set of Inputs 1	Set of Inputs 2	Other Indicators
1 h	24	$q_{t-1}, q_{t-2}, q_{t-3}, q_{t-23}$	$q_{t-1}, q_{t-2}, q_{t-3}, q_{t-22}, q_{t-23}, q_{t-24}, q_{t-143}, q_{t-144}, q_{t-167}, q_{t-168}$	Actual hour, Actual month, Day type
6 h	4	$q_{t-1}, \dots, q_{t-6}, q_{t-18}, \dots, q_{t-23}, q_{t-43}, q_{t-44}, q_{t-45}$	$q_{t-1}, \dots, q_{t-12}, q_{t-18}, \dots, q_{t-24}, q_{t-43}, \dots, q_{t-48}, q_{t-163}, \dots, q_{t-168}$	Actual hour, Actual month, Day type
24 h	1	$q_{t-1}, \dots, q_{t-24}, q_{t-143}, \dots, q_{t-148}$	$q_{t-1}, \dots, q_{t-24}, q_{t-142}, \dots, q_{t-168}$	Actual hour, Actual month, Day type
168 h	1	$q_{t-1}, \dots, q_{t-24}, q_{t-142}, \dots, q_{t-168}$	$q_{t-1}, \dots, q_{t-168}$	Actual hour, Actual month, Day type

The iteration is performed N times for each of the 8 problems consisting of 4 prediction horizons for the 2 case studies. The number of iteration has been selected after a stochastic analysis of the process, which has been reported in Section 3.1.

Table 2. Combination of layers and neurons adopted for the proposed investigation for the four neural networks architectures.

Layers	Nodes
1	[18], [36], [48], [72], [96]
2	[18-18], [18-36], [36-36], [36-48], [48-48], [48-72], [72-72], [72-96], [96-96]
3	[18-18-18], [18-36-18], [36-36-36], [36-48-36], [48-48-48], [48-72-48], [48-96-48], [72-72-72], [72-96-72], [96-96-96]

2.2.2. ANN Architectures

This section aims to present the four neural network architectures adopted in this investigation. The proposed models have been developed with the KERAS [32] and TensorFlow [33] Python frameworks.

Feed Forward Neural Network

The first proposed architecture is probably the most classical and used one, known also as multilayer perceptron. These networks are based on a succession of units, called neurons, that are grouped in a sequence of layers. The first called the input layer, whose purpose is to connect the input information to the subsequent layers, which are the hidden ones. In this case, it is possible to have one or more hidden layers, producing in this way a deep neural network. Lastly, the output layer provides the outcome of the model. Due to its relative simplicity, this class of network has been widely adopted in many studies [4] and has become a benchmark for the forecasting process.

Simple Recurrent Neural Network

The simple recurrent neural network (SRNN) is a class of neural networks whose purpose is to remember information during the time. For this reason, their connections are different from the multilayer perceptron and they are designed to have not only feedforward connections but also feedback connections. Despite that this feature should allow the SRNN to remember past information with long term dependencies, some studies showed that this class of network was actually not capable to remember, and the reason is the vanishing gradient problem [34]. To overcome this issue, a different variation of recurrent neural networks has been proposed, like the LSTM [35] or the GRU [36].

Long Short-Term Memory

LSTM has been designed to preserve long term information, avoiding the vanishing gradient problem. These particular networks have each computing units organised in a series of gates aimed at the management of the memory. These include the forget, the input, the cell and the output gates. With this complex sequence of individual hidden units, the LSTM structure has to learn how to store information, how to remember information and what output to give. It is worth noting that this is a simplified explanation of the complex mechanism that is behind the LSTMs. There exists a plethora of studies developed in recent years that deal with this architecture, for instance [37].

Gated Recurrent Unit

The LSTM networks were proposed in 1997 to solve the vanishing gradient problem and perform in language processing problems [35]. Despite their effectiveness and their ability to deal with long-term information, these networks have a complex structure that makes their training much slower compared to conventional neural networks. To make the process faster, the GRU networks were proposed as a modification of LSTM networks. Similarly to the LSTM, the GRU has to capture information with dependencies at long time scales, with the difference that it is simpler to compute and easier to implement [36]. The GRU network has the same gate mechanism as the LSTM with some differences. The most important is the combination of the input and of the forget gates into a single one and the lack of a distinct cell state.

2.2.3. Evaluation Metrics

To evaluate the models of the proposed investigation, the following statistical metrics are used:

$$MAPE = \frac{100}{N} \sum_{i=1}^N \left| \frac{x_i - \hat{x}_i}{x_i} \right|, \quad (1)$$

and

$$R^2 = 1 - \frac{\sum_{i=1}^N (x_i - \hat{x}_i)^2}{\sum_{i=1}^N (x_i - X)^2}, \quad (2)$$

where N is the number of predicted values, while x_i and \hat{x}_i are the observed and the predicted values at the i time step, respectively. X represents the average of the observed values. Thus, lower values of MAPE means better models performance, whereas lower values of R^2 indicates the worst models. The latter represents the explained variance by the models; its best value is 1. These two different metrics are widely used in models evaluation in different applications, comprising several engineering fields such as hydrology [38], hydraulics [39,40] and consumption forecasting [41].

3. Results and Discussion

3.1. Study of ANN Tuning Stochasticity

As already discussed, ANN are a class of machine learning algorithms with an important stochastic component, which is necessary to improve the training of the algorithms. In addition, the high data variability that characterises the water demand makes the hyperparameters tuning complex. These two aspects lead to an increase of the tuning results variability, making the hyperparameters' selection a tricky task. In order to assess the stochasticity of ANN algorithms, 20 runs of the grid search tuning have been performed using the combinations of nodes and layers reported in Table 2. This analysis has been done on *ts2* for the 4 horizons with set of inputs 1 (Table 1) involving 2 ANN architectures with different characteristics: the FFNN and the LSTM architectures. This analysis set up has been taken due to the intensive computational requirements of this test. The results in terms of MAPE are reported in Figure 4 under Box and Whisker plots.

Figure 4 highlights the variability that affects the simulations. In fact, each box, which represents a different configuration of nodes and layers of a defined problem and ANN architecture, shows a varying degree of variability of the simulations results by confirming the ANN stochasticity. This change of the results makes the choice of the best model more difficult and the tuning less reliable. It is also worth noting that the results of the 2 architectures have different behaviour. The LSTM shows overall better performance but also less variability compared to the FFNN. Besides, the FFNN resulting model with a simple structure is not properly able to deal with the more complex problem with large horizons like 24 and 168 h (panels (e) and (g)). This behaviour is even more underlined in Figure 5.

Figure 5 shows the MAPE distribution of the models by aggregating the models based on the number of layers. This figure highlights, even more, the better performances achieved by the LSTM models compared to the FFNN. In addition, the variability trends that affect these models are well explained for the different ANN structures, consisting of 1, 2 and 3 layers. The variability of the models appears to significantly increase for the FFNN models along with the widening of prediction horizons. This behaviour affects the parameters tuning. The variability that affects the different solution makes the selection of the more performing configuration difficult. A solution for this issue can be to select the median, or some other statistical indicators, to compare the different configuration. In this study, the median is chosen. To underline the problem of selecting the more suitable neural network configuration, it is proposed to show how the rank changes with the increase of the number of simulations. The rank assesses with a number ranging from 1 to 24 which is the configuration with the lowest MAPE achieved by the median. In other words, it shows

for each iteration which is the configuration that is making the less MAPE with the median. This analysis is reported only for the horizon of 24 h in Figure 6.

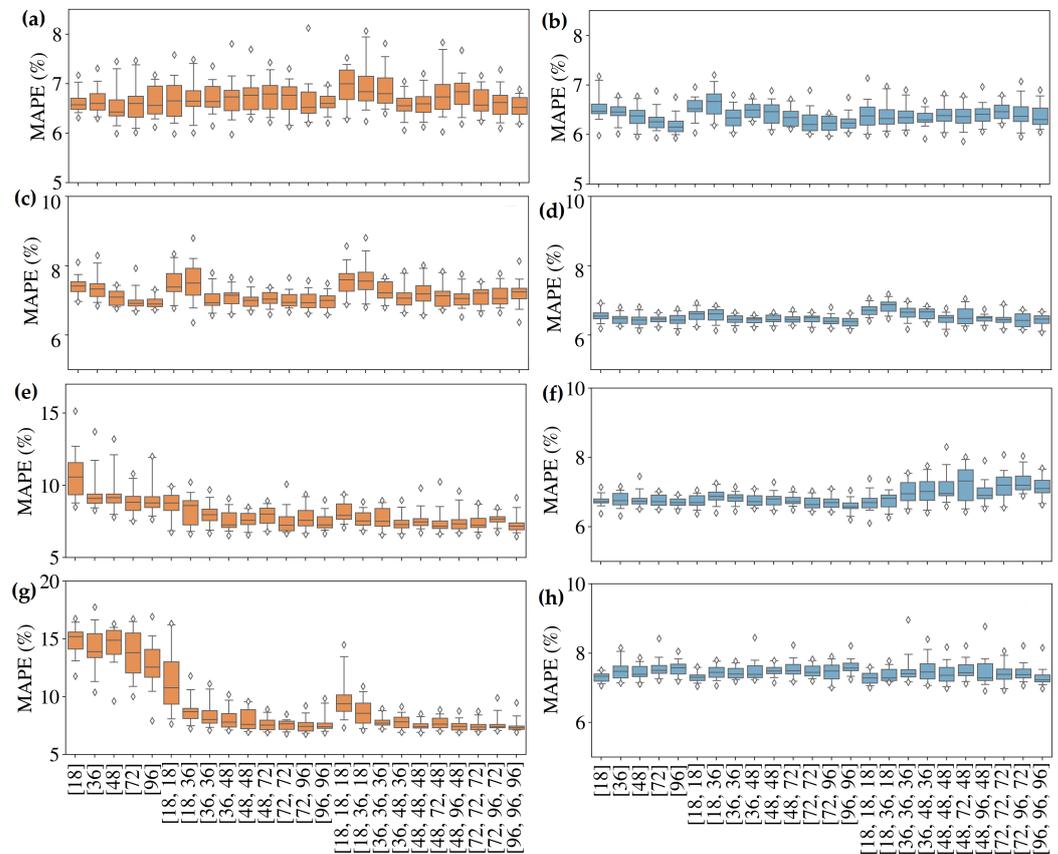


Figure 4. MAPE of the 20 simulations for each combination of nodes and layers of the *ts2* with set of lags 1. The results for the FFNN with 1 h, 6 h, 24 h and 168 h of horizon are reported in panel (a,c,e,g), respectively. Whereas, the results for the LSTM with 1 h, 6 h, 24 h and 168 h of horizon are reported in panel (b,d,f,h), respectively.

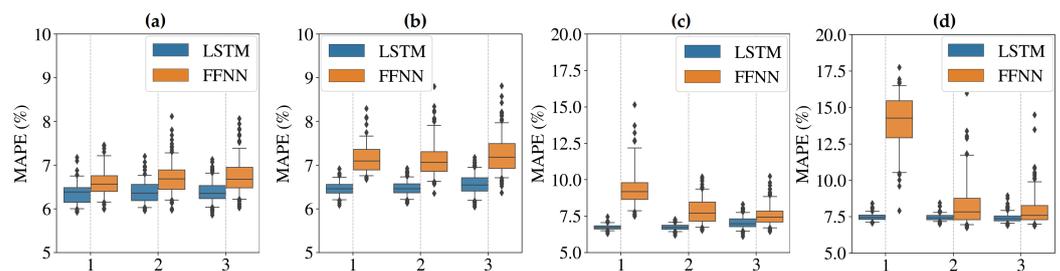


Figure 5. MAPE of the 20 simulation for each combination of nodes and layers grouped by the number of layers. In panel (a–d) are reported the results for the horizon of 1, 6, 24 and 168 h, respectively.

Figure 6 highlights the variability issue through the rank of the median results for each configuration updated at each iteration. Despite the rank keep changing during the simulations, some trends are stable and can be noticed in these plots. For instance, the panel (a) of Figure 6 shows the FFNN models for the 24 h horizon. In this case, the configuration with at least 2 layers have the better ranks, and consequently they are preferable compared to the single layers. In fact, as also depicted in panel e of Figure 2, the performance is significantly better both in terms of variability and also lower MAPE. With this analysis, it is possible to detect this behaviour and to select the more suitable configuration to deal with the variability ANN issue at least after the first 3 simulations. Regarding the LSTM model

depicted in panel (b) of Figure 4, a configuration over the three layers with 18 neurons each is not capable of achieving performance comparable to the configuration with less layers. As for the FFNN case, the behaviour of the models starts to be visible just after the first 3 simulations. Despite the high number of simulations that would be required to stabilise the ranks, due to the high computational requirements to perform the number of simulations that are needed, using 20 or more simulations can be difficult to perform. For this reason, it is proposed to make only 3 simulation as a compromise between results accuracy and time requirements.

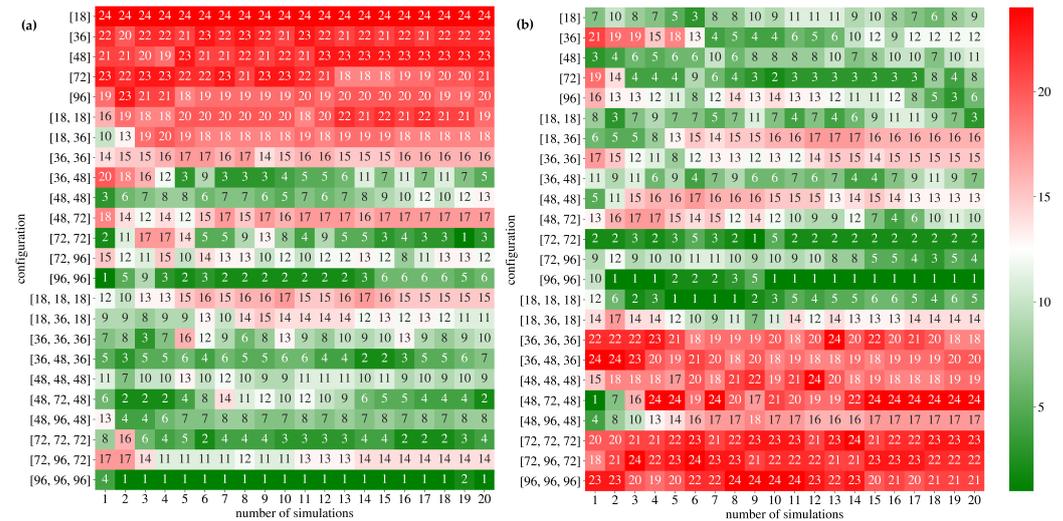


Figure 6. Heatmap of the medians rank for each configuration in the case of 24 h horizon. Panel (a) shows the FFNN models and Panel (b) the LSTM models.

3.2. Results Analysis and Comparison

In this section, the results of the iterative grid search procedure are presented via the 8 addressed problems, which regard the forecasting of two different datasets on the 4 prediction horizons. The synthetic and real datasets allow us to have a more robust analysis of ANN tuning in drinking water forecasting and also to check for possible differences due to the peculiarities of individual water demand time series. On the other side, forecasting horizons of 1 h, 6 h, 24 h and 168 h have been investigated in order to understand the tuning behaviour of different ANN architectures in dealing with challenges of various complexities; the larger the prediction horizon the more complex is the forecasting problem. In fact, ANN architectures respond differently to various forecasting horizons and also the required number of layers and nodes comprising the models changes.

On these 8 problems, the ANN architectures FFNN, LSTM, SRNN and GRU are analysed through the proposed iterative grid search involving 24 combination of layers and nodes and 2 set of inputs for each horizon as described in detail in Section 2.2 to Tables 1 and 2, respectively. According to the findings of Section 3.1, 3 repetitions of the grids search have been performed and therefore the models with the median MAPE selected, for the total number of 4608 models trained. Given the high number of results to present and comment about ANN tuning, the rest of the subsection is structured as follows: (Section 3.2.1) influence of time series, (Section 3.2.2) influence of inputs (lags selection) and (Section 3.2.3) models comparison within and between architectures. All results, consideration and findings are related exclusively on the ANN tuning of drinking water demand time series.

3.2.1. The Influence of Water Demand Data on Tuning

ANNs belongs to data-driven models meaning that the modelling is fully compelled by data. Therefore, the data have a significant impact on tuning due to the data-driven nature of ANN models. However, the analysis of 2 different time series is presented afterwards looking

to analogies related to the same principles governing drinking water demand. The comparisons between the synthetic *ts1* and real *ts2* are highlighted through Figures 7 and 8. The former shows the performance in R^2 of the FFNN, LSTM, SRNN and GRU architectures of the 2 datasets on the 4 horizons. In particular, Figure 7a depicts the best models, while Figure 7b shows the mean of the 24 models for each ANN architectures. Figure 8 illustrates the models variance for each architectures and the number of nodes of the ANN structure, in Figure 8a,b, respectively.

Several points emerge from Figure 7. First, the main difference of the two datasets is reflected in the different levels of the models' performance: best models of *ts1* ranging between 0.83 and 0.77 in terms of R^2 and between 0.95 and 0.86 models related to *ts2*. Second, the rank of the ANNs best models performance for each problem is largely stable. Indeed, LSTM and GRU are the most performing models followed by FFNN and SRNN for both the time series along the prediction horizons. Third, the performances of best models slightly decrease with increasing of prediction horizons according to the growing complexity of the problem. The only anomaly of the aforementioned considerations realises on the R^2 trend of SRNN for *ts1* that increases from 1 h to 24 h and only for 168 h the performance drops. It is likely due to the character of SRNN along with the synthetic nature of the dataset. On the other side, although Figure 8b shows a similar general behaviour for the mean models, it adds that the FFNN quickly decays with horizons extension. This highlights how important the layers and nodes tuning in the selection of the best ANN models is, especially when sharp performance changes are presented, such as for SRNN.

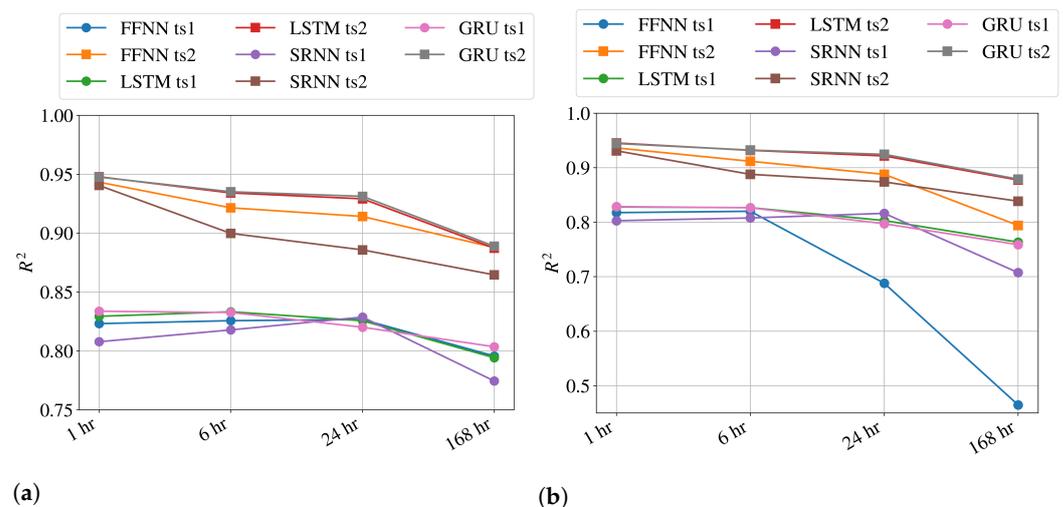


Figure 7. Results of ANN tuning (R^2 in y-axis) of *ts1* synthetic time series (solid lines and circle markers) and *ts2* real time series (solid lines and square markers) for the 4 forecasting horizons (x-axis) with the set of inputs 2. (a) depicts the best model for each architecture per horizon, while (b) shows the mean R^2 of the models for each architecture per horizon.

Figure 8a depicts the models variability within the 4 ANN architectures finding high variability over large horizons. In addition, the great variability of SRNN models is confirmed especially for one day and one week horizons. The number of nodes of the different best models are analysed in Figure 8b. A larger forecasting horizon causes a slight increase of the number of nodes of the best models independently on ANN architectures and time series. After this analysis, we can say that the influence of the water demand time series on ANN tuning regards mainly the performance levels rather than general trends on problem complexity and architectures. Thus, the results of this study can be deemed as general findings into the specific challenge of the drinking water forecasting because of the specificity of the problem driven usually by the same dynamics on different case studies.

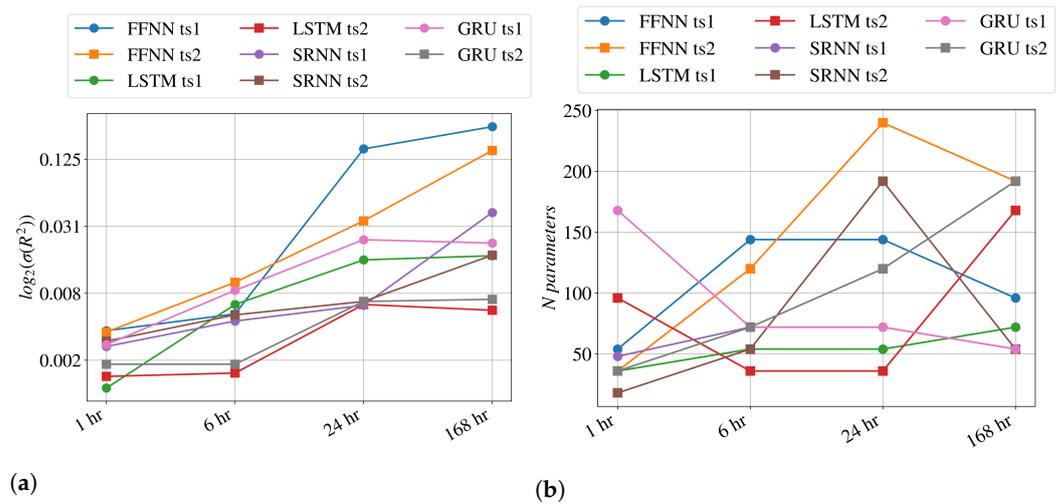


Figure 8. Models' performance variability within ANN architectures in $\log_2(\sigma(R^2))$ of ts_1 (solid lines and circle markers) and ts_2 (solid lines and square markers) for the 4 forecasting horizons (x-axis) with the set of inputs 2 in (a). Numbers of nodes of the best models in (b).

3.2.2. The Influence of Set of Inputs on Tuning

2 sets of inputs have been studied with the proposed tuning methodology; set of inputs 1 and 2 representing a shorter and a larger selection of time series lags, respectively. Both *set1* and *set2* have been defined for each prediction horizon as reported in Table 1.

Figure 9 shows the MAPE of the best models for each ANN architectures on ts_2 over the 4 horizons. Generally, the ANN tuning illustrates slightly better results of the models using *set2*, meaning a larger number of lags. This trend is inverted for the 168 h horizon even though the difference is very limited. Nevertheless, FFNN models present better results with lower number of lags (*set1*) for all the horizons except for 1 h.

The inference we can make about the number of input analyses is that not always so many inputs are better. Thus, not only the selection of the more significant inputs is important for a proper ANN tuning and then forecasting but also testing different numbers of significant inputs can help this process. Although generally the problem is more complex when the set of inputs has to be larger, testing different numbers of inputs is recommended given that it is not always true and depends on ANN architectures and prediction horizons.

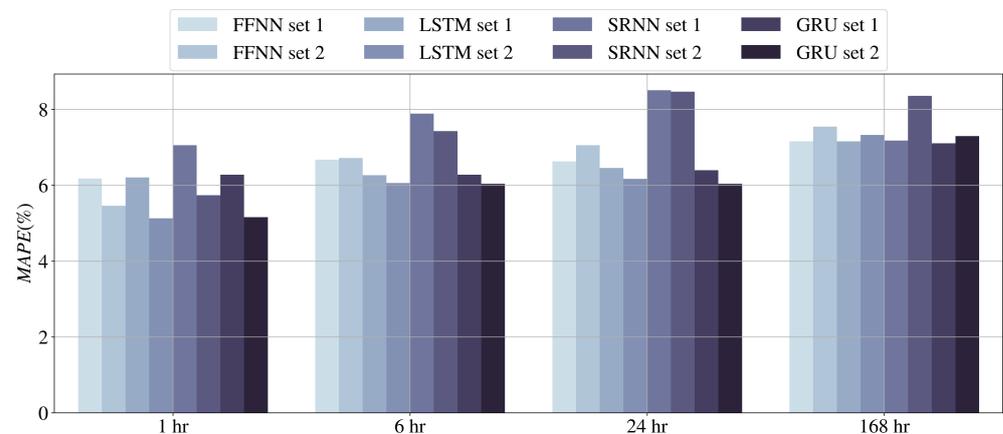


Figure 9. MAPE (%) of the best ANN models on the 4 forecasting horizons (x-axis) with two sets of inputs: *set1* (smaller number of lags) and *set2* (larger number of lags).

3.2.3. Within and between Architectures Comparison

Hereafter, the tuning results of the ts_2 are presented and commented on focusing on how the best performing models change within and between ANN architectures varying

the prediction horizons. All the models are summarised in Figure 10 through the MAPE of the median models resulting 3 iterations.

N. Layers	N. Nodes	FFNN				LSTM				SRNN				GRU			
		1 hr	6 hr	24 h	168 h	1 hr	6 hr	24 h	168 h	1 hr	6 hr	24 h	168 h	1 hr	6 hr	24 h	168 h
1 [18]		5.5	7.8	11.3	11.6	5.4	6.4	6.4	7.6	5.7	8.5	8.9	8.5	5.4	6.4	6.1	7.4
1 [36]		5.5	8.0	12.7	11.6	5.2	6.1	6.2	7.8	6.0	8.0	9.4	9.5	5.2	6.3	6.3	7.7
1 [48]		5.6	7.8	12.1	11.4	5.3	6.2	6.3	7.4	6.3	8.3	8.8	10.4	5.4	6.2	6.1	7.5
1 [72]		5.5	7.9	10.1	11.4	5.3	6.1	6.6	7.6	6.3	8.7	9.3	9.3	5.2	6.0	6.2	7.3
1 [96]		5.5	7.9	9.7	12.8	5.1	6.2	6.2	7.7	6.0	8.8	8.9	10.2	5.2	6.1	6.1	7.7
2 [18, 18]		5.5	7.3	9.8	7.9	5.4	6.5	6.4	8.1	5.8	8.1	8.8	8.8	5.6	6.4	6.1	7.8
2 [18, 36]		5.7	8.0	9.1	9.2	5.3	6.3	6.4	7.9	5.9	8.2	9.0	9.2	5.3	6.3	6.2	7.8
2 [36, 36]		5.5	6.8	7.5	8.1	5.3	6.3	6.3	7.6	6.0	8.5	9.0	9.5	5.3	6.2	6.1	7.9
2 [36, 48]		6.0	7.2	9.1	8.1	5.4	6.2	6.3	7.6	6.4	8.6	9.2	9.3	5.3	6.2	6.2	7.4
2 [48, 48]		5.8	7.1	7.8	7.7	5.3	6.1	6.4	7.3	6.2	8.3	9.2	9.0	5.2	6.1	6.1	7.5
2 [48, 72]		5.6	6.8	7.6	7.8	5.3	6.2	6.2	7.5	6.3	8.9	9.0	10.1	5.2	6.2	6.0	7.9
2 [72, 72]		5.8	7.0	7.7	7.9	5.2	6.1	6.3	7.8	6.2	9.2	8.5	9.7	5.2	6.1	6.2	7.4
2 [72, 96]		5.6	6.9	9.0	7.7	5.2	6.1	6.5	7.4	6.1	8.3	9.6	9.8	5.2	6.1	6.1	7.7
2 [96, 96]		5.9	6.8	8.2	8.4	5.1	6.1	6.3	7.8	6.1	8.7	8.5	10.2	5.2	6.1	6.2	7.3
3 [18, 18, 18]		5.8	7.7	8.8	7.8	5.4	6.5	6.5	7.5	6.6	7.4	9.3	8.4	5.8	6.6	6.0	7.9
3 [18, 36, 18]		5.8	6.9	8.0	8.2	5.6	6.6	6.3	7.8	6.5	7.8	9.4	8.4	6.0	6.3	6.4	7.6
3 [36, 36, 36]		5.8	7.4	7.7	9.0	5.4	6.3	6.3	7.5	6.6	7.8	8.9	9.0	6.0	6.3	6.2	7.4
3 [36, 48, 36]		5.7	6.9	7.7	8.0	5.3	6.5	6.4	8.0	6.2	8.0	9.5	8.4	5.4	6.6	6.2	7.6
3 [48, 48, 48]		6.1	6.9	7.7	7.9	5.4	6.3	6.4	7.9	6.4	8.0	8.8	9.1	5.9	6.7	6.4	8.0
3 [48, 72, 48]		5.7	7.4	7.8	8.3	5.4	6.1	6.5	7.9	6.7	8.2	9.4	8.6	5.8	6.6	6.3	7.9
3 [48, 96, 48]		5.7	6.8	9.3	7.6	5.5	6.3	6.5	7.7	6.6	8.0	9.2	9.6	5.7	6.4	6.4	7.6
3 [72, 72, 72]		5.8	7.0	7.8	7.7	5.4	6.5	6.7	7.6	6.3	8.1	9.3	10.2	5.6	6.1	6.5	7.9
3 [72, 96, 72]		5.6	6.7	7.1	7.7	5.6	6.2	6.8	7.7	6.7	8.2	9.1	9.1	5.8	6.5	6.7	7.8
3 [96, 96, 96]		5.8	7.0	7.3	8.1	5.4	6.2	6.5	8.3	6.4	8.2	9.5	9.3	5.7	6.2	6.6	8.1

Figure 10. MAPE of the models of ts_2 time series for each architectures on the 4 forecasting horizons with the set of inputs 2.

Starting from the FFNN, it is clear that this ANN architecture requires more articulated structures to deal with longer forecasting horizons. In fact, the best model for 1 h horizon has 1 layer; instead it has 2 layers for 6 h and for 24 h, while the best model for 168 h comprises 3 layers. On the other side, except for the simplest problem of 1 h, the worst model results are those with 1 layer meaning that FFNN needs a deep structure to perform on the forecasting problem. Generally, the variability between the models of FFNN is marked, making the layers and nodes a tricky step in ANN modelling and forecasting.

The LSTM and GRU shows similar results both in terms of overall performance and models trends. The best models are identified in simple ANN structures, independently from the horizons size. It means that 1 and 2 layers are recommended over 3 layer models because of better performance, computational time and parsimony in the selection of the configuration numbers. The variability of these models is low with a slight increase at wider horizons ensuring a lower impact in the case of a wrong selection of best model during the tuning. Generally, the worst models of these two architectures are identified among the complex structures with 3 layers for the 2 time series analysed.

Regarding the SRNN architectures, the models' performance shows quite significant variability for each prediction horizons. No clear trends are detectable which can indicate high variability of SRNN. This is highlighted by the similarity in structure between the best and worst models per each horizons. Specifically, 1 layer nodes results the best for 1 h horizon, whereas low number of nodes on 3 layers or large number of nodes on 2 layers are the best configurations for the others larger horizons. For this ANN architecture, the worst models are generally for the 2 proposed cases the complex ones consisting of 3 layers and high number of nodes.

The between models comparison of the different ANN architectures is proposed afterwards with the support of Figure 11, which shows a synthesis of Figure 10 by dividing the models in homogeneous groups. The models are grouped according to the ANN layers and nodes configuration into single-layer models, two-layers and low numbers of nodes up to [48, 48], two-layers and high numbers of nodes from [48, 48] onwards, three-layers and low numbers of nodes up to [48, 48, 48] and three-layers and high numbers of nodes from [48, 48, 48] onwards. The tuning shows the LSTM and GRU best results are obtained with 2 layers and low number of nodes independently of the forecasting horizons. On the other side, FFNN has a significant dependence between number of layers, nodes and horizon size. No clear trend is found for SRNN. On the performance point of view, all the

ANN architectures are high performance and low variability over 1 h horizon. The other horizons, which refer to more challenging forecasting problems, highlight that the LSTM and GRU are more accurate than FFNN and SRNN.

N. Layers	N. Nodes	FFNN				LSTM				SRNN				GRU			
		1 hr	6 hr	24 h	168 h	1 hr	6 hr	24 h	168 h	1 hr	6 hr	24 h	168 h	1 hr	6 hr	24 h	168 h
1	[18]	5.5	7.9	11.2	11.8	5.3	6.2	6.3	7.6	6.1	8.4	9.1	9.6	5.3	6.2	6.2	7.5
1	...																
1	[96]																
2	[18, 18]	5.7	7.3	8.7	8.2	5.4	6.3	6.3	7.7	6.1	8.3	9.0	9.2	5.3	6.2	6.2	7.7
2	...																
2	[48, 48]																
2	[48, 72]	5.7	6.9	8.2	7.9	5.2	6.1	6.3	7.6	6.2	8.8	8.9	9.9	5.2	6.1	6.1	7.6
2	...																
2	[96, 96]																
3	[18, 18, 18]	5.9	7.2	8.0	8.2	5.4	6.4	6.4	7.7	6.5	7.8	9.2	8.7	5.8	6.5	6.3	7.7
3	...																
3	[48, 48, 48]																
3	[48, 72, 48]	5.7	7.0	7.9	7.9	5.4	6.3	6.6	7.8	6.6	8.2	9.3	9.3	5.7	6.4	6.5	7.8
3	...																
3	[96, 96, 96]																

Figure 11. MAPE per group of models of ts_2 time series for each architectures on the 4 forecasting horizons with the set of inputs 2.

4. Conclusions

This article presents a novel in-depth study about the tuning of ANN structure, including nodes and layers, in the field of drinking water forecasting. The four state-of-the-art ANN architectures, which are the Feed Forward Neural Network, the Long Short Term Memory the Simple Recurrent Neural Network and the Gated Recurrent Unit, have been investigated on different short prediction horizon ranging from 1 h and 1 week. At this aim, an iterative grid search procedure has been proposed. This investigation comprises 2 time series, 4 forecasting horizon and several combinations of nodes, layers and set of inputs. The final intent is to identify meaningful indications for a proper build-up of ANN models related to water demand time series. Thus, the final remarks about ANN structure tuning for drinking water demand forecasting are summarised in the following points.

- i Layers and nodes tuning is affected by variability in the results due to the stochastic character of ANN which use randomness in several ways, such as weight initialisation, sample shuffle and k-fold cross-validation. This causes changes in the models’ performance which can lead to the misleading model selection with a traditional tuning search. To overcome this limit, an interactive tuning procedure is recommended with the number of iteration defined with a compromise between accuracy and computational intensity.
- ii Generally, both the numbers of inputs and the number of layers and nodes grows with the increasing size of the forecasting horizon. However, this study shows that different number of inputs have to be tested in the hyperparameters tuning because a prior selection is not possible among the various significant time series lags. In addition, the structure of the best ANN models changes depending on the forecasting horizons, showing different trends also depending on the ANN architectures.
- iii The tuning of LSTM and GRU provide the best performing models with the lowest variability among models. For the tested time series, 2 layers models provide stable and accurate performance for the different horizons. Instead, FFNN and SRNN have good performance for the 1 h horizons but lose accuracy with the extension of the prevision window. FFNN needs more articulated ANN structures for more complex forecasting problems, while SRNN do not shows clear trends.

Author Contributions: Conceptualisation, A.M., A.Z. and M.R.; Data curation, A.M. and A.Z.; Formal analysis, A.M. and A.Z.; Funding acquisition, M.R.; Investigation, A.M. and A.Z.; Methodology, A.M. and A.Z.; Project administration, M.R.; Software, A.Z.; Supervision, M.R.; Validation, A.M. and A.Z.; Visualisation, A.M. and A.Z.; Writing—original draft, A.M. and A.Z.; Writing—review and editing, A.M., A.Z. and M.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research has been funded by the Interfaculty Project of the Free University of Bozen-Bolzano: “TESES-URB (ID 2019)”.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ANN	Artificial Neural Networks
FFNN	Feed Forward Neural Network
GRU	Gated Recurrent Unit
LSTM	Long Short Term Memory
MAPE	Mean Absolute Percentage Error
SARIMA	Seasonal Autoregressive Integrated Moving Average
SRNN	Simple Recurrent Neural Network
WDS	Water Supply Systems

References

1. Rahaman, M.M.; Varis, O. Integrated water resources management: Evolution, prospects and future challenges. *Sustain. Sci. Pract. Policy* **2005**, *1*, 15–21. [[CrossRef](#)]
2. Brown, C.M.; Lund, J.R.; Cai, X.; Reed, P.M.; Zagona, E.A.; Ostfeld, A.; Hall, J.; Characklis, G.W.; Yu, W.; Brekke, L. The future of water resources systems analysis: Toward a scientific framework for sustainable water management. *Water Resour. Res.* **2015**, *51*, 6110–6124. [[CrossRef](#)]
3. Righetti, M.; Bort, C.M.G.; Bottazzi, M.; Menapace, A.; Zanfei, A. Optimal Selection and Monitoring of Nodes Aimed at Supporting Leakages Identification in WDS. *Water* **2019**, *11*, 629. [[CrossRef](#)]
4. Ghalekhondabi, I.; Ardjmand, E.; Young, W.A.; Weckman, G.R. Water demand forecasting: Review of soft computing methods. *Environ. Monit. Assess.* **2017**, *189*, 1–13. [[CrossRef](#)]
5. Donkor, E.A.; Mazzuchi, T.A.; Soyer, R.; Alan Roberson, J. Urban water demand forecasting: Review of methods and models. *J. Water Resour. Plan. Manag.* **2014**, *140*, 146–159. [[CrossRef](#)]
6. Guo, G.; Liu, S.; Wu, Y.; Li, J.; Zhou, R.; Zhu, X. Short-term water demand forecast based on deep learning method. *J. Water Resour. Plan. Manag.* **2018**, *144*, 04018076. [[CrossRef](#)]
7. Herrera, M.; Torgo, L.; Izquierdo, J.; Pérez-García, R. Predictive models for forecasting hourly urban water demand. *J. Hydrol.* **2010**, *387*, 141–150. [[CrossRef](#)]
8. Jain, A.; Ormsbee, L.E. Short-term water demand forecast modeling techniques—Conventional methods versus AI. *J. Am. Water Work. Assoc.* **2002**, *94*, 64–72. [[CrossRef](#)]
9. Braun, M.; Bernard, T.; Piller, O.; Sedehizade, F. 24-h demand forecasting based on SARIMA and support vector machines. *Procedia Eng.* **2014**, *89*, 926–933. [[CrossRef](#)]
10. Mombeni, H.A.; Rezaei, S.; Nadarajah, S.; Emami, M. Estimation of water demand in Iran based on SARIMA models. *Environ. Model. Assess.* **2013**, *18*, 559–565. [[CrossRef](#)]
11. House-Peters, L.A.; Chang, H. Urban water demand modeling: Review of concepts, methods, and organizing principles. *Water Resour. Res.* **2011**, *47*. [[CrossRef](#)]
12. Ghiassi, M.; Zimbra, D.K.; Saidane, H. Urban water demand forecasting with a dynamic artificial neural network model. *J. Water Resour. Plan. Manag.* **2008**, *134*, 138–146. [[CrossRef](#)]
13. Adamowski, J.; Karapataki, C. Comparison of multivariate regression and artificial neural networks for peak urban water-demand forecasting: Evaluation of different ANN learning algorithms. *J. Hydrol. Eng.* **2010**, *15*, 729–743. [[CrossRef](#)]
14. Ambrosio, J.K.; Brentan, B.M.; Herrera, M.; Luvizotto, E.; Ribeiro, L.; Izquierdo, J. Committee machines for hourly water demand forecasting in water supply systems. *Math. Probl. Eng.* **2019**, *2019*, 1–11. [[CrossRef](#)]
15. Xenochristou, M.; Kapelan, Z. An ensemble stacked model with bias correction for improved water demand forecasting. *Urban Water J.* **2020**, *17*, 212–223. [[CrossRef](#)]
16. Weerts, H.J.; Mueller, A.C.; Vanschoren, J. Importance of tuning hyperparameters of machine learning algorithms. *arXiv* **2020**, arXiv:2007.07588.
17. Bergstra, J.; Bengio, Y. Random search for hyper-parameter optimization. *J. Mach. Learn. Res.* **2012**, *13*, 281–305.

18. Candelieri, A.; Giordani, I.; Archetti, F. Automatic configuration of Kernel-based clustering: An optimization approach. In Proceedings of the International Conference on Learning and Intelligent Optimization, Nizhny Novgorod, Russia, 19–21 June 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 34–49.
19. Joy, T.T.; Rana, S.; Gupta, S.; Venkatesh, S. Hyperparameter tuning for big data using Bayesian optimisation. In Proceedings of the 2016 23rd International Conference on Pattern Recognition (ICPR), Cancun, Mexico, 4–8 December 2016; pp. 2574–2579.
20. Candelieri, A.; Giordani, I.; Archetti, F.; Barkalov, K.; Meyerov, I.; Polovinkin, A.; Sysoyev, A.; Zolotykh, N. Tuning hyperparameters of a SVM-based water demand forecasting system through parallel global optimization. *Comput. Oper. Res.* **2019**, *106*, 202–209. [[CrossRef](#)]
21. Shirkoobi, M.G.; Doghri, M.; Duchesne, S. Short-term water demand predictions coupling an Artificial Neural Network model and a Genetic Algorithm. *Water Supply* **2021**. [[CrossRef](#)]
22. Perea, R.G.; Poyato, E.C.; Montesinos, P.; Díaz, J.A.R. Optimisation of water demand forecasting by artificial intelligence with short data sets. *Biosyst. Eng.* **2019**, *177*, 59–66. [[CrossRef](#)]
23. Napolitano, J.; Sechi, G.M. Scenario Optimization of Complex Water Supply Systems for Energy Saving and Drought-Risk Management. In Proceedings of the International Conference on Numerical Computations: Theory and Algorithms, Crotone, Italy, 15–21 June 2019; Springer: Berlin/Heidelberg, Germany, 2019; pp. 557–569.
24. Tricarico, C.; Morley, M.S.; Gargano, R.; Kapelan, Z.; De Marinis, G.; Savić, D.; Granata, F. Integrated optimal cost and pressure management for water distribution systems. *Procedia Eng.* **2014**, *70*, 1659–1668. [[CrossRef](#)]
25. Napolitano, J.; Sechi, G.M.; Zuddas, P. Scenario optimisation of pumping schedules in a complex water supply system considering a cost–risk balancing approach. *Water Resour. Manag.* **2016**, *30*, 5231–5246. [[CrossRef](#)]
26. Raciti, M.; Cucurull, J.; Nadjm-Tehrani, S. Anomaly detection in water management systems. In *Critical Infrastructure Protection*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 98–119.
27. Tricarico, C.; Morley, M.S.; Gargano, R.; Kapelan, Z.; Savić, D.; Santopietro, S.; Granata, F.; de Marinis, G. Optimal energy recovery by means of pumps as turbines (PATs) for improved WDS management. *Water Sci. Technol. Water Supply* **2018**, *18*, 1365–1374. [[CrossRef](#)]
28. Schratz, P.; Muenchow, J.; Iturrutxa, E.; Richter, J.; Brenning, A. Hyperparameter tuning and performance assessment of statistical and machine-learning algorithms using spatial data. *Ecol. Model.* **2019**, *406*, 109–120. [[CrossRef](#)]
29. Wang, H.; Emmerich, M.; Preuss, M.; Plaat, A. Analysis of Hyper-Parameters for Small Games: Iterations or Epochs in Self-Play? *arXiv* **2020**, arXiv:2003.05988.
30. Khalid, R.; Javaid, N. A survey on hyperparameters optimization algorithms of forecasting models in smart grid. *Sustain. Cities Soc.* **2020**, *61*, 102275. [[CrossRef](#)]
31. Menapace, A.; Zanfei, A.; Felicetti, M.; Avesani, D.; Righetti, M.; Gargano, R. Burst Detection in Water Distribution Systems: The Issue of Dataset Collection. *Appl. Sci.* **2020**, *10*, 8219. [[CrossRef](#)]
32. Chollet, F. Keras. 2015. Available online: <https://github.com/fchollet/keras> (accessed on 5 January 2021).
33. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. Tensorflow: A system for large-scale machine learning. In Proceedings of the 12th USENIX Symposium on Operating Systems design and Implementation (OSDI 16), Savannah, GA, USA, 2–4 November 2016; pp. 265–283.
34. Bengio, Y.; Simard, P.; Frasconi, P. Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. Neural Netw.* **1994**, *5*, 157–166. [[CrossRef](#)] [[PubMed](#)]
35. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)]
36. Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv* **2014**, arXiv:1406.1078.
37. Graves, A. Generating sequences with recurrent neural networks. *arXiv* **2013**, arXiv:1308.0850.
38. Jalalkamali, A.; Sedghi, H.; Manshouri, M. Monthly groundwater level prediction using ANN and neuro-fuzzy models: A case study on Kerman plain, Iran. *J. Hydroinform.* **2011**, *13*, 867–876. [[CrossRef](#)]
39. Ebtahaj, I.; Bonakdari, H. Bed load sediment transport estimation in a clean pipe using multilayer perceptron with different training algorithms. *KSCE J. Civ. Eng.* **2016**, *20*, 581–589. [[CrossRef](#)]
40. Brentan, B.M.; Luvizotto, E., Jr.; Herrera, M.; Izquierdo, J.; Pérez-García, R. Hybrid regression model for near real-time urban water demand forecasting. *J. Comput. Appl. Math.* **2017**, *309*, 532–541. [[CrossRef](#)]
41. Shabani, S.; Candelieri, A.; Archetti, F.; Naser, G. Gene expression programming coupled with unsupervised learning: A two-stage learning process in multi-scale, short-term water demand forecasts. *Water* **2018**, *10*, 142. [[CrossRef](#)]