

## Article

# A Hybrid Bimodal LSTM Architecture for Cascading Thermal Energy Storage Modelling

Athanasios Anagnostis <sup>1,2</sup>, Serafeim Moustakidis <sup>3,4</sup>, Elpiniki Papageorgiou <sup>1,5,\*</sup> and Dionysis Bochtis <sup>1</sup>

- <sup>1</sup> CERTH/IBO—Centre for Research and Technology Hellas, Institute of Bio-Economy and Agri-Technology, 57001 Thessaloniki, Greece; a.anagnostis@certh.gr (A.A.); d.bochtis@certh.gr (D.B.)
- <sup>2</sup> Department of Computer Science, University of Thessaly, 35100 Lamia, Greece
- <sup>3</sup> Systems & Control Research Centre, City University of London, Northampton Square, London EC1V 0HB, UK; serafeim.moustakidis.1@city.ac.uk or s.moustakidis@aideas.eu
- <sup>4</sup> AIDEAS OÜ, Narva mnt 5, 10117 Tallinn, Estonia
- <sup>5</sup> Department of Energy Systems, Geopolis Campus, University of Thessaly, 41500 Larisa, Greece
- \* Correspondence: elpinikipapageorgiou@uth.gr; Tel.: +30-6936064613

**Abstract:** Modelling of thermal energy storage (TES) systems is a complex process that requires the development of sophisticated computational tools for numerical simulation and optimization. Until recently, most modelling approaches relied on analytical methods based on equations of the physical processes that govern TES systems' operations, producing high-accuracy and interpretable results. The present study tackles the problem of modelling the temperature dynamics of a TES plant by exploring the advantages and limitations of an alternative data-driven approach. A hybrid bimodal LSTM (H2M-LSTM) architecture is proposed to model the temperature dynamics of different TES components, by utilizing multiple temperature readings in both forward and bidirectional fashion for fine-tuning the predictions. Initially, a selection of methods was employed to model the temperature dynamics of individual components of the TES system. Subsequently, a novel cascading modelling framework was realised to provide an integrated holistic modelling solution that takes into account the results of the individual modelling components. The cascading framework was built in a hierarchical structure that considers the interrelationships between the integrated energy components leading to seamless modelling of whole operation as a single system. The performance of the proposed H2M-LSTM was compared against a variety of well-known machine learning algorithms through an extensive experimental analysis. The efficacy of the proposed energy framework was demonstrated in comparison to the modelling performance of the individual components, by utilizing three prediction performance indicators. The findings of the present study offer: (i) insights on the low-error performance of tailor-made LSTM architectures fitting the TES modelling problem, (ii) deeper knowledge of the behaviour of integral energy frameworks operating in fine timescales and (iii) an alternative approach that enables the real-time or semi-real time deployment of TES modelling tools facilitating their use in real-world settings.

**Keywords:** bi-modal LSTM; cascading energy framework; district heating; thermal energy storage



**Citation:** Anagnostis, A.; Moustakidis, S.; Papageorgiou, E.; Bochtis, D. A Hybrid Bimodal LSTM Architecture for Cascading Thermal Energy Storage Modelling. *Energies* **2022**, *15*, 1959. <https://doi.org/10.3390/en15061959>

Academic Editor: Abu-Siada Ahmed

Received: 19 January 2022

Accepted: 3 March 2022

Published: 8 March 2022

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



**Copyright:** © 2022 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/>).

## 1. Introduction

District heating and cooling (DHC) networks are highly complex systems consisting of a large number of distributed entities. Modelling and optimization play a crucial role towards the effective management of such multi-vector energy systems. Several attempts have been reported in the recent literature using physics-based approaches for the modelling of individual DHC components [1]. Most of these studies decompose complex energy systems into a series of simpler input-output energy hubs [2]. Design stage assumptions are often adopted for modelling the thermal behavior of buildings, whereas mathematical functions are used to calculate the performance of energy conversion units assuming constant thermal and electrical efficiencies. Pivotal to the optimization

of a decentralized DHC network is the adoption of holistic management methodologies that take into account all aspects of the system. To optimize today's multi-vector energy systems, accurate internal models of all the primary and secondary energy sources need to be produced.

Advanced data-driven techniques and machine learning (ML) present a promising alternative solution that could enhance or even replace the simplified and static modeling approaches with more detailed and at the same time more effective models. ML is a subfield of data science in which algorithms and data-driven models capture and predict complex non-linear relationships that exist in data [3]. The algorithms for ML have been around since the middle of the previous century, based on statistical theories that go back to the 18th century (such as Bayes' theorem [4]). However, modern computers have reached such a performance level that made ML easy to implement and fast to train, thus rebooting the interest in this area. In recent years, ML has seen exceptional growth, and new areas have sprung from it including energy management and optimization. Deep Learning [5] is a relatively new sector that utilizes Artificial Neural Networks (ANNs) that are built with a large number of layers (deep network) outperforming the conventional algorithms in heavy tasks with the use of big data.

Several ML algorithms have been investigated and deployed for modelling, load forecasting and optimization of processes in DHC systems [6]. Among the different ML algorithms, ANNs have found the largest applicability as an alternative to the numerical models for multiple applications [7]. Some examples include model-based predictive control (MPC) for heating, ventilation, and air conditioning (HVAC) systems [8], as well as for thermal analysis of heat exchangers [9]. Focusing on the application of ML in energy load prediction, there are many research studies demonstrating high performance and providing insights and information that improve the efficiency and productivity of energy-related tasks [10,11]. Specifically, ANNs have been used to predict energy consumption in bioclimatic buildings [12], to provide reliable control mechanisms for TES systems [13], as well as for cooling demand prediction in commercial buildings [14–16]. They have been also applied for performance prediction of a solar thermal energy system used in a domestic hot water and space heating application [17], for modelling heating and cooling loads of DHC systems [18], and for process optimization by predicting energy-efficient building designs [19]. ANNs combined with genetic algorithms were applied for process optimization of a solar-energy system with the ultimate object of maximizing energy efficiency and the associated economic benefits [20]. Finally, an ANN-based-framework for the optimal integration of solar-assisted district heating in different urban sized communities has been proposed in [21].

Aside from ANNs, other various ML models were applied for the prediction of energy demand [22]; Bayesian nets and reinforcement methods were used for heat load prediction in district heating systems [23,24], fuzzy networks were implemented for the prediction of energy demand concerning renewable energy systems [25], SVMs were employed for predictive energy management between solar energy source and an energy storage [26] and ensembles of online ML algorithms were used for operational demand forecasting in DH systems [18,27] as representative examples. The need of incorporating ML-based forecasting algorithms within advanced control strategies has been highlighted in [28,29] for the transformation of the current district heating and cooling systems to more efficient automated systems.

The popularity of deep learning has been considerably increased over recent years leading many studies to focus on finding solutions for energy-based problems with the use of deep architectures of common ML methods [30]. Specifically, deep reinforcement learning has been employed for simulating energy savings and demand response in buildings [31], as well as in the optimization of the energy demand and supply of smart grids [32]. Deep artificial neural networks have been employed on load prediction of a district cooling system combined with physics-based TES modelling [33], whereas in [34] deep recurrent neural networks, that offer great performance in time-dependent problems, were used for the prediction of the heating demand in commercial buildings. Deep reinforcement

learning has been adopted in [35] as a temperature control strategy in a Chinese district heating use case. Long Short-Term Memory (LSTM) and deep neural networks have been proposed for district heating load prediction in [36,37], respectively. Deep learning and specifically auto-encoders have also been developed for assisted energy consumption profiling in buildings using smart meter data in [38].

In the context of the aforementioned increased adoption of ML in various DHC domains, TES modeling has attracted some interest from the scientific community, however the modelling task has been mainly treated with analytical mathematical tools. Thermal energy storage is a key technology in DHC networks that brings a number of benefits such as full recovery of the heat produced, maximization of the system's efficiency and reliability as well as low operating and maintenance costs. To the best of the authors' knowledge, the efforts made so far on the modelling of DHC thermal energy storage are limited to analytical mathematical models or conventional ML techniques. A quasi-steady state mathematical model was employed in [39] to model the performance of a storage tank which was part of a DHC system of an institutional building located in a Mediterranean climate. Petrocelli and Lezzi investigated the effect of storage tank size on a wood pellet boiler in the control strategy [40]. Fluid dynamics, both three-dimensional numerical and computational, have been also utilised to simulate the performance and dynamics of various storage tank configurations [41,42]. Thermofluiddynamic models have also been investigated for carrying out systematic calculations on different design options for parabolic trough collectors [43]. Particularly in applications where solar collectors are part of the system, analytical approaches that rely on thermodynamic principles are rather complicated [44], computationally expensive and cannot be used in real-time application. A finite difference model with an electrical analogy was investigated in [45] for calculating the outlet temperature in a novel concept of solar collectors. Thermodynamic parameters and weather conditions were, respectively, used in [46] to calculate outlet temperatures and the efficiency of solar collectors that had their glass surface replaced with aerogel-filled polycarbonate panels. Similar thermodynamic parameters and weather conditions were used in [47] to predict air temperatures with efficiency that reached relative error of 7%. Even though these analytical models offer high accuracy, they are computationally intensive, therefore online control is not feasible, due to difficulties in exhaustively exploring the parametric space within specific time intervals.

A few studies have been reported in the recent literature where ML have been applied for TES modelling. Géczy-Víg et al. [48] used ANNs to calculate outlet temperatures in various layers of thermal storage tanks, Caner et al. [49] and Sözen et al. [50] calculated the efficiency of solar collectors, whereas in [51] both outlet temperatures and efficiency of a solar air heater system were estimated though the use of ML. Variations of ANNs along with SVMs were used by Liu et al. [52] in order to calculate the heat collection rate and heat loss coefficient of water-in-glass evacuated tube solar water heaters, whereas in García et al. [53] ANNs were employed for predictive modelling in biomass torrefaction within an energy treatment process. ANNs were implemented by Kalogirou et al. [54] in solar water heater systems to calculate both the inlet/outlet temperatures of components and establish a fault detection mechanism by comparing actual and predicted outlet temperatures of the TES system. Data-driven models have been commonly used as fault detection systems as well as for the online control of systems in [55]. Kalogirou et al. [56] used ANNs to predict both the energy output of a large-scale solar thermal system and the output temperature of a storage tank. However, despite the fact that operational optimization requires fine timescales, this study was limited to total daily energy outputs. One of the challenges that the aforementioned ANN-based approaches face is that they are not designed to capture sequential information in the input data. Moreover, most of the reported studies focus on modelling specific components of the integrated system and they typically lack a holistic solution that could include and correlate data coming from different components.

As an alternative to the above limitations, this paper proposes a novel data-driven DL-based cascading methodology of thermal energy storage modelling in DHC. A custom

architecture of a hybrid bi-modal LSTM neural network is being proposed for tackling a number of technical challenges that have been commonly encountered in the specific modelling paradigm: (i) Compared to the existing analytical solutions, the proposed modelling methodology does not require any prerequisites or domain knowledge with respect to the operation of the different components; (ii) It is suitable for modelling time series since it possesses an internal state or short-term memory that captures the sequential information present in the input data i.e., dependency between successive time steps while making predictions; (iii) It has been designed to provide a holistic solution in which data from different energy components are combined in order to produce the final outputs; (iv) It enables the real-time or semi-real time deployment of the modeling tools facilitating its use in real-world settings. A variety of ML models, covering most of the algorithmic families, is used as benchmark in order to evaluate and compare the efficiency of the proposed methodology. All models were trained on data collected from sensors over a long time period to model the operation of a heat storage tank facility. The modelling approach was conducted under two approaches: (i) the storage tank was modelled as a separate unit, (ii) the energy storage model was integrated in a cascading architecture that also involves units that model the operation of the connected sub-systems (solar panels and heat exchanger). The performance of all ML approaches for the TES modeling is demonstrated by requiring only the availability of historical datasets of inlet-outlet temperatures of components. The proposed modelling of thermal energy storage can provide operational insights on the storage capacity of the system, thus providing exploitable information to network operators.

The rest of the paper is structured as follows. Section 2 gives a short description of the problem along with information about the acquired data. The proposed DL methodology is presented in Section 3, whereas Section 4 presents the modelling results along with discussions. Conclusions and future directions are provided in Section 5.

## 2. Materials and Methods

### 2.1. Problem Description

In most district heating systems, energy storage is a tank that contains water, which acts as an energy buffer for the distribution of hot water in the network. A real DHS located in Vranksko (Slovenia) was considered in our analysis [57]. Vranksko's TES consists of a 100 m<sup>3</sup> stratified water tank that contains water, has three different inputs/outputs (on three levels, bottom, middle and top), and receives its energy from two sources. The first source is an array of solar panels (840 m<sup>2</sup> of solar collectors, 370 kW) that provides energy in the form of hot water to the system. The second source is a cluster of biomass boilers (with a total heating power capacity of 4.8 MW) that operate on wood chips and oil, as well as a combined heat and power (CHP) plant that operates on wood chips. These separate boilers are considered as a unified, single source in our analysis.

The storage tank in Vranksko has a semi-automatic operation selected based on the system operators' experience. The function ensures that the valves will automatically change the direction of the biomass feed directly into the network in the need of increased consumption demands within a small timeframe (sudden demand). In principle, when the water temperature in the solar collectors is high, the system prioritizes the operation of the solar panels starting the pumps and turning the valves to the direction so that they feed in the storage tank. On the other hand, if there is insufficient solar power (cloudy days) the CHP plant takes over providing enough energy to the tank. Additionally, when the tank is at full capacity (has reached 90 °C) at the top section, the valve at the bottom of the tank automatically opens so the solar collectors and/or the CHP and/or the biomass boiler feed the tank from the bottom. A graphical representation of the components of the energy storage facility is shown in Figure 1. Where the basic components, that need to be modelled, are presented as functions in boxes, and the inputs are presented in three different categories, normal, return and universal.

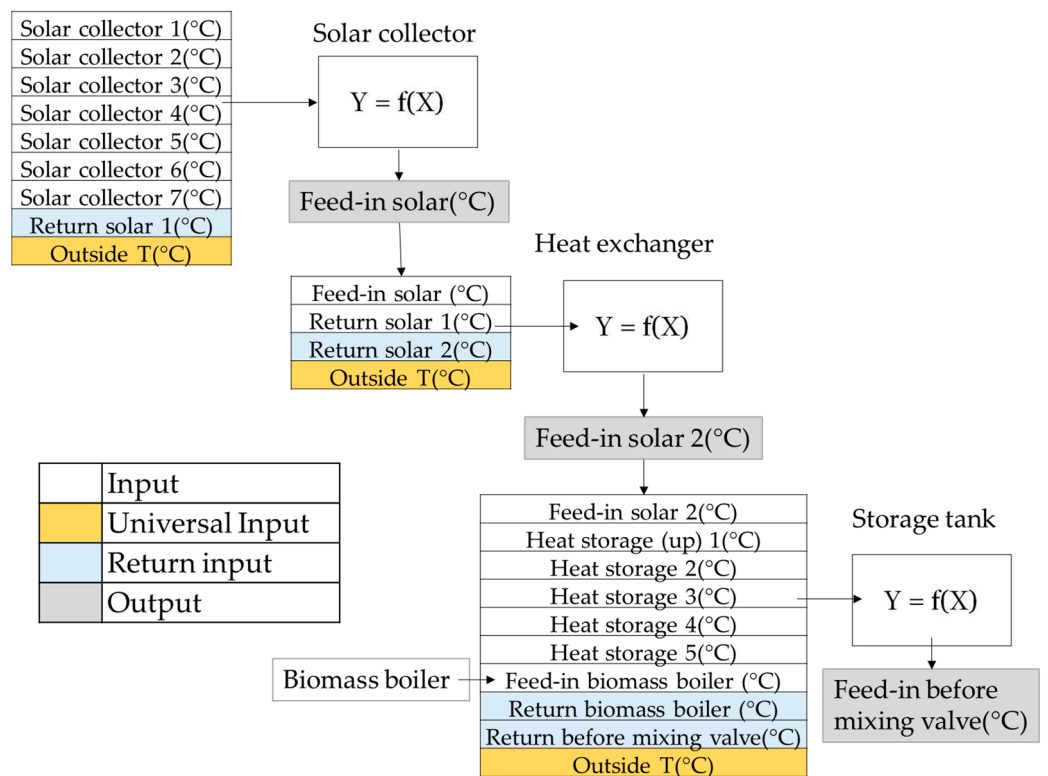


Figure 1. Detailed representation of the interconnection of the different components of the system, along with the respective inputs and outputs.

The physical interconnection of the different components of the TES system under investigation along with respective input-output information are depicted in detail in Figure 2.

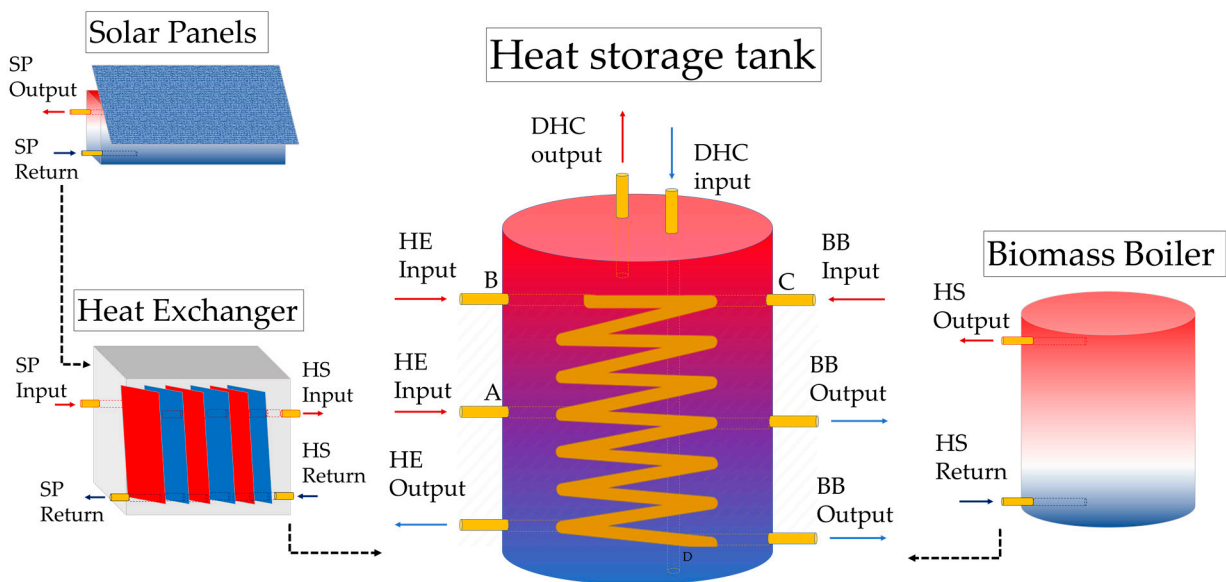
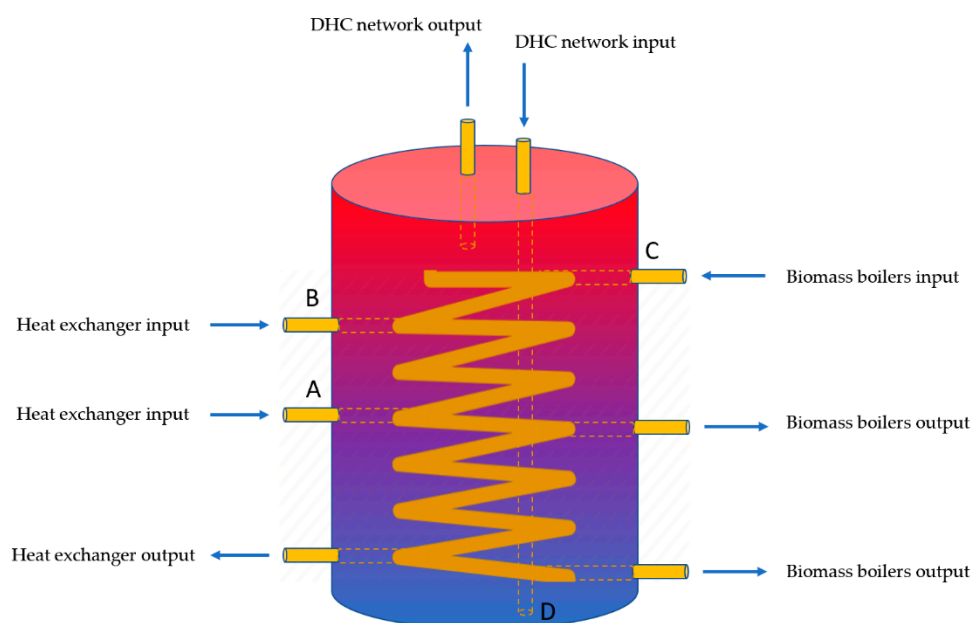


Figure 2. Schematic of the physical connection between the solar panels (SP), heat exchanger (HE), biomass boiler (BB) and the heat storage tank (HS).

The tank itself is equipped with an immersed heat exchanger (IHX) coil that contains a coolant, deionized (DI) water. The water within the coil absorbs heat from the different heating sources and rejects the heat to the storage tank water. To avoid contamination,

the DI water remains decoupled from the thermal storage medium. Specifically, there are three points of input for the hot water in the storage tank. Two of them come from the heat exchanger, which is connected with the solar panels, and one comes from the biomass heater. The inputs that originate from the heat exchanger/solar panels system enter the tank from the middle (point A) and the top (point B) of the tank. The input that originates from the biomass boiler enters only in the top of the tank (point C). Alternatively, the biomass boiler can supply heat directly to the network in case this is necessary. Hot water enters, and the heat diffuses in the tank via conduction. The water from the top section of the tank, which is the hottest, is fed and circulated into the network, and the water that returns from the network, goes directly to the bottom of the tank, serving as an extra input (point D). An in-depth schematic of the cylindrical thermal energy storage tank and the connected pipes is shown in Figure 3.



**Figure 3.** Schematic of the colloidal pipes and the basic structure of the heat storage tank.

The increased complexity in the structure of the storage tank poses significant challenges that deteriorate the accurate modelling of the temperature dynamics of the TES. A data-driven modelling approach is proposed in this paper to (i) tackle the aforementioned barriers avoiding all the unnecessary simplifications that are typically employed and (ii) enable semi-real time modelling of the induced complexity without affecting the predictive ability of the model.

## 2.2. Data Acquisition and Characteristics

A database that contained 351,148 inputs from temperature sensors starting from 5 May 2014 08:45, until 10 October 2017 02:50, collecting data every 5 min was acquired and used in the present study. Each data entry consists of 21 features that contain temperature information from various locations on the energy storage facility, both internal (from the piping system), as well as external (e.g., the outside temperature). The total of 7,374,108 entries is considered as a satisfactory amount of data for evaluating the performance of data-driven models.

In order to model each component properly, we used three different configurations of the dataset based on the inputs that are required by each component, and the output it produces. The group of solar panels constitutes the first component (A) that receives 8 input temperatures and produces one output. The heat exchanger is the second component (B) with 3 temperatures as inputs and one output. Finally, the heat storage tank (component C) receives 9 input temperatures and produces the final output temperature of the water

feeding the DH network. The external temperature serves as a universal input for all the components. Table 1 cites all the features and their role in each component as well as the number of training samples multiplied by the features for each component.

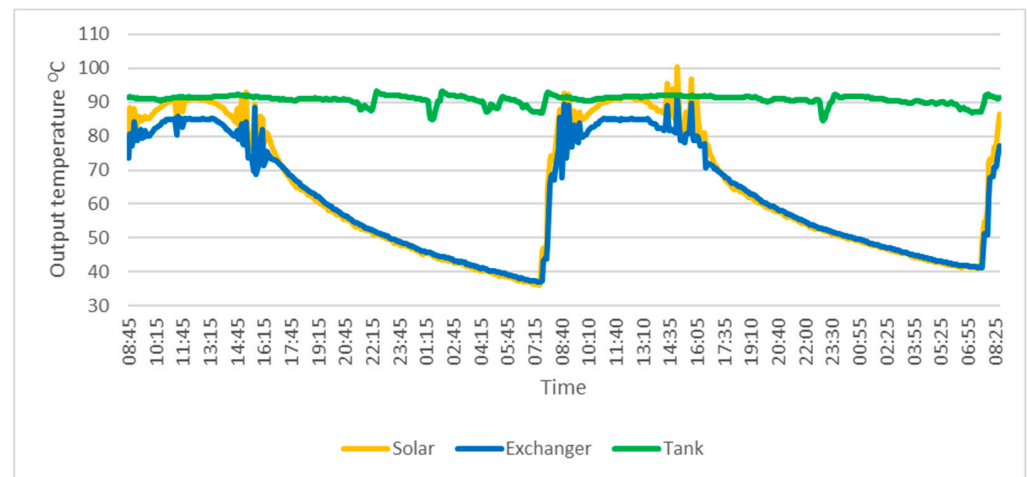
**Table 1.** The features of the dataset and how they are used within each component.

Feature Name	Solar Panels	Heat Exchanger	Heat Storage Tank
T1 Solar collector field 1 (°C)	Input	-	-
T2 Solar collector field 2 (°C)	Input	-	-
T3 Solar collector field 3 (°C)	Input	-	-
T4 Solar collector field 4 (°C)	Input	-	-
T5 Solar collector field 5 (°C)	Input	-	-
T6 Solar collector field 6 (°C)	Input	-	-
T7 Solar collector field 7 (°C)	Input	-	-
T8 feed-in Solar prim. (°C)	Output	Input	-
T9 return Solar prim. (°C)	Return Input	Input	-
T10 feed-in Solar sec. (°C)	-	Output	Input
T11 return Solar sec. (°C)	-	Return Input	-
T12 Temp. 1 heat storage (up) (°C)	-	-	Input
T13 Temp. 2 heat storage (°C)	-	-	Input
T14 Temp. 3 heat storage (°C)	-	-	Input
T15 Temp. 4 heat storage (°C)	-	-	Input
T16 Temp. 5 heat storage (°C)	-	-	Input
T17 feed-in biomass boiler (°C)	-	-	Input
T18 return biomass boiler (°C)	-	-	Input
T19 feed-in before mixing valve (°C)	-	-	Output
T20 return before mixing valve (°C)	-	-	Return Input
T21 outside temp. (°C)	Universal Input		
Total number of training samples multiplied by the features of each component	3,511,480	1,755,740	3,862,628

The output temperatures of each component within a duration of two days, specifically starting from 5 May 2014 08:45 until 7 May 2014 08:25, are shown in Figure 4. The output temperatures of each module fluctuate during the day and are affected by seasonal external conditions such as temperature and cloud coverage, as expected. Since the heat exchanger is connected only with the solar panels, their daily output fluctuation looks similar, however a comparison of the two shows a difference in temperature that is explained by the heat loss from the piping system. It is also observed that the storage tank's output fluctuation differs from the others due to the fact that the heat storage tank's energy level and water temperature is heavily dependent on the operation of the biomass boilers and the input they provide.

### 2.3. Preprocessing Phase

Data collection and organization: Data were collected from 21 temperature sensors, located on various locations all over the heat storage tank facility. Data entries were collected in 5-min intervals, over a period of 3.5 years, leading to the generation of the dataset. The dataset was segmented into 3 feature subgroups ( $D_A$ ,  $D_B$ , and  $D_C$ ) each one containing only the relevant features that serve as inputs/output for each component (as shown in Table 1). This way, each model was built by the right features that have actual importance and physical meaning for each particular model.



**Figure 4.** Output temperature of components A, B and C for 24 h.

To model the temperature dynamics of the TES system, a standardized procedure was followed. The proposed data-driven TES modelling methodology includes the following processing steps:

**Handling of missing values:** Real life datasets are susceptible to hardware malfunctions, software errors, or various random events that lead to errors, and/or corrupted/missing values. The first step is to clear any abnormalities that appear as NaN (Not a Number) values in the dataset. In general, the missing values could either be replaced with a regression-based prediction median of the values, or the whole measurement could be removed from the dataset. In our case, since the dataset was rather complete and only a few lines of data contained NaN values, it was decided that these entries should be removed completely.

**Data split:** To validate the performance of the proposed ML models, the generated dataset was randomly split into three subgroups: the training dataset in which the models were trained and fit, the validation dataset which was used to properly train the models and avoid overfitting, and the testing dataset in which all models were tested and evaluated. An 80%–10%–10% split was selected for the training, validation and testing sets, respectively.

**Data normalization:** Feature scaling standardizes the range of all the independent variables/features of the dataset. This method is a common requirement for most ML models that typically ensures a smoother implementation of the ML algorithms and in some cases leads to improved performance. Feature normalization was applied to the training and the testing datasets so that they are centered around 0 with a standard deviation of 1. This is called standard scaling and the mathematical equation that describes it is as follows:

$$\hat{x}_{i,j} = \frac{x_{i,j} - \mu_j}{\sigma_j}, \quad (1)$$

where  $x_{i,j}$  and  $\hat{x}_{i,j}$  denote the value of feature  $j$  for the initial and normalized data entry  $i$ , respectively, with the mean value of feature  $j$  calculated by:

$$\mu_j = \frac{1}{N} \sum_{i=1}^N (x_{i,j}), \quad (2)$$

and standard deviation of feature  $j$  as:

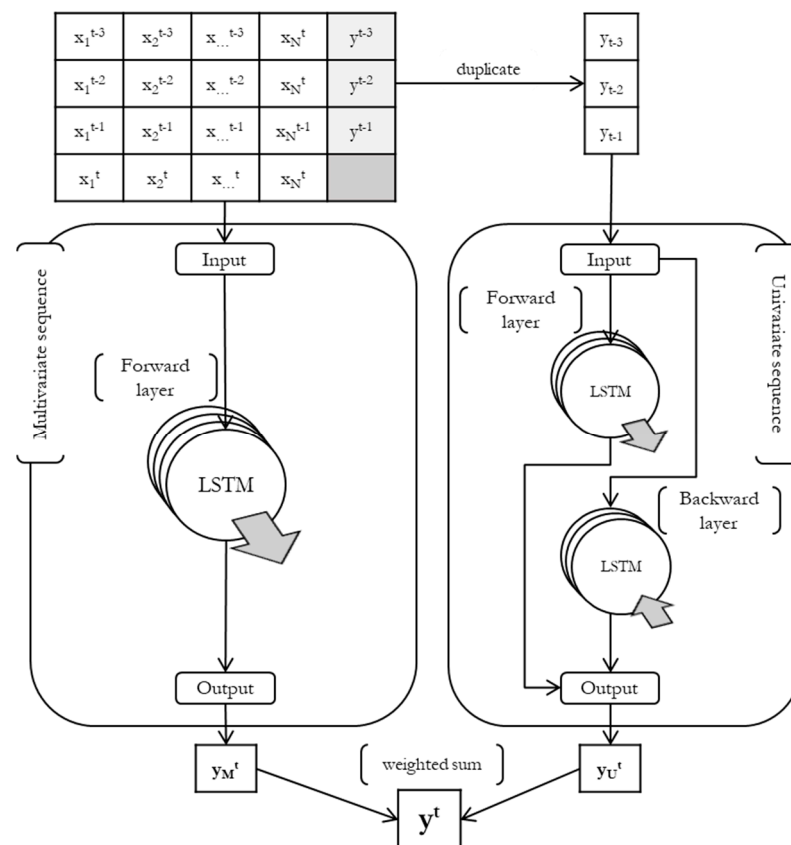
$$\sigma_j = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_{i,j} - \mu_j)^2} \quad (3)$$

#### 2.4. Proposed LSTM Architecture

Recurrent Neural Networks (RNN) is a special category of ANN, which are built in a recurrent fashion, aiming at tackling problems with sequential features, such as timeseries



forecasting [58,59]. Simple RNNs offer improvements on specific applications, however, they face the problem of vanishing gradients [60]. This problem was solved by Long Short-Term Memory (LSTM) networks with the introduction of memory gates within a recurrent architecture [61]. In the present study, a hybrid bimodal architecture of LSTM neural networks is proposed (H2M-LSTM) in order to fit the specific challenges that arise from the problem of modelling heat storage tanks. The novelty of the proposed architecture lies in a bimodal structure where one component of the network takes multivariate, multistep input and feeds it to a regular forward-direction LSTM, and the other mode focuses on fine-tuning the predictions by learning the trends of the desired feature by feeding the univariate input on a bidirectional LSTM. The multivariate component focuses on learning the dependencies and relationships of the output variable with the rest of the features, and the univariate component fine-tunes the output by learning the trends that govern the storage tank's behaviour. The final target variable of the H2M-LSTM is the weighted average of each mode's predictions. For the present study, the multivariate component was built with 2 layers and 50 memory cells each, whereas the univariate component was built with 1 layer and 50 memory cells for the forward- and backward-layer. The architecture of the proposed network is shown in Figure 5.



**Figure 5.** Architecture of the proposed LSTM network.

### 2.5. Benchmarking Machine Learning Algorithms

For benchmarking purposes, an extensive set of algorithms has been selected based on their performance, and their applicability on the TES modelling problem of the present study. The applied methodologies cover most of the main algorithmic families that are currently used in machine learning.

Several algorithms were used from the family of linear regression (such as Ordinary Least Squares (OLS) [62] and Least Absolute Shrinkage and Selection Operator Lasso (LASSO) [63,64]) that is one of the most fundamental methods used in machine learning. LASSO conducts variable selection alongside with L1 regularization for the enhancement

of the prediction accuracy. Additionally, another two algorithms were employed from the same category: (i) Bayesian Ridge (BR) [65] which is a linear regression approach where Bayesian inference takes the place of statistical analysis and (ii) Elastic Net (EN) [66] that linearly combines the regularization penalties of both the LASSO and the Bayesian Ridge methods. Least-Angle Regression (LARS) [67], which is strongly preferred when the data for the regression problem are high-dimensional, was also explored along with Stochastic Gradient Descent (SGD) [68] which is best known for the optimization of differentiable objective functions in an iterative way. Both LARS and SGD are strong representative algorithms related to linear regression.

A different approach is followed in Decision trees (DT) [69,70] where there is gradual split and organization of a dataset in reduced homogeneous datasets, following a top down structure of a tree with the root being at the top, whereas branches and leaf nodes are progressively generated towards the bottom. DT offer the explanation, however, that they are prone to overfitting, something that can be dealt with the application of ensemble methods. Random Forest (RF) [71] is the most famous algorithm of the ensemble methods family where, in principal, the algorithms construct a linear combination of base learners, in this case DT, for improving their predictive ability. Gradient Boosting (GB) [72] produces a prediction model as an ensemble of several weak prediction models in a stage-wise fashion. AdaBoost [73] is an estimator that fits a weak regressor, then fits more duplicates of the same regressor upon the original dataset and finally a boosted model derives from the weighted sum of these weak learners, constantly adjusted based on the error of a prediction. Last algorithm of the ensemble methods, Bootstrap Aggregating, or Bagging [74], builds black-box estimators that train on multiple random subsets of the initial dataset, and then produces a final prediction based on the combination of their individual predictions.

One of the most important categories of ML algorithms is Support Vector Machines (SVM). They were initially developed based on the statistical learning theory [75] to classify data instances with the construction of a linear separating hyperplane. One of their greatest attributes is that they can utilize a selection of various kernels in order to transform an initial feature space into a higher dimensional space, consequently improving their performance in particular tasks. SVM perform consistently in different tanks, due to the fact that they can resolve the overfitting issue in high dimensional spaces with global optimization [76,77]. In the present study, three kernels were used as different approaches for our support vector regression (SVR) implementation, namely a linear, a polynomial and radial basis function kernel.

Artificial Neural Networks (ANN) [78] is the most popular family of ML algorithms due to its consistently high performance in a variety of regression and classification tasks in the recent years. Some of the most famous learning algorithms include the perceptron [79], multilayer perceptron [80], back-propagation [81], resilient backpropagation [82], radial basis function networks [83], autoencoders [84], adaptive-neuro fuzzy inference systems [85] and much more. New types of neural networks have emerged such as recurrent neural networks [86] and convolutional neural networks [87], as part of the larger deep learning group [88]. For this study, a Multi-Layer Perceptron (MLP) was used with an architecture of 4 hidden layers with a “384, 384, 256, 192” node configuration, Rectified Linear Unit (ReLU) [89] as our activation function for the hidden layers, Adam optimization [90] and early stopping with a 10 epoch patience to avoid overfitting. The hyperparameters used in each approach were carefully selected with a grid search method, with which various parameters are selected, and a range of values is tested.

### *2.6. Modelling Approaches: Component-Specific and Cascading*

Energy component-specific modelling: Each component of the heat storage system was modelled using all the aforementioned machine learning algorithms. The goal of this step is to benchmark the algorithms which were presented in Section 3.3 and then compare them with the proposed hybrid bimodal LSTM that was developed for the particular application and presented in Section 3.2. For the individual components’ modelling, the solar panels, heat exchanger and storage tank have all been modelled using the same hyperparameter values

for all individual models. In this validation setup, input-output data collected from sensors installed directly on the energy components were used for the training of the ML algorithms.

**Cascading TES modelling:** Additionally to proposing a tailor-made learning architecture that fits the addressed problem and its challenges, this study also proposes a holistic modelling methodology that incorporates all the TES components and their processes within the heat storage tank facility. In order to accomplish that, a new modelling methodology is proposed to address the need for a realistic representation of the components considering the dependencies between their inputs and outputs and the hierarchical order in which they are connected. Therefore, a cascading architecture was set up where each model provides its output values to be used as inputs by the adjacent one, until the final output of the storage tank is generated. Specifically, the solar panel's output is used as input for the exchanger, where its output, along with the output of the boilers, are used as inputs for the heat storage tank. At the same time, the return from the network is also used as input for the heat storage tank, whose return is used as input for the exchanger, whose return is used as input for the solar panels.

The cascading architecture's components are trained consequentially, starting from the solar panel's model, followed by the exchanger, and finally having the heat storage tank to generate the values of the temperature of the water that feeds the network. The models are arbitrarily presented in Figure 1 by  $y = f(x)$  boxes. All the ML algorithms are compared based on their suitability in being the core model of the cascading architecture. The performance of the cascading architecture in comparison to the individual components' modelling will offer a deeper insight into whether a unified approach to a multi-component system is applicable in real life. The results of the cascading modelling approach are presented in Section 3.2.

A simplified flowchart that summarizes and visually presents the process that has been followed in our study is shown in Figure 6.

The dataset was segmented into 3 feature subgroups ( $D_A$ ,  $D_B$ , and  $D_C$ ) each one containing only the relevant features that serve as inputs/output for each component (as shown in Table 1). This way, each model was built by the right features that have actual importance and physical meaning for each component. For the validation and evaluation of the models, each subgroup was split into two portions. The training portion, denoted as  $tr$ , which is a random 80% of the initial subgroup, is the amount of data upon which the algorithms are fitted, and the models are built. The testing portion, denoted as  $te$ , which is the remaining random 20%, is the amount of data that is kept hidden during the training process and is used for the validation of the models' performance. The following six datasets were generated as the output of this step:  $D_{A\_tr}$ ,  $D_{A\_te}$ ,  $D_{B\_tr}$ ,  $D_{B\_te}$ ,  $D_{C\_tr}$  and  $D_{C\_te}$ . The evaluation of each model's performance for the present study was based on three common performance metrics, the mean absolute error (MAE), mean squared error (MSE) and root mean squared error (RMSE), that are explained below.

MAE is the average of the absolute errors of the estimated and the real values. In this average, the individual differences are all equally weighted. Mathematically it is represented by the following equation:

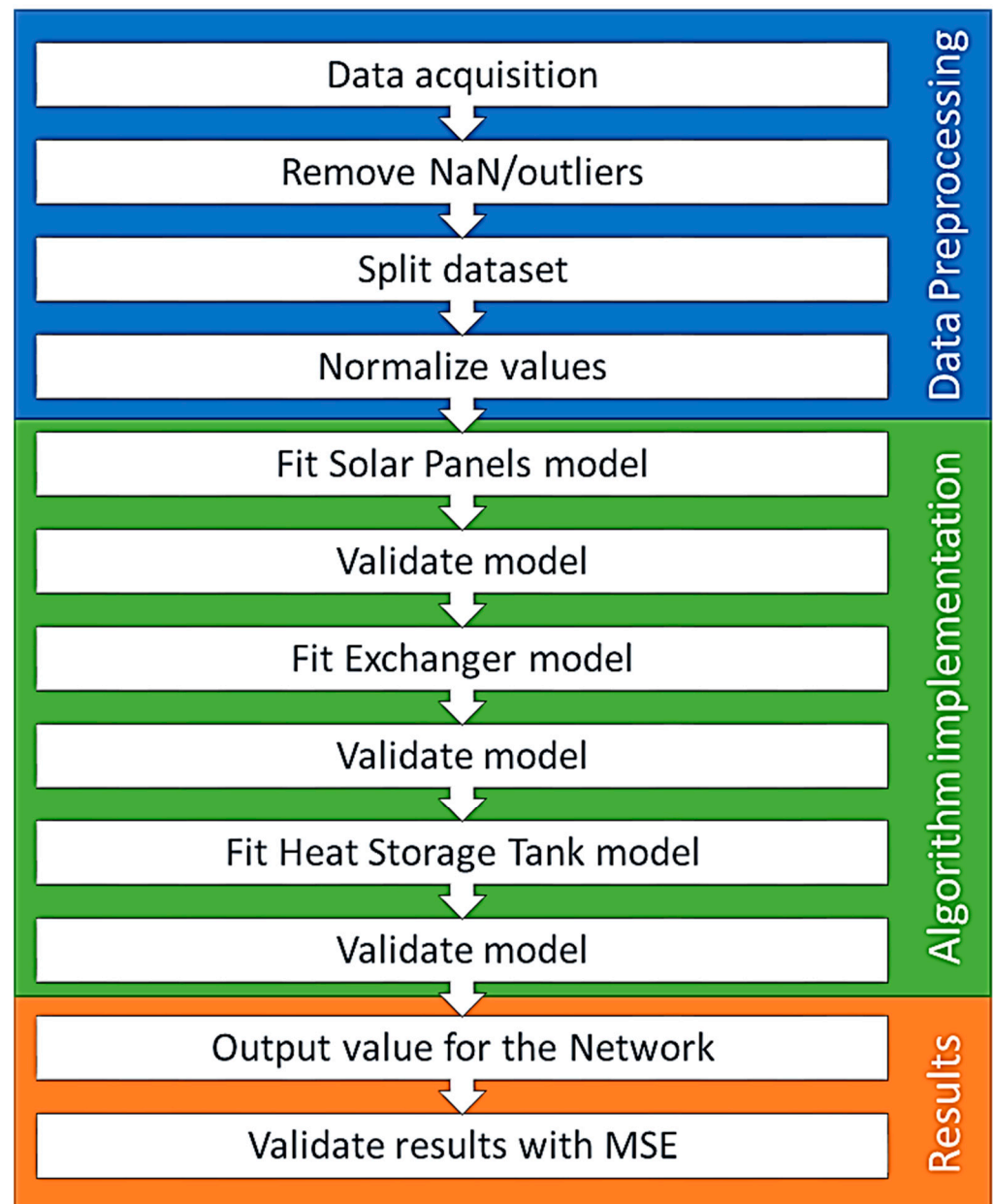
$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - x_i| \quad (4)$$

MSE measures the average squared difference between the estimated values and the real values and it is always non-negative. MSE's usefulness lies in the fact that large errors have bigger consequences than equivalent small ones when penalized. The mathematical equation is the following:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - x_i)^2, \quad (5)$$

RMSE is the squared root of the average of squared errors. The reason that RMSE was used alongside MSE was because it is particularly sensitive to outliers since large errors have a disproportionately large effect. The mathematical equation is similar to the MSE:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - x_i)^2} \quad (6)$$



**Figure 6.** Flowchart of the methodology followed in this study.

### 3. Results

#### 3.1. Experimental Design

The performance of the proposed architecture and the competing methodologies is thoroughly presented in this chapter. TES modelling of individual components was initially performed to evaluate the predictive capacity of the proposed LSTM method compared to benchmarks. At a second phase, the cascading modelling architecture was implemented in which the outputs of a component were given as inputs to the subsequent system

component. The latter modelling approach takes into account the intercorrelation of the different energy components providing a holistic solution to TES modeling.

Modelling of individual components: For each one of the three components A, B and C, the proposed LSTM approach and a variety of competing ML models (as presented in Sections 3.2 and 3.3) were trained on  $D_{i\_tr}$ ,  $i = A, B, C$ . The resulted trained ML models were tested on the subsets  $D_{i\_te}$ ,  $i = A, B, C$ . The validation of each ML model was conducted with the application of three performance metrics, MSE, MAE and RMSE and a comparative analysis was performed.

TES modelling using the proposed cascading architecture: A cascading architecture was set up in such a way that the predicted outcomes of component A were used as input for component B, and subsequently the predicted outcomes of component B were used as input for component C. The previous step was repeated using the proposed LSTM approach and the competing ML models (as presented in Sections 3.2 and 3.3). At each iteration, each one of the ML models was applied in all three components. The validation of each ML model used within the cascading architecture was conducted with the application of the same three performance metrics, MSE, MAE and RMSE. A comparative analysis was performed, and the best method was chosen by taking into consideration the accuracy of each method.

### 3.2. Performance of The Individual Modelling Approaches for Each Component

The results (performance metrics) for all the ML algorithms are presented in the following tables, each one for the particular component that it was used for. Table 2, Figures 7 and 8 show the achieved metric values of the ML models on the thermal modeling of component A (solar panels), Table 3, Figures 9 and 10 for component B (heat exchanger) and Table 4, Figures 11 and 12 for component C (heat storage tank). All programming was done in Python 3.4, mostly based on libraries such as NumPy [91] for mathematical operations, Pandas [92] for data structures handling and Matplotlib [93] for visualizations. All models were trained either with Scikit-Learn [94] on an i7-6950X CPU or with Keras [95] and Tensorflow [96] on a local computer equipped with a Titan 1080Ti GPU, depending on the potential parallelization of operations.

**Table 2.** Performance metrics for the solar panels.

Method	MAE	MSE	RMSE
Decision Tree	0.073	0.029	0.172
Random Forest	0.067	0.015	0.124
AdaBoost	0.353	0.195	0.441
Bagged Trees	0.067	0.015	0.124
Boosted Trees	0.168	0.063	0.252
Linear Regression	0.259	0.125	0.354
Bayesian Ridge	0.259	0.125	0.354
Stochastic Gradient	0.259	0.125	0.354
Lasso Regression	0.820	0.997	0.999
Elastic Net	0.623	0.558	0.747
Least Angle	0.259	0.125	0.354
SVR linear	0.257	0.128	0.358
SVR poly	0.419	0.305	0.552
SVR rbf	0.154	0.058	0.242
ANN MLP	0.096	0.024	0.154
H2M-LSTM	0.059	0.012	0.110

A small value for the MSE points out that the model successfully describes the real-life temperature dynamics of the storage tank components.

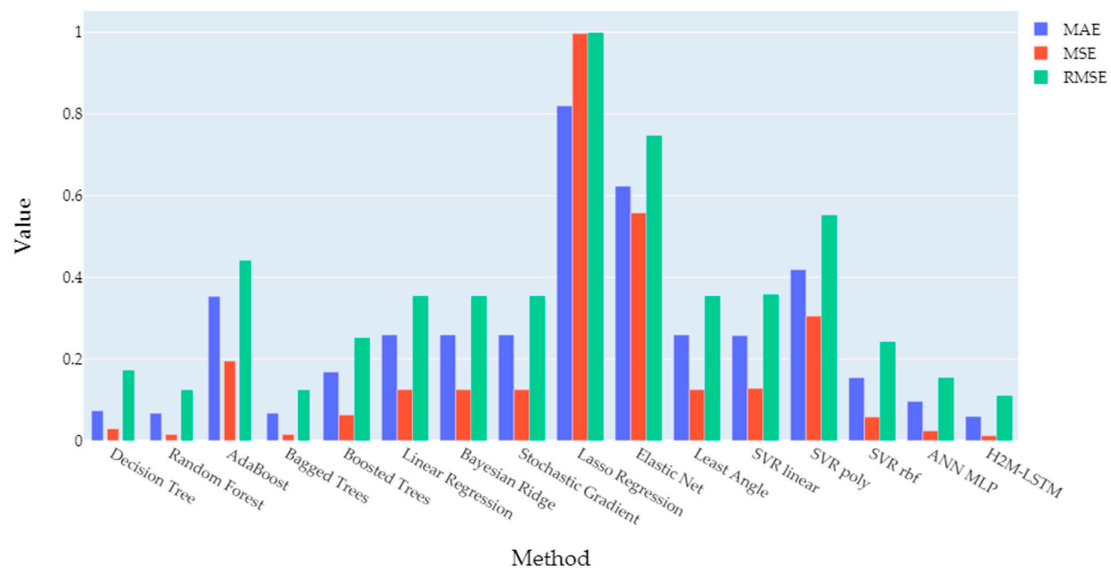


Figure 7. Barplot comparison of the performance metrics for the solar panels.

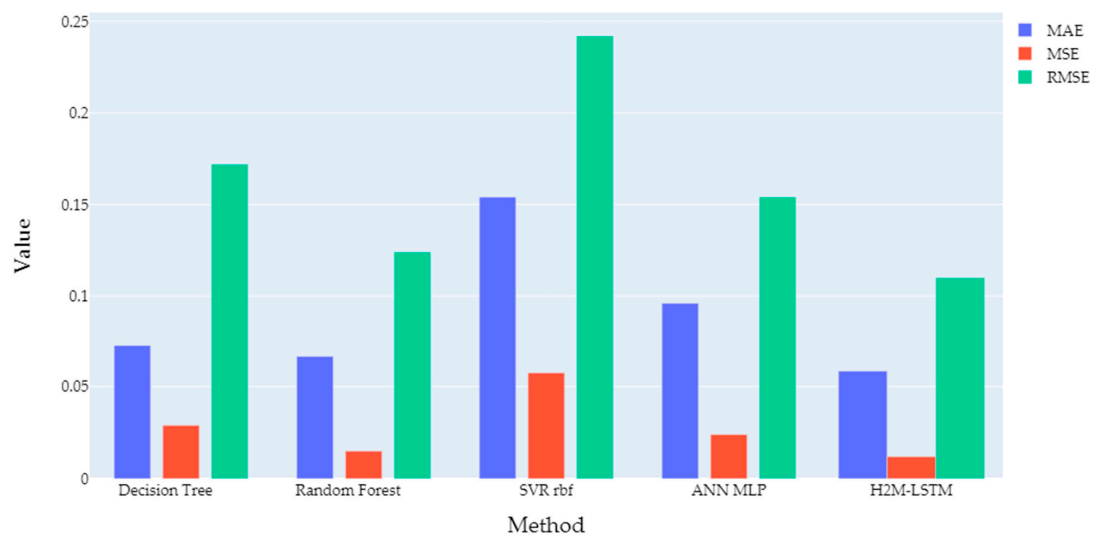


Figure 8. Barplot comparison of the best performance metrics for the solar panels.

Table 3. Performance metrics for the heat exchanger.

Method	MAE	MSE	RMSE
Decision Tree	0.026	0.005	0.069
Random Forest	0.023	0.003	0.055
AdaBoost	0.172	0.048	0.219
Bagged Trees	0.023	0.003	0.055
Boosted Trees	0.057	0.010	0.099
Linear Regression	0.081	0.019	0.136
Bayesian Ridge	0.081	0.019	0.136
Stochastic Gradient	0.081	0.019	0.136
Lasso Regression	0.830	0.998	0.999
Elastic Net	0.525	0.396	0.629
Least Angle	0.081	0.019	0.136
SVR linear	0.082	0.019	0.139
SVR poly	0.361	0.718	0.847
SVR rbf	0.059	0.008	0.091
ANN MLP	0.033	0.004	0.060
H2M-LSTM	0.020	0.002	0.045

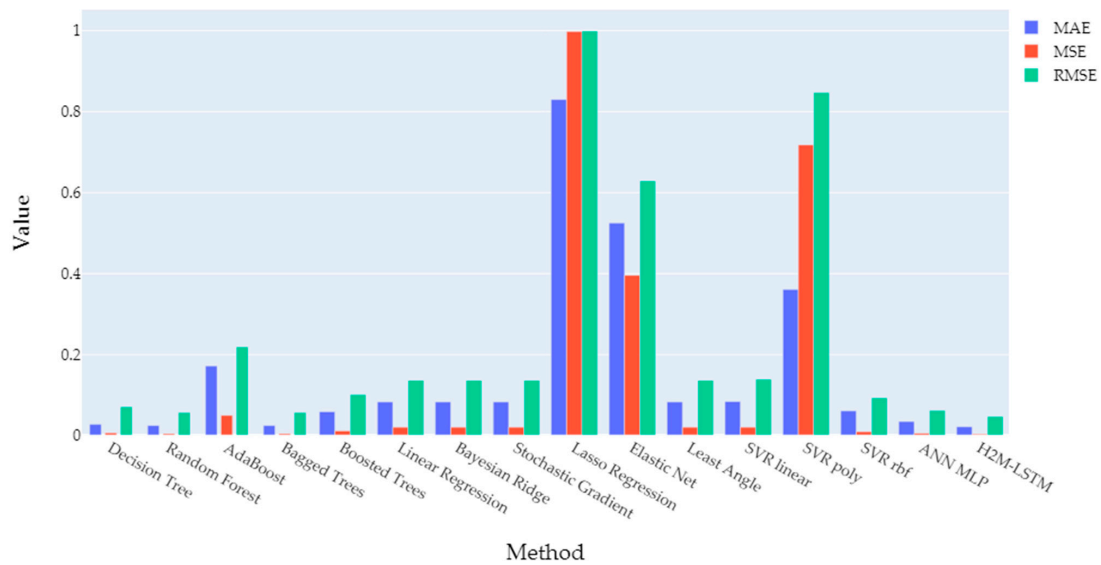


Figure 9. Barplot comparison of the performance metrics for the heat exchanger.

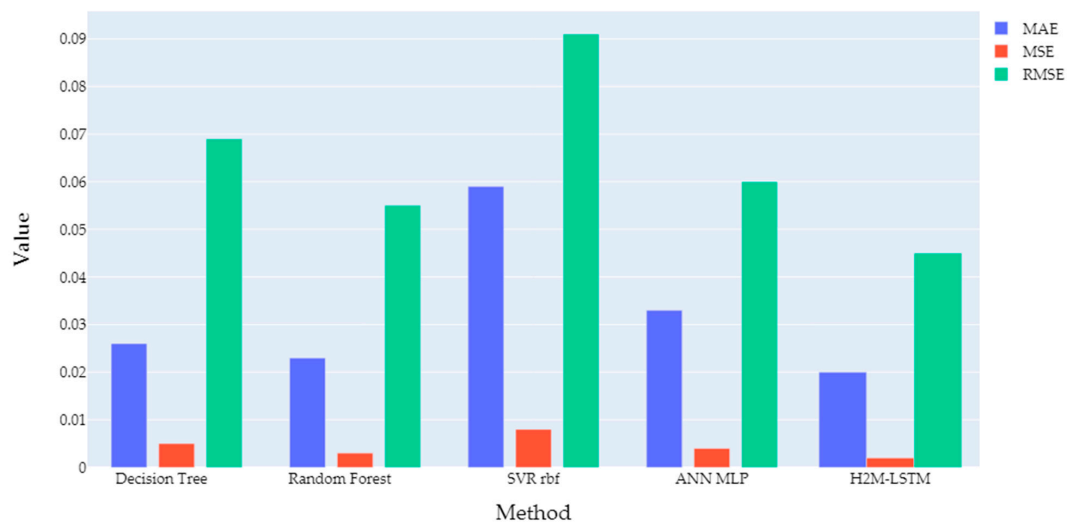


Figure 10. Barplot comparison of the best performance metrics for the heat exchanger.

Table 4. Performance metrics for the heat storage tank.

Method	MAE	MSE	RMSE
Decision Tree	0.042	0.009	0.094
Random Forest	0.033	0.005	0.071
AdaBoost	0.192	0.060	0.244
Bagged Trees	0.033	0.005	0.072
Boosted Trees	0.074	0.014	0.120
Linear Regression	0.145	0.052	0.229
Bayesian Ridge	0.145	0.052	0.229
Stochastic Gradient	0.144	0.052	0.229
Lasso Regression	0.818	1.001	1.000
Elastic Net	0.584	0.513	0.716
Least Angle	0.145	0.052	0.229
SVR linear	0.129	0.057	0.238
SVR poly	0.252	0.154	0.392
SVR rbf	0.055	0.008	0.089
ANN MLP	0.039	0.005	0.071
H2M-LSTM	0.028	0.003	0.055

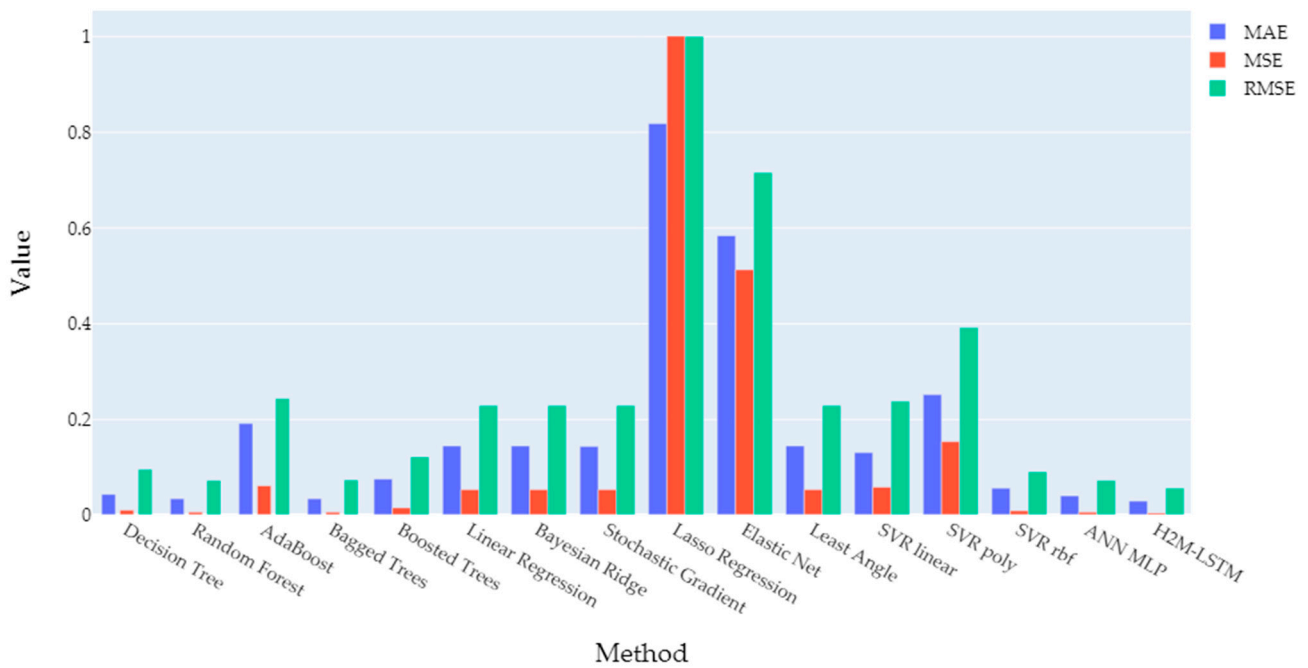


Figure 11. Barplot comparison of the performance metrics for the heat storage tank.

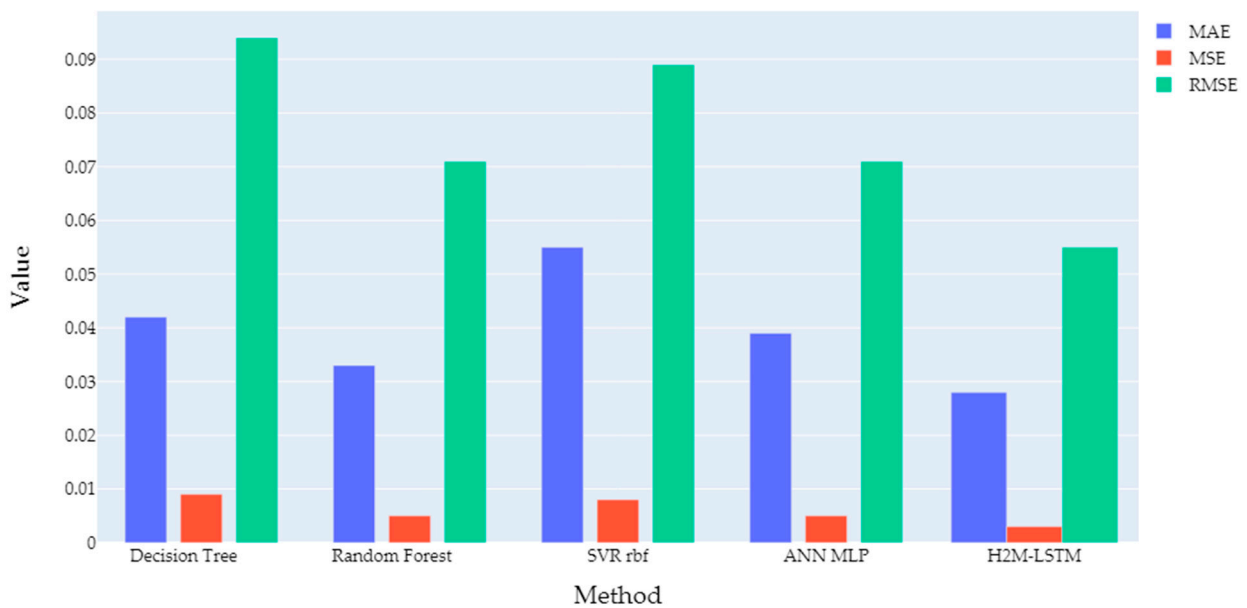


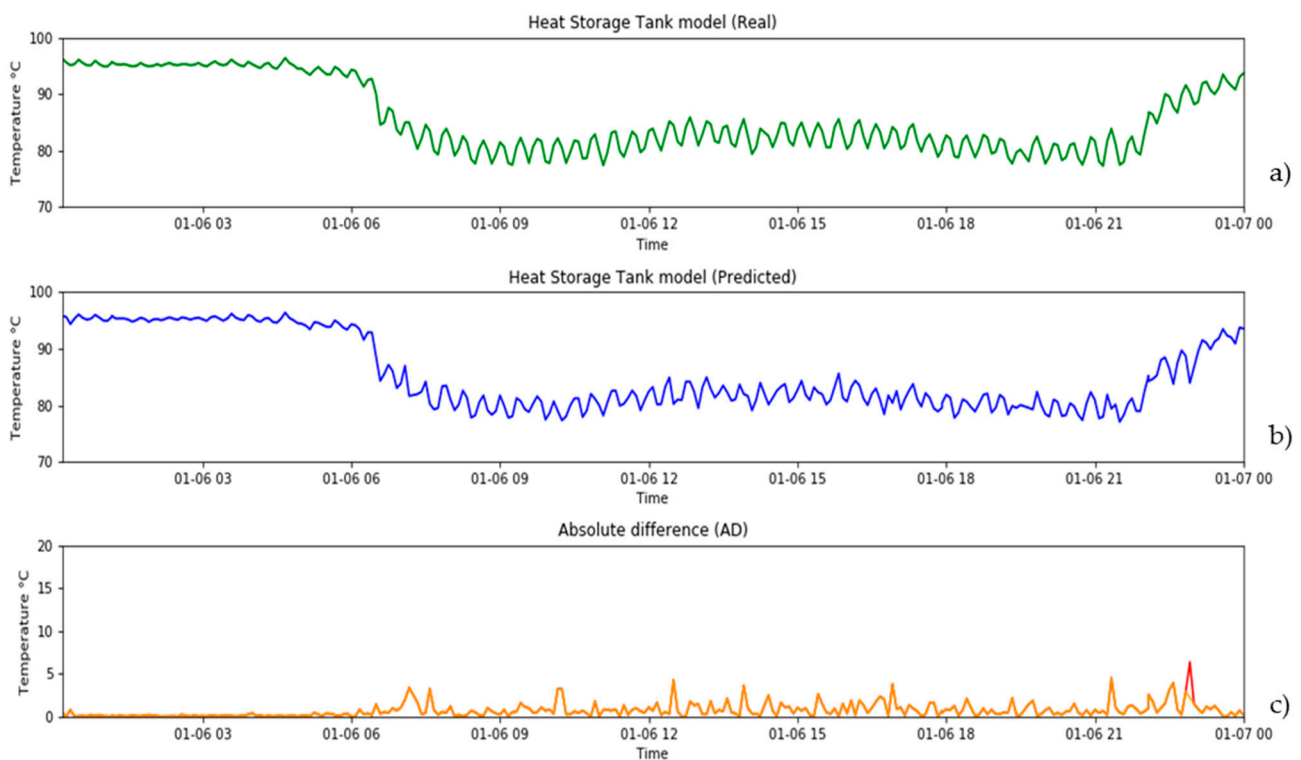
Figure 12. Barplot comparison of the best performance metrics for the heat storage tank.

Linear regression and its variants, along with SVR with linear kernel and the boosting methods, achieved similarly poor and inconsistent performances in the range of 0.015–0.124 for MSE, due to their linear nature. Elastic-net and SVR with polynomial kernel scored MSE higher than 0.154, whereas lasso regression systematically produced the worst MSE of approximately 1 for all components. On the other hand, RF, BT and ANN produced accurate predictions, specifically RF achieved 0.015 MSE for the solar panel, RF and BT both scored 0.003 on the heat exchanger, and ANN together with RF both achieved 0.005 MSE on the heat storage tank. While each of the ML algorithms used in the study offer discrete benefits, none was able to surpass H2M-LSTM which was built specifically for the particular application. H2M-LSTM managed to outperform all other algorithms across all models and on all performance metrics by scoring 0.012 MSE for the solar panels, 0.002 for the exchanger and 0.003 for the heat storage tank. This consistency in performance demonstrates the



effectiveness of a tailor-designed architecture for a specific problem, as well as validates the much-desired robustness any method requires, especially when applied in different components of a system.

A visual representation of the heat tank's performance for the H2M-LSTM implementation is shown in Figure 13. The modeling performance is demonstrated on Figure 6, for a randomly selected day, namely the 6th of January 2015, by plotting the real values (a), the predicted ones (b), along with their absolute difference (AD) (c). A limit of 5 °C is set in order to visualize the magnitude of error. AD values below 5 °C are visualized in orange, whereas red represents AD values that exceeded the limit. It is apparent that, especially during times of low fluctuations of temperature, the predicted values are very close to the real ones.



**Figure 13.** Real (a) vs. predicted (b) output of the individual heat storage model for the 6 January 2015, along with their absolute difference (c) (AD).

### 3.3. Performance of the Cascading Architecture

All algorithmic methods were also applied to the cascading model with the aim to model the whole system as a unified system. Each method was implemented and evaluated again in order to determine whether the cascading model maintains the performance of the individual models. A detailed table with the results of all the metrics (MSE, MAE and RMSE) for all the tested ML algorithms is presented below. The results are gathered and presented in Table 5, Figures 14 and 15:

**Table 5.** Performance metrics for the machine learning algorithms used within the cascading architecture.

Method	MAE	MSE	RMSE
Decision Tree	0.044	0.010	0.099
Random Forest	0.034	0.005	0.073
AdaBoost	0.180	0.053	0.231
Bagged Trees	0.034	0.006	0.074
Boosted Trees	0.075	0.015	0.121
Linear Regression	0.145	0.052	0.229

Table 5. Cont.

Method	MAE	MSE	RMSE
Bayesian Ridge	0.145	0.052	0.229
Stochastic Gradient	0.145	0.053	0.229
Lasso Regression	0.818	1.001	1.000
Elastic Net	0.584	0.513	0.716
Least Angle	0.145	0.052	0.229
SVR linear	0.129	0.057	0.239
SVR poly	0.269	0.209	0.457
SVR rbf	0.055	0.008	0.089
ANN MLP	0.037	0.005	0.073
H2M-LSTM	0.029	0.003	0.055

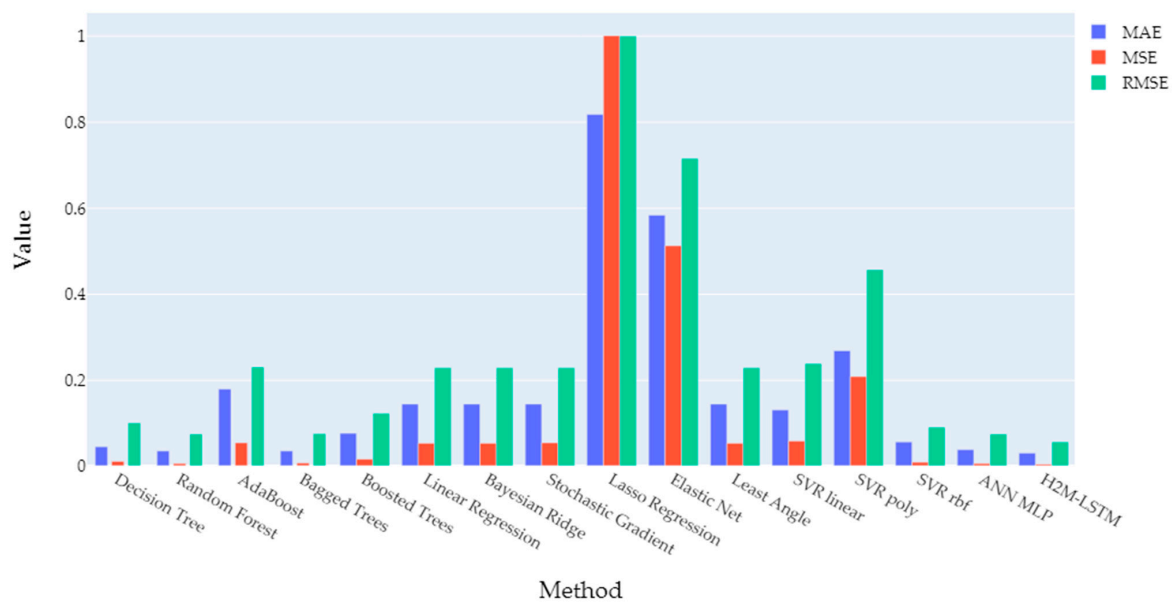


Figure 14. Barplot comparison of the performance metrics for the cascading architecture.

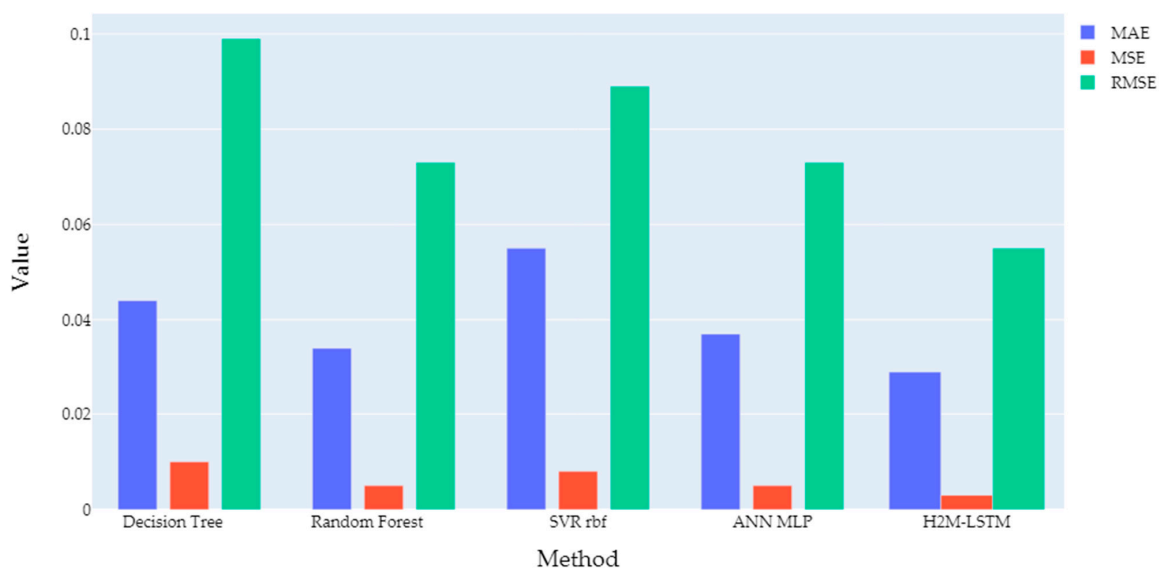
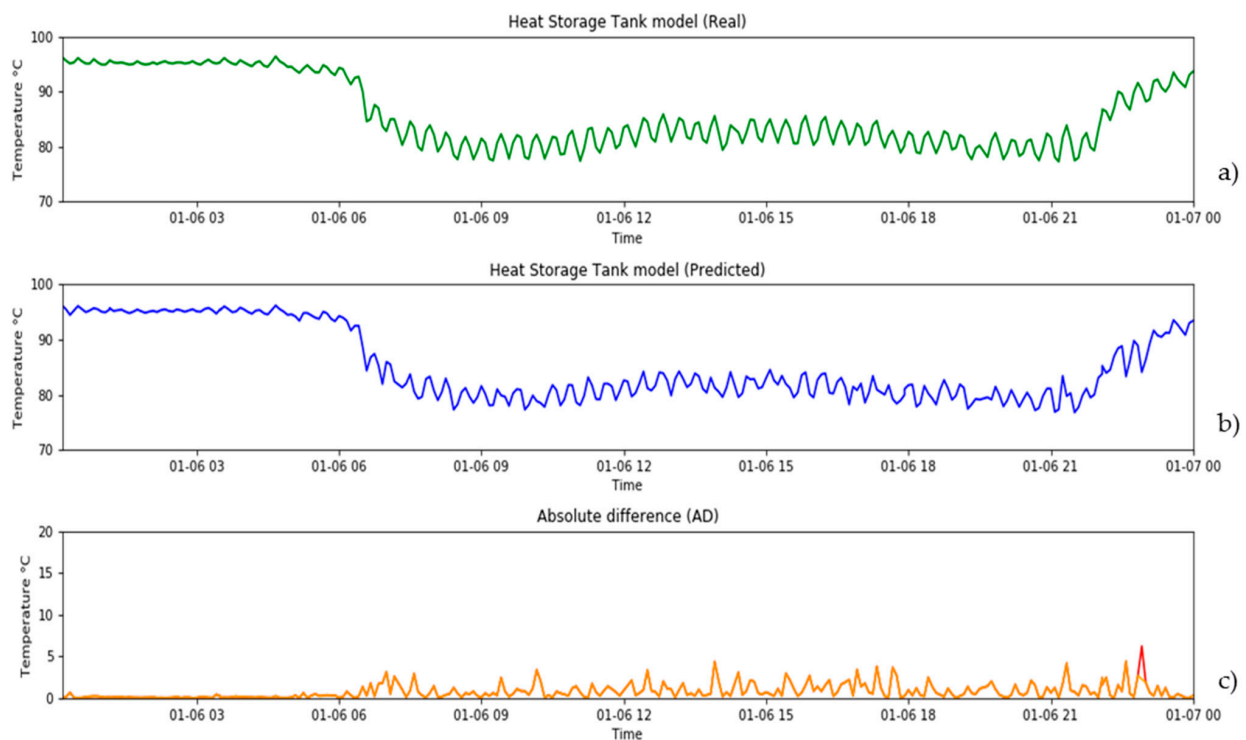


Figure 15. Barplot comparison of the best performance metrics for the cascading architecture.

The proposed hybrid bimodal LSTM architecture (H2M-LSTM) exhibits robust performance and consistent behaviour in the cascading modelling approach as well, achieving the

lowest values in all performance metrics. It is noteworthy to mention that its performance is almost identical to the performance of the individual modelling of the heat storage tank. In detail, H2M-LSTM achieved 0.029 MAE, 0.003 MSE and 0.055 RMSE, surpassing the RF in MAE by 0.005 and the RF and the ANN in MSE and RMSE by 0.002 and 0.018, respectively. As it is observed from, DT, BT, and SVR(rbf) achieved moderate performances which are within the same order of magnitude of the RF and ANN, but still less accurate than the H2M-LSTM. The comparative analysis between the individual component modelling of the heat tank and the cascading model resulted in differences less than  $10^{-4}$  for the MSE and MAE that were therefore considered as insignificant. This small observed deviation demonstrates that the proposed cascading model works with almost the same accuracy as with the model that was individually trained with the real input-output tank data.

The visual representation of the heat storage tank model's output values, under the cascading implementation with the hybrid bimodal LSTM model is presented in Figure 16, for the 6 January 2015. The real temperature output is shown in green (a), the predicted in blue (b) and the absolute difference (AD) in orange, unless it surpasses the 5 °C limit where it turns red (c).



**Figure 16.** Real (a) vs. predicted (b) output of the heat storage model within the cascading architecture for the 6 January 2015, along with their absolute difference (c) (AD).

#### 4. Discussion

Heat storage tanks, which have gradually become key components of today's DH systems, are extremely complex systems. The accurate modelling of a heat storage tank's operation can be used as a tool that will assist an operator to evaluate different scenarios of sudden changes in energy demand from the network. The present study tackled the problem of thermal energy storage modelling with a data-driven approach and a cascading methodology for hierarchical modelling. A proposed hybrid bimodal LSTM architecture (H2M-LSTM) was built for the particular problem for the sole purpose of utilizing both the plethora of sensors in the system, as well as the time-dependent trend that is apparent on the output temperature of the tank. Thus, a multi-head approach was designed where one component of the architecture is a multivariate forward-directional LSTM, and the other component is a univariate bidirectional LSTM, where the outcomes of both components are

merged into a weighted average. Various well-proven ML techniques, tested and evaluated for the modelling of the temperature dynamics of the tank, were also used for benchmarking and comparison purposes. A complete energy storage framework, where each component of a facility is modelled based on its operation and its connection to the other components, was also proposed as an alternative methodology to individual component modelling and tested in this study. This proposed framework has a cascading architecture and feeds the modelling data in a hierarchical manner, leading to a seamless modelling of the whole operation as a single system. The implementation of the cascading methodology yielded satisfactory accuracy, and therefore was proved a good fit for real-life application, since it can simplify the modelling of a multi-component system in a hierarchical manner, without any compromise in the performance.

The proposed hybrid bimodal LSTM architecture manages to achieve high performance regarding solely the temperature dynamics of the TES, without taking into consideration any other variable such as heat power, heat flows, and mass flows. The performance metrics indicate that H2M-LSTM achieved consistent, high performance for all components and subsequently the cascading framework as a whole. This consistency in performance denotes a robustness of the proposed algorithm, which is much desired in real-life applications, since it signifies steady performance. An expected shortcoming of the proposed method is that it cannot explain the causes of the temperature evolution and it only models the temporal evolution of thermal dynamics. Although this approach cannot offer the insight or the accuracy of a fine-tuned analytical model, it can potentially present some advantages, namely: (i) semi-real time execution in deployment phase enabling its integration in online control systems, (ii) scalability (as it can be adapted to different systems without any significant change on the methodology and without any knowledge of the systems' characteristics), (iii) easiness to be implemented by the operators without deep knowledge of the system characteristics. Analytical models are able to reach higher confidence intervals, with limited computational time, for temperature calculation of simple components such as solar panels, heat exchangers and heat storage tanks. Even more, these physical models can explain the causes behind their results, which is a crucial advantage against data-driven methods. Approaches such as the one presented in the present study provide various new findings and deeper knowledge on the data-driven models' behavior. The data-driven model could also be integrated within a system's optimization mechanism [28] as a means to capture the current or even predicted storage capacity of the DHC network. Target for this modelling approach is to provide a fast-and-dirty implementation to any thermal system, where there is a limited amount of information regarding the details of the TES plant's operation, other than temperature measurements, by trying to achieve the best possible accuracy for a data-driven model.

## 5. Conclusions

Analytical models have already proven their performance in modelling TES plants. However, since ML approaches are being applied in more and more applications throughout the energy domain, detailed investigation on their capabilities is required to be able to identify their advantages as well as their limitations. Further examination of the ML algorithms in energy-related applications is crucial for the advancement of the domain, providing an alternative solution to the existing analytical models, that can tackle specific problems based only on data, and can be used either as standalone, or alongside analytical simulations.

Future plans include the further development of the proposed LSTM-based modelling approaches working on some of its deficiencies. Given that there is no activation function that performs well in all data problems, an extensive exploration on the suitability of various transfer functions (such as the one proposed in [97]) would be beneficial to avoid possible gradient instability issues. Training the proposed LSTM-based model for short-term and long-term predictions could be also considered as an interesting topic. The ultimate goal is the integration of data-driven TES modeling mechanisms into intelligent

control strategies that would guarantee efficient energy management and reduction in energy costs. Reliable TES modelling could facilitate the smoother transformation of current DHC networks into smarter ones that would be robust to stochastic uncertainties and adaptable to changes in the network status. Interdisciplinarity and combination of various technological advancements such as control, energy forecasting and modelling are crucial towards the development of a complete solution that could fully achieve all the challenges of the DHC sector.

**Author Contributions:** For the present study, the following contributions are credited. For conceptualization, S.M.; methodology, S.M. and A.A.; software, A.A.; validation, A.A.; formal analysis, A.A.; investigation, A.A.; resources, A.A., S.M. and E.P.; data curation, A.A.; writing—original draft preparation, A.A.; writing—review and editing, A.A., S.M., E.P. and D.B.; visualization, S.M. and A.A.; supervision, E.P. and D.B. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was supported by the European Union’s Horizon 2020 research and innovation program under Grant Agreement #696174: Innovative Technology for District Heating and Cooling (InDeal).

**Data Availability Statement:** Data availability upon request from the authors and the consortium of the InDeal project.

**Acknowledgments:** Special thanks to Marko Krajnc and Energetika Projekt, partners of the InDeal project. (<https://www.energetika-projekt.eu/> (accessed on 25 April 2020)).

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

## References

1. Reynolds, J.; Ahmad, M.W.; Rezugui, Y. Holistic modelling techniques for the operational optimisation of multi-vector energy systems. *Energy Build.* **2018**, *169*, 397–416. [[CrossRef](#)]
2. Ayele, G.T.; Haurant, P.; Laumert, B.; Lacarrière, B. An extended energy hub approach for load flow analysis of highly coupled district energy networks: Illustration with electricity and heating. *Appl. Energy* **2018**, *212*, 850–867. [[CrossRef](#)]
3. Nassif, A.B.; Azzeh, M.; Banitaan, S.; Neagu, D. Guest editorial: Special issue on predictive analytics using machine learning. *Neural Comput. Appl.* **2016**, *27*, 2153–2155. [[CrossRef](#)]
4. Bayes, T. An Essay towards Solving a Problem in the Doctrines of Chances. *Philos. Trans.* **1763**, *45*, 296–315. [[CrossRef](#)]
5. Lecun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [[CrossRef](#)]
6. Ntakolia, C.; Anagnostis, A.; Moustakidis, S.; Karcianas, N. Machine learning applied on the district heating and cooling sector: A review. *Energy Syst.* **2021**, *13*, 1–30. [[CrossRef](#)]
7. Kalogirou, S.A. Applications of artificial neural-networks for energy systems. *Appl. Energy* **2000**, *67*, 17–35. [[CrossRef](#)]
8. Afram, A.; Janabi-Sharifi, F. Theory and applications of HVAC control systems—A review of model predictive control (MPC). *Build. Environ.* **2014**, *72*, 343–355. [[CrossRef](#)]
9. Mohanraj, M.; Jayaraj, S.; Muraliedharan, C. Applications of artificial neural networks for thermal analysis of heat exchangers—A review. *Int. J. Therm. Sci.* **2015**, *90*, 150–172. [[CrossRef](#)]
10. Yildiz, B.; Bilbao, J.I.; Sproul, A.B. A review and analysis of regression and machine learning models on commercial building electricity load forecasting. *Renew. Sustain. Energy Rev.* **2017**, *73*, 1104–1122. [[CrossRef](#)]
11. Idowu, S.; Saguna, S.; Åhlund, C.; Schelén, O. Applied machine learning: Forecasting heat load in district heating system. *Energy Build.* **2016**, *133*, 478–488. [[CrossRef](#)]
12. Mena, R.; Rodríguez, F.; Castilla, M.; Arahall, M.R. A prediction model based on neural networks for the energy consumption of a bioclimatic building. *Energy Build.* **2014**, *82*, 142–155. [[CrossRef](#)]
13. Henze, G.; Schoenmann, J. Evaluation of Reinforcement Learning Control for Thermal Energy Storage Systems. *HVAC R Res.* **2003**, *9*, 259–275.
14. Yokoyama, R.; Wakui, T.; Satake, R. Prediction of energy demands using neural network with model identification by global optimization. *Energy Convers. Manag.* **2009**, *50*, 319–327. [[CrossRef](#)]
15. Luo, N.; Hong, T.; Li, H.; Jia, R.; Weng, W. Data analytics and optimization of an ice-based energy storage system for commercial buildings. *Appl. Energy* **2017**, *204*, 459–475. [[CrossRef](#)]
16. Henze, G.P. An Overview of Optimal Control for Central Cooling Plants with Ice Thermal Energy Storage. *J. Sol. Energy Eng.* **2003**, *125*, 302. [[CrossRef](#)]
17. Yaïci, W.; Entchev, E. Performance prediction of a solar thermal energy system using artificial neural networks. *Appl. Therm. Eng.* **2014**, *73*, 1348–1359. [[CrossRef](#)]
18. Chou, J.S.; Bui, D.K. Modeling heating and cooling loads by artificial intelligence for energy-efficient building design. *Energy Build.* **2014**, *82*, 437–446. [[CrossRef](#)]

19. Ben-Nakhi, A.E.; Mahmoud, M.A. Cooling load prediction for buildings using general regression neural networks. *Energy Convers. Manag.* **2004**, *45*, 2127–2141. [[CrossRef](#)]
20. Kalogirou, S.A. Optimization of solar systems using artificial neural-networks and genetic algorithms. *Appl. Energy* **2004**, *77*, 383–405. [[CrossRef](#)]
21. Abokersh, M.H.; Vallès, M.; Cabeza, L.F.; Boer, D. A framework for the optimal integration of solar assisted district heating in different urban sized communities: A robust machine learning approach incorporating global sensitivity analysis. *Appl. Energy* **2020**, *267*, 114903. [[CrossRef](#)]
22. Dalipi, F.; Yildirim Yayilgan, S.; Gebremedhin, A. Data-Driven Machine-Learning Model in District Heating System for Heat Load Prediction: A Comparison Study. *Appl. Comput. Intell. Soft Comput.* **2016**, *2016*, 3403150. [[CrossRef](#)]
23. Vlachopoulou, M.; Chin, G.; Fuller, J.C.J.C.; Lu, S.; Kalsi, K. Model for aggregated water heater load using dynamic bayesian networks. *Proc. Int. Conf. Data Sci.* **2012**, *1*, 818–823.
24. Sajjadi, S.; Shamsirband, S.; Alizamir, M.; Yee, P.L.; Mansor, Z.; Manaf, A.A.; Altameem, T.A.; Mostafaeipour, A. Extreme learning machine for prediction of heat load in district heating systems. *Energy Build.* **2016**, *122*, 222–227. [[CrossRef](#)]
25. Yaïci, W.; Entchev, E. Adaptive Neuro-Fuzzy Inference System modelling for performance prediction of solar thermal energy system. *Renew. Energy* **2016**, *86*, 302–315. [[CrossRef](#)]
26. Chia, Y.Y.; Lee, L.H.; Shafiabady, N.; Isa, D. A load predictive energy management system for supercapacitor-battery hybrid energy storage system in solar application using the Support Vector Machine. *Appl. Energy* **2015**, *137*, 588–602. [[CrossRef](#)]
27. Johansson, C.; Bergkvist, M.; Geysen, D.; De Somer, O.; Lavesson, N.; Vanhoudt, D. Operational Demand Forecasting in District Heating Systems Using Ensembles of Online Machine Learning Algorithms. *Energy Procedia* **2017**, *116*, 208–216. [[CrossRef](#)]
28. Moustakidis, S.; Meintanis, I.; Halikias, G.; Karcianas, N. An innovative control framework for district heating systems: Conceptualisation and preliminary results. *Resources* **2019**, *8*, 27. [[CrossRef](#)]
29. Moustakidis, S.; Meintanis, I.; Karkanias, N.; Halikias, G.; Saoutieff, E.; Gasnier, P.; Ojer-Aranguren, J.; Anagnostis, A.; Marciniak, B.; Rodot, I.; et al. Innovative Technologies for District Heating and Cooling: InDeal Project. *Proceedings* **2019**, *5*, 1. [[CrossRef](#)]
30. Popa, D.; Pop, F.; Serbanescu, C.; Castiglione, A. Deep learning model for home automation and energy reduction in a smart home environment platform. *Neural Comput. Appl.* **2018**, *31*, 1317–1337. [[CrossRef](#)]
31. Vázquez-Canteli, J.R.; Ulyanin, S.; Kämpf, J.; Nagy, Z. Fusing TensorFlow with building energy simulation for intelligent energy management in smart cities. *Sustain. Cities Soc.* **2019**, *45*, 243–257. [[CrossRef](#)]
32. Sogabe, T.; Malla, D.B.; Takayama, S.; Shin, S.; Sakamoto, K.; Yamaguchi, K.; Singh, T.P.; Sogabe, M.; Hirata, T.; Okada, Y. Smart Grid Optimization by Deep Reinforcement Learning over Discrete and Continuous Action Space. In Proceedings of the 2018 IEEE 7th World Conference on Photovoltaic Energy Conversion, WCPEC 2018—A Joint Conference of 45th IEEE PVSC, 28th PVSEC and 34th EU PVSEC, Waikoloa, HI, USA, 10–15 June 2018.
33. Cox, S.J.; Kim, D.; Cho, H.; Mago, P. Real time optimal control of district cooling system with thermal energy storage using neural networks. *Appl. Energy* **2019**, *238*, 466–480. [[CrossRef](#)]
34. Rahman, A.; Smith, A.D. Predicting heating demand and sizing a stratified thermal storage tank using deep learning algorithms. *Appl. Energy* **2018**, *228*, 108–121. [[CrossRef](#)]
35. Le Coz, A.; Nabil, T.; Courtot, F. Towards optimal district heating temperature control in China with deep reinforcement learning. *arXiv* **2020**, arXiv:2012.09508.
36. Xue, G.; Pan, Y.; Lin, T.; Song, J.; Qi, C.; Wang, Z. District heating load prediction algorithm based on feature fusion LSTM model. *Energies* **2019**, *12*, 2122. [[CrossRef](#)]
37. Gong, M.; Zhou, H.; Wang, Q.; Wang, S.; Yang, P. District heating systems load forecasting: A deep neural networks model based on similar day approach. *Adv. Build. Energy Res.* **2019**, *14*, 372–388. [[CrossRef](#)]
38. Ullah, A.; Haydarov, K.; Haq, I.U.; Muhammad, K.; Rho, S.; Lee, M.; Baik, S.W. Deep learning assisted buildings energy consumption profiling using smart meter data. *Sensors* **2020**, *20*, 873. [[CrossRef](#)]
39. Corberan, J.M.; Finn, D.P.; Montagud, C.M.; Murphy, F.T.; Edwards, K.C. A quasi-steady state mathematical model of an integrated ground source heat pump for building space control. *Energy Build.* **2011**, *43*, 82–92. [[CrossRef](#)]
40. Petrocelli, D.; Lezzi, A.M. Modeling operation mode of pellet boilers for residential heating. *Proc. J. Phys. Conf. Ser.* **2014**, *547*, 012017. [[CrossRef](#)]
41. Campos Celador, A.; Odriozola, M.; Sala, J.M. Implications of the modelling of stratified hot water storage tanks in the simulation of CHP plants. *Energy Convers. Manag.* **2011**, *52*, 3018–3026. [[CrossRef](#)]
42. Shin, M.S.; Kim, H.S.; Jang, D.S.; Lee, S.N.; Lee, Y.S.; Yoon, H.G. Numerical and experimental study on the design of a stratified thermal storage system. *Appl. Therm. Eng.* **2004**, *24*, 17–27. [[CrossRef](#)]
43. Montes, M.J.; Abánades, A.; Martínez-Val, J.M. Thermofluidynamic Model and Comparative Analysis of Parabolic Trough Collectors Using Oil, Water/Steam, or Molten Salt as Heat Transfer Fluids. *J. Sol. Energy Eng.* **2010**, *132*, 021001. [[CrossRef](#)]
44. Duffie, J.A.; Beckman, W.A. *Solar Engineering of Thermal Processes: Fourth Edition*; Wiley: New York, NY, USA, 2013; ISBN 9780470873663.
45. Notton, G.; Motte, F.; Cristofari, C.; Canaletti, J.L. New patented solar thermal concept for high building integration: Test and modeling. *Energy Procedia* **2013**, *42*, 43–52. [[CrossRef](#)]
46. Dowson, M.; Pegg, I.; Harrison, D.; Dehouche, Z. Predicted and in situ performance of a solar air collector incorporating a translucent granular aerogel cover. *Energy Build.* **2012**, *49*, 173–187. [[CrossRef](#)]

47. Karim, M.A.; Perez, E.; Amin, Z.M. Mathematical modelling of counter flow v-grove solar air collector. *Renew. Energy* **2014**, *67*, 192–201. [[CrossRef](#)]
48. Géczy-Víg, P.; Farkas, I. Neural network modelling of thermal stratification in a solar DHW storage. *Sol. Energy* **2010**, *84*, 801–806. [[CrossRef](#)]
49. Caner, M.; Gedik, E.; Keçebaş, A. Investigation on thermal performance calculation of two type solar air collectors using artificial neural network. *Expert Syst. Appl.* **2011**, *38*, 1668–1674. [[CrossRef](#)]
50. Sözen, A.; Menlik, T.; Ünvar, S. Determination of efficiency of flat-plate solar collectors using neural network approach. *Expert Syst. Appl.* **2008**, *35*, 1533–1539. [[CrossRef](#)]
51. Esen, H.; Ozgen, F.; Esen, M.; Sengur, A. Artificial neural network and wavelet neural network approaches for modelling of a solar air heater. *Expert Syst. Appl.* **2009**, *36*, 11240–11248. [[CrossRef](#)]
52. Liu, Z.; Li, H.; Zhang, X.; Jin, G.; Cheng, K. Novel method for measuring the heat collection rate and heat loss coefficient of water-in-glass evacuated tube solar water heaters based on artificial neural networks and support vector machine. *Energies* **2015**, *8*, 8814–8834. [[CrossRef](#)]
53. García Nieto, P.J.; García-Gonzalo, E.; Paredes-Sánchez, J.P.; Bernardo Sánchez, A.; Menéndez Fernández, M. Predictive modelling of the higher heating value in biomass torrefaction for the energy treatment process using machine-learning techniques. *Neural Comput. Appl.* **2018**, *31*, 8823–8836. [[CrossRef](#)]
54. Kalogirou, S.; Lalot, S.; Florides, G.; Desmet, B. Development of a neural network-based fault diagnostic system for solar thermal applications. *Sol. Energy* **2008**, *82*, 164–172. [[CrossRef](#)]
55. Ahmad, M.W.; Mourshed, M.; Yuce, B.; Rezgui, Y. Computational intelligence techniques for HVAC systems: A review. *Build. Simul.* **2016**, *9*, 359–398. [[CrossRef](#)]
56. Kalogirou, S.A.; Mathioulakis, E.; Belessiotis, V. Artificial neural networks for the performance prediction of large solar systems. *Renew. Energy* **2014**, *63*, 90–97. [[CrossRef](#)]
57. Energetika Project. Available online: <http://www.energetika-projekt.eu> (accessed on 11 January 2019).
58. Anagnostis, A.; Papageorgiou, E.; Bochtis, D. Application of artificial neural networks for natural gas consumption forecasting. *Sustainability* **2020**, *12*, 6409. [[CrossRef](#)]
59. Anagnostis, A.; Papageorgiou, E.; Dafopoulos, V.; Bochtis, D. Applying Long Short-Term Memory Networks for natural gas demand prediction. In Proceedings of the 2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA), Patras, Greece, 15–17 July 2019; pp. 1–7.
60. Hochreiter, S. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.* **1998**, *6*, 107–116. [[CrossRef](#)]
61. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)]
62. Galton, F. Anthropological Miscellanea. Regression towards Mediocrity in Hereditary Stature. *J. Anthropol. Inst. Great Br. Irel.* **1886**, *15*, 246–263.
63. Santosa, F.; Symes, W.W. Linear Inversion of Band-Limited Reflection Seismograms. *SIAM J. Sci. Stat. Comput.* **1986**, *7*, 1307–1330. [[CrossRef](#)]
64. Tibshirani, R. Regression Selection and Shrinkage via the Lasso. *J. R. Stat. Soc. B* **1996**, *58*, 267–288. [[CrossRef](#)]
65. Pasanen, L.; Holmström, L.; Sillanpää, M.J. Bayesian LASSO, scale space and decision making in association genetics. *PLoS ONE* **2015**, *10*, e0120017. [[CrossRef](#)] [[PubMed](#)]
66. De Mol, C.; De Vito, E.; Rosasco, L. Elastic-net regularization in learning theory. *J. Complex.* **2009**, *25*, 201–230. [[CrossRef](#)]
67. Efron, B.; Hastie, T.; Johnstone, I.; Tibshirani, R.; Ishwaran, H.; Knight, K.; Loubes, J.M.; Massart, P.; Madigan, D.; Ridgeway, G.; et al. Least angle regression. *Ann. Stat.* **2004**, *32*, 407–499. [[CrossRef](#)]
68. Robbins, H.; Monro, S. A Stochastic Approximation Method. *IEEE Trans. Syst. Man Cybern.* **1971**, *1*, 338–344.
69. Belson, W.A. *Matching and Prediction on the Principle of Biological Classification*; Wiley: New York, NY, USA, 1959; Volume 8, ISBN 0000000779.
70. Breiman, L.; Friedman, J.H.; Olshen, R.A.; Stone, C.J. *Classification and Regression Trees*; Routledge: New York, NY, USA, 2017; ISBN 9781351460491. [[CrossRef](#)]
71. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [[CrossRef](#)]
72. Freund, Y.; Schapire, R.; Abe, N. A Short Introduction to Boosting. *J. Jpn. Soc. Artif. Intell.* **1999**, *14*, 1612.
73. Freund, Y.; Schapire, R.R.E. Experiments with a New Boosting Algorithm. In Proceedings of the 13th International Conference on Machine Learning, Bari, Italy, 3–6 July 1996.
74. Breiman, L. Bagging predictors. *Mach. Learn.* **1996**, *24*, 123–140. [[CrossRef](#)]
75. Vapnik, V.N. An overview of statistical learning theory. *IEEE Trans. Neural Netw.* **1999**, *10*, 988–999. [[CrossRef](#)]
76. Suykens, J.A.K.; Vandewalle, J. Least squares support vector machine classifiers. *Neural Process. Lett.* **1999**, *9*, 293–300. [[CrossRef](#)]
77. Chang, C.; Lin, C.; Tieleman, T. LIBSVM: A Library for Support Vector Machines. *ACM Trans. Intell. Syst. Technol.* **2008**, *2*, 1–27. [[CrossRef](#)]
78. McCulloch, W.S.; Pitts, W. A logical calculus of the ideas immanent in nervous activity. *Bull. Math. Biophys.* **1943**, *5*, 115–133. [[CrossRef](#)]
79. Rosenblatt, F. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychol. Rev.* **1958**, *65*, 386. [[CrossRef](#)] [[PubMed](#)]

80. Pal, S.K.; Mitra, S. Multilayer Perceptron, Fuzzy Sets, and Classification. *IEEE Trans. Neural Netw.* **1992**, *3*, 683–697. [[CrossRef](#)] [[PubMed](#)]
81. Linnainmaa, S. Taylor expansion of the accumulated rounding error. *BIT* **1976**, *16*, 146–160. [[CrossRef](#)]
82. Riedmiller, M.; Braun, H. A direct adaptive method for faster backpropagation learning: The RPROP algorithm. In Proceedings of the IEEE International Conference on Neural Networks—Conference Proceedings, Nagoya, Japan, 25–29 October 1993.
83. Broomhead, D.; Lowe, D. Multivariable functional interpolation and adaptive networks. *Complex Syst.* **1988**, *2*, 321–355.
84. McClelland, J.L.; Rumelhart, D.E.; McClelland, J.L. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 2: Psychological and Biological Models*; The MIT Press: Boston, MA, USA, 1986; ISBN 0262132184.
85. Jang, J.S.R. ANFIS: Adaptive-Network-Based Fuzzy Inference System. *IEEE Trans. Syst. Man Cybern.* **1993**, *23*, 665–685. [[CrossRef](#)]
86. Jordan, M.I. Attractor dynamics and parallelism in a connectionist sequential machine. In Proceedings of the Eighth Annual Conference Cognitive Science Society, Amhurst, MA, USA, 15–17 August 1986; pp. 531–546.
87. Fukushima, K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biol. Cybern.* **1980**, *36*, 193–202. [[CrossRef](#)]
88. Ivakhnenko, A.G. Polynomial Theory of Complex Systems. *IEEE Trans. Syst. Man Cybern.* **1971**, *4*, 364–378. [[CrossRef](#)]
89. Nair, V.; Hinton, G.E. Rectified linear units improve Restricted Boltzmann machines. In Proceedings of the ICML 2010—Proceedings, 27th International Conference on Machine Learning, Haifa, Israel, 21–24 June 2010.
90. Kingma, D.P.; Ba, J.L. Adam: A method for stochastic optimization. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015—Conference Track Proceedings, San Diego, CA, USA, 7–9 May 2015.
91. Harris, C.R.; Millman, K.J.; van der Walt, S.J.; Gommers, R.; Virtanen, P.; Cournapeau, D.; Wieser, E.; Taylor, J.; Berg, S.; Smith, N.J.; et al. Array programming with NumPy. *Nature* **2020**, *585*, 357–362. [[CrossRef](#)]
92. McKinney, W. Data Structures for Statistical Computing in Python. In Proceedings of the 9th Python in Science Conference, Austin, TX, USA, 28 June–3 July 2010.
93. Hunter, J.D. Matplotlib: A 2D graphics environment. *Comput. Sci. Eng.* **2007**, *9*, 90–95. [[CrossRef](#)]
94. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
95. Chollet, F. Others Keras. 2015. Available online: <https://github.com/fchollet/keras> (accessed on 11 January 2019).
96. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. *arXiv* **2016**, arXiv:1603.04467.
97. Kiseľák, J.; Lu, Y.; Švihra, J.; Szépe, P.; Stehlík, M. “SPOCU”: Scaled polynomial constant unit activation function. *Neural Comput. Appl.* **2020**, *33*, 3385–3401. [[CrossRef](#)]