ECAI 2024
U. Endriss et al. (Eds.)

doi:10.3233/FAIA241051

Predicting Solar Heat Production to Optimize Renewable Energy Usage

Tatiana Boura[a,*], Natalia Koliou[a], George Meramveliotakis[b], Stasinos Konstantopoulos[a] and George Kosmadakis[b]

[a]Institute of Informatics and Telecommunications, NCSR 'Demokritos', Ag. Paraskevi, Greece
[b]Institute of Nuclear & Radiological Sciences and Technology, Energy & Safety, NCSR 'Demokritos', Ag. Paraskevi, Greece
ORCID (Tatiana Boura): https://orcid.org/0009-0008-0656-4372, ORCID (Natalia Koliou): https://orcid.org/0009-0004-3920-9992, ORCID (George Meramveliotakis): https://orcid.org/0000-0002-0110-347X, ORCID (Stasinos Konstantopoulos): https://orcid.org/0000-0002-2586-1726, ORCID (George Kosmadakis): https://orcid.org/0000-0002-3671-8693

Abstract. Utilizing solar energy to meet space heating and domestic hot water demand is very efficient (in terms of environmental footprint as well as cost), but in order to ensure that user demand is entirely covered throughout the year needs to be complemented with auxiliary heating systems, typically boilers and heat pumps. Naturally, the optimal control of such a system depends on an accurate prediction of solar thermal production.

Experimental testing and physics-based numerical models are used to find a collector's performance curve — the mapping from solar radiation and other external conditions to heat production — but this curve changes over time once the collector is exposed to outdoor conditions. In order to deploy advanced control strategies in small domestic installations, we present an approach that uses machine learning to automatically construct and continuously adapt a model that predicts heat production. Our design is driven by the need to (a) construct and adapt models using supervision that can be extracted from low-cost instrumentation, avoiding extreme accuracy and reliability requirements; and (b) at inference time, use inputs that are typically provided in publicly available weather forecasts.

Recent developments in attention-based machine learning, as well as careful adaptation of the training setup to the specifics of the task, have allowed us to design a machine learning-based solution that covers our requirements. We present positive empirical results for the predictive accuracy of our solution, and discuss the impact of these results on the end-to-end system.

Keywords: Solar energy; Renewable energy; Deep learning

1 Introduction

Efficient utilization of renewable energy, especially solar energy through solar thermal or PV/thermal (PVT) collectors, is a promising solution to meet the needs for heating and domestic hot water, offering substantial advantages with respect to both environmental footprint and operational costs. Particularly during mid-seasons and summer, the available solar radiation effectively eliminates the need for conventional boilers. However, to ensure that user demand is entirely covered throughout the year, collectors are complemented with auxiliary heating systems, typically boilers and heat pumps.

The controller of such a system should minimize the use of auxiliary heating by employing insulated hot water tanks to shift the load from when demand is high to when there is excess production. Although this is relatively straight-forward for space heating, combined space heating and domestic hot water usage is more challenging as the latter experiences demand spikes. Individual usage patterns can be user-parameterized or statistically estimated, but optimal control also requires reliable prediction of solar thermal production to match the demand and reduce heat losses from the storage tank.

Given the importance of estimating solar thermal collectors' performance, it is only natural that the problem has been approached from multiple angles. Based on experimental testing, the methods for estimating the efficiency of collectors *under specified conditions* have been standardized (ASHRAE 93, ISO 9806, EN12976-2). These methods provide the parameters required to predict the long term performance of solar thermal collectors [18]. Numerical modelling methods have also been developed to estimate the heat production and efficiency of collectors [1, 8]. These models account for heat transfer mechanisms such as conduction, convection, and radiation under known plate geometry and operating conditions and serve as useful tools for the theoretical design and optimization of these components.

Accurate as they might initially be, the major shortcoming of experimental and numerical methods is that the performance curve of the collector changes over time once the collector is exposed to outdoor conditions. In order to deploy advanced control strategies at scale, we need an approach that does not rely on prior knowledge of the performance curve, but can automatically construct and continuously adapt a model that predicts heat production based on expected external conditions (solar radiation and ambient temperature). Machine learning is a natural fit for such a task, and unsurprisingly, there is a rich literature on applying machine learning to construct predictors for thermal collectors' heat output.

In the work presented here, we focus on the additional requirements for the mass-deployment of advanced control strategies in

* Corresponding Author. Email: tatianabou@iit.demokritos.gr.

small domestic installations. Besides the continuous adaptation covered by using machine learning in general (presented in Section 2), we also aim for a system that relies only on data that can be obtained from relatively low-cost instrumentation and public weather forecasts. To achieve this, we rely on recent developments in deep learning (presented in the second part of Section 2) and the adaptation of the machine learning setup to the specifics of the task (Section 3). We then present empirical results (Section 4) and conclude (Section 5).

2 Background

2.1 Predicting thermal collector performance

As already stated above, machine learning has been extensively applied to construct models of solar thermal and PVT collectors. Ghritlahre and Prasad [6] give a comprehensive review, and show that these machine learning exercises are generally successful. Solar radiation and ambient temperature serve are the key inputs in most of these models, unsurprisingly, as physics-based models also depend on these two variables.

What is worth noting about the relevant literature, is that it is to a large extent addressing the design phase of collectors or otherwise studying collectors outside the context of controlling solar and auxiliary heating systems. For instance, and moving forward to more recent works, Sadeghzadeh et al. [20] compared multi-layer perceptron, RBF, and Elman back-propagation ANNs on the task of modelling the efficiency of solar collectors at various flow rates of the working fluid; whereas Mirzaei and Mohiabadi [14] compared how the use of different working fluids affected the accuracy of ANN models in predicting collector performance. The study demonstrated the effectiveness of ANN in predicting collector performance across all three fluids, highlighting the potential of machine learning as a cost-effective and time-saving alternative to conventional testing and modelling methods.

Closer to our work, Gunasekar et al. [7] used Multi-Layered Perceptron Neural Networks to predict the thermal performance of a PVT evaporator of a solar-assisted heat pump, demonstrating close alignment with experimental data and also identifying solar radiation and ambient temperature as the primary input variables. Du et al. [3] trained a convolutional neural network on a richer set of inputs (solar radiation, ambient temperature, wind speed, fluid flow rate, and fluid inlet temperature). The developed model showed strong predictive capability in sunny conditions, but the authors do not elaborate on the impact of the extended inputs on the prediction. Additionally, Du et al. also note that clustering analysis to screen the data and eliminate outliers improves the accuracy of the prediction.

2.2 Time representation

One of the practical considerations often encountered in timeseries processing is that fully observed, uniformly sampled inputs are almost impossible to gather in realistic conditions. Reasons include gaps in the data, varying sampling rates, and (for multivariate timeseries) misalignment between variables' timing.

One approach for facing the reality of *irregularly sampled and sparse multivariate timeseries* is to first reconstruct a full, uniformly sampled timeseries while another is to directly look for the underlying structure of the data that is available. Reconstruction methods range from simple imputation and aggregation [12, 10] to sophisticated interpolation methods [26, 22].

Other works adjust the structure of known recurrent neural networks to directly handle irregularly sampled timeseries as input. Che et al. [2] present several methods based on Gated Recurrent Units (GRUs), Pham et al. [16] propose to capture time irregularity by modifying the forget gate of an LSTM and Neil et al. [15] introduce a new time gate to be utilized within an LSTM. Another, more recent algorithm is proposed by Schirmer et al. [21] assuming a hidden state that evolves according to a linear stochastic differential equation and is integrated into an encoder-decoder framework. Rubanova et al. [19] present the ODE-RNNs, another auto-encode architecture, whose hidden state dynamics are specified by neural ordinary differential equations. These methods, however, fail to learn directly from partially observed vectors, which often occur when dealing with multiple variables.

Some early works [9, 11] utilize the idea of Gaussian Processes (GPs) to model irregular timeseries. These works first optimize GP parameters and then train the corresponding model. Futoma et al. [5] achieve end-to-end training for multivariate timeseries by using the re-parametrization trick to back-propagate the gradients through a black-box classifier. Though GPs provide a systematic way to deal with uncertainty, they are expensive to learn and are highly dependent on the chosen mean and covariance functions.

To go past these challenges, many researchers have also explored the use of Transformers. Unlike recurrent and differential equation-based architectures, which process inputs sequentially, Transformers use self-attention mechanisms [24] to capture relationships between all input positions simultaneously. Although they can be more efficient due to their parallel processing nature, they lack an inherent mechanism for representing temporal order. To address this limitation, a proposed method is to augment the input features with *time embeddings*. When the data is known or suspected to exhibit periodicity, Wen et al. [25] propose using sinusoidal functions that separate periodicity and phase. For instance, adding as features the sine and cosine of (numerical representation of) months, days, hours as input features provides the model with the means to capture periodicity at across different time scales. We shall refer to this model as CycTime.

Naturally, this representation is static and assumes prior knowledge of which periodicities make sense for the phenomenon being modelled. As expected, a line of research was developed to enhance the capabilities of attention-based architectures to inherently address the lack of time information. The multi-time attention network (mTAN) [23], is similar to kernel-based interpolation with the difference that the attention-based similarity kernel is learnt. This model *acquires* the time embeddings from a shallow network that learns both *periodic time patterns* and *non-periodic trends*. These embeddings are then utilized by a multi-head attention mechanism that produces an embedding module composed of a set of reference time points and also re-represents the input features in a fixed-dimensional space.

This discussion motivates our choice of candidate models: CycTime can take advantage of our prior knowledge of the existence of annual and daily periodicites in solar heat production, while mTAN can discover the interaction between the annual/daily periodicity and the efficiency degradation trend of the collectors.

3 Optimal Control for PVT Installations

As mentioned in the Introduction, our application targets the mass-deployment of advanced control strategies in small domestic installations. The high-level description of such an installation and its operation is as follows:

Table 1. Q_{PVT} class thresholds (kWh)

Balanced Ranges	Balanced Classes	Max Margins
0.05	0.05	0.05
0.50	0.29	0.21
1.00	1.24	0.53
1.50	3.70	1.05

- Hot water is produced by solar thermal or by PVT collectors that simultaneously produce thermal and electrical energy. Electricity production in the PVT case is not taken into account for the purposes of the work described here.
- Hot water is used for both space heating and to cover domestic hot water demand. Ideally, only water from the solar thermal collector will be used for both usages, but in order to ensure meeting demand the system is also equipped with (less efficient) auxiliary heating systems.
- The system decides automatically how to distribute the available hot water between space heating and domestic hot water demand and when to use the auxiliary heating system.

Our experimental installation is instrumented with a flow rate meter, as well as inlet and outlet temperature sensors from which hot water production can be quantified.[1] Prior work has used this experimental installation on NCSR 'Demokritos' premises to test and validate this configuration for both office (space heating only) and domestic (space heating and hot water) usage profiles [13, 17]. The system is controlled by static summertime and wintertime rules that are tailored to the local climate. In our domestic application, the crucial quantity for these rules is how the expected *heat production (Q_{PVT})* compares to the expected *domestic hot water (DHW)* demand.

Q_{PVT} can be directly calculated from sensor data as a function of the input and output temperatures of the collector and the water flow. Note that although some parameters are hardware dependent, they are constant throughout an installation's life-cycle and do not need to be re-evaluated.[2] But it can also be measured through the (experimentally determined) *collector efficiency curve* that maps external conditions to Q_{PVT}. Using the collector efficiency curve has the advantage that we can use weather forecasts to *predict* the expected Q_{PVT}, but (as already noted) the curve changes as the collector ages and needs to be re-evaluated.

What our application achieves is to *continuously maintain* a machine-learned surrogate of the collector efficiency curve by using the installation's instrumentation to directly calculate Q_{PVT}, which is then used to label external conditions as provided by local weather reports.

There is significant inherent uncertainty in the processes we aim to model, thus it helps to abstract the prediction to bands of Q_{PVT} values that are meaningful for the given task. Further to this, as the system includes an insulated hot water tank, the timing of the production does not need to closely match the timing of the demand, not even for the morning spike as the previous day's water can be used, if there was an excess. In other words, we are not interested in the exact timing of the production, but only if the system expects to run out of hot water at some point during the day.

This gives us the opportunity to reduce uncertainty by abstracting the predictive task to wide bands of heat production over wide time periods, instead of aiming to predict specific Q_{PVT} values at specific time-points. Since weather forecasts are published for 3h periods, it makes sense to target this granularity in the time dimension:

[1] These sensors are currently installed for our experimental purposes, but similar information can be obtained from low-cost domestic heat meters.

[2] Assuming the mirrors are regularly cleaned.

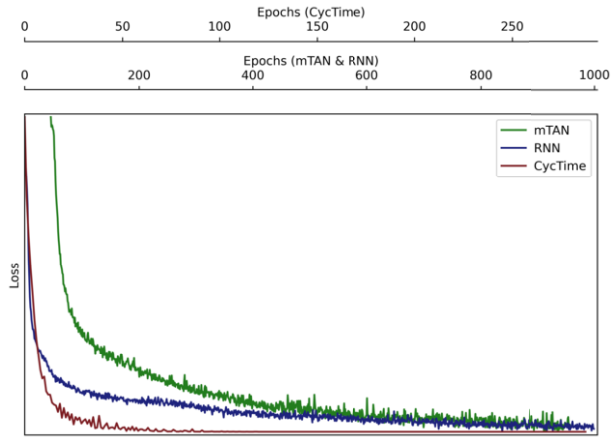

Figure 1. Normalized training loss progression for each model. Task is predicting the 'Max Margins' bins.

Any finer predictions would simply repeat the same inputs, or rely on interpolations that would add uncertainty; any coarser predictions would have to aggregate the inputs thus reducing the information available to the system.

Regarding the granularity at which the value of Q_{PVT} is predicted, we have tested three alternative approaches:

- Balanced ranges, simply separating the range of Q_{PVT} values into bands of equal width.
- Balanced classes, selecting thresholds that give a balanced dataset by simply sorting all values in the training data and cutting into equal sub-arrays.
- Maximum margins, selecting thresholds that are as far as possible from expected levels of *demand*, in order to offer downstream applications robustness to small divergence between expected and actual demand.

For the maximum-margins thresholds we referred to the *DHW demand profiles* that are used to validate energy savings from ecodesign of components [4, Annex III, Table 1]. We set the thresholds to be exactly between the distinct values appearing in that table. The specific threshold values are given in Table 1.[3]

4 Experiments and Results

4.1 Experimental Setup

Our data was collected from the pilot building at NCSR 'Demokritos'. The building uses PVT for space heating and hot water, with a heat pump as auxiliary heating system. The building is equipped with all necessary sensors to monitor the performance of the various components tested there, including, among others, a pyranometer (for solar radiation), a temperature sensor with radiation shield (for ambient temperature), temperature sensors (for water/glycol inlet/outlet temperatures), and flow rate meters (for water/glycol and hot water discharge). Our focus for the work described here is on the part of the test facility that includes 4 PVT collectors (total surface of 8.6 sq.m.) charging a 300-litre water tank. The medium tapping profile is programmed to remove heat from this tank with the use of a controllable valve.

[3] The maximum-margins thresholds in this table are based on the *medium profile*. It is straightforward to adjust them for different usage profiles.

Table 2. Evaluation results for the classification tasks. Each model was trained using six different initialization seeds, and the final metrics emerge from the aggregation (mean and standard deviation) of the results from these trained models.

Balanced Ranges			
Metric	RNN	mTAN	CycTime
Precision (Micro)	0.383 ± 0.201	0.740 ± 0.032	$\mathbf{0.845} \pm 0.008$
Recall (Micro)	0.383 ± 0.201	0.740 ± 0.032	$\mathbf{0.845} \pm 0.008$
F-Score (Micro)	0.383 ± 0.201	0.740 ± 0.032	$\mathbf{0.845} \pm 0.008$
Precision (Macro)	0.278 ± 0.055	0.580 ± 0.028	$\mathbf{0.584} \pm 0.053$
Recall (Macro)	0.298 ± 0.027	$\mathbf{0.567} \pm 0.071$	0.553 ± 0.020
F-Score (Macro)	0.237 ± 0.060	$\mathbf{0.547} \pm 0.039$	0.544 ± 0.026
Precision (Weighted)	0.532 ± 0.102	$\mathbf{0.843} \pm 0.017$	$\mathbf{0.843} \pm 0.018$
Recall (Weighted)	0.383 ± 0.201	0.740 ± 0.032	$\mathbf{0.845} \pm 0.008$
F-Score (Weighted)	0.372 ± 0.192	0.784 ± 0.025	$\mathbf{0.833} \pm 0.014$
Balanced Classes			
Metric	RNN	mTAN	CycTime
Precision (Micro)	0.395 ± 0.221	0.793 ± 0.010	$\mathbf{0.794} \pm 0.008$
Recall (Micro)	0.395 ± 0.221	0.793 ± 0.010	$\mathbf{0.794} \pm 0.008$
F-Score (Micro)	0.395 ± 0.221	0.793 ± 0.010	$\mathbf{0.794} \pm 0.008$
Precision (Macro)	0.332 ± 0.120	$\mathbf{0.704} \pm 0.021$	0.653 ± 0.013
Recall (Macro)	0.361 ± 0.087	$\mathbf{0.760} \pm 0.020$	0.691 ± 0.020
F-Score (Macro)	0.295 ± 0.128	$\mathbf{0.718} \pm 0.021$	0.661 ± 0.016
Precision (Weighted)	0.548 ± 0.093	$\mathbf{0.825} \pm 0.013$	0.783 ± 0.013
Recall (Weighted)	0.395 ± 0.221	0.793 ± 0.010	$\mathbf{0.794} \pm 0.008$
F-Score (Weighted)	0.392 ± 0.203	$\mathbf{0.800} \pm 0.010$	0.784 ± 0.010
Max Margins			
Metric	RNN	mTAN	CycTime
Precision (Micro)	0.453 ± 0.143	0.765 ± 0.031	$\mathbf{0.831} \pm 0.009$
Recall (Micro)	0.453 ± 0.143	0.765 ± 0.031	$\mathbf{0.831} \pm 0.009$
F-Score (Micro)	0.453 ± 0.143	0.765 ± 0.031	$\mathbf{0.831} \pm 0.009$
Precision (Macro)	0.310 ± 0.041	0.634 ± 0.033	$\mathbf{0.684} \pm 0.037$
Recall (Macro)	0.381 ± 0.039	$\mathbf{0.701} \pm 0.037$	0.638 ± 0.011
F-Score (Macro)	0.283 ± 0.048	$\mathbf{0.651} \pm 0.037$	0.636 ± 0.019
Precision (Weighted)	0.622 ± 0.028	$\mathbf{0.822} \pm 0.016$	0.817 ± 0.012
Recall (Weighted)	0.453 ± 0.143	0.765 ± 0.031	$\mathbf{0.831} \pm 0.009$
F-Score (Weighted)	0.447 ± 0.153	0.779 ± 0.025	$\mathbf{0.813} \pm 0.008$

From the operation of the pilot building we have acquired one year's worth of data, from which Q_{PVT} ground-truth can be calculated.[4] For the same period we have also obtained weather data i.e., temperature, humidity levels, pressure, wind speed, accumulated rain and snow, measured for the wider region that includes the installation. We do not use weather forecasts, but actual weather data in order to factor out the effect of weather forecasting uncertainty, but the weather data we used is of the same kind and detail as the forecasts published daily by the National Observatory of Athens. As we mentioned already, weather forecasts are published for 3h periods, so for our dataset we also aggregated Q_{PVT} accordingly.

It is a common occurrence when collecting data from sensors to observe missing values within the dataset due to defective sensors, networking issues etc. In our case, we often have water temperature values falling outside the accepted temperature range of -20°C to 100°C. As Q_{PVT} is calculated from multiple water temperatures, it has approximately 15% missing values due to one or more of the variable needed to calculate Q_{PVT} being missing.

Putting everything together, our machine learning task is to predict a time-series of length 8 (because of the 3h aggregation) where each prediction places Q_{PVT} in one of the 5 classes (value bins) defined by the thresholds in Table 1.

[4] Considering the 3h aggregation, the data comprises 8 datapoints per day. The dataset is available at https://zenodo.org/records/12818885

4.2 Model Architectures

To perform the classification task we compare three methods: (a) a conventional RNN approach, (b) the CycTime transformer that uses fixed-time embeddings and (c) the mTAN module.

Baseline We include in out experiments a standard timeseries processing approach as the baseline: linear interpolation to fill missing and out-of-range sensor values, RNN, linear classification layer.

Transformer with fixed time embedding (CycTime) Our next approaches to tackle the classification problem are based on the idea of Transformers. This first method uses fixed positional time encodings. Regarding the modelâĂŹs architecture, it consists of an encoder, a decoder, and a classifier. The encoder maps the input into a latent representation, while the decoder reconstructs it, preserving the original dimensionality of its features. The classifier layer then outputs a probability distribution over the five classes, with each element representing the probability of the input belonging to a particular class.

Multi-time attention network (mTAN) Adhering to the trajectory of attention-based approaches but employing a methodology based on learning the positional encodings of a timeseries, we apply the mTAN [23] module. We deployed this module into an encoder-decoder architecture with the decoder being a single linear classification layer, the same as the one used in the previously discussed architectures. We found that using 32 time-reference points achieved better results. It is expected that the system should be given some

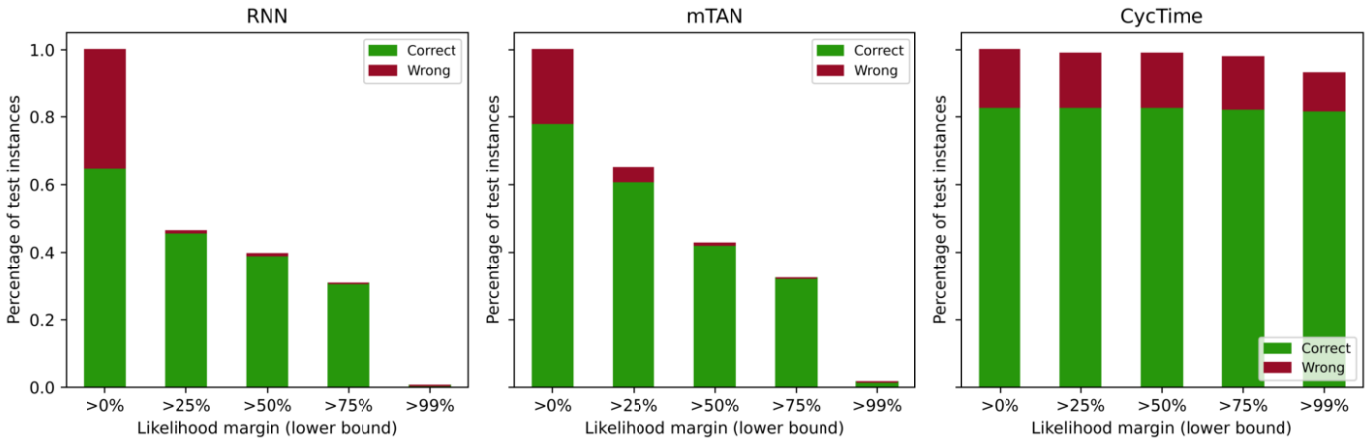

Figure 2. The percentage of correct (green) and wrong (red) predictions, split on lower bounds on the likelihood margin between the predicted (most likely) class and the second most likely class. Naturally, the left-most bar gives the accuracy on the complete test set. Bars to the right give the accuracy on increasingly-confident decisions.

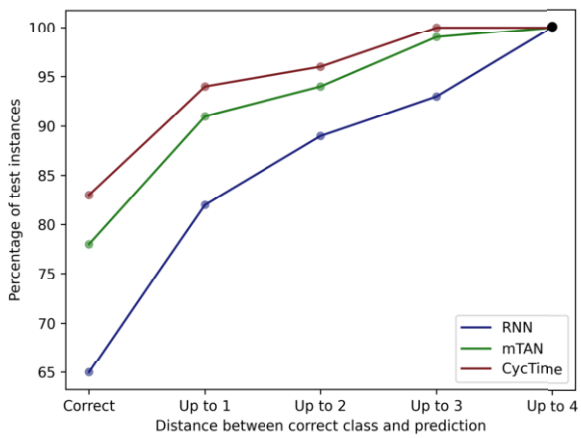

Figure 3. The percentage of test instances that are predicted of be of a class that is up to N classes away from the ground-truth class, following the natural ordering of the value ranges that each class represents.

freedom to organize time in finer steps than the eight three-hour steps it receives as input, and using a multiple of 8 makes it straightforward to aggregate the outputs back to three-hour steps. We found empirically that 32 steps and using mean as an aggregator works best.

All models were trained using multi-class cross-entropy as the loss function, with an implemented mechanism to mask unobserved predicted labels.[5] Figure 1 presents the models' training loss progression over epochs. Note that although the CycTime model appears to have the longest training time per epoch (10 times longer than the mTAN model), it requires 10 times fewer epochs on average to converge. Consequently, both models require roughly the same overall training time, which is 13-15 minutes on a single NVIDIA RTX 5000 Ada Generation GPU. This level of computational load should not pose any challenge for monthly or even weekly model updates.

[5] The experimental setup is implemented in Python and is available from https://zenodo.org/records/12819005

4.3 Results and Discussion

We evaluated the performance of each model on the final two days of every month, data that we had excluded from the training set. The non-domain-specific evaluation metrics, which capture the micro, macro, and weighted recall, precision, and F-score achieved by the models in each predictive task, are presented in Table 2. The micro average calculates the overall performance across all classes, while the macro average computes the average score across all classes, treating each class equally, and the weighted measurements calculate each metric with each class weighted by its support. As anticipated, the RNN model exhibits poorer performance across all tasks, while the other attention-based models compete closely with each other.

Regarding the choice of Q_{PVT} value quantization, there does not seem to be any dramatic advantage in class-balancing. This affords us the flexibility to prefer the max-margins quantization, which offers an advantage for downstream processing. For this reason, we will continue the discussion using results from the max-margins quantization.

Besides knowing which is the most accurate model, it is also important to analyse their behaviour when they err. Since the task is one of inherent uncertainty, we expect errors and we expect that the downstream controller will be aware of the possibility of errors and behave accordingly. This means that we are interested in the graceful degradation of model accuracy so that the controller is not completely thrown off and in having models that do not fail with confidence, so that the control can trust high-confidence predictions.

Failing with confidence Figure 2 shows the confidence with which the different models make predictions. Specifically, it displays the percentage of test instances that were classified correctly or misclassified with respect to different confidence margins of the prediction. Each prediction represents the distribution of likelihoods among five different value ranges. As a measure of confidence, we denote the likelihood difference between the best guess and the second best guess. Ideally, we would like the likelihood mass for correctly predicted examples to be concentrated on a single output, while the likelihood mass for misclassified samples to be spread across the five different classes. Simply put, model confidence is a positive quality only when its predictions are also accurate. Although the RNN model

Table 3. Percentage of all errors that occurred during the early daylight period and during the main heat-producing period.

	Balanced Ranges		Balanced Classes		Max Margins	
	7am-10am	**10am-7pm**	**7am-10am**	**10am-7pm**	**7am-10am**	**10am-7pm**
RNN	57%	41%	54%	43%	66%	34%
mTAN	68%	32%	69%	31%	62%	31%
CycTime	56%	44%	58%	42%	49%	51%

exhibits the desired confidence behaviour, its accuracy is worse than that of the other two models. The mTAN architecture has better accuracy than RNN, but with a similar behaviour when failing. On the other hand the CycTime model, although more accurate than the other models, when it fails it fails with confidence.

Distance between true and predicted class Figure 3 shows the distance between the predicted class and the ground-truth class for the different models. There are no 'reversals' of the relative ordering of the three models, so as far as graceful degradation is concerned CycTime is clearly preferred.

Morning errors Most errors occur between 7am and 10am, with the practically all of rest falling between 10am and 7pm (cf. Table 3). The high concentration of errors during the 7-10am period can be attributed to the morning's dual nature, resembling both day and night depending on the season. For instance, summer mornings are sunny, resembling daytime, while winter mornings may still appear dark, resembling night-time. It seems natural that CycTime is able to handle this slightly better than the rest, since it is equipped with the means to represent the annual cycle as priors. We should remind at this point that we cannot change the time aggregation in order to mitigate this, as it based on how weather forecasts are published.

5 Conclusions and Future Work

We have presented an application of the state-of-the-art in Transformer models to the high-impact domain of renewable energy. Specifically, we applied Transformer models to predict heat production from solar collectors to identify whether it is adequate to cover the demand and, ultimately, to minimize the usage of (less efficient) auxiliary heating systems.

Based on the analysis presented above, we have decided to install the mTAN model based on both theoretical and empirical reasons. From the theoretical point of view, mTAN should be able to separate the annual/daily cycles from the longer trend of collector efficiency deterioration. We will revisit this empirically once the collectors at our installation have aged enough to see perceptible changes, but having this capability supports further applications such as predictive maintenance.

From the empirical analysis, we have observed that having a static prior time representation gives a noticeable but not impressive advantage in prediction accuracy. However, mTAN has the advantage that is less prone to failing with confidence: When it fails, mTAN tends to have called a prediction on a slim difference in likelihood, whereas the CycTime tends to have much higher confidence in success as well as in failure. mTAN's behaviour is more appropriate for our application, since the controller can be configured to use the more conservative or the more optimistic predictions, depending on the expected demand and on the current condition of the system's hot water tanks.

With the mTAN predictor installed and actively in use, we will now be able to validate the actual energy gains by comparison to the conventional (season-based) control previously used. This will allow us to experiment with different policies on how often to re-train the model and, in general, to validate the complete concept of using installation's sensor data to have the system automatically adapt.

In the analysis presented here we have used actual weather conditions instead of weather forecasts, as the focus was to see how well we can model the installation's internal state. In future work, we will investigate the combined effect of weather forecasting uncertainty and model uncertainty when controlling conservatively or aggressively. Finally, in further future work we plan to revisit the smart control strategy with the aim to define a more dynamic strategy that can 'absorb' these uncertainties. Such a strategy would exploit the system's internal state (e.g., the current charge of the hot water tanks) and the knowledge of the margins of uncertainty of future demand, weather forecasts, and the heat production predictor to balance these uncertainties into an control that is optimal in the long run.

Acknowledgements

This research has been co-financed by the European Union and Greek national funds through the program 'Flagship actions in interdisciplinary scientific areas with a special interest in the connection with the production network' — GREEN SHIPPING — TAEDR-0534767 (Acronym: NAVGREEN). For more information please visit https://navgreen.gr

AWS resources were provided by the National Infrastructures for Research and Technology GRNET and funded by the EU Recovery and Resiliency Facility.

References

[1] J. Cadafalch. A detailed numerical model for flat-plate solar thermal devices. *Solar Energy*, 83, 2009. doi: doi:10.1016/J.SOLENER.2009.08.013.

[2] Z. Che, S. Purushotham, K. Cho, D. Sontag, and Y. Liu. Recurrent neural networks for multivariate time series with missing values. *Scientific Reports*, 8, 04 2018.

[3] B. Du, P. D. Lund, and J. Wang. Improving the accuracy of predicting the performance of solar collectors through clustering analysis with artificial neural network models. *Energy Reports*, 8, Nov. 2022. doi: doi:10.1016/J.EGYR.2022.03.013.

[4] European Commission. Regulation (EU) no 814/2013. OJ L 239 6.9.2013, p. 162, 2013. URL http://data.europa.eu/eli/reg/2013/814/2017-01-09. Ecodesign requirements for water heaters and hot water storage tanks.

[5] J. Futoma, S. Hariharan, and K. Heller. Learning to detect sepsis with a multitask gaussian process rnn classifier. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 1174âĂŞ1182. JMLR.org, 2017.

[6] H. Ghritlahre and R. Prasad. Application of ANN technique to predict the performance of solar collector systems — a review. *Renewable and Sustainable Energy Reviews*, 84, 2018. doi: doi:10.1016/J.RSER.2018.01.001.

[7] N. Gunasekar, M. Mohanraj, and V. Velmurugan. Artificial neural network modeling of a photovoltaic-thermal evaporator of solar assisted heat pumps. *Energy*, 93, 2015. doi: 10.1016/J.ENERGY.2015.09.078.

[8] A. Khelifa, K. Touafek, H. Ben Moussa, and I. Tabet. Modeling and detailed study of hybrid photovoltaic thermal (PV/T) solar collector. *Solar Energy*, 135, 2016. doi: doi:10.1016/J.SOLENER.2016.05.048.

[9] S. C.-X. Li and B. Marlin. Classification of sparse and irregularly sampled time series with mixtures of expected gaussian kernels and random features. In *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*, UAI'15, page 484âĂŞ493, Arlington, Virginia, USA, 2015. AUAI Press.

[10] Z. C. Lipton, D. Kale, and R. Wetzel. Directly modeling missing data in sequences with rnns: Improved classification of clinical time series. In *Proceedings of the 1st Machine Learning for Healthcare Conference*, volume 56 of *Proceedings of Machine Learning Research*, pages 253–270, Northeastern University, Boston, MA, USA, 18-19 Aug 2016. PMLR.

[11] Z. Lu, T. K. Leen, Y. Huang, and D. Erdogmus. A reproducing kernel hilbert space framework for pairwise time series distances. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, page 624âĂŞ631, New York, NY, USA, 2008. Association for Computing Machinery.

[12] B. M. Marlin, D. C. Kale, R. G. Khemani, and R. C. Wetzel. Unsupervised pattern discovery in electronic health care data using probabilistic clustering models. In *Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium*, page 389âĂŞ398, New York, NY, USA, 2012. Association for Computing Machinery.

[13] G. Meramveliotakis, G. Kosmadakis, M. Pilou, and S. Karellas. Testing a flexible configuration of a solar-assisted heat pump with PVT collectors for domestic hot water production. In *Proceedings of EuroSun 2022: ISES and IEA SHC International Conference on Solar Energy for Buildings and Industry, Freiburg, Germany, January 2022*, 2022. doi: 10.18086/eurosun.2022.08.07.

[14] M. Mirzaei and M. Mohiabadi. Neural network modelling for accurate prediction of thermal efficiency of a flat plate solar collector working with nanofluids. *International Journal of Ambient Energy*, 42(2), 2021. doi: doi:10.1080/01430750.2018.1525576.

[15] D. Neil, M. Pfeiffer, and S.-C. Liu. Phased lstm: Accelerating recurrent network training for long or event-based sequences. In *Advances In Neural Information Processing Systems*, pages 3882–3890, 2016.

[16] T. Pham, T. Tran, D. Phung, and S. Venkatesh. Predicting healthcare trajectories from medical records: A deep learning approach. *Journal of Biomedical Informatics*, 69:218–229, 2017. ISSN 1532-0464.

[17] M. Pilou, G. Kosmadakis, G. Meramveliotakis, and A. Krikas. Towards a 100% renewable energy share for heating and cooling in office buildings with solar and geothermal energy. *Solar Energy Advances*, 2, 2022. doi: 10.1016/J.SEJA.2022.100020.

[18] D. Rojas, J. Beermann, S. Klein, and D. Reindl. Thermal performance testing of flat-plate collectors. *Solar Energy*, 82(8), Aug. 2009. doi: doi:10.1016/J.SOLENER.2008.02.001.

[19] Y. Rubanova, R. T. Q. Chen, and D. Duvenaud. Latent odes for irregularly-sampled time series. In *Advances In Neural Information Processing Systems*, volume 32, page 5320âĂŞ5330. Curran Associates, Inc., 2019.

[20] M. Sadeghzadeh, M. H. Ahmadi, M. Kahani, H. Sakhaeinia, H. Chaji, and L. Chen. Smart modeling by using artificial intelligent techniques on thermal performance of flatâĂŘplate solar collector using nanofluid. *Energy Science and Engineering*, 7(5), June 2019. doi: doi:10.1002/ese3.381.

[21] M. Schirmer, M. Eltayeb, S. Lessmann, and M. Rudolph. Modeling irregular time series with continuous recurrent units. In *Proceedings of the 39th International Conference on Machine Learning, 17-23 Jul 2022*, volume 162, 2022.

[22] S. N. Shukla and B. Marlin. Interpolation-prediction networks for irregularly sampled time series. In *International Conference on Learning Representations*, 2019.

[23] S. N. Shukla and B. Marlin. Multi-time attention networks for irregularly sampled time series. In *International Conference on Learning Representations*, 2021.

[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[25] Q. Wen, T. Zhou, C. Zhang, W. Chen, Z. Ma, J. Yan, and L. Sun. Transformers in time series: A survey. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, Survey Track. Macao, 19–25 August 2023*, 2023. doi: 10.24963/ijcai.2023/759.

[26] J. Yoon, W. R. Zame, and M. van der Schaar. Estimating missing data in temporal data streams using multi-directional recurrent neural networks. *IEEE Transactions on Biomedical Engineering*, 66:1477–1490, 2017.