CHALMERS
UNIVERSITY OF TECHNOLOGY

Hierarchical LSTM-Based Classification of Household Heating Types Using Measurement Data

Downloaded from: https://research.chalmers.se, 2025-07-01 09:46 UTC

Citation for the original published paper (version of record):
Fürst, K., Chen, P., Gu, I. (2024). Hierarchical LSTM-Based Classification of Household Heating Types Using Measurement Data. IEEE Transactions on Smart Grid, 15(2): 2261-2270.
http://dx.doi.org/10.1109/TSG.2023.3296020

N.B. When citing this work, cite the original published paper.

© 2024 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, or reuse of any copyrighted component of this work in other works.

(article starts on next page)

Hierarchical LSTM-based Classification of Household Heating Types using Measurement Data

Kristoffer Fürst, *Student Member, IEEE*, Peiyuan Chen, *Member, IEEE*, Irene Yu-Hua Gu, *Senior Member, IEEE*

***Abstract*—A lack of knowledge of the heating systems used by electricity consumers impedes distribution system operators in developing a sound grid upgrade plan and estimating potential demand flexibility from these consumers. The large-scale rollout of smart meters for electricity consumers provides an excellent opportunity to identify end users' heating types. This paper proposed a hierarchically structured deep-learning framework for identifying heating types of individual electricity consumers. The main contributions of the paper are: (a) We propose an effective framework based on long short-term memory (LSTM) that offers an effective automatic feature learning from sequential electricity consumption data and weather conditions. (b) We apply the proposed deep-learning architecture for household heating type classification which is among the first few successful reports on this application. We evaluate the performance using hourly measurement data collected over four years from one and two-family dwellings with either district heating, exhaust air heat pumps or direct electric heating as the heating type. Good performance was shown from the test results using the proposed framework, with an average test accuracy of 94.2%. Comparisons with four existing machine learning algorithms using handcrafted features and a single-layer LSTM-based deep-learning algorithm have shown marked improvement of the proposed method.**

***Index Terms*—classification algorithms, deep learning, energy consumption, energy measurement, feature extraction, heating systems, long short term memory, recurrent neural networks, smart meter**

I. Introduction

THE energy requirement for interior space heating and domestic hot water production for residential buildings constitutes an essential share of European energy demand [1]. By utilizing a building's thermal inertia, electric heating has considerable potential to supply demand flexibility to the electrical grid [1], [2]. For grid-planning decisions, the distribution system operators (DSOs) need a better knowledge of their customers' heating types. This is needed to make a more accurate peak load estimation for grid dimensioning purposes, especially in newly planned load areas. Furthermore, DSOs would be continuously informed on the hosting capacity of their grid if they could keep track of the change in the heating system of their consumers. Moreover, knowing the consumers' heating types offers the possibility to estimate the demand flexibility potential available in their grids. Thus, the improved knowledge of their customers' heating types can help the DSOs to reduce the safety margin needed during both the operation and planning phase of the grid in a controlled way and allows integration of more electric vehicles and solar PVs into their grid while fulfilling power quality requirements. However, there is no obligation for electricity consumers to notify their grid operator of energy efficiency measures or the types of heating systems. The large-scale rollout of smart meters for electricity consumers, plus publicly available data, provides a great opportunity for DSOs to determine end users' heating types automatically, without needing to contact them.

Many approaches have been used to identify different characteristics of electrical consumers, including statistical techniques [3], [4], conventional unsupervised machine learning (ML) [5]–[7] and supervised ML approaches using handcrafted features [8]–[11]. References [3] and [4] both developed Bayesian frameworks to predict the match between load profiles and their associated heating types. Reference [3] used time-series data measured directly from the heat pumps (not usually available). Furthermore, [4] was only able to identify a single heating type. Reference [5] used support vector regression (SVR) to extract energy signatures from electricity consumption and outdoor air temperature measurements, followed by k-means clustering to find the signatures of heating-type clusters. Reference [6] used fuzzy clustering to group load profiles extracted from daily load profiles. Both [5] and [6] used handcrafted features defined by human experts requiring prior knowledge of consumer behaviors. Furthermore, cluster-based approaches need post-analysis and interpretation to identify the types of heating systems associated with the clusters. Using supervised learning, [8] employed support vector machines (SVM) to classify household heating types using features from smart meter measurements. Reference [9] used random forest (RF) and SVM with different sets of features extracted from smart-meter for household classification. The work in [10] and [11] further extended a set of pre-determined features in [9] to classify heating types. However, these conventional ML approaches extract features defined by human experts (i.e., handcrafted features). This may be a challenging task as it needs highly skilled human experts. Incomplete knowledge from human experts may lead to important information being overlooked, and thus less adequate heating type classification.

Recently, some deep-learning (DL) approaches were studied for automatic feature learning [12]–[17]. Long short-term memory (LSTM) is a DL approach suitable for automatic feature-learning from data sequences [18], [19]. It has been used in a variety of areas, such as natural language-processing [17], as well as in electricity system applications (power grid impedance estimation [12], residential load forecasting [13], and consumer type classification [14]–[16]). Although [14]

The project was financed by The Swedish Energy Agency under the *Digitalisering möjliggör energi- och klimatomställningen* program.

Fürst, K., Chen, P. and Gu, I. Y. H are with Dept. of Electrical Engineering, Chalmers University of Technology, Gothenburg, Sweden (e-mail: kristoffer.furst@chalmers.se)

used a combination of handcrafted features, LSTM and conventional supervised ML to classify such household appliances as heat pumps, dishwashers and TVs, it was only applied to specific household appliances using synthetic high-resolution profiles. Such an approach cannot be used directly to identify heating types at a large scale as measurements from individual appliances are not distinguishable from smart meter data collected at a household level. References [15] used a CNN-LSTM DL network for electricity theft detection. Similarly, [16] used a CNN-LSTM DL network for dryer identification. The convolutional neural network (CNN) layer was first used on segmented data for feature extraction before the LSTM was applied. For handling long data segments, we propose adding an additional LSTM layer instead of a CNN layer as LSTM is suitable for feature learning from data sequences. In the area of natural language processing, [17] resorted to a hierarchical LSTM-based approach. Two LSTM layers were used to classify a text document where the first layer captured the short-term dependency in a sentence, followed by the second layer that captured the long-term dependency of the document. However, the feasibility of such a hierarchical LSTM-based approach was not tested on classifying household heating types using smart meter measurements.

Motivated by the above, we propose a novel, hierarchical, LSTM-based DL framework specifically for the application of heating type classification. This aim is to identify consumer heating types by automatically learning the features of household consumption data gathered by smart meters, and the corresponding weather data. In particular, an LSTM-based framework is proposed for feature-learning and classifying multiple heating types. The automatic feature learning is convenient as the classification does not rely on human expert knowledge which may be incomplete or unavailable. To the best of our knowledge, this is the first reported DL-based algorithm that offers high performance for successfully classifying consumer heating types based on smart meter measurements and weather data. The main contributions of the paper are:

- proposing a novel hierarchical LSTM-based DL architecture for classifying consumer heating system types. It uses measurement data gathered by smart meters and corresponding weather data.
- developing a hierarchical LSTM-based classifier end-to-end by first extracting features automatically followed by identification of households' heating types.

Experiments and performance evaluations have been conducted on hourly smart meter measurements and weather data sequences over four years. Our case study results demonstrate that the proposed framework has successfully identified electricity consumers' heating types. In addition, a comparison has been made with four existing ML algorithms and a single-layer LSTM-based DL algorithm, where the proposed framework classifier we developed has shown improved classification performance.

The remainder of this paper is organized as follows. The proposed framework is described in Section II. A brief review of the vanilla LSTM is followed by an overview of the proposed framework. Next, there is a detailed description of several important modules, including the hierarchical LSTM network for automatic learning heating-type features, and an end-to-end network for classifying consumer heating types. Section III describes the experiment setup and shows the test results and performance evaluation, accompanied by a comparison and some discussion. Section IV further discusses limitations and power system planning applications. The paper concludes with Section V.

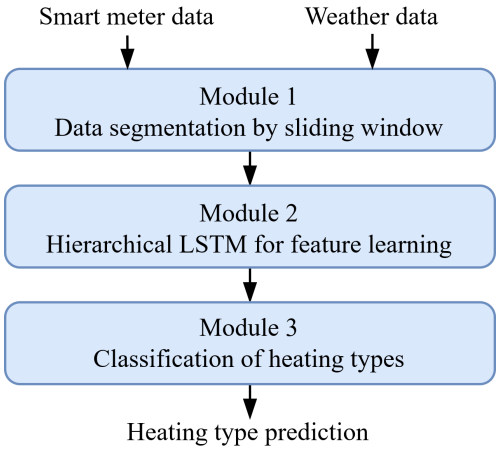

Fig. 1. Block diagram of the proposed deep-learning framework.

II. Proposed Framework

A. Overview of the Proposed Framework

Essentially, the proposed framework uses a supervised DL network to classify the heating types of individual households. The proposed approach uses a hierarchical LSTM network architecture to *automatically* learn the discriminating features of consumers' heating types, based on smart meter data and corresponding weather data. Fig. 1 shows the block diagram of the proposed framework, consisting of the following modules: Module 1 - Data segmentation by sliding window; Module 2 - Hierarchical LSTM layers for feature learning; and Module 3 - Classification of heating types. The details of the framework are described in the following subsections.

B. Long Short-Term Memory - a Brief Review

For the sake of mathematical and notational convenience, this section gives a brief overview of long short-term memory (LSTM). See [18] and [19] for further details. LSTM is a recurrent neural network with memory units capable of learning long-term dynamics. The core of the network is the cell state $\boldsymbol{c}$ and hidden state $\boldsymbol{h}$ which encodes the input sequence $\boldsymbol{x}$. The cell state serves as the long-term memory of the network and the hidden state as the working memory. There are many variants of LSTM, but here we only review the basic vanilla LSTM shown in Fig. 2, where the hidden states and cell states are regulated by *three* gates: a forget gate $\boldsymbol{f}$, an update gate $\boldsymbol{i}$, and an output gate $\boldsymbol{o}$. The hidden states and cell states at time t are updated by:

$$\boldsymbol{f}^t = \sigma\left(\boldsymbol{W}_{f,h}\boldsymbol{h}^{t-1} + \boldsymbol{W}_{f,x}\boldsymbol{x}^t + \boldsymbol{b}_f\right) \tag{1a}$$
$$\boldsymbol{i}^t = \sigma\left(\boldsymbol{W}_{i,h}\boldsymbol{h}^{t-1} + \boldsymbol{W}_{i,x}\boldsymbol{x}^t + \boldsymbol{b}_i\right) \tag{1b}$$
$$\tilde{\boldsymbol{c}}^t = \tanh\left(\boldsymbol{W}_{c,h}\boldsymbol{h}^{t-1} + \boldsymbol{W}_{c,x}\boldsymbol{x}^t + \boldsymbol{b}_c\right) \tag{1c}$$
$$\boldsymbol{c}^t = \boldsymbol{f}^t \odot \boldsymbol{c}^{t-1} + \boldsymbol{i}^t \odot \tilde{\boldsymbol{c}}^t \tag{1d}$$
$$\boldsymbol{o}^t = \sigma\left(\boldsymbol{W}_{o,h}\boldsymbol{h}^{t-1} + \boldsymbol{W}_{o,x}\boldsymbol{x}^t + \boldsymbol{b}_o\right) \tag{1e}$$
$$\boldsymbol{h}^t = \boldsymbol{o}^t \odot \tanh\left(\boldsymbol{c}^t\right) \tag{1f}$$

where $\boldsymbol{x}^t$ denotes the input at time t, $\boldsymbol{W}$ the weight matrices, $\boldsymbol{b}$ the bias, σ the logistic sigmoid function, and $\odot$ the element-wise product of vectors. The length of the state vectors is determined by the number of hidden units n, whereas the size of the weights and biases are determined by the number of hidden units and the length of the input vector N (defined before training the model). The length of the state vectors are $\boldsymbol{h}^t \in \mathbb{R}^{n\times 1}$ and $\boldsymbol{c}^t \in \mathbb{R}^{n\times 1}$, the weight matrices corresponding to the hidden state $\boldsymbol{W}_h \in \mathbb{R}^{n\times n}$, the weight matrices corresponding to the input $\boldsymbol{W}_x \in \mathbb{R}^{n\times N}$ and the biases $\boldsymbol{b} \in \mathbb{R}^{n\times 1}$. The complexity of the LSTM network is thereby dependent on the number of hidden units and the input size, If the number of hidden units and/or the input size is larger, more parameters will need to be trained.

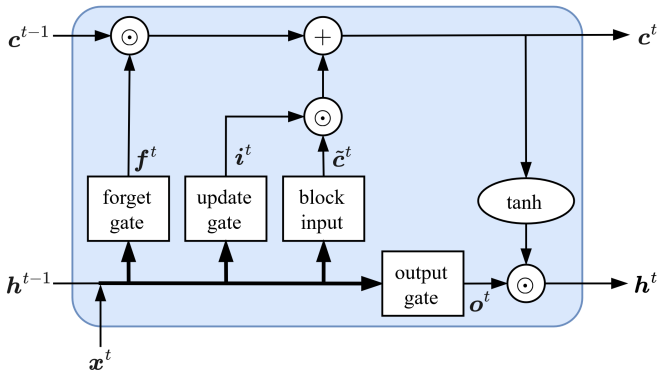

Fig. 2. A typical vanilla long short-term memory (LSTM) unit. In the figure, $\odot$ is the element-wise product of vectors, and $+$ is the addition of vectors.

C. Description of the Proposed Framework

This subsection gives a detailed description of the three modules in the proposed framework.

1) Data segmentation by using a sliding window: The rationale for applying data segmentation is to break a very long sequence of measurement data across several years. It is equivalent to using a sliding data window of length T without overlap. First, for *each* consumer, one smart meter data sequence and N_P weather data sequences were collected over the same time interval and using nearby geographical area. Each data sequence contains L measurement samples. The smart meter sequence is denoted as: $\boldsymbol{x} = [x^1, x^2, \ldots, x^L]$ and the sequences for the pth weather sequences as $\boldsymbol{v}_p = [v_p^1, v_p^2, \ldots, v_p^L]$. Our dataset used hourly data sampling. However, the framework is not limited to hourly data values. The following steps describe the data preparation steps and sequence segmentation.

i) Normalize the smart meter data sequence $\boldsymbol{x}$ by its mean m_x and standard deviation s_x to highlight the electricity consumption shape, i.e., $\tilde{x}^l = (x^l - m_x)/s_x$ for all values $l = 1, \ldots, L$.

ii) Combine the normalized smart meter sequence $\tilde{\boldsymbol{x}} = (\tilde{x}^1, \tilde{x}^2, \ldots, \tilde{x}^L)$ and N_P weather sequences (not normalized) to form a sequence of vectors $\boldsymbol{z}$:

$$\boldsymbol{z} = \begin{bmatrix} \tilde{\boldsymbol{x}} \\ \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \\ \vdots \\ \boldsymbol{v}_{N_P} \end{bmatrix} \in \mathbb{R}^{(1+N_P)\times L} \tag{2}$$

iii) Rearrange the sequence of vectors $\boldsymbol{z}$ into a new sequence of vectors $\boldsymbol{Z}$, such that the rows correspond to the smart meter and weather values in a 24-hour period:

$$\boldsymbol{Z} = \begin{bmatrix} \tilde{x}^{1,1}, \ldots, \tilde{x}^{D,1} \\ \tilde{x}^{1,2}, \ldots, \tilde{x}^{D,2} \\ \vdots \\ \tilde{x}^{1,24}, \ldots, \tilde{x}^{D,24} \\ v_1^{1,1}, \ldots, v_1^{D,1} \\ \vdots \\ v_{N_p}^{1,24}, \ldots, v_{N_p}^{D,24} \end{bmatrix} \in \mathbb{R}^{24(1+N_P)\times D} \tag{3}$$

where $\tilde{x}^{d,t}$ and $v_p^{d,t}$ corresponds to day d and hour t in the sequence, $d = 1, \ldots, D$ and $t = 1, \ldots, 24$. The new sequence is now $D = L/24$ days long. Such a rearrangement increases the size of the weight matrices $\boldsymbol{W}$. However, experiments have shown that better classification performance was obtained when using $\boldsymbol{Z}$ with a shorter sequence length instead of $\boldsymbol{z}$, which has a longer sequence length.

iv) Add the mean and standard deviation into the feature vector $\boldsymbol{Z}$, as they also play an important role in characterizing the heating types. This means using the mean m_x and standard deviation s_x from the original smart meter sequence $\boldsymbol{x}$ to construct two additional sequences of length D, $\boldsymbol{m}_x = [m_x, \ldots, m_x] \in \mathbb{R}^{1\times D}$ and $\boldsymbol{s}_x = [s_x, \ldots, s_x] \in \mathbb{R}^{1\times D}$. The result is a sequence of vectors $\boldsymbol{S}$ consisting of $N = 24(1 + N_P) + 2$ features,

$$\boldsymbol{S} = \begin{bmatrix} \boldsymbol{Z} \\ \boldsymbol{m}_x \\ \boldsymbol{s}_x \end{bmatrix} \in \mathbb{R}^{N\times D} \tag{4}$$

v) Normalize each row (feature component) of the sequence of vectors $\boldsymbol{S}$ by using its mean and standard deviation across all customers in the training set. Thus, each feature component in the normalized sequence of vectors $\tilde{\boldsymbol{S}}$ lies within the same scale.

vi) Segment $\tilde{\boldsymbol{S}}$ into K segments, $(\tilde{\boldsymbol{S}}_1, \tilde{\boldsymbol{S}}_2, \ldots, \tilde{\boldsymbol{S}}_K)$, using a non-overlapping sliding window of length T. The number of segments is defined by $K = D/T$.

This leads to a total of K segments, each one a sequence of vectors consisting of N rows (feature components) and T columns (segment length). Although reducing the input data length was one of the reasons for adopting a hierarchical

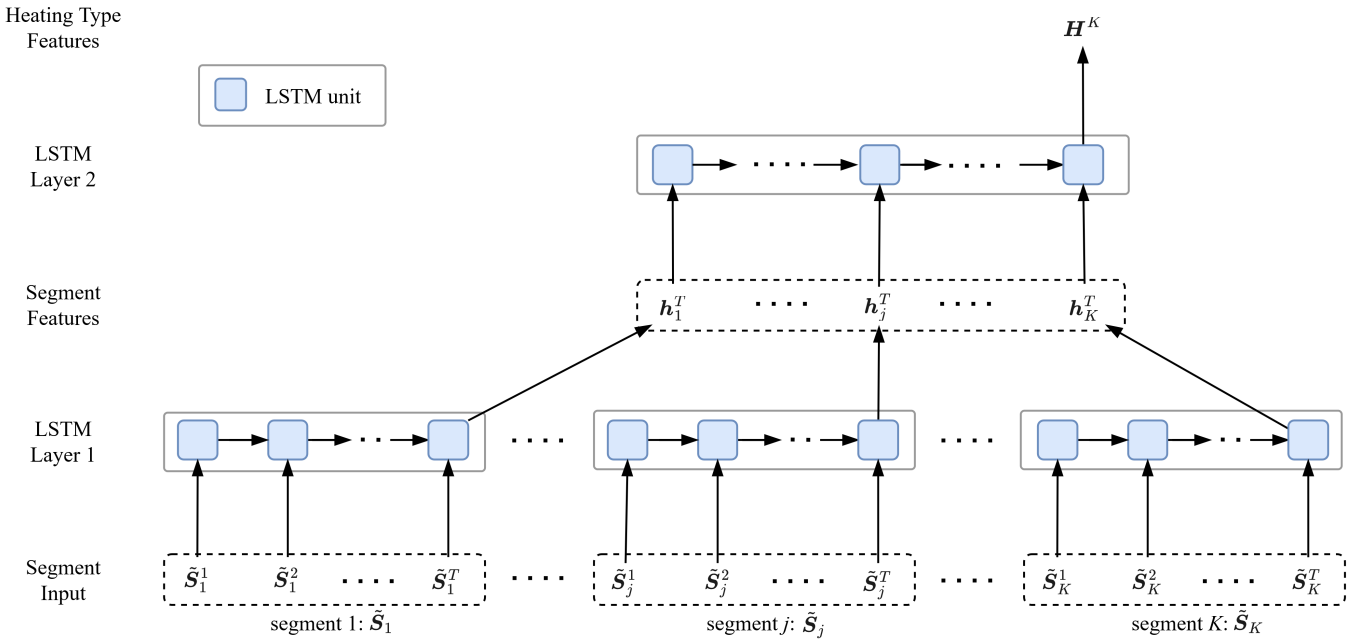

Fig. 3. Module 2 of the proposed framework: automatic feature extraction using a hierarchical, two-layer LSTM. The parameters of the LSTM units are shared across all time steps and all segments for each respective LSTM layer.

structure, an LSTM could fail to capture the dynamic if the segment length T were too short. A segment length of $T = 30$ days was chosen for this work and performed well on our dataset.

2) Hierarchical LSTM *for Feature Learning:* The proposed hierarchical LSTM network consists of two LSTM layers, as shown in Fig. 3. Each LSTM layer applies *many-to-one* input to output. First, each segment ($j = 1$ to K) is fed into the nodes of the first LSTM layer. For the jth input segment $\tilde{\boldsymbol{S}}_j$, the first LSTM layer with n_1 hidden units outputs a corresponding hidden-state vector $\boldsymbol{h}_j^T \in \mathbb{R}^{n_1 \times 1}$. This vector $\boldsymbol{h}_j^T$ represents the characteristics of segment $\tilde{\boldsymbol{S}}_j$. These hidden-state outputs are then used to form a new sequence of vectors $[\boldsymbol{h}_1^T, \boldsymbol{h}_2^T, \ldots, \boldsymbol{h}_K^T] \in \mathbb{R}^{n_1 \times K}$. As shown in Fig. 3, the new sequence is fed into the second LSTM layer with n_2 hidden units. The hidden state $\boldsymbol{H}^K \in \mathbb{R}^{n_2 \times 1}$ output from the second LSTM layer is then used as the *final feature vector* for classifying heating types. Note that the same set of parameters (weights and biases) are used for all LSTM units in each *respective* LSTM layer. Note also that the LSTM layers in the function of DL Keras library [20] imply a three-dimensional tensor (batch size, number of time steps, number of features), whereas the first LSTM layer of the proposed hierarchical framework consists of a four-dimensional tensor (batch size, number of time steps in layer 2 (K), number of time steps in layer 1 (T), and number of features (N)). In the DL Keras library, this is implemented by using an LSTM layer with a time-distributed wrapper [20].

3) Classification of Heating Types: The final feature vector $\boldsymbol{H}^K$ (obtained from the output of the second LSTM layer) is then fed into the classifier to determine an electricity consumer's heating type. As shown in Fig. 4, the classifier is formed by a fully connected network with C neurons, in which C equals the number of heating types considered. The feature vector $\boldsymbol{H}^K = [H_1^K, H_2^K, \ldots, H_{n_2}^K]^\top$ is then feedforward through the network, with softmax applied to the network output to obtain the class probability for individual classes, $\hat{\boldsymbol{y}} = [\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_C]^\top$. Finally, an electricity consumer's heating type is classified according to the type with the highest probability, i.e. $\hat{y}^* = \arg\max_c(\hat{\boldsymbol{y}})$.

D. Training and Classification

1) Training: During the training process, the network uses supervised learning to try and learn a set of parameters. The network's coefficients (weights and biases) are optimized by minimizing categorical cross-entropy among the training samples. In other words, the difference between the true and estimated target vector is minimized. The loss function L for each training sample is defined as:

$$L(\boldsymbol{y}, \hat{\boldsymbol{y}}) = -\sum_{c=1}^{C} y_c \cdot \log \hat{y}_c, \tag{5}$$

where y_c takes the value 0 or 1 to indicate the correct class, and $\hat{y}_c$ is the estimated probability (softmax output) of class c. Once the network has been trained, the parameters are frozen for use in the classification process.

2) Classification Process: Once the coefficients from the training process have been obtained, the trained network can be test-run to classify a consumer's heating type. That is, using consumers whose data have not been seen by the classifier as the input for predicting their heating types.

3) Partition between Training and Test Sub-Datasets: Note that training and test data subsets should be split according to different consumers, i.e., each consumer's data are used either for training or testing. Despite superior test performance may

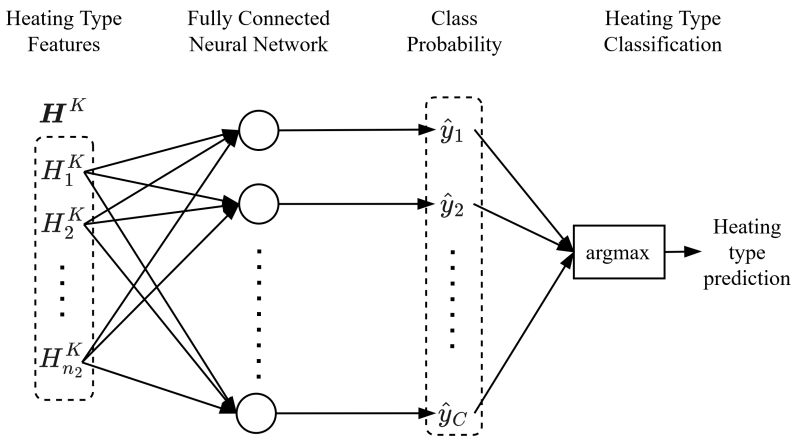

Fig. 4. Module 3 of the proposed framework: classification. The fully connected neural network includes a softmax activation function.

be obtained by partitioning each consumer's data and using different parts for the training and testing. This is because different parts of data from the same consumer are highly correlated. However, the performance may drop significantly when the classifier is used to classify new consumers. Thus, we adopt strict consumer-based data partitioning between the training and test subsets.

III. Results and Performance Comparison

A. Setup of Experiments

1) Data Description: The dataset consists of hourly smart meter measurements from a city in Sweden, taken over four years (2016-2019). The labels were collected from the energy declarations of buildings [21]. In this study, we selected one and two-family dwellings with only one smart meter. Three of the most common heating types were selected: district heating district heating (DH), exhaust air heat pump (EAHP), and direct electric heating (DEH). Outdoor weather data with a spatial resolution of 2.5×2.5 km was also included, as the weather influences the indoor climate, and can shift the operating point of heating systems. Over the same four years, hourly values were collected for outdoor air temperature, wind speed, solar irradiance, and relative humidity [22], [23]. The consumer data recordings were then synchronized with the weather data for the closest grid point, with a total of 40 grid points used.

2) Dataset Partition: The dataset was partitioned into training/validation/test subsets with a $60/20/20$ ratio, according to the number of consumers/households. That is, data sequences from individual consumers were used *only* in training, validation, or testing. Table I summarizes the size of the subsets for each heating type studied.

3) Pre-Processing: Due to some changes in the sampling frequency (faulty communication), measurement data with a difference of less than 0.1 kWh/h between two adjacent hours for 20 consecutive hours were filtered out. However, data from only two successive missing values were interpolated linearly. Lastly, a masking layer was added to the network to deal with missing values. In other words, time steps with missing values were skipped. This made it possible to classify consumers using various data lengths.

TABLE I
Dataset partition: number of individual customers used (exclusively) in each subset.

Heating type [a]	Train	Validation	Test	Total
DH	1264(33%)	422(11%)	421(11%)	2107(54%)
EAHP	380(10%)	126(3%)	127(3%)	633(16%)
DEH	686(18%)	229(6%)	229(6%)	1144(29%)
Total	2330(60%)	777(20%)	777(20%)	3884(100%)

[a] DH: district heating; EAHP: exhaust air heat pump; DEH: direct electric heating

4) Network Hyperparameters: The *number of hidden units* in our experiments was determined by performing a grid search over numbers of hidden units, including $4, 8, 16,$ and 32 for each LSTM layer. The combination with the lowest *validation* loss (using the validation set) was selected. We used an Adam *optimizer* [24] with a learning rate of $1e^{-3}$ and $\beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 1e^{-7}$. A reduce-on-plateau *learning rate scheduler* was used to reduce the learning rate by 10% if the validation loss had not been reduced over five epochs. An *early-stopping* strategy was used where the learning was terminated if there had been no gain for 50 epochs. The network coefficients corresponding to the lowest validation loss were allocated to the final classifier. A *mini-batch* size of 128 was selected for training.

Furthermore, one can see from Table I that the number of consumers in the three types of heating was unbalanced. One main reason is that some heating technologies were more popular at different years of construction, as well as the availability, potential and investment of the different technologies. To manage the imbalanced dataset, the training loss was inversely proportional to the class frequencies of the training set. Oversampling or undersampling techniques have also been evaluated to make the data more balanced in different classes. However, they did not show improved classification performance in our preliminary tests. Other techniques, such as GANs (generative adversarial networks) [25] could be useful to enrich the training dataset by adding synthetic data with

the same distribution of that class. However, it is beyond the scope of this paper.

5) Criteria for Performance Evaluation: Precision, recall, and F_1-score were used to evaluate the performance of each heating type class c, and the accuracy of the test set was used to measure the total classification accuracy, defined as:

$$\text{Precision}_c = \frac{TP_c}{TP_c + FP_c}, \tag{6}$$

$$\text{Recall}_c = \frac{TP_c}{TP_c + FN_c}, \tag{7}$$

$$\text{F}_{1,c} = 2\frac{\text{Precision}_c \cdot \text{Recall}_c}{\text{Precision}_c + \text{Recall}_c}, \tag{8}$$

$$\text{Accuracy} = \frac{\sum_{c=1}^{C} TP_c}{\sum_{c=1}^{C} TP_c + FP_c}, \tag{9}$$

where TP_c is the true positive number of class c (in which consumers are correctly classified), FP_c is the false positive number for class c (in which consumers that do not belong to class c are wrongly classified as class c), and FN_c is the false negative number (in which consumers belonging to class c are misclassified).

6) Implementation: The proposed framework was programmed whose performance was tested on a PC with a 3.7 GHz Intel i9 processor with 128 GB of memory, and an NVIDIA GeForce RTX 3080 GPU with 10 GB of memory. The network was implemented, trained, and evaluated, using the Keras library's basic functions (including the Keras layers *LSTM, Dense, TimeDistributed*, and *Masking*) [20]. Furthermore, the network was trained using the computer's GPU to speed up the computation, though it could also be trained using the CPU.

B. Results and Discussion

Ten random dataset partitions were used in the experiments, with the network re-trained for each new one and the performance re-tested.

1) Overall Performance: Tests were conducted to demonstrate the proposed framework's effectiveness, and the performance of unseen consumers was calculated. A sliding window of $T = 30$ days was used for data segmentation of training, validation, and test data alike. Table II shows the classification performance by using smart meter measurements and their corresponding outdoor air temperature, solar irradiance, relative humidity, and wind speed. The results showed an average accuracy of 94.2%. Furthermore, the classifier showed better performance in identifying consumers with DH than those with electric heating, including both EAHP and DEH. One reason could be that consumers with DH generally consumed less electricity during the heating period than similar buildings with electric heating sources. The confusion matrix obtained from the test set (Table III) further illustrates the performance of individual classes. It indicates that the classifier has some difficulty distinguishing between the two electric heating types. This could be due to their similarities; both EAHP and DEH have increased electricity usage during the heating period, albeit increasing by different amounts.

TABLE II
TEST PERFORMANCE OF THE PROPOSED FRAMEWORK WITH DATA SEGMENTS OF $T = 30$ DAYS. ALL PERFORMANCE VALUES IN THE TABLE ARE AVERAGED ± STANDARD DEVIATION OVER 10 RUNS*.

Heating type[a]	Precision (%)	Recall (%)	F_1−score (%)	Total accuracy (%)
DH	98.5 ± 0.6	97.7 ± 0.8	98.1 ± 0.4	
EAHP	86.3 ± 3.7	85.5 ± 3.8	85.8 ± 2.6	94.2 ± 0.9
DEH	90.9 ± 2.0	92.6 ± 2.5	91.7 ± 1.2	

* By dataset re-partitions, followed by re-training and re-testing, [a] DH: district heating; EAHP: exhaust air heat pump; DEH: direct electric heating

TABLE III
CONFUSION MATRIX: TEST PERFORMANCE USING PROPOSED FRAMEWORK WITH A ROLLING WINDOW OF $T = 30$ DAYS. VALUES ARE AVERAGED ± STANDARD DEVIATION OVER 10 RUNS* NORMALIZED TO THE CLASS SIZE.

		Predicted heating type (%)		
		DH[a]	EAHP[b]	DEH[c]
Actual[d] heating type (%)	DH	97.7 ± 0.8	1.0 ± 0.5	1.3 ± 0.5
	EAHP	2.0 ± 1.1	85.5 ± 3.8	12.4 ± 3.6
	DEH	1.7 ± 0.9	5.8 ± 1.8	92.6 ± 2.5

* By dataset re-partitions, followed by re-training and re-testing, [a] DH: district heating, [b] EAHP: exhaust air heat pump, [c] DEH: direct electric heating, [d] the actual heating-type label indicates the heating type declared *at the time* the energy declaration was issued.

To further visualize the class separability of the automatically extracted features, the high-dimensional feature space is reduced to two dimensions using t-distributed stochastic neighbor embedding (t-SNE) [26]. Fig. 5 shows this two-dimensional feature space, with each sub-figure corresponding to one of the ten runs of the experiment. Three distinct clusters can be observed, each corresponding to one heating type. It is also evident that a small number of consumers were wrongly classified as another type. This relates to the minor classification errors seen in Table II, which is further analyzed below.

2) Error analysis: An error analysis was performed to further analyze the misclassifications. This was done by evaluating the classification errors in relation to the heated area and the age of the building (Fig. 6), and to the geographical location of consumers (Fig. 7). It is worth mentioning that such analysis is only a hint of the possible causes of errors as the information on building and behavior characteristics is incomplete. Fig. 6 (left) shows that for households with EAHP or DEH, there was only a small difference between the heated area for the correct and incorrect classified samples. However, there was a larger difference for buildings with DH, where the correct classified samples had a median of 142 m^2, whereas the incorrect ones had a larger heated area of 174 m^2. The confusion matrix in Table III shows that approximately half of the misclassified DH was classified as EAHP and half as DEH (1.0% as EAHP, 1.3% as DEH). The potential reasons for the classification error may include: i) large buildings with

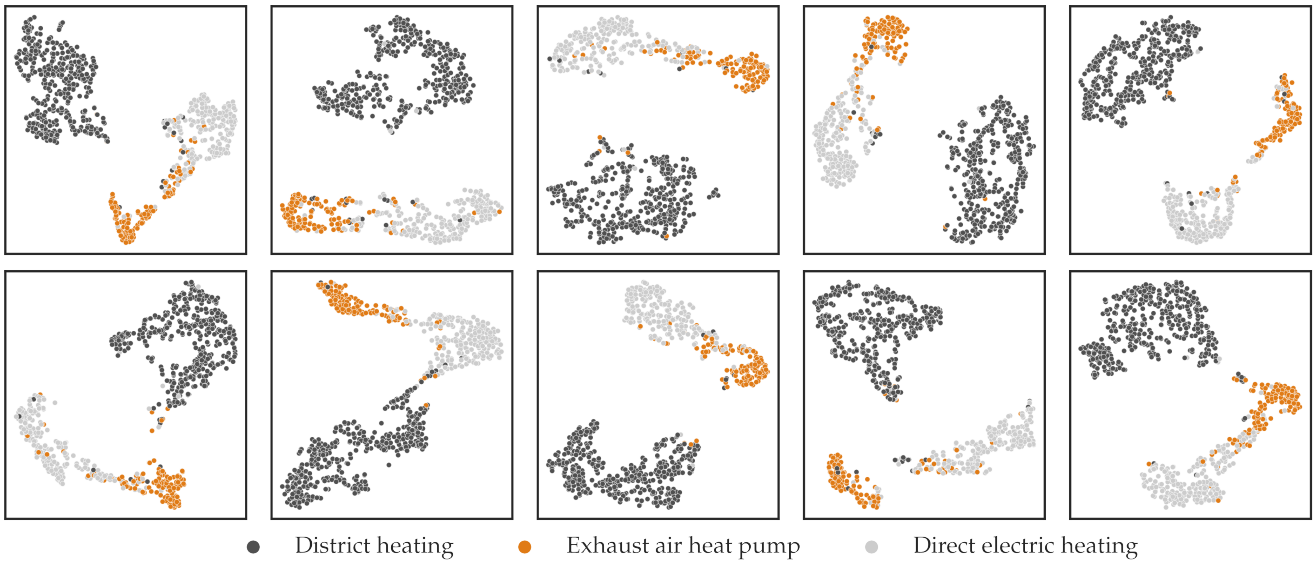

Fig. 5. Plots of the heating-type features learned from the proposed hierarchical LSTM layers, with four years' worth of data segmented over 30 days. The Scikit-learn [27] function t-SNE was used for these plots. These sub-figures show the results in ten runs, in which the dataset was re-partitioned, then re-trained and re-tested.

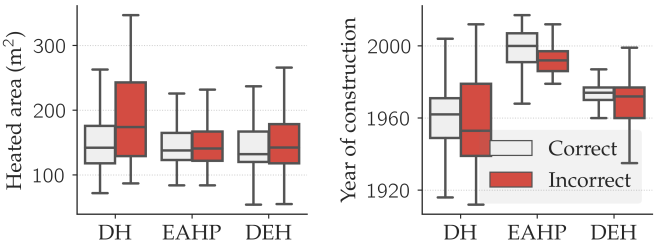

Fig. 6. Classification performance as a function of left) heated area and right) year of construction, where DH = district heating, EAHP = exhaust air heat pump, and DEH = direct electric heating. The boxplot was based on the test performance over all ten dataset re-partitions.

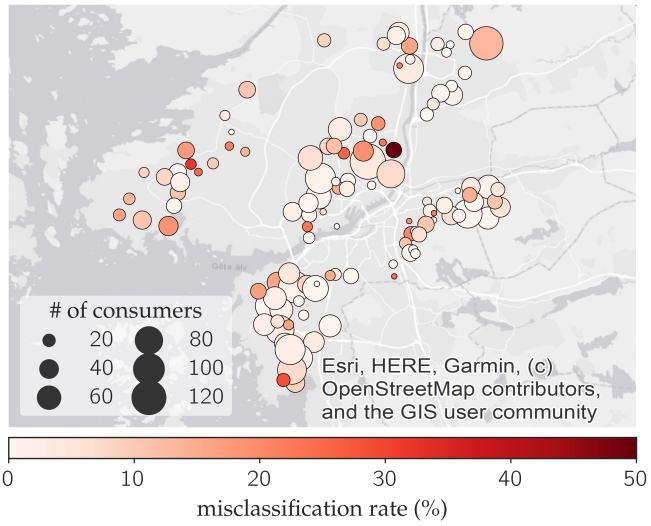

Fig. 7. Misclassification rate per zip code, based on the test performance over all ten dataset re-partitions. (This map was created using ArcGIS® software by Esri. ArcGIS® and ArcMap™ are the intellectual property of Esri and are used herein under license. Copyright © Esri. All rights reserved. For more information about Esri® software, please visit www.esri.com)

DH could have a higher electricity consumption which the DL model did not capture; ii) the buildings had supplementary electric heating which was not included in the dataset, leading to an increased weather-dependent electricity consumption; iii) buildings that have changed heating system had, in general, a higher heated area. This would however need to be verified by contacting the consumers individually.

Furthermore, Fig. 6 (right) shows a large difference between the age of the buildings and the different heating types where DH was seen in both older and newer houses, EAHP mainly in houses from the 80s and forward, and DEH mainly in houses from the 70s. Even though the age of the building could be included in the model, there is a risk that the classifier could show a bias. Moreover, the median of incorrect classified samples was on average eight years older for houses with EAHP than the correct ones. The confusion matrix in Table III shows that most of the misclassified EAHP samples were classified as DEH (12.4% as DEH, 2.0% as DH). This could indicate that an older heat pump with a lower coefficient of performance (COP) and/or lower rated capacity was installed, which is more dependent on additional heat from for instance the heat pump's immersion heater, thus using more electricity as a similar building with a modern and larger EAHP. This however also needs to be verified by contacting the consumers individually. Similarly, houses with DH which were correctly classified had a median that was on average eight years older than the incorrect samples. This could also be due to older buildings with poor isolation in need of supplementary electric heating. However, the result also showed a large spread. Even though DEH did not show a large difference in the median,

TABLE IV
IMPACT OF TEST PERFORMANCE WHEN DIFFERENT WEATHER VARIABLES WERE ADDED TO THE SMART METER DATA USING THE PROPOSED FRAMEWORK. IN ALL EXPERIMENTS, A DATA SEGMENT LENGTH OF $T = 30$ DAYS WAS USED. ALL PERFORMANCE VALUES WERE AVERAGED $\pm$ STANDARD DEVIATION OVER 10 RUNS*. THE BEST PERFORMANCE APPEARS IN BOLD.

Items of data used[a]	Precision (%)			Recall (%)			F_1−score (%)			Accuracy (%)
	DH[b]	EAHP[c]	DEH[d]	DH	EAHP	DEH	DH	EAHP	DEH	Total
SM	97.9 ± 0.9	78.0 ± 5.2	89.6 ± 3.0	96.0 ± 1.0	82.6 ± 4.8	89.7 ± 3.0	97.0 ± 0.6	80.2 ± 4.0	89.6 ± 2.0	92.0 ± 1.4
SM+T	98.4 ± 0.7	82.7 ± 4.9	90.5 ± 2.1	97.3 ± 1.0	84.6 ± 3.3	90.7 ± 3.4	97.8 ± 0.6	83.5 ± 2.0	90.5 ± 1.1	93.3 ± 0.8
SM+S	97.6 ± 1.0	81.0 ± 3.9	90.4 ± 2.1	96.6 ± 0.6	83.6 ± 3.3	90.4 ± 3.0	97.1 ± 0.4	82.2 ± 3.1	90.4 ± 1.7	92.7 ± 1.2
SM+H	97.9 ± 1.0	83.8 ± 4.5	90.8 ± 1.9	96.9 ± 1.0	85.2 ± 3.9	91.4 ± 2.3	97.4 ± 0.5	84.3 ± 2.1	91.1 ± 1.0	93.4 ± 0.8
SM+W	98.0 ± 1.0	83.3 ± 5.1	90.1 ± 2.2	97.3 ± 1.2	83.6 ± 5.2	90.9 ± 2.4	97.6 ± 0.9	83.3 ± 3.5	90.5 ± 1.4	93.2 ± 1.3
SM+T+S+H+W	$\mathbf{98.5 \pm 0.6}$	$\mathbf{86.3 \pm 3.7}$	$\mathbf{90.9 \pm 2.0}$	$\mathbf{97.7 \pm 0.8}$	$\mathbf{85.5 \pm 3.8}$	$\mathbf{92.6 \pm 2.5}$	$\mathbf{98.1 \pm 0.4}$	$\mathbf{85.5 \pm 2.6}$	$\mathbf{91.7 \pm 1.2}$	$\mathbf{94.2 \pm 0.9}$

* By dataset re-partitions, followed by re-training and re-testing,[a] SM: smart meter; T: outdoor air temperature; S: solar irradiance; H: relative humidity; W: wind speed, [b] DH = district heating, [c] EAHP = exhaust air heat pump, [d] DEH = direct electric heating

it still misclassified 5.8% as EAHP (and 1.7% as DH). This could be due to the training process and decision boundary being affected by some EAHP behaving similarly to DEH, or that some consumers have changed class (upgraded their system).

Fig. 7 shows that some areas have a higher share of misclassified samples. This indicates that there could be a geographical bias which could be further analyzed. For instance, in the area with the highest share of misclassifications (see Fig. 7), buildings with DH were misclassified as EAHP. Through manual investigation, we found that the buildings in this area had collectively changed the heating system from DH to EAHP [28]. On one hand, this shows that the information from the energy declaration used for the training process may be outdated since issued. This can affect the training process and the evaluation of the model. On the other hand, it also shows that the model was able to identify whether a consumer had changed the heating system.

3) Impact of Adding Weather Data: Test performance in various scenarios was evaluated to examine the impact of adding different types of weather measurement data to the smart meter measurements. Table IV shows the test performance upon the addition of outdoor weather data variables (i.e., outdoor air temperature, relative humidity, global irradiance, and wind speed) *individually*. The results show that adding weather data increased the test accuracy from 92.0% to between 92.7% and 93.4%. Further, these weather variables were complementary, and the highest average test accuracy (94.2%) was obtained when all four weather data variables were used in combination with smart meter data. In particular, a large improvement can be seen for the precision of EAHP which increased from 78.0% to 86.3%. The increased performance of the F1-score for all three heating types also shows that adding weather data increases the separability between all three classes.

4) Impact of Training Set Size: DL methods require a large set of training samples. A sensitivity analysis was performed to evaluate how the performance was affected by the size of the training data set. The size of the training subset (randomly selected from the training set) varied from 50 customers to the full training set sized 2330 customers. Fig. 8 shows the performance on the test set versus the classifier trained by different-sized training sets. The general trend was that the test performance improved as the size of the training set increased. In particular, there was a steep increase at the beginning of the graph. Furthermore, the results indicate that even higher performance could be obtained if more data were used. Note that the stochastic variations of the curve were due to the random subset sampling and the random nature of the ML implementation (e.g., dropout, batch partition).

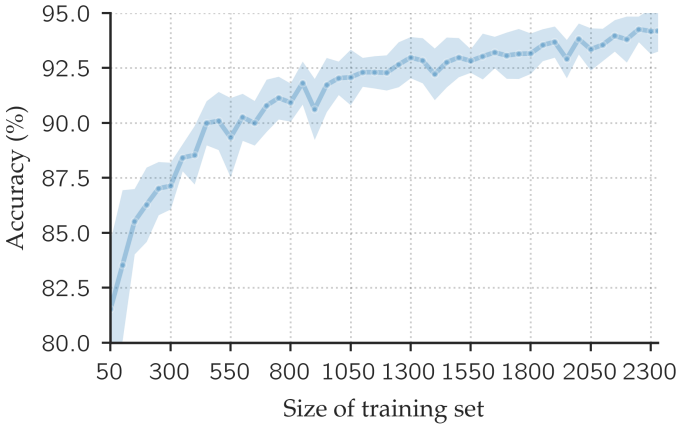

Fig. 8. Test accuracy as a function of training set size. The line and shaded area show the average $\pm$ standard deviation of the test accuracy over all ten dataset re-partitions.

C. Comparison to other Machine Learning Algorithms

The classifier obtained from the proposed framework was compared to four existing ML-based methods using *manually extracted features*, including clustering by k-means with dynamic time warping (DTW) [5], and classification by support vector machine (SVM), random forests (RF) and k-NN [10], [11]. Furthermore, a single-layer LSTM-based DL algorithm was also implemented for comparison. These methods and algorithms were re-implemented using the same data set and dataset partitions as described in Section III-A.

For the implementation of clustering by k-means with DTW [5], the energy profiles extracted from the daily electricity consumption and outdoor air temperature using support-vector regression were clustered into k clusters. k was then searched from $2, 3, 4, 5, 6, 7$, with a Silhouette score as the selection criteria (see [5] for further details). The clusters

were categorized using the majority of heating-type samples. For comparison, we categorize each cluster according to the majority of heating-type samples of the training set. The unknown samples were then classified according to the cluster they belonged to.

For the implementation of classifiers using SVM, RF and k-NN [10] and [11], 91 features from smart meter measurements and 8 features per weather variable were extracted for each week. In both papers, the default hyperparameter settings were used as stated in the ML package used by the authors. We conducted a grid search of the hyperparameters selected in [10] and [11] for fair comparison (for SVM, searching grids for C and γ were $0.001, 0.01, 0.1, 1, 10, 100$; for k-NN, k was searched from 1 to 30; for RF, the minimum number of samples required to be a leaf node was searched from $2, 4, 8, 16, 32$)). The best-performing hyperparameters from the validation set were then chosen.

Lastly, for a single-layer LSTM, no segmentation was applied to the input sequence $\tilde{\boldsymbol{S}}$. Instead, it takes the entire sequence of length D as input, see Section II-C1. After the single-layer LSTM, the classifier was formed as described in II-C3. The number of hidden units n was searched from $4, 8, 16, 32$. The best-performing hyperparameter on the validation set was also chosen here.

Table V summarizes the test performance from these methods in terms of average accuracy, F1-score, precision and recall. One can see from the table that the classifier in the proposed framework generated the best average accuracy (94.2%) in the test set among the six methods. However, there were two cases (EAHP in precision, and DEH in recall) in which the classifier did not top the list. On the other hand, the classifier in the proposed framework shows the best F1-score (harmonic mean of precision and recall) for all three heating types. For k-means, despite the high F1-score for DH, the low value for DEH and zero value for EAHP showed that the model was not able to distinguish between DEH and EAHP, thus resulting in a lower accuracy of 76.8%. SVM showed the best performance out of the conventional classifiers with an accuracy of 92.0%. Comparing the F1-score, it also shows that the proposed model in particular improved upon the classification of EAHP and DEH. The table also shows the effectiveness of the proposed framework over a single-layer LSTM network, which showed a test accuracy of 90.1%. Furthermore, as for the proposed framework, the conventional classifiers as well as the single-layer network also showed a lower F1-score for EAHP as compared to DEH, and in particular to DH. This could be tied back to the error analysis in Section III-B2, where the misclassifications were analyzed in relation to the heated area, the age of the building, and the geographical location. One potential reason for the lower F1-score for EAHP could be that the incorrectly classified EAHP samples had an older heat pump with a lower COP and/or lower rated capacity installed. The heat pump's immersion heater could then for instance be used for additional heat, thus using more electricity than a similar building with a larger EAHP and with a higher COP.

IV. Discussion

Due to the limitation on already collected measurements from past years, our study was limited to hourly-based smart measurement sequences. It will be possible to have higher sampling rate measurements in the future with the second generation of smart meters rolling out in Sweden. It would be of interest for further study to see whether a high sampling rate can further improve the performance of the classifier.

Furthermore, experiments were conducted on smart meter data collected from one large city in Sweden. It would be of interest to evaluate the performance of the model with buildings from other areas or countries. This can be useful, especially for smaller DSOs with a limited number of training samples as training the proposed model with a small set could lead to less desirable classification performance as shown in Fig. 8.

If the demand characteristics change over time, e.g. through demand-response and behind-meter generation and/or storage, ML methods that depend on expert-defined features need to be re-evaluated to see if the existing features would be still capable of discriminating the different heating types and whether new features need to be added. The proposed DL method, however, automatically extracts features without requiring specific knowledge from experts. This is one of the main advantages of the proposed DL method for heating system classification over existing ML methods on such applications. The improved accuracy of the proposed method over the existing methods is another merit. In many countries, DSOs use Verlander's/Rusck's method and/or typical load curves for peak load estimation. The parameters of Verlander's method and the typical load curve are developed for consumers of different types and heating systems [29]. The improved heating type classification directly affects the estimation accuracy of these parameters, which impacts the accuracy of the resulting peak load estimation. This has also a direct impact on the estimation of the grid hosting capacity for electric vehicles and solar PVs.

V. Conclusions

The proposed hierarchical LSTM-based framework has successfully classified electricity consumers' heating types by using smart meters and weather measurement sequences. The experiments were conducted on the electricity data on consumers consisting of one and two-family dwellings, for identifying their usage of either district heating, exhaust air heat pumps, or direct electric heating. Our experimental results on the test set using the proposed method have shown good performance (average accuracy 94.2%). Further detailed empirical tests have shown that adding outdoor weather data to smart meter measurements has improved performance (increased accuracy by 2.2%). Comparing with four existing ML algorithms using expert-defined handcrafted features as well as an LSTM algorithm with a simple architecture also showed marked improvement (with accuracy increased between 2.2% to 17.4%). As the developed framework is able to classify the heating types automatically using available measurement data, the need for such tedious activities as

TABLE V
COMPARISON OF TEST PERFORMANCE WITH FOUR EXISTING ML METHODS AS WELL AS A SINGLE-LAYER LSTM

. All performance values were averaged ± standard deviation over 10 runs*. The best performance is highlighted with bold font.

Method[a]	Precision (%)			Recall (%)			F_1-score (%)			Accuracy (%)
	DH[b]	EAHP[c]	DEH[d]	DH	EAHP	DEH	DH	EAHP	DEH	Total
k-Means DTW [5]	95.1 ± 1.1	0 ± 0	57.2 ± 1.3	91.1 ± 1.4	0 ± 0	93.2 ± 1.9	93.1 ± 1.1	0 ± 0	70.9 ± 1.4	76.8 ± 1.1
SVM [10]	96.5 ± 0.7	$\mathbf{96.4 \pm 2.0}$	83.5 ± 2.2	97.3 ± 1.0	65.9 ± 4.3	$\mathbf{96.7 \pm 1.0}$	96.9 ± 0.6	78.2 ± 3.0	89.6 ± 1.4	92.0 ± 1.1
RF [10]	96.6 ± 0.7	92.5 ± 2.5	82.6 ± 1.6	95.7 ± 1.1	68.3 ± 3.0	95.9 ± 1.6	96.2 ± 0.7	78.6 ± 1.9	88.8 ± 1.2	91.3 ± 0.8
k-NN [10]	96.4 ± 1.4	96.3 ± 2.3	83.0 ± 4.3	97.3 ± 0.8	64.3 ± 13.3	96.6 ± 1.1	96.8 ± 0.9	76.4 ± 11.2	89.2 ± 2.9	91.7 ± 2.5
Single layer LSTM	97.7 ± 1.0	76.3 ± 3.2	84.3 ± 2.9	95.9 ± 0.9	74.6 ± 5.2	87.9 ± 3.2	96.8 ± 0.3	75.3 ± 2.5	86.0 ± 1.3	90.1 ± 0.7
Proposed	$\mathbf{98.5 \pm 0.6}$	86.3 ± 3.7	$\mathbf{90.9 \pm 2.0}$	$\mathbf{97.7 \pm 0.8}$	$\mathbf{85.5 \pm 3.8}$	92.6 ± 2.5	$\mathbf{98.1 \pm 0.4}$	$\mathbf{85.8 \pm 2.6}$	$\mathbf{91.7 \pm 1.2}$	$\mathbf{94.2 \pm 0.9}$

* By dataset re-partitions, followed by re-training and re-testing, [a] SVM: support vector machine; RF: random forest; k-NN: k-nearest neighbor; k-means DTW: k-means with dynamic time warping; proposed: proposed hierarchical two-layer LSTM with a segment length of one month ($T = 30$ days), [b] DH: district heating, [c] EAHP: exhaust air heat pump, [d] DEH: direct electric heating.

contacting household consumers individually is minimized. A reliance on expert knowledge for selecting features can also be avoided. The improved knowledge of heating types facilitates a reliable estimation of the demand flexibility potential from these electricity consumers. Future work would be on training with larger data sets from different countries, including more heating types.

ACKNOWLEDGMENTS

This project was financed by the Swedish Energy Agency [50237-1]. The technical discussions and comments from the Energiforsk reference group have been greatly appreciated.

REFERENCES

[1] P. Kohlhepp, H. Harb, H. Wolisz, S. Waczowicz, D. Müller, and V. Hagenmeyer, "Large-scale grid integration of residential thermal energy storages as demand-side flexibility resource: A review of international field studies," *Renewable and Sustainable Energy Reviews*, vol. 101, pp. 527–547, 2019.

[2] Nothern European Power Perspectives, NEPP, "Reglering av kraftsystemet med ett stort inslag av variabel produktion," Stockholm, NEPP, Tech. Rep., 2016.

[3] W. Zhao, Y. Shang, Z. Zhang, J. Zhang, and X. Du, "Non-intrusive electric heating load identification method based on bayesian classification," in *2022 7th Asia Conference on Power and Electrical Engineering (ACPEE)*, 2022, pp. 2193–2197.

[4] G. L. Ray, M. H. Christensen, and P. Pinson, "Detection and characterization of domestic heat pumps," in *2019 IEEE Milan PowerTech*, 2019, pp. 1–6.

[5] P. Westermann, C. Deb, A. Schlueter, and R. Evins, "Unsupervised learning of energy signatures to identify the heating system and building type using smart meter data," *Applied Energy*, vol. 264, p. 114715, 2020.

[6] Z. Jiang, R. Lin, and F. Yang, "A hybrid machine learning model for electricity consumer categorization using smart meter data," *Energies*, vol. 11, p. 2235, 08 2018.

[7] C. Sandels and J. Widén, "End-user scenarios and their impact on distribution system operators - a techno-economic analysis," Energiforsk, Stockholm, Tech. Rep., 2018.

[8] H. Fei, Y. Kim, S. Sahu, M. Naphade, S. K. Mamidipalli, and J. Hutchinson, "Heat pump detection from coarse grained smart meter data with positive and unlabeled learning," ser. KDD '13. New York, NY, USA: Association for Computing Machinery, 2013, p. 1330–1338.

[9] C. Beckel, L. Sadamori, T. Staake, and S. Santini, "Revealing household characteristics from smart meter data," *Energy*, vol. 78, pp. 397–410, 2014.

[10] K. Hopf, M. Sodenkamp, and T. Staake, "Enhancing energy efficiency in the residential sector with smart meter data analytics," *Electronic Markets*, vol. 28, 03 2018.

[11] A. Weigert, K. Hopf, N. Weinig, and T. Staake, "Detection of heat pumps from smart meter and open data," *Energy Informatics*, vol. 3, pp. 1–14, 2020.

[12] A. Bagheri, M. Bongiorno, I. Y. H. Gu, and J. R. Svensson, "Estimation of frequency-dependent impedances in power grids by deep lstm autoencoder and random forest," *Energies*, vol. 14, no. 13, 2021.

[13] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on lstm recurrent neural network," *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 841–851, 2019.

[14] M. Aslan and E. Zurel, "An efficient hybrid model for appliances classification based on time series features," *Energy and Buildings*, vol. 266, p. 112087, 04 2022.

[15] M. N. Hasan, R. N. Toma, A.-A. Nahid, M. M. M. Islam, and J.-M. Kim, "Electricity theft detection in smart grid systems: A cnn-lstm based approach," *Energies*, vol. 12, no. 17, 2019.

[16] T. Toma, K. Basu, W. Rodrigues, and S. J. Galsworthy, "A deep learning based method for heat pump dryer user classification," in *IECON 2018 - 44th Annual Conference of the IEEE Industrial Electronics Society*, 2018, pp. 3455–3460.

[17] R. Zhang, H. Lee, and D. Radev, "Dependency sensitive convolutional neural networks for modeling sentences and documents," 2016.

[18] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, pp. 1735–80, 12 1997.

[19] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "Lstm: A search space odyssey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2222–2232, 2017.

[20] F. Chollet *et al.* (2015) Keras. [Online]. Available: https://github.com/fchollet/keras

[21] Boverket, 2020. Accessed 4 Apr. 2020. [Online]. Available: https://www.boverket.se/sv/energideklaration/sok-energideklaration/

[22] SMHI. (2022) Meteorologisk analysmodell MESAN (AROME) - API. [Online]. Available: https://www.smhi.se/data/utforskaren-oppna-data/meteorologisk-analysmodell-mesan-arome-api

[23] SMHI. (2017) STRÅNG - en modell för solstrålning. [Online]. Available: https://www.smhi.se/forskning/forskningsenheter/atmosfarisk-fjarranalys/strang-en-modell-for-solstralning-1.329

[24] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations*, 12 2014.

[25] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, vol. 27. Curran Associates, Inc., 2014.

[26] L. van der Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. 86, pp. 2579–2605, 2008.

[27] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[28] K. Fürst, "Electricity consumer classification using supervised machine learning," Energiforsk, Stockholm, Tech. Rep., 2021.

[29] S. Elverksföreningen, "Belastningsberäkning med typkurvor," Svenska Elverksföreningen, Stockholm, Tech. Rep., 1991.