Sub-Hourly Load Forecasting for Community-Level Flexible Appliance Management

Christoforos Menos-Aikateriniadis
National Technical University of Athens
Intracom Telecom, Telco & Enterprise Software Dpt.
Athens, Greece
christoforosmenos@mail.ntua.gr

Andreas Akarepis
Intracom Telecom, Telco & Enterprise Software Dpt.
Athens, Greece
anakarepis@intracom-telecom.com

Isidoros Kokos
Intracom Telecom, Telco & Enterprise Software Dpt.
Athens, Greece
isik@intracom-telecom.com

Pavlos S. Georgilakis
Division of Electric Power
National Technical University of Athens
Athens, Greece
pgeorg@power.ece.ntua.gr

***Abstract*—Very Short-Term Load Forecasting (VSTLF) at a residential level constitutes a challenge mainly due to the highly volatile end-user consumption patterns as well as data integrity issues that jeopardize the quality of collected data. VSTLF can enable improved flexibility quantification and more efficient residential Demand-Side Management (DSM) for load aggregators, not only on a consumer level but also on an aggregated appliance level. Inspired by the ongoing research debate between statistical-based and data-driven models for VSTLF, this work investigates the performance of a Long Short-Term Memory (LSTM) model with Feed-Forward Error Correction (FFEC), the widely-used XGBoost method and the State-of-the-Art (SoA) models N-BEATS and Prophet. Models were trained and tested on 15-minute aggregate Electric Vehicle (EV) and Air Conditioning (A/C) loads, using real measurements by the Pecan Street dataset. In contrast to common research practices, this work considers solely temporal features and historical consumption for training the models. Results indicate that firstly LSTM with FFEC and secondly XGBoost models capture more accurately sub-hourly load dependencies for both EVs and A/Cs, showcasing their ability to generalize efficiently across different appliances and sparse datasets. N-BEATS model performs adequately well but it cannot outperform the aforementioned models, while Prophet fails to capture very short term dependencies in the data.**

***Index Terms*—Very Short-Term Load Forecasting, LSTM, Prophet, N-BEATS, Aggregate Flexible Load**

I. Introduction

A. Context

Residential demand-side flexibility is universally acknowledged as a pivotal element in the transition towards a more cost-effective and environmentally sustainable energy system [1]. Precise VSTLF, which commonly refers to sub-hourly time horizons, can strongly impact the quality of services provided by energy suppliers or load aggregators to residential end-users, ensuring smart grid reliability and efficient operation [2]. Modelling and predicting very short-term electricity consumption on a household or even appliance level can be a catalyst in decision making tasks such as shaping dynamic pricing strategies, quantification and utilisation of available flexible capacity and automatic control and scheduling of flexible distributed energy resources on sub-hourly intervals [3], [4]. The necessity for accurate VSTLF is also evident when noticing operational issues faced in existing energy markets due to potential imbalances creation. For instance, the UK Energy System Operator requires a re-declaration of capacity (usually storage) every 15 minutes by all participants in the balancing mechanism [5]. Similar challenges may arise as load aggregators become more involved in energy markets.

Despite its importance in various applications, currently sub-hourly load forecasting presents significant challenges due to [6]: (1) the highly volatile energy consumption patterns of residential end-users, (2) inadequate data quality due to unavoidable transmission and storage data loss, and (3) lack of Internet-of-Things (IoT)-based sensoring data (e.g. temperature, humidity, occupancy) that can support a more precise load prediction. Lack of these parameters can increase uncertainty to decision-making systems, such as Home Energy or Demand Response Management Systems, that support end-users participation in local or regional flexibility markets.

B. Related Work

The use of smart metering readings for residential load forecasting has been widely investigated during the last years. However, the majority of existing research works focuses on models that predict community or household-level consumption on a monthly, daily or hourly granularity, even if lower granularity of data is being utilized. For instance, works [7]–[11] predicted total power consumption either on a community level (multiple households) or building-level

This project has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No 955422.

(residential or commercial) considering data that are at an hourly, 30 or 15-minute temporal resolution (or has been consolidated to such intervals). Various models and methods have been used such as LSTM [7], [9]–[11], eXtreme Gradient Boosting (XGBoost)[7], Gated Recurrent Unit (GRU) [7], Recurrent Neural Network (RNN) [7], Autoregressive Integrated Moving Average (ARIMA) [7], Support Vector Machines (SVM) and Regression (SVR) [7], [9], Bayesian Networks [8], Random Forest [7], Back Propagation Neural Network (BPNN) [10], Extreme Learning Machine (ELM) [10] and K-Nearest Neighbors (KNN) [10] algorithms. Work [12] aggregated low resolution (1-min) consumption data to a 30-minute granularity, training a Deep Neural Network (DNN) with residual blocks to predict total household consumption. In work [13] an improved XGBoost model is compared with a Light Gradient-Boosting Machine (LightGBM) and a deep LSTM model to predict total household load consumption on a 15-minute horizon, using historical data with similar resolution. XGBoost and LSTM were also trained in [14] with hourly data to predict household electricity consumption on a multi-time scale basis. Works [15] and [16] utilized IoT-related features (e.g. humidity, temperature, occupancy) for sub-hourly load forecasting. In the former, forecasting took place on a household level while in the latter demographics were also considered for community-load stochastic forecasting. However, IoT data is not universally accessible in households, thus limiting their actual applicability.

Regardless the data resolution and forecasting horizon, the aforementioned works share all the same characteristic: they all focus on total load forecasting. To the best of the authors knowledge, limited literature exists for appliance-level load forecasting on a sub-hourly basis. For instance, work [17] develops a Hidden Semi-Markov Model (HSMM) to predict 15-minute consumption of EVs, A/C units and Electric Water Heaters (EWH) using 1-minute measurements. However, that work performed forecasting on a household-level and not on a community-level. Work [18] performed appliance-level forecasting from a community perspective, proposing an LSTM model with a Feed Forward Neural Network (FFNN) for error correction. However, that work evaluated predictions on a day-ahead basis. Last but not least, works [19] and [20] predicted District Heating (DH) and EV load, respectively. Even if, in both cases, load forecasting took place on an aggregated (regional or charging station) level, an hourly forecasting horizon has been selected. The majority of the literature reviewed in this work is summarized in Table I.

TABLE I
RELATED WORK FOR RESIDENTIAL SHORT-TERM LOAD FORECASTING

Ref.	Aggregation Level	Forecasting Models	Forecasted Variables	Prediction Horizon
[4]	Household	SVM, DNN	Total Load	1-min
[7]	Commercial Buildings	LSTM, GRU, RNN, ARIMA, XGBoost, Random Forest, SVR	Total Load	30-min
[8]	Community	Bayesian Networks	Total Load	1-hour
[9]	Household	SVR, LSTM	Total Load	1-hour
[10]	Community	BPNN, LSTM, KNN, ELM	Total Load	30-min
[12]	Household	DNN + ResBlock	Total Load	30-min
[13]	Household	LightGBM, LSTM, XGBoost	Total Load	15-min
[14]	Household	XGBoost, LSTM	Total Load	1-hour
[15]	Household	CNN, LSTM	Total Load	10-min
[17]	Household	Conditional HSMM	EV, EWH, A/C	15-min
[18]	Community	LSTM with FF Control, GB, MLP, SVM	EV, EWH, A/C	1-hour
[19]	Grid (Regional)	Prophet, LightGBM, XGBoost	DH	1-day
[20]	Grid (Charging Stations)	Prophet-LSTM, ARIMA, VMD	EVs	1-hour
[21]	Grid (City)	XGBoost, CatBoost	Total Load	1-hour
[22]	Community	Deep-Autoformer, Transformer, LSTM, CNN, ARIMA	Total Load	15-min
This work	Community	LSTM with FFEC, XGBoost, Prophet, N-BEATS	EV, A/C	15-min

C. Structure and Contribution

This work addresses the identified research gap of sub-hourly (15-minute) appliance-level load forecasting for an energy community. In contrast to total household load forecasting, where general consumption advice can be deducted, obtaining focused insights into the consumption patterns of an EV fleet or a high volume of thermostatically controlled loads (e.g. A/Cs) can lead to the creation of targeted demand response programs with sub-hourly granularity. In that way, load aggregators can obtain a better estimate of the controlled assets' flexibility potential, thus providing more efficient demand-side management services. Additionally, the absence of IoT-related data in typical households prompted the authors to rely solely on historical consumption data and temporal features from the dataset. An LSTM Model with Feed-Forward Error Correction, inspired by [18], is developed and compared with the XGBoost method. Additionally, the SoA models of Prophet and N-BEATS are firstly investigated for VSTLF. The contributions of this work are the following:

- Community Sub-Hourly Appliance-level Predictions: Aggregated consumption forecasting of specific flexible assets (e.g. EVs, A/Cs) on a sub-hourly level contributes towards a more accurate estimation of the flexibility potential for aggregators offering DSM services.
- Extensive Models Evaluation: Evaluation of an LSTM Model with Feed-Forward Error Correction and comparison with XGBoost and the SoA methods of Prophet and N-BEATS, firstly tested in VSTLF problems.

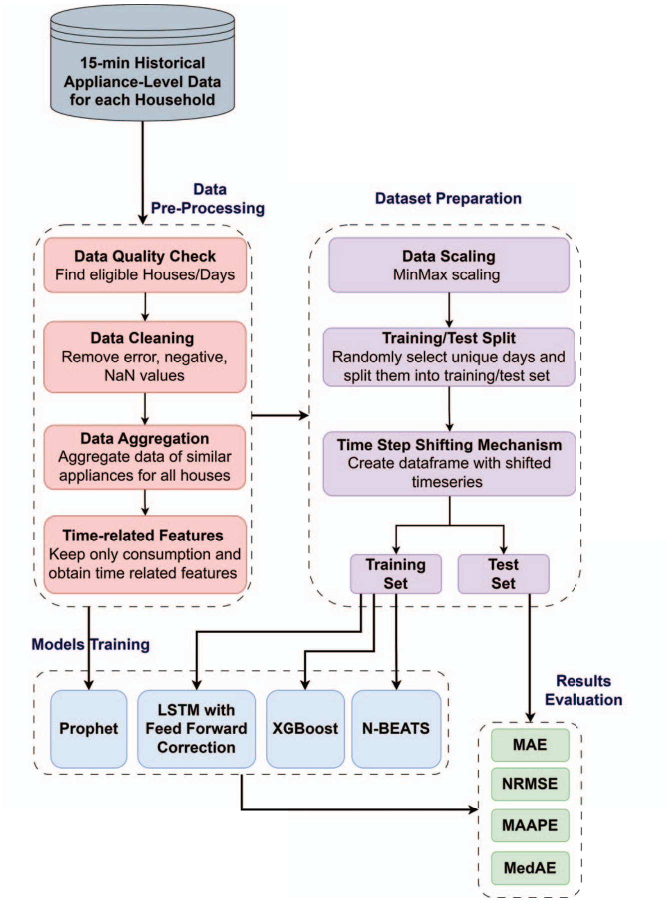

Fig. 1. Proposed methodology for aggregated Appliance-Level Sub-Hourly Load Forecasting in Residential Energy Communities

- High Model Applicability: In this work, forecasting is based solely on historical consumption and temporal-related features. This can increase models applicability since IoT data are rarely available in real-life applications.

This work is structured as follows. In Section II, the modelling framework and the investigated models are described. In Section III, the data analysis, the experimental setup and results evaluation are presented, and in Section IV the main conclusions and future research orientations are provided.

II. Methodology

A. Data Pre-Processing and Preparation

An overview of the modelling framework, showing the breakdown between data pre-processing, data preparation, models training and results evaluation is presented in Fig. 1. The first step to obtain appliance-level aggregated consumption profiles of the community is to isolate appliance-specific measurements and identify both the eligible days and houses, meaning the houses with considerable 15-minute consumption (max $>$ 1kW) throughout the year (85% of non-NaN values) and days without many missing (NaN) values. Then, data should be cleaned by removing error measurements (negative or irrationally low) and interpolate in between of NaN values. Last step involves aggregating individual household data to derive the community load and generating time-related features.

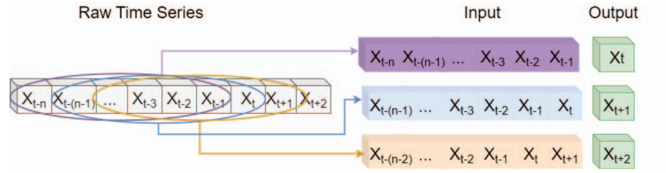

Fig. 2. Time Step Shifting Mechanism for training/test data transformation

The output of the data pre-processing module will ultimately be a dataframe with the following input features:

1) Consumption (kW): Aggregate appliance-level (EV and A/C) consumption for each 15-minute time step
2) Time: Time indices in range [1-1440] indicating the minute of the day
3) Weekday: Weekday indices in range [0-6] indicating the day of the week (0 = 'Monday')
4) Month: Month indices in range 1-12 indicating the month of the year (1 = 'January')

Dataset preparation for training and testing varies depending on the model under consideration. The Prophet model is a highly automated, statistics-based model with the capability to receive raw historical data, without any additional data transformation/modification, to train the model. On the contrary, data normalization and time step shifting mechanism are essential before data being fed into LSTM, XGBoost and N-BEATS models. In this work, MinMax scaling is applied for each feature of the processed dataset following the transformation shown in Eq. (1) and Eq. (2):

$$X_{\text{norm}} = \frac{X - \min(X)}{\max(X) - \min(X)}, \qquad \forall X \in \chi \tag{1}$$

$$X_{\text{scaled}} = X_{\text{norm}} \cdot (L_{up} - L_{low}) + L_{low}, \qquad \forall X \in \chi \tag{2}$$

where χ represents the set of input features X and $\{L_{low}, L_{up}\}$ are the lower and upper boundaries of the desired feature range. The training and test sets are defined by randomly assigning days of each month to each set. The next step is to select the temporal horizon utilized for aggregated appliance-level training and testing, as shown in Fig. 2, where n stands for the previous time steps selected as input time window. In this work, a 3-hour window has been selected, meaning $n = 12$ 15-minute time periods.

B. Short-Term Load Forecasting Models

This section offers a brief summary of the working principles underlying the investigated models, namely the proposed LSTM model with Feed-Forward Error Correction, the XG-Boost, the N-BEATS and the Prophet models.

1) LSTM with Feed-Forward Error Correction: LSTM architecture firstly introduced by [23] and further improved by [24], is a special type of RNNs that operate as sequence-based models, enabling the establishment of temporal correlations between past information and current circumstances. Given that LSTM-based models have found extensive application in VSTLF, as elaborated in Section I, a comprehensive depiction of the LSTM architecture is deemed unnecessary.

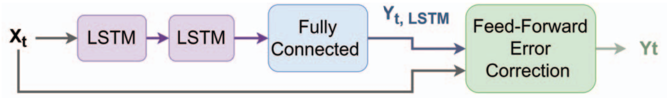

Fig. 3. LSTM with Feed-Forward Error Correction Architecture

In this work, an LSTM with Feed-Forward Error Correction architecture is investigated, inspired by a model proposed in [18]. The detailed architecture developed in this work is outlined in Fig. 3. More specifically, the selected input features $X_t = [x_{t-n}, \ldots, x_{t-3}, x_{t-2}, x_{t-1}]$ are being used to train the LSTM model, leading to a prediction (singular value) of $Y_{t,LSTM}$. Then, this prediction is being refined by feeding a Feed-Forward Neural Network with the LSTM prediction and the historical appliance-level aggregate load to produce a more accurate estimate of the energy consumption for the target time interval t. This approach aims to improve the accuracy of the aggregate predictions for each appliance considered (EV, A/C), by incorporating additional error correction mechanisms provided by the FFNN.

2) XGBoost Model: XGBoost is a high-accuracy ensemble learning model rooted in Gradient Boosting Decision Trees, published in 2016 [25]. It incorporates a regularization term to control model complexity, preventing overfitting and improving generalization. Additionally, second-order Taylor expansion helps XGBoost accelerate optimization speed and reduce modelling complexity [21]. The primary objective of XGBoost is to enhance prediction accuracy by leveraging insights gained from previous weak learners (Tree 1-t) while introducing new weak learners designed to target and correct residual errors. The model accumulates calculation results from all trees to derive conclusions. Through an iterative process of combining multiple learners, the approach ultimately leads to predictions that surpass the accuracy achieved by any individual learner.

3) N-BEATS Model: Neural Basis Expansion Analysis for Interpretable Time Series Forecasting (N-BEATS), proposed in 2020 [26], is a deep neural architecture which approaches forecasting as a non-linear multivariate regression problem. The fundamental building block is a multi-layer Fully Connected (FC) network incorporating Rectified Linear Unit (ReLU) non-linearities (Fig. 4, left). The first model block receives the input features for each time step t while the rest of the blocks receive as inputs the previous block(s) outputs. The model produces a forward forecast and a backcast, which is block's best estimate of the input. The arrangement of blocks into stacks follows a doubly residual topology (Fig. 4, middle and right), where one branch processes the backcast and one the forecast outputs of each layer. The fact that forecasts can be aggregated hierarchically enables the build-up of a very deep, explainable neural network.

4) Prophet Model: Prophet, published in 2018 by the former Facebook - currently Meta team [28], constitutes an explainable model for time series forecasting that relies on an additive model. The utilization of Markov Chain Monte Carlo methods helps Prophet to accommodate efficiently non-linear trends, incorporating yearly, weekly, and daily seasonality,

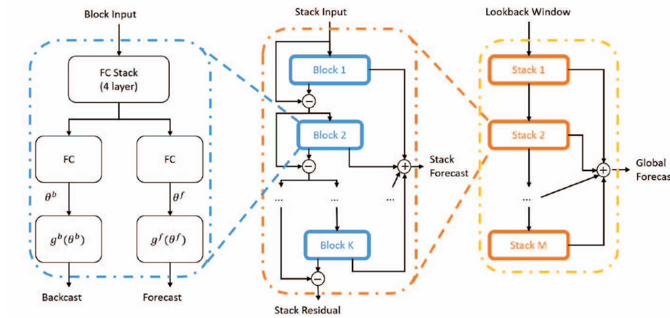

Fig. 4. N-BEATS Model Architecture obtained by [27]

along with holiday effects. Its fully automatic nature ensures reliable forecasts even on complex and incomplete datasets, displaying resilience to missing data, trend shifts, and outliers. In this work, we model and evaluate Prophet's performance on sub-hourly load forecasting, enhancing existing research where the model is being tested in longer time horizons.

C. Forecasting Performance Metrics

Performance evaluation of load forecasting models typically involves commonly used metrics such as the Root Mean Squared Error (RMSE), the Mean Absolute Error (MAE), and the Mean Absolute Percentage Error (MAPE), being a metric similar to MAE expressed in percentage terms. Given the high load volatility and the high number of time periods when community appliance-level consumption is zero, using the MAPE metric can be considered misleading for VSTLF, since such conditions may lead to outliers with disproportionately large MAPE values. For that reason, the Mean Arctangent Absolute Percentage Error (MAAPE) was proposed in literature [29]. MAAPE, ranges in $(0, \pi/2)$, given that arc tangent equals to $\pi/2$ when its argument tends to infinity: $\lim_{x\to\infty} \tan^{-1} x = \frac{\pi}{2}$. Another metric commonly used in VSTLF is the Median Absolute Error (MedAE), which is robust to outliers when values are close to (or equal to) zero.

In this work, the metrics of Normalized RMSE (NRMSE), MAE, MedAE and MAAPE are selected, as already depicted in Fig.1. NRMSE is preferable over RMSE when comparing the forecasting accuracy of appliances with different consumption levels. MedAE and MAAPE consider the existence of outliers and near-zero values in the dataset, while MAE is selected as a commonly used metric in STLF problems. The selected metrics are computed follows:

$$\text{NRMSE} = \sqrt{\frac{\frac{1}{N}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}{Y_{\text{max}} - Y_{\text{min}}}}, \tag{3}$$

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{n}\left|Y_i - \hat{Y}_i\right|, \tag{4}$$

$$\text{MedAE} = \text{median}(\left|Y_1 - \hat{Y}_1\right|, \ldots, \left|Y_n - \hat{Y}_n\right|), \tag{5}$$

$$MAAPE = \frac{1}{N}\sum_{i=1}^{n}\text{arctan}\left(\left|\frac{Y_i - \hat{Y}_i}{\hat{Y}_i}\right|\right) \tag{6}$$

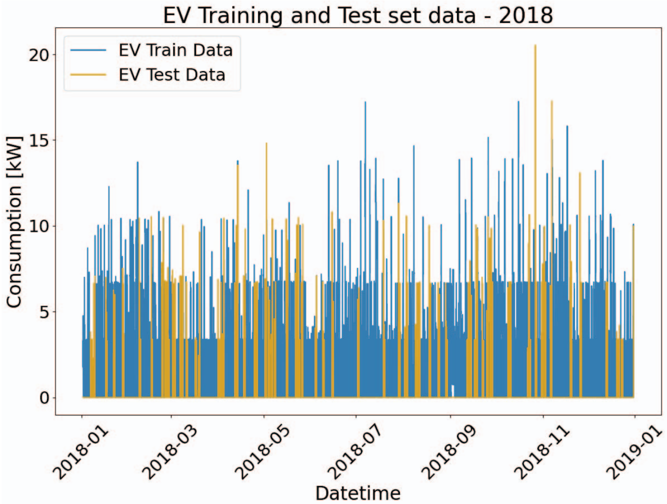

Fig. 5. Community-Level EV consumption training and test set data

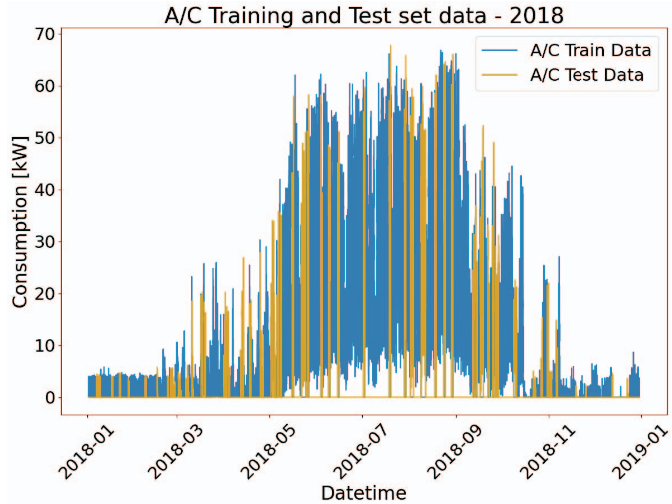

Fig. 6. Community-Level A/C consumption training and test set data

Where:

- N is the total number of forecasts.
- Y_i is the actual (true) value of i^{th} observation.
- $\hat{Y}_i$ is the forecasted value of i^{th} observation.
- Y_{max} is the test set maximum value.
- Y_{min} is the test set minimum value.

III. Evaluation

A. Experimental Setup

In this work, Python 3.11.3 is being used to train and test the investigated models. The experimental results have been conducted using a computer with Windows 11 Pro, 11th Gen Intel(R) Core(TM) i5-11300H @ 3.10GHz CPU, NVIDIA GeForce RTX 3050 Ti GPU and 16GB RAM installed.

The investigated Sub-Hourly Load Forecasting models are developed using the submetered data of EVs and A/Cs from 8 households in Austin, Texas, USA, as obtained from the Pecan Street dataset [30]. The raw data, which refer to a full 2018 calendar year, has been processed as explained in Section II, leading to the community-level EV and A/C aggregated loads presented in Fig.5-6. As it can be seen, A/C community load showcases a strong seasonal pattern, with higher consumption during summer months due to cooling requirements. On the contrary, EV communal consumption is lower than the A/C total load, showing daily/weekly patterns without any significant differentiation among seasons. 80% of the pre-processed days has been randomly selected for training and 20% for testing purposes. The modelling hyper parameter settings shown in Table II have been selected after multiple attempts (e.g. learning rate: 0.001 - 0.00001, batch size: 256 - 1024, etc), being tailored to each specific model.

TABLE II
PARAMETER SETTINGS FOR ALL MODELS CONSIDERED

Model	Hyperparameters	Value
LSTM with FFEC	LSTM, FFEC Optimizers	adam
	LSTM Units	200
	LSTM Dropout Rate	0.2
	LSTM Learning Rate	1e-4
	LSTM, FFEC Loss Functions	MAE
	LSTM, FFEC Epochs	200
	LSTM, FFEC Batch sizes	256
	FFEC Layer1,2 Sizes	256, 128
	FFEC Layer1,2 Activation Functions	ReLU
XGBoost	Loss Function	MSE
	Colsample Bytree	0.9
	Learning Rate	0.1
	Tree Maximum Depth	3
	Number of Trees	100
	Tree Subsample	0.8
NBEATS	Model Type	generic
	Lookback - Horizon	12 - 1
	Generic Neurons, Stacks, Layers	512, 20, 8
	Loss Function	MAE
	Learning Rate	1e-5
	Epochs	100
	Batch Size	512
Prophet	Interval Width	0.95

B. Modelling Results

Modelling results are being evaluated using the mean and standard deviation values of metrics NRMSE, MAE, and MAAPE calculated over the test set predictions, computed on a 15-minute basis for each appliance and model, while MedAE is calculated, by definition, over the full test set. Additionally, a ”Naive Persistence” model has been used as benchmark, predicting that the upcoming value matches the previous one.

EV community load can be characterized by a high sparsity

TABLE III
MODELLING RESULTS

Evaluation Metric	Electric Vehicles					Air Conditioning				
	Naive	LSTM with FFEC	XGBoost	NBEATS	Prophet	Naive	LSTM with FFEC	XGBoost	NBEATS	Prophet
NRMSE Mean	0.100	**0.095**	0.098	0.102	0.336	0.259	0.230	**0.222**	0.261	0.711
NRMSE Std	0.212	0.208	0.173	0.210	0.326	0.333	0.287	0.274	0.315	0.535
MAE Mean [kW]	0.455	**0.429**	0.445	0.462	1.522	2.135	1.889	**1.829**	2.146	5.850
MAE Std [kW]	0.958	0.940	0.785	0.953	1.476	2.740	2.360	2.252	2.597	4.400
MedAE [kW]	**0.000**	**0.007**	0.130	0.013	1.004	1.175	1.113	**1.025**	1.166	5.080
MAAPE Mean	1.034	1.042	**1.026**	1.037	1.133	0.640	0.635	**0.618**	0.638	0.842
MAAPE Std	0.664	0.649	0.665	0.659	0.559	0.617	0.613	0.623	0.618	0.582

(constant presence of zero or near-zero consumption intervals), while high consumption intervals are less frequent. Consequently, this EV load pattern ”helps” sequence-based models, such as the LSTM with Feed-Forward Error Correction, to perform more accurately than others. As shown in results Table III, LSTM with FFEC marginally outperforms XGBoost model in Electric Vehicles load forecasting which, in turn, showed lower prediction errors than the Naive Persistence and N-BEATS models. In absolute terms, all the aforementioned models showcase great results with a mean MAE of less than 0.5 kW, due to the volatile and sparse nature of EV consumption as well as due to the short forecasting horizon (15-minutes) that makes Naive Persistence model performing considerably well. However, EV predictions can be characterized by high dispersion, given that standard deviations of NRMSE and MAE are almost twice as much as the corresponding mean values of these metrics.

In the case of Air Conditioning load, XGBoost outperforms the rest of the investigated models, showing 14% lower NRMSE, MAE and MedAE and 4% lower MAAPE than the Naive Persistence model. We also observe that forecasting errors of LSTM with FFEC are considerably low and close to XGBoost’s performance. The higher frequency and volumes of consumption, when compared to EVs, and the lower number of zero-consumption intervals, create a larger performance gap among models, showing the superiority of both XGBoost and LSTM with FFEC models.

From Table III, it is also evident that Prophet shows the weakest performance for both EV and A/C load forecasting, with a mean 15-minute MAE of above 1.5kW and NRMSE more than twice as much as the Naive Persistence model. This indicated that Prophet model cannot capture very short term dependencies in the data since its design is heavily reliant on the identification of trend and seasonality patterns, which cannot be observed in such short term appliance-level forecasting periods. With regards to N-BEATS, being tested for the first time in appliance level VSTLF, it shows high forecasting accuracy with a mean 15-minute MAE of less than 0.5kW in EV and 2.2kW in A/C community loads, respectively. However, N-BEATS fails to strongly outperform the Naive Persistence, LSTM with FFEC and XGBoost models. This can be explained by the fact that N-BEATS has been originally designed for optimal performance in non-sparse data (e.g. stock prices, loans), that differ from volatile residential sub-hourly energy patterns where measurements can be zero.

To further showcase the effectiveness of the proposed methods, Fig. 7 presents two example days illustrating EV and A/C aggregate community loads forecasting. Except for Prophet model that is a statistics-based model, thus cannot map such short term load fluctuations, the rest of the models predict reasonably well EV and A/C total loads. It is also noticeable that the investigated models have a slight time delay in predictions when compared to the historical real values, showing the high impact of the last interval measurements in future predictions.

IV. CONCLUSIONS

In this work, the problem of sub-hourly (15-minute) appliance-level load forecasting for an energy community has been investigated. Real consumption data of Electric Vehicle and Air Conditioning community loads have been used, obtained from 8 households in Austin, Texas, USA collected during a calendar year, without any additional features usage (e.g. IoT-based data) in models training and testing. Statistical-based method of Prophet and data-driven models of N-BEATS and XGBoost have been compared with an LSTM with Feed-Forward Error Correction model. In addition, a ”Naive Persistence” model has been used as benchmark, matching the upcoming value with the previous observation. Forecasting accuracy was evaluated on a 15-minute basis, with the use of NRMSE, MAPE, MedAE and MAAPE metrics that can accommodate the sparse and volatile nature of appliance-level consumption in the calculation.

LSTM with Feed-Forward Error Correction model marginally outperforms the rest of the investigated models in the EV community load forecasting, showing a mean MAE of less than 0.5kW and a nearly 4% smaller error than the Naive Persistence model. XGBoost model closely follows the performance of LSTM with FFEC in EV community load forecasting but outpeforms all models in the A/C community

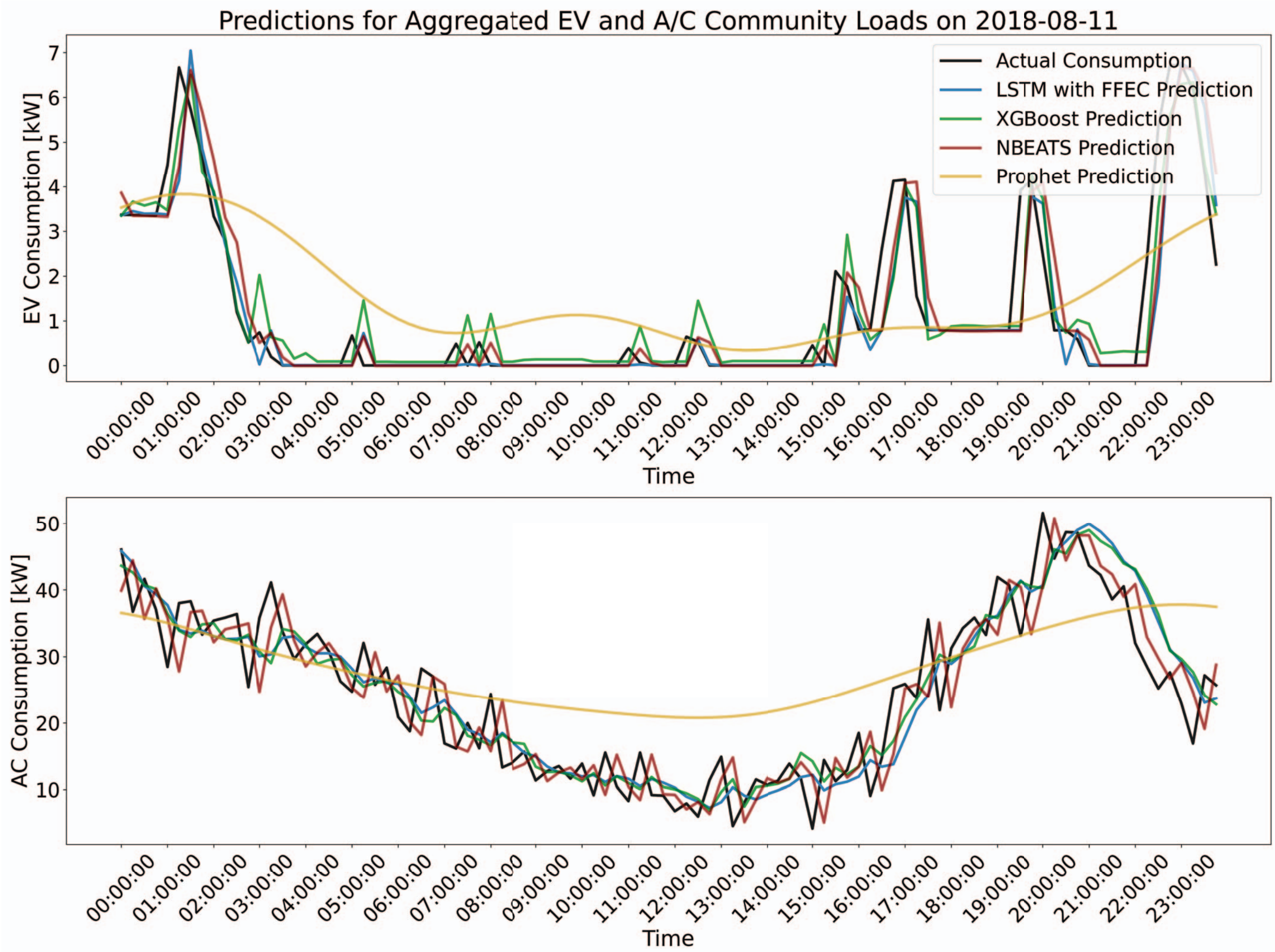

Fig. 7. Example of EV and A/C aggregate community load forecasting. Historical consumption can be seen with the black-coloured line and the community appliance-level forecasting models with the coloured lines. The impact of past values in future predictions, in such short term horizon, is evident

load forecasting, showing 14% lower NRMSE, MAE and MedAE and 4% lower MAAPE than the Naive Persistence model. However, the very short term forecasting horizon and the high number of near-zero (or zero) consumption intervals make the Naive Persistence model a highly accurate solution, especially in EV load forecasting, where it matches the performance of the N-BEATS model. Important conclusions of this work are also the fact that N-BEATS, initially designed for less sparse time series forecasting, cannot outperform LSTM with FFEC and XGBoost models in appliance-level VSTLF and that Prophet model lacks in performance since it cannot capture very short term dependencies in the data.

Future research directions can focus on the integration of human-related input (e.g. end-user preferences) in the training process. For instance, Electric Vehicles time of arrival/departure or the minimum and maximum indoor temperature requirements can improve modelling performance. In addition, the seasonal character of the A/C community load and the intra-weekly trends seen in the EV community load raises concerns on whether separate seasonal/weekly/daily (e.g. weekends vs weekday) models should be trained and utilized separately, rather than training and using a unique model for all periods.

References

[1] I. Saviuc, C. Zabala López, A. Puskás-Tompos, K. Rollert, and P. Bertoldi, "Explicit Demand Response for small end-users and independent aggregators – Status, context, enablers and barriers," JRC, Tech. Rep., 2022.

[2] S. Akhtar, S. Shahzad, A. Zaheer, *et al.*, "Short-term load forecasting models: A review of challenges, progress, and the road ahead," *Energies*, vol. 16, no. 10, 2023.

[3] E. Commission, D.-G. for Energy, M. Antretter, *et al.*, *Digitalisation of energy flexibility*. Publications Office of the European Union, 2022.

[4] R. Gonzalez, S. Ahmed, and M. Alamaniotis, “Implementing very-short-term forecasting of residential load demand using a deep neural network architecture,” *Energies*, vol. 16, no. 9, 2023.

[5] ESO, *Balancing transformation - storage stakeholder group*, Internal Report, Jul. 2023.

[6] P. Ma, S. Cui, M. Chen, S. Zhou, and K. Wang, “Review of family-level short-term load forecasting and its application in household energy management system,” *Energies*, vol. 16, no. 15, 2023.

[7] A. M. N. C. Ribeiro, P. R. X. do Carmo, P. T. Endo, P. Rosati, and T. Lynn, “Short- and very short-term firm-level load forecasting for warehouses: A comparison of machine learning and deep learning models,” *Energies*, vol. 15, no. 3, 2022.

[8] M. Bessani, J. A. Massignan, T. M. Santos, J. B. London, and C. D. Maciel, “Multiple households very short-term load forecasting using bayesian networks,” *Electric Power Systems Research*, vol. 189, p. 106 733, 2020, ISSN: 0378-7796.

[9] S. Atef, K. Nakata, and A. B. Eltawil, “A deep bi-directional long-short term memory neural network-based methodology to enhance short-term electricity load forecasting for residential applications,” *Computers and Industrial Engineering*, vol. 170, p. 108 364, 2022, ISSN: 0360-8352.

[10] 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.

[11] M. Wood, E. Ogliari, A. Nespoli, T. Simpkins, and S. Leva, “Day ahead electric load forecast: A comprehensive lstm-emd methodology and several diverse case studies,” *Forecasting*, vol. 5, no. 1, pp. 297–314, 2023, ISSN: 2571-9394. DOI: 10.3390/forecast5010016.

[12] Y. Hong, Y. Zhou, Q. Li, W. Xu, and X. Zheng, “A deep learning method for short-term residential load forecasting in smart grid,” *IEEE Access*, vol. 8, pp. 55 785–55 797, 2020.

[13] W. Cao, Y. Liu, H. Mei, H. Shang, and Y. Yu, “Short-term district power load self-prediction based on improved xgboost model,” *Engineering Applications of Artificial Intelligence*, vol. 126, p. 106 826, 2023, ISSN: 0952-1976.

[14] W. Yang, J. Shi, S. Li, Z. Song, Z. Zhang, and Z. Chen, “A combined deep learning load forecasting model of single household resident user considering multi-time scale electricity consumption behavior,” *Applied Energy*, vol. 307, p. 118 197, 2022.

[15] T. Ploysuwan, “Deep cnn and lstm network for appliances energy forecasting in residential houses using iot sensors,” in *2019 7th International Electrical Engineering Congress (iEECON)*, 2019, pp. 1–4.

[16] A. Amin and M. Mourshed, “Community stochastic domestic electricity forecasting,” *Applied Energy*, vol. 355, p. 122 342, 2024, ISSN: 0306-2619. DOI: https://doi.org/10.1016/j.apenergy.2023.122342.

[17] Y. Ji, E. Buechler, and R. Rajagopal, “Data-driven load modeling and forecasting of residential appliances,” *IEEE Transactions on Smart Grid*, vol. 11, no. 3, pp. 2652–2661, 2020.

[18] Y. Zhou, A. S. Nair, D. Ganger, A. Tripathi, C. Baone, and H. Zhu, “Appliance level short-term load forecasting via recurrent neural network,” in *2022 IEEE Power and Energy Society General Meeting (PESGM)*, 2022, pp. 1–5.

[19] A. Shakeel, D. Chong, and J. Wang, “District heating load forecasting with a hybrid model based on light-gbm and fb-prophet,” *Journal of Cleaner Production*, vol. 409, p. 137 130, 2023.

[20] N. Cheng, P. Zheng, X. Ruan, and Z. Zhu, “Electric vehicle charging load prediction based on variational mode decomposition and prophet-lstm,” *Frontiers in Energy Research*, vol. 11, 2023.

[21] L. Zhang and D. Jánošík, “Enhanced short-term load forecasting with hybrid machine learning models: Catboost and xgboost approaches,” *Expert Systems with Applications*, vol. 241, p. 122 686, 2024, ISSN: 0957-4174.

[22] Y. Jiang, T. Gao, Y. Dai, *et al.*, “Very short-term residential load forecasting based on deep-autoformer,” *Applied Energy*, vol. 328, p. 120 120, 2022, ISSN: 0306-2619.

[23] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[24] F. A. Gers, J. A. Schmidhuber, and F. A. Cummins, “Learning to forget: Continual prediction with lstm,” *Neural Computation*, vol. 12, no. 10, pp. 2451–2471, 2000.

[25] T. Chen and C. Guestrin, “Xgboost: A scalable tree boosting system,” New York, NY, USA: Association for Computing Machinery, 2016, pp. 785–794, ISBN: 9781450342322.

[26] B. N. Oreshkin, D. Carpov, N. Chapados, and Y. Bengio, “N-beats: Neural basis expansion analysis for interpretable time series forecasting,” in *International Conference on Learning Representations*, 2020.

[27] N. P. Singh, A. R. Joshi, and M. N. Alam, “Short-term forecasting in smart electric grid using n-beats,” in *2022 Second International Conference on Power, Control and Computing Technologies (ICPC2T)*, 2022, pp. 1–6.

[28] S. J. Taylor and B. Letham, “Forecasting at scale,” *The American Statistician*, vol. 72, no. 1, pp. 37–45, 2018.

[29] S. Kim and H. Kim, “A new metric of absolute percentage error for intermittent demand forecasts,” *International Journal of Forecasting*, vol. 32, no. 3, pp. 669–679, 2016, ISSN: 0169-2070.

[30] *Dataport - Pecan Street Inc.* https://www.pecanstreet.org/dataport/, [Online; accessed 20-May-2023], 2009.