Forecasting Gas Usage for Big Buildings using Generalized Additive Models and Deep Learning

Nilavra Pathak*, Amadou Ba†, Joern Ploennigs† and Nirmalya Roy*
*Department of Information Systems, University of Maryland, Baltimore County
† IBM Research, Dublin, Ireland
nilavra1@umbc.edu, AMADOUBA@ie.ibm.com, Joern.Ploennigs@ie.ibm.com, nroy@umbc.edu

Abstract—Time series behavior of gas consumption is highly irregular, non-stationary, and volatile due to its dependency on the weather, users' habits and lifestyle. This complicates the modeling and forecasting of gas consumption with most of the existing time series modeling techniques, specifically when missing values and outliers are present. To demonstrate and overcome these problems, we investigate two approaches to model the gas consumption, namely Generalized Additive Models (GAM) and Long Short-Term Memory (LSTM). We perform our evaluations on two building datasets from two different continents. We present each selected feature's influence, the tuning parameters, and the characteristics of the gas consumption on their forecasting abilities. We compare the performances of GAM and LSTM with other state-of-the-art forecasting approaches. We show that LSTM outperforms GAM and other existing approaches, however, GAM provides better interpretable results for building management systems (BMS).

Index Terms—Gas Forecasting, Deep Learning, Long Short Term Memory, Generalized Additive Model, Forecasting

I. INTRODUCTION

Gas consumption forecasting has emerged as a potential approach to enhance energy management by balancing supply and demand, improve investment planning, and for more rational price bidding. There are four types of consumers, namely, residential, commercial, industrial, and electric power generation stations. The residential sector has the highest necessity for natural gas usage followed by commercial, industrial and power generation respectively [1]. In present days, commercial buildings are adopting smart building management technologies. They help combine sensors and software solutions to manage energy, retrofit heating, ventilation, and air conditioning (HVAC), lighting, access control etc. These are steps taken by the companies to reduce operating cost and to improve the safety and security of the employees. HVAC control is one of the most important aspects of Building Management Systems (BMS). We envision that a gas consumption forecasting model specific to a building will be a part of a Building Management System (BMS), e2Diagnoser [2]. The e2Diagnoser is a real-time data mining system for the energy management of smart, sensor-equipped buildings. It predicts energy consumption at the sub-meter level and diagnoses abnormal energy leakage. Commercial buildings have periodic gas usage which varies depending on the time of the day, occupancy, weather and scheduling periods. HVAC usage forecasting is useful for optimizing peak usage by shifting the preheating and precooling hours as suggested in [3].

There is no known literature that addresses the issue of forecasting for large commercial buildings. However, prior works focus on forecasting in citywide cases. The most common approaches are variations of auto-regressive models, for example, Auto-regressive moving average (ARIMA), seasonal auto-regressive moving average (SARIMA) and seasonal auto-regressive moving average with exogenous input (SARIMAX), or Artificial Neural Networks (ANN), or a combination of SARIMA and ANN. The SARIMA models the historicity of the data and the ANN captures the non-linearity [4], [5]. We also take inspiration from works on building energy modeling and forecasting as it has close ties to gas usage [6], [7].

In this paper, we propose two approaches for building gas usage forecasting Generalized Additive Model (GAM) and a Deep Learning variant knows as Long-short term memory network (LSTM), and compare the results with the latest approaches. GAM allows representing each feature influencing the gas consumption by an identifiable and interpretable transfer function, represented by spline basis. The interpolation characteristics of GAM help to simultaneously address the problem of missing values and outliers. Also, GAM presents the relationship between data and it's covariates in an interpretable form which allows gaining insight regarding gas usage [8]. The LSTM exploits the use of several processing layers in the network to discover complex patterns in the gas consumption data, where each layer learns a particular concept that used by the subsequent layers.

We test the proposed approaches on two real-world building gas consumption datasets which have different characteristics and usage patterns. We perform a day ahead forecast and explain in details the various assumptions and analytics required to make the algorithms work for the different datasets, and compare the results with the existing state-of-the-art methods like Recurrent Neural Network (RNN), Random Forest (RF), SARIMA, SARIMAX, and Linear Regression (LR).

The rest of the paper is organized as follows. Section II reviews some prior works on time series modeling and forecasting, particularly in the context of energy consumption. In Section III, we place the emphasis on the different methods used for gas consumption forecasting. We present our case study in Section IV, where we interpret our dataset, present the preprocessing approach and the feature selection method. We present two test cases of gas consumption obtained from buildings and explain the application of GAM and LSTM to

the gas data, and evaluate the performance of LSTM compared to GAM, RF, SARIMAX, and LR. The conclusions and future works is addressed in Section V.

II. RELATED WORKS

In this section, we review some of the works related to building gas forecasting. This includes gas forecasting gas consumption on a city-wide scale, building electricity load prediction and the applications of GAM and Deep Learning.

Prior works on gas-forecasting primarily focus on city-wide cases [4]. Two common methods, for forecasting, are the usage of ARIMA or ANN models or both. ARIMA models the temporal nature of the data and ANNs can model the stochasticity in data. A comparison of the two methods in [9] shows that ANN provides better results than ARIMA because it can capture the non-linearity in the data. However, a two-step modeling achieves better results [5]. Here, ARIMA models the auto-regressiveness in the data and ANNs captures the complex nonlinearities.

Deep Learning algorithms are extensions of ANNs, particularly the variants of recurrent neural networks (RNN) are more applicable to time-series analytics. For example, in [10], predicted energy usage using RNN. LSTM [11] is a variant of RNN, particularly useful for time series modeling because it does not suffer from the problems of vanishing gradient, unlike simple feed-forward networks and recurrent neural networks.

Another class of algorithms which is quite popular for forecasting is Generalized Additive Models (GAM) [8]. The benefit of GAM over most of the other forecasting models is its ability to model complex features with transfer functions, which is useful for visualization and interpretation. Gehrke cites the interpretability as one of the major challenges in data science [12]. It enables analyst and operators to better understand the driving forces behind their energy consumption. In prior works, GAM was used for electricity consumption modeling [13], for forecasting in [6], [7] and for both in [14].

We perform a day ahead gas usage forecast using the GAM and LSTM and compare the results with RNN, SARIMA, SARIMAX, LR, and RF.

III. APPROACHES

In this section, we present our forecasting methodologies for gas consumption forecasting: GAM and LSTM. GAM is effective for modeling electricity usage for buildings in earlier work and it is interpretable. We choose LSTM as a recurrent neural network which has been effective in time series analytics and proposes a multivariate single step and a multivariate multi-step time series forecasting LSTM models.

A. GAM

In this section we review the Generalized Additive Models. GAM has the following form:

$$y_t = \sum_{i=1}^{I} f_i(x_t) + \epsilon_t \tag{1}$$

In this formulation, x_t is a vector of covariates which either is categorical or continuous. y_t is the dependent variable, which is continuous. The noise term ϵ_t is assumed Gaussian,

independent and identically distributed with mean zero and finite variance. The functions f_i are the transfer functions of the model, which are of the following types: constant (exactly one transfer function, representing the intercept of the model), categorical, or continuous. The continuous transfer functions are either linear functions of covariates (representing simple linear trends) or smoothing splines. An interesting possibility is to combine smoothing splines with categorical conditions; in the context of gas consumption modeling, this allows, e.g., for having different effects of the time of the day depending on the day of the week. In our work, we chose GAM with 1-d cubic spline basis (see [8], [14]). These spline representations allow GAM to work under random missing data or outliers exist.

In our experiments, we use 1-dimensional cubic splines, which allows us to write the smoothing splines in the following form

$$f_i(x_t) = \beta_i^T b_i(x_t) = \sum_{j=1}^{J_i} \beta_{ij} b_{ij}(x_t) \tag{2}$$

where β_{ij} are the spline coefficients and b_{ij} are the spline basis functions which depend on 1 or 2 components of x_t. Note that a (fixed) sequence of knot points define the basis functions. The measure J_i in equation (1) is the number of spline coefficients associated with the transfer function f_i. Now, let β denote the stacked vector containing the spline coefficients, and $b(x_t)$ the stacked vector containing the spline basis functions of all the transfer functions.

Baseline GAM prediction with Auto-regressive models: The prior approaches in [13], [15] took a two-step approach where the exogenous and the general electricity usage behavior were modeled using GAM. The residuals are modeled with auto-regressive models like ARMA and Auto-Regressive Conditional Heteroskedasticity (AR-ARCH) models respectively, to model the temporal for obtaining the prediction bands. We proposed three versions of GAM -

- *Baseline GAM* explains the general behavior of the usage integrating the exogenous variables
- *Hierarchical GAM* is a two-stage approach, where the autoregressive models capture the temporal behavior of the baseline GAM's residuals. As the residuals have multiple seasonality we apply Seasonal ARIMA (SARIMA) and Double-Seasonal Holt-Winters Forecasting (DSHW) [16] depending on the type of seasonality present.
- *Auto-regressive GAM* is a combination of an autoregressive model and the baseline GAM.

B. Deep Learning: LSTM

A recurrent neural network (RNN) is a type of artificial neural network (ANN) which has a feedback loop and is capable of storing information. RNN is a feed-forward neural network, which repeatedly returns the output of the same layer through time. It deconstructs into multiple layers of recurrent network where each layer corresponds to a given time step. This gives rise to a problem known as vanishing gradient problem. Since the network uses backpropagation, in a large

network, it results in an exponential decrease of the gradient through time which leads to the network to train very slowly. LSTM [11] is a special type of RNN which capable of learning long-term dependencies and thus mitigates the problem. The long-term dependency is particularly interesting in our study, as gas usage shows seasonality across different periodicity - daily, weekly and monthly.

We construct two LSTM models, where the first one acts like a regression model that consists of a single output model that forecasts the consumption at time t, y_t. It is a multivariate time series forecasting model. The second model is a multivariate multi-step forecasting model that has far fewer features than the other models and provides a multi-step (24-hour ahead forecast). The intuition here is that given the auto-regressive and exogenous data over time, the deep learning can forecast. In the Analysis section, we discuss the individual models in more details tailored to the individual datasets.

IV. ANALYSIS

We evaluate how our approaches perform on a real dataset obtained from a building. We present two case studies, where the first one is from a building and have weather data. The second case study is done, gas dataset is from publicly available dataset, from Lawrence Berkeley National Lab campus. For both the case studies, we describe the building, the associated features and the details regarding the data in the first subsection. Subsequently, we delve into the characteristics of the dataset and the adopted processing mechanism. Finally, we evaluate the performance of GAM and LSTM for the forecasting of gas consumption.

A. Case Study Building 1: IBM B3 Building

Our study is conducted with the dataset obtained from a building. The dataset that we use is 6 years old building which is actively occupied by around 200 employees. There are 2,500 sensors installed within this 3,300 sq meter to provide real-time data on heating and cooling systems, lighting, water, gas and electricity meters, footfall and motion. We consider the data within 2012 to 2016, a total of 19000 observations. Besides these measurements, weather information like solar luminescence, humidity, temperature and wind speed are also captured by some of these sensors.

1) Dataset: We use more than 4 years of gas consumption time series data having an hourly granularity. The amplitude of the consumption for 2012 is different from those of 2015-2016. The startup consumption changed from 2012 to 2016, as a new unit was added to the building, bringing in more users, which requires more heating in the starting phase. In general, there is no consumption on the weekends but on Sundays, the system starts at late night to heat up the building overnight, which results in the highest peak over the week. One of the major challenges is the presence of missing data due to faults in recording data. The other exogenous factors used are solar luminescence, humidity, outside air temperature and wind speed. The fundamental characteristics of the dataset are - no consumption data at night, as the control system switches off the boiler. This is followed by a sharp peak in the morning for preheating the building before work hour starts. Then, the amplitude of the gas consumption decreases when the building reaches the set temperature and takes a stochastic nature to maintain the setpoint.

2) Data Cleaning- Missing data handling & Outlier removal: The dataset presents several irregularities and is highly noisy which requires processing before any exploration. Furthermore, the dataset exhibits several discontinuous lasting several hours and sometimes days, and many missing gas consumption data. Also, random missing data is present when the boiler operates. The boilers don't work from the evening till the next morning, for which we assign the values to zero. If there is missing data during the operation hours we use splines to interpolate the missing values. We assume that the values of the time series that are higher than (μ + 3 * σ ; μ : mean, σ : standard deviation) are anomalies, and consider them as missing data and interpolate those values.

TABLE I: List of features with correlation with gas consumption

Features Used	Type	Symbols	Correlation Coefficient
Solar luminescence	Exogenous	LUX	0.132
Wind speed	Exogenous	Wind	0.053
Humidity	Exogenous	Humidity	-0.017
Outside Air temperature	Exogenous	OAT	-0.30
Time of Day	Heuristic	TOD	-
Hour	Heuristic	hr	-
Day	Heuristic	D	-
Month	Heuristic	m	-
LastWeekGas	Autoregressive	x_{t-168}	0.72
LastDayGas	Autoregressive	x_{t-24}	0.49
HourLastOn	Heuristic	ON	-

3) Features Extraction: In Table I, we list the features used. Solar luminescence (LUX), outside air temperatures (OAT), wind speed (Wind) and Humidity are the exogenous weather features. *LastWeekGas* (x_{t-168}) is the gas consumption in the same hour on the previous week while *LastDayGas* (x_{t-24}) is the consumption for the same hour on the last day. We computed the Pearson correlation coefficients between the features and the gas consumption to find the most influential features. Table I shows that the most influential features are *LastWeekGas* and *LastDayGas*, and the least ones are *Humidity* and *Wind*. The better Pearson correlation coefficient for *LastWeekGas* is better than *LastDayGas*, as the latter does not capture the gas consumption patterns associated with the weekends when the boilers are not operating. The exogenous variables *LUX* and *OAT* have some correlation with the gas consumption but is much lower for *Humidity* and *Wind*. We constructed another temporal feature *HourLastOn*, which measures the last time the gas in the previous day at the same hour. We inspected the data for seasonality and found three types of seasonality - daily, weekly and yearly. We considered two lags - 24 hours (daily lag) and 168 hours (weekly lag), and found both seasonality is present from the autocorrelation

function (ACF) and partial autocorrelation (PACF) plots of the differenced series.

4) Choice of Metrics: We compare with three standard metrics used for evaluating forecasting - Root Mean Square Error (RMSE), Normalized Mean Square Error (NRMSE) and Coefficient of Determination (The R-squared measure of goodness of fit).

5) Analysis of GAM: We start with the analysis of the effects of the exogenous and temporal variables on the gas consumption. Then, we apply our approach to 24 hours ahead forecasting. In IV-A2, we showed the features selected, in this section, we show their effects on forecasting. Challenges of building GAM models for forecasting the gas dataset are:

- **Averaging the effect of peaks:** As GAM is essentially based on using splines, it consequently averages the sudden peaks. By adjusting the number of knot points of the transfer functions we can change the characteristics of the splines.
- **Anomalous forecasts of weekends:** The boilers are generally stopped at night and on weekends. Although there is a general schedule where there is no consumption on weekends, for some weekends some boilers operated like weekdays, which makes the forecasting difficult.
- **GAM formulation:** We introduce the exogenous variables one by one and check the improvement on test error, and the deviation in training and testing errors.

6) Quantitative Evaluation: In this section, we explain and compare the results of the different GAM based models, baseline GAM and DSHW model and the auto-regressive GAM model.

a) Baseline Modeling: We investigated the different autoregressive, heuristic and exogenous features. Here, we show the GAM performance for each feature. To capture the temporal dependencies, we use temporal features *TimeofDay*, *Day*, *month* and *isWeekend*. The baseline that gave the best validation error is given as -

$$y_t \sim s(hour, k = 24, by = Day) + s(OAT, by = TOD) \\ + s(LUX, by = TOD) + s(ON, by = TOD) + TOD + D + M \quad (3)$$

The exogenous weather features used were *OAT* and *LUX*. The weather feature *OAT* is efficient, while adding *LUX* combined with *OAT* doesn't increases the training and cross-validation error, the test error does not increase significantly. Adding the temporal feature *HourLastOn* also improves the training and cross-validation errors, but does not cause significant increase in test error. The formula $s(hr/Day_{k=24}) + s(OAT/TOD)$ gives the best test error.

b) Residuals modeling: The residuals of the baseline GAM model is then used to forecast the auto-regressive part. The data have double seasonality at 24 and 168 hours for which we used DSHW to forecast 24 hours ahead, using the residuals for the last two weeks. DSHW is an additive exponential smoothing approach, where a model for the additive seasonal HW method decomposes the series value y_t into an error ϵ_t, a level l_t, a trend b_t and a seasonal component (s_t), as per the following equations:

Method	Eqn No.	RMSE		NRMSE		R-sq	
		Train	Test	Train	Test	Train	Test
Baseline GAM	(9)	10.64	18.68	0.13	0.25	0.73	0.60
Baseline GAM + DSHW	(9) + DSHW	-	15.5	-	0.10	-	0.73
Auto regressive GAM	(17)	10.77	13.84	0.12	0.12	0.73	0.76

TABLE II: Comparison of Baseline, Baseline Autoregressive and Autoregressive GAM

$$y_t = l_{t-1} + b_{t-1} + s_{t-m} + \epsilon_t \quad (4)$$
$$l_t = l_{t-1} + b_{t-1} + s_{t-m} + \alpha\epsilon_t \quad (5)$$
$$b_t = b_{t-1} + \beta\epsilon_t \quad (6)$$
$$s_t = s_{t-m} + \gamma_w\epsilon_t \quad (7)$$

where $\epsilon_t \sim$ N(0, σ^2), and α, β and γ_w are smoothing parameters for the level, trend and seasonal terms, respectively. The value m represents the number of seasons in one seasonal cycle. Thus, for hourly data, we get a HW(24) model that has a cycle of length 24 (a daily cycle). The result of modeling the residual with DSHW is given in Table III.

c) Auto-regressive GAM: We hypothesized that an autoregressive GAM is equally applicable for forecasting. The following equation gives the best result -

$$y_t \sim LastWeekGas + s(OAT, by = TOD) + s(LUX, by = TOD) \\ + s(ON, by = TOD) + TOD + D + M \quad (8)$$

The most significant feature that helps to model the data *OAT* and adding the *HourLastOn* further improves the performance. The feature *LUX* also improves the results but the difference is not that significant. We do not vary the knot points for GAM as we found experimentally that it does not add any benefit.

d) Comparison: In Table III, the results of the three models have been compared. The multi-stage GAM model constructed with the GAM and DSHW outperforms the baseline model. However, the auto-regressive GAM performs better than the multi-stage GAM. The DSHW also is more complex to model, to forecast 24 hours ahead as it requires at least 336 data points and every time we have to retrain the model on the new residuals. We found that an additive model with both auto-regressive factors - daily and weekly can perform better in terms of training and validation error, however, the test error does not get any better. The best model for GAM gives a training, testing and validation error of 10.64, 13.85 and 11.55 respectively. Table II shows the results for the three versions of the GAM.

7) Qualitative analysis: Now we look into the qualitative analysis of GAM by inspecting the transfer functions. We understand the general nature of the hourly consumption for the different *Day* classes in Figures 1a-1d. In general, on Saturdays, there is no consumption. On Sundays, the boiler does not work almost the entire day except at night, when

(a) Monday   (b) Tuesday - Friday
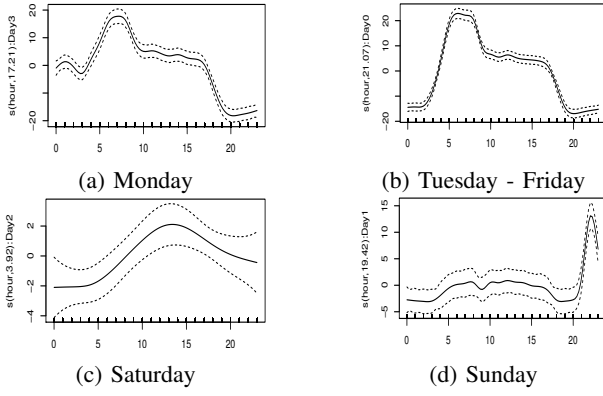
(c) Saturday   (d) Sunday

Fig. 1: Hourly consumption using transfer functions for a GAM model of the different *Day* classes.

there is a huge spike of consumption, which is because the building needs pre-heating as the boiler is off for almost two days. The Mondays' consumptions are different from the rest of the weekdays. This helps us to construct the *TimeofDay* and *Day* features. We can approximate the range of hours for each category of *TimeofDay* by observing the transfer functions. We categorize the *TimeofDay* feature in three-time regimes - night-time when boiler is off, pre-heating time during the start of the day, and the normal daytime usage. For the *Day* feature, we observed that Tuesday to Friday has similar characteristics and we choose as one category, while Monday, Saturday, and Sunday have unique features of their own, and we consider each of them as separate categories.

8) Analysis of LSTM: In the Section IV-A5, we described our approach to modeling and forecasting gas consumption with GAM. In this section we evaluate the results of gas consumption forecasting using the LSTM. LSTM captures the high level abstraction of the features which allows it to deal with the complexities of the gas consumption. We propose two versions of LSTM for single step and multi-step forecasting. The considerations to model gas consumption using LSTM are:

- **Model Selection:** We experimented different models and came across two different structures for LSTM. The first one is an LSTM network with 1-D input and 1-D output, the objective of which is too see how well the LSTM performs in a regression like forecasting. In the second LSTM model, we provided the network with a 24-hour time-step input for each the previous week's consumption, the previous day's consumption and the temperature data of the day and obtain a 24-hour ahead forecast. In Section III-B we provided a brief overview of LSTM.
- **Set of Features:** GAM provides us with some insights into selecting the features of LSTM. In LSTM we select the subset of features which provides us with the best outcome, with the objective of choosing the least number of features for generating the results.
- **Comparison of results with GAM:** We compared the

generalization error, testing error and the feature set between GAM and LSTM. The objective here is check how much accuracy can we achieve using LSTM over GAM.

9) Model 1 : Multivariate time-series Forecasting with LSTM: We experimented several recurrent models and settled for the following model for forecasting:

- Input: We make sure that the data for at least every month is available.
- LSTM layer 1 with nodes N = 64.
- Stacked LSTM layer 2 with nodes N = 64.
- Fully connected layer with 1 output node and linear activation function.

At each time step, the network receives a vector of input data and outputs a single sample of gas consumption for the target hour for the next day. We use different set of combinations of the following features {*LastWeekGas*, *LastDayGas*, *TimeofDay*, *Month*, *Day*, *LUX*, *OAT*, *HourLastON*}, and we also add *Humidity* and *Wind* to evaluate the LSTM performance over GAM and noted the performance of the forecasting decreases also in this case. We represent the LSTM Model 1 as: $< x_{t-168}, x_{t-24},$ OAT, LUX, TOD, D, M, ON $> \rightarrow y_t$.

10) Model 2: Multivariate Time-Series & Multi-step forecasting: In this section, we frame the task of forecasting as a pattern matching the task to achieve a multi-step-ahead forecast. The objective is to forecast the entire day's gas consumption as a pattern when the input provided is the entire day's data from the same day of the previous week and the day before, along with the outside air temperature. The outside air temperature has a negative correlation with gas consumption, as presented in Table I. However, it is a lagged correlation and the lag varies over different time of the year. We represent the Multi-step forecasting model as follows -

1) Input: 24 hour data $<${*LastWeekGas*, *LastDayGas*, *OAT*$>$ We make sure that the data for at least every month is available.
2) LSTM layer 1 with nodes N = 48 and ReLu activation function.
3) Stacked LSTM layer 2 with nodes N = 24 and ReLu activation function.
4) Fully connected layer with 48 output nodes and ReLu activation function.
5) Fully connected layer with 24 output nodes and linear activation function.

At each time step, the network receives a matrix of input data consisting of previous week and day's consumption, and outputs the entire next day's gas consumption. Figure 2 describes the structure of the network. We fed the inputs to the Stacked LSTM, the output of which flows into the Dense Layer. The Dense layer finally has 24 outputs for a 24-hour ahead forecast.

11) Hyperparameter Selection: For all the LSTM models we optimize the following hyperparameters - the number of layers, hidden nodes, and the activation function types. We
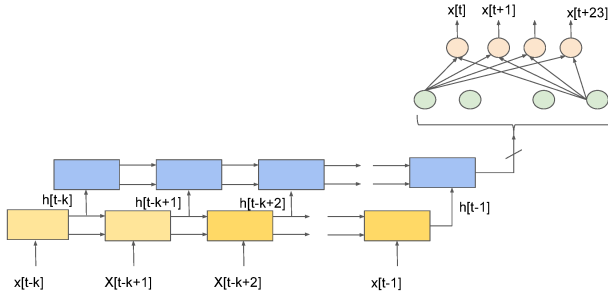
5

Fig. 2: Multivariate Multi-step Model

TABLE III: Training and testing errors of different models

Method	RMSE		NRMSE		R-sq	
	Train	Test	Train	Test	Train	Test
LR	11.24	13.89	0.08	0.08	0.70	0.66
RF	8.36	13.73	0.06	0.08	0.73	0.66
Baseline GAM	10.64	18.68	0.13	0.25	0.73	0.60
Baseline GAM + DSHW	-	15.5	-	0.103	-	0.73
Auto-regressive GAM	10.77	13.84	0.12	0.12	0.73	0.76
RNN	10.18	12.46	0.06	0.07	0.77	0.83
LSTM 1	9.86	11.34	0.06	0.07	0.79	0.86
LSTM 2	13.32	17.26	0.08	0.10	0.62	0.68
SARIMA	12.27	14.11	0.09	0.08	0.83	0.64
SARIMAX	12.03	14.15	0.08	0.08	0.83	0.64

performed a grid search to decide the activation function and settled on rectified linear units (ReLU) for the inner layers and linear for the final layers. For selecting the number of hidden layer nodes and the layers, we experimented with the different architectures and chose the one which gave us the best result.

12) Number of Epochs: We inspected the change of validation error and the generalization error with the number of epochs. We set a threshold of 200 epochs and added a condition for an early stoppage to stop executing if the validation error does not improve after 10 consecutive epochs.

13) Testing dataset results: For Model 1 we compare all possible sets of features and present the comparative results in Table III). We compared the forecasting results between Simple Recurrent Network (RNN) and LSTM Model 1. LSTM performs better than RNN for Model 1. Model 2 does not do well. Our aim here was to find whether the LSTM can perform a 24-hour ahead forecast, given week's data and the weather data. Using the current models, multivariate multi-step forecasting do not show promising results for this particular dataset.

14) Comparison between GAM and LSTM: In this section, we summarize the results provided by GAM and LSTM. The auto-regressive GAM performs best among the GAM models. The LSTM Model 1 works best for the feature sets ⟨ *LastWeekGas*, *LastDayGas*, *OAT*, *LUX*, *Humidity*, *Wind*, *ON*, *TOD*, *Day*, *month* ⟩ and has the best test error of 11.34.

The best model for both cases are different as GAM performs better without the LUX feature while LSTM can capture the LUX characteristics. Also for GAM *HourLastOn* deteriorates the performance when used with *LastWeekGas* and *LastDayGas*, this is unlike LSTM. GAM provided us with interpretable transfer functions which helped us construct

some of the features and gain better insights into the boilers' operations. We remarked that GAM overfits during the weekends and fails to capture some of the peaks if we remove the *LastDayGas* feature. In both cases, the peak on Mondays is difficult to capture. The intuition behind Model 2 is to construct a recurrent model that can utilize the multiple time series vectors and forecast ahead without much requirement for feature engineering. Although it gives some preliminary success, further research is required to come up with better multi-step forecasting models.

15) Comparison with other models: We compare GAM and LSTM with other models like LR, simple RNN and RF. We also compared GAM with SARIMAX. The features set for RNN and RF is the same as that of the Model 1 and chose the best result. To determine the parameters for the SARIMAX model, from the ACF and PACF (detailed explanation for parameter estimation of SARIMA models [17]. We found that GAM outperforms Linear Regression, however, Random Forest works pretty well on this dataset. Deep learning shows better results and further model improvement can further enhance forecasting performance. We constructed a forecasting mechanism for seasonal autoregressive gas consumption data that has influenced by exogenous factors. Although Random Forest performs better results than GAM (in terms of RMSE), the choice of GAM for a base case model is because it is easily interpretable. We construct the GAM model deduces the different contributing factors for the various features, and the transfer functions visualizes the different components. The SARIMAX model is particularly difficult to implement in cases where there is missing data present (such as unreliable sensor data due to instrumentation failures, loss of power etc.). GAM, RF, and LSTM can forecast in those cases as they learn the average patterns over time.

B. Case Study Building 2: LBNL building

The second case study is gas consumption data from Lawrence Berkeley National Lab's Building (LBNL) 74 [18], and is a publicly available dataset. The weather data for the LBNL 74 is not available so we compare the results of forecasting without them. The dataset is smaller than that in Case Study of Building 1 and has data from 2014 to mid 2015.

1) Dataset Analysis: The gas data available is at 5-minutes interval which we aggregated to hourly rate. The dataset is clean and requires no further processing. Figure 4 shows the different characteristics of the gas consumption for the LBNL building. The data only exhibits a diurnal seasonality only and the consumption pattern is less complicated than that of Building 1 which has a pre-heating characteristic that results in a sharp peak. Also unlike Building 1, as the building does not employ pre-heating or pre-cooling strategies, and there is energy consumption at night.

2) Comparison between the Different Models: In this section, we review the different methods similar to the previous case study, however, we describe the analysis in pithy.
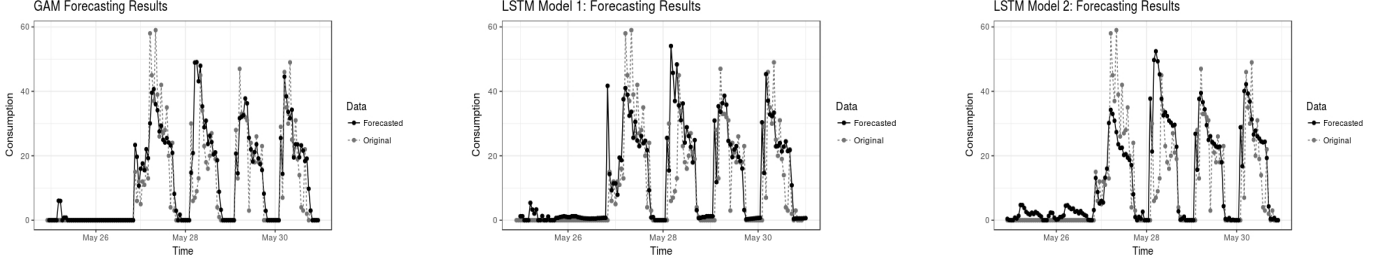
Fig. 3: Example outputs produced by all GAM, LSTM 1 and LSTM 2 for four different time durations for Building 1. Each column represents the forecasting result of the three different methods and each row is a different time duration.
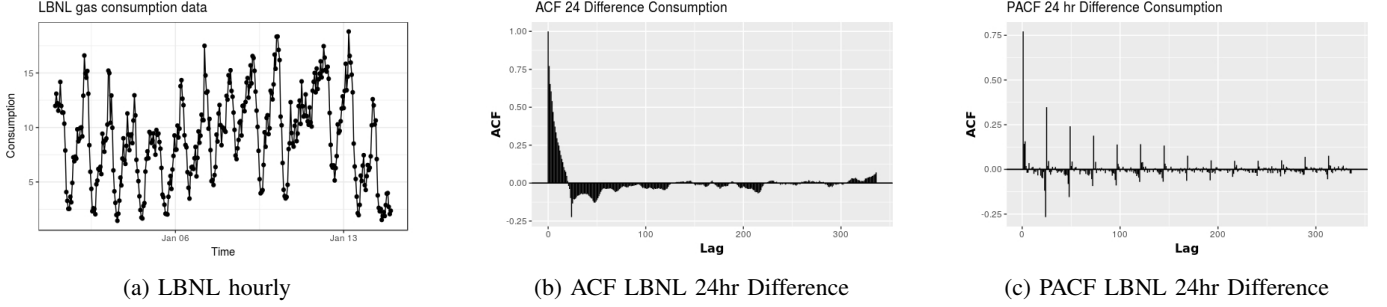

(a) LBNL hourly   (b) ACF LBNL 24hr Difference   (c) PACF LBNL 24hr Difference

Fig. 4: Characteristics of LBNL Building

Method	Parameters	RMSE		NRMSE		R-sq	
		Train	Test	Train	Test	Train	Test
Baseline GAM	Eqn 9	2.77	3.79	0.18	0.26	0.56	0.22
Baseline GAM + ARIMA	Eqn 9 + ARIMA $(0,0,1)(0,0,1)_{24}$	-	2.81	-	0.11	-	0.50
Auto-regressive GAM	Eqn 10	2.23	2.68	0.10	0.15	0.71	0.59
Linear Regression	$y_t \sim x_{t-24} + day + month + hour$	2.35	2.77	0.11	0.14	0.68	0.56
SARIMA	ARIMA $(0,1,1)(0,1,1)_{24}$	1.11	2.86	0.042	0.12	0.92	0.60
Random Forest	-	2.02	2.78	0.10	0.18	0.75	0.56
RNN	Look back = 48	1.89	2.33	0.07	0.09	0.80	0.68
LSTM	Look back = 24	1.97	2.28	0.07	0.08	0.78	0.70
LSTM	Look back = 48	1.89	2.43	0.07	0.09	0.80	0.65
LSTM	Look back = 96	1.82	3.16	0.07	0.12	0.82	0.40
LSTM	Look back = 168	1.93	2.83	0.07	0.11	0.79	0.52
LSTM	Look back = 336	2.06	2.68	0.08	0.10	0.76	0.57

TABLE IV: Comparison of Forecasting Results for LBNL Gas Data

a) **Baseline GAM**: The baseline GAM model we constructed is

$$y_t \sim s(hour, k = 24, by = Day) + Hour + Day + Month \quad (9)$$

Since we don't have any exogenous variables hence they cannot be modeled in this case.

b) **Residual Modeling**: We model the residuals of the baseline model with a SARIMA$(p,d,q)(P,D,Q)_{24}$ model. Box-Jenkins test [17] gives the parameters of the SARIMA model.

c) **Auto-regressive GAM**: The auto regressive GAM model is given as

$$y_t \sim LastDayGas + s(hr, k = 24, by = D) + hr + D + M \quad (10)$$

d) **LSTM**: Since the data is univariate, we perform a multi-step forecast to obtain 24 hours ahead forecast. We varied the training window size from 2 weeks to a day to check the effects on the forecast. The LSTM requires a large dataset so

we decided to shift the window at an hourly rate. We trained on 1-year data, and test on the remaining 6 months and split the training data into 80 and 20 ratio to create training and validation sets. The variable parameter here is **Look back**, which is the window of time that we look back to forecast ahead. The model applied for performing forecast with LSTM:

1) Input: Previous time window hour data.
2) LSTM layer 1 with nodes N = 32 and ReLu activation function.
3) Stacked LSTM layer 2 with nodes N = 32 and ReLu activation function.
4) Fully connected layer with 24 output nodes and linear activation function.

e) **Comparison with other Models**: We compare the results with other models like LR, SARIMA, RF, and RNN. In

Table IV we present our results. The baseline GAM model used is given in Eqn 9 and we model the residuals by an ARIMA$(0,1,1)(0,1,1)_{24}$ model. The model for auto-regressive GAM is given in Equation 10 which outperforms LR, SARIMA, multi-stage GAM and even RF. However, the recurrent networks give the better results and of which the LSTM with a **Look back** of 24- hours gives the best result. Unlike the Building 1 model, the multi-step forecast works best for the LBNL dataset. We checked for the different time windows for consumption and found that a window of 24 hours is enough to forecast 24 hours ahead.

C. Building Forecasting

We investigated a number of approaches for building gas consumption forecasting for two different buildings' data. Availability of exogenous data like weather, occupancy, electricity usage etc, helps generalize the consumption model. For a BMS system, where interpretability is a key issue that helps understand. In the heart of this empirical study, are two methods which lie in two opposite spectrum - GAM and LSTM, where there is a tradeoff between comprehension of the process and the accuracy.

V. Conclusions and Future Works

In this study, we presented two approaches to gas consumption forecasting, where we demonstrated the efficacy of LSTM and GAM for forecasting and interpreting the model, compared the performances of these approaches to LR, RF, RNN, and SARIMAX. To the best of our knowledge, this represents the first study of gas consumption forecasting with these methods, and is a significant step towards a better management of energy production systems. The approaches were validated on two datasets - one from 4 years of gas consumption data and the influence of each feature on the forecasting performance was provided. The other one is a publicly available dataset which does not have any weather features.

The best performance for GAM is obtained when we use in our model an auto-regression of gas consumption, on a weekly and daily basis, in conjunction with the weather variables and restricted to the temperature. While for LSTM the best results were obtained also with both the auto-regression of gas consumption, on a weekly and daily basis, along with the weather variables which are the outside air temperature and the solar luminescence.

The benefit of GAM is that it is capable of handling random missing data by modeling the data as a sum of splines which interpolate the missing data. Also, the other advantage of GAM is to gain interpretable insights regarding the data and the covariates. Deep Learning, on the other hand, can provide better forecasting results with lesser generalization error and higher accuracy, but falls short when it comes to handling missing data and providing insights.

Although LSTM provides the best forecasting gas consumption, there are rooms for improvements. In future work, we will investigate multi-step forecasting and check whether Convolutional LSTM [19] and Seq2Seq LSTM [20] model can help improve the performance, for periodic and multi-seasonal data.

References

[1] "Annual Energy Outlook 2017 ."
[2] J. Ploennigs, B. Chen, P. Palmes, and R. Lloyd, "e2-diagnoser: A system for monitoring, forecasting and diagnosing energy usage," in *Data Mining Workshop (ICDMW), 2014 IEEE International Conference on*. IEEE, 2014, pp. 1231–1234.
[3] A. Vishwanath, V. Chandan, C. Mendoza, and C. Blake, "A data driven pre-cooling framework for energy cost optimization in commercial buildings," in *Proceedings of the Eighth International Conference on Future Energy Systems*. ACM, 2017, pp. 157–167.
[4] B. Soldo, "Forecasting natural gas consumption," *Applied Energy*, vol. 92, pp. 26–37, 2012.
[5] M. Khashei and M. Bijari, "An artificial neural network (p, d, q) model for timeseries forecasting," *Expert Systems with applications*, vol. 37, no. 1, pp. 479–489, 2010.
[6] S. Fan and R. J. Hyndman, "Short-term load forecasting based on a semi-parametric additive model," *IEEE Transactions on Power Systems*, vol. 27, no. 1, pp. 134–141, 2012.
[7] A. Pierrot and Y. Goude, "Short-term electricity load forecasting with generalized additive models," *Proceedings of ISAP power*, vol. 2011, 2011.
[8] S. N. Wood, *Generalized additive models: an introduction with R*. Chapman and Hall/CRC, 2006.
[9] N. Kohzadi, M. S. Boyd, B. Kermanshahi, and I. Kaastra, "A comparison of artificial neural network and time series models for forecasting commodity prices," *Neurocomputing*, vol. 10, no. 2, pp. 169–181, 1996.
[10] E. Busseti, I. Osband, and S. Wong, "Deep learning for time series modeling," *Technical report, Stanford University*, 2012.
[11] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[12] W. Gao, J. Gehrke, and V. Kumar, "Keynotes," in *2014 IEEE International Conference on Data Mining*, Dec 2014, pp. xxix–xxxii.
[13] J. Ploennigs, B. Chen, A. Schumann, and N. Brady, "Exploiting generalized additive models for diagnosing abnormal energy use in buildings," in *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings*. ACM, 2013, pp. 1–8.
[14] A. Ba, M. Sinn, Y. Goude, and P. Pompey, "Adaptive learning of smoothing functions: Application to electricity load forecasting," in *Advances in neural information processing systems*, 2012, pp. 2510–2518.
[15] B. Chen, M. Sinn, J. Ploennigs, and A. Schumann, "Statistical anomaly detection in mean and variation of energy consumption," in *Pattern Recognition (ICPR), 2014 22nd International Conference on*. IEEE, 2014, pp. 3570–3575.
[16] P. G. Gould, A. B. Koehler, J. K. Ord, R. D. Snyder, R. J. Hyndman, and F. Vahid-Araghi, "Forecasting time series with multiple seasonal patterns," *European Journal of Operational Research*, vol. 191, no. 1, pp. 207–222, 2008.
[17] R. J. Hyndman and G. Athanasopoulos, *Forecasting: principles and practice*. OTexts, 2014.
[18] "LBNL Building 74 gas consumption dataset," https://openei.org/datasets/dataset/lbnl-building-74.
[19] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," in *Advances in neural information processing systems*, 2015, pp. 802–810.
[20] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.