A comparison of models for forecasting the residential natural gas demand of an urban area

Rok Hribar [a, b, *], Primož Potočnik [c], Jurij Šilc [b], Gregor Papa [a, b]

[a] Jožef Stefan International Postgraduate School, Ljubljana, Slovenia
[b] Computer Systems Department, Jožef Stefan Institute, Ljubljana, Slovenia
[c] Laboratory of Synergetics, Faculty of Mechanical Engineering, University of Ljubljana, Ljubljana, Slovenia

ARTICLE INFO

ABSTRACT

Forecasting the residential natural gas demand for large groups of buildings is extremely important for efficient logistics in the energy sector. In this paper different forecast models for residential natural gas demand of an urban area were implemented and compared. The models forecast gas demand with hourly resolution up to 60 h into the future. The model forecasts are based on past temperatures, forecasted temperatures and time variables, which include markers for holidays and other occasional events. The models were trained and tested on gas-consumption data gathered in the city of Ljubljana, Slovenia. Machine-learning models were considered, such as linear regression, kernel machine and artificial neural network. Additionally, empirical models were developed based on data analysis. Two most accurate models were found to be recurrent neural network and linear regression model. In realistic setting such trained models can be used in conjunction with a weather-forecasting service to generate forecasts for future gas demand.

1. Introduction

Forecasting the residential natural gas demand of a large group of buildings is vital for efficient logistics in the energy sector. Natural gas is the main energy resource in the residential sector of EU, accounting for 37% of total energy consumed [1]. Forecasting its consumption is therefore an essential part of energy management and transportation. An important example where such forecasting is required is the problem of transporting natural gas via pipelines. Such transport has to be orchestrated according to the forecasted demand that will occur a couple of days into the future. While forecasts of industrial gas demand can be obtained from each client individually, this is not the case for residential sector. This part of gas demand needs to be forecasted using models.

Gas demand forecasting is strongly connected to heating load forecasting problem. This is supported by the fact that 76% of residential natural gas in EU is consumed for space heating purposes, while 19% and 5% is consumed for water heating and cooking,

respectively [1]. Therefore, an accurate gas consumption model also includes an accurate heating load model for residential buildings. This in turn widens the usability of gas demand forecast models. If total energy consumption in buildings is considered, space heating is still the most energy intense end-use in EU homes and accounts for around 70% of total energy consumption [2]. Knowing the future consumption of heat in buildings is therefore essential for the efficient transportation of all energy resources used for space heating. In addition, an accurate forecast model for the heat consumption in buildings can substantially improve district heating and combined heat and power workflows. In both cases a successful operation requires the optimal scheduling of heating resources in order to satisfy the heating demand. The scheduling operation requires accurate, short-term, intraday forecasts of the future heat consumption in order to optimally assign the heating resources.

Modeling all factors that contribute to residential gas demand is an extremely difficult task. Since heating is the primary end-use of residential gas, an accurate forecast model need to take into account the physics of heat transfer and the accumulation of heat within buildings. This part of the model is primarily weather dependent, where the most important variable is the outdoor temperature and its past dynamics [3]. However, the model must

* Corresponding author. Computer Systems Department, Jožef Stefan Institute, Ljubljana, Slovenia.
E-mail addresses: rok.hribar@ijs.si (R. Hribar), primoz.potocnik@fs.uni-lj.si (P. Potočnik), jurij.silc@ijs.si (J. Šilc), gregor.papa@ijs.si (G. Papa).

also account for socio-economic factors. These are, for example, the dynamics of the indoor temperature preferences of the population, the heating needs of commercial buildings, the daily migrations of the population, the growth of the heating space area, the change of buildings' efficiency and others. Additionally, gas consumption model must also model dynamics of water heating and cooking.

Because there is no widely accepted model that would include socio-economic factors, the gas demand forecast model needs to be general enough to capture the theoretically unknown behavior of the system. Therefore, sophisticated computer models are used for this problem, which are otherwise found in machine learning, statistics and artificial intelligence.

The goal of this paper is to find easily trainable and accurate models for forecasting residential gas demand of an urban area with an hourly resolution for up to 60 h into the future. In this paper different forecast models are compared in order to find the most appropriate one. Three machine-learning models were applied to this problem. Those are linear regression (LR), the kernel machine (KM) and the recurrent neural network (RNN). LR and RNN model was implemented using advanced techniques that are new to the energy-modeling sector. Additionally, three empirical models were developed. Their structure is based on historical data analysis and theoretical considerations.

The layout of the paper is as follows. In Section 2 a literature review of the topic is presented. In Section 3 the mathematical formalization of the problem is outlined and methods common to all models are presented. In Section 4 three machine-learning methods are applied to the problem and presented in detail. In Section 5 simple empirical models are constructed based on an analysis of the historical data. In Section 6 the implementation of the models is described. In Section 7 the results and model comparison are presented. The implications of this comparison are discussed in Section 8. Concluding remarks are given in Section 9.

2. Related work

There are several good reviews on forecasting energy demand. Soldo [3] reviewed published research in the area of gas demand forecasting. Influence of covered area and forecasting horizon on the methods is examined. Several forecasting methods are discussed and include curve extrapolation, time series regression, artificial neural networks (ANNs), genetic algorithms and models based on fuzzy logic. Zhao and Magoulès [4] reviewed research concerning forecasting energy consumption of buildings (heating, cooling and electricity). Their review is centered around methods used in this domain. Engineering models, time series regression, ANN, KM and Gray models are mentioned. Hong and Fan [5] made a comprehensive survey of electricity demand forecasting. They review a great number of modeling techniques and weigh several side considerations regarding e.g. variable selection, error metrics, practical measures for forecasting accuracy and reliability criteria.

Based on the before mentioned reviews, forecasting horizon, the area covered and the methods used are the most important criteria for categorization of approaches for demand forecasting of buildings. The forecasting horizon can span anywhere from few hours to days, weeks, months or even years. Based on the area covered the forecasts can cover a single building, a group of buildings, a whole town or even a whole country or continent. The categorization of papers shown in Table 1 indicates that the forecasting of the energy consumption in buildings was studied for various combinations of the forecasting horizon and the area covered.

The forecasting horizon and the area covered should influence the choice of features on which the forecasts are based. When the area covered is small, then building indicators and socio-economic characteristics of occupants become important variables for forecasting. Kalogirou and Bojic [6] investigated the influence of wall thickness and insulation on the heating load of a building. Based on the data generated using simulations of heat transfer dynamics they trained a forecasting model for heating load that accounts for specific characteristic of the building. Tso and Yau [7] studied the influence of socio-economic indicators on the electricity demand of a household. They built several models that depend on indicators such as income, type of ownership, age, etc. To forecast for large covered area and large forecasting horizon Adom and Bekoe [8] used macro-socio-economic indicators, such as population count, gross national product, degree of urbanization, etc. Otherwise, the weather and time indicators are the most important features.

Methods for buildings' energy consumption forecasting include approaches from time-series modeling, machine learning, use of hybrid models and ensembles of models. Probably, the simplest method from time-series modeling is LR, where the appropriate variables are linearly mapped to produce a consumption forecast. LR models become competitive when appropriate features are found. Vondráček et al. [9] used specially engineered nonlinear mappings of observables to features, i.e., new predictors. Using those features as input to a LR model they were able to train an accurate prediction model for gas demand of a single building. Another beneficial nonlinear feature mapping is redundant Haar wavelet transform. Nguyen and Nabney [10] used such transform for generation of beneficial features which were used for several models including LR to forecast electricity demand of UK. By applying nonlinearity in this manner, models can capture nonlinear dependencies, even though the model is trained using linear regression.

Using an additional autoregressive term in the LR model can further improve the accuracy of the forecasts. Unfortunately, this makes the model nonlinear with respect to the parameters of the model, which makes the training more difficult. Tratar and Strmčnik [11] compared several autoregressive models for forecasting heating load on daily, weekly and monthly timescale. They found that there is no best model for all timescales and that the choice of the model strongly depends on forecasting horizon. Vaghefi et al. [12] developed an adaptive model that is a combination of a LR model and a seasonal autoregressive moving average model and applied it to cooling and heating load forecasting problem.

Another group of models used in buildings' energy consumption forecasting and which come from machine learning are kernel machines. Such models are nonlinear and work by nonlinearly mapping the feature space to a high-dimensional space from which the forecasted quantities are calculated linearly. Oğcu et al. [13] used support vector regression (SVR) with radial basis kernel function to forecast electricity demand of Turkey and showed that it outperforms ANN. To search for the optimal parameters of the kernel grid search or quadratic programming algorithm is usually used. Al-Shammari et al. [14] used firefly algorithm to find kernel parameters and showed that this results to more accurate forecasts of heating load compared to grid search algorithm. Protić et al. [15] showed on the case of heating load forecasting that the power of SVR can be increased by using features obtained by discrete wavelet transform. Zhu et al. [16] further upgraded SVR method to filter out data points that are close to the state at prediction time but have a different time dynamics. By this method SVR is more suitable for time series prediction. The proposed model was shown to outperform classical SVR, ANN and autoregressive integrated moving average (ARIMA) in the case of forecasting daily gas demand of UK.

Artificial neural networks (ANNs) are also a widely used model for buildings' energy consumption forecasting. Such models are

Table 1
Papers that deal with heating load, natural gas or electricity consumption forecasting, categorized with respect to the forecasting horizon and the area covered. The shaded area shows the two categories to which this paper belongs to.

	building	town	country
hour	[28]	[14, 15, 21]	—
day	[7, 17, 22–24, 26, 31, 33]	[12, 13, 17, 20, 30, 32]	[10, 16]
week	[7]	[11, 19]	—
month	—	[11, 18]	[13]
year	[9]	—	[8, 27]

nonlinear and harder to train, but they have proven to be successful for such forecasts. Potočnik et al. [17] made a comparison of a feed-forward ANN with a single hidden layer with several other models for forecasting heating load and found that autoregressive models outperform ANN. Izadyar et al. [18] used ANN with three layers and compared it with other models and reached an analog conclusion. The reason why feed-forward ANN architectures are less competitive is that they can not model long-range correlations which is why they are not the most appropriate for such problems. ANN architecture that can model long-range dependencies is RNN (which includes feed-back loops). Kato et al. [19] used RNN for heating load forecasting and showed that it outperforms a three layered feed-forward ANN. Song et al. [20] used a much more easily trainable variant of ANN called extreme learning machine to forecast electricity demand of college campus. Shamshirband et al. [21] forecasted heating load by using a ANN variant called adaptive neuro-fuzzy inference system (ANFIS), which is a kind of ANN based on the Takagi−Sugeno fuzzy inference system.

In cases where only a rough estimate of the buildings' energy consumption is needed, clustering can be used. El-Baz and Tzscheutschler [22] used k-nearest neighbors clustering on historical data of electricity demand of a single building. Using this method they were able to forecast the probability of demand being in some consumption range. A different clustering method was used by Valgaev and Kupzog [23] who performed clustering in the space of different realizations of consumption dynamics, i.e., in the function space. In this way a continuation of the current consumption dynamics can be generated by taking dynamics from the same cluster that the current dynamics belongs to.

In case of a small covered area, it is possible to develop more theoretically founded models. De Rosa et al. [24] derived differential equations for heating and cooling demand of a building and built a computational model of the demand. For a single building the physics of the consumption is manageable, however, for larger groups of building such approach is not practical. A more general approach was used by Karabulut et al. [25] by using genetic programming to forecast electricity demand of a large covered area. This is a procedure that uses evolutionary algorithms to search for a formula for the calculation of energy consumption. The benefit of this approach is that the resulting model is highly interpretable because the model is a human readable formula.

Hybrid models are also widely used to forecast the energy consumption of buildings. This usually means that the output of one model is fed as an input to a different model. One option for a hybrid model follows the attempt to join the strength of the autoregressive-type model that is able to model long-range correlations well, and a model that can model nonlinear behavior well. De Nadai and van Someren [26] combined ARIMA and ANN for anomaly detection in gas consumption. In this model features are fed to ARIMA and the output of ARIMA together with starting input features are then fed to ANN. Barak and Sadegh [27] fed ARIMA output

to ANFIS to forecast electricity demand. Amara et al. [28] used the output of nonparametric kernel density estimation as an additional feature to a trainable autoregressive model to forecast electricity demand. Hsu [29] combined linear prediction and the k-means clustering in a single hybrid model known as cluster-wise regression, which has been used to forecast buildings' electricity consumption.

Another way of joining the strengths of different models is to use an ensemble of different models. This means that different models vote about the value of the forecast. Jovanović et al. [30] used an ensemble of ANNs to forecast heating load. Ensemble was clustered so that the ANNs used for voting were guaranteed to be diverse. But the models inside the ensemble can also be of a different type. Fan et al. [31] constructed an ensemble to forecast electricity demand that consisted of LR, ARIMA, ANN, SVR, random forest, boosting tree, multivariate adaptive regression splines and k-nearest neighbors. They showed that the ensemble outperformed all included models.

3. Methods

In this section the methods that are common to all the models used in this paper are described.

3.1. Data

The models considered in this paper were trained on residential gas demand data that was gathered in the city of Ljubljana, Slovenia. The data excludes industrial gas demand, however some unregistered small-scale industrial users might still be included. The gas demand data includes the total gas demand in Ljubljana for every hour over the past 8 winter seasons (36, 106 hours of data). Time range of the available data during a winter season varies among different seasons. The beginning of the winter season ranges from September to November and the ending ranges from April to May. Because the actual values of the gas demand are confidential, all values in this paper are in units of maximum daily demand (MDD). The weather data used was gathered by the Slovenian Environment Agency. The weather forecast data used in this paper was generated using the ALADIN/SI model, as used by the Slovenian Environment Agency. The weather forecasts have a forecasting horizon up to 60 h. Both the gas demand data and the weather forecasts have an hourly resolution.

3.2. Feature selection

A comprehensive analysis of the gas demand data and the weather data was made. This data analysis consisted of Spearman's rank correlation among different observables. It was found that temperature is by far the most important observable. Table 2 lists the correlations between the gas demand and the weather variables. The correlation was calculated using the entire data set without applying any time shift to the variables. Given the correlations our models will use temperature as the only weather observable. This is also convenient with respect to data acquisition.

Table 2
Spearman's rank correlation coefficient ρ of gas demand and a given weather variable.

weather variable	ρ
outdoor temperature	−0.89
solar irradiance	0.06
air humidity	−0.02
wind speed	0.01

Using this method the wind speed, air humidity and solar irradiance were found to be much less correlated with the gas demand and therefore irrelevant for a gas demand forecasting.

Gas demand is also strongly time dependent, which might relate to different social factors. Weekly and daily seasonality is clearly observable, which can be seen in Fig. 1. Additionally, there is a discrepancy on public and school holidays and also on the working days that are between a public holiday and the weekend. Table 3 lists all the variables used by the models in this paper.

Heating degree days (HDD) were not considered as a feature because all models in this paper incorporate a temperature bias term. Therefore, HDD is implicitly included in all models which makes explicit usage of HDD as a feature unnecessary.

3.3. Forecast model framework

The framework of the forecast models is intended to be as realistic as possible. Under realistic circumstances the forecast model can only use measured weather variables from the past and forecasted ones for the future. This is also the framework of the models used in this paper. This is important because forecasted weather variables are less accurate and result in less accurate gas demand forecasts. Using this method the error in the models is not underestimated.

The model's input is a certain subset of the available time data, past temperatures, forecasted temperatures and past gas demand. The output of all the forecast models is the gas demand inside a 24-h window with an hourly resolution. The forecasting horizon specifies the position of this window. A depiction of this framework is shown in Fig. 2.

It must be noted that all forecast models in this paper are short-term, with 60 h being the maximum forecasting horizon. In this regard, they are not designed to account for long-term gas

Fig. 2. The problem of forecasting future gas demand. The shaded area is a 24-h time range for the forecast and the cross-shaped ticks are the desired output of the forecast model for each hour in that range.

consumption trends which are guided by gas availability, price, population size, etc. To account for such long-term trends specialized forecast models need to be constructed [5]. Therefore, to ensure the accuracy of the short-term model for a long period of time, the model should be periodically adapted to reflect the current state of long-term gas demand. This can be achieved by using adaptive models [17] or by simply retraining the model from time to time using the newly gathered data.

3.4. Model comparison and training

Cross-validation was used in order to obtain a reliable estimate of the model's accuracy. This was implemented by setting one season as a test set and the other seasons as a training set. For each run a different season was excluded to be a validation set. Therefore, 8 independent runs were made for each model and the mean was used for the model's error estimation. This estimation should be very close to the error of the model if it were used in real life.

When training the models there is no need to use the less-accurate forecasted temperatures because the measured ones are available for all times. The benefit of using measured variables is that the models are able to train on more accurate data. However, when using less-accurate forecasted weather data for training, a model might learn some useful features that are typical for forecasted data. In the extreme case, the model might learn to correct some errors made by weather forecasting. For example, it might learn to transform the forecasted temperature dynamics that are not observed in practice to a more likely version. On the other hand, training on less-accurate data should intuitively produce a less accurate model.

To test whether training on forecasted weather data is beneficial, each model was trained on measured data and also on forecasted data, with the forecasting horizon being up to 60 h. This can be easily implemented if the first training is conducted on measured data and the resulting model is then used as the initial point for the training on forecasted data. The forecasting horizon can be incrementally increased using the previous result as the initial point of the training. With this procedure, initial points for the training that are close to the optimal one can be found. This substantially decreases the training time.

Fig. 1. Density of data points for different values of gas demand and hour of the day. The median curve is represented as a weighted sum of triangular basis functions ψ^d.

Table 3
Variables used by the models in this paper. Variables ph, sh and hw are boolean whereas the others are real.

symbol	variable
P	gas demand
T	outdoor temperature
t_d	time of the day, i.e., time (mod) 1 day
t_w	time of the week, i.e., time (mod) 1 week
ph	presence of public holiday
sh	presence of school holiday
hw	day between public holiday and weekend

3.5. Error metric

The chosen error metric should reflect the cost that results from a bad gas demand forecast. For example, this can be a financial loss or the quantity of CO_2 emissions that is a direct consequence of a bad gas demand forecast. These costs are usually linear with respect to the absolute error, i.e., it costs as much as the model misses the correct value [32]. It is less likely that the cost is proportional to the square of the error, even though this is the prevalent error metric found in the literature [5].

The metric used in this work, both for testing the accuracy of the models and for training them, is the mean absolute error (*MAE*)

$$MAE = \frac{1}{n} \sum_{i=1}^{n} \left| P_i^{\text{forecast}} - P_i \right|,$$ (1)

where *i* is for all *n* hours of forecasted gas demand. P^{forecast} is the gas demand forecasted by the model and *P* is the measured gas demand. This means that the error for each hour of the forecast is added to the *MAE* separately.

The choice of error metric, however, did not influence the implementation of the models and their training methods. For any other cost metric, one can easily adjust the implementation so that the optimal model for that specific metric is searched for.

For the purpose of the analysis the mean absolute percentage error (MAPE) is also used, which is defined as

$$MAPE = \frac{100\%}{n} \sum_{i=1}^{n} \left| \frac{P_i^{\text{forecast}} - P_i}{P_i} \right|.$$ (2)

4. Machine-learning models

In this section three machine-learning methods are presented and applied to the problem of the gas demand forecasting.

4.1. Linear regression

LR is a standard and straightforward approach to the prediction of time series. Its use is also common for the heating load forecasting [33]. In LR the forecast is calculated linearly from hand-chosen regressors. Because the model is linear with respect to the model parameters, the training can be easily accomplished. To choose the best subset of regressors, stepwise regression can be used, where the regressors are iteratively added and removed based on their statistical significance [34]. This makes the model more robust and without any unnecessary regressors.

The first important group of regressors for this problem are the current and past temperatures which are essential for the part of gas demand used for space heating. The temperature before the forecast is important because of the heat accumulation. The rationale for this assumption is the example of a period of warm weather, followed by colder weather. In this case it is reasonable to expect that because of the accumulated heat, the heating load will not rise instantly when the temperature drops. Therefore, the heating load should increase with some delay. The regressors for the temperature *T* can be written as

$$p(D)T_i = \sum_{j=0}^{r} a_j T_{i-j},$$ (3)

where *p(D)* is an order *r* polynomial of the shift operator *D* and a_j are the model parameters.

Besides the temperature, the time of the day t_d and the time of the week t_w are important for the gas demand forecast. It was observed in the data that the dependency of the gas demand with respect to t_d and t_w is nonlinear. The nonlinearity with respect to t_d is shown in Fig. 1. But those dependencies can be represented as a linear combination of some basis functions. In this way linearity with respect to the parameters is still retained. In this paper this basis consists of differently translated periodic triangular functions ψ. The shape of basis functions ψ is shown in Fig. 1. The median curve in Fig. 1 is a weighted sum of basis functions ψ and shows how arbitrary continuous function can be decomposed using basis ψ.

The regressors used to capture the time dependencies can be written as

$$n_d(t_d) = \sum_{j=1}^{k} b_j \psi_j^d(t_d)$$
$$n_w(t_w) = \sum_{j=1}^{l} c_j \psi_j^w(t_w),$$ (4)

where *k* and *l* are the numbers of basis functions ψ and b_i and c_i are model parameters. The width of triangular functions was chosen to be 2 h for ψ^d and 2 days for ψ^w. In this way, the contribution of day of the week to gas demand changes continuously through time without step like discontinuities that would appear when going from one day of the week to the next. The information about whether a day is special (*ph*, *sh* and *hw*) was also added into the n_w function basis. In this way the presence of special days is also additively included in the model. The use of triangular basis functions allows a continuous dependency of gas demand with respect to both t_d and t_w, even in the presence of special days.

The last group of regressors are the values of the measured gas demand *P* before the forecast.

$$q(D)P_i = \sum_{j=h}^{m} d_j P_{i-j},$$ (5)

where *q(D)* is the order *m* polynomial of the shift operator *D* and d_j are the model parameters. The minimal lag *h* ensures that the model knows only about the gas demand values before the forecast. Without this, the model would be nonlinear with respect to the parameters.

Using all the regressors described above the LR model is

$$P_i^{\text{forecast}} = p(D)T_i + n_d(t_d) + n_w(t_w) + q(D)P_i.$$ (6)

4.2. Kernel machine

Because a large set of historical data is available, it is possible to construct a density distribution of the points. This, in turn, allows one to forecast the gas demand based on that distribution. This is what the KM allows the user to do generically. The state of the system for the KM model is defined as:

$$y = (x; P) = (t_d, t_w, T; P).$$ (7)

This system state does not know anything about the past temperatures, which should make the model less accurate. This will be corrected with a variation of the KM model that is described below.

From the known points y_i in the historical data a probability distribution $w(y)$ can be constructed. Knowing w, a forecast for the gas demand P_{new} can be calculated for some new state x_{new} using a conditional expectation[1]

$$P_{new} = E[w(y)|x = x_{new}] = E[w(x_{new}; P)]. \tag{8}$$

Knowing the distribution of P, given a state x_{new}, also allows the user to calculate the error of the forecast. More precisely, for an arbitrary x_{new} the conditional distribution $w(x_{new}; P)$ can be calculated. The center of this conditional distribution is the forecasted gas demand P_{new} and the width of this distribution is the error of this forecast.

A drawback of this model is that past temperatures do not have any role in the forecast. For this purpose new features were constructed that also hold information about past temperatures. The new variables T^i are short-term, medium-term and long-term temperature averages. They are defined using a convolution (marked with $*$) with an exponential function in the following way

$$T^{short}(t) = T(t)*\exp(-t/\tau_{short})/\tau_{short}. \tag{9}$$

By using T^i, a new feature space can be defined as

$$\tilde{y} = (\tilde{x}; P) = \left(t_d, t_w, T^{short}, T^{medium}, T^{long}; P\right). \tag{10}$$

KM with feature space $\tilde{y}$ will be referred to as a kernel machine with memory (KMM). This form was inspired by LR where past temperatures also contribute to the gas demand forecast in a linear way. The exponential was chosen because it has a clear time scale τ_i and because a convolution with an exponential can be calculated with $\mathscr{O}(n)$ time complexity.

In this paper $\tau_i = \{0, 24\,h, 48\,h\}$. Roughly speaking, T^j corresponds to the current temperature, the 24-h average and the 48-h average. It is important to note that the points $\tilde{y}_i$ are less densely packed than the points y_i because the dimensionality of space displayed in Eq. (10) is higher than in Eq. (7).

The problem of a low density of points can be a major one. A reliable forecast of P_{new} requires an abundance of known points near x_{new}. It is unlikely that the data includes all the common temperature dynamics for all the days of the week and every hour of the day. Because of that, the kernels have to be wide enough to compensate for the lack of points, which causes wider distributions and therefore larger errors. The only remedy for such lack of points is gathering more data instances. In order to avoid this problem, variables like public holidays and school holidays had to be omitted from x, even though they affect P substantially [17]. Those events are simply too rare and those regions do not have a sufficient number of points for kernel methods. This is why it is expected that the KM would have the largest errors during those occasional events.

4.3. Recurrent neural network model

An ANN is a common and successful model for gas demand forecasting. Various architectures are used in the literature and include feed-forward [18], recurrent [19] and ensemble [30] architectures. It is also known that an ANN can approximate an arbitrary function to an arbitrary precision [35]. This means that an ANN can capture any function if it is large enough. If there is an accurate gas demand prediction function based on time and

temperature, a large enough ANN can approximate this function. It is only a question of the user's ability to find those ANNs using optimization techniques.

An ANN is a model that consists of the composition of parameterized linear maps and fixed element-wise nonlinear maps interchangeably. ANNs that consists of several applications of linear and nonlinear maps are called deep ANNs. The depth of an ANN significantly improves the capacity of the model, which means that deeper ANNs are able to capture a wider range of functions [36].

It is desirable that an ANN model also takes past temperatures into account. Therefore, a wide enough ANN is needed, one that can take several days of data as a single input. Since the number of parameters of an ANN grows quadratically with the width, an increasing width can quickly lead to over-fitting. Another way for the ANN to know of past temperatures is the use of feedback loops, which allow the ANN to keep a memory of previous inputs. An ANN that includes feedback loops is called a RNN, an example of which is shown in Fig. 3. A RNN can be fed one temperature at a time, which means that the network can have a small width (and therefore a manageable number of model parameters) and also takes past temperatures into account. Therefore, the RNN will be used in this paper.

In the past RNNs were not widely used because they were difficult to train due to the problem of vanishing and exploding gradients [37]. These problems were overcome by introducing gated units such as a long short-term memory (LSTM) and gated recurrent unit (GRU) [38]. Accuracy wise, GRUs are comparable to LSTMs [39]. In this paper GRUs are used because they have a smaller number of parameters per neuron. The definition of one GRU layer is

$$\begin{aligned} \mathbf{z}_t &= \sigma_g(W_z\mathbf{x}_t + U_z\mathbf{y}_{t-1} + \mathbf{b}_z) \\ s_t &= \sigma_g(W_s\mathbf{x}_t + U_s\mathbf{y}_{t-1} + \mathbf{b}_s) \\ y_t &= (1 - \mathbf{z}_t)\circ\sigma_h\big(W_y\mathbf{x}_t + U_y(\mathbf{s}_t\circ\mathbf{y}_{t-1}) + \mathbf{b}_y\big) + \mathbf{z}_t\circ\mathbf{y}_{t-1}, \end{aligned} \tag{11}$$

where $\mathbf{x}$ and $\mathbf{y}$ are the input and output columns. W_i, U_i and $\mathbf{b}_i$ are the parameters of the layer. $\mathbf{z}$ and $\mathbf{s}$ are the update and reset gate columns. The proposed [39] nonlinear maps σ_g and σ_h are hard
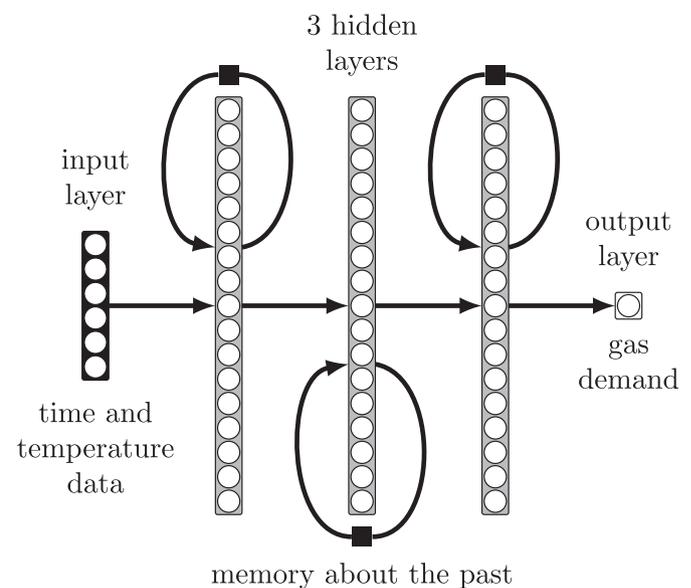
Fig. 3. RNN model architecture where all the layers and units are explicitly shown. The connections between the layers are shown with arrows. Hidden layers consist of GRU units, which need values from the previous time step. For this purpose a one-step delay is represented by a black square.

---

[1] In the case that the metric is the absolute error and not the square error, the expectation is replaced with the median.

sigmoid and tanh functions, respectively.

A three-layered GRU with a width of 16 was used in this paper with a total of 4337 parameters. The model is shown in Fig. 3. The size and architecture of the RNN was chosen with respect to the number of data points available $(36, 106)$ in order to prevent over-fitting and to have enough width for the data to pass through the RNN. The input of the model is the vector

$$(T, t_d, t_w, ph, sh, hw). \tag{12}$$

For energy related forecasting, one-layered RNN is also widely used [5], however this might be due to historical reasons. Before the development of deep learning, one-layered neural networks were the only trainable neural network models, while deeper architectures suffered problems due to exploding or vanishing gradients [37]. The three-layered architecture (Fig. 3) was chosen after experimentation with different number of layers. These experiments showed that the error of one- and two-layered models are 7.9 and 4.3 times higher compared to the three-layered one.

5. Empirical models

In this section simple and easily trainable models for gas demand forecasting are constructed. Unlike LR, KM and RNN models, the design of these models is based on the characteristics of historical gas demand data compared to weather and time data.

5.1. Two-reservoir model

By analyzing the historical data an empirical model was constructed

$$P = n_w(t_w)n_d(t_d)f(T) + g(T) + u_w(t_w)u_d(t_d), \tag{13}$$

where n_i, f, g and u_i are unknown functions. The interpretation of the unknown functions is outlined below. The form of this model stems from the following observations based on the historical gas demand data.

- the influence of T on P is different in the nighttime compared to the daytime, hence the two temperature-dependent functions $f(T)$ and $g(T)$ (see Fig. 4),
- with the proper scaling the influence of the hour of the day on P is similar for all the days of the week, and hence the product form $n(t) = n_w(t_w)n_d(t_d)$,
- gas demand is non-zero even at high temperatures, hence the temperature-independent term $u(t) = u_w(t_w)u_d(t_d)$.

This model will be referred to as the two-reservoir model (TRM). The name stems from the interpretation of first two summands of Eq. (13) which represent space heating contribution to gas demand. The two summands represent the total heat consumption of two different reservoirs: one that consists of buildings that are always heated and the other that consists of buildings where only a fraction $n(t)$ of them are heated. The two reservoirs are allowed to have different heat-transfer characteristics, hence the two temperature dependencies f and g. The remaining summand $u(t)$ from Eq. (13) represents the gas demand that is independent of the outdoor temperature and can also account for other end uses of gas consumption. Within this framework unknown functions from Eq. (13) have a clear interpretation.

The unknown functions n_i and u_i are parameterized as in Eq. (4). Therefore, TRM model is able to continuously shift from daytime to nighttime regime and also continuously adjust gas demand for different days of the week. For the functions f, g a basis of differently
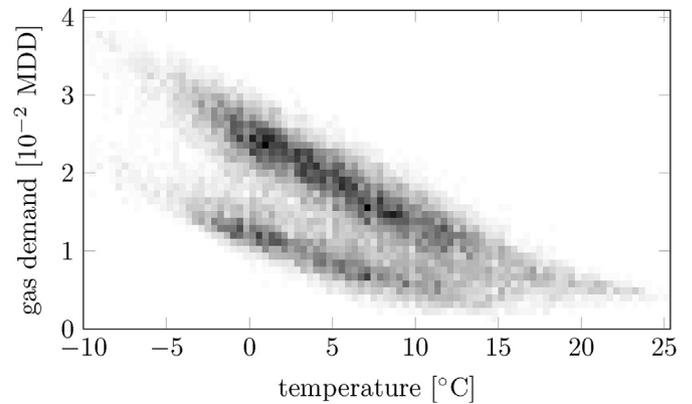
Fig. 4. Density of the data points for different values of the gas demand and temperature. Two branches are clearly seen, from which the lower one happens mostly during the nighttime and the other one mostly during the daytime. The branches are not only scaled differently but actually have different shapes.

translated hard sigmoid functions and two ramp functions for high and low temperatures was used.

The calculation of the model forecasts for a large set of data consists of matrix multiplications with the column of model parameters $\mathbf{p}$ to obtain columns with different function values. These columns are then element-wise multiplied (marked with $\circ$) and added in accordance with the model design:

$$\mathbf{P} = M_{n_w}\mathbf{p} \circ M_{n_d}\mathbf{p} \circ M_f\mathbf{p} + M_g\mathbf{p} + M_{u_w}\mathbf{p} \circ M_{u_d}\mathbf{p}. \tag{14}$$

Eq. (14) is equivalent to Eq. (13) and shows how the calculation of the column of gas demand $\mathbf{P}$ is carried out for this parameterization of functions. The parameters of these functions are in column $\mathbf{p}$. The matrices M_i are constructed based on basis functions and time and temperature data. This calculation is fast, but it is nonlinear with respect to the parameters. This means that searching for the optimal values of the model parameters requires the use of nonlinear optimization techniques. It is, however, possible to analytically calculate the derivative of the forecasted values with respect to the model parameters. Therefore, the gradient optimization method can be used.

An extremely important property of this model is that it is easy to construct an initial point for the optimization process that searches for the optimal parameters of the model given the data. Having an initial point that is close to the global minimum is very beneficial, especially for nonlinear optimization. Such an initial point can be easily constructed here by fixing some parameters of the model and using linear optimization to find the non-fixed parameters. For example, one can fix the parameters that define functions n_w, n_d and u_d, which results to a linear model and the parameters of the other functions can be fitted to the data. This process can be iteratively repeated by fixing different parameters to find an estimation for all the parameters of the model. By this procedure an initial point is constructed that is close to the optimal one. This is the case because this model can be linearized with respect to the group of parameters by simply fixing the parameters of the model that are not in that group. Since linear optimization is easy to perform, such an initial point is easy to construct.

5.2. Two-reservoir model with linear memory

The fact that past temperatures are important guides us to upgrade the TRM to take past temperatures into account. But it is extremely difficult to include both the nonlinearity and the temperature dependency seen in Fig. 4 and the dependency based on

past temperatures. The number of parameters for the parameterization of a nonlinear function grows exponentially with the number of arguments. Having a multitude of past temperatures as arguments simply means that the number of parameters soon becomes too large.

It is desirable to keep the simple form of TRM (Eq. (14)) that is easy to calculate, analytically differentiable and with an easily constructed initial point. In the TRM case the calculation of $f(\mathbf{T})$ and $g(\mathbf{T})$ is a multiplication of the model parameters by a matrix. In the upgraded model this property needs to be retained.

One option is inspired by linear prediction. In this case the temperatures are transformed via convolution (marked with $*$) with some unknown response function $\mathbf{r}$. Let us take

$$f(\mathbf{T}) = \mathbf{r}_f * \mathbf{T} \quad \text{and} \quad g(\mathbf{T}) = \mathbf{r}_g * \mathbf{T}, \qquad (15)$$

where $\mathbf{r}_j$ holds the parameters of the model. The maps in Eq. (15) are linear with respect to the parameters. Therefore, the calculations of $f(\mathbf{T})$ and $g(\mathbf{T})$ are just multiplications with matrices. This is in accordance with the calculation scheme of the TRM. This model will be referred to as the two-reservoir model with linear memory (TLM). It is worth noting that this model cannot capture the nonlinear dependency seen in Fig. 4 because it is linear with respect to the temperature.

5.3. Two-reservoir model with nonlinear memory

The nonlinearity in the temperature dependency can be enforced if the response functions $\mathbf{r}_j$ are fixed (independent of the model parameters). With fixed temperature transformations the transformed temperatures can be nonlinearly mapped in the following way

$$f(\mathbf{T}) = \sum_j f_j(\mathbf{r}_j * \mathbf{T}), \qquad (16)$$

where the functions f_j are parameterized (hold the parameters of the model), while $\mathbf{r}_j$ are fixed response functions that are chosen beforehand.

This model will be referred to as the two-reservoir model with nonlinear memory (TNM). If the r_j functions are monotonically decreasing, the $\mathbf{r}_j * \mathbf{T}$ can be interpreted as the temperature average on some time scale. Therefore, the temperature dependency in Eq. (16) can be interpreted as the sum of the nonlinear functions of the temperature averages on different time scales.

The parameterization of the functions f_j and g_j is the same as for the f and g used in the TRM (Section 5.1). A good choice for the fixed response functions r_j are exponentially decreasing functions

$$r_j(t) = \exp(-t/\tau_j)/\tau_j \qquad (17)$$

that have a clear time scale τ_j and the convolution with such $\mathbf{r}_j$ can be computed with $\mathcal{O}(n)$ time complexity.

The TLM and TNM models are only upgraded versions of the TRM that include past temperatures without increasing the model's computational complexity. The upgraded models can be implemented with the same ease as the TRM and are analytically differentiable with respect to the model parameters, so gradient-optimization techniques can be used to search for the optimal model parameters. Also, the initial point for optimization can be constructed in the same way as explained in Section 5.1, which is extremely important for nonlinear optimization in a large dimensional space. To retain all these TRM properties, either the nonlinearity of the temperature dependency was sacrificed as in TLM or the temperature transformations were fixed as in TNM.

6. Implementation

The LR model was constructed using stepwise regression with the function stepwise fit from the octave environment [40]. The chosen p-values of the F-statistics to enter and remove the new regressors were set to 0.05 and 0.10, respectively.

For the KM and KMM models, the distribution density of the data points was constructed using a Gaussian kernel with a diagonal bandwidth. The optimal bandwidths were searched for using the plug-in selection method [41]. This means that the kernel bandwidths were not selected by the user, but constructed in an optimal way based on the data. The library ks [42] available in the programming language R [43] was used in the implementation of the algorithm. The constructed density estimation was employed to forecast future gas demand using Eq. (8).

The RNN model was trained using the root-mean-square propagation (RMSProp) algorithm and Keras [44] and Theano [45] library. The training for each forecasting horizon was conducted for 10^4 epochs. Without regularization neural networks tend to overfit which is indicated by the increase of test error during training. This effect was not observed in this case, presumably due to large training dataset. The progress of training and test error during RNN training is shown in Fig. 5. Gradient decent, however, was partially unstable because of the very long sequences (approx. 5000 points). The problem of instability was solved by restarting the optimization from already visited, stable regions. This means that when the gradient was calculated, which included special values (nan, inf), the gradient decent was restarted from a valid point that was visited recently by the same algorithm.

To search for optimal parameters of the empirical models the gradient-decent and Nelder-Mead methods were used. The functions fminunc and fminsearch from the octave environment [40] were used to implement them. Auxiliary methods for the gradient calculation and the initial point generation were written from scratch in an octave environment.

All the models were trained on a computer with an Intel i7-4500U processor and with 8 GB of RAM.

7. Results

The error of all the models increases with the forecasting horizon. This is because weather forecasts are less accurate for larger forecasting horizons. The comparison of errors for the different models is shown in Fig. 6. The models can be clearly ranked, with the RNN and LR models being the most accurate ones. Models of
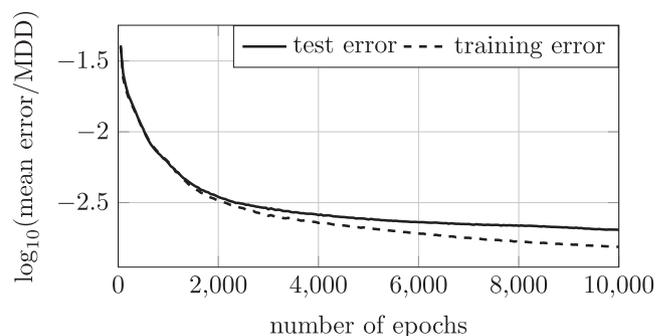
Fig. 5. Mean training and test error of the RNN model during training. The x-axis shows the number of times algorithm went through the entire data set during training (epoch).
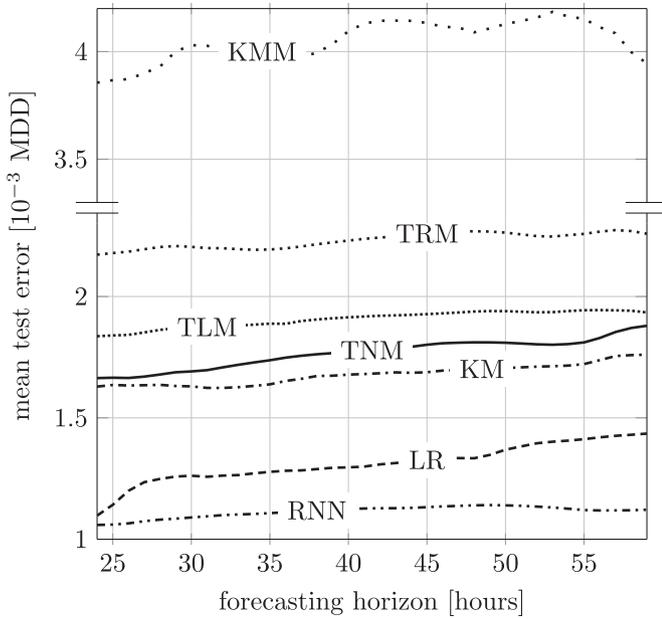
Fig. 6. Mean test error with respect to the forecasting horizon for different models.

Fig. 7. Ratio of errors when the model is trained on forecasted weather data and when it is trained on measured weather data. The KM and KMM models are not included because they were not trained on forecasted data due to the very long forecasting time.

medium accuracy are the KM and empirical models (TRM, TLM and TNM). The least accurate model is the KMM, which has much larger errors than the rest.

The KM model is approx. 2.2 times more accurate than the KMM model. This is despite the fact that the KMM model has access to past temperatures. This means that including more dimensions in the KM spreads the data points to such a degree that the larger models perform poorly. This is the result of the low density of points discussed in Section 4.2.

In the case that only daily gas demand forecasts are required, the errors of the models are smaller. This is because the errors for an hourly resolution balance each other out, so that the total error inside the 24-h window is smaller in proportion to the mean hourly error. Table 4 shows the errors of both versions of the forecasts for one day ahead, i.e., for a 24-h forecasting horizon. It must be noted that Fig. 6 shows the errors for all the forecasting horizons considered in this paper, while Table 4 shows the errors for one specific forecasting horizon. It also must be stressed that *MAE* reflects the cost of operation, while *MAPE* does not. Furthermore *MAPE* is a biased estimator that will overestimate the error when gas demand is low and put a heavier penalty on negative errors, than on positive errors.

The results show that the models trained on forecasted weather data can outperform the models trained on measured weather data. This was the case for the TRM, TNM and RNN models, while other models are more accurate if trained on measured weather

data. Fig. 7 shows the ratio of both errors for the different models. This is not anticipated because training on data with more noise should bring less, and not more, accuracy. The three models that are more accurate when trained on forecasted data are nonlinear with respect to temperature, while the LR and TLM models are not. This might indicate that nonlinear models are able to represent useful characteristics of forecasting noise to a greater degree than the other models.

The most untypical gas demand dynamics happens on public holidays. Fig. 8 shows the relative increase in the error during public holidays, compared to ordinary days. As expected, the KM model has the largest error increase on public holidays because the model does not include them. LR and TNM have the lowest increase in error, while others have a medium increase.
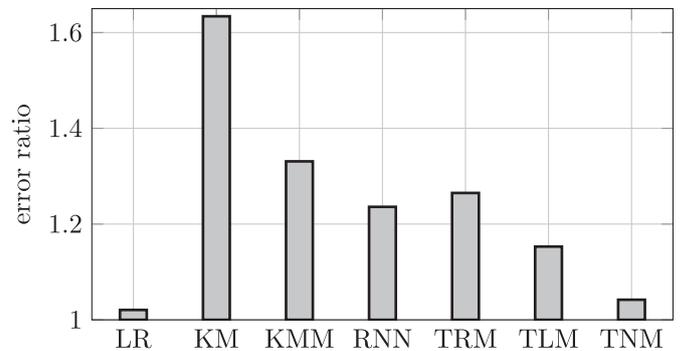
Fig. 8. Mean error on public holiday divided by the mean error on an ordinary day for different models. Public holidays are the most significant outliers.

Table 4
Errors of the models when hourly resolution is required (hour) and when daily resolution is required (day). Forecasting horizon is 24 h.

model	*MAE* [10^{-3} MDD]		*MAPE* [%]	
	hour	day	hour	day
LR	1.10	20.1	8.1	6.1
KM	1.62	27.5	14.0	9.7
KMM	3.87	89.3	42.4	38.5
RNN	1.06	18.3	9.3	6.8
TRM	2.17	32.9	19.8	11.2
TLM	1.84	32.5	15.4	10.5
TNM	1.66	25.2	15.8	8.6

Table 5
Training time for different models. Empirical models (TRM, TLM and TNM) were trained using Nelder-Mead (NM) or gradient decent (GD) optimization algorithm.

model	mean training time
LR	11.3 s
KM	2.4 h
KMM	9.1 h
RNN	5.6 h
TRM (NM)	33.8 s
TLM (NM)	7.1 min
TNM (NM)	8.8 min
TRM (GD)	1.1 s
TLM (GD)	1.6 s
TNM (GD)	3.6 s

The training times for each model are presented in Table 5. For empirical models it is easy to construct an initial point for optimization close to optimal one, which is why the training time is substantially shorter for those models. Training times for KM, KMM and RNN are, however, orders of magnitude larger compared to other models. But since training is only performed once, neither of those training times are problematic. A much greater problem appears when forecasting is very resource intensive, which is the case for the KM models. This is problematic because forecasting needs to be performed over and over again, e.g. on an hourly basis. The time needed to test the KM models was several orders of magnitude larger compared to the time needed to train them. For this reason training on forecasted data was not conducted for those models.

To search for the optimal parameters of the empirical models the gradient-decent and Nelder-Mead methods were used. It was found that the two optimization algorithms produce models of the same quality. For empirical models the values of the optimal parameters can actually help us understand the characteristics of the system (see Figs. 9 and 10). From this, one can deduce the laws for how this system behaves from the physical and social points of view. This cannot be done with black-box models from machine learning.

The most accurate forecast model in this paper is the RNN model. When hourly resolution is needed its error is, on average, 6.4% of the average gas demand in 1 h. If only daily gas demand forecast is required, the errors in the hourly resolution balance each
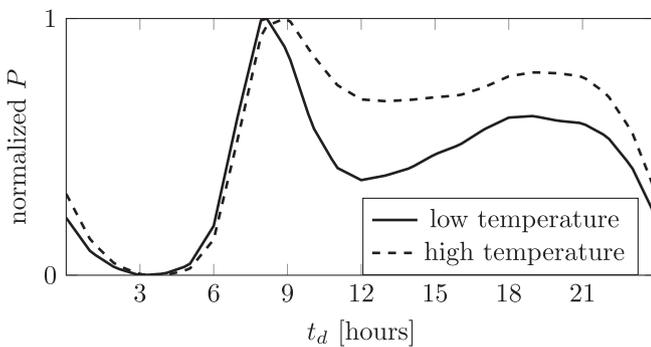
Fig. 9. How variable t_d influence the gas demand forecast of TNM. The graph shows normalized gas demand forecast (function P from Eq. (13) with fixed t_w and T) with respect to t_d for two temperature extremes.
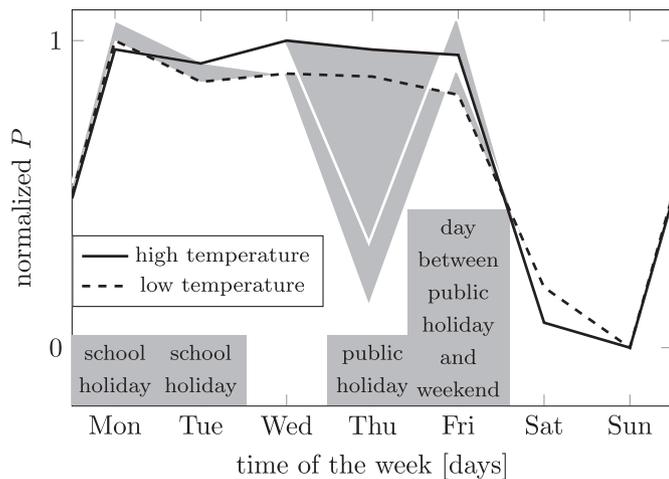
Fig. 10. How variables t_w, ph, sh and hw influence the gas demand forecast of TNM. The graph shows normalized gas demand forecast (function P from Eq. (13) with fixed t_d and T) with respect to t_w for two temperature extremes. The deviation due to the presence of occasional events (ph, sh and hw) is shown in gray.

other out and the average error drops to 5.4% of average gas demand in a day.

8. Discussion

Comparing the accuracies of very different models used in this paper can guide us to deduce the properties that an accurate gas demand forecast model should have. It is evident that a model is more accurate if it includes the influence of past temperatures. This claim is supported by the fact that the TLM is more accurate than the TRM and the LR model is more accurate than the KM. Furthermore, the model is more accurate if it is nonlinear with respect to temperature. Arguments for this claim are both the superiority of the TNM over the TLM and the RNN over the LR model. Also, it appears that past temperatures are more important than the temperature nonlinearity without the inclusion of past temperatures, which is indicated by the fact that LR is more accurate than the KM model.

This means that an accurate model should include past temperatures and, on top of that, it should be nonlinear with respect to temperatures. In Fig. 6, one can see clear exceptions to that claim. However, those exceptions can be explained by other properties of the models. One such exception is the superiority of the KM model over the KMM model, even though the KMM model includes past temperatures and the KM model does not. One way to explain this behavior is that the KMM state-space dimensionality might be too large for the number of data points available. This in turn causes a wider kernel and consequentially larger errors.

Another question is why LR is more accurate than the TNM model, even though the TNM model includes temperature nonlinearity and the LR does not. One explanation for this behavior is that the LR model is much easier to train compared to the TNM. It is known that LR training has a unique error minimum that is easy to find. On the other hand, TNM training might have a multitude of local error minima and there is no guarantee that the trained model is the optimal one.

Another explanation for the superiority of LR over the TNM model might be the TNM model structure which was hand-crafted based on historical data and theoretical considerations. Given the accuracies of empirical models it seems that it is very hard to predetermine overall model structure and the form of the nonlinearity in the model that would result to high accuracy. Hand-crafted models do not pay off in this case. This claim is further supported by the fact that even the KM model, which does not include past temperatures, is more accurate than the TNM model.

Empirical models are, however, useful because they give an insight into the characteristics of the system, as shown in Figs. 9 and 10. Unlike black-box models from machine learning, empirical models are crafted for this specific problem and the values of their parameters have a clear interpretation. If one is interested in discovering the qualitative laws of how the gas demand is governed, such empirical models can be of great help. For example, Fig. 10 shows that the presence of school holidays has an effect on gas demand only when temperatures are low. On the other hand, the impact of public holidays is independent of the temperature. Such knowledge is very hard to acquire using statistics because of the high dimensionality of the state space.

Models that are nonlinear with respect to temperature (TRM, TNM and RNN) become more accurate if the training is performed on forecasted temperatures. While others are more accurate when trained on measured temperatures. It seems that nonlinearity allows a model to find useful features in forecasted data that are not present in measured data. These features are useful because forecasts are based on both measured and forecasted temperatures. This also affects the error increases when the forecasting horizon

increases (see Fig. 6). In the case of the RNN model the error is practically constant, i.e., independent of the forecasting horizon, while the LR model has a notable increase in the error when the forecasting horizon is increased (approx. 32% increase).

The most accurate model in this paper is our implementation of the RNN model. This is in contrast to the findings of [17], where both the ANN and RNN models were less accurate than LR. The superiority of our RNN model can be attributed to advanced techniques used to implement it, such as the use of gated recurrent units and the use of a more suitable optimization algorithm for training the model. It should be noted that RNN has the disadvantage that the training is very resource intensive. Its implementation is also time consuming because of the recurrent loops and an instability that is a result of very long sequences.

The second most accurate model is LR, which is, unlike RNN, very easy to train and also easy to implement. Given that its accuracy is close to the RNN's, it is a sound choice. Therefore, the LR model should be tried first and if the accuracy of the model is satisfactory, there is no need for the resource-intensive implementation and training of the RNN model.

The third most accurate model is the KM, whose both training and forecasting are very resource intensive. In the case that extensive testing is required, the KM model is not the best choice because forecast generation is extremely time consuming.

Empirical models (TNM, TLM and TRM) are less accurate; however, their training time is very short. This can be attributed to the generation of initial points and the use of gradient-based optimization. However, the implementation of empirical models is time consuming because the methods for initial-point generation and gradient calculation need to be implemented from scratch.

The models in this paper have the largest error on occasional events like holidays. The LR model has the smallest error increase. For some forecasting horizons the LR error is even smaller than the RNN's on occasional events. This is unexpected, given that RNN has a larger capacity compared to LR. The superiority of LR for those events might be a consequence of the sub-optimal RNN training, given imbalanced data. In this respect it is a good idea to supplement the RNN model with the LR model on such occasional events, which results in an RNN-LR ensemble.

9. Conclusion

Given the high accuracy of some models presented in this paper, forecast models should be used more widely. Their use can reduce costs and improve the efficiency of the energy sector by improving the transportation of energy. The use of forecast models can also reduce overall energy consumption and CO_2 emissions. This is possible in the case of district heating and combined-heat-and-power plants where more optimal scheduling can bring substantial savings. An accurate forecast model can even eliminate the need for costly energy reserves altogether. Considering the wide variety of savings that are the result of the use of gas demand forecast models, they have the potential to reduce the cost of energy and its consumption and so reduce CO_2 emissions.

A less obvious trait of the models trained in this paper is their ability to model socially driven gas and in turn heat consumption within buildings. Therefore, such models can be used for the better optimized design of buildings and their heating systems. For example, it is possible to model both sources of natural heat (like the sun) and heat consumption with respect to a building's parameters. This allows us to find building parameters that minimize the heating costs using a computer simulation. Similarly, heat-consumption modeling allows us to optimize the control of heating systems more realistically because human behavior is also included in the models.

Gas demand models can also be used to recognize bad heating strategies practiced by humans. By analyzing the gas demand models one can find instances when humans consume unusually large amounts of heat. Knowing the circumstances under which this happens can help in advising about which heating practices are the most wasteful and costly.

The results of this paper show that the gas demand can be accurately modeled and forecasted. An accurate gas demand model should use past weather data and, on top of that, be nonlinear. The most accurate model found in this paper is the recurrent neural network and its accuracy is much higher if its training is conducted on forecasted weather data. In the case that implementation and computational resources are limited, a linear regression model is the best alternative. When following the guidelines stated in this paper one can train an accurate model that can be used to solve a variety of important problems found in the energy sector.

References

[1] Energy consumption in households. 2018. http://ec.europa.eu/eurostat/statistics-explained/index.php?title=Energy_consumption_in_households&oldid=369015. [Accessed 9 May 2018].

[2] Economidou M, Atanasiu B, Despret C, Maio J, Nolte I, Rapf O. Europe's buildings under the microscope. A country-by-country review of the energy performance of buildings. Tech. rep. Buildings Performance Institute Europe (BPIE); 2011.

[3] Soldo B. Forecasting natural gas consumption. Appl Energy 2012;92:26–37.

[4] Zhao H-X, Magoulès F. A review on the prediction of building energy consumption. Renew Sustain Energy Rev 2012;16(6):3586–92.

[5] Hong T, Fan S. Probabilistic electric load forecasting: a tutorial review. Int J Forecast 2016;32(3):914–38.

[6] Kalogirou SA, Bojic M. Artificial neural networks for the prediction of the energy consumption of a passive solar building. Energy 2000;25(5):479–91.

[7] Tso GK, Yau KK. Predicting electricity energy consumption: a comparison of regression analysis, decision tree and neural networks. Energy 2007;32(9):1761–8.

[8] Adom PK, Bekoe W. Conditional dynamic forecast of electrical energy consumption requirements in Ghana by 2020: a comparison of ARDL and PAM. Energy 2012;44(1):367–80.

[9] Vondráček J, Pelikán E, Konár O, Čermáková J, Eben K, Malý M, Brabec M. A statistical model for the estimation of natural gas consumption. Appl Energy 2008;85(5):362–70.

[10] Nguyen HT, Nabney IT. Short-term electricity demand and gas price forecasts using wavelet transforms and adaptive models. Energy 2010;35(9):3674–85.

[11] Tratar LF, Strmčnik E. The comparison of Holt–Winters method and multiple regression method: a case study. Energy 2016;109:266–76.

[12] Vaghefi A, Jafari M, Bisse E, Lu Y, Brouwer J. Modeling and forecasting of cooling and electricity load demand. Appl Energy 2014;136:186–96.

[13] Oğcu G, Demirel OF, Zaim S. Forecasting electricity consumption with neural networks and support vector regression. Proc Soc Behav Sci 2012;58:1576–85.

[14] Al-Shammari ET, Keivani A, Shamshirband S, Mostafaeipour A, Yee L, Petković D, Ch S. Prediction of heat load in district heating systems by support vector machine with firefly searching algorithm. Energy 2016;95:266–73.

[15] Protić M, Shamshirband S, Petković D, Abbasi A, Kiah MLM, Unar JA, Živković L, Raos M. Forecasting of consumers heat load in district heating systems using the support vector machine with a discrete wavelet transform algorithm. Energy 2015;87:343–51.

[16] Zhu L, Li M, Wu Q, Jiang L. Short-term natural gas demand prediction based on support vector regression with false neighbours filtered. Energy 2015;80:428–36.

[17] Potočnik P, Soldo B, Šimunović G, Šarić T, Jeromen A, Govekar E. Comparison of static and adaptive models for short-term residential natural gas forecasting in Croatia. Appl Energy 2014;129:94–103.

[18] Izadyar N, Ong HC, Shamshirband S, Ghadamian H, Tong CW. Intelligent forecasting of residential heating demand for the district heating system based on the monthly overall natural gas consumption. Energy Build 2015;104:208–14.

[19] Kato K, Sakawa M, Ishimaru K, Ushiro S, Shibano T. Heat load prediction through recurrent neural network in district heating and cooling systems. In: Systems, man and cybernetics, 2008. SMC 2008. IEEE international conference

on, IEEE; 2008. p. 1401—6.

[20] Song H, Qin AK, Salim FD. Multivariate electricity consumption prediction with extreme learning machine. In: Neural networks (IJCNN), 2016 international joint conference on, IEEE; 2016. p. 2313—20.

[21] Shamshirband S, Petković D, Enayatifar R, Abdullah AH, Marković D, Lee M, Ahmad R. Heat load prediction in district heating systems with adaptive neuro-fuzzy method. Renew Sustain Energy Rev 2015;48:760—7.

[22] El-Baz W, Tzscheutschler P. Short-term smart learning electrical load prediction algorithm for home energy management systems. Appl Energy 2015;147:10—9.

[23] Valgaev O, Kupzog F. Building power demand forecasting using k-nearest neighbors model-initial approach. In: Power and energy engineering conference (APPEEC), 2016 IEEE PES asia-pacific, IEEE; 2016. p. 1055—60.

[24] De Rosa M, Bianco V, Scarpa F, Tagliafico LA. Heating and cooling building energy demand evaluation; a simplified model and a modified degree days approach. Appl Energy 2014;128:217—29.

[25] Karabulut K, Alkan A, Yilmaz AS. Long term energy consumption forecasting using genetic programming. Math Comput Appl 2008;13(2):71—80.

[26] De Nadai M, van Someren M. Short-term anomaly detection in gas consumption through ARIMA and artificial neural network forecast. In: Environmental, energy and structural monitoring systems (EESMS), 2015 IEEE workshop on, IEEE; 2015. p. 250—5.

[27] Barak S, Sadegh SS. Forecasting energy consumption using ensemble ARIMA—ANFIS hybrid algorithm. Int J Electr Power Energy Syst 2016;82: 92—104.

[28] Amara F, Agbossou K, Dubé Y, Kelouwani S, Cardenas A. Estimation of temperature correlation with household electricity demand for forecasting application. In: Industrial electronics society, IECON 2016-42nd annual conference of the IEEE, IEEE; 2016. p. 3960—5.

[29] Hsu D. Comparison of integrated clustering methods for accurate and stable prediction of building energy consumption data. Appl Energy 2015;160: 153—63.

[30] Jovanović RŽ, Sretenović AA, Živković BD. Multistage ensemble of feedforward neural networks for prediction of heating energy consumption. Therm Sci 2015;20(4):1321—31.

[31] Fan C, Xiao F, Wang S. Development of prediction models for next-day building energy consumption and peak power demand using data mining techniques. Appl Energy 2014;127:1—10.

[32] Potočnik P, Govekar E. Applied forecasting of short-term natural gas consumption. In: Proc. 2011 19th IASTED international conference on applied simulation and modelling; 2011. p. 1—7.

[33] Spoladore A, Borelli D, Devia F, Mora F, Schenone C. Model for forecasting residential heat demand based on natural gas consumption and energy performance indicators. Appl Energy 2016;182:488—99.

[34] Draper NR, Smith H, Pownell E. Applied regression analysis, vol. 3. New York: Wiley; 1966.

[35] Hornik K. Approximation capabilities of multilayer feedforward networks. Neural Network 1991;4(2):251—7.

[36] Montufar GF, Pascanu R, Cho K, Bengio Y. On the number of linear regions of deep neural networks. In: Advances in neural information processing systems; 2014. p. 2924—32.

[37] Goodfellow I, Bengio Y, Courville A. Deep learning. MIT Press; 2016. http://www.deeplearningbook.org.

[38] Z. C. Lipton, J. Berkowitz, C. Elkan, A critical review of recurrent neural networks for sequence learning, arXiv preprint arXiv:1506.00019.

[39] J. Chung, C. Gulcehre, K. Cho, Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, arXiv preprint arXiv:1412.3555.

[40] Eaton JW, Bateman D, Hauberg S, Wehbring R. GNU Octave version 4.2.0 manual: a high-level interactive language for numerical computations. 2016. http://www.gnu.org/software/octave/doc/interpreter.

[41] Wand MP, Jones MC. Multivariate plug-in bandwidth selection. Comput Stat 1994;9(2):97—116.

[42] Duong T. ks: kernel Smoothing, R package. 2017., version 1.10.6. https://CRAN.R-project.org/package=ks.

[43] R Core Team. R: a language and environment for statistical computing. Vienna, Austria: R Foundation for Statistical Computing; 2016. https://www.R-project.org/.

[44] Chollet F. Keras. 2015. https://github.com/fchollet/keras.

[45] Theano Development Team, Theano: A Python framework for fast computation of mathematical expressions, arXiv e-prints abs/1605.02688. URL http://arxiv.org/abs/1605.02688.