A Data Science Solution to Integrate Weather Data for Energy Consumption Analysis

Thanh Huy Daniel Mai, Carson K. Leung ✉, Junyi Lu, Nathaniel Giesbrecht, Owen A. Hnylycia
Department of Computer Science, University of Manitoba, Winnipeg, MB, Canada
✉ Carson.Leung@UManitoba.ca

***Abstract*—For a modern grid to be reliable, energy efficiency and identifying consistent energy consumption patterns are crucial. In this paper, we present a data science solution that integrates weather data with historical energy consumption for energy consumption analysis. Consequently, it predicts temporal energy consumption patterns via techniques like frequent pattern mining, traditional machine learning, and deep learning. Our solution mines and forecasts energy consumption based on meteorological and environmental conditions over time series and examines how weather conditions affect energy usage variation. Evaluation results on a real-world dataset show that our solution identifies several distinct frequent patterns with frequent pattern mining, revealing a significant relationship between irradiance and energy consumption, as well as a positive correlation between temperature and energy usage. Moreover, our solution predicts and compares energy consumption for a specific year using decision tree, gradient boosting, linear regression, and random forest models with daily weather data. Additionally, we applied a long short-term memory (LSTM) model to analyze energy consumption as time-series data, uncovering patterns based on given time steps. These results demonstrate the practicality of our data science solution for energy consumption analysis.**

Keywords*—*data science, information reuse, information integration, energy, weather, meteorological data, environmental data, binning, machine learning, random forest, gradient boosting, deep learning, long short-term memory* (*LSTM*)*, data mining, frequent pattern mining

I. Introduction

In the current era of big data, data science solutions—which make good use of machine learning [1-3] (including deep learning [4]), data mining [5-10], mathematical and statistical techniques [11, 12], and visualization [13]—help reuse and/or integrate information in various real-life applications for public good. These include healthcare informatics [11], social network analysis [], transportation analytics [12-14], and predictive analytics []. In this paper, we focus on energy consumption analysis. The motivation behind mining and predicting energy consumption in the Canadian province of Ontario extends beyond just the immediate benefits for energy suppliers. By accurately forecasting energy demand, a wide range of stakeholders, including policymakers, consumers, and researchers, can benefit from the insights generated. By predicting energy demand, suppliers can optimize their energy generation and storage capabilities, ensuring a reliable supply of energy to consumers while minimizing costs and reducing carbon emissions. The results of this study could lead to a better understanding and prediction of energy consumption across Ontario based on variable weather conditions throughout the year. In this study, we used a combination of the frequent pattern mining to find and assess trends in energy consumption, and machine learning techniques to predict future trends in energy consumption based on weather variability. By leveraging the strengths of both approaches, we hope to generate novel insights and more accurate predictions that can help guide decision-making for energy suppliers, policymakers, consumers, and researchers alike.

Key contributions of our paper include our design and implementation of a data science system that consists of frequent pattern mining algorithm (e.g., Apriori, FP-growth), machine learning algorithms (e.g., linear regression, decision tree, random forest, gradient boosting regression) and a neural network (e.g., LSTM) algorithm. Such a system integrates information from weather data and energy consumption data. We applied the resulting data science system to mine and forecast energy consumption based on weather data.

The reminder of this paper is organized as follows. The next section provides background and related work. Section III starts presenting our data science system by describing its pre-processing step, mining step, and prediction step. Section IV shows our evaluation results on real-life energy consumption data in Ontario. Finally, conclusions are drawn in Section V.

II. Background and Related Works

A. Data Mining

There have been only a couple of other investigations into the link between weather and energy consumption using data mining. For instance, Kuo et al. [1] analyzed the energy consumption characteristics and affecting factors of Taiwan's convenience stores-using the big data mining approach. They looked at many variables (e.g., local climatic conditions of each store). Their analyses found that temperature and sunshine hours were the most influential climactic factors, although the significance of the impact was not consistent across locations. To mine the data, they used the WEKA data mining application. Ashouri et al. [4] developed building energy saving advisory by using] a data mining approach. They did not directly investigate the effect of weather but trying to control for it. The largest contributor that they controlled for was temperature. While there are many studies that use data mining to determine the effect of variables on household, many less use data mining approaches on weather data [2]. Blázquez et al. [3] examined residential energy usage using aggregated data. With the many papers connecting the weather to how energy is used, very few use a data mining approach. In contrast, we use both a machine

2835-5776/24/$31.00 ©2024 IEEE
DOI 10.1109/IRI62200.2024.00026

learning approach as well as a data mining approach. Through using both data mining and machine learning, we will also be able to determine the efficacy of using data mining.

B. Prediction

Many existing works have been done in an effort to produce prediction models for energy consumption. There are quite number of different models have been introduced, which are:

- Statistical models (e.g., K-means clustering [5], ARIMA [6], statistical model and its physical principles [7])
- Machine learning models (e.g., support vector machine (SVM) [8], fuzzy SVM [9], linear regression [10], artificial neural network (ANN) [11])
- Deep learning models (e.g., deep neural network [12], recurrent neural network (RNN) [13], auto-encoder [14])

Here, our approach is to experiment on different machine learning models and a LSTM model (which is a version of RNN model) to learn how well each of them perform on Ontario dataset.

III. Our Data Science System

A. Pre-Processing Data

Data are usually captured by a single CSV file recording energy data ordered by date/hour, paired with weather readings for that time slot. To run frequent pattern mining on this data, we first preprocess them. Since our data was hourly, energy readings for the day were additive. Because of this, we combined all 24 rows of each day into a single entry. The result of this combination lead to single date rows, where energy consumption was added to a total daily consumption, and the max values for weather data were taken. This was done using the python pandas library. We made sure to maintain the order by date during this process.

As data may be very precise (e.g., down to many decimal places), it can be challenging for frequent pattern mining algorithms to pick up the frequent patterns due to numerous unique precise values. To rectify this, we split our data into bins with labels ranging from very low to very high. We create two implementations, one with 6 bins per header, and one with 10 bins per header. The 6-bin implementation made our data less accurate but allowed us to raise our minimum support threshold when searching for frequent items. Conversely, the implementation with 10 bins made our data set more accurate but required us to have a low minimum support threshold (e.g., 10%) in order to extract interesting patterns.

The energy usage data are captured on an inconvenient scale, and we want the features to be on a similar scale and near to a normal distribution. We normalize the energy data to the interval [0, 1]. Min-max normalization is used to implement this concept:

$$x^* = \frac{x - \min(x)}{\mathrm{maax}(x) - \min(x)} \tag{1}$$

A factor that had a significant impact on our data was the North-East blackout of 2003. During this blackout, many parts of Ontario were without power for many hours. This caused a significant outlier in our dataset, as seen in Fig. 1. To ensure the integrity of our data, we ignored any dates that were affected by this event. While it would likely not affect the outcome of our frequent pattern mining since it is an infrequent anomaly, it had potential to affect our bin/label categorization and machine learning prediction model.

To give a quick overview of the data, we kept in our dataset post pre-processing. Moreover, we also list the headers to be kept and a brief description of each where necessary. These include: date, precipitation, temperature, irradiance surface: the measure of irradiance on earth's surface layer, top of atmosphere (TOA) irradiance (which is a measure of irradiance in earth's atmosphere), snowfall, snow depth, cloud coverage, air density, and energy consumption. It is important to note that the maximum values for all headers excluding energy consumption and snow depth were taken when we converted our data set to daily rows from hourly rows. Energy consumption and snow depth were additive values in the hourly data set, and thus they were summed and totaled during conversion.

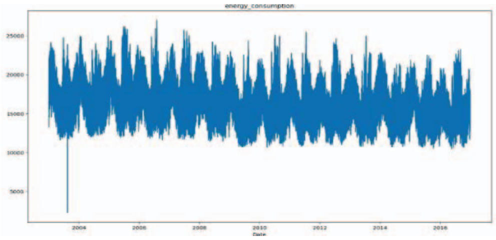


Fig. 1. Total energy consumption 2003-2016

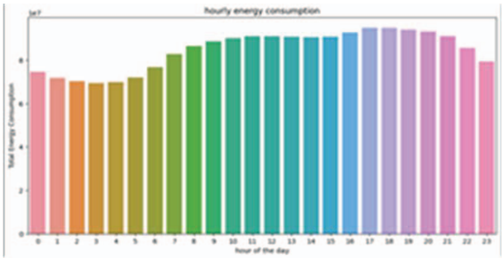


Fig. 2. Hourly energy consumption

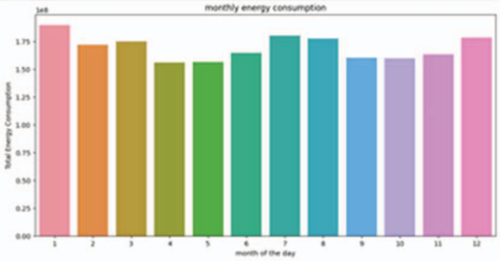


Fig. 3. Monthly energy consumption

Figs. 2 and 3 summarize the unmined energy consumption rates in our original data set. Observed from Fig. 2, energy consumption dips during the early hours of dawn, and then stays at a constant rate throughout the rest of the day. Observed from Fig. 3, energy consumption seems to be highest in the months that have the most extreme temperatures: January & December (extreme cold) and July & August (extreme hot). This correlation would be as expected based off our current knowledge of energy consumption. The months with extreme temperatures are the months where heating/cooling of buildings is most prevalent, and thus energy consumption would be expected to be higher.

B. *Mining Data*

To mine frequent patterns, our data science solution consider each row or transaction in the dataset, with Date/Hour being the transaction ID. Then, it creates bins and labels from the data. Afterwards, it assigns the values from the dataset into bins. Once these data are binned, it starts mining frequent patterns. We split numeric values into bins consisting of the labels (very low, low, high, very high). This not only made our data much easier to interpret, but also lead to much better frequent pattern generation. While this adjustment has manipulated the data in some way, we tried to mitigate as much risk as possible. For example, our cloud coverage bin for "very low" in our six-bin implementation contained all values between zero to twenty-five. This change does affect how frequent patterns are generated. Using precise data points may not have generated the latter frequent pattern, but using binned data points allowed us to generalize our results. Additionally, the binned data allows energy suppliers to view energy consumption rates in a range, rather than based on specific detailed values.

Binning. Due to the nature of data, they were continuous values with many significant figures. To get any frequent patterns an extremely low minimum support would have been necessary, and any frequent patterns would have arisen more based on an anomalous match of data points, than because of any real correlations. To extract meaningful results, we reduce the number of significant figures by placing the data into bins.

An advantage of using bins rather than just rounding values is that the points at which values are placed into bins can be adjusted away from round numbers. Deciding where to break bins apart also allows one to make the decision on how many bins to have, with just rounding we are limited to orders or magnitude. When creating bins for the data it is possible to decide the amount of precision desired, more bins create more precision but if taken to far will run into the same problems as the original data, whereas to few bins would remove any useful information from being extracted. When deciding how to separate bins, it also allows you to decide how to separate your bins. With this necessity to create bins, it then becomes important how you choose the bin sizes and values.

We split the bins at points that would result in the same number of elements in each bin, this approach was effectively binning on quantiles. We wrote a program that would give us the partition points of our data for any number of quantiles desired. This approach gave us more interesting frequent patterns than any other approach. When picking a minimum support with this method it is important to place it relative to the frequency of the bins and not in absolute terms. With this binning method we did have to consolidate some bins. With some of those data points, their distribution was heavily skewed to one extreme, precipitation and snow fall were some of these variables. For the variables that had quantiles with the same separating values or extraordinarily small differences between bins, we combined the values into a single bin. The binning decisions are ultimately guided by the ability to extract non-trivial information from the data.

Further optimization. Finally, the last step we took with mining our data was to narrow down our results even further. We found that our frequent pattern mining algorithm was returning results that was did not care about. For example, we were returned frequent item sets such as [Precipitation high, Cloud Coverage high]. The purpose of our study was to determine frequent item sets with relation between energy consumption and the weather. Thus, we decided to trim our frequent item sets, and frequent item sets such that they did not contain energy data.

Frequent pattern mining. Our frequent pattern mining algorithm, using the equally distributed bin method, resulted in thirty-one unique frequent patterns. From these patterns we can extract some useful information regarding energy consumption relating to weather conditions. For example, we can see a direct correlation between both top of atmosphere irradiance and energy consumption, as well as surface irradiance and energy consumption. When surface and/or top of atmosphere irradiance are categorized as *very low* we can see that energy consumption falls into the *extremely high* category. Conversely, when surface and/or top of atmosphere irradiance is *extremely high* we can see that energy consumption is categorized as *very low*. Further data collection would likely be required to draw any concrete conclusions, but some general assumptions can be made from this pattern. The correlation could be the cause of solar power usage, so when irradiance is high, less energy is used possibly due to the use of local solar energy rather than other forms of energy. This hypothesis is further supported by the expectations we would have when irradiance is high. Since many buildings in Ontario rely on air conditioning to cool the building in hot weather, we would expect energy consumption to be higher when irradiance is high (since when irradiance is high, we would expect temperature to rise). Seeing that energy consumption is lower with higher irradiance supports the idea that the lower energy consumption is due to solar power usage. Another hypothesis as to why energy usage is not higher, is that the hottest days occur late in the summer when the days have already started to shorten.

Both the extreme ends of temperature (very low and extremely high) were observed to correlate with high energy consumption, as we would expect to see. This is likely due to the energy cost of heating/cooling buildings during these extreme temperatures. Further, energy consumption during temperatures within these extreme ranges tends to stay on the lower end of the scale. From these, energy suppliers can expect increased power usage when the forecast predicts extreme hot or cold temperatures.

C. *Making Predictions*

Machine learning-based prediction. Common international practice for benchmarking energy consumption entails predicting and comparing various machine learning models based on building category [15]. This time, we extend the novel direction by predicting energy consumption based on varying environmental and meteorological conditions within the same region over time [16]. Meanwhile, we have also developed multiple energy consumption prediction models using machine learning and have prioritized probabilistic and artificial intelligence-based approaches to obtaining results. We will utilize the data headers provided as input data.

Consequently, we employed four machine learning models based on historical data on the meteorological environment: *linear regression, decision tree, random forest, and gradient boosting*. These models are applied to the region of Ontario and provide results for a particular energy data.

Long-Short Term Memory Model. Our motivation for choosing the LSTM model is to capture the long-term dependencies in the input sequences. As we can clearly notice that the energy each year follows a very similar pattern for post-COVID dataset (i.e., the dataset is from 2003 and 2016), which does not get affected with the information of energy during COVID time – Where people stayed at home much more and used significantly more energy than usual. Normally, the weights of our model are initialized to small values at the beginning of training, so the gradient quickly diminishes when it's computed with respect to inputs several time steps into the past, severely limiting our model's ability to learn long-term dependencies – this is commonly known as *vanishing gradients.* To combat the problem of vanishing gradients, Hochreiter and Schmidhuber [17] introduced the *long short-term memory (LSTM)* architecture. Moreover, as we observe, based on the dataset of energy consumption in Ontario, we want to apply the LSTM model on only the energy consumption (i.e., removing all weather attributes) – the point is to only capture the capture the long-term frequent pattern of the energy consumption over a year. The disadvantage of deep learning architectures is that they require a huge quantity of parameters, and are complex to train [18]. A unit of LSTM consists of several components. There are three types of gates in an LSTM model:

- Keep gate: determines which information from the previous time step should be kept.
- Input gate: decides which new information should be stored in the memory cell.
- Output gate: controls which information should be outputted to the next time step.

In addition to these gates, the LSTM model also has a memory cell that allows the network to maintain long-term dependencies [19]. Memory cells are the core components of the LSTM model, which holds the critical information that it has learned over time, and the network is designed to effectively maintain the useful information in the memory cell over many time steps. At every time step, the LSTM unit modifies the memory cell with new information with three different phases: keeping phase, writing phase, output phase, whose idea is related to each of the aforementioned gates.

D. Implementing Our Solution

Since the focus is on finding the frequent pattern (trend) of energy consumption, all the weather attributes were removed from our dataset. Therefore, a long sequence of hourly energy consumption is introduced and used to create an input-output pairs, which means that to predict an output energy at given time index *j,* we would use all the energy from time index $j - k - 1$, to $j - 1$ as our input data points. The following function is the one we create the input-output pair for our LSTM model.

The function *create_io_pair* takes two inputs: a sequence of energy data and a time step value *k*. It generates a set of (input, output) pairs from the given dataset, where the input is a sequence of *k* data points, and the output is the next data point after the sequence. The function starts by creating two empty lists, *dataX* and *dataY*, which will be used to store the number of input points and the expected output for that input. Then, it loops through the sequence of energy data using a for loop, with the range starting from 0 and going up to the length of the data minus *k* minus 1. Inside the for loop, the function extracts a sequence of *k* data points starting from the current index *i* up to *i+k* and appends this sequence to the *dataX* list. This sequence will serve as the input to the LSTM model. The function also extracts the next data point after the sequence (i.e., the data point at index *i+k*) and appends it to the *dataY* list. This value will serve as the output of the LSTM model. Finally, the function returns the input and output pairs as two numpy arrays, *dataX* and *dataY.*

In the implementation, four LSTM layers are added by using the sequential method from using Keras library along with its dropout rate. Deep neural networks with many parameters can cause overfitting, especially when the datasets are small [18]. Big improvements in model performance can be observed when dropout is applied to the model since dropout encourages each hidden unit to identify useful features without relying on other hidden units to do its correction [20]. A brief description of dropout is when a neural network model is updating its hidden layer where the dropout is applied, it arbitrarily does not update neurons in the layer [21].

IV. Evaluation

For frequent pattern mining, we vary the minimum support between 0.10 and 0.15 yielded us good results in terms of frequent item sets depending on the number of bins we created, meaning an item is frequent if it occurs in between 500-750 rows out of 5000 rows. The **best** results we found in our study occurred when we evenly distributed the bins and set the minimum support threshold to 4%. When the bins are evenly distributed, each bin will occur 1/bins of the time. With 6 bins each item will occur about 16% of the time. As each item can occur at most 16% of the time a frequent item can occur at most 16% of the time. We found that framing the minimum support not in absolute terms but relative to the maximum possible frequency helpful. Our minimum support in absolute terms is 4%, it is 25% of a relative maximum.

We make predictions using the same form of code, but we need a performance metric. Mean Absolute Error (MAE) is used to evaluate the four models. Namely, MAE is a performance metric used to evaluate the accuracy of regression models, including the Decision Tree Regressor, Linear Regression, Random Forest, and Gradient Boosting models, similar research in [22] used the same metric as us. MAE measures the average absolute difference between the predicted values (produced by the model) and the actual observed values (from the test dataset):

$$MAE = \frac{\sum_{i=1}^{n}|y_i - x_i|}{n} - \frac{\sum_{i=1}^{n}|e_i|}{n} \qquad (2)$$

which is an arithmetic average of the absolute errors $|e_i|=|y_i - x_i|$ where y_i is the prediction and x_i the true value.

A. Machine Learning Model for Hourly Prediction

We evaluated our machine learning models by using a training set capturing the first 13 years of data (from 2003 to 2015) and a testing set capturing the final year in the given data set (2016). There were 113,952 training examples with eight features (or columns) and 8,784 test examples with the same eight features:

- The X_train array, the input of the training data, has a shape of (113952, 8), meaning there are 113,952 rows (or examples) and 8 columns (or features).
- The y_train array, the output of the given input from training data, has a shape of (113952,), meaning it is a 1-dimensional array with 113,952 elements, corresponding to the labels for each of the training examples.
- The X_test array, the input of the test data, has a shape of (8784, 8), meaning there are 8,784 rows and 8 columns.
- The y_test array, the output of the given input from test data, has a shape of (8784,), meaning it is a 1-dimensional array with 8,784 elements, corresponding to the labels for each of the test examples.

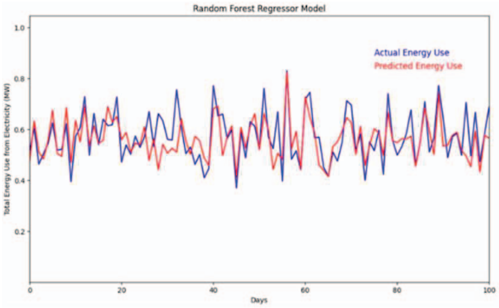
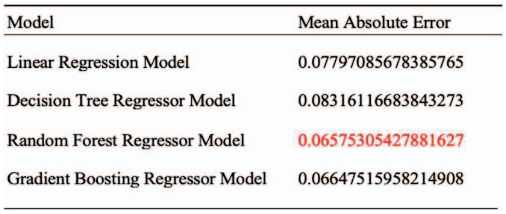


Fig. 4. Random forest model in 100 days.

Furthermore, we test in two different scenarios: with 2003 and without 2003 in the training dataset due to the North-East Blackout event.

Overall, based on Fig. 4 with only 100 days of 2016 data given that the training data, all models perform well in predicting the overall shape of the demand curve. However, the linear regression model did not do well at capturing large deviations, while the random forest and gradient boosting demonstrated a good ability to capture the variability in the data, and hence produce a more accurate prediction result. Based on the overall performance, there is no doubt that the best model to forecast for 2016 is the random forest regression model.

TABLE I. MAE OF 4 MODELS



Model	Mean Absolute Error
Linear Regression Model	0.07797085678385765
Decision Tree Regressor Model	0.08316116683843273
Random Forest Regressor Model	0.06575305427881627
Gradient Boosting Regressor Model	0.06647515958214908

As a result, the actual and predicted energy use is positively correlated between the two models. Also, it is evident that all the points in the random forest plot are closer to the line than in the gradient boosting regression, indicating that the random forest has superior performance.

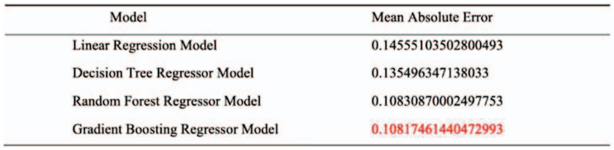
B. Machine Learning Model for Daily Prediction

We repeated the aforementioned evaluation on daily prediction. See Tables II-III.

TABLE II. SETUP

Training Features Shape (X_train)	(4748, 8)
Training Labels Shape (y_train)	(4748,)
Testing Features Shape (X_test)	(366, 8)
Testing Labels Shape (y_test)	(366,)

TABLE III. MAE OF 4 MODELS FOR DAILY PREDICTION



Model	Mean Absolute Error
Linear Regression Model	0.14555103502800493
Decision Tree Regressor Model	0.135496347138033
Random Forest Regressor Model	0.10830870002497753
Gradient Boosting Regressor Model	0.10817461440472993

C. LSTM for Daily Prediction

We will split the training and testing dataset as follows: 80% of the dataset is used for training, 20% of the dataset is used for testing. To summarize, there will be 5114 data points in total, the number of training data points will be 3274 and 921 data points for testing. After that, we apply the function to create the input-output pair with the time–step k = 100:

- There are 3173 training pairs – each of them will have 100 of the data-points to from time–step $j - k - 1$ to $j - 1$ as the input and 1 data-point as the output at time–step j.
- There are 921 testing pairs – to predict one output at time–step j, we will 100 of the data-points to from time–step $j - k - 1$ to $j - 1$.

For further improvement on the model, we tuned the hyper-parameter using GridSearchCV to test the effectiveness of number of units in layer (64, 128, 256), different Dropout rate (0.1, 0.2, 0.3) and two different optimizer methods (“rmsprop”, “adam”). The resulting best hyper-parameter is recorded as dropout rate of 0.3 with 256 units and Adam optimizer.

Overall, the LSTM has done a great job at capturing the pattern of the data with time–step k = 100. See Fig. 5.

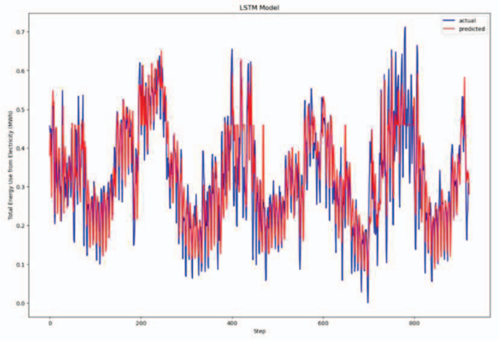


Fig. 5. LSTM with k=100 on test data for daily prediction

Moreover, we also evaluated the hourly consumption with dataset split into the training and testing dataset as follows: 80% of the dataset is used for training, 20% of the dataset is used for testing. To summarize, there will be 122736 data points in total, the number of training data points will be 78383 and 24378 data points for testing. After that, we apply the function to create the input-output pair with the time–step k = 168. Here, we choose 168 as our step since our purpose is to take the hourly consumption of seven days to predict the eighth day' energy consumption:

- There are 78383 training pairs – each of them will have 168 of the data-points to from time–step $j - k - 1$ to $j - 1$ as the input and 1 data-point as the output at time–step j.
- There are 24378 testing pairs – to predict one output at time–step j, we will 168 of the data-points to from time–step $j - k - 1$ to $j - 1$.

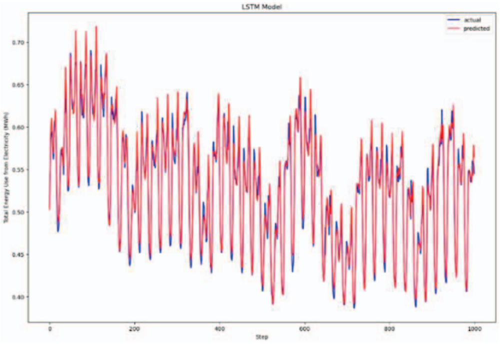


Fig. 6. LSTM for hourly prediction

Fig. 6 shows first 1,000 data points for better visualization. As we can see, they did a very good job of capturing the high variability of the hourly consumption. Moreover, they also give a good result on the MAE metric: 0.008067364346490155, which outperforms all the machine learning models.

V. Conclusions

The use of energy is very dependent on the weather, and so when predicting energy usage, it is a vital tool. In this paper, we presented our data science solution that mines and predicts energy consumption based on weather data. *It integrates information from both energy consumption data and weather data*. When applying our solution to real-life data, the random forest model and LSTM model were the best at predicting energy usage. Extreme temperatures resulted in high energy usage. Energy usage was bounded tighter to low temperatures and winter weather than with high temperatures and summer weather. We have shown that basic data mining techniques can give accurate and useful information on energy demand given weather. Moreover, the LSTM and random forest models are able to predict with precision on a shorter time frame. The use of several weather variables allowed us to get more accurate predictions than were possible with only temperature. As climate change continues to make extreme weather events a normal occurrence, data mining and machine learning models will become extraordinarily important to giving grid operators the ability to predict energy usage and keep the grid functioning. As *ongoing and future work*, we would explore other prediction models and techniques (e.g., incorporate frequent sequential mining in forecasting models).

Acknowledgment

This work is partially supported by NSERC (Canada) and University of Manitoba.

References

[1] M.S. Daoud, et al., "Enhancing intrusion detection systems accuracy using machine learning," IEEE SDS 2023, 103-106.

[2] S. Emmenegger, et al., "Mastering fencing techniques with machine learning: a video-based classification and correction system," IEEE SDS 2023, 120-127.

[3] L.G. Huber, et al., "Physics-informed machine learning for predictive maintenance: applied use-cases," IEEE SDS 2023, 66-72.

[4] C.K. Leung, et al., "AI-based sensor information fusion for supporting deep supervised learning," Sensors 19(6), 1345:1-1345:12, 2019.

[5] M.T. Alam, et al., "Discovering interesting patterns from hypergraphs," ACM TKDD 18(1), 32:1-32:34, 2024.

[6] M.T. Alam, et al., "Mining frequent patterns from hypergraph databases," PAKDD 2021, Part II, 3-15.

[7] M.A. Islam, et al., "Graph-based substructure pattern mining with edge-weight," Applied Intelligence 54, 3756-3785, 2024.

[8] C.K. Leung, "Frequent itemset mining with constraints," Encyclopedia of Database Systems, 2nd edn., 1531-1536, 2018.

[9] C.K. Leung, D.A. Brajczuk, "Efficient algorithms for mining constrained frequent patterns from uncertain data," ACM SIGKDD Workshop on Knowledge Discovery from Uncertain Data 2009 (KDD-U 2009), 9-18.

[10] R.A. Rizvee, et al., "A new tree-based approach to mine sequential patterns," ESWA 242, 122754:1-122754:28, 2024.

[11] L. Lac, et al., "Computational frameworks integrating deep learning and statistical models in mining multimodal omics data," Journal of Biomedical Informatics 152, 104629:1-104629:13, 2024.

[12] R. Langone, et al., "Interpretable anomaly prediction: predicting anomalous behavior in Industry 4.0 settings via regularized logistic regression tools," DKE 130, 101850, 2020.

[13] K.E. Barkwell, et al., "Big data visualisation and visual analytics for music data mining," IV 2018, 235-240.

[14] C.K. Leung, "Biomedical informatics: state of the art, challenges, and opportunities," BioMedInformatics 4(1), 89-97, 2024.

[15] X. Zhou, et al., "Deep learning-empowered big data analytics in biomedical applications and digital healthcare," IEEE/ACM TCBB 21(4), 2024.

[16] A. Alostad, et al., "An application to manage widespread social media accounts with one smart touch," SDS 2018, 176-181.

[17] D. Choudhery, C.K. Leung, "Social media mining: prediction of box office revenue," IDEAS 2017, 20-29.

[18] P. Howlader, et al., “Predicting Facebook-users' personality based on status and linguistic features via flexible regression analysis techniques,” ACM SAC 2018, 339-345.

[19] C.K. Leung, C.L. Carmichael, “Exploring social networks: a frequent pattern visualization approach,” IEEE SocialCom 2010, 419-424.

[20] C.K. Leung, et al., “Interactive discovery of influential friends from social networks,” Social Network Analysis and Mining 4(1), 154:1-154:13, 2014.

[21] A. Amirshahi, et al., “Predicting survey response with quotation-based modeling: a case study on favorability towards the United States,” IEEE SDS 2023, 1-8.

[22] R.C. Camara, et al., “Fuzzy logic-based data analytics on predicting the effect of hurricanes on the stock market,” FUZZ-IEEE 2018, 576-583.

[23] F. Saritas, et al., “Using acoustic signal to predict grain size of bedload particles,” SDS 2022, 59-64.

[24] C.-F. Jeffrey Kuo, C.-H. Lin, and M.-H. Lee, “Analyze the energy consumption characteristics and affecting factors of Taiwan's convenience stores-using the big data mining approach,” Energy and Buildings 168, pp. 120–136, Jun. 2018.

[25] J. Kang and D.M. Reiner, “Off seasons, holidays and extreme weather events: using data-mining techniques on smart meter and energy consumption data from China,” Energy Research & Social Science 89, p. 102637, Jul. 2022.

[26] L. Blázquez, N. Boogen, and M. Filippini, “Residential electricity demand in Spain: New empirical evidence using aggregate data,” Energy Economics 36, pp. 648–657, Jun. 2013.

[27] M. Ashouri, et al, “Development of building energy saving advisory: a data mining approach,” Energy and Buildings 172, pp. 139–151, 2018.

[28] Munz, G.; Li, S.; Carle, G. Traffic Anomaly Detection Using k-means Clustering. In Proceedings of the GI/ITG Workshop MMBnet, Hamburg, Germany, 13–14 September 2007; pp. 13–14.

[29] Kandananond, K. Forecasting electricity demand in thailand with an artificial neural network approach. Energies 2011, 4, 1246–1257.

[30] De Cauwer, C.; et al. Energy consumption prediction for electric vehicles based on real-world data. Energies 2015, 8, 8573–8593.

[31] Dong, B.; et al. Applying support vector machines to predict building energy consumption in tropical region. Energy Build. 2005, 37, 545–553.

[32] Xuemei, L.; Yuyan, D.; Lixing, D.; Liangzhong, J. Building cooling load forecasting using fuzzy support vector machine and fuzzy c-mean clustering. CCTAE 2010, pp. 438–441. Energies 2019, 12, 739

[33] Ma, Y.; Yu, J.Q.; Yang, C.Y.; Wang, L. Study on power energy consumption model for large-scale public building. ISA 2010; pp. 1–4.

[34] Ekici, B.B.; Aksoy, U.T. Prediction of Building Energy Consumption by Using Artificial Neural Networks. Adv. Eng. Softw. 2009, 40, 356–362.

[35] Ahmad, M.W.; Mourshed, M.; Rezgui, Y. Trees vs Neurons: Comparison between Random Forest and ANN for High-resolution Prediction of Building Energy Consumption. Energy Build. 2017, 147, 77–89.

[36] Lee, D.; Kang, S.; Shin, J. Using Deep Learning Techniques to Forecast Environmental Consumption Level. Sustainability 2017, 9, 1894.

[37] Li, C.; et al. Building Energy Consumption Prediction: An Extreme Deep Learning Approach. Energies 2017, 10, 1525.

[38] M. K. Kim, Y.-S. Kim, and J. Srebric, “Predictions of electricity consumption in a campus building using occupant rates and weather elements with sensitivity analysis: Artificial neural network vs. linear regression,” Sustainable Cities and Society 62, p. 102385, 2020.

[39] H. Pombeiro, et al., “Comparative assessment of low-complexity models to predict electricity consumption in an institutional building: linear regression vs. fuzzy modeling vs. neural networks,” Energy and Buildings 146, pp. 141–151, 2017.

[40] Sepp Hochreite, Jürgen Schmidhuber, “Long Short-term Memory”, Neural Computation, December 1997.

[41] X. Lü, T. Lu, C.J. Kibert, M. Viljanen Modeling and forecasting energy consumption for heterogeneous buildings using a physical–statistical approach Appl. Energy, 144 (2015), pp. 261-275.

[42] Nithin Buduma, Nikhil Buduma, Jor Papa, Nicholas Locascio, Fundamentals of Deep Learning – Designing Next Generation Machine Intelligence Algorithms, second edition. May 2022.

[43] Close G.E. Hinton, et al., Improving neural networks by preventing co-adaptation of feature detectors, CoRR, vol. abs/107.0580, 2012.

[44] Muhammad Faiq, et al., “Prediction of energy consumption in campus buildings using long short-term memory”, Alexandria Engineering Journal, p.65-76, 2023.

[45] Barinder Thind, et al., “Predicting Hourly Residential Electricity Demand in Ontario Using Signal Separation Approaches and Autoregressive Models”, Case study, Simon Fraser University