

Article

Intelligent Room-Based Identification of Electricity Consumption with an Ensemble Learning Method in Smart Energy

Vincent Le, Joshua Ramirez and Miltiadis Alamaniotis *

Department of Electrical and Computer Engineering, University of Texas at San Antonio, San Antonio, TX 78249, USA; Vincenle515@yahoo.com (V.L.); jramirez0777@gmail.com (J.R.)

* Correspondence: miltos.alamaniotis@utsa.edu; Tel.: +1-210-458-6237

Abstract: This paper frames itself in the realm of smart energy technologies that can be utilized to satisfy the electricity demand of consumers. In this environment, demand response programs and the intelligent management of energy consumption that are offered by utility providers will play a significant role in implementing smart energy. One of the approaches to implementing smart energy is to analyze consumption data and provide targeted contracts to consumers based on their individual consumption characteristics. To that end, the identification of individual consumption features is important for suppliers and utilities. Given the complexity of smart home load profiles, an appliance-based identification is nearly impossible. In this paper, we propose a different approach by grouping appliances based on their rooms; thus, we provide a room-based identification of energy consumption. To this end, this paper presents and tests an intelligent consumption identification methodology, that can be implemented in the form of an ensemble of artificial intelligence tools. The ensemble, which comprises four convolutional neural networks (CNNs) and four k-nearest neighbor (KNN) algorithms, is fed with smart submeter data and outputs the identified type of room in a given dwelling. Results obtained from real-world data exhibit the superiority of the ensemble, with respect to accuracy, as compared with individual CNN and KNN models.

Keywords: consumption identification; room consumption; ensemble learning; CNN; KNN; smart metering



Citation: Le, V.; Ramirez, J.; Alamaniotis, M. Intelligent Room-Based Identification of Electricity Consumption with an Ensemble Learning Method in Smart Energy. *Energies* **2021**, *14*, 6717. <https://doi.org/10.3390/en14206717>

Academic Editor: Muhammad Aziz

Received: 24 August 2021
Accepted: 7 October 2021
Published: 15 October 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the rapid growth of technology and the increasing presence it has in peoples' lives, the significance of finding ways to efficiently monitor and deliver power is becoming more prominent. Although society currently has the capability to generate energy that meets high demands, such processes tend to result in excessive power losses [1]. To help abate this issue, the concept of smart energy was introduced, bringing intelligent technologies and data collection systems into smart homes. In addition, the advent of IoT devices makes the continuous networked communication of a consumer's utility usage possible [2]. This collection of information opens opportunities for a given individual to have better insight into their consumption patterns, make adjustments that will reduce unnecessary load draws, or even detect anomalies in their household's power usage [3]. This information can also be made accessible to power grid utilities, enabling them to offer tailored demand response and management programs that reduce or shift energy consumption; additionally, this information can aid power grid utilities in securing grid operations [4].

The aim of this process is to achieve a relationship where a consumer's daily energy consumption can be predicted by their suppliers and utilities. These, in turn, would control the amount of power to send the consumer's way, ensuring optimal 24/7 operation of the power grid. One approach by which this relationship might be established is via bilateral

contracts between consumer and utility. Through these processes, in the long term, power losses and costs can be minimized [5].

Artificial intelligence (AI) algorithms can be utilised to make robust and accurate consumption predictions for people across different regions [6]. Currently, smart homes have submeters that periodically collect data on how much energy is being drawn from the power grid [7]. Submeters collect a substantial amount of data, and AI models can be trained to identify patterns and make predictions from this large dataset [8]. For instance, using data about the time of day and the kilowatt-hour used, an AI model would be able to identify a specific room in a given smart home. By extension, this could even lead to making load forecasts of the resident's consumption patterns for the next few days, weeks, or the coming year [9–11].

Previous works have centered on non-invasive load monitoring; a method of extrapolating the load draw of individual appliances from a single submeter [12]. In Ref. [13], a very deep one-dimensional convolutional neural network (VDOCNN) was used to classify the power signatures of various appliances from one submeter. The work carried out in [14] aimed to achieve similar results, but with the use of a k-nearest neighbor algorithm. The uses of a DNN and an LSTM in making load forecasts for the energy system of a given residential district were studied in [15]. A similar load forecasting technique was used in [16]; however, this study used a support vector machine (SVM). Sambandam et al. [17] disaggregated the power delivered into a house and compared the individual accuracies in classifying individual appliances, as obtained by a naive Bayes classifier, an SVM, and a KNN. In all these cases, the experiments yielded decent accuracies; however, each of the studies exhibits a high reliance on the prediction and performance of a single model. There is potential for more robust and consistent load predictions when the classifications of multiple models are accounted for, especially in regions that exhibit large temperature variations.

In this work, we present a new approach for load identification by departing from the traditional appliance-based load identification in a consumer's profile and proposing a room-based consumption identification. Faustine et al. [18] conclude that appliance-level breakdown does not lead to quantifiable energy efficiency. In fact, [19] found that more appliances increased the complexity of the disaggregation process. Kelly et al. [20] determined that a broader power analysis could lead to significant average energy reduction. As such, the proposed approach implicitly groups the appliances and provides an upper-level identification that may be used in non-intrusive load monitoring [21]. To this end, an ensemble learning method is proposed for room classification. The ensemble learning method employs a majority voting system that has been proven to obtain consistently correct predictions and classifications [22]. The inputs for the ensemble method are the watt-hour data obtained from several submeters. Through training a convolutional neural network (CNN) [23] and a k-nearest neighbor (KNN) algorithm [24] on pre-processed data obtained from a resident in Paris (France), the model performs an ensemble on nine different classifiers to provide the room type [25]. The contributions of this paper are summarized as follows:

- (i) A novel approach implementing a room-based energy consumption identification.
- (ii) The development and application of a new intelligent ensemble method in load identification.

The rest of this paper is split into sections as follows. Section 2 presents the methodology developed for classification. Section 3 provides a comparison between the results of our methodology and other related works. Section 4 concludes the article.

2. Methodology

2.1. Data Pre-Processing

The UCI Machine Learning Repository has a comprehensive dataset that was collected from a resident in Paris, France [26], whose home was equipped with smart submeters. Three submeters were used to monitor the load draw from the kitchen, the laundry room,

and the HVAC/water heater room. Over the course of four years, each submeter collected data for every minute. The resulting dataset contained well over two million samples of watt-hour energy. Figure 1 displays a sample of the recorded data for a randomly selected day in March 2008.

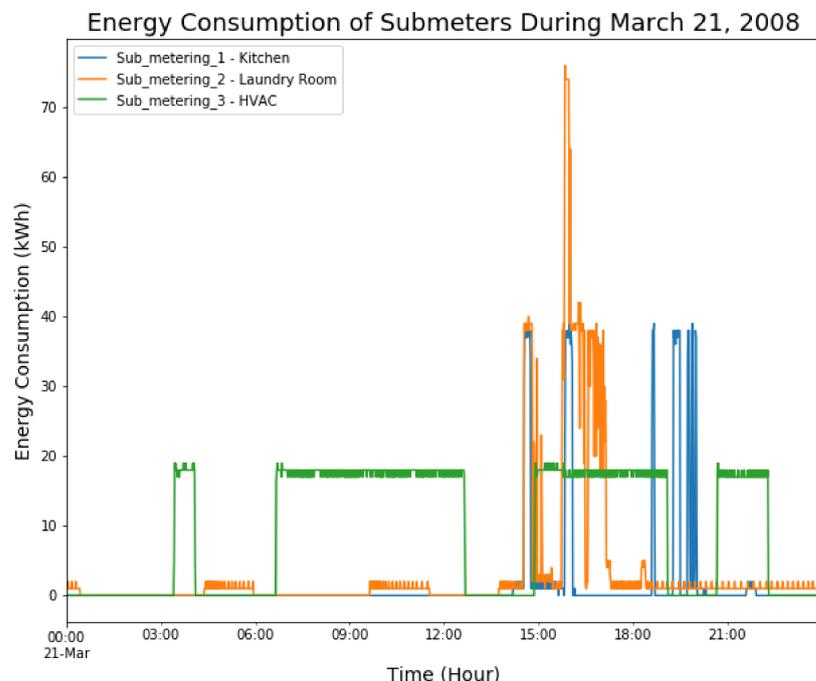


Figure 1. Sample of energy usage data acquired from the submeters.

In order better to account for the variations in power usage, which are mostly due to variation in temperature, the dataset is split into four sub-datasets that correlate with the seasons across the four years. For the models, data from 2006 to 2009 were used for training and validation, while the data from 2010 were used for testing the performance of the trained model. Each model received the watt-hour usage that was sampled every minute over a three-hour period, resulting in 180 data points. Using one-hour intervals of the data was avoided, since the zero power consumption in early hours of the day would cause difficulties in training the model. Instead, three-hour intervals were used to improve training. Additionally, data formatted in this manner allow the model to be more robust and to ignore any minor abnormalities in power usage that occur over a few minutes.

The CNN model uses all data points from the 180 min as its input to make a prediction, while the KNN model uses the average of the 180 min, as well as the start time of the interval. Both of these models are explained in more detail in the next two subsections.

2.2. K-Nearest Neighbor Algorithm

The k-nearest neighbor (KNN) model accelerates the process of clustering data points into groups that each represent a label for the data. In this work, KNN is used as a classification tool. In general, the inputted data points are compared to the k-nearest data points, which subsequently group themselves with whichever class is most represented.

To train the KNN model, a set of qualitatively labeled data is plotted in a two-dimensional space based on its quantitative features. When a new data point, $M_{in}(x_{in}, y_{in})$, is entered into the model, the specified K number of closest data points are compared to M_{in} . The distance is D . The training data point is $Y_K(x_K, y_K)$, which is from M_{in} and is calculated based on Euclidean distance, as shown in Equation (1), where Y_K is a data point

being measured from M_{in} . x_{in} and y_{in} are the x and y coordinates of the input point, and x_K and y_K are the x and y coordinates of the measured data point [21] as follows:

$$D(M_{in}, Y_K) = \sqrt{(x_{in} - x_K)^2 + (y_{in} - y_K)^2}. \quad (1)$$

M_{in} will group the K number of training data points into a cluster, and M_{in} will classify itself as the class that is most represented in the group. In a case where all classes are equally represented in the cluster, M_{in} is randomly classified into one of the classes. In most cases, the K parameter is tuned to find a value that yields the highest overall accuracy of the model. In this work, the label or class will be the room that the model has classified for any given data points of time and power.

Using the seasonal data, four KNN models were created and trained using data from 2006 to 2009 for each specific season. The data from 2006 to 2009 were split into train–test datasets, where 70% of the data was used for training and 30% was used for testing. The data from 2010 were used as a holdout dataset and used to determine the performance of the models on unseen data.

2.3. Convolutional Neural Network Model

Convolutional neural networks (CNN) excel in extracting features from temporal or spatial data. These models are feed-forward multilayer neural networks with multidimensional inputs. The inputs are transformed as they go through each of the hidden layers; resulting in an output with extracted features that are mapped to a label. The beginning of the process involves the convolution layer, where a convolutional kernel (having the same width as the time series data) slides over the input data to convolve every element in the vector through the following equation:

$$s(t) = (x * w)(t) = \sum_{a=-\infty}^{\infty} x(a)w(t - a) \quad (2)$$

where x is the input, w is the kernel, and t is time [27]. The steps following the convolution involve reducing the dimensionality of the mapping through max pooling, concatenating the output of the convolution layers through the process of flattening, and then proceeding through the multilayer feed-forward network for classification [28].

The publicly available software TensorFlow and Keras were used to generate the four seasonal CNN models [29]. The model architecture, seen in Figure 2, remained the same for all the CNN models. The only two parameters that varied between the models were the batch size and the number of epochs; these parameters varied because each model had different seasonal input data, and because the quality of data varied between models. For training and validating the model, a 70–30 split was used. The 2010 dataset was used as a holdout dataset, in order to measure the performance of the CNN models on data that had not been seen by the model.

2.4. Ensemble Learning

The eight independent models were joined together to implement ensemble learning. Figure 3 displays the flow of data in our proposed model. The output of each model determines which submeter the data belong to. The model then uses a majority vote, or obtains the mode of each prediction. The submeter with the highest count will be the final output. Given that there are eight models in the entire system, it is possible that there will be a tie between four models. In this case, we propose the following approach: the chosen output prediction should be that of the submeter that is not submeter 3. For example, if submeter 1 and submeter 3 are tied, submeter 1 should be chosen. The same will procedure should be followed if submeter 2 and submeter 3 are tied. Lastly, if submeter 1 and submeter 2 are tied, submeter 1 should be chosen.

This method for solving a tie was chosen because submeter 3 was likely to be over-predicted due to the large number of small watt-hour readings. If submeter 1 was inputted

into the model and it was an early morning reading where most appliances were not being used, the model would likely predict submeter 3 because a majority of its readings are small or 0.

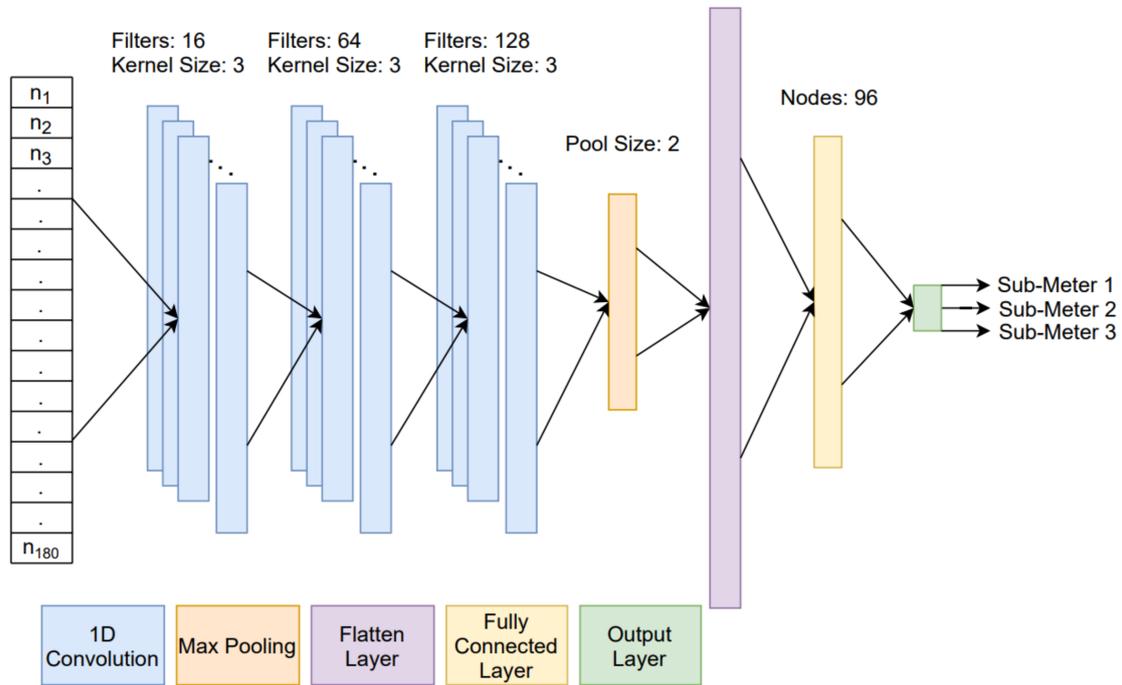


Figure 2. CNN model architecture.

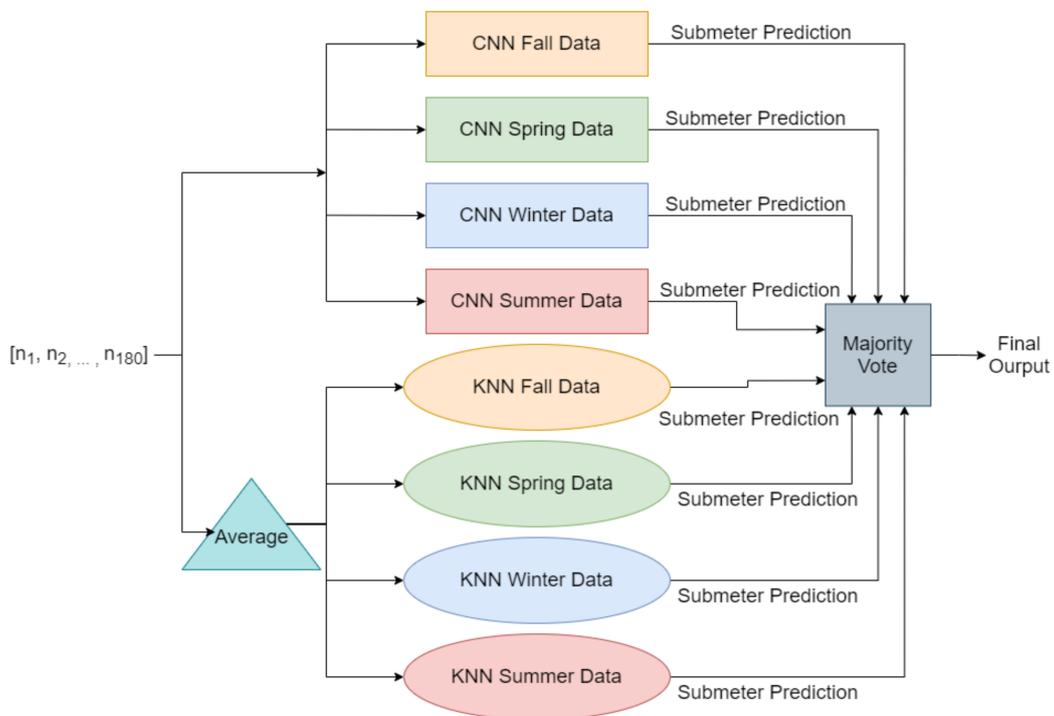


Figure 3. Ensemble methodology: the array input represents the watt-hour data from the UCI database; n_1 is the first data point of the 3-h interval; n_{180} is the last data point of the 3-h interval.

3. Results

3.1. CNN Model Results

The training for the CNN models varied between ten and thirty epochs, depending on the season, since some of the models took longer than others to minimize the loss. Figures 4 and 5 show the training accuracy and loss after each epoch. Table 1 shows the final training results for each of the CNN models. When using the 2010 holdout dataset for the specific season, the fall dataset had the lowest accuracy of 84.7% and the summer model had the highest accuracy of 98.6%. One observable trend is in the correlation between accuracy and the number of epochs in the models that performed well versus the models that did not. A higher number of epochs often leads to a higher final accuracy. The training was stopped when the validation loss and the accuracy began to plateau.

Table 1. Final CNN model results.

	Fall	Spring	Winter	Summer
Epochs	30	15	20	10
Batch Size	#	#	#	#
Final Testing Accuracy/Final Loss	91.4%/0.2505	93.2%/0.1950	90.1%/0.2704	91.9%/0.257
Final Validation Accuracy/Final Loss	92.4%/0.276	92.3%/0.214	90.2%/0.265	91.1%/0.286
Final Testing Accuracy	84.7%	94.5%	85.7%	98.6%

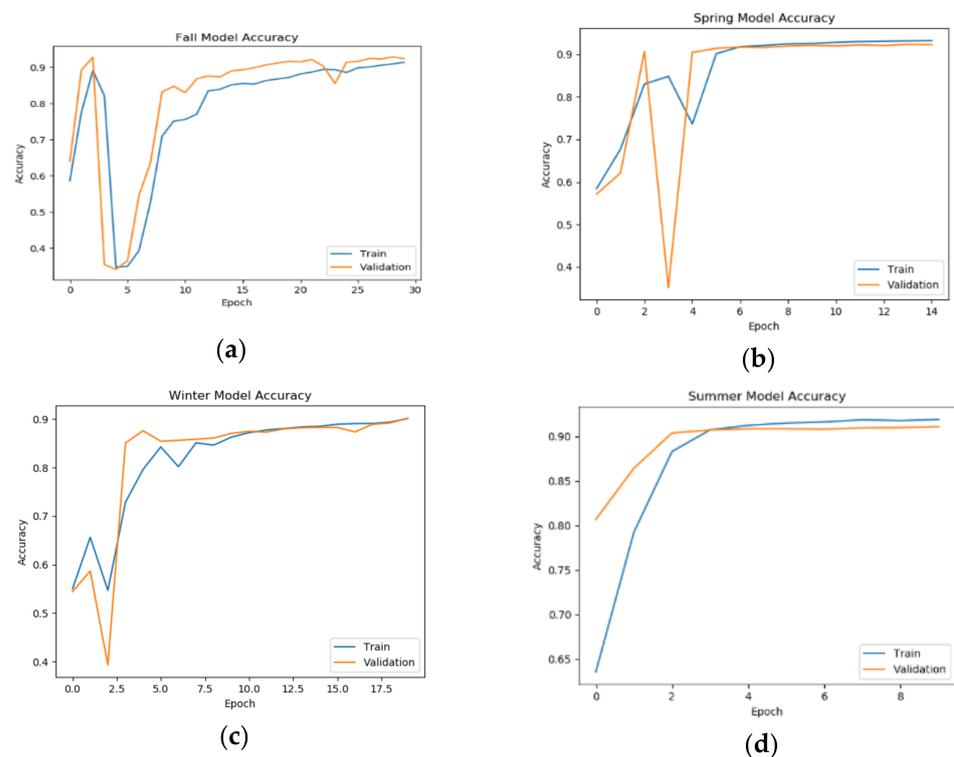


Figure 4. CNN models' training and validation accuracy: (a) fall model training accuracy; (b) spring model training accuracy; (c) winter model training accuracy; (d) summer model training accuracy.

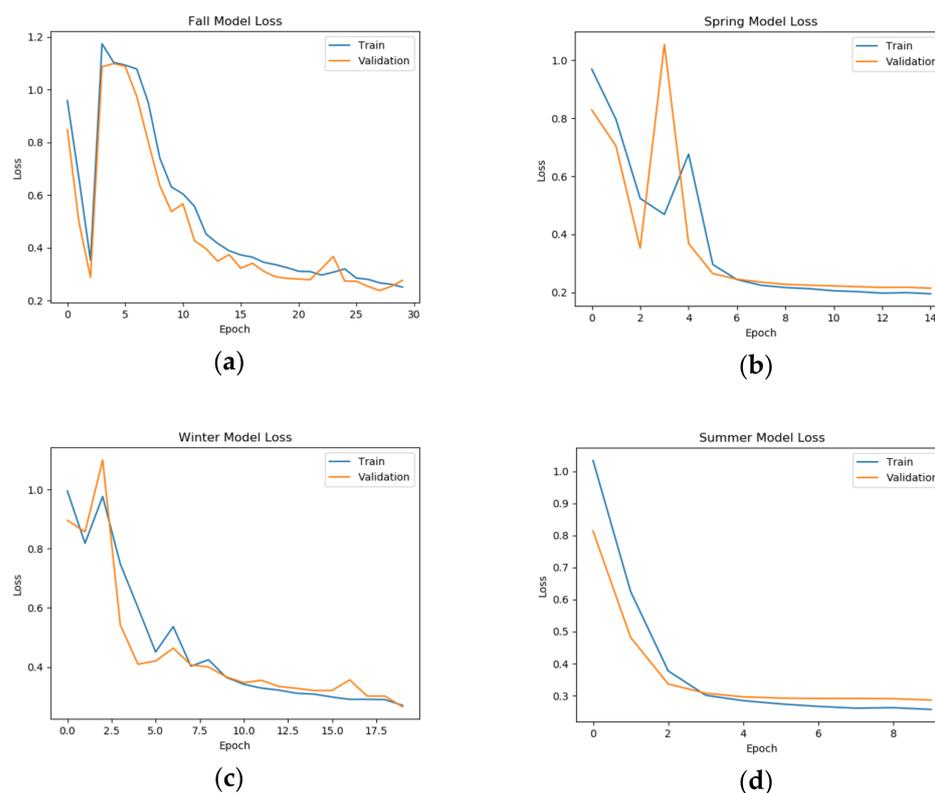


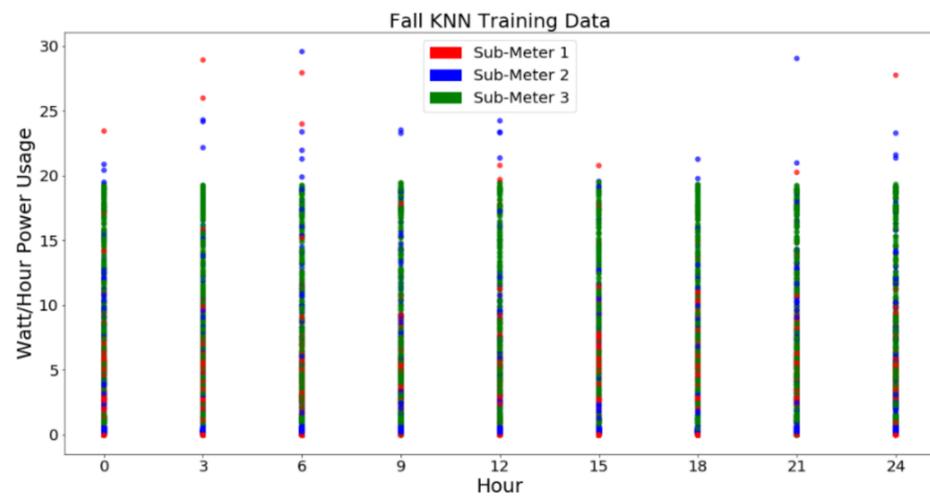
Figure 5. CNN models' training and validation loss: (a) fall model training loss; (b) spring model training loss; (c) winter model training loss; (d) summer model training loss.

3.2. KNN Model Results

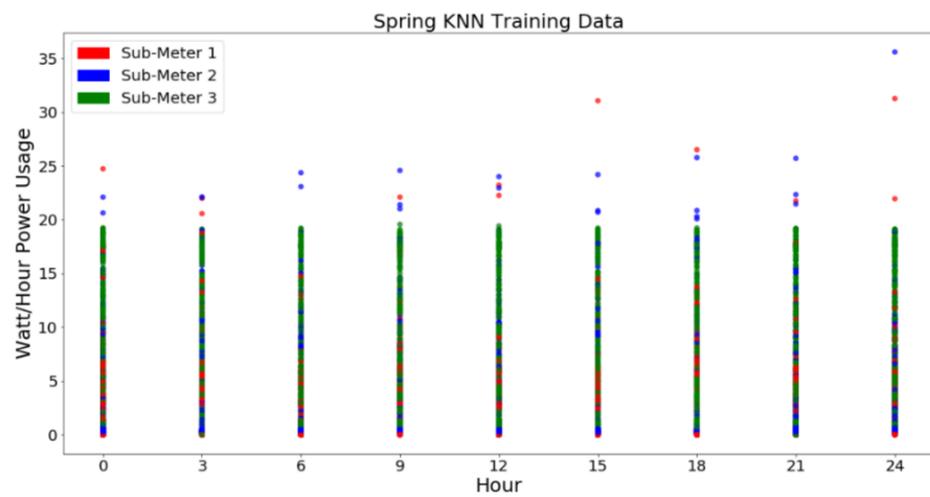
The KNN models were trained using the data shown in Figure 6. Each KNN model had different K neighbor values, as shown in Table 2. With the differences in the seasonal data, the values were not expected to be the same. From the K values in our models, a higher value would mean that the submeters had a large number of energy usages that were similar to each other. Table 2 shows the final accuracy metrics of these models. The winter model had the lowest accuracy of 80.3% and the summer model had the highest accuracy of 81.7%.

Table 2. Final KNN model results.

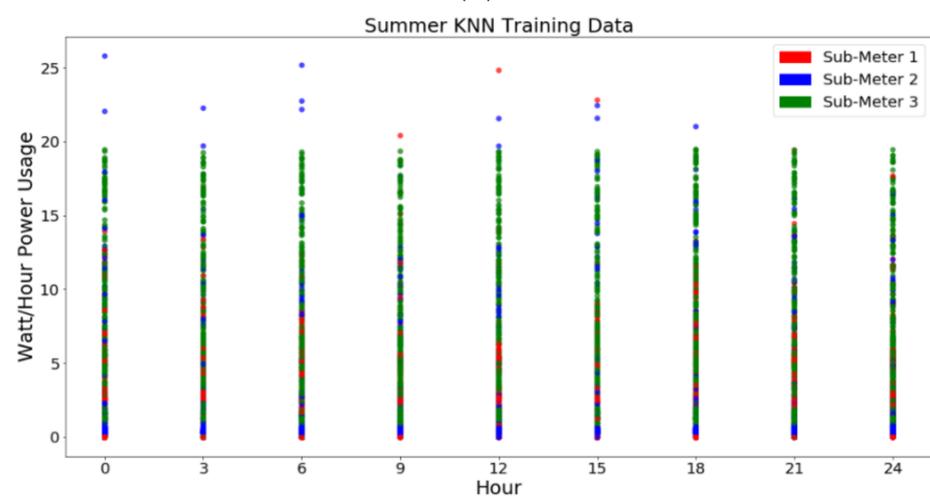
	Fall	Spring	Winter	Summer
K Neighbors	27	37	19	27
Validation Accuracy	83.1%	78.8%	78.3%	80.7%
Testing Accuracy	81.1%	80.8%	80.3%	81.7%



(a)



(b)



(c)

Figure 6. Cont.

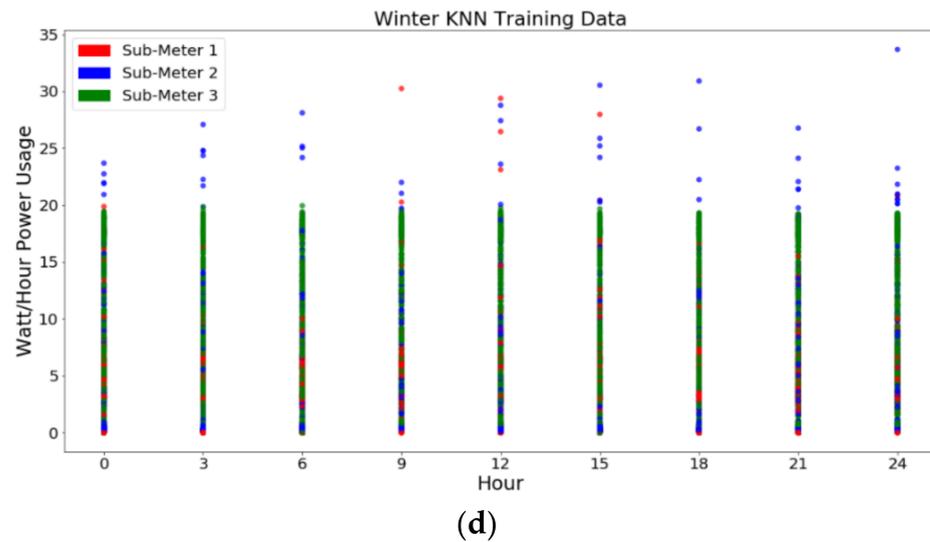
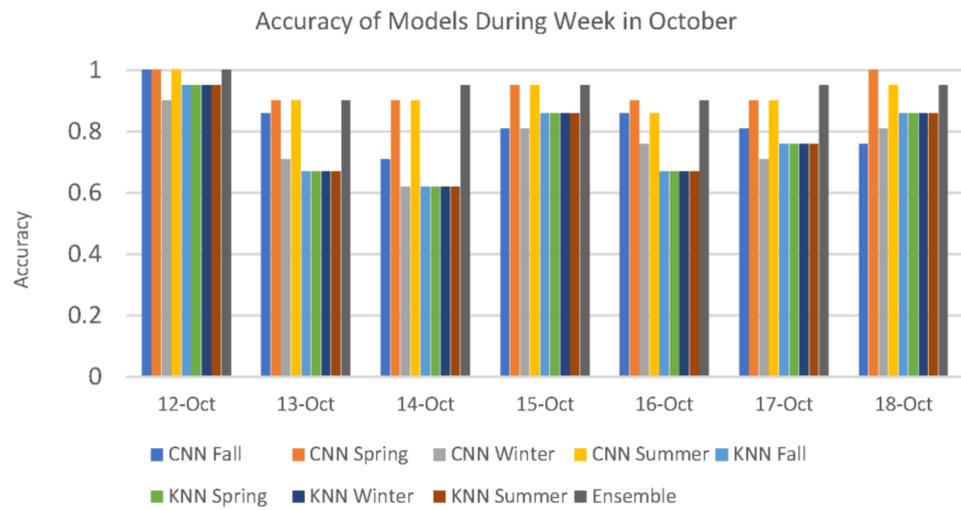


Figure 6. KNN data distribution among all the seasons: (a) fall data distribution; (b) spring data distribution; (c) winter data distribution; (d) summer data distribution.

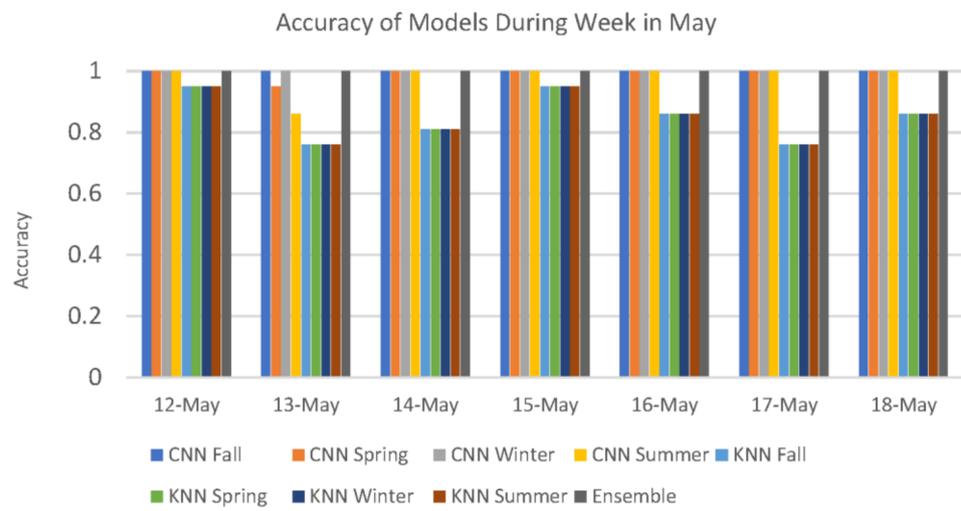
3.3. Ensemble Model Results

In the CNN and the KNN models, the summer models performed the best; this is most likely due to the consistency of the data points, e.g., the constant HVAC usage. During other seasons with cooler temperatures, submeter 3 would read zero or a low value for the energy usage, so the models probably over-predicted submeter 3 whenever it showed low energy readings.

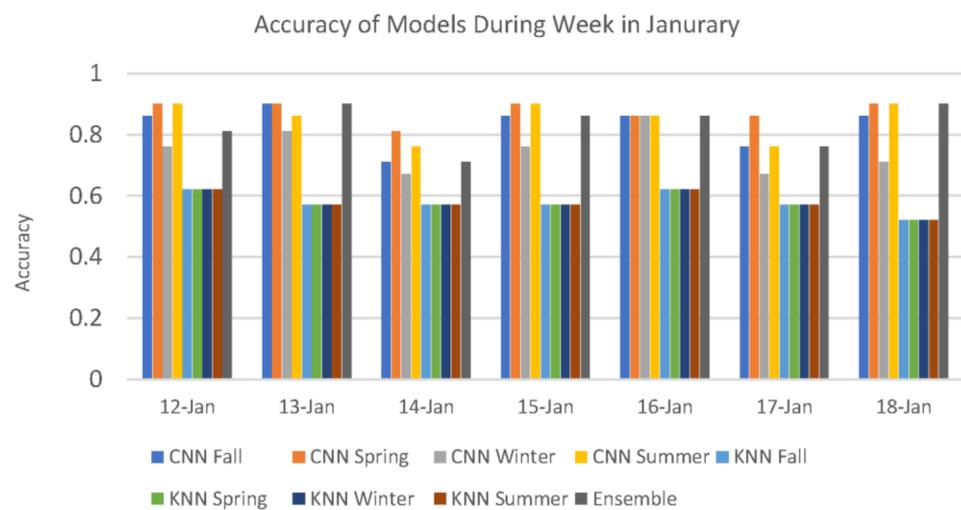
Lastly, the models were combined to use ensemble learning to create an output. For each prediction made, a majority vote was used to determine the final prediction of the models. To determine the overall performance of the ensemble learning model, a daily prediction was made for one week during the approximated middle month of each season: October for fall, January for winter, April for spring, and July for summer. The daily accuracy of each model is plotted in Figure 7. Out of the twenty-eight predictions made with the ensemble learning model, twenty-three of the days showed the highest accuracy among all the models; thus, in most cases, the ensemble model outperformed the singular, independent CNN and KNN models.



(a)



(b)



(c)

Figure 7. Cont.

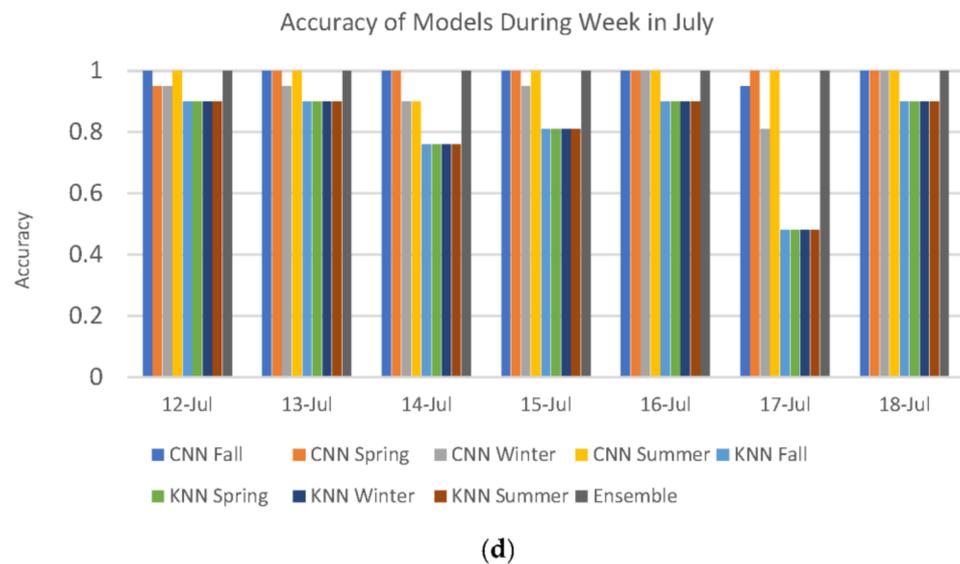


Figure 7. Model accuracies during various seasons: (a) model accuracies during a week of October using all models; (b) model accuracies during a week of May using all models; (c) model accuracies during a week of January using all models; (d) model accuracies during a week of July using all models.

4. Conclusions

In this paper, a robust method for classifying rooms is proposed, based on power consumption and the time of day. Through the implementation of ensemble learning, predictions were made more robust and consistent by combining the predictions from multiple models. The models that made up the ensemble model in this work were a convolutional neural network and a k-nearest neighbor algorithm.

To better account for variations in power usage, especially in regions that exhibit large variations in temperature, the dataset was pre-processed into four smaller datasets that represented the seasons throughout the four-year timespan. The CNN and KNN models were then trained on each season, respectively, giving the eight predictions for the ensemble learning model. A majority voting mechanism was used to implement the ensemble model. As a result, the ensemble model was proven to outperform the predictions of the individual models in the majority of cases; thus, the proposed ensemble learning method was proven to enhance the performance of the overall model.

Since the model handled the classification well, a future work to pursue would be to explore forecasting algorithms equipped to make near- or long-term load forecasts. In this way, power grid suppliers would benefit by being equipped to ensure optimal 24/7 power distribution throughout any part of the year. Moreover, on a consumer's end, the opportunity arises to change their power consumption habits when this information is provided to them.

With regard to ensemble creation, in the future, we plan to modify the ensemble model by employing and testing other machine learning tools and record their efficiency. Furthermore, we plan to extensively test whether the knowledge about the variation in energy demands over time contributes significantly to energy savings, and if so, how our ensemble models may contribute to that.

Author Contributions: Conceptualization, M.A.; methodology, V.L. and J.R.; data preparation, J.R.; code development, V.L. and J.R.; draft preparation, V.L. and J.R.; draft review, V.L, J.R and M.A.; supervision, M.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are publicly available at UCI Machine Learning Repository, under the title “Individual household electric power consumption Data Set”.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Tsoukalas, L.H.; Gao, R. Inventing energy internet: The role of anticipation in human-centered energy distribution and utilization. In Proceedings of the IEEE SICE Annual Conference, Tokyo, Japan, 20–22 August 2008; pp. 399–403.
2. Belligianni, F.; Alamaniotis, M.; Fevgas, A.; Tsompanopoulou, P.; Bozani, P.; Tsoukalas, L.H. An Internet of Things Architecture for Preserving Privacy of Energy Consumption. In Proceedings of the 10th Mediterranean Conference on Power Generation, Transmission, Distribution and Energy Conversion (Med Power 2016), Transmission, Belgrade, Serbia, 6–9 November 2016; pp. 1–7.
3. Alamaniotis, M.; Tsoukalas, L.H. Anticipation of Minutes-Ahead Household Active Power Consumption Using Gaussian Processes. In Proceedings of the International Conference on Information, Intelligence, Systems and Applications, Corfu, Greece, 6–8 July 2015; pp. 1–6.
4. Alamaniotis, M. Multi-kernel Analysis Paradigm Implementing the Learning from Loads Approach for Smart Power Systems. In *Machine Learning Paradigms*; Tsihrintzis, G., Virvou, M., Sakkopoulos, E., Jain, L., Eds.; Springer: Berlin/Heidelberg, Germany, 2019; Chapter 8, pp. 131–148.
5. Alamaniotis, M. Fuzzy Leaky Bucket System for Intelligent Management of Consumer Electricity Elastic Load in Smart Grids. *Front. Artif. Intell. Fuzzy Syst.* **2020**, *3*, 1–14.
6. Kuster, C.; Rezgui, Y.; Mourshed, M. Electrical load forecasting models: A critical systematic review. *Sustain. Cities Soc.* **2017**, *35*, 257–270. [[CrossRef](#)]
7. Gupta, D.; Hazarika, B.B.; Berlin, M.; Sharma, U.M.; Mishra, K. Artificial intelligence for suspended sediment load prediction: A review. *Environ. Earth Sci.* **2021**, *80*, 1–39. [[CrossRef](#)]
8. Ayub, N.; Irfan, M.; Awais, M.; Ali, U.; Ali, T.; Hamdi, M.; Alghamdi, A.; Muhammad, F. Big Data Analytics for Short and Medium-Term Electricity Load Forecasting Using an AI Techniques Ensembler. *Energies* **2020**, *13*, 5193. [[CrossRef](#)]
9. Alamaniotis, M.; Ikononopoulos, A.; Tsoukalas, L.H. Evolutionary Multiobjective Optimization of Kernel-based Very Short-Term Load Forecasting. *IEEE Trans. Power Syst.* **2012**, *27*, 1477–1484. [[CrossRef](#)]
10. Shirzadi, N.; Nizami, A.; Khazen, M.; Nik-Bakht, M. Medium-term regional electricity load forecasting through machine learning and deep learning. *Designs* **2021**, *5*, 27. [[CrossRef](#)]
11. Sánchez-Durán, R.; Luque, J.; Barbancho, J. Long-term demand forecasting in a scenario of energy transition. *Energies* **2019**, *12*, 3095. [[CrossRef](#)]
12. Zoha, A.; Gluhak, A.; Imran, M.A.; Rajasegarar, S. Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey. *Sensors* **2012**, *12*, 16838–16866. [[CrossRef](#)] [[PubMed](#)]
13. Dash, P.; Naik, K. A Very Deep One Dimensional Convolutional Neural Network (VDOCNN) for Appliance Power Signature Classification. In Proceedings of the 2018 IEEE Electrical Power and Energy Conference (EPEC), Toronto, ON, Canada, 10–11 October 2018. [[CrossRef](#)]
14. Khan, M.M.; Siddique, M.A.; Sakib, S. Non-Intrusive Electrical Appliances Monitoring and Classification using K-Nearest Neighbors. In Proceedings of the 2nd International Conference on Innovation in Engineering and Technology (ICIET), Dhaka, Bangladesh, 23–24 December 2019; pp. 1–5.
15. Wang, Q.; Li, C.; Zhang, Y.; Chen, L.; Liu, L. LSTM Model for Various Types of Load Forecasting in Energy System Integration. In Proceedings of the 12th IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC), Nanjing, China, 20–23 September 2020; pp. 1–4.
16. Kim, S.; Jung, S.; Baek, S.-M. A Model for Predicting Energy Usage Pattern Types with Energy Consumption Information According to the Behaviors of Single-Person Households in South Korea. *Sustainability* **2019**, *11*, 245. [[CrossRef](#)]
17. Faustine, A.; Mvungi, N.H.; Kaijage, S.; Michael, K. A Survey on NonIntrusive Load Monitoring Methodies and Techniques for Energy Disaggregation Problem. *arXiv* **2017**, arXiv:1703.00785.
18. Egarter, D.; Pochacker, M.; Elmenreich, W. Complexity of Power Draws for Load Deggregation. *arXiv* **2015**, arXiv:1501.02954.
19. Kelly, J.; Knottenbelt, W. Does disaggregated electricity feedback reduce domestic electricity consumption? A systematic review of the literature. *arXiv* **2016**, arXiv:1605.00962.
20. Alamaniotis, M.; Tsoukalas, L.H.; Buckner, M. Privacy-driven electricity group demand response in smart cities using particle swarm optimization. In Proceedings of the IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI), San Jose, CA, USA, 6–8 November 2016; pp. 946–953.
21. Hansen, L.K.; Salamon, P. Neural network ensembles. *IEEE Trans. Pattern Anal. Mach. Intell.* **1990**, *12*, 993–1001. [[CrossRef](#)]
22. Wu, Q.; Wang, F. Concatenate convolutional neural networks for non-intrusive load monitoring across complex background. *Energies* **2019**, *12*, 1572. [[CrossRef](#)]
23. Fan, G.F.; Guo, Y.H.; Zheng, J.M.; Hong, W.C. Application of the weighted k-nearest neighbor algorithm for short-term load forecasting. *Energies* **2019**, *12*, 916. [[CrossRef](#)]

24. Tan, M.; Yuan, S.; Li, S.; Su, Y.; Li, H.; He, F. Ultra-short-term industrial power demand forecasting using LSTM based hybrid ensemble learning. *IEEE Trans. Power Syst.* **2020**, *35*, 2937–2948. [[CrossRef](#)]
25. UCI Machine Learning Repository: Individual Household Electric Power Consumption Data Set. 2020. Available online: <https://archive.ics.uci.edu/ml/datasets/individual+household+electric+power+consumption> (accessed on 13 December 2020).
26. Rothman, D. *Artificial Intelligence by Example: Acquire Advanced AI, Machine Learning, and Deep Learning Design Skills*; Packt: Birmingham, UK, 2020.
27. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning (Adaptive Computation and Machine Learning series), Illustrated ed.*; The MIT Press: Cambridge, MA, USA, 2016.
28. Jin, H.; Wang, H.; Tian, F.; Zhao, C. An EEMD and convolutional neural network based fault diagnosis method in intelligent power plant. In Proceedings of the 2019 Chinese Automation Congress (CAC), Hangzhou, China, 22–24 November 2019.
29. Géron, A. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*; O'Reilly Media: Sebastopol, CA, USA, 2019.