IoT Augmented Physical Scale Model of a Suburban Home

Thomas Burns†, Gregory Fichthorn‡, Sharare Zehtabian*, Salih Safa Bacanlı*, Mina Razghandi*
Ladislau Bölöni*, Damla Turgut*
†Department of Computer Science, Rutgers University
‡Department of Computer Science, Stetson University
*Department of Computer Science, University of Central Florida
tjburns28@gmail.com, gfichthorn@stetson.edu, sharare.zehtabian@knights.ucf.edu,
{sbacanli, mrazghan, lboloni, turgut}@cs.ucf.edu

Abstract— **Green homes require informed energy management decisions. For instance, it is preferable that a comfortable internal temperature is achieved through natural, energy-efficient means such as opening doors or lowering shades as opposed to turning on the air conditioning. This requires the control agent to understand the complex system dynamics of the home: will opening the window raise or lower the temperature in this particular situation? Unfortunately, developing mathematical models of a suburban home situated in its natural environment is a significant challenge, while performing real-world experiments is costly, takes a long time and depends on external circumstances beyond the control of the experimenter.**

In this paper, we describe the architecture of a physical, small scale model of a suburban home and its immediate exterior environment. Specific scenarios can be enacted using Internet of Things (IoT) actuators that control the doors and windows. We use a suite of IoT sensors to collect data during the scenario. We use deep learning-based temporal regression models to make predictions about the impact of specific actions on the temperature and humidity in the home.

Index Terms: **Smart home modeling, internet of things, temperature regulation.**

I. Introduction

Residential and commercial buildings represent about 40% of the total energy consumption in the United States [1]. Thus, making more energy efficient and environmentally responsible decisions in homes can have a significant economic and environmental impact.

Green home technologies aim to find more energy efficient ways to manage the temperature, humidity, light and other metrics in the home without sacrificing user comfort. For instance, whenever possible, the thermal management of the home should be achieved through natural means such as opening doors or lowering shades. Such actions can be contrasted with energy intensive approaches such as central air conditioning. Such approaches not only reduce the environmental impact of energy generation, but also have direct economic benefits to the inhabitants. If we calculate with the average US energy cost to the customer, 13.2 cents per kWh cost, running an air conditioning unit for eight hours a day costs an average suburban home between one hundred ten dollars and two hundred forty five dollars per month [2].

In order for a green home to take the optimal actions that lower cost and minimize environmental impact, it needs a high quality predictive model of the various actions that can be taken. Due to the complex interactions between internal and external factors, the geometry of the home and the actuators, such a predictive model is difficult to build. For instance, opening a window at night might lower the temperature, while during the day it might raise it, especially if the window is on the sunny side of the home. Naturally, this depends on the location (Arizona versus Minnesota), season and other factors (is the home shaded by a tree?).

In this paper, we describe the architecture of a small scale physical model of a suburban home, together with an enclosure that models the exterior environment. Specific scenarios can be enacted using Internet of Things (IoT) actuators which control the doors and windows. We use a suite of IoT sensors to collect data during the scenario. We use deep learning based temporal regression models to make predictions about the effect of specific actions affecting temperature and humidity in the home.

The remainder of this paper is organized as follows. Section II review related work. Section III explains the scaled smart home design and data collection. Experiment details can be find in Section IV. Results are demonstrated in Section V. Finally, the paper ends in Section VI with a conclusion.

II. Related Work

Lin et al. [3] examined the relationship between in-home behavior and indoor air quality based on the data collected from smart home sensors and chemical indoor air quality measurements. This was done by collecting data from two smart homes and analyzing the impact of smart home behavior on indoor air quality, as well as the relationship between different groups of smart home features and indoor air quality variables. For data analysis, random forest, linear regression, and support vector regression machine learning classifiers were used. The study concludes that there is a strong relationship between in-home human behavior and air quality and that this observation could be generalized across multiple smart homes. The temperature was found to be the most frequently selected feature. The temperature changes within the homes were caused by multiple human activities, making it the most impactful feature within the dataset.

Lee et al. [4] developed a virtual smart home as well as an agent behavior-based simulation model. The virtual smart home and the sensors within the house were created in Unity. They designed a human-like virtual agent which acted based on a motivation-based behavior planning model. Using the simulated agent and virtual sensors together, they were able to verify the smart home structure and the arrangement of the sensors within the house. They concluded that this was a cheaper alternative for researchers to simulate a configurable smart home environment that enables autonomous agent generation.

Jin et al. [5] designed a prediction model for the optimization of power consumption for heaters within a smart home environment. They achieved this through the use of a recurrent neural network (RNN) and long short term memory (LSTM) which utilized datasets of temperature and humidity collected inside the house for their prediction models of energy consumption. They combined these models with a predicted comfort index to optimize power consumption. The results showed that the proposed optimization scheme saves energy as well as providing a comfortable environment at the user-desired temperature and humidity.

Mateo et al. [6] used different forms of regression to predict temperature within a larger building. They were able to use machine learning techniques to accurately predict the temperature with an average error of about 0.1°C. This study applied to larger buildings and only indirectly addresses the prediction of temperatures within a smart home.

Chen and Irwin [7] present Weatherman, a model that analyzes energy consumption data as well as wind and solar generation data to predict where on earth a set of coarse energy consumption data has occurred. Their analysis takes advantage of the idea of the distinct weather signatures that appear in different environments around the world. This research applies to the construction of different earth environments and more specifically their energy consumption.

Teich et al. [8] explore and present a prototype of a neural network to maintain comfortable temperatures in the smart home in an energy-efficient way. The presented model automatically supervises and re-trains its components based on activities within the home. This allows for the simplification of the tenant's lives through the home's automated services.

Kim et al. [9] propose different probability-based algorithms for the recognition of human activities while it could be applied to other domains such as healthcare. The research focuses on the recognition of patterns of behavior, multiple behaviors at once, and the ambiguity of different actions. This theoretical research relates to human activity and modeling patterns of such within a home environment utilizing machine learning models.

Cook et al. [10] designed a smart home that acts as an intelligent agent. The goal is to accurately predict inhabitant action prediction. The prediction algorithm used a back-propagation neural network that made a final prediction for the home based on a series of algorithm accuracies that the network generated based on the data within the home and through the algorithms. The prediction algorithm facilitated an adaptive and automated environment that meant to improve the experience of its inhabitants.

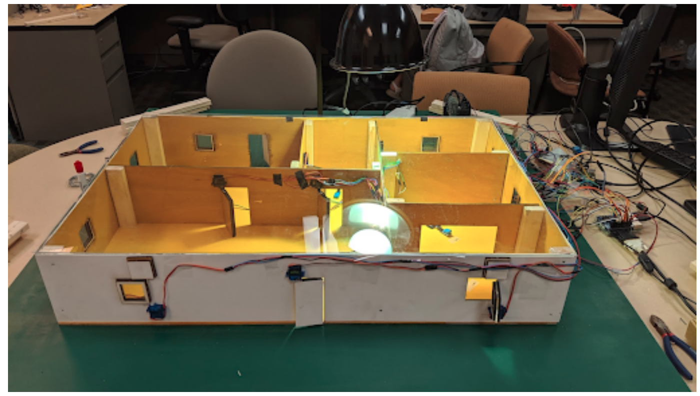

Fig. 1. Scaled home prototype.

III. The scaled smart home

A. Design

Our scaled home models the architecture of a typical small suburban home in the Southeast US. The home has two bedrooms, a bathroom, a living room and a kitchen with an attached dining room. The model was built using plywood for the walls and floor, wooden posts for supporting beams, and acrylic plastic for the roof.

Inside the house, we placed seven sensors: one in each room, and two in the living room and kitchen due to the size of the rooms. Each sensor gathers data on temperature and humidity. Additionally, we attached fifteen Adafruit's Raspberry Pi micro Servo motors, eight to the doors and seven to the windows as actuators for opening and closing them. The prototype of the scaled home can be seen in Fig. 1.

The temperature and humidity were measured using the DHT11 sensors, which have a low power requirement and small size, allowing them to be inserted into the rooms of the scaled home. These sensors provide an accuracy of ±5% in the humidity range of 20 to 80%, and ±2 °C in the temperature range of 0-50 °C. We balanced the positions of the sensors such that if there were multiple sensors in a room, they were placed on opposite sides of the room, otherwise each sensor was placed in the middle of its assigned room.

B. Data collection

Two different Raspberry Pi were employed to collect data from the scaled home. One of them was used for collecting temperature and humidity information from the seven embedded sensors as well as switching the heat lamp and the fan on and off while the other one controlled fifteen motors for opening and closing the scaled home doors and windows. All changes in motor and sensor status have been thoroughly recorded.

One problem was that the Pi with all of the motors did not have enough power supply to handle the standing current and simultaneous running of motors. To solve this, we used an Adafruit Pi HAT, a module that adds an external power source to the Pi. This allowed us to keep all fifteen motors hooked up to one Pi.

One of our objectives was to learn a model that can predict the temperature inside the home based on known variables such as the state of the lamp and fan and the open/closed state of the doors and windows. To provide training data for the learning process, we run a number of randomized scenarios to study a wide range of combinations of doors and windows being opened or closed under various environmental conditions. The sunlight and the wind were modeled by using states of the heat lamp and the fan. Over the course of experiments stretching over eighteen hours, we turned the lamp on and off every fifteen minutes, turned the fan on and off every five minutes, opened random doors and windows to let the air circulate through the home every thirty seconds. To model the movement of the sun, three different lamp positions were used throughout the experiment. At each interval, we recorded the temperature and humidity from all the sensors. Thus, approximately 900 rows of data were collected, each containing a timestamp, state of the lamp, state of the fan, the humidity and temperature information for each sensor and the state of each door and window.

IV. Experiments

For the experiments, we are following two main goals:

- Data analysis in order to study the temperature and humidity changes based on the state of the doors and windows, the heating lamp and the fan.
- Temperature and humidity prediction for each room to analyze the generalizability of the model.

The code for data collection and experimentation is currently hosted at `https://github.com/tjburns/not-a-SmartHome`.

A. Scaled home data analysis

One of the challenges of collecting data from smart home test beds is having a robust and continuous data collection from the sensors [11]. In this section, we aim to study how temperature and humidity change against the time based on the state of the heating lamp and the fan. This can help us to evaluate the robustness of our data collection procedure from the scaled home.

In developing machine learning models for a specific functionality, especially when you are using a simulation for data collection, it is very important to have a dataset that can represent the real world features. Considering that, we briefly study the collected data from each room before using the prediction model and investigate the patterns in data over time.

We simulate the days and nights by turning on and off the heating lamp and changing the direction and location of the lamp and the fan. We expect that these variations change the temperature and humidity respectively in the scaled home.

TABLE I

Selected values for hyperparameters of the regularized LSTM.

Hyperparameters	**Values**
time steps	10
number of features	32
learning rate	0.001
training steps	700
size of hidden layer	16
λ (regularization factor)	0.5
batch size	16

B. Regularized LSTM-based temperature/humidity prediction

In our previous scaled smart home, Ling et al. [12], we collected humidity and temperature data for the entire home but not each room separately. In current scaled home prototype, we have sensors in each room as well as in the kitchen; therefore, we collected humidity and temperature data from all the rooms and the kitchen. The inputs to our model (i.e. features) are state of the doors and windows, temperature and humidity in previous and current time steps. The output is the temperature and humidity in the target room for the next time step.

Recurrent neural networks (RNNs), such as long short-term memory networks (LSTMs), are used as a basic block for many applications in sequence learning and prediction. From our previous scaled home, Ling et al. [12], we concluded that the long short term memory (LSTM) model was the more accurate machine learning model over a fully connected neural network (FCNN). Considering that, we decided to focus on the creation of an LSTM model applied to the data we collected from the new scaled home prototype. However, as Rafiq et al. [13] propose, "regularization" can improve the performance of the LSTM models. Therefore, we impose commonly used L2 regularization on weights of the LSTM network to improve the generalization of the model [14].

We use 80% ($\simeq$ 720 data points) of the data for training and 20% ($\simeq$ 180 data points) of the data for testing. In our experiments, we consider $\lambda = 0.5$ for the regularization factor of the L2 regularizer. Table I presents the hyperparameters of the Regularized LSTM model we used for training phase. Our model is able to predict temperature and humidity of each room in the home for the next time step, based on the states of the doors and windows, the heat lamp and the fan of current and previous time steps. In the results section (section V), we compare the accuracy on train and test sets and analyze the generalizability of the prediction model.

V. Results

A. Evaluation of the data collection procedure

In Fig. 2-top, we present the recorded temperatures from the home against time, with dotted vertical lines showing the times that the lamp changed states, from on to off respectively over the course of the day. In Fig. 2-bottom, we show the recorded

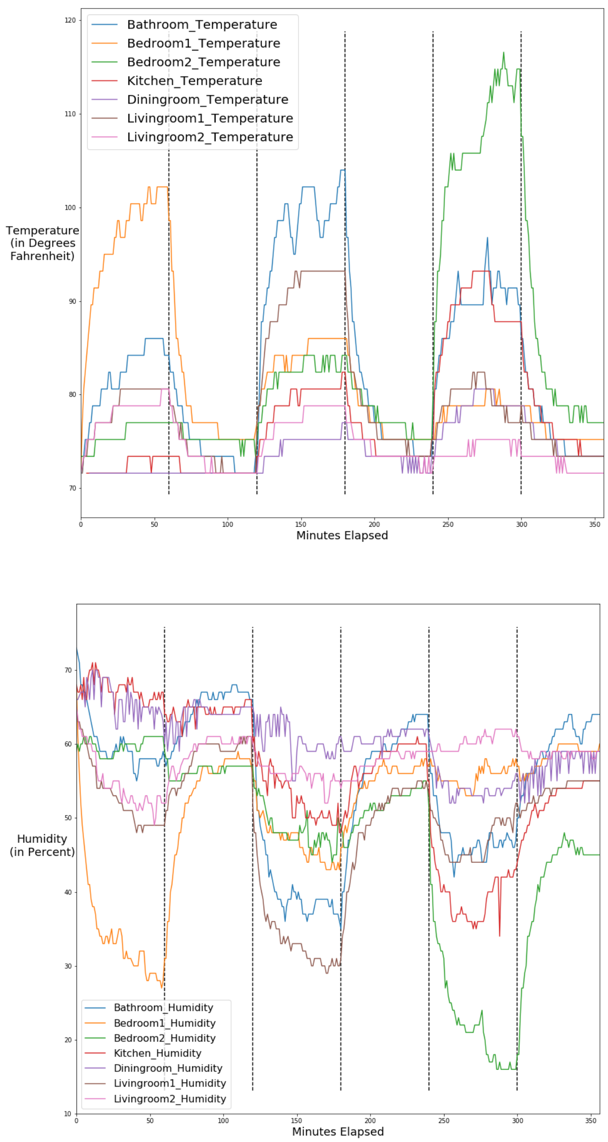

Fig. 2. Temperature (top) and humidity (bottom) recorded from each of the sensors within the home.

humidity from each room against time, again with the dotted lines showing the same information about the lamp state.

As expected, the temperature within the home rises quickly when the lamp is on and falls gradually when it is turned off. The inverse is true for humidity, it decreases when the lamp is turned on and rises when it is turned off.

B. Temperature/humidity prediction using regularized LSTM model

In Fig. 3, loss of the training step is shown for each room, with 80% of the data points for training set and 700 epochs. The reason that the loss on training data does not reach near 0 is that we used regularization to train the LSTM network in order to increase generalizability of the model.

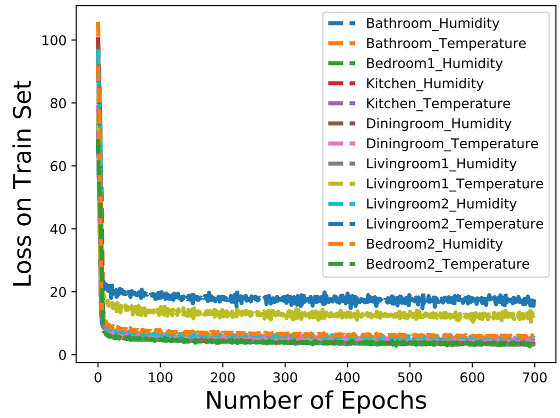

Fig. 3. Loss of Regularized LSTM model with train set size of 80% ($\simeq$ 720 data points) after 700 epochs.

TABLE II

ACCURACY OF REGULARIZED LSTM PREDICTION MODEL ON TRAIN AND TEST SETS FOR EACH ROOM AFTER 700 EPOCHS.

Prediction Target	Accuracy on Train	Accuracy on Test
Bathroom Humidity	0.90	0.51
Bathroom Temperature	0.91	0.72
Bedroom1 Humidity	0.94	0.83
Bedroom1 Temperature	0.97	0.93
Kitchen Humidity	0.87	0.31
Kitchen Temperature	0.91	0.79
Dining room Humidity	0.85	0.62
Dining room Temperature	0.98	0.99
Living room1 Humidity	0.74	0.54
Living room1 Temperature	0.83	0.63
Living room2 Humidity	0.79	0.69
Living room2 Temperature	0.85	0.87
Bedroom2 Humidity	0.94	0.47
Bedroom2 Temperature	0.97	0.70

We have also listed the accuracy of the prediction model on train and test sets for each room in Table II. It can be seen that the accuracy of the model for predicting the temperature for the next time step based on features in current and previous time steps is much better than the accuracy of predicting the humidity. The model can learn (or memorize) the humidity in the train data, however, the size of the train data or the selected features to predict the humidity are not sufficient enough to have better results on test set.

The accuracy of a model for each room with test set size of 20% and seven hundred epochs is shown in Fig. 4.

VI. CONCLUSION

Our goal was to construct a scaled home prototype to collect real life data from the home to be used in machine learning models. We analysed the collected dataset in order to validate the correctness of the data collection phase. We were able to create multiple LSTM models on the data and evaluate them.

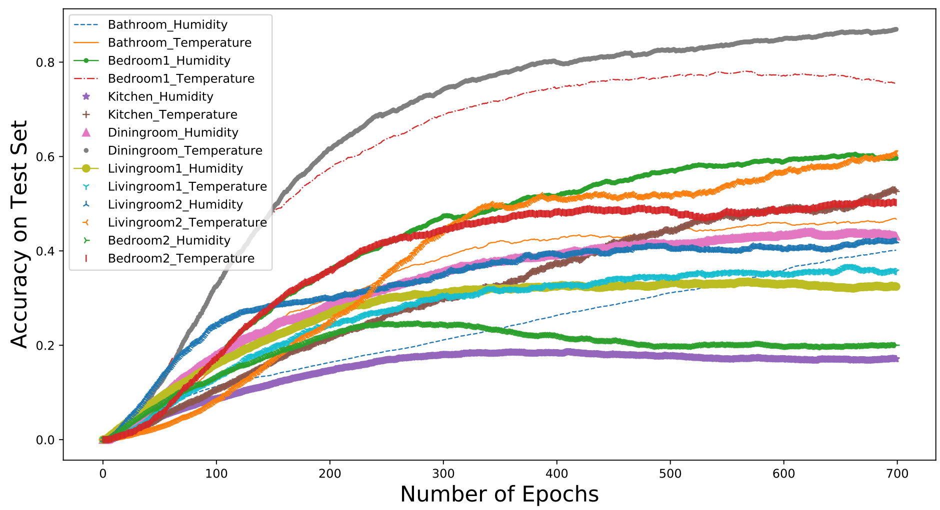

Fig. 4. Accuracy of model with test set size of 20% after 700 epochs.

We investigated different approaches such as alleviating the overfitting problem by using regularized LSTM model. Our model was successful in predicting the temperature and the humidity was also predicted to some degree.

The next step in our research is to collect more data from the scaled home through additional realistic scenarios for multiple days. This research is an starting point to design prototypes which can simulate the different environments and weather conditions of homes more accurately. This would give us a good benchmark to investigate models that can more accurately predict the temperature and humidity within the home.

Another interesting area of research in this context would be to learn a policy for the scaled home. In other words, to have a smart agent which is able to understand which actions would be appropriate to change the humidity and also temperature within the home based on the environment conditions to maintain a more comfortable environment.

Acknowledgement

The support for this work was provided by the National Science Foundation REU program under Award No. 1852002. Any opinions, findings, and conclusions and recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] EIA U.S. Energy Information Administration, "How much energy is consumed in U.S. residential and commercial buildings?" 2019. [Online]. Available: https://www.eia.gov/tools/faqs/faq.php?id=86&t=1

[2] D. Ketchum, "How much will it cost me to run an air conditioner?" https://budgeting.thenest.com/much-cost-run-air-conditioner-23306.html, November 2018, accessed: 2019-06-26.

[3] B. Lin, Y. Huangfu, N. Lima, B. Jobson, M. Kirk, P. OKeeffe, S. Pressley, V. Walden, B. Lamb, and D. Cook, "Analyzing the relationship between human behavior and indoor air quality," *Journal of Sensor and Actuator Networks*, vol. 6, no. 3, p. 13, August 2017.

[4] W. Lee, S. Cho, P. Chu, H. Vu, S. Helal, W. Song, Y.-S. Jeong, and K. Cho, "Automatic agent generation for IoT-based smart house simulator," *Neurocomputing*, vol. 209, pp. 14–24, October 2016.

[5] W. Jin, I. Ullah, S. Ahmad, and D. Kim, "Occupant comfort management based on energy optimization using an environment prediction model in smart homes," *Sustainability*, vol. 11, no. 4, p. 997, February 2019.

[6] F. Mateo, J. J. Carrasco, A. Sellami, M. Millán-Giraldo, M. Domínguez, and E. Soria-Olivas, "Machine learning methods to forecast temperature in buildings," *Expert Systems with Applications*, vol. 40, no. 4, pp. 1061–1068, March 2013.

[7] D. Chen and D. Irwin, "Weatherman: Exposing weather-based privacy threats in big energy data," in *Proc. of the IEEE International Conference on Big Data (Big Data)*, December 2017, pp. 1079–1086.

[8] T. Teich, F. Roessler, D. Kretz, and S. Franke, "Design of a prototype neural network for smart homes and energy efficiency," *Procedia Engineering*, vol. 69, pp. 603–608, 2014.

[9] E. Kim, S. Helal, and D. Cook, "Human activity recognition and pattern discovery," *IEEE Pervasive Computing*, vol. 9, no. 1, pp. 48–53, 2010.

[10] D. J. Cook, M. Youngblood, E. O. Heierman, K. Gopalratnam, S. Rao, A. Litvin, and F. Khawaja, "Mavhome: An agent-based smart home," in *Proc. of the IEEE International Conference on Pervasive Computing and Communications (PerCom)*, March 2003, pp. 521–524.

[11] N. Alshammari, T. Alshammari, M. Sedky, J. Champion, and C. Bauer, "OpenSHS: open smart home simulator," *Sensors*, vol. 17, no. 5, p. 1003, May 2017.

[12] J. Ling, S. Zehtabian, S. Bacanli, L. Bölöni, and D. Turgut, "Predicting the temperature dynamics of scaled model and real-world iot-enabled smart homes," in *Proc. of the IEEE GLOBECOM*, December 2019.

[13] H. Rafiq, H. Zhang, H. Li, and M. K. Ochani, "Regularized LSTM based deep learning model: first step towards real-time non-intrusive load monitoring," in *Proc. of the IEEE International Conference on Smart Energy Grid Engineering (SEGE)*, August 2018, pp. 234–239.

[14] S. Merity, N. S. Keskar, and R. Socher, "Regularizing and optimizing LSTM language models," *arXiv preprint arXiv:1708.02182*, 2017.