Automated Storytelling using Occupancy and Appliance Data

Abhishek Mangla
University of California, Berkeley
amangla@berkeley.edu

Ming Jin
University of California, Berkeley
jinming@berkeley.edu

ABSTRACT

As Internet of Things (IoT) and occupancy detection technologies become more common in households, more information about living patterns become accessible. Previous research explored data visualization techniques that convert numbers into informative graphs so that a user can determine a smart-home policy. We want to adapt this end-user development approach to one in which the home learns by itself how to operate using appliance and occupancy-detection data. In this paper, we use data sets covering a 3 month period from a Dutch household to create inference models which generate a data-driven storytelling of household activity. Automated storytelling could help people make energy-efficient choices, alarm users of real world changes and dangers, and elucidate interesting patterns about one household amongst others in the neighborhood.

CCS CONCEPTS

• **Computing methodologies → Information extraction**; **Natural language generation**; **Probabilistic reasoning**; Lexical semantics;

KEYWORDS

IoT, smart homes, smart buildings

ACM Reference format:
Abhishek Mangla and Ming Jin. 2017. Automated Storytelling using Occupancy and Appliance Data. In *Proceedings of ACM Conference, Washington, DC, USA, July 2017 (Conference'17),* 5 pages.
https://doi.org/10.475/123_4

1 INTRODUCTION

Smart buildings are becoming more common and thus providing tremendous amounts of data such as appliance usage and occupancy detection [1]. Understanding big data has many benefits for advancing technology. For example, medical assistance systems track anomalies within the living space to ensure that the elderly people within the home are able to call for medical help in time [2]. Furthermore, people often want to know what is going on in their homes when absent and thus real-time data visualization of energy usage provides clients that peace of mind [3]. Perhaps one of the most driving incentives for using smart home data is energy efficiency. Understanding where most consumption occurs can save businesses and families money and simultaneously encourages a greener lifestyle. For example, researchers controlled and smartly managed Heating Ventilation and Air Conditioning (HVAC) systems such that they achieved 38% reduction in energy consumption while maintaining indoor thermal comfort in 3 large warehouses [4]. Moreover, some retail stores used clustering techniques on raw electricity data to compare building loads across several stores [5]. This type of information provides critical business intelligence that simultaneously encourages smart grid solutions.

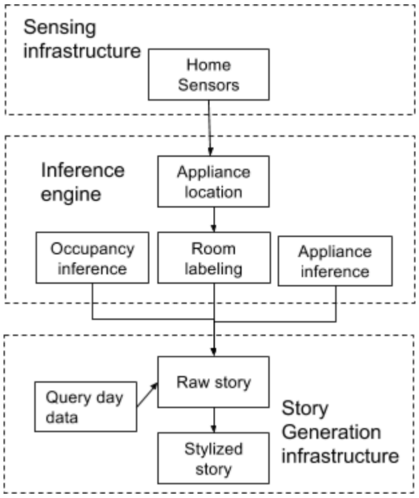

Figure 1: Storytelling work flow.

While significant effort has gone into cleanly displaying IoT data, little work has been made to aggregate multiple data sources to understand people's lifestyles. Our method attempts to learn trends related to room occupancy and activity at specific hours of the day through probabilistic inference. Furthermore, appliance data is the foundation for characterizing rooms with English labels (i.e. kitchen, bathroom). These inference models and room mappings facilitates an automated storytelling which is further stylize to provide greater entertainment value to consumers through smart agents like Amazon Alexa.

2 METHODOLOGY

The following sections provide more details on the steps of the story generation work flow (Fig. 1). We used data from the Dutch Residential Energy Dataset [6]. The data was collected from a single household in the Netherlands over a period of 6 months. The DRED dataset includes electricity monitoring, room-level indoor temperature, outdoor temperature, environmental parameters, room-level location information of occupants, Wi-Fi information, and house layout.

Our method requires only two datasets. The first is occupancy which contains an array of rooms that were occupied, sampled

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 123-4567-24-567/08/06…$15.00
https://doi.org/10.475/123_4

every 1 min. The second one has voltage of several appliances in the home, sampled at a frequency of 1 Hz.

2.1 Appliance Location

Appliance locations within the household are not always going to be provided to a smart home system and those locations can change based on consumer use and preference as well. Therefore the first step in our work-flow as shown in Fig. 1 is to map each appliance to its most likely room location. We want to estimate the probability of the appliance being in some location given occupancy states of all rooms at all times and appliance states for all times.

X_r^t = occupancy state of room r at time t
Y_a^t = appliance a state at time t (used or not used)
L_a = location of appliance a

$$Pr(L_a|X_1^t, X_2^t ... X_R^t, Y_a^t) = \frac{Pr(X_1^t, X_2^t \ldots X_R^t, Y_a^t|L_a)P(L_a)}{Pr(X_1^t, X_2^t \ldots X_R^t, Y_a^t)}$$

$$\propto [\prod_{r=1}^{R} Pr(X_r^t|Y_a^t, L_a)Pr(Y_a^t|L_a)]Pr(L_a) \quad (1)$$

$$\propto [\prod_{r=1}^{R}\prod_{t=1}^{T} Pr(X_r^t|Y_a^t, L_a)Pr(Y_a^t)]Pr(L_a) \quad (2)$$

In Eq. 1, we assume room occupancy is independent from each other. In Eq. 2, influence of time on occupancy and appliances is assumed independent. We also assume appliance state is not dependent on its location. To solve for Eq. 2, we use a prior distribution for $Pr(L_a)$ and $Pr(X_r^t|Y_a^t, L_a)$. Parts of both prior distributions are represented in Tables 1 and 2. Our method estimates $Pr(Y_a^t)$ using inference by enumeration on appliance data.

Table 1: Sample Prior Distribution for TV Location

Room	$Pr(L_a)$
livingRoom	0.65
kitchen	0.08
bedroom	0.25
bathroom	0.01
Storeroom	0.01

Table 2: Sample Prior Distribution: r = Living Room, a = TV, t = 3 PM

X_r^t	Y_a^t	L_a	$Pr(X_r^t|Y_a^t, L_a)$
not occupied	not used	living room	0.8
not occupied	not used	bathroom	0.1
not occupied	not used	kitchen	0.1
occupied	used	living room	0.95
occupied	used	kitchen	0.01

2.2 Room Labeling

At this stage of the work flow, occupancy data indicates if someone is in Room 1, Room 2, or some other generic rooms, but since each household has its own unique room layout, the system needs to learn the best English word that represents each room based on the appliances being used frequently in that room. Our method uses Natural Language Toolkit's (NLTK) WordNet tool to search a hyponym[1] tree of the word "room" by performing a Breadth-first tree traversal. Once all possible English room labels are collected, the Wikipedia API is used to obtain 50-100 word summaries about each room type and then another Natural Language Processing (NLP) library called TextBlob extracts noun phrases for each summary. Some noun phrases are not in Word Net so we disassemble the phrases to individual words. For each possible English room type, we now have a list of associated noun phrases.

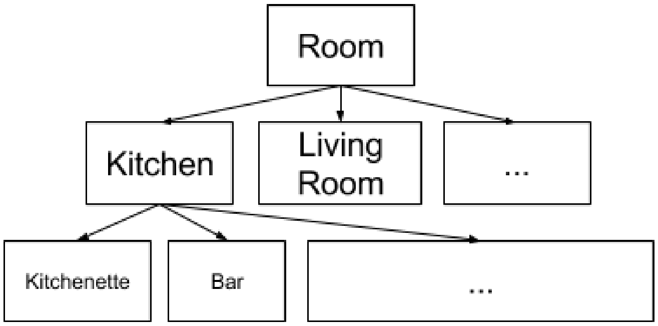

Figure 2: Partial WordNet Hyponym Tree for Room.

The final step is to use NLTK's two similarity indexing methods to figure out the best label for each room based on its appliance usage. The two similarity methods return a score between 0 and 1 but Leacock-Chodorow Similarity (LCH) is based on the shortest path that connects the senses while Wu-Palmer Similarity (WUP) is based on the depth of the two senses in the taxonomy and their most specific ancestor node [7]. For each generic room, we know the appliances that reside there and how frequently it is used compared to the other appliances. Alg. 1 illustrates how we use WUP and LCH on the extracted noun phrases to determine the best room label. Our method makes the assumption that the lesser an appliance is used compared to other appliances in the same room, the less it defines the room's usage. Thus the weight for each appliance is $Pr(L_a|X_1^t, X_2^t ... X_R^t, Y_a^t)$ and we can compute this probability (Eq. 2).

2.3 Occupancy Model Generation

To maintain an understanding of room occupancy, a 7 day, 24 hour model tracks how often occupancy was detected and where for each hour of the day, each day of the week. Our method retrieves $Pr(X_r^t, time = t|$ past occupancy data) using inference by enumeration.

I = set of time instances i in which L is occupied during t's hourly interval
$N(i)$ = total number of rooms occupied during i
$\alpha = \sum_{\forall i \in I} \frac{1}{N(i)}$

[1] A word of more specific meaning than a general or superordinate term applicable to it. For example, 'spoon' is a hyponym of 'cutlery'.

Algorithm 1 Room Labeling

```
procedure RoomLabeling(generic Room as GR)
    $GRSims = map()$    ▷ keys= room labels, values= similarity
    for each room label as RL do
        $appliancesInGR \leftarrow getApps(GR)$.
        for each APP in appliancesInGR do
            $total \leftarrow 0, counter \leftarrow 0$
            for noun phrase in nounPhrases(RL) as NP do
                $C = avg(wupSim(APP, NP), lchSim(APP, NP))$
                $total \mathrel{+}= f(GR, APP) * (C)$    ▷ $f$ is Eq. 2 where
    $L_a = GR$, $Y_a^t$ = APP being used, $X_r^t$ = occupied
                $counter \mathrel{+}= 1$
            $GRSims[RL] = \frac{total}{counter}$
    RETURN $argmax(GRSims)$
```

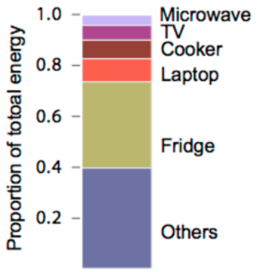

Figure 3: Most used appliances.

$\beta = \sum_{\forall i \in I} N(i)$ P = past occupancy data.

$$Pr(location = L, time = t|P) = \frac{\alpha}{\beta} \tag{3}$$

Retrieving the location with the highest probability produces the most likely location of a person at any given hour and day of the week.

2.4 Appliance Model Generation

The appliance model is similar to the occupancy model except it uses appliance data as a given knowledge.

λ = number of times A is used during t's hourly interval.
ι = total number of times any appliance used during t's hourly interval.
$P = Y_a^t, Y_b^t ... Y_z^t$ ∀ appliances

$$Pr(appliance = A, time = t|P) = \frac{\lambda}{\iota} \tag{4}$$

2.5 Raw Story

Our method attempts to identify trends in appliance usage and then decides which deviations from these trends are worth mentioning in a summary of the queried day (QD).

Our first step in generating a story is to create a list of noteworthy deviations from trends- this list is called Interesting Variations (IV). Our method loops through each hour of QD data and if the room most likely to be occupied is different from the QD observed location, information about this deviation is added to the IV (Table 1). Information includes a start time and end time of the variation, which room is expected to be occupied and probability, and which room is actually occupied and respective probability.

In Alg. 2, we look at each variation in IV, figure out how likely this occupancy variation occurs and mention it in the story. Each variation corresponds to a sentence in the story and we add to this sentence while executing within the initial for-loop. For each appliance in the actual location, a 5th degree polynomial regression model is used to estimate model and QD usage. Subtracting the two regressions and then looking at peak differences indicate either unusually high or unusually low use of that particular appliance. ϵ and α are arbitrarily set as experimental parameters.

Algorithm 2 Storytelling given Interesting Variations

```
procedure storyGeneration(day)
    $IV = InterestingVariations$
    $story = ""$
    for each variation in IV as v do
        if $v.expectedProb > \epsilon$ then
            add words about high probability to story
        else
            add words about low probability to story
        for each appliance in $v.actualLocation$ do
            diff = model data - QD data
            diffReg = calculate regression for diff
            if $diff[v.start \rightarrow v.end] > \alpha$ then
                add words about higher use of appliance
            else
                add words about lower use to appliance
```

2.6 Stylized Story Generation

We tested current neural nets models for Shakespeare stylization and the resulting story was not evidently stylized and instead even more confusing. Thus, our current method extracts defined patterns from a sample Trump text and then performs word-to-word or even word-to-phrase mapping. If the word is not seen in the sample text, our method simply does identity mapping.

3 RESULTS

Evaluating the story generation quantitatively and through automation is difficult, but we can test by creating line plots of appliance data against QD data and see where the big differences lie and how those differences are captured in the outputted story.

Our sample QD data for some Monday generates the Interesting Variations shown below in Table 1. It depicts the start time (S-time) and end time (E-time) of when the expected occupancy differed from the observed one. Expected occupancy (EO) is the place our occupancy model predicts a person should most likely be residing which requires the probability of EO (Eq. 2). Actual occupancy (AO) is the observed room and probability of this room being occupied is also calculated using Eq. 2.

Given the IV above, we generate the raw story below with Alg. 2:

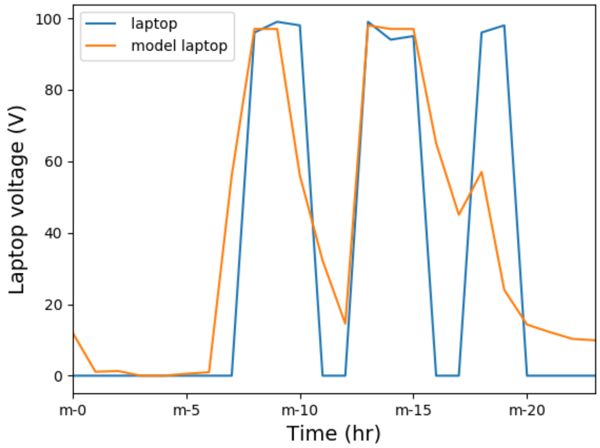

Figure 4: QD and model plot for laptop.

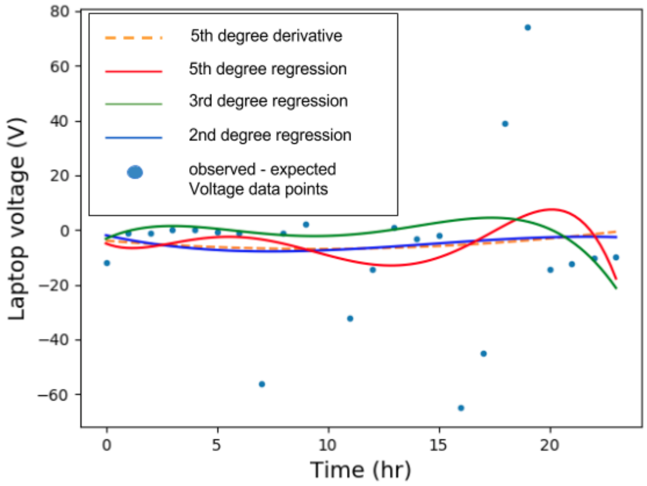

Figure 5: QD and model difference regressions.

On Monday, At 12:00 AM, a person is very likely to be in the bedroom but is instead in the kitchen and using laptop less than regular, mains, and sockets. At 08:00 AM, a person should be in the Kitchen but is in the bedroom. At 03:00 PM, a person is probably in the LivingRoom but is instead in the kitchen and using laptop less than regular, TV more than regular.

A small evaluation of this raw story output can be made if we look at the laptop appliance usage closely. Fig. 4 shows how the QD laptop (blue line) varies with model laptop (orange line). The story says the person is using the laptop less than normal and indeed we can confirm this by looking at Fig. 5.

4 CONCLUSION

In this paper we have demonstrated how IoT data can be processed and transformed into a story that has interesting information for people. While currently the story outputs abnormal usage of appliances when a person is at a different location than normal, we hope to add increased data correlation strategies so that perhaps the program will notice that normally a TV is on along with BlueTooth speakers, so if it ever encounters a QD where one is on without the other, this is output as an abnormality. Our method also discusses a novel way to characterize rooms based on appliance usage using NLP methods.

For future work, we hope to fix the problem of over-fitting to past model data and better accommodate learning new lifestyle patterns that may be adapted. To do this, we can use temporal techniques to gradually weigh out past experiences and favor new ones or even separately track recent trends in the past week and compare against overall tendencies of the past. While the current method executes the Room Labeling and Appliance Location every cycle, ideally, we want to only do them upon initialization in a new smart home environment. Other optimizations include parallelizing data processing from CSV files, learning what the best α and ϵ values are for an appropriate story length, and figuring out the best regression models to discover appliance energy differences.

We also hope to improve our stylization method significantly. A specific approach is to apply a sequence to sequence neural machine translation model [9]. The basic model consists of an encoder recurrent neural network (RNN) that learns a state vector of the input sentence, and an decoder RNN that extracts the state vector and produces the stylized output. Previous works have shown success of such model in performing Shakespearean style conversion [10]. However, to apply the model over smart home data, we first need relevant data and training set.

REFERENCES

[1] Jiakang Lu, Tamim Sookoor, Vijay Srinivasan, Ge Gao, Brian Holben, John Stankovic, Eric Field, and Kamin Whitehouse. 2010. The smart thermostat: using occupancy sensors to save energy in homes. *In Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems (SenSys '10)*. ACM, New York, NY, USA, 211-224.

[2] Himmel, Simon, and Martina Ziefle. 2016. Smart Home Medical Technologies: UsersâĂŹ Requirements for Conditional Acceptance. *i-com* 15.1: 39-50.

[3] Nico Castelli, Corinna Ogonowski, Timo Jakobi, Martin Stein, Gunnar Stevens, and Volker Wulf. 2017. What Happened in my Home?: An End-User Development Approach for Smart Home Data Visualization. *In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 853-866.

[4] Omid Ardakanian, Arka Bhattacharya, and David Culler. 2016. Non-Intrusive Techniques for Establishing Occupancy Related Energy Savings in Commercial Buildings. *In Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments (BuildSys '16)*. ACM, New York, NY, USA, 21-30.

[5] Almir Mehanovic, Emil Sebastian RÃÿmer, Jakob Hviid, and Mikkel Baun KjÃęrgaard. 2016. Clustering and Visualisation of Electricity Data to identify Demand Response Opportunities: Poster Abstract. *In Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments (BuildSys '16)*. ACM, New York, NY, USA, 233-234.

[6] S. N. A. U. Nambi, A. Reyes Lua and R. Prasad, âĂIJLocED: Location- aware energy Disaggregation Framework,âĂİ *in Proceedings of the 2nd ACM International Conference on Embedded Systems For Energy-Efficient Built Environments (BuildSys), 2015.*

[7] Pedersen, Ted, Siddharth Patwardhan, and Jason Michelizzi. "WordNet:: Similarity: measuring the relatedness of concepts." *Demonstration papers at HLT-NAACL 2004. Association for Computational Linguistics, 2004.*

[8] Bird, Steven, Edward Loper and Ewan Klein (2009), *Natural Language Processing with Python*. OâĂŹReilly Media Inc.

[9] Dzmitry Bahdanau, Kyunghyun Cho, Yoshua Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. *CoRR, 2014*.
[10] Mark Kwon, Jesik Min and Se Won Jang. Writing Style Conversion using Neural Machine Translation.