CS-308-2015 Final Report
Smart Lab

Deepali Adlakha	11D170020
Divyam Bansal	110050086
J Guna Prasaad	110050082
Vipul Venkataraman	110050084

Contents

1 Introduction

In recent years, automatic control of appliances have gained a lot of interest. The need for efficient and reliable control of appliances bring in the idea of automation with reduced human effort. Industries and manufacturing plants have been using both centralized and distributed versions of automation and serve as evidence for automation leading to improved productivity and energy conservation.

Building automation is a venture that has gained consensus recently. It is the control of a building's heating, ventilation, air conditioning, lighting, and other systems through a centralized building management system. *Home automation* is the residential extension of the idea. The major objectives of building automation are improved occupant comfort, and reduced energy consumption and operating costs. In this project we explore and implement a relatively simpler subset of the problem applied to a situation where multiple users use a pool of appliances based on personal preferences.

2 Problem Statement

Smart Lab A room automation system that predicts and controls the lights and fans in a room where a large number of such appliances are deployed and each user has his/her own preferences. The prediction employed is influenced by factors such as time of the day, user and history of appliance usage.

3 Requirements

3.1 Functional Requirements

The lab layout we consider consists of the following components:

1. **Identity Detector**
   We require an identity detector in order to detect the identity of the user for whom the system must predict the preferences. This system can be as simple a tag detector or as complicated as a biometric system.

2. **Entry/Exit Detector**
   We naturally time the trigger for switching on/off appliances when a user enters or exits the lab. We need a system to detect the entry and exit of users. Please note that such a system needs to be combined with the identity detector to predict the desired change in state of the appliances.

3. **Intelligent Predictor** We require a system to intelligently predict the preferences of a particular user based on past entries and also actively learn preferences of a new user or change in preferences online. It also needs to account for change in preferences due to time variations.

4. **Lab Layout** The lab layout we consider consists of fans and lights in a specific organisation as shown in Fig. 1. We further require to connect the power control of these appliances to our system so that we can toggle the working status of the appliances as and when needed. The lines L1, L2, L3 and L4 are operated through single switched each and each fan is operated with a switch of its own.

3.2 Non-functional Requirements

Following are the major non-functional requirements :

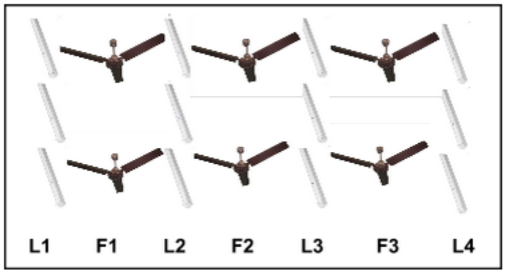

Figure 1: Lab Layout

Item	Qty.	Remarks
Raspberry Pi	2	One to act as an interface to the relay and another to detect entry, predict and communicate to the other raspberry pi about the change in status
Relay Circuit	1	To amplify Raspberry Pi output for the appliances
Fans	6	As per the layout
Tube lights	4	As per the layout

Table 1: Hardware Requirements

1. **Usability**
   The system needs to be easily configurable and usable by the target community, who are the users of the lab. It must essentially reduce the time and effort spent in switching on/off the devices the user wants.

2. **Sensitivity**
   The system needs to be sensitive to changes in preferences of the users. This often occurs during seasonal variations, relocation in the lab. The system must be sensitive enough to adapt its prediction to such changes.

3. **Extensibility**
   The system is initially built with a pool of users and it must be extensible in a way that adding or removing a user is not difficult. This requirement is important as such changes occur frequently in typical lab scenarios.

4. **Accuracy**
   The performance of the system must also be measured against the accuracy of the prediction system. It should match with the actual preferences of the users.

5. **Response time**
   This is important both for user experience and energy conservation. They delay between user entry or exit and the actual toggling of appliances should be minimized.

3.3 Hardware Requirements

Hardware requirements are mentioned in Table 1.

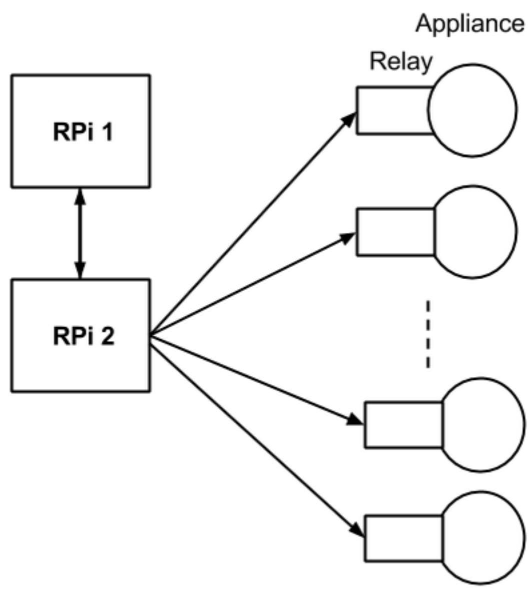

Figure 2: System Design

3.4 Software Requirements

Following are the software that were used in the implementation of the Smart Lab system:

- MySQL Server (`version : 5.5.37-0ubuntu0.12.04.1`)
- python (`version : 2.7.6`)

4 System Design

The system mainly consists of two raspberry pis and relay connected to the appliances to amplify the output of the raspberry pi. This is shown in Figure 2. When the user enters/exits the lab, he/she has to register it in RPi1. RPi1 records the preferences

5 Working

We have presented the workflow of the system and the interaction with the user in terms of the flow diagram in Figure 3. A user will typically follow the steps as mentioned in the diagram:

- Enter the lab. As soon as the user enters the lab he/she has to login to our portal. Please note that in our system this acts both as entry/exit detector and identity detector.
- The system will generate preferences for the corresponding user based on the time of the day and toggle the appliances as per the generated suggestions.
- If the user is fine with the current configuration he/she can continue working. Else, the user is given an option to override the system's suggestions. The new suggestion is also recorded by the system and will be taken into account in future predictions. This is essentially the *online* part of our learning model.

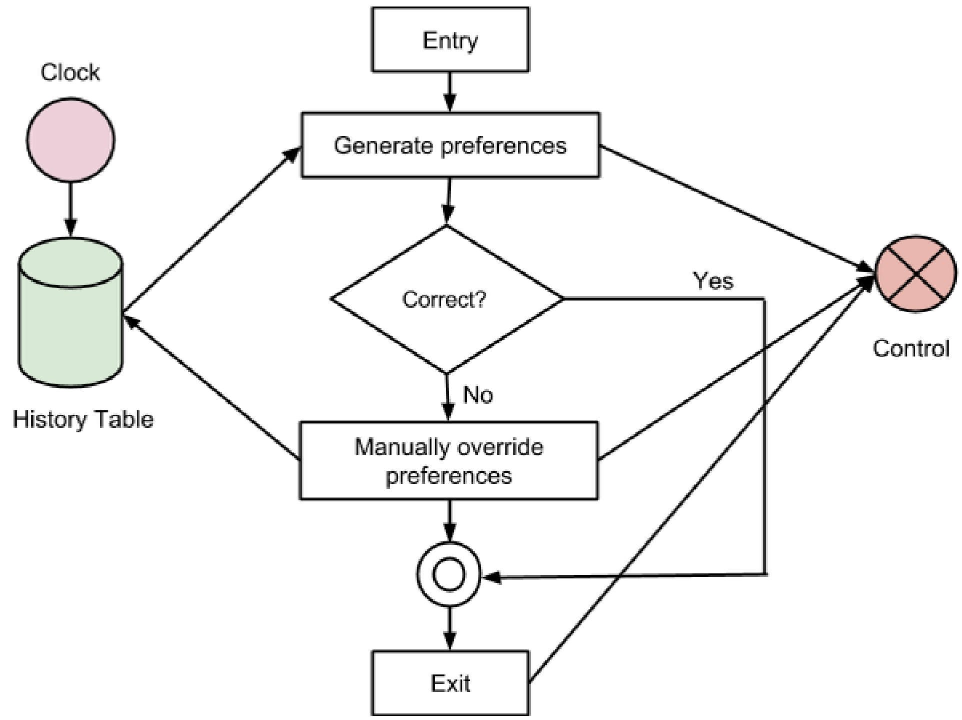

Figure 3: Workflow of the system

- Exits the lab. When the user exits the lab, he/she logs out of the smart lab and hence the system automatically switches off those appliances that were switched on specifically for this user (either based on generated preference or overridden rule)

6 Challenges and Discussion of the system

6.1 Collected Data

We have spent a considerable amount of time in pruning the data as the initial dataset that we got had a lot of problems. Following were some of the casesProblems with datacaused problems in the data

- Accidentally switching on some other appliance instead of the preferred appliance
- Switching on a different appliance due to orientation change in using the portal on mobile phones
- One user testing all the appliances leading to bias in the learning algorithm
- Manually switching on/off the fan and hence some records were missing in the data
- Some of the entry/exit data were missing
- Power consumption data was not usable because the two clocks were not synchronous

Data Pruning Data pruning was performed to remove certain entries that do not record preferences. However, nothing could be done about records that were not present. Following were some of the steps we performed in pruning the given dataset:

- If a single user operates all or majority of the appliances in a limited time window (5-10 mins), we remove all these entries as they might introduce bias. We assume that these entries denote that the user wanted to check the system rather than register actual preference.
- If a single appliance is operated too frequently in a small time window (5 mins), we remove them. In this case, we assume that it was for testing purposes.

6.2 Choice of Learning Model

Why not decision trees? We started out with the objective of using a learning algorithm such as decision trees to map appliance configuration to people configuration in the room. But in the data we got each record corresponds to a user switching on/off an appliance. So, to get the appropriate data we needed to maintain the room state and process each new row to obtain a training example. We can train a model on such training examples so that the model provides us the most preferred appliance configuration for a given people configuration. This procedure is very erroneous because absence of one record in the history table will make a lot of the subsequent training examples incorrect. Hence, we had to discard such learning algorithms

Frequency based prediction Given the constraints in the data, we wanted to reduce the impact of absence of some rows in the table (either due to faulty records or manual control of appliances). We hypothesized that switching on an appliance strongly denotes preference of the user towards using that particular appliance. This is recorded by the relative frequency with which the user has operated this appliance compared to other appliances. We further prune our choice based on a probability threshold. This is a tunable parameter in the system. The default is 0.25.

7 Future Work

- The system can be extended to include air conditioners and preferred temperature
- Given a more robust data collection mechanism, we can implement a better learning model to accurately model seasonal and time variations
- We can use temperature and light sensors to predict the need for lighting or fans and switch them on/off automatically. This however eliminates the personalization our current system provides

8 Minutes of Presentation

We presented a demo of the following tasks before the instructor and the teaching assistants:

- Entry and exit of a user already existing in the database
- Entry of a user whose one or more preferences are already on
- Exit of a user whose one or more preference were switched on by some other user
- Ability to override the prediction, if it is not preferred
- Adding a new user and learning his/her preferences
- Ability of system to handle temporary change in preferences and revert back to old suggestion

Further, we had discussions about comparing our learning algorithm with a simple heuristic : *last choice next*. We presented points such as sensitivity of the system to temporary changes but a common pattern in user choices was observed to be that a change is preserved for quite sometime ie.. user behaviour happens in temporal chunks. We can compare the performance of our algorithm with this heuristic using the historical data and measuring the performance using false positives.

Problems in the provided data such as missing records posed a great challenge in adapting the data for a machine learning algorithm such as decision trees. We could have used simulated data to evaluate our system. However, the close to reality factor gets undermined in such a setup, which both the instructor and we wanted to avoid.

We also planned to implement the system for air conditioners, which could have posed new challenges such as dealing with ordinality of the temperature as against binary decision of switching on/off lights or fans. But, this could not be implemented mainly because of logistic difficulties during the time-line of the project, thanks to the AC system renovation in the building.

We also presented the problems in integration of the system as we had different parts of our system located in different systems and the probability of failure increased with every new component we added. We also suggested a better way to handle the connection to RPi that is interfacing the appliances with the other RPi. A simple python bottle server that can handle HTTP requests could have worked better than socket connections, as they are found to be more reliable.

9 Conclusion

In summary, this project was a very good learning experience in designing and implementing embedded systems. We learnt how information at hand can completely influence the way system is designed. The discussions we had about the various possibilities of errors in the data collected, and on how to model the preference prediction taking into account the multitude of possibilities.