ARIIMA: A Real IoT Implementation of a Machine-learning Architecture for reducing energy consumption

Daniela Ventura[1], Diego Casado-Mansilla[2], Juan López-de-Armentia[2], Pablo Garaizar[2], Diego López-de-Ipiña[2], and Vincenzo Catania[1]

[1] University of Catania
{daniela.ventura,vincenzo.catania}@dieei.unict.it
[2] Deusto Institute of Technology - DeustoTech, University of Deusto
{dcasado,juan.lopezdearmentia,garaizar,dipina}@deusto.es

Abstract. As the inclusion of more devices and appliances within the IoT ecosystem increases, methodologies for lowering their energy consumption impact are appearing. On this field, we contribute with the implementation of a RESTful infrastructure that gives support to Internet-connected appliances to reduce their energy waste in an intelligent fashion. Our work is focused on coffee machines located in common spaces where people usually do not care on saving energy, e.g. the workplace. The proposed approach lets these kind of appliances report their usage patterns and to process their data in the Cloud through ARIMA predictive models. The aim such prediction is that the appliances get back their next-week usage forecast in order to operate autonomously as efficient as possible. The underlying distributed architecture design and implementation rationale is discussed in this paper, together with the strategy followed to get an accurate prediction matching with the real data retrieved by four coffee machines.

Keywords: IoT, RESTful Infrastructure, Machine Learning, ARIMA Models, Eco-aware Everyday Things, Energy Efficiency, Coffee-Maker

1 Introduction

The potential of the IoT to drive a sustainable everyday life is more than probable. This fact is easily evidenced through its current application domains such as agriculture, energy saving at home or in industrial settings and the pollution and traffic control within the cities. One example of such potential is the Google's Nest Thermostat, perhaps the most famous IoT gadget during 2014. Their designers disclosed that it can become carbon neutral in a period of just eight weeks after its first usage. Carbon neutrality refers to the greenhouse gases that were created by manufacturing and distributing the device are offset by the energy savings one obtains from using it[3].

[3] https://nest.com/blog/2014/04/22/todays-earth-day-tomorrow-should-be-too/

However, it is still controversial how other myriads of IoT devices (everyday consumer appliances, fitness trackers or kitchen appliances) can be also labeled as green devices along their life-cycle: from manufacturing to disposal[4]. These new devices are designed to replace old-fashioned ones. Therefore, their inclusion will rise to an augment of electronic waste that probably will end in the landfill.

This paper describes the implementation of an approach that addresses this latter IoT challenge. Our proposal lies in two pillars: First, it is focused on embedding intelligence through open hardware electronics within everyday appliances of shared use (e.g. beamers, coffee-makers, printers, screens, portable fans, kettles, etc.). Our aim is transforming these electronic devices into Internet-connected eco-aware everyday things rather than replacing them by new ones. As a proof of concept, in the presented work we have focused on electronic coffee machines located in four different work-laboratories. The second pillar, it is to design and implement a RESTful infrastructure that enables to these eco-aware appliances to reduce their energy waste. It is devised to intelligently process in the back-end the most efficient operation mode at any time for each shared device and to give back such information to them, i.e. the appliances are able to operate autonomously in an eco-friendly manner.
We have named this architecture ARIIMA (the capital letters of the six former words in the paper title) as an analogy with the predictive model used to forecast the appliance's usage, ARIMA model. The presented paper makes a reality the theoretical design reported in a previous authors article [7] by implementing it.

2 Background

In the literature we have found several IoT architectures (e.g. LinkSmart Project [1], RestThing [2], S3OiA [3], Gao et al. [4], Wang [5] and Weiss et al. [6]). Their main features are: *1)* ability to integrate heterogeneous devices (even constrained [1]); *2)* REST-based architectures [2–4]; *3)* easy to merge with Cloud-services [5]; *4)* simple services and applications composition [1–3] and *5)* standardized formats to store the data [4]. Among the works reviewed, the architecture proposed by Wang [5] is the only one similar to ARIIMA in its application domain, i.e. energy efficiency. In that article the author focused on making a campus more energy efficient. He used Zigbee and RFID readers to control the laboratories' temperature and their occupation. The aim was to switch on or off the air conditioner efficiently by means of real time temperature measures. The architecture that we propose borrows some of the ideas of the works surveyed to create a RESTful platform with JSON as the established data format to exchange information. Our proposal is not restricted to a local domain like [6], i.e. in ARIIMA the Internet-connected devices can update their data from everywhere, therefore intermediate gateways are not needed. One of the main differences of our approach compared with the others, is the addition of an energy forecasting web-service based on the energy consumption reported by the same devices in the past. Another important outcome of our approach lies in its own implementation.

[4] http://www.wired.com/2014/06/green-iot/

Several architecture proposals that has been reviewed lack of a real development only presenting a conceptual design. Comparing ARIIMA with Wang's architecture, we can emphasize our completely open hardware/software solution[5] and its wide range of application. Thus, the presented architecture can be extrapolated to any context of appliances' automation towards energy efficiency.

3 Design Rationale

In a early author's work [7], we already studied the electric coffee machines' functionality. We distinguished the two typical operation modes of these appliances and we disclosed their associated power consumption: *1)* `On-Off mode`, consisting in repetition of actions "switch on", "waiting for the coffee machine to heat", "prepare the coffee" and immediately "switch off"; *2)* `Standby mode`, in which the appliance is permanently ready to be used and no long warming time is needed when one wants to prepare a coffee. Our major contribution was to theoretically demonstrate that depending of the number of people that use the appliance in each hour (note that it is not the same the usage of a couple in their private setting as the usage of many workers in a workplace), it is convenient to introduce a new appliance's operating mode in order to save energy. Thus, one that adjusts the coffee-machine's operation depending of the usage it is subjected to. In rush hours (3 coffees or more per hour) the coffee-maker should remain on (`Standby mode`), while periods of lower use it has to be switched off (`On-off mode`). In order to predict the appliance's utilization, time-series forecasting was applied, specifically ARIMA models. This methodology assumes that past patterns (number of coffee intakes per hour) will similarly occur in the future, and therefore are predictable. In the next sections, the implementation of the overall architecture that holds the theoretical idea presented in [7] is disclosed.

4 Implementation

Our ARIIMA proposal tries to overcome two problems: the lack of Internet connection in old-fashioned devices and their energy inefficiency. Thus, it transforms non-sustainable appliances into more eco-friendly Internet-connected smart ones. To enable Internet access to the coffee machines, we have attached to them a microcontroler which features Ethernet interface and that is compatible with Arduino MEGA (`iBoard Pro`). To overcome the energy efficiency issue, we have managed the forecasting of coffee machines' usage directly on a RESTful server. This architecture delegates the computing intensive coffee consumption forecasting process to a Cloud-based service, thus reducing the microcontrollers' processing as much as possible. The RESTful server provides REST APIs to receive energy data from appliances and to generate the weekly forecast associated to each sustainable coffee machine. According to RESTful principles, it exposes stateless services that can be observed in Table 1.

[5] https://github.com/dieguich/linked_data_coffee-maker/tree/UNO/ARIIMA

Table 1. The three different Cloud-based services offered by the ecoserver.

URI	HTTP Method	Description
/ecoserver/	POST	request to save new energy consumption events
/ecoserver/predictions/	GET	get the prediction for all coffee-makers
/ecoserver/predictions/[*deviceID*]	GET	get the prediction of the coffee-maker with parameter [*deviceID*]

For a better understanding of the interaction among the ARIIMA's components, we have divided the logic in two subsections: Data storing and forecasting.

4.1 Data storing of energy consumption events

It is necessary to keep track of the timestamps of every coffee intake to predict the working mode that the coffee machine should hold in each hour-slot along the working day. Each `iBoard Pro` features a RTC clock which is synced within a one second precision once a day by means of a pool of NTP servers depending of the country where the appliance is located. The retrieved time is stored within the RTC clock that has its own battery. If the NTP servers are not reacheable or the mains goes down, the Arduino board can always remains in synchronization by using its local time.
These data have to be stored to be used later for time series analysis. Taking into account that an Arduino board is a resource constrained device with reduced memory storage resources (Arduino MEGA have 4 Kb of EEPROM memory), it is not suitable to store large amounts of data. Therefore, we have designed the architecture shown in Figure 1 to manage the data storing of consumed coffees.

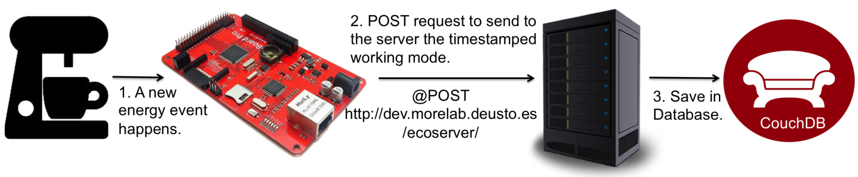

Fig. 1. ARIIMA architecture used for data storing of energy consumption events.

Whenever a new energy event is detected, the Arduino board captures some information like its timestamp, the energy value consumed in Wh, the state in which the machine is set, and so forth (an example of the complete JSON object in which these data are structured can be observed in Listing 1). Then, the microcontroler sends the JSON to the ecoserver via a POST request. Finally, the server carries out the storing of the JSON object as a document inside a CouchDB NoSQL database. CouchDB is itself an HTTP server acepting CRUD operations over JSON documents.

Listing 1 Sample of a JSON energy event sent by Lab1's coffee-maker


```
{
    "deviceID": "Lab1",
    "device_type": "COFFEE-MAKER",
    "datetime": "2014-03-05T10:23:41Z",
    "time_secs": 43215,
    "consumption_type": "MAKING_COFFEE",
    "consumption_time_in_secs": 34,
    "energy_consumption_kWh": 16.8
}
```


4.2 Forecast of coffee machine's next-week usage

The data collection process described in the previous subsection is used to predict the appliance's operating mode for the working days of the next-week (from Monday to Friday). The computational phase is done by a Cloud-based web service and it is performed weekly (on Sundays). The sorted data-flow related with the time-series prediction can be observed in the Figure 2.

The first task is to search for the number of coffees consumed in each of the hour-

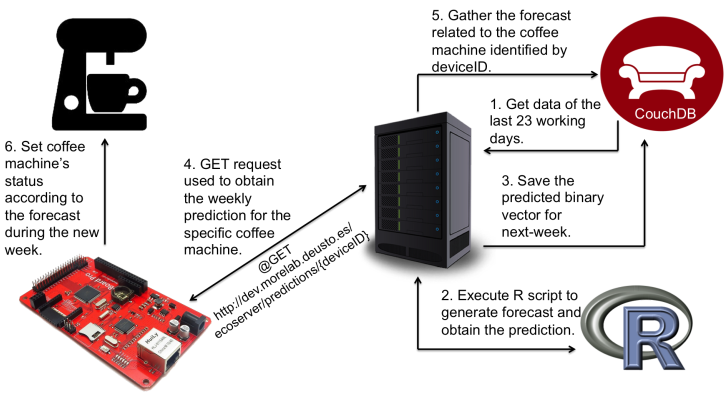

Fig. 2. ARIIMA architecture used to forecasting the coffee machine's next-week usage

slots along the previous 23 working days (about 1 month of data). These data are used to infer the operating mode that the appliance should perform in each hour-slot for each working day on the new week. The information returned by the CouchDB database has to be transformed by the server in a dataset processable by an R script. To perform this conversion, every working-day is divided into slots of 12 hours (from 7am to 7pm) and the total amount of coffees made in each hour is calculated. Using this vector as input parameter, the ARIMA forecasting is executed (second step in Figure 2). The outcome prediction gives the number of coffees that are expected to be consumed in 5 days ahead for each hour slot. The prediction is furthermore valuated in different confidence bounds (80% and 95%). We have tested the different forecasting intervals and we selected 80% confident value as the more accurate. Its selection is discussed in the next Section. Since the Arduino board needs to know the operation mode to assume for each hour-slot, the forecast is translated to a binary vector. For this aim, we logically evaluated whether each predicted number of coffee intakes exceeds the threshold of three coffees. In that case we set the correspondent time slot to 1 (work in `Standby mode`), or contrary set it to 0 (work in `On-Off mode`). This binary vector is saved as JSON object inside the database with the format showed in Listing 2.

Listing 2 Predicted 60 bits of binary data. Each bunch of 12 bit corresponds to each working-day (7am-7pm) of the new week.

```
{
    "deviceID": "Lab1",
    "prediction": "001110000110011100001l..."
}
```

In the 4th step of the Figure 2, each smart coffee-maker gathers weekly its prediction through a HTTP GET request sending its *deviceID*. When the server receives the request, it queries the database filtering by the *deviceID* received getting a JSON object (see Listing 2). The server sends the prediction vector back to the microcontroler and the Arduino board saves it in its EEPROM memory. In this way, everyday of the new week, it reads the sequence of 12 bits corresponding to that working day. In each hour-slot it applies the forecast automatically by using a relay leveraged within each device's On-Off button.

5 Evaluation and Results

To select the prediction which is closer to the real data, we compared the 5 days-ahead forecasted values with the real data observed during the forecasted period. The "training set" used to compute the prediction refers to 23 days from 21st May 2014 to 20th June 2014. Therefore, the empirical data are the next five working days (from 23rd June 2014 to 27th June 2014).
The ARIMA forecast issued by the R script give us different confident intervals (80% and 95%) for the exact number of coffees that are predicted in each slot, a.k.a. point forecast. Hence, we measured the binary closeness between the real data, and four different predicted data: *1)* point forecasted values; *2)* the values corresponding to the upper bound with 80% rate of confident; *3)* those corresponding to the upper bound with 95% rate of confident; and *4)* the mean between the point forecasted values and the values corresponding to their upper bound with 95% rate of confident.

The prediction's vectors that we compared are those already transformed to binary data like the presented in Listing 2. Therefore, we reviewed a survey of binary distances and similarities [8] to select, among 76 methods proposed there, Hamming, Jaccard and Sokal-Michener (also called Simple Matching) as candidate distances. The Hamming distance gives us the exact number of binary mismatching, while Jaccard and Simple Matching give equal weight for matches and non-matches as expressed in[6]. To ease the selection of the most accurate measure, we distinguished two type of prediction's errors: *1)* heavy (false negative); and *2)* light (false positive). The former refers to the mismatching that occurs when the real coffee machine's operating mode was `Standby` but the value predicted was to set `On-Off mode`; The latter alludes to the error occurred when the real coffee machine's operating mode was `On-Off` but the value predicted was to set `Standby mode`. The energy wasted related to heavy errors is greater than light errors.
The closeness of the four different predictions intervals using each of the proposed distance-measures for four coffee machines are shown in Table 2.

Table 2. Distances calculated for 4 different coffee machines placed in four laboratories. Hamming distance gives an scalar measure (**h**, refers to heavy errors and **l** to light ones), while Jaccard and SM give their closeness value normalized between 0 and 1.

Coffee Maker	Distances	P.Forecast	80%	90%	Mean
Lab1	Hamming	10 (**h**:10, **l**:0)	21 (**h**:3, **l**:18)	37 (**h**:1, **l**:36)	16 (**h**:7, **l**:9)
	S.Matching	0,16	0,35	0,61	0,26
	Jaccard	1,0	0,75	0,80	0,84
Lab2	Hamming	6 (**h**:6, **l**:0)	5 (**h**:5, **l**:0)	27 (**h**:0, **l**:27)	6 (h:6, **l**:0)
	S.Matching	0,1	0,08	0,45	0,1
	Jaccard	1,0	0,83	0,88	1,0
Lab3	Hamming	6 (**h**:5, **l**:1)	11 (**h**:0, **l**:11)	34 (**h**:0, **l**:34)	8 (**h**:1, **l**:7)
	S.Matching	0,1	0,18	0,56	0,13
	Jaccard	0,6	0,55	0,79	0,5
Lab4	Hamming	7 (**h**:6, **l**:1)	7 **h**:5, **l**:2)	7 (**h**:5, **l**:2)	7 (**h**:5, **l**:2)
	S.Matching	0,11	0,11	0,11	0,11
	Jaccard	1,0	0,87	0,89	0,87

The analysis of the results shows that for Lab2 and Lab4's coffee machines the prediction closest to the real data is provided by the upper bound with 80% rate of confidence for all the evaluated distances. Lab1's coffee-maker shows that the 80% confidence bound is the most accurate when applying the Jaccard coefficient while using Simple Matching and Hamming distances is more accurate the point forecast. The point forecast presents a higher number of heavy errors than 80% (10 vs. 3). Thus, the amount of waste energy using the point forecast is greater. For Lab3, the average seems to be the most accurate prediction,

[6] http://tinyurl.com/murubf3

however, the upper bound with 80% rate of confidence presents closer results in every distance evaluated. In base of these considerations, we have decided to use always the upper bound with 80% of confidence level whenever we want to calculate the forecast of coffee machine's next-week usage for any coffee-maker. The Jaccard coefficient is selected as the more suitable for our scatter dataset since this distance does not take into account the number of zero-matching.

6 Conclusions

The approach presented in this article has two targets. First, it aims to reduce the ecological impact that would cause the replacement of every old-fashioned consumer appliance by new Internet-connected ones. Our proposal is to embed an electronic adaptor within everyday appliances so they can become sustainable-IoT devices avoiding the replacement, and therefore the grow of electronic waste. Second, we provide a Cloud-based infrastructure that enables the eco-aware appliances to approximate their energy consumption to their optimal efficiency. This paper has described the REST-based architecture implementation and it has opened the path to demonstrate how the eco-aware devices (in this paper, a set of capsule-based coffee machines) could reduce their ecological impact towards carbon neutrality by means of ARIMA-based predictive models computed in the Cloud. Our ongoing work is to measure the energy consumption related with the EcoAdaptor and the Cloud based infrastructure to find out whether it can be offset with the saving attributed to the approach presented (15% per week, i.e. around 140 Wh per week as was proved in [7]). The whole ARIIMA infrastructure is currently deployed in our own servers, we plan to install it within any Cloud-platform such as Amazon EC2 or Google Cloud Platform. With this future enhancement, our solution can scale, be sustainable and reliable as the IoT requires.

References

1. The LinkSmart Project: `http://www.hydramiddleware.eu/`. Aug. 2014.
2. W. Qin et al., "RestThing: A Restful Web service infrastructure for mash-up physical and Web resources". Proc. of EUC'11, pp. 197-204, 2011.
3. Vega-Barbas, M., Casado-Mansilla, D., et al. "Smart Spaces and Smart Objects Interoperability Architecture (S3OiA)". In Proc. of IMIS'12, pp. 725-730, 2012.
4. Gao, L., Zhang, , et al. "RESTful Web of Things API in sharing sensor data". Proc. of ICITST-2011, pp. 1-4, 2011.
5. Wang, H-I., "Constructing the Green Campus within the Internet of Things Architecture". In Journal of Distributed Sensor Networks. pp. 1-8, 2014.
6. Weiss, M., Guinard, D. "Increasing Energy Awareness Through Web-enabled Power Outlets", In MUM '10, pp. 20-30, 2010.
7. López-de-Armentia, J., Casado-Mansilla, D., and López-de-Ipiña, D. "Reducing energy waste through eco-aware every-day things", In journal of MIS, vol. 10, no. 1, 2014.
8. Seung-Seok,C., et al. "A Survey of Binary Similarity and Distance Measures". In Journal of Systemics, cybernetics and informatics, vol. 8, no. 1, 2010.