

Home Energy Market Simulator: The Appliance Usage Model Manager

Andrea Monacchi *
Institute of Networked Embedded Systems,
Alpen-Adria-Universität Klagenfurt, Austria

February 6, 2015[†]

Abstract

The appliance usage model manager is a tool that can be used to model device demand by means of machine learning and data mining techniques. We propose a general purpose interface to handle such kind of models, which is used to query various types of models which can be learned from energy consumption datasets. The tool provides a hub where models can be loaded and used in the HEMS simulator[MZE14a]. As such we provide examples to aid users towards their first demand modelling experiments.

1 A manager of appliance usage models

The appliance usage model is an external tool that can be used to extract and handle appliance usage models (see Fig.1). Every appliance usage model is required to imple-



Figure 1: The appliance usage model manager GUI

ment the device model interface, as in Fig. 2. The HEMS simulator, can request the server (running on the model manager) to retrieve the starting probability of a specific device for a certain time interval (See Fig.3). For the sake of simplicity, we use hourly time intervals. Therefore requests within the same interval are supposed to return the same starting probability. The other input is used to model multiple usage of the same device within the time interval. This is the case of user-driven devices, such as a vacuum cleaner and a coffee machine, which are usually used various times to satisfy the needs of multiple inhabitants or moved to different rooms. At the current time, the

*Supported by Lakeside Labs, Klagenfurt, Austria and funded by the European Regional Development Fund (ERDF) and the Carinthian Economic Promotion Fund (KWF) under grant 20214-23743-35469 and 35470 (project MONERGY <http://monergy-project.eu>)

[†]First drafted on September 9th 2014

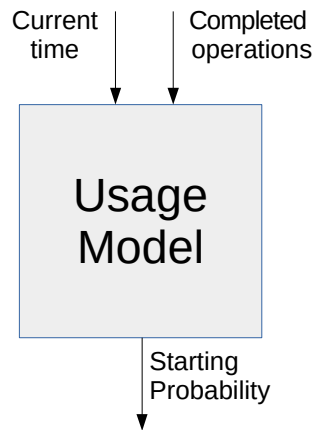


Figure 2: The device model interface

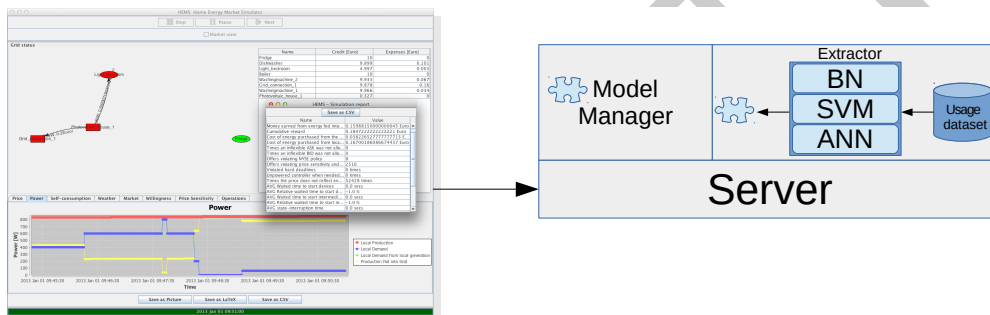


Figure 3: The usage model manager

manager can handle bayesian networks, artificial neural networks and support vector machines.

2 Handling multiple machine-learning techniques

The appliance usage model manager allows for a simple and quick extraction of usage models (See Fig.4).

2.1 Modeling a Bayesian Network

To handle Bayesian Networks we rely on the Netica Java library¹, whose manual is available at ². The structure used for the model is reported in Fig.5. We refer to [MEE⁺14] for further details on using bayesian networks to mine appliance usage from a consumption dataset. The Bayesian network is learned using the Expectation max-

¹<http://www.norsys.com/netica-j/>

²http://www.norsys.com/netica-j/docs/NeticaJ_Man.pdf

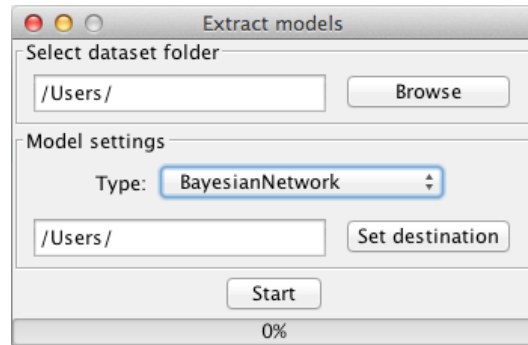


Figure 4: The model extraction interface

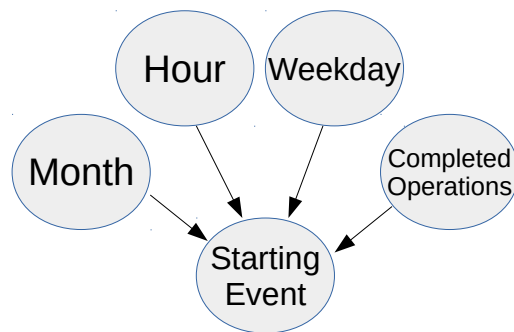


Figure 5: The Bayesian network

imisation algorithm, whereas the estimation is performed by means of the Junction tree algorithm.

2.2 Modeling an Artificial Neural Network

To handle Artificial Neural Networks we rely on the Neuroph Java library³. The library comes with an IDE onto which experiments with the models can be assessed, as well as a core library that powers our models. The ANN uses a multilayer perceptron with dynamic back propagation⁴. The default number of nodes in the hidden layer is 9 (i.e., $2 * n + 1$).

2.3 Using a Support Vector Machine

To handle Support Vector Machines, we rely on the libSVM java library⁵[CL11]. The model can be extracted both from CSV and SVMLight training datasets.

³<http://neuroph.sourceforge.net>

⁴<http://neuroph.sourceforge.net/javadoc/org/neuroph/nnet/learning/DynamicBackPropagation.html>

⁵<http://www.csie.ntu.edu.tw/~cjlin/libsvm/>

3 Extracting appliance usage events

The format for the CSV dataset files used to train appliance usage models is reported in Table 1. It is also worth to remark that without an appropriate feature scaling⁶ any

Table 1: Event data format

Month	Hour	Weekday	ConcludedOperations	Starting
...
March	H7	Weekday	0	Start
March	H7	Weekday	1	Hold
March	H7	Weekday	2	Hold
March	H7	Weekday	3	Hold
March	H7	Weekday	4	Hold
...

machine learning technique would suffer of the problem by which features with greater numeric ranges might dominate the others. To prevent this we scale each value with

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

which yields to a dataset in the range [0, 1].

A usage event dataset can be extracted from a consumption dataset, such as GREEND[MEE⁺14], using the Python script available on the UMMA sourceforge project page⁷.

Usage is: `processor_v2.py [dataset_path] [profiles_path] [offset_seconds]`

The appliance profiles and the offset are optional

Example: `processor_v2.py dataset/ profiles.json 30`

The processor receives by stdin the following arguments:

- **path to the dataset folder** where a bunch of csv files are available;
- **path to profiles** a json format file where device-specific thresholds are specified (e.g., `{"profiles": [50, 4, 4, 300, 1000, 10, 500, 50, 40]}`), the default threshold is otherwise 10 W;
- **offset** timeoffset to be added to the dataset timestamp (useful in case of discrepancy between the real and measured samples)

4 Experiments

The appliance usage model manager also provides benchmarking tool by which learned models can be visualised and assessed. The explorer (Fig. 6) plots the starting probability of devices over hourly time intervals and it is therefore meant to be used as a quick forecasting and visualisation means. The experimenter (Fig. 7) provides a set of predefined experiments under which models can be assessed. Each experiment implements the interface listed in Code 1.

⁶http://en.wikipedia.org/wiki/Feature_scaling

⁷<https://sourceforge.net/projects/umma/files>

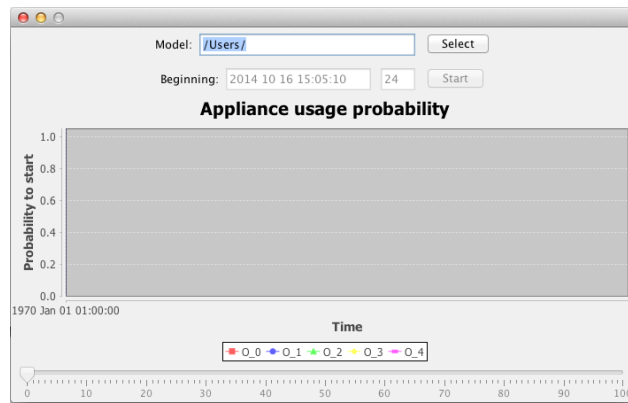


Figure 6: The usage model explorer

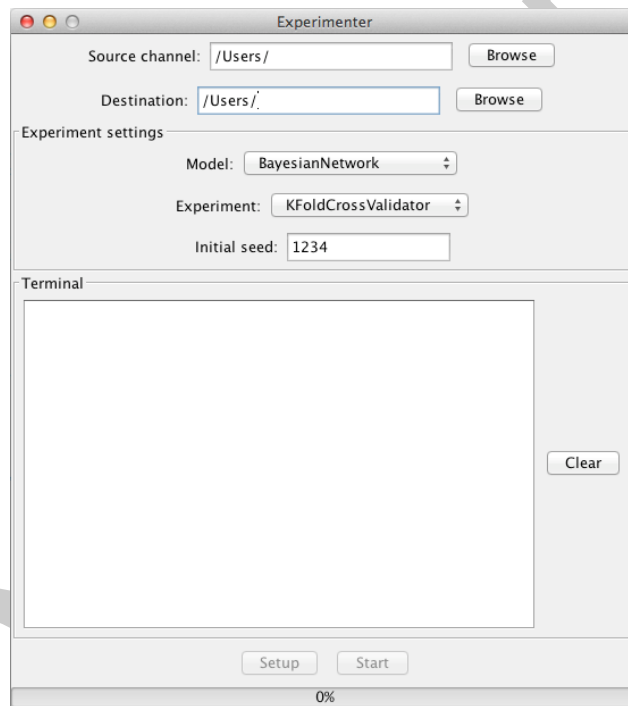


Figure 7: The usage model experimenter

Code 1: Experiment

```

1 public interface Experiment {
2     public void setProgressBar(JProgressBar bar);
3     public boolean setupExperiment(int initialSeed, String datasetPath, String destination);
4     public boolean performExperiment(String destination, String modelType);
5 }

```

A common experiment used to assess models and chosen parameters is k-fold cross validation⁸. UMMA provides a specific experiment, which extracts k randomised folds from a usage event dataset and trains k models, by using k-1 folds for training and 1 fold as test set per each model. The accuracy, mean squared error and root mean squared error per each experiment and on average are then computed and saved as an output report file.

5 The server

As also shown in [MZE14a, MEE⁺14], appliance usage models can be exploited for simulation purposes. The scenario description format described in [MZE14b] can be used to include appliance usage models within smart micro grid simulation frameworks. Model management is handled by the the appliance usage model manager, which also exposes the models through a network interface (Fig 8). In this way, each model can be

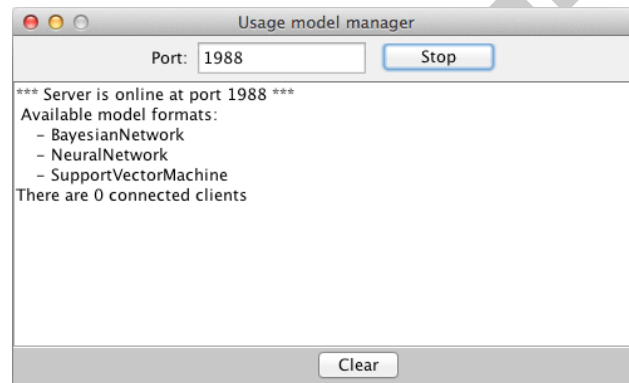


Figure 8: The usage model server

remotely loaded and handled, without worrying of importing dependent libraries and management into a specific simulation tool.

References

- [CL11] Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):27:1–27:27, May 2011.
- [MEE⁺14] Andrea Monacchi, Dominik Egarter, Wilfried Elmenreich, Salvatore D’Alessandro, and Andrea M. Tonello. GREEND: an energy consumption dataset of households in Italy and Austria. In *Proc. of IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Venice, Italy, Nov 2014.

⁸[http://en.wikipedia.org/wiki/Cross-validation_\(statistics\)](http://en.wikipedia.org/wiki/Cross-validation_(statistics))

- [MZE14a] Andrea Monacchi, Sergii Zhevzyk, and Wilfried Elmenreich. HEMS: A home energy market simulator. In *Computer Science - Research and Development*. Springer, Nov 2014.
- [MZE14b] Andrea Monacchi, Sergii Zhevzyk, and Wilfried Elmenreich. Home energy market simulator: The scenario definition format. Technical report, Alpen-Adria Universität Klagenfurt, September 2014.

DRAFT