A new approach for supervised power disaggregation by using a deep recurrent LSTM network

GlobalSIP 2015, 14th Dec.

Lukas Mauch and Bin Yang

Institute of Signal Processing and System Theory
University os Stuttgart

Content

Motivation

Limitations of current NILM approaches

Unsupervised event based

event detection	event matching	clustering	reconstruction

- difficult for multi-state loads
- not suitable for variable loads
- not scalable to a large number of loads and events

- no load specific disaggregation
- hand crafted feature extraction
- sampling frequency higher than the line freqency needed

missing robustness

Supervised eventless

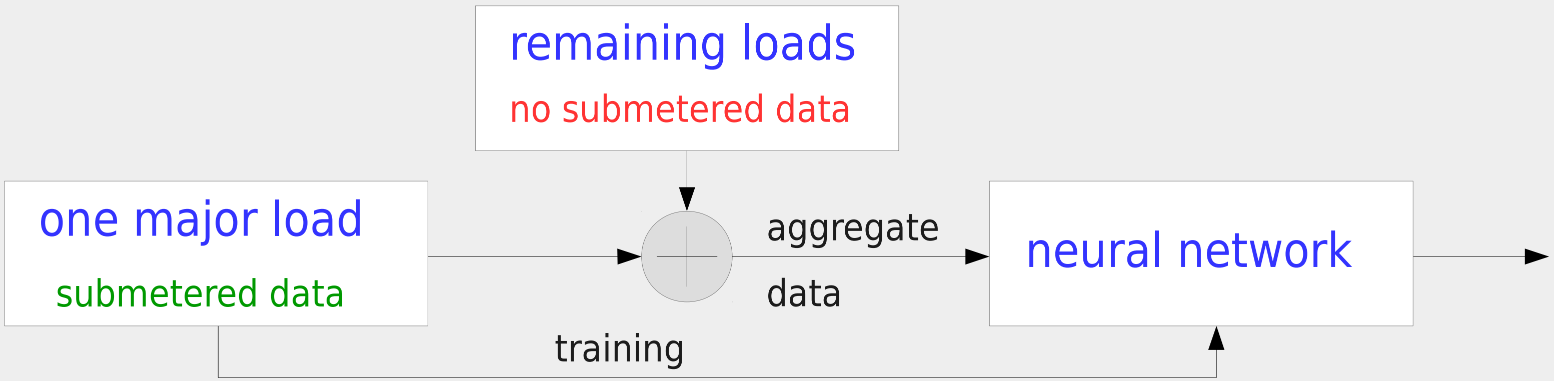
Factorial Hidden Markov Model (FHMM) for single channel source separation

- not scalable due to exponential complexity
- exact training and inference intractable
- HMM of each load has to be known

missing scalability

Our approach

Supervised Neural Network based approach for single channel source extraction



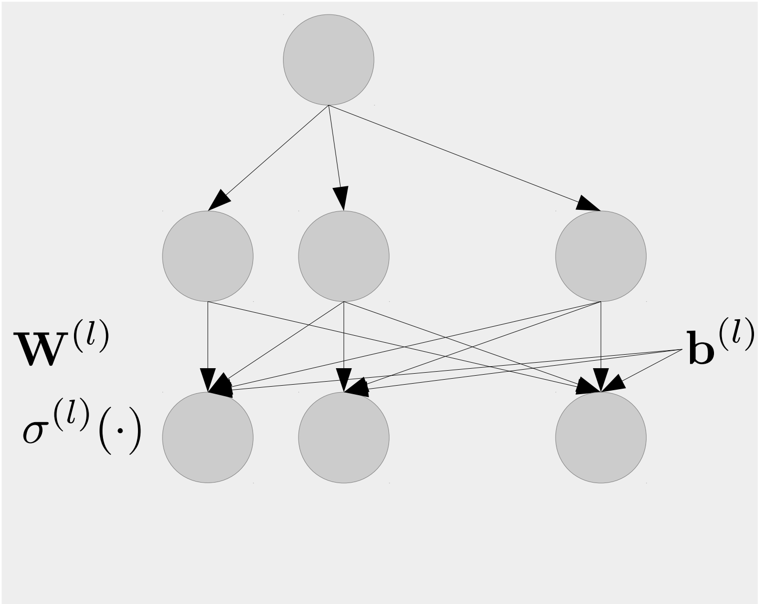
- remaining loads treated as time varying noise

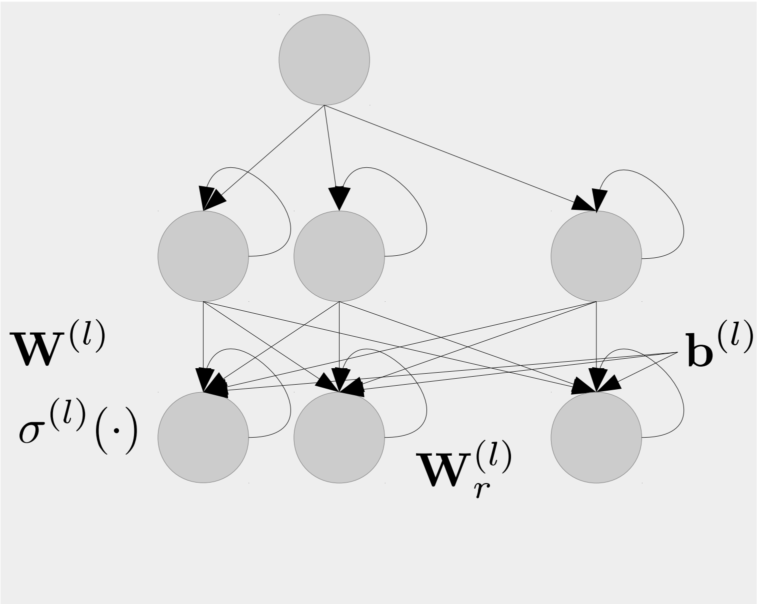
- scalable to many loads
- no hand crafted feature extraction
- assignment of power traces to specific loads possible
- suitable for multi-state and variable load devices
- suitable for low frequency (<1Hz) real power data only

- submetered training data needed

Recurrent Neural Network (RNN)

Feedforward Neural Network	Recurrent Neural Network





Feedforward Neural Network

- multiple layers of units
- feedforward connections

- universal static mapper
- used for classification and regression

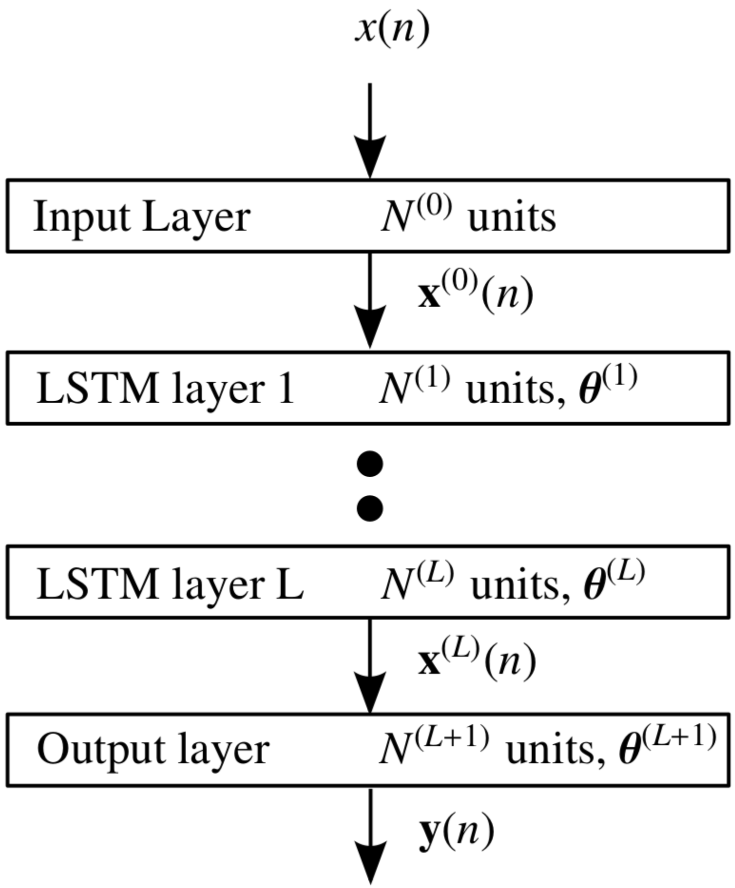
Recurrent Neural Network

- feedback connections allowed in each layer

- can learn any causal time-varying mapping
- used for sequence labeling and prediction

Model layout

Layout



- Use forward-backward processing to allow noncausal mapping

Mapping

$$\mathbf{x}^{(0)}(n) = [x(n), x(n-1), \ldots, x(n - N^{(0)} + 1)]^T \in \mathbb{R}^{N^{(0)}}$$

Gates $\mathbf{i}^{(l)}(n) = g(\mathbf{x}^{(l-1)}(n), \mathbf{x}^{(l)}(n-1), \mathbf{s}^{(l)}(n-1))$

$$\mathbf{o}^{(l)}(n) = \ldots$$

$$\mathbf{f}^{(l)}(n) = \ldots$$

$$g(\mathbf{x}, \mathbf{y}, \ldots, \mathbf{z}) = \mathbf{W}_x\mathbf{x} + \mathbf{W}_y\mathbf{y} + \ldots + \mathbf{W}_z\mathbf{z} + \mathbf{b}$$

States

$$\mathbf{s}^{(l)}(n) = \mathbf{i}^{(l)}(n) \circ \tanh\left(g(\mathbf{x}^{(l-1)}(n), \mathbf{x}^{(l)}(n-1))\right) + \mathbf{f}^{(l)}(n) \circ \mathbf{s}^{(l)}(n-1)$$

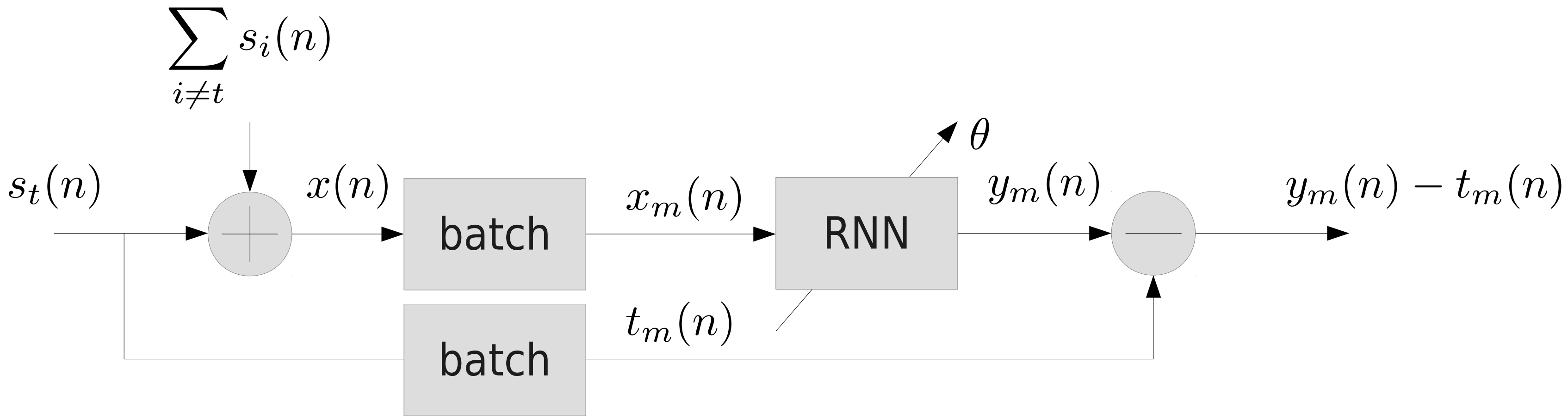
Output

$$\mathbf{x}^{(l)}(n) = \mathbf{o}^{(l)}(n) \circ \tanh(\mathbf{s}^{(l)}(n))$$

$$\mathbf{y}(n) = \sigma^{(L+1)}(\mathbf{W}^{(L+1)}\mathbf{x}^{(L)}(n) + \mathbf{b}^{(L+1)}) \in \mathbb{R}^{N^{(L+1)}}$$

Application to NILM

Extraction of target signal $s_t(n)$ with bidirectional RNN



Training pairs

$x_m(1), \dots x_m(B)$

$t_m(1), \dots, t_m(B)$

...signals divided into M blocks of length B

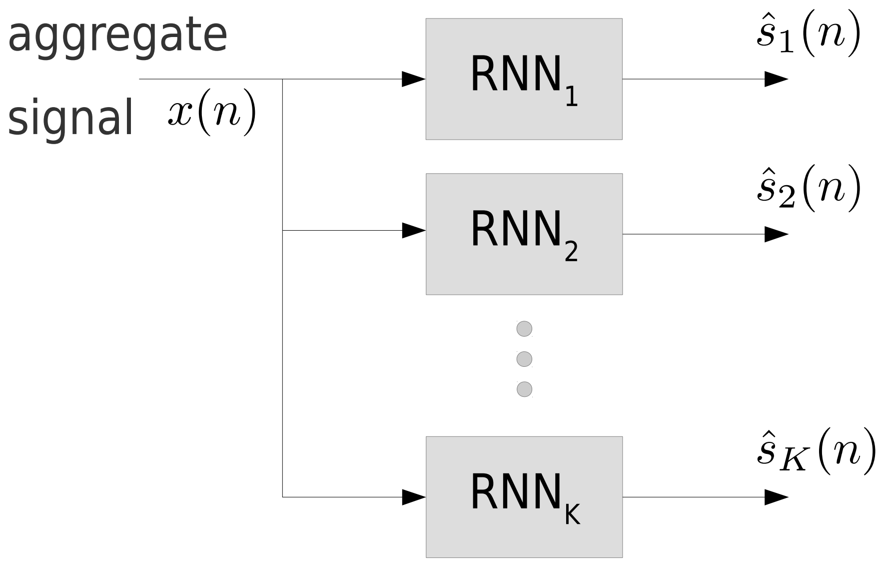
Cost

$$J(\boldsymbol{\theta}) = \sum_{m=1}^{M} \sum_{n=1}^{B} (y_m(n) - t_m(n))^2 + \lambda_1 ||\boldsymbol{\theta}||_1 + \lambda_2 ||\boldsymbol{\theta}||_2^2$$

Optimization

stochastic gradient descent | momentum | learning reate decay

Extraction of multiple loads

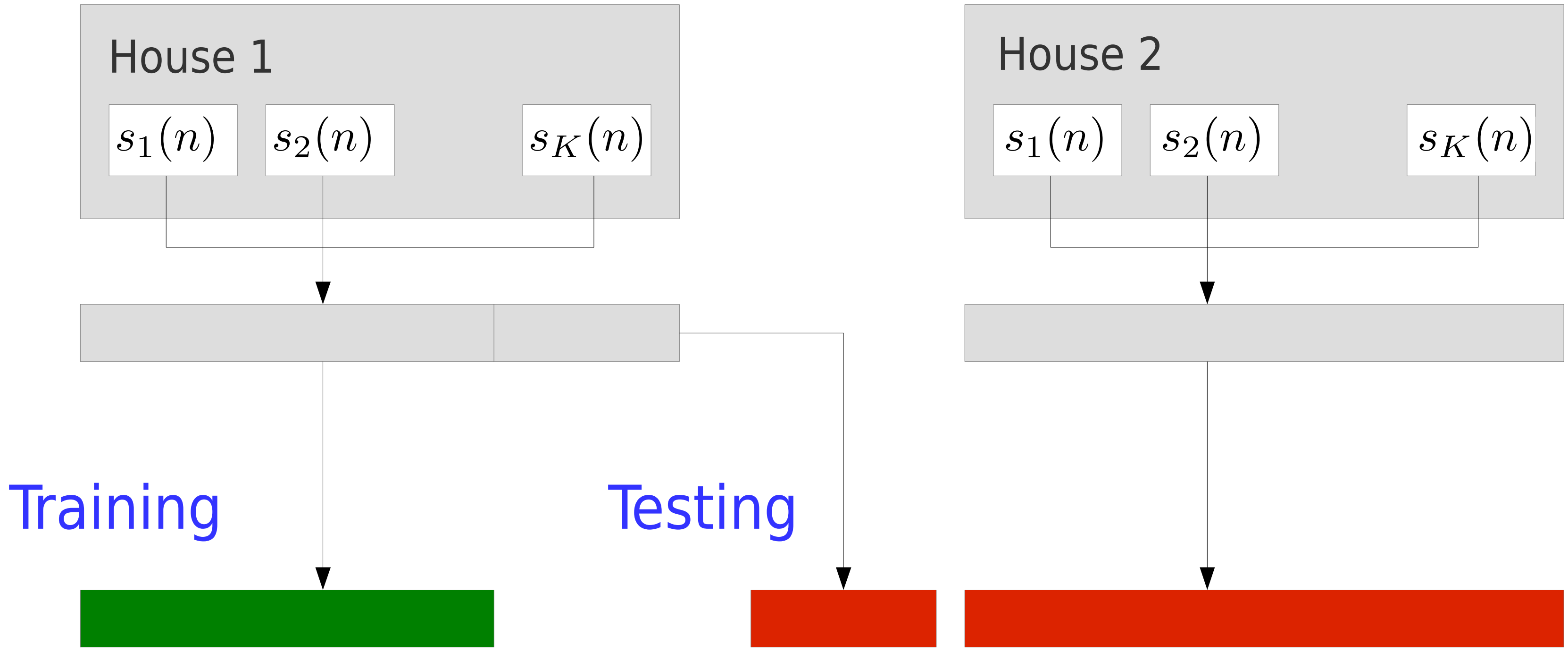


- Train multiple models by using mltiple submeter measurements
- Use one model to extract one major load separately out of the aggregate signal
  → easily extendable to new loads

Using Reference Energy Disaggregation Dataset (REDD)

- #loads: K=16
- #hours: 620h

- #loads: K=9
- #hours: 258h



Experiments

Network setup

- Input layer
  - $N^{(0)} = 10$
- Two recurrent layers
  - $N^{(1)} = N^{(2)} = 140$
- Output layer
  - $N^{(L+1)} = 1$
- #Parameters 485801

Target appliances

- Refrigerator
  - on/off device
  - periodic power consumption
  - small amplitude
- Dishwasher
  - multi-state device
  - nonperiodic
  - fixed pattern
- Microwave
  - multi-state device
  - nonperiodic
  - random pattern

Metrics

- Estimated energy

$$\hat{E}_t = \frac{1}{F_s} \sum_{n=1}^{N} \hat{s}_t(n)$$

- Consumed energy

$$E_t = \frac{1}{F_s} \sum_{n=1}^{N} s_t(n)$$

- NRMS

$$\text{NRMS} = \sqrt{\frac{\sum_{n=1}^{N} (\hat{s}_t(n) - s_t(n))^2}{\sum_{n=1}^{N} s_t^2(n)}}$$

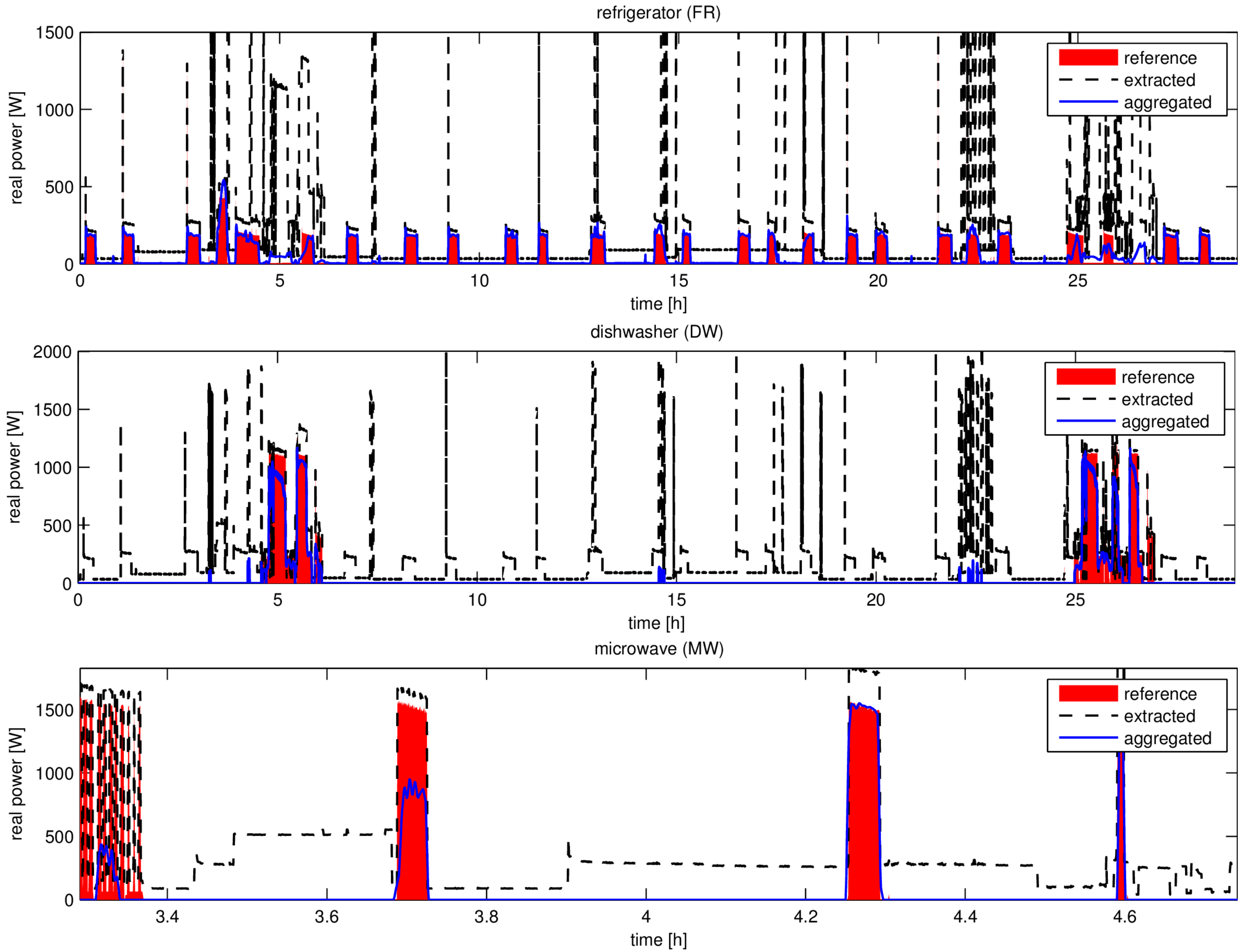
For active periods

$$s_t(n) \geq \gamma, \ \hat{s}_t(n) \geq \gamma$$

- Precision
- Recall
- F1 score

Experiments

Results for house 1



- details in following MATLAB demonstration

Experiments

Metrics for validation on house 1

Appl.	E_t	$\hat{E}_t$	NRMS	F1	R	P
FR	23.9	23.0	0.33	0.91	0.98	0.85
DW	11.1	10.50	0.35	0.79	0.87	0.73
MW	7.8	7.9	0.74	0.66	0.83	0.54

Table 1. Validation on test set of house 1 with $E = 63.37kWh$

Metrics for validation on house 2

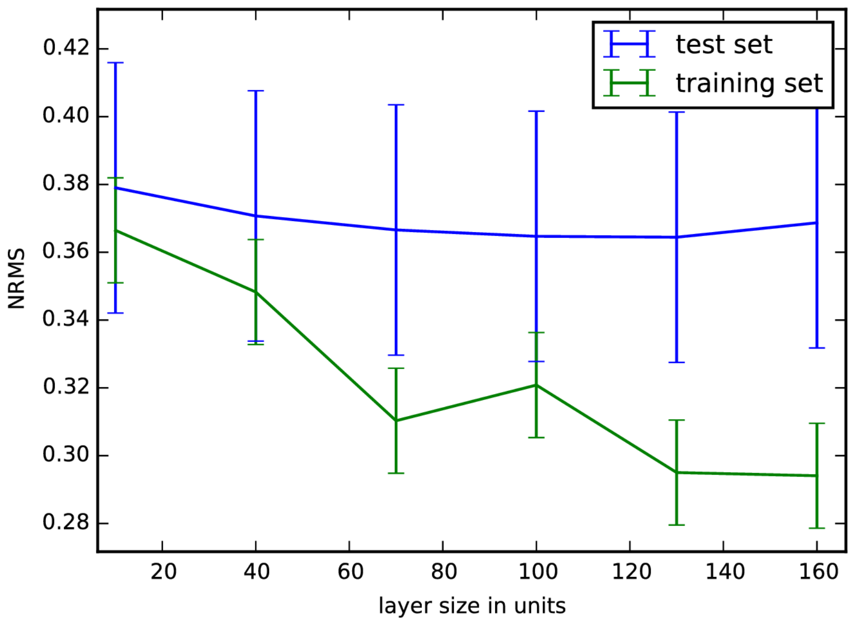
Appl.	E_t	$\hat{E}_t$	NRMS	F1	R	P
FR	20.7	20.6	0.35	0.93	0.96	0.91
DW	2.36	3.26	0.31	0.68	1.0	0.52
MW	4.0	2.11	0.58	0.09	0.05	0.5

Table 2. Validation on house 2 with $E = 36.6kWh$

- Models trained from house 1 work well for house 2 → high robustness

Experiments

Overfitting to training set



- result heavily dependent on initialization
- larger layer allows for more complex mappings
- network tends to overfit to training data
- increase of validation error between 120 and 160 units layer size chosen to 140 units

Conclusion

Advantages of the approach

- Bidirectional RNN can be used for supervised load disaggregation
- Good performance for appliances with recurring patterns
- Eventless for all types of loads
- Allow low-frequency (<1Hz) power meter
- No feature engineering

Drawbacks

- Need submeter data
- Networks tend to overfit for little training data

Future work

- Combination of DNN and HMM for disaggregation
- Domain adaption for different loads of same kind