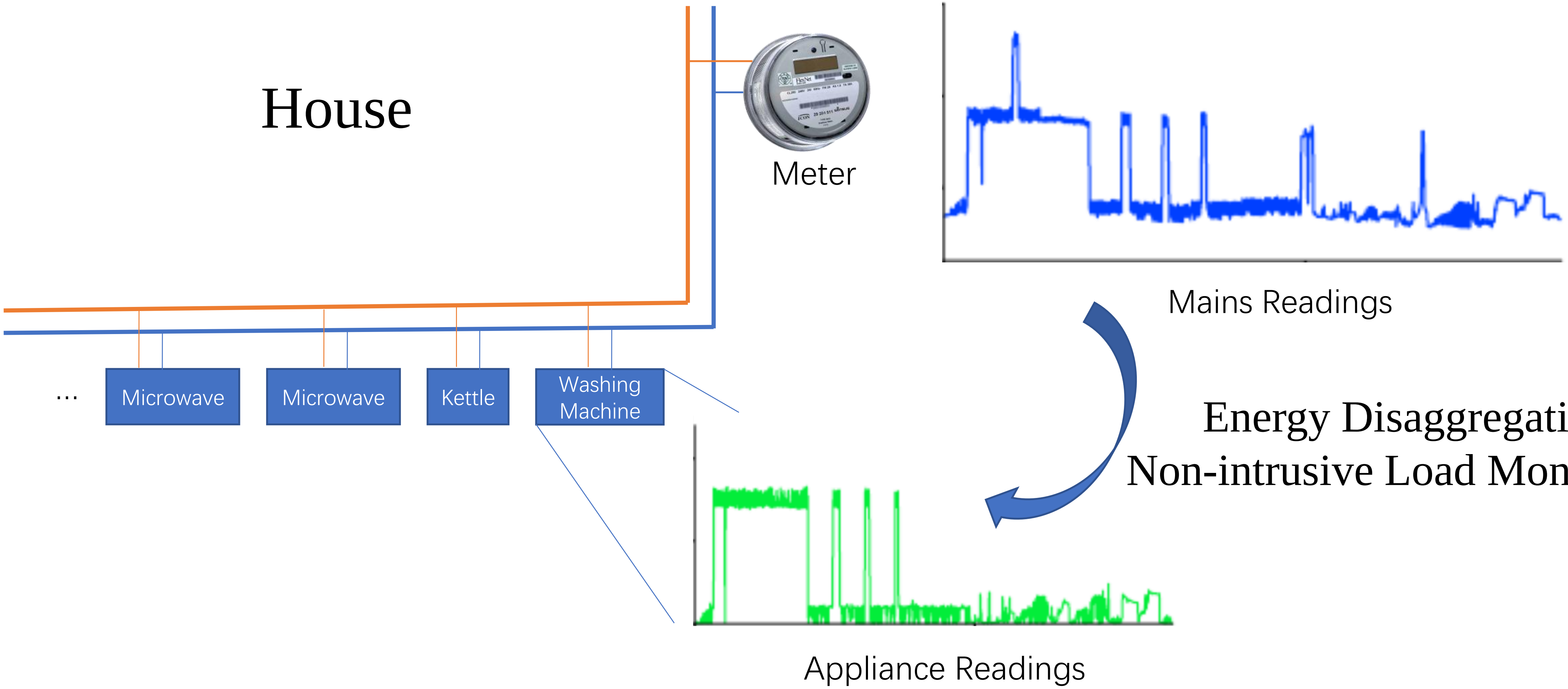
SEQUENCE-TO-SUBSEQUENCE LEARNING WITH CONDITIONAL GAN FOR POWER DISAGGREGATION

Yungang Pan, Ke Liu, Zhaoyan Shen*, Xiaojun Cai, Zhiping Jia

Outline

- Introduction
- Method
  - Sequence-to-Subsequence
  - Architecture
    - Overview
    - Design principles
    - Model
  - Training and Inference
- Evaluations and Results
- Summary

Introduction



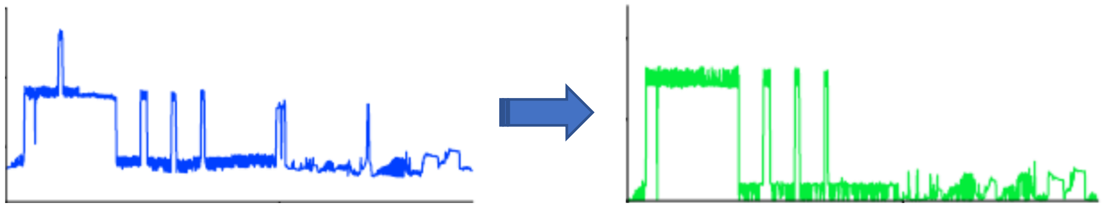
Energy Disaggregation
Non-intrusive Load Monitoring

Introduction

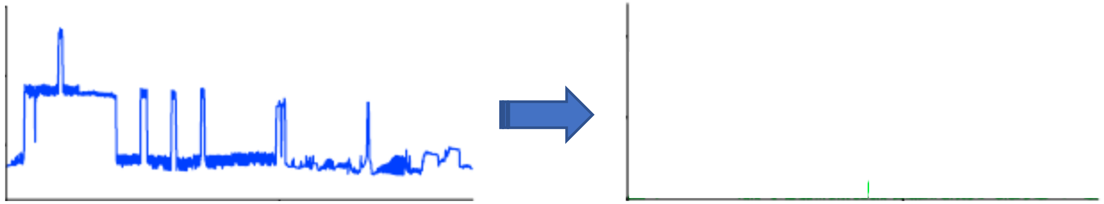
- Applications:
  - Detailed bill information
  - Occupancy detection
  - New companies services
  - Illegal load detections
- Common techniques:
  - Autoencoders
  - Hidden Markov Models
  - Deep Learning
    - Convolutional Neural Network
    - Long Short Term Memory

Sequence-to-Subsequence

Sequence-to-Sequence[1] **Difficult to converge**

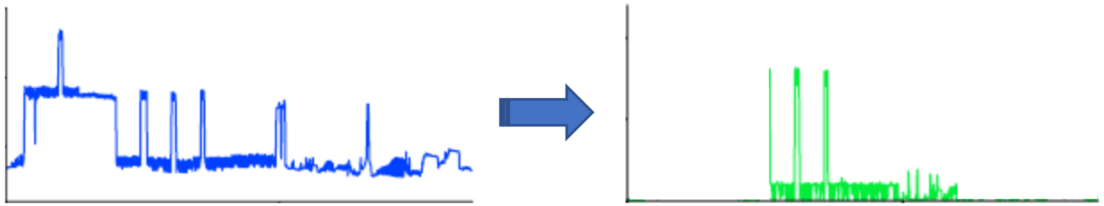


Sequence-to-Point[2] **Too much computation**



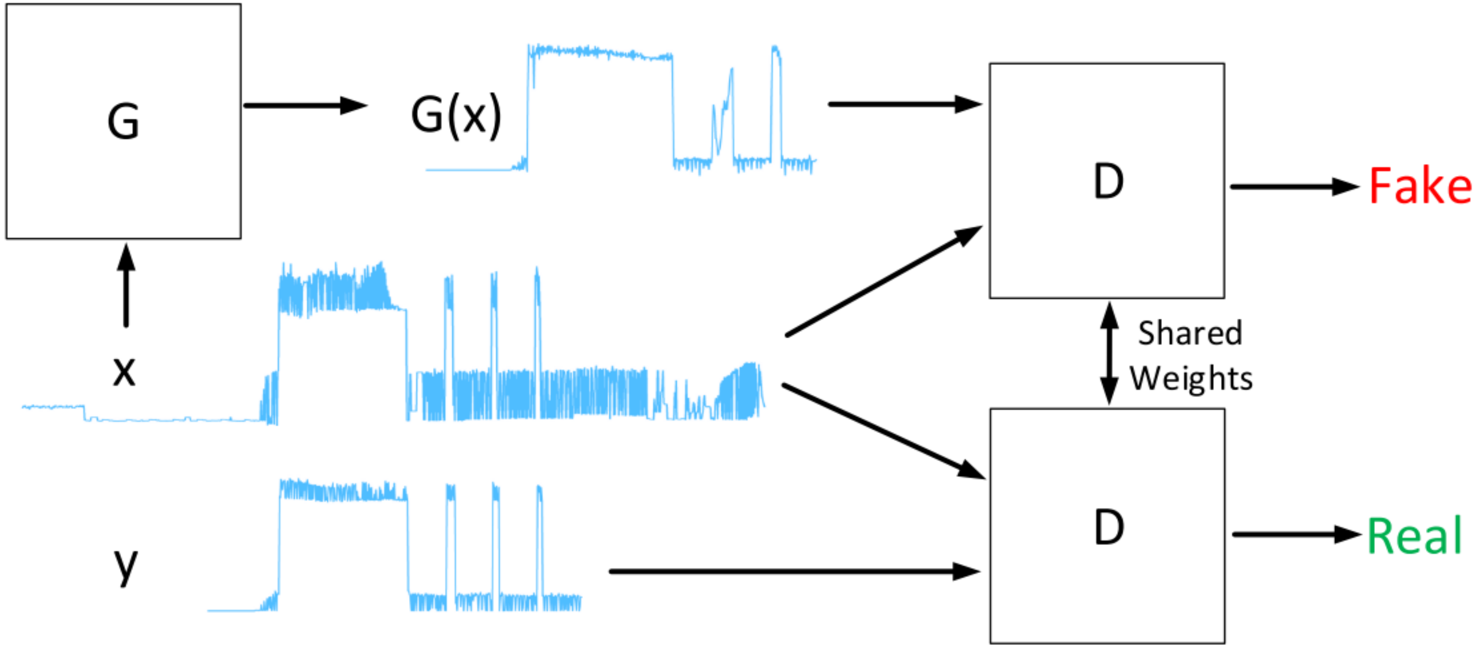
Tradeoff

Sequence-to-Subsequence



Overview

We model the energy disaggregation problem as a **sequence generation** problem.

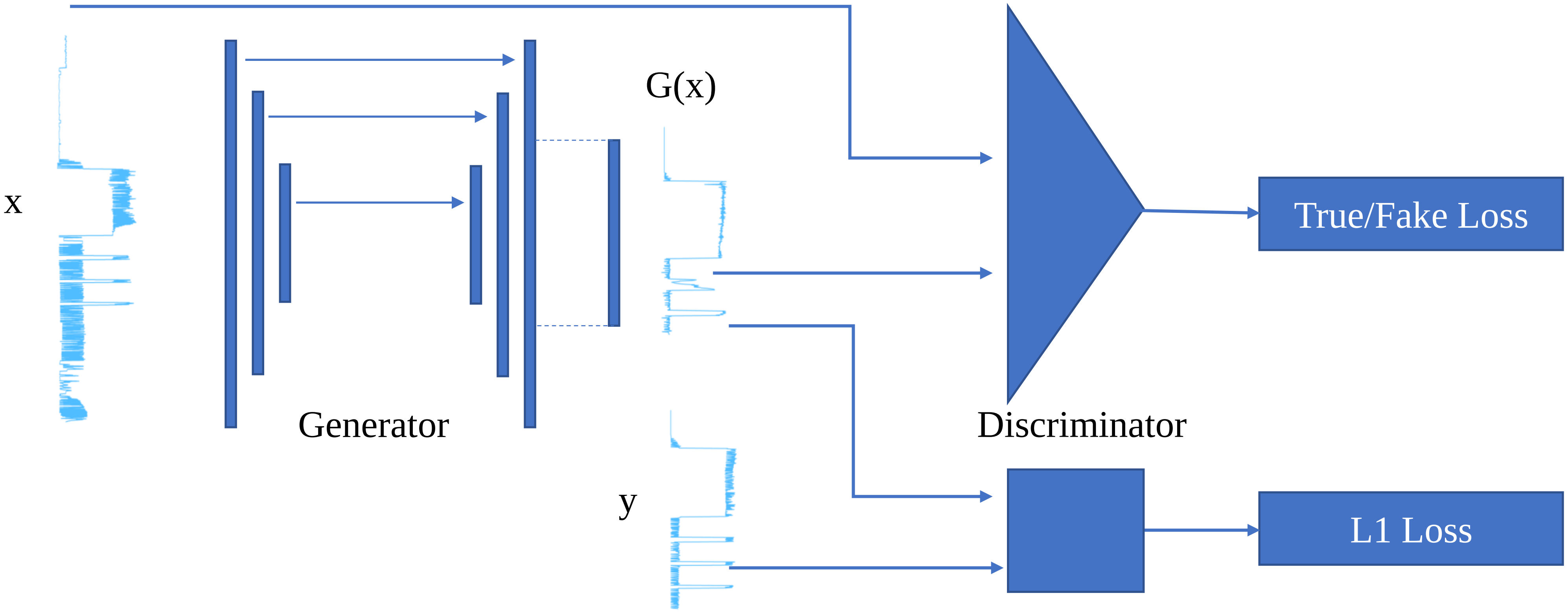


x: Mains readings
y: The corresponding real appliance readings
G(x): The generated appliance readings

Design Principles

- The underlying structure between the mains readings and the individual appliance readings should share some common characteristics or be similar in many occasions. (U-Net [3])
- Learning the model from too many examples yield poorer qualitative results compared with a smaller example set. (Instance Normalization [4])
- Use L1 loss to encourage the model to capture the low-frequency information, while applying the fully convolutional network (FCN [5]) into discriminator to optimize the quality of high-frequency information.

Model



Objective function of traditional GAN:

$$\min_G \max_D V(D,G) = \mathbb{E}_{y\sim P_{data}(y)}[\log \mathrm{D}(\mathrm{y}|\mathrm{x})] + \mathbb{E}_{z\sim P_z(z)}[\log(1 - \mathrm{D}(\mathrm{G}(\mathrm{z}|\mathrm{x})))])$$

Objective function of our model:

$$\min_G \max_D V_{\mathcal{L}1}(D,G) = V(D,G) + \lambda\mathbb{E}_{z\sim P_z(z),y\sim P_{data}(y)}[||y - G(z|x)||_1])$$

Training and Inference

- Training phase:
  - One model corresponds to one kind of target appliance.
  - Applying instance normalization requires that the batch size equals to 1.
  - Discriminator and Generator are trained alternatively-one step on Discriminator, then another step on Generator. We use SGD optimizer for Discriminator and Adam optimizer for Generator.

- Inference phase:
  - Only Generator works. Each execution produces one corresponding appliance sequence.

Evaluations

- Competitor solutions:
  - Sequence-to-Point Method proposed in literature [2]
- Data sets:
  - UK-DALE [6]
  - REFIT [7]
- Essential parameters:
  - Readings interval: 6 seconds
  - Window width of mains readings : 1024
  - Window width of target appliance readings : 512
  - Batch size: 1
  - The ratio between off-class and on-class of the target appliance: 1
  - The ratio between L1 loss and discriminator loss λ: 100

Evaluations

- Measure Metrics:
  - normalized Signal Aggregate Error (SAE)
  - Mean Absolute Error (MAE)
  - Estimated Accuracy (Acc.)
- For transferability:
  - Test the model on ”unseen” houses during the training process
- Implementation:
  - Python
  - TensorFlow
  - NVIDIA RTX 2080
- Source code:
  - https://github.com/DLZRMR/seq2subseq

Results

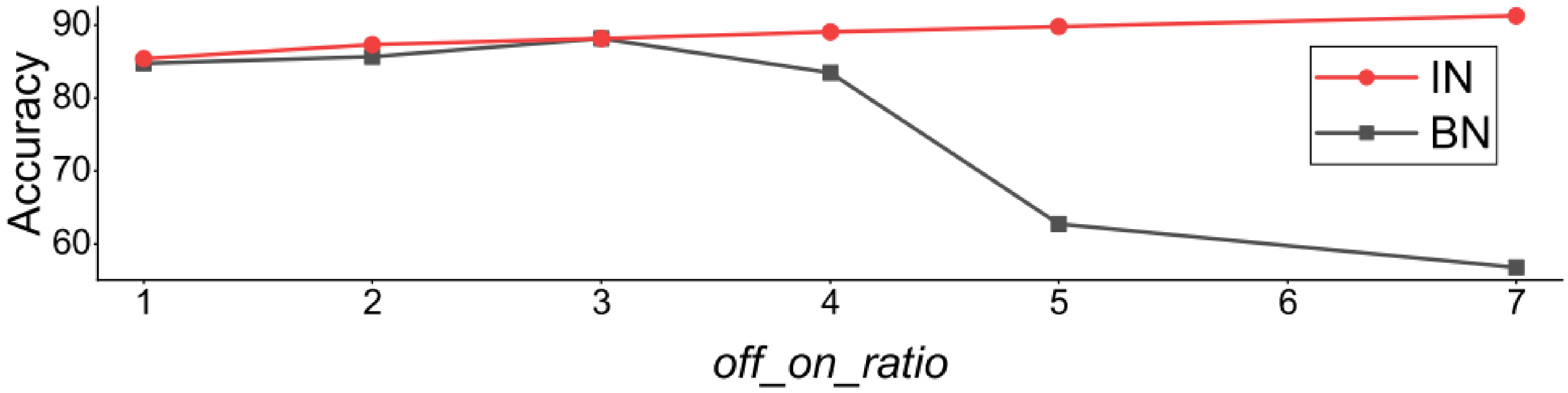
Performance comparison between seq2subseq and seq2point on UK-DALE and REFIT. The best results are shown in bold.

Metric	Methods	WM	KE	MW.	FR	DW	Avg
MAE	U(ours)	**7.11**	**3.59**	**3.14**	**11.86**	**13.52**	**7.84**
	U(s2p)	12.66	7.44	8.66	20.89	27.70	15.47
	R(ours)	**16.72**	**6.43**	**5.88**	**16.77**	**4.80**	**10.12**
	R(s2p)	16.85	6.83	12.66	20.02	12.26	**13.72**
SAE	U(ours)	**0.120**	**0.026**	**0.030**	**0.070**	**0.370**	**0.123**
	U(s2p)	0.284	0.069	0.486	0.121	0.645	0.321
	R(ours)	**0.162**	**0.106**	**0.125**	**0.100**	**0.172**	**0.133**
	R(s2p)	2.610	0.130	0.170	0.330	0.260	0.700

Our method outperforms seq2point (s2p) in all metrics for all appliances, reducing MAE by 97%, SAE by 161% for UK-DALE and MAE by 35%, SAE by 426% for REFIT on average, respectively.

Results

Accuracy of Washing Machine on ”unseen” house 2 from UK-DALE using Instance Normalization and Batch Normalization (batch size=32), respectively.



This phenomenon shows that too many examples and high *off_on_ratio* will have negative effects on the results.

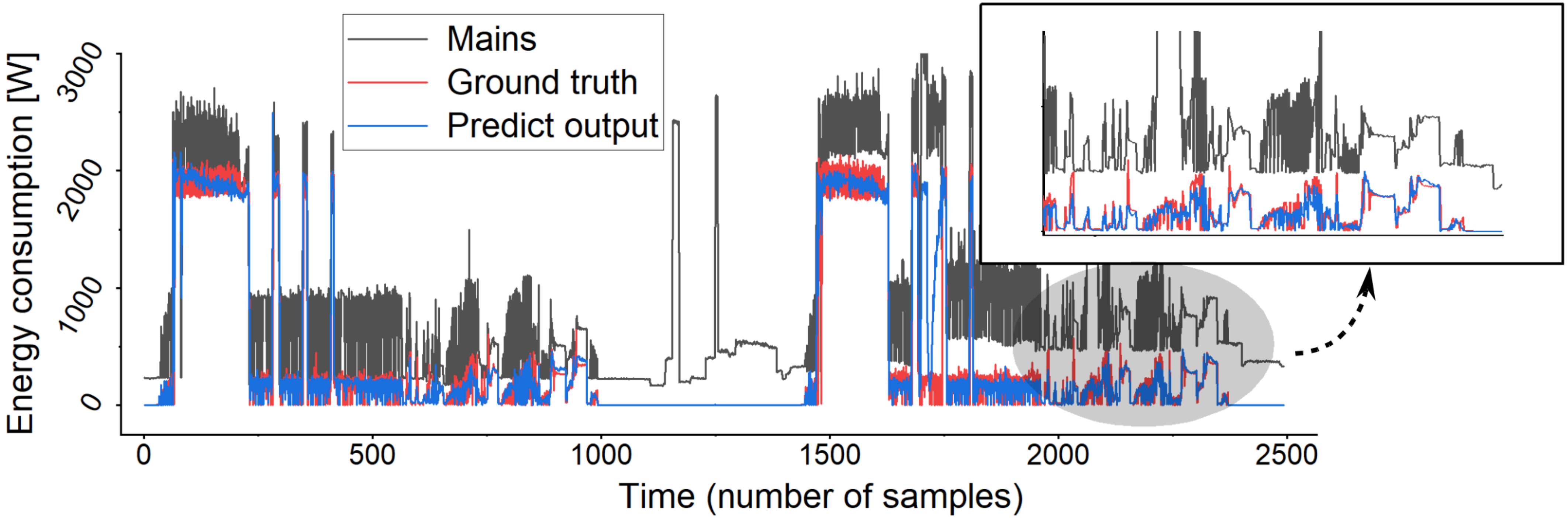
Results

Testing on ”unseen” UK-DALE house 2, after training our models on all other UK-DALE houses. The best results are shown in bold.

Metric	Methods	WM	KE	MW	FR	DW	Avg
Acc.	I+U	**85.2**	**90.3**	**77.5**	**85.3**	**77.2**	**83.1**
	N-I+U	57.0	72.2	58.5	56.2	51.6	59.1
	I+N-U	79.5	50.0	49.9	79.9	50.0	61.9
	N-I+N-U	50.0	50.0	50.0	50.0	50.0	50.0
MAE	I+U	**7.11**	**3.59**	**3.14**	**11.86**	**13.52**	**7.84**
	N-I+U	20.65	10.34	5.77	35.44	28.68	20.18
	I+N-U	9.87	18.6	6.96	16.21	29.64	16.26
	N-I+N-U	24.03	19.86	6.96	40.47	29.65	24.19
SAE	I+U	0.120	**0.026**	**0.030**	0.070	**0.370**	**0.123**
	N-I+U	0.317	0.081	0.365	0.477	0.709	0.390
	I+N-U	**0.034**	1.000	1.000	**0.062**	1.000	0.619
	N-I+N-U	1.000	1.000	1.000	1.000	1.000	1.000

Applying both Instance Normalization and U-Net to our model can improve the performance dramatically.

Example Outputs



Example outputs of Washing Machine on ”unseen” house 2 from UK-DALE (We enlarge the right part of the figure to make it clear).

Summary

- We propose the **sequence-to-subsequence** learning method, balancing the convergence difficulty in deep neural networks and the amount of computation during the inference period.
- Build the model based on conditional GAN.
- With discriminator and L1 loss, our model has the ability to capture the low-level and high-level information simultaneously.
- The U-Net and instance normalization techniques improve the performance of our model.

References

[1] Jack Kelly and William Knottenbelt, “Neural nilm:deep neural networks applied to energy disaggregation,” 2015.
[2] Chaoyun Zhang, Mingjun Zhong, ZongzuoWang, Nigel Goddard, and Charles Sutton, “Sequence-to-point learning with neural networks for nonintrusive load monitoring,” 2017.
[3] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, “U-net: Convolutional networks for biomedical image segmentation,” in International Conference on Medical Image Computing & Computer-assisted Intervention, 2015.
[4] Dmitry Ulyanov, Andrea Vedaldi, and Victor S. Lempitsky, “Instance normalization: The missing ingredient for fast stylization,” CoRR, vol. abs/1607.08022, 2016.
[5] Long, Jonathan, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.
[6] Jack Kelly and William Knottenbelt, “The uk-dale dataset, domestic appliance-level electricity demand and whole-house demand from five uk homes,” Scientific Data, vol. 2, pp. 150007, 2015.
[7] D Murray, L Stankovic, and V Stankovic, “An electrical load measurements dataset of united kingdom households from a two-year longitudinal study,” Scientific Data, vol. 4, pp. 160122, 2017.

Thank you!

Questions?