Non-Intrusive Load Monitoring

Mahir Jethanandani, Chester Leung, Andrew Nam
University of California, Berkeley

Introduction

Non-intrusive load monitoring (NILM) is the process of predicting individual appliance contribution to aggregate, i.e. whole house, electrical consumption. We modified this task to focus on the opposite direction: this project aims to cluster individual device behavior and recover power consumption information. The motivation for focusing on this direction, rather than the opposite (disaggregation) direction, is to create an inverse representational form of vectorized device states with respect to the disaggregation task. The aggregation model utilizes silhouette values and K-Means/DBSCAN clustering, alongside the sister Seq2Seq model utilized in similar non-intrusive load monitoring tasks [0].

Previous Work

- In 1992, Hart published the genesis NILM paper [1]. The simplified paper focused on two-mode appliances.
- Later work included Hidden Markov Models [2] [3] and clustering.
- Deep neural networks [4] were applied in 2015. LSTMS and denoising autoencoders were used.
- Similar work [0] predicted removed middle energy points in a sliding window.

Approach

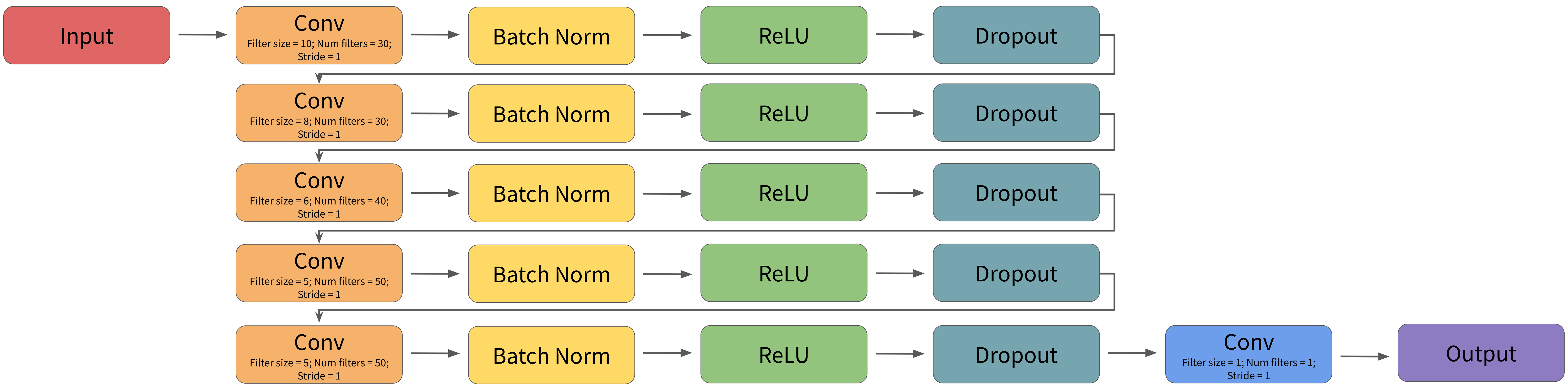
Preprocessing began with cleaning input power data and clustering methods. Using silhouette values to select the number of kernels, K-Means clustering categorized power data into modes. K-Means and DBSCAN formed an ensemble for clustering, which also transformed power values into normalized and centered cluster values.

The final model was inspired by previous work done in a slightly different setting [0]. Instead of training a Seq2Point model to predict the power consumption at a specific time, this project designed a Seq2Seq model to forecast individual appliance power consumption at each time step over 500 steps.

We used the AMPds dataset [6] to train, validate, and test our model. The 1M training points over the course of months were split into 70/20/10 percentages. Our loss measurement is mean squared error (MSE) and absolute error. Each device ranges in watts range, so loss comparison between devices is impractical.

The pipeline was developed in Pandas, NumPy, Scikit-Learn, and PyTorch, and the model was trained on a Tesla K80 GPU. Special thanks to Oladapo Afolabi for mentorship throughout the project.

Model Architecture



Results

The final model was trained with a scheduled learning rate over 50 epochs and a batch size of 32. The data was divided into 700,000 training, 200,000 validation, and 100,000 test samples. Training used a stepped learning rate with an initial value of 1e-2 and decays by 0.01 every 10 steps. For dropout, the hyperparameter was tuned to $p = 0.3$.

The model did well at predicting lower power consumption but took much longer to learn higher values. Over training, the bias and variance of the high power predictions decreased [Figure 2].

Between devices, the model performed worse on devices with more modes (Dryer, Pump).

Devices	Training (Final)		Validation (Final)		Testing (Mean)	
	MSE	Absolute	MSE	Absolute	MSE	Absolute
Fridge	112.5487	10.6089	118.1467	10.8695	142.0701	11.9193
Thermostat	259.8036	16.1184	392.4260	19.8097	497.8734	22.3131
Clothes Dryer	4802.7510	69.3019	10331.4512	101.6437	8248.1973	90.8196
Security System	24.8668	4.9867	27.2975	5.2247	40.1975	6.3401
Heat Pump	5020.4248	70.8550	9601.7207	97.9884	24132.6777	155.3470

Figure 1: Table of Losses. Included the training, validation and testing losses across different household device types. Note: Devices have different power ranges which will impact the magnitude of the losses

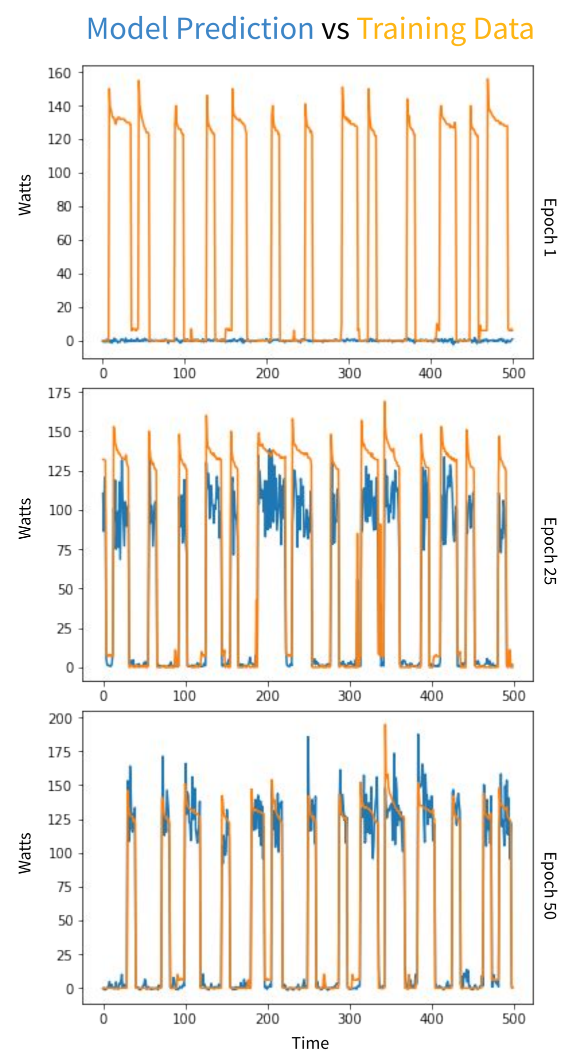


Figure 2: Model and Prediction data for Fridges

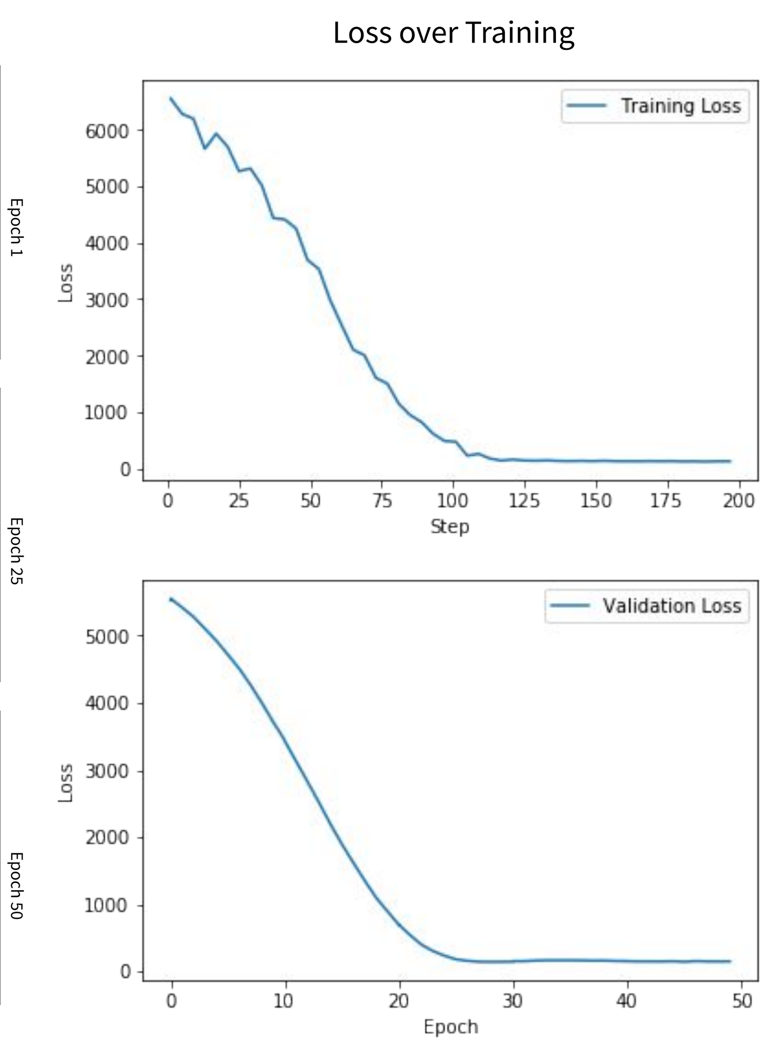


Figure 3: Training and Validation loss for Fridges

Future Work

This project accomplishes the aggregation direction, i.e. mapping sequential cluster values into power data. The disaggregation direction attempts the reverse direction. The motivation for this inverse aggregation direction is to provide a trained model to predict power usage given appliance modes. With additional layers and modifications, the current model architecture can possibly be adapted to solve disaggregation. As it stands, the architecture can handle multiple devices by adding up predicted power values for all devices across each time step. Convolutional layers and batch normalization are ever-present in time-series related tasks, so should deep learning advancements arise to replace such layers, the model could achieve greater representation. Finally, the model is invariant to time, so input as a function of time could provide unforeseen information gain.

References

[0] Zhang, C., Zhong, M., Wang, Z., Goddard, N., & Sutton, C. (2018, April). Sequence-to-point learning with neural networks for non-intrusive load monitoring. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
[1] Hart, G. "Nonintrusive appliance load monitoring", Proceedings of the IEEE, v. 80, n. 12, pp. 1870–1891, Dec 1992. ISSN: 0018-9219. doi: 10.1109/5.192069.
[2] Ghahramani, Z., Jordan, M. I. "Factorial Hidden Markov Models", Mach. Learn., v. 29, n. 2-3, pp. 245–273, nov. 1997. ISSN: 0885-6125. doi: 10.1023/A:1007425814087.
[3] Parson, O. "Unsupervised Training Methods for Non-intrusive Appliance Load Monitoring from Smart Meter Data", April 2014.
[4] Kelly, J., & Knottenbelt, W.J. (2015). Neural NILM: Deep Neural Networks Applied to Energy Disaggregation
[5] Batra, N., Kelly, J., Parson, O., et al. "NILMTK: An Open Source Toolkit for Non-Intrusive Load Monitoring". In: International Conference on Future Energy Systems (ACM e-Energy), 2014.
[6] http://ampds.org/