Applying Gated Recurrent Unit (GRU) Method for Appliance Identification Across Multiple Household Smart-Meter Datasets

Darius Ghomashchian
Department of Electrical and Electronic Engineering
Heriot-Watt University
Edinburgh, Scotland
Dg125@hw.ac.uk

***Abstract*— Energy disaggregation is the method of monitoring aggregate power loads at the end-user point to determine appliance level energy demand. Recent smart-meter technology has enabled this with non-intrusive load monitoring (NILM) techniques which allow access to the end point data using sensor technology, without having to use sub-metering methods to measure every power load in a building or home. As smart meters collect the power aggregation of multiple devices in the home, AI-based techniques can be deployed to determine the type of appliance and the time of usage amongst other useful data. These are key classifications which can be used to save energy and money for the end user and energy providers. In this paper, relevant techniques at the forefront of NILM will be reviewed, an adapted LSTM method (Gated Recurrent Units) will be proposed with a focus on classifying highly demanding loads across multiple merged datasets. This is to test reliability of datasets being used in isolate and the integrity of the appliance data being used. Results of the method are compared with benchmark algorithms using the NILMTK API. Key issues and future work in the field will be discussed.**

Keywords—Energy-Demand, Energy Forecasting, Smart-meter analysis, Energy Disaggregation, NILM.

I. INTRODUCTION

Interest in smart meter data and energy disaggregation has been growing in recent years while energy companies become more aware of the opportunity that it brings to save on money and resources in parallel with the adoption of Big Data Analysis (BDA) techniques transforming many other industries. Residential living is accustomed to high demand with an expected growth of 30% between 2018 and 2050. As energy companies expand their resources to meet demand, newer technologies have been introduced to assist in demand monitoring and network management.

With the introduction of smart-meter technologies much of the aggregate energy demand can be monitored to a high degree of accuracy by placing meters in user households to monitor end point usage. By introducing modern sensor technology, these meters can provide data on power usage without having to measure this from the source. Huge amounts of smart meter readings will increase as more are installed and frequency of data collection increases. Research has been shown that 5-15% of the residential energy consumption can be saved on the demand side [2] by analysing the smart meter data and enhancing the efficiency of generation networks on the supply side. Smart meter technology translates to reduced prices to the end customers and enables the possibility for energy suppliers to have a more efficient and safer supply network.

One of the motivations for using energy disaggregation methods is to be able to find the appliances being used and therefore be able to offer consumers an itemised energy bill. By making the consumers more aware of their appliance energy demand it encourages them to make more deliberate decisions towards appliances which have less energy consumption. There is also the opportunity to automate the devices recognised through some sort of energy demand software.

II. Key Issues In NILM

Although there has been steps forward in recent years largely due to the energy sector adopting data methods to improve their networks, there are still many challenges in NILM and limitations in its application. The ever-increasing number of multi-state devices being introduced into homes make classification of appliance loads challenging as well as the variance in existing loads. Looking at some of the leading methods for classification most of the datasets utilise a known dataset with ideal environmental conditions. There are many issues to consider in the datasets and in NILM in general.

A. *Load Variance*

Sampling from reliable data sources has its challenges as the appliance data itself does not have a consistent, universal power demand. There is a high variability in appliance numbers, size, type and make, as well as user privacy restrictions to consider. For example, the power load of a washing machine in one home may share some aggregate signal characteristics such as transient power signatures and time-of-day usage features but there will be a high variance in overall power consumption due to different parameter settings on each washing machine. In addition, there are several appliance types which are difficult to distinguish between, such as low-powered devices which share similar signatures, variable state devices such as washing machines, and continuous state devices such as dimming lights. In general, NILM performs better on two-state devices such as lights or toasters. Much of the research requires the devices to be trained and labelled offline which created problems when new devices are introduced to the ecosystem. If the considered goal of NILM is to apply a one-fits all algorithm across a full energy network, then this problem will need to be addressed and should be considered in the experimentation.

B. *Sampling Frequency*

The smart-meter networks deployed can generally be classified as low-frequency or high-frequency sampling meters. High-frequency data is costly; however, it obviously provides much more detail in energy signatures and therefore there is more opportunity to deploy AI-methods of varying types. Unfortunately, low frequency has had to be made the primary data repository format as this is what is currently available at scale.

It should also be mentioned that as all datasets use different sampling frequencies, it is not always possible to conduct research across multiple datasets. Experiments are limited to using low-frequency sampling techniques for this reason.

C. *Dataset Dependancy*

Much of the research done in this area use predominant set of datasets such as REDD or UK_DALE which are reliable, free to anyone and offer high frequency sampling. However, basing the research on each dataset individually leaves each independent experiment vulnerable to overfitting. Many features of the dataset such as the location, temperatures, cannot be implicated in the research as they cannot be considered as a feature unless compared to other datasets. Therefore, experimentation of algorithms across datasets is an interesting and useful method to explore.

It is also important to consider the environments of the datasets which are being used in NILM. The five popular datasets benefit from ideal conditions experienced in developed countries [3]. There is no high frequency dataset available from less prosperous regions which have higher populations and therefore higher long-term energy needs.

III. General Framework Of NILM

To disaggregate smart meter data the standard approach is to install individual plugs to send utility specific data for each device it is expensive (50 dollars each plug) therefore we want to accurately report it without installing hardware therefore we would use NILM. At the household level given a smart meter can report minute to minute hour to hour consumption can desegregate this to device level data without having to have plugs installed on appliances in-the home.

This is an analytical approach an alternative to hardware approach (high-priced) traditional sub-metering), where the upside is to have the same level of information on appliances in the house using a computer algorithm cost effectively, efficiently aggregate demand-response data. This analytical involves the processes of NILM needed to carry out data analysis.

NILM techniques typically follow the same framework as seen in figure 1:

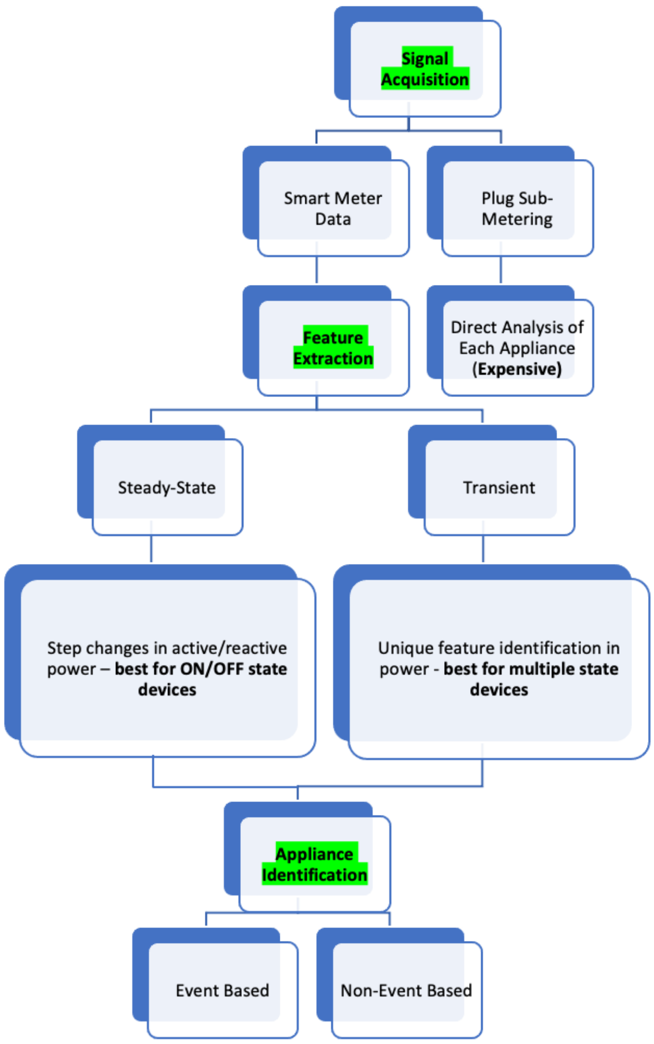

Figure 1 : Taxonomy of NILM Technique.

A. *Signal Acquisition*

Typically, the datasets used in research contain data on the aggregate level which would be obtained from the mains electrical supply of the building/home (smart meter) and the individual level which will be obtained from a sub-metering method (attaching plug meters) on each load in the home/building. Frequency varies but it will usually have a high sampling frequency of around 1-2Hz.

B. *Feature Extraction*

There are two main classes of appliance signatures:

- Steady state.
- Transient.

Both approaches refer to identifying changes in the operation when an appliance modulates from one operational state or level to another, but the two approaches differ in what data they focus on.

- Steady state signature is one of most widely used signatures in literature, refers to step changes in the steady-state active and/or reactive power consumption levels. Easy to use, does not require fast sampling data and works well on large ON/OFF alliances such as water heaters and air-conditioners.
- Transient appliance signatures more complicated, refers to unique features such as shape, duration, size, harmonics (frequencies) of transient power fluctuations of appliances that can be used to distinguish different appliances. These short-term transient fluctuations usually occur immediately after changes in an appliance's operating state (e.g. OFF/ON) and before a new stead-state is reached. Although transient analysis gives more precise information, extracting this type of signature requires high-frequency data sampling. As a result, more resources are needed to construct and maintain a complex transient signature database.

Transient state analysis might be more suited to appliances which do not have clear on/off signatures or may have multiple power levels attached to them.

C. *Appliance Identification*

In general, disaggregation algorithms can be categorized as event-based, or non-event based. The difference being that event-based refer to approaches that rely on edge detection algorithms to detect occurrences of events, such as an appliance turning ON/OFF or a change in the operating mode of the appliance. The extracted features around the neighborhood of the event points are then classified using supervised models.

Non-event based appliance identification does not have an event-detection mechanism, instead it relies on every sample of the aggregated power signatures to be logged [4]. An example algorithm would be Hidden Markov Models (HMMs) which will use all sample points as data input.

IV. Review of Existing Methods

There are several benchmark algorithms that are used in energy disaggregation which are regularly researched and implemented. Of course, there is no universal algorithm which has been discovered as the models usually optimise for a specific purpose and for reasons discussed with the data it would be unrealistic to expect any one algorithm to work ideally at this point. Promise has been shown in some algorithms to work across different datasets and will be discussed below.

A. FHMM

Factorial Hidden Markov Models is one of the most commonly applied algorithms in the field of NILM as it has been used successfully in other fields to model stochastic and discrete time series data. The basic idea of Markov Models is similar to how the human brain operates at the individual neuron level, where a neuron's action potential (or spike) has an all-or-nothing reaction to input signals and depends only on the current state of inputs, it is not affected by any other neuron states. In other words, its future states depend only on its current and prior state.

HMMs assume two properties which must be satisfied; the first is that an observation at time t is generated by a process whose state at time t is hidden, therefore implying that the Markov model underlying the process is hidden and the observations are an implication of these hidden states, not the observable ones. The hidden state is a multinomial variable which encodes information on the time series history. The second property is the Markov Property which states that the current state only depends on the value of the previous states and no prior states.

HMM's use a series of observations $\{Y_t\}$ where t = 1,…T which are modelled in relation to a series of hidden states [11] $\{S_t\}$ as shown in figure (…a). Two assumptions are made; first that Yt is independent of all other observations and states. Second, that state S_t is independent of S1…St-2 given St-1 (the first-order Markov property). Therefore, the joint probability for a sequence of states and observations is in eq(1):

$$\mathrm{P}\ (\{S_t, Y_t\}) \ = \ \mathrm{P}\ (S_1)\mathrm{P}\ (Y_1|S_1) \prod_{t=2}^{T} \mathrm{P}\ (S_t|S_{t-1})\mathrm{P}\ (Y_t|S_t) \qquad (1)$$

Equation (1) can be seen in graph form in figure 2(a). The state transition probabilities P $(S_t|S_{t-1})$ represent the arrows between states, while the observation probabilities P $(Y_t|S_t)$ represent the arrows from each state to each observation.

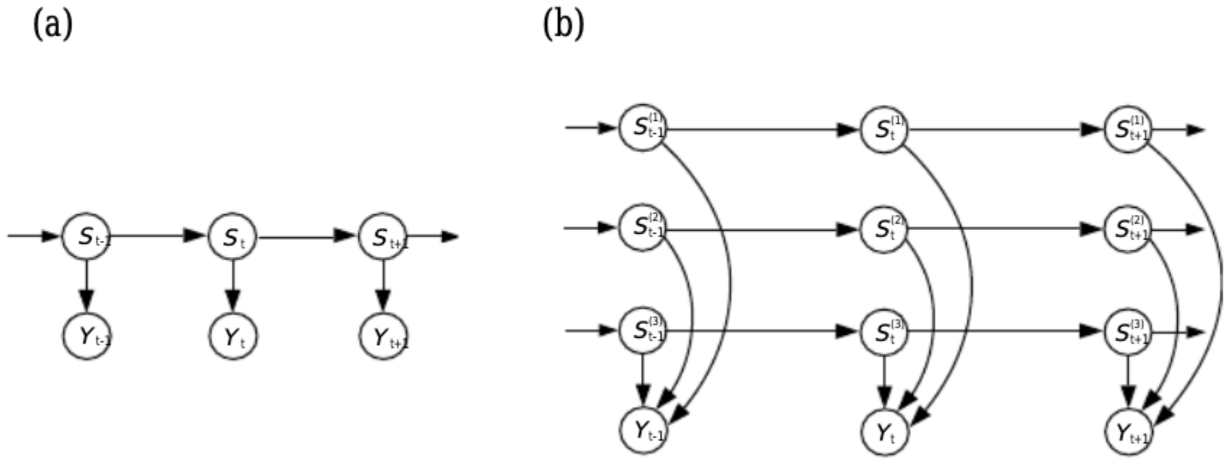

Figure 2 : a) HMM structure. b) FHMM structure.

In the context of FHMM, the state representation of the HMM can be abstracted to have multiple hidden state variables, all of which effect the observation variable as seen in figure (…b). The state space consists of the cross product of these state variables, which is why the method name is preceded by 'Factorial'. The FHMM attempts to break through some of the fallacies of HMM such as the time complexity and lack of state memory (Markov property) by decoupling state variables [11].

Ref [12] used FHMM in disaggregating appliance loads using the multiple hidden states to represent multi-state appliances. The results showed that *"the tested methods worked well for simple or modestly complex signatures"*. It was able to classify the loads of simple devices, however it doesn't perform well for more complex loads. The accuracy was around 78% for households with 8 appliances. Another limitation was that the algorithm required the number of active appliances in the signal as input before being able to disaggregate them.

Another application used hierarchical FHMM which utilized the relationship between device to improve the HMM inferencing speed and accuracy [8]. The algorithm was able to log the correlation between devices by saving the states of multiple appliances in each set of HMM state variables and then categorising them in a hierarchy by the states that occur most frequently. This demonstrates improved performance with adjustments to the FHMM capacity with inferencing between devices.

In general, FHMM performs consistently well however, the restrictions such as the high time and space complexity, as well as the memoryless property of Markov processes make it difficult to model and disaggregate complex loads with multiple states such as washing machines and dishwashers. It will likely remain one of the leading algorithms in NILM and there is ongoing active research on FHMM variants.

B. ANN

Artificial Neural Networks (ANN) are commonly used in the image recognition domain. They have the ability to recognize extreme variability in patterns [10]. ANNs are another brain-inspired algorithm which use sequences of nodes (neurons) which are interconnected with real-valued weights (ω) between 0 and 1. A neuron will become active if the sum of all weights going into it total a threshold value and this value holds after being passed through an activation function (e.g., sigmoid or ReLu). Typically, an ANN will be made up of an input layer, one more hidden layers and an output layer (figure 3).

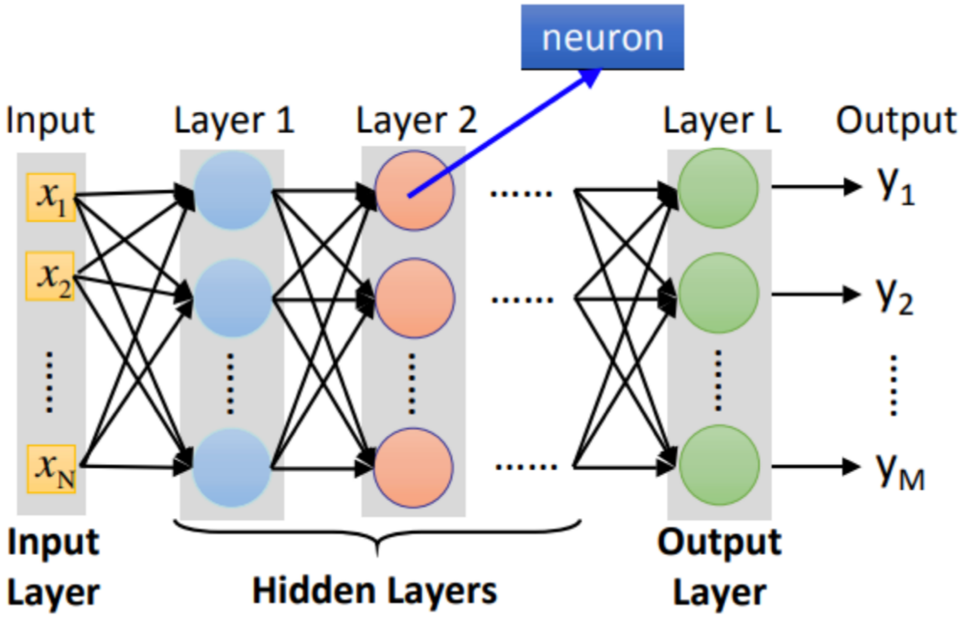

Figure 3 : ANN layout.

The output can be measured against the ground truth of the testing set and a loss function will be generated. There is then a learning algorithm applied (such as backpropagation) in order to readjust the weights to minimise the loss function.

Steady-state real and reactive power signatures were used to train an ANN in [17] on four commonly used appliances. The results showed it was able to disaggregate with a 95% correct detection rate. This, however, was only tested with one load ON or OFF while other loads remain unchanged. Although the classification accuracy and efficiencies were favourable, this algorithm also has trouble tackling loads of higher complexity.

One issue which occurs in ANNs is the 'vanishing/exploding gradients' problem which can occur when you apply some of the popular learning algorithms to retroactively correct the network weights such as back propagation or gradient descent. In the case of "vanishing gradients", whilst the learning algorithm tries to find optimised weights for each layer of the neural network, it is subject to an ever-decreasing value (<1) which proceeds exponentially shrink the values. This effectively halts the learning at the earlier layers as the value becomes too small. The same occurs inversely when the value is increasing exponentially.

C. DNN

ANNs that consist of more than two hidden layers are considered Deep Neural Networks (DNN). Deeper architectures have better representation capabilities which comes at an expense of increased demands of data and computational resources [18]. DNNs have seen considerable growth in NILM in recent years and has already successfully produced state-of-the-art algorithms across multiple domains including automatic speech recognition and image classification. Using multi-layered neural networks, they have the ability to automatically infer hierarchical features from raw input data with large datasets [21]. DNNs have an abstract architecture and can therefore be established in multiple forms and extend to different types of learning algorithms [19].

One interesting application of DNNs in NILM is in [20] where DNNs are adapted to detect the operational state change in household appliances. Using the REDD dataset three households are used to train the model on identifying characteristics such as falling/rising edges and horizontal lines in the time series data. The DNN uses 5 layers with 1200 trainable parameters and only trains on a small fraction of the dataset for each house. The algorithm offers light-weight, low-complexity approach with application across variable load types. In [20] the DNN is compared with benchmark HMM, RNN and Graph Signal Processing-based approaches and was able to achieve a competitive performance even by using a small portion of the available data for training and testing.

Overfitting is one of the issues attached to DNNs, although this is largely dependent on the dataset, the algorithms have a tendency to non-avertedly inference local features in power signatures. This said, DNNs still prove to be one of the most effective algorithms, in one study [22] outperforming HMMs, SVMs and KNNs with a regression model accuracy of 88.71%.

D. RNN

Recurrent Neural Networks (RNNs) are a family of DNNs derived from feed forward neural networks, they are good at modelling sequence data for predictions. They do this by using "sequential memory", a method of memorizing something based on its sequential order. Traditional feed-forward neural networks have a hidden layer, an input layer and an output layer. If one would like to use previous information to effect later information, a loop can be added to pass previous information forward. This is the basis for an RNN, it uses loops to transfer information between steps, the information being the 'hidden state' which is the representation of previous steps. RNNs have been used

Unfortunately like many other neural networks, RNNs suffer from the 'vanishing gradient' problem, therefore as the RNN processes more steps it has more trouble retaining information from its previous steps. The 'short-term memory' and 'vanishing gradient' features stem from the nature of back propagation where the gradients will shrink as they feed back to the earlier layers preventing them from learning. The main disadvantage this causes is an inability for the RNN to inference long range dependencies across time steps [23].

To combat these Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU) networks were introduced. Functioning similar to an RNN, however LSTMs and GRUs both have the ability to learn long-term dependencies by using gates. The gates are tensor operations which can learn what to add or remove from hidden states.

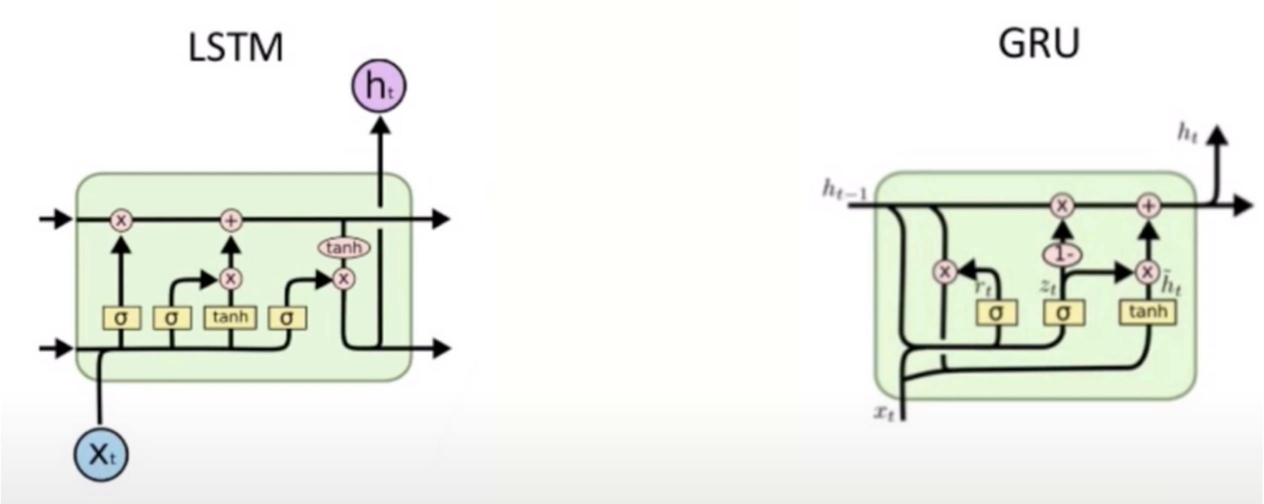

Figure 4 : a) Diagram of a LSTM. b) Diagram of a GRU. (source: https://www.youtube.com/watch?v=LHXXI4-IEns)

The Window GRU mechanism presented in NeuralNilm [23] introduces a sampling window of the aggregate power demand to train the dataset using three DNN architectures. The testing is done on data from a house which is not trained with at all. The results showed that the F1 scores for all three of the implementations surpassed benchmark algorithms such as FHMM.

V. METHOD

As mentioned in section II, the dataset dependency is an issue when considering experimentation of a new algorithm, and it can be considered a research case which is relevant to understand the difference of using one at a time. Therefore, an attempt to build a more robust load classification model is made utilising a benchmark GRU method across two datasets with four loads in each dataset attempting to classify the load in a third dataset which the algorithm has not been trained on.

Using the NILMTK API, the datasets for GREEND REDD and UK-DALE are loaded, and the same four appliances are parsed from each. Each is set to give a 1-month period of 1Hz sampling of data. This means that the GRU window will sample from each dataset at a fixed length. Figure 5 shows a basic illustration of the setup.

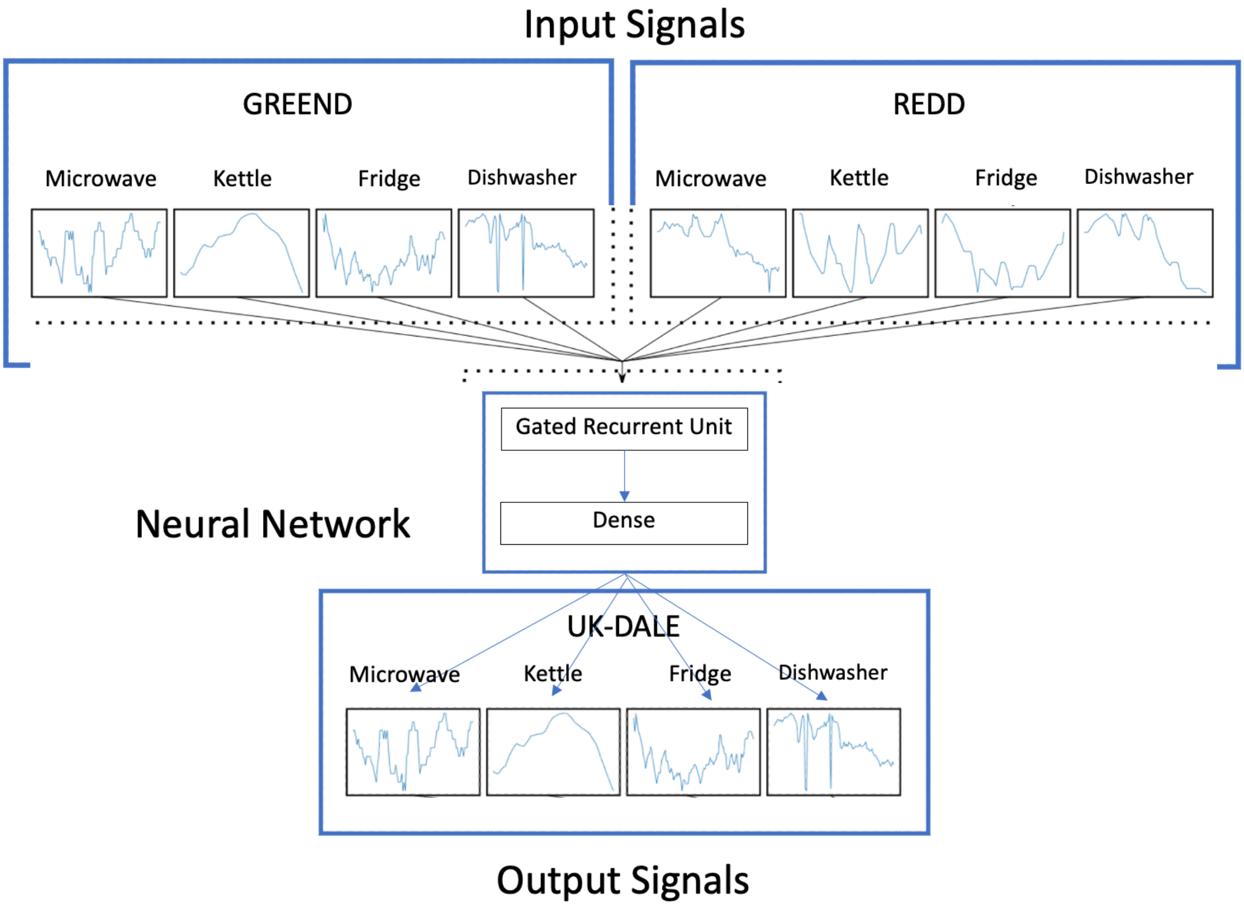

Figure 5 : a) Diagram of data input/output to neuralNILM algorithm used.

The algorithm is based off the neuralNILM architecture uses the RNNDisaggregator using both training datasets. The sampling period is kept at 1Hz and 5 epochs are used to train the algorithm on each appliance power signature within its mains.

VI. RESULTS

The results show the overall performance of the algorithm on each appliance:

APPLIANCE	PRECISION	RECALL	ACCURACY	F1 SCORE
MICROWAVE	0.81	0.032	0.28	0.061
KETTLE	0.49	0.021	0.15	0.04
FRIDGE	-	-	-	-
DISHWASHER	-	-	-	-

TABLE 1 : OVERALL RESULTS ON EACH APPLIANCE FROM THE UK-DALE DATASET.

The algorithm performed best on Microwave and Kettle Appliances and the values for both fridge and dishwasher were extremely small and somewhat inconclusive.

VII. CONCLUSIONS

The GRU method for load disaggregation was tested using three publicly available and widely used datasets. The results show that it is possible to train and test on two different datasets meaning that there is consistency between appliance characteristics across datasets. This report outlines the viability of fusing multiple datasets to incorporate a wider range of signal characteristics and reduce overfitting. Although the results did not show competitive metrics, it was shown that there is some level of classification happening across datasets. The algorithm was still able to classify the less complex state devices such as microwaves and kettles. Continuous loads such as fridges can be difficult to classify across multiple datasets as they are continuous state devices, there isn't very many transient properties in their signatures.

It may be possible that the commonly used datasets are inadvertently overfitting models with specific signature characteristics which are local to a particular dataset for several possible reasons including region, temperature, time of year, etc. With the emergence of more data at high frequency sampling rates, deep learning models are favoured.

NILM will inevitably continue growing in popularity as more smart meters are introduced and access to larger amounts of data mean there is more need for AI-based algorithms for classification. There are many factors effecting the direction this field may take including hardware limitations (sampling frequencies),

VIII. FUTURE WORK

There is room for a lot more work to be done by testing benchmark algorithms across multiple datasets to factor in appliance variability. This is especially true for algorithms capable of high dimensionality and data intensive such as DNNs and GRUs as they can adapt more features into their classifications. A number of benchmark algorithms can be tested in the same way, across datasets to compare their scores with using single datasets.

References

[1] Faustine, Anthony & Pereira, Lucas & Bousbiat, Hafsa & Kulkarni, Shridhar. (2020). UNet-NILM: A Deep Neural Network for Multi-tasks Appliances State Detection and Power Estimation in NILM. 84-88. 10.1145/3427771.3427859.

[2] Joshi, Hitarth & Parikh, Abhishek & Shah, Sandeep. (2021). A Different Neural NILM based Energy Disaggregation.

[3] Zhuang, Mengmeng & Shahidehpour, M. & Li, Zuyi. (2018). An Overview of Non-Intrusive Load Monitoring: Approaches, Business Applications, and Challenges. 4291-4299. 10.1109/POWERCON.2018.8601534.

[4] T. Lu, Z. Xu and B. Huang, "An Event-Based Nonintrusive Load Monitoring Approach: Using the Simplified Viterbi Algorithm," in IEEE Pervasive Computing, vol. 16, no. 4, pp. 54-61, October-December 2017, doi: 10.1109/MPRV.2017.3971125.Klemenjak, Christoph & Kovatsch, Christoph & Herold, Manuel & Elmenreich, Wilfried. (2020). A synthetic energy dataset for non-intrusive load monitoring in households. Scientific Data. 7. 10.1038/s41597-020-0434-6.

[5] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[6] M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

[7] Kukunuri, Rithwik & Chauham, Jainish & Bhagtani, Kratika & Walia, Sumit & Patil, Rohan & Aglawe, Anup & Batra, Nipun. (2020). EdgeNILM: Towards NILM on Edge devices. 10.1145/3408308.3427977.

[8] Zhang, Chaoyun & Zhong, Mingjun & Wang, Zongzuo & Goddard, Nigel & Sutton, Charles. (2016). Sequence-to-point learning with neural networks for nonintrusive load monitoring.

[9] Pukale, D. D., Bhirud, S. G., & Katkar, V. D. (2018). Content-based Image Retrieval using Deep Convolution Neural Network. 2017 International Conference on Computing, Communication, Control and Automation, ICCUBEA 2017, 1–5.

[10] Ghahramani, Zoubin & Jordan, Michael. (1995). Factorial Hidden Markov Models. Machine Learning. 29. 10.1023/A:1007425814087.

[11] Kim, Hyungsul & Marwah, Manish & Arlitt, Martin & Lyon, Geoff & Han, Jiawei. (2011). Unsupervised Disaggregation of Low Frequency Power Measurements. Proc. SIAM Conf. Data Mining. 11. 747-758. 10.1137/1.9781611972818.64.

[12] P. E. Group, “Renewable Energy: What’s the Most Efficient Energy Source?,” [Online]. Available: https://www.phoenixenergygroup.com/blog/renewable-energy-whats-the- most-efficient-energy-source.

[13] Machlev, Ram & Belikov, Juri & Beck, Y. & Levron, Yoash. (2019). MO-NILM: A multi-objective evolutionary algorithm for NILM classification. Energy and Buildings. 199. 10.1016/j.enbuild.2019.06.046.

[14] Huber, Patrick & Calatroni, Alberto & Rumsch, Andreas & Paice, Andrew. (2021). Review on Deep Neural Networks Applied to Low-Frequency NILM. Energies. 14. 10.3390/en14092390.

[15] Kelly, Jack & Knottenbelt, William. (2015). Neural NILM: Deep Neural Networks Applied to Energy Disaggregation. 10.1145/2821650.2821672.

[16] S. Biansoongnern and B. Plangklang, "Nonintrusive load monitoring (NILM) using an Artificial Neural Network in embedded system with low sampling rate," 2016 13th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2016, pp. 1-4, doi: 10.1109/ECTICon.2016.7561398.

[17] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. <i>Deep Learning</i>. The MIT Press.

[18] Bonfigli, Roberto & Felicetti, Andrea & Principi, Emanuele & Fagiani, Marco & Squartini, Stefano & Piazza, Francesco. (2017). Denoising Autoencoders for Non-Intrusive Load Monitoring: Improvements and Comparative Evaluation. Energy and Buildings. 158. 10.1016/j.enbuild.2017.11.054.

[19] Xiao, Peng & Cheng, Samuel. (2019). Neural Network for NILM Based on Operational State Change Classification.

[20] Sremath Tirumala, Sreenivas & Narayanan, Ajit. (2015). Hierarchical Data Classification Using Deep Neural Networks. 9489. 492-500. 10.1007/978-3-319-26532-2_54.

[21] Schirmer, Pascal & Mporas, Iosif. (2019). Statistical and Electrical Features Evaluation for Electrical Appliances Energy Disaggregation. Sustainability. 11. 3222. 10.3390/su11113222.

[22] I. Goodfellow, Y. Bengio, and A. Courville, “Deep learning (adaptive computation and machine learning series), mit press,” 2016.

[23] Kim, Jihyun & Le, Thi-Thu-Huong & Kim, Howon. (2017). Nonintrusive Load Monitoring Based on Advanced Deep Learning and Novel Signature. Computational Intelligence and Neuroscience. 2017. 1-22. 10.1155/2017/4216281.