Sliding Window Approach for Online Energy Disaggregation Using Artificial Neural Networks

Odysseas Krystalakos
Aristotle University of Thessaloniki
School of Informatics
Greece
okrystal@csd.auth.gr

Christoforos Nalmpantis
Aristotle University of Thessaloniki
School of Informatics
Greece
christofn@csd.auth.gr

Dimitris Vrakas
Aristotle University of Thessaloniki
School of Informatics
Greece
dvrakas@csd.auth.gr

ABSTRACT

Energy disaggregation is the process of extracting the power consumptions of multiple appliances from the total consumption signal of a building. Artificial Neural Networks (ANN) have been very popular for this task in the last decade. In this paper we propose two recurrent network architectures that use sliding window for real-time energy disaggregation. We compare this approach to existing techniques using six metrics and find that it scores better for multi-state devices. Finally, we compare ANNs that use Gated Recurrent Unit neurons against those using Long Short-Term Memory neurons and find that they perform equally.

KEYWORDS

energy disaggregation, non-intrusive load monitoring, artificial neural networks

1 INTRODUCTION

The electrical demands of the modern household are constantly changing with the introduction of new types of devices (smart appliances, smartphones, tablets) and new energy sources (solar cells, batteries). This coupled with the fact that the modern buildings are monitored by smart meters, gives the opportunity to retrieve information from a variety of sources. Energy disaggregation allows the user to extract power consumption data, turn on/off events and behavioral patterns of the occupants, using a single point of measurement.

Energy disaggregation is the process of extracting the energy consumption of electrical appliances, from the power demand of a group of appliances, as measured by a single meter. For example, a user that needs to monitor the electricity consumption of each of their appliances can install one meter that measures the aggregate power draw. Then, utilize an energy disaggregation algorithm to approximate the individual consumption signals. Using this technique it is possible to acquire information about the status of electric devices without installing multiple meters. Instead, a single power reading is used, indicating the behavior of the appliances as a whole.

Energy disaggregation was first proposed as Non-Intrusive Appliance Load Monitoring (NALM or NILM) by George Hart in the 1980s [1]. His original NILM algorithm [2] was based on combinatorial optimization techniques. The main idea was to find the optimal states of the monitored appliances so that the sum of power consumption would be the same as the meter reading. However, this approach works only for appliances that have a finite number of states, during which the power consumption remains the same. This means that devices with continuously variable consumption cannot be monitored. Common examples are computers, electric drills and light dimmers.

In a lot of research areas, machine learning techniques have shown their ability to detect complicated patterns. NILM research has turned to machine learning as well. Usually, it is used in a scenario in which an appliance-specific model is trained with existing datasets. This trained model can later be used to monitor consumption in unseen buildings. The advantage of this approach is that it tries to solve the problem in a building-agnostic way and thus creates reusable models. During the past decade, Bayesian models [3], Hidden Markov Models [4,5,6] and Artificial Neural Networks [7,8,9,10,11] have been the most popular techniques used in NILM.

In this paper we present a novel approach to NILM, using artificial neural networks on a sliding window of consumption data. All of the networks are optimized to deliver real-time results. The paper is structured as follows: In section 2 we provide a brief review of proposed neural network techniques on NILM. In section 3 we describe the three architectures used in the experiments and their differences with other proposed networks. Finally, in section 4 we comment on the results and in section 5 we propose ways to improve the performance.

2 ARTIFICIAL NEURAL NETWORKS IN NILM

In the recent years, the most popular approach in NILM was Hidden Markov Models (HMM) and specifically Factorial HMM.

SETN '18, July 9–15, 2018, Rio Patras, Greece

ACM ISBN 978-1-4503-6433-1/18/07…$15.00
https://doi.org/10.1145/3200947.3201011

This type of models is suited for sequential data and appears to perform very well in many supervised NILM scenarios. During the last decade, Neural Networks have managed to outperform their counterparts in several research fields including Computer Vision [12] and Natural Language Processing [13]. Following this trend, from 2013 onward, researchers focused on adapting these techniques for energy disaggregation purposes.

In 2015, Kelly et al., in their paper titled *Neural NILM: Neural Networks Applied to Energy Disaggregation* [8], described three different architectures of ANNs: a Recurrent Neural Network (RNN), a Denoising Autoencoder and a Convolutional Regressor. All three networks were trained, using the UK-DALE [14] dataset, to infer the power draw of a specific appliance, given the aggregate energy consumption. Kelly et al. produced results that outperformed Hart's combinatorial optimization algorithm and the Factorial HMM.

Lange et al. [9] described a different approach with neural networks. Instead of using an end-to-end network that produces disaggregated data, they proposed an architecture to decompose the aggregated signal. Their network is trained to compute the active and reactive power using the current reading. The last layer is restricted to have a binary output. These binary values are considered additive subcomponents of the total power draw. Thus, by using a combinatorial optimization algorithm it is possible to find which of these subcomponents correspond to each appliance. Overall, this technique manages to train the network without any ground truth data and, at the same time, compresses the aggregated signal to a binary representation.

Following these two major studies, the research mainly has focused on recurrent networks based on Neural NILM. Mauch et al. implemented a deep recurrent network using Long-Short Term Memory (LSTM) neurons [10]. This network predicted the consumption of an appliance at a specific time point using the aggregate value of the same time point. Working on the same principle, Thi-Thu-Huong et al. compared the performance of Gated Recurrent Units (GRU) against traditional recurrent networks on energy disaggregation tasks [15]. In their experiments, GRU networks seemed to achieve better results.

In 2017, Zhang et al. [11] in their experiments, achieved state of the art results with a deep convolutional architecture. The main advancement is that the network uses a window of aggregate data to predict the midpoint value of the same window of appliance consumption. The input window has a size of 600 samples which translates to 1 hour of data (6 second sampling period). They named this technique Sequence-to-Point as it uses a sequence of mains power consumption to infer a point of meter consumption. Zhang et al compared the results produced by their seq2point architecture against the works of Kelly et al and found that it performs better by more than 80% on two commonly used error metrics. Finally, they visualized the features learned by the convolutional layers and they found that they describe the behavior of the observed appliance in regards of consumption levels and time of operation.

3 NEURAL NETWORK ARCHITECTURES

In this section we describe the architecture of the proposed neural networks.

During our research we focused on recurrent neural networks. The main reason is that they are proved to perform well with sequential data. This makes them suitable for energy disaggregation as the network receives the timeseries of power consumption as input. Kelly et al. [8] and Mauch et al. [10] worked on the same approach.

Both RNNs that were proposed for NILM by Kelly et al. and Mauch et al., use a single point of the aggregate timeseries to predict the power consumption of the appliance at the same point. In other words, in order to predict the power draw of an appliance at time t, the aggregate draw at t is given as input.

The technique we propose looks at a window of past aggregate data and infers the consumption at a single point. A window of length w means that the neural network will receive the timeslice $[t-w, t]$ to predict the device consumption at time t. An example can be seen in Figure 1.

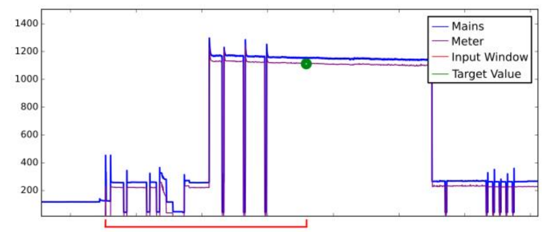

Figure 1: Example parts of the mains and meter signal used as input and target.

All of the following networks were developed using Keras with Tensorflow backend on GPUs. NILMTK [16] was used for the data loading and preprocessing.

We found that the Adam optimization algorithm [17] worked best for training. Finally, the loss was measured using mean squared error.

3.1 LSTM Network

The first attempt to create a recurrent network with a window is based on the LSTM network from Neural NILM [8]. The architecture is kept the same but with an input vector size of w instead of 1. The full architecture details can be seen in Figure 2.

Networks that use LSTM neurons suffer from high computational cost. First of all, each LSTM cell performs several mathematical operations before they produce their output. This makes them computationally demanding for both training and inference. Moreover, they have an internal memory cell which raises the memory demands of the model. Therefore, this kind of network may be unsuitable as NILM systems may have to run on low-cost embedded devices or consumer systems.

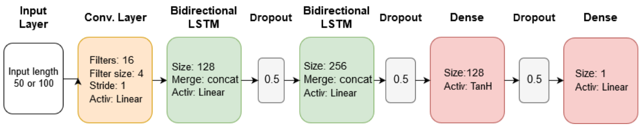

Figure 2: Architecture of the LSTM network with sliding window.

3.2 GRU Network

The limitations of the LSTM network inspired a new design that performs the same whilst being less demanding. The first step was to replace LSTM neurons with GRU. Gated Recurrent Units have a simpler architecture with no internal memory. This makes them more computationally efficient while training and less memory demanding for disaggregation. Inspired from previous works on the comparison of the two architectures [18], we tested both networks to evaluate the difference in prediction accuracy.

The second step was to reduce the number of neurons per layer. As Kelly et al. mention, their networks were not optimized. Through experiments we found that, with recurrent layers of half the size, the accuracy of the network remains the same. This dropped the number of trainable parameters by about 60%.

Finally, we added dropout between the layers to prevent overfitting. This is especially useful when training an appliance model with data from multiple houses. Moreover, dropout is helpful in case some of the input values are missing. The full architecture details can be seen in Figure 3.

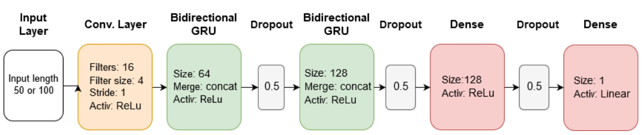

Figure 3: Architecture of the GRU network with sliding window.

3.3 Short Sequence-to-Point Network

Since Sequence-to-Point [11] is using a window of consumption data, we found it useful to compare it with the proposed recurrent networks. However, the original Sequence-to-Point attempt uses an input window of 1 hour to infer a single point. Moreover, half of the window consists of time points after the target consumption time point. This means that this network is unsuitable in scenarios of online disaggregation in which the user expects results with minimal delay. In order to tackle this issue, we shrunk the input window to 10-20 minutes of data. Finally, we added dropout between the convolutional layers and tested the performance. The full architecture details can be seen in Figure 4.

Even though this network has more layers with more neurons, convolutional networks are less computationally intensive than their recurrent counterparts. This mainly happens because neurons in a recurrent layer, receive the output of every previous neuron of its layer as input. This is not the case for CNN as they only use the input vector to compute the result.

During our experiments it was evident that Sequence-to-Point took the same or less time to train and test than the LSTM and GRU networks.

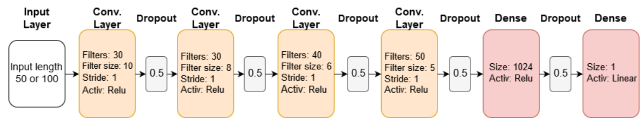

Figure 4: Architecture of the Short Sequence-to-Point network.

4 EXPERIMENT DATA AND RESULTS

We trained all of the models using the UK-DALE dataset [14] with a sampling period of 6 seconds. The buildings were split for training and evaluation as mentioned in Table 1.

Table 1: Buildings used for training and testing

	Training	Testing
Dish Washer	1, 2	5
Fridge	1, 2, 4	5
Kettle	1, 2, 3, 4	5
Microwave	1, 2	5
Washing m.	1, 5	2

This is the same way Kelly et al. split the data. Contrary to their work, no synthetic data were used during training. For the disaggregation part, all of the scenarios were run on unseen buildings from UK-DALE. For the UK-DALE tests, in order to make direct comparison with Neural NILM, we evaluated the models with one week of ground truth data provided by Neural NILM [8].

For all experiments, we normalized the inputs and targets to [0,1] by dividing with a handpicked maximum consumption value. The only exception is Sequence-to-Point for which we subtracted the mean and divided with the standard deviation in the same way as Zhang et al. did. All of the handpicked features are listed in Table 2.

Table 2: Appliance features used for normalization and state detection

	Max Power	On power threshold	Mean Power	Std of Power
Dish Washer	3000	10	700	1000
Fridge	200	50	200	400
Kettle	3000	2000	700	1000
Microwave	3000	200	500	800
Washing m.	2500	20	400	700

We tried to directly compare the three proposed architectures with the works of J. Kelly et al [8]. For this reason, all of the networks were tested on the same data using the same metrics. There are two categories of metrics: Precision, Recall, Accuracy and F1 Score that measure the ability of the network to detect on/off events and Mean Absolute Error and Relative Error in Total Energy that measure the ability to infer the correct power consumption value.

The size of the input vector is different for each algorithm and device. At first, all windows were set to a default length of 50 samples (5 minutes). Then, through experiments we found that some appliances perform better with slight alterations of *w*. The final vector sizes that were used to produce the results are mentioned in Table 3.

Table 3: Sliding window sizes for each device and network (in samples)

	LSTM	GRU	Sequence to Point
Dish Washer	50	50	100
Fridge	50	50	50
Kettle	50	50	100
Microwave	50	50	50
Washing m.	100	100	200

Similarly to Kelly et al. and Zhang et al. we assumed that a device is turned on when its consumption is above a threshold provided by the UK-DALE dataset. All thresholds can be seen in Table 2. The metrics used were:

$$TP = \text{number of true on state predictions}$$
$$FP = \text{number of false on state predictions}$$
$$FN = \text{number of false off state predictions}$$
$$E = \text{total energy consumed}$$
$$E' = \text{total predicted energy consumed}$$
$$y_t = \text{true appliance consumption at time } t$$
$$y_t' = \text{predicted appliance consumption at time } t$$
$$T = \text{length of the predicted timeseries}$$

$$recall = \frac{TP}{TP + FN}$$

$$precision = \frac{TP}{TP + FP}$$

$$f1\ score = 2\frac{precision \cdot recall}{precision + recall}$$

$$accuracy = \frac{TP + TN}{T}$$

$$relative\ error\ in\ total\ energy = \frac{|E' - E|}{\max(E', E)}$$

$$mean\ absolute\ error = \frac{1}{T}\sum|y_t' - y_t|$$

Parts of the resulting signal produced by the networks during the testing phase are displayed in Figure 5. As it is visible in the results of the two networks, some device activations are not detected. Also, some networks learn to output the average consumption at all times. This is especially visible in the plots of the outputs for fridge.

The result produced by our experiments along with the results of the RNN from Neural NILM can be seen in Figure 6.

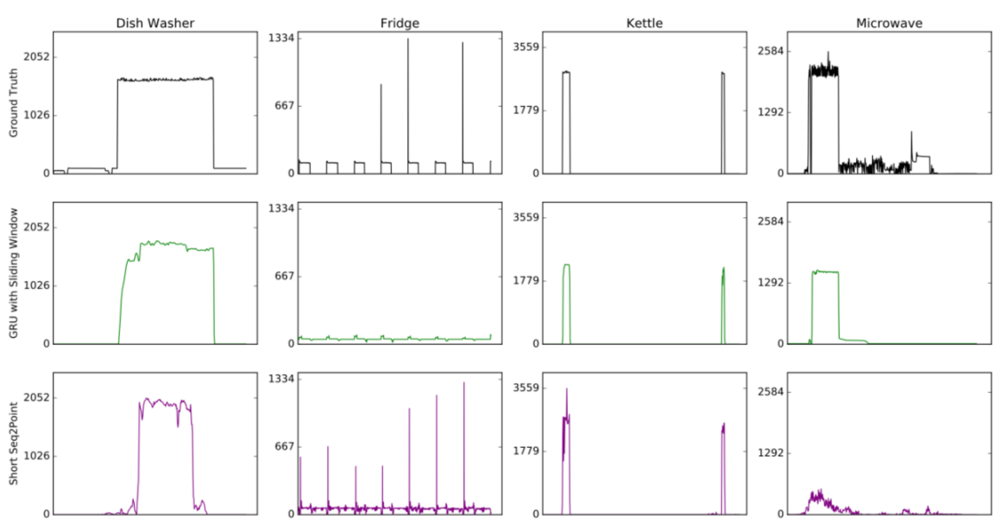

Figure 5: Example outputs of the GRU and the Short Sequence-to-Point networks.

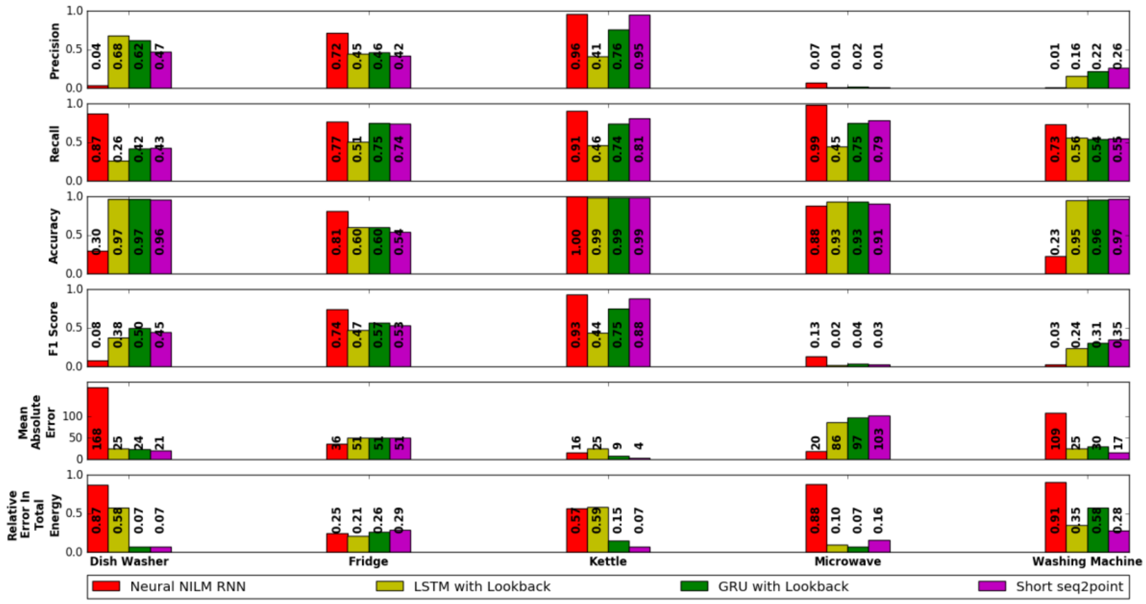

Figure 6: All experiment results.

5 NETWORK COMPARISONS AND CONCLUSIONS

The results, were mixed, with some networks producing superior results on certain devices and certain metrics.

The Neural NILM RNN performs very well on detecting events of two-state appliances like fridge and kettle. However, it fails to predict the power consumption, which becomes obvious from the Relative Error in Total Energy of the kettle. On the contrary, the networks that used sliding window achieved better results on multi-state devices such as the dishwasher and the washing machine. This suggests that using information about the consumption in the previous and/or following time points helps the network to recognize the behavior of the appliance.

The results of the GRU network show that they can at least perform on par to LSTM neurons on all five appliances. Even though they scored almost the same on the washing machine, the results on the rest of the appliances favor the GRU network. Taking into account that LSTM neurons are much harder to train and more memory consuming on deployment, we came to the conclusion that GRU networks are better suited for the task of energy disaggregation.

The short Sequence-to-Point network seems to perform relatively the same with the GRU network. The only exception is the kettle where Sequence-to-Point achieves better scores on all metrics. Judging from the results of our experiments, both networks are well suited for online disaggregation when the user needs to receive results with as short delay as possible.

The appliance that all four of the algorithms failed to disaggregate was the microwave. This mainly lies on the fact that, it is a multi-state appliance, with on-state behavior that varies greatly both on the power consumption (ranging from 1300 to 2800 Watt) as well as on the appliance duration (lasting from 20 seconds to 10 minutes).

6 PROPOSALS FOR FUTURE WORK

It should be mentioned that the three networks are relatively deep architectures with a lot of points that can be optimized.

One point that requires more attention is the length of the sliding window that is used as input. For our experiments we used a default length of 50 samples (5 minutes). Through experiments we found that the networks perform much better on the washing machine when using a window of 100 samples. However this is not true for the rest of the appliances. This indicates that the sliding window size is dependent on the behavior of the device and should be optimized accordingly.

Different types of networks succeed on different appliance types. While the sliding window variants performed better on multi-state devices, the original RNN scored higher on two-state. Extensive tests should be done on different appliances as well as different models of the same appliance as it should help find the pattern that links specific network architectures to device types.

Most NILM attempts based on neural networks pursue to train a network that is able to generalize to any unseen house. In theory, this is possible for appliances that display similar behavior. We found that all of the UK-DALE houses have devices that behave very similarly regarding their on-state consumption. This hints that disaggregating within the dataset is an easier task than a real world scenario. It would be useful to see how the proposed architectures scale using bigger training sets and different test sets.

ACKNOWLEDGEMENTS

The authors would like to thank the Hellenic Artificial Intelligence Society (EETN) for its support for attending this conference.

This work has been funded by the ΕΣΠΑ (2014–2020) Erevno-Dimiourgo-Kainotomo 2018/EPAnEK Program ‘Energy Controlling Voice Enabled Intelligent Smart Home Ecosystem’, General Secretariat for Research and Technology, Ministry of Education, Research and Religious Affairs.

REFERENCES

[1] George W. Hart. 1985. Prototype nonintrusive appliance load monitor.
[2] George W. Hart. 1992. Nonintrusive appliance load monitoring. Proceedings of the IEEE. 80, 12, 1870-1891.
[3] Alan Marchiori, Douglas Hakkarinen, Qi Han. and Lieko Earle. 2011. Circuit-Level Load Monitoring for Household Energy Management. IEEE Pervasive Computing. 10, 1, 40-48.
[4] Dominik Egarter, Venkata P. Bhuvana and Wilfried Elmenreich 2015. PALDi: Online Load Disaggregation via Particle Filtering. IEEE Transactions on Instrumentation and Measurement. 64, 2, 467-477.
[5] Oliver Parson, Siddhartha Ghosh, Mark Weal and Alex Rogers. 2014. An unsupervised training method for non-intrusive appliance load monitoring. Artificial Intelligence. 217, 1-19.
[6] Francesca Paradiso, Federica Paganelli, Dino Giuli and Samuele Capobianco. 2016. Context-Based Energy Disaggregation in Smart Homes. Future Internet. 8, 1, 4.
[7] Francesca Paradiso, Federica Paganelli, Antonio Luchetta, Dino Giuli and Pino Castrogiovanni. 2013. ANN-based appliance recognition from low-frequency energy monitoring data. 2013 IEEE 14th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM).
[8] Jack Kelly and William Knottenbelt. 2015. Neural NILM. Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments - BuildSys '15.
[9] Henning Lange. and Mario Berges. 2016. The Neural Energy Decoder: Energy Disaggregation by Combining Binary Subcomponents. Proceedings of the 3rd International Workshop on Non-Intrusive Load Monitoring.
[10] Lukas Mauch and Bin Yang. 2015. A new approach for supervised power disaggregation by using a deep recurrent LSTM network. 2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP).
[11] Chaoyun Zhang, Mingjun Zhong, Zongzuo Wang, Nigel Goddard and Charles Sutton. 2017. Sequence-to-point learning with neural networks for nonintrusive load monitoring. ArXiv e-prints. https://arxiv.org/abs/1612.09106.
[12] Alex Krizhevsky, Ilya Sutskever and Geoffrey Hinton. 2012. ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems 25, 1097-1105.
[13] Alex Graves. 2013. Generating Sequences With Recurrent Neural Networks. ArXiv e-prints. https://arxiv.org/abs/1308.0850.
[14] Jack Kelly and William Knottenbelt. 2015. The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes. Scientific Data. 2, 150007.
[15] Thi-Thu-Huong Le, Jihyun Kim and Howon Kim. 2016. Classification performance using gated recurrent unit recurrent neural network on energy disaggregation. 2016 International Conference on Machine Learning and Cybernetics (ICMLC).
[16] Nipun Batra, Jack Kelly, Oliver Parson, Haimonti Dutta, William Knottenbelt, Alex Rogers, Amarjeet Singh and Mani Srivastava.2014. NILMTK. Proceedings of the 5th international conference on Future energy systems - e-Energy '14.
[17] Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. . ArXiv e-prints. https://arxiv.org/abs/1612.09106.
[18] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho and Yoshua Bengio. 2014. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. ArXiv e-prints. https://arxiv.org/abs/1412.3555.