



Instituto Federal de Educação, Ciência e Tecnologia da Paraíba
Campus Campina Grande
Coordenação do Curso de Engenharia de Computação

Análise da Utilização de Datasets Sintéticos na Desagregação de Cargas Utilizando NILM

Victor Andrade Perone

Orientador: Igor Barbosa da Costa

Campina Grande - PB, Novembro de 2022
Victor Andrade Perone[®]



Instituto Federal de Educação, Ciência e Tecnologia da Paraíba
Campus Campina Grande
Coordenação do Curso de Engenharia de Computação

Análise da Utilização de Datasets Sintéticos na Desagregação de Cargas Utilizando NILM

Victor Andrade Perone

Monografia apresentada à Coordenação do
Curso de Engenharia de Computação do
IFPB, Campus Campina Grande - PB, como
requisito parcial para conclusão do curso de
Bacharelado em Engenharia de Computação.

Orientador: Igor Barbosa da Costa

Campina Grande - PB, Novembro de 2022

P453a Perone, Victor Andrade.

Análise da utilização de Datasets sintéticos na desagregação de cargas utilizando NILM / Victor Andrade Perone. - Campina Grande, 2022.

45 f. : il.

Trabalho de Conclusão de Curso (Curso de Bacharelado em Engenharia de Computação) - Instituto Federal da Paraíba, 2022.

Orientador: Prof. Dr. Igor Barbosa da Costa.

1.Desagregação de cargas 2. Dados sintéticos 3. Datasets I. Costa, Igor Barbosa II. Título.

CDU 004

Análise da Utilização de Datasets Sintéticos na Desagregação de Cargas Utilizando NILM

Victor Andrade Perone

Prof Dr. Igor Barbosa da Costa
Orientador

Prof. Ms. Victor André Pinho de Oliveira
Membro da Banca

Prof. Dr. Emanuel Dantas Filho
Membro da Banca

Campina Grande, Paraíba, Brasil
Novembro de 2022

Dedico este trabalho a minha família, esposa e amigos que sempre me apoiaram e nunca me deixaram desistir.

Se eu vi mais longe, foi por estar sobre ombros de gigantes.

Isaac Newton

Agradecimentos

Agradeço a todos que me apoiaram nesta longa jornada.

Resumo

Identificação de aparelhos e seu consumo é um problema que vem sendo estudado desde a década de 1990 quando foi proposta a técnica de monitoramento não intrusivo de cargas (*NILM*). Em 2014 o *NILMTK* foi apresentado como uma ferramenta e é considerada referência para desagregação de cargas. A rede neural proposta utilizada, proposta pelo *WaveNILM* consiste em uma rede neural causal dilatada, aplicada em conjunto com a ferramenta *NILMTK* para avaliar o impacto da adição de datasets sintéticos ao treinamento de redes neurais para desagregação de cargas em cenários reais. Para tais experimentos foram utilizados os conjuntos de dados sintéticos *SmartSim* e *SynD*, o resultado final foi uma acurácia na desagregação de refrigerador em cenários reais de 65,6%.

Palavras-chave: dados sintéticos; desagregação de cargas; *NILMTK*, *WaveNILM*.

Abstract

Appliance identification and their consumption is a problem that has been studied since the 1990s, when the non intrusive load monitoring (NILM) technique was proposed. In 2014 NILMTK was presented as a tool and is insidered a reference for load disaggregation. The proposed neural network used, proposed by WaveNILM consists of a dilated causal neural network, applied together with the NILMTK tool to evaluate the impact of adding synthetic datasets to the training of neural networks for load disaggregation in real scenarios. For such experiments, the synthetic datasets SmartSim and SynD were used, and 65,6% was the final accuracy in disaggregation of refrigerators in real scenarios.

Keywords: synthetic data; load disaggregation; NILMTK, WaveNILM.

Sumário

Lista de Figuras	xii
Lista de Tabelas	xiii
1 Introdução	1
1.1 Objetivos do Trabalho	3
1.2 Metodologia	3
1.3 Contribuições	4
1.4 Organização do Documento	4
2 Fundamentação Teórica	5
2.1 Tecnologias utilizadas	5
2.2 <i>Datasets</i>	5
2.3 WaveNILM	6
2.4 NILMTK	7
2.5 Métrica de Avaliação	10
2.6 Considerações Finais	10
3 Revisão Bibliográfica	11
3.1 Trabalhos Relacionados	11
3.2 SmartSim	12
3.3 SynD	13
3.4 Considerações Finais	16
4 Desenvolvimento	17
4.1 Escolha dos Datasets	17
4.2 Análise Exploratória	18
4.2.1 Design de Experimento	18
4.3 Resultados e Discussões	21
4.3.1 Simulações dados sintéticos individuais	21
4.3.2 Simulações dados reais e sintéticos	22
4.3.3 Todos os dados disponíveis	25
4.4 Considerações Finais	27

5 Conclusão	29
5.1 Sugestões para Trabalhos Futuros	29
Referências Bibliográficas	31

Lista de Figuras

2.1	Comparativo de uma Rede Neural Causal padrão com uma Rede Neural Causal Dilatada. Fonte: Adaptado de [Oord <i>et al.</i> 2016]	7
2.2	NILMTK pipeline. Em cada estágio da arquitetura de execução, resultados e dados são armazenados ou carregados de uma memória. Fonte: adaptado de [Batra <i>et al.</i> 2014].	8
4.1	Comparativo Refrigeradores	18
4.2	Resultados do treinamento individual refrigerador <i>SmartSim</i>	21
4.3	Resultados do treinamento individual refrigerador <i>SynD</i>	22
4.4	Resultados do treinamento refrigerador <i>SmartSim</i> e dados reais 3 meses de dados	22
4.5	Resultados do treinamento refrigerador <i>SmartSim</i> e dados reais 6 meses de dados	23
4.6	Resultados do treinamento refrigerador <i>SmartSim</i> e dados reais 1 ano de dados	23
4.7	Resultados do treinamento refrigerador <i>SynD</i> e dados reais 3 meses de dados	24
4.8	Resultados do treinamento refrigerador <i>SynD</i> e dados reais 6 meses de dados	25
4.9	Resultados do treinamento refrigerador <i>SynD</i> e dados reais 1 ano de dados	25
4.10	Resultados do treinamento refrigerador ambos <i>datasets</i> e dados reais 3 meses de dados	26
4.11	Resultados do treinamento refrigerador ambos <i>datasets</i> e dados reais 6 meses de dados	26
4.12	Resultados do treinamento refrigerador ambos <i>datasets</i> e dados reais 1 ano de dados	26

Lista de Tabelas

- 4.1 Tabela acurácia de avaliação *SmartSim* 24
- 4.2 Tabela acurácia de avaliação *SynD* 25
- 4.3 Tabela acurácia de avaliação com todos os dados disponíveis 27
- 4.4 Tabela resumo das avaliações das redes neurais 28

Capítulo 1

Introdução

Quando a maioria das pessoas escutam falar sobre "Inteligência Artificial", logo visualizam a imagem de um robô: um mordomo dependente ou o Exterminador do Futuro [Géron 2019]. Entretanto, a Inteligência Artificial (IA) não é apenas uma fantasia futurística [Géron 2019], ela já faz parte do dia a dia da sociedade há décadas em algumas aplicações específicas, tais como, reconhecimento de caracteres (OCR – *Optical Character Recognition*), *chatbots* que utilizam a linguagem para conversar com pessoas de maneira natural e os assistentes pessoais que podem auxiliar nas atividades do cotidiano como definir alarmes e marcar reuniões.

Uma das subáreas da Inteligência artificial é a aprendizagem de máquina (AM). A área da AM faz uso de conjuntos de dados para que máquinas possam aprender padrões a partir desses dados. Por exemplo, é possível construir e treinar redes neurais artificiais voltadas para [Géron 2019; Tran *et al.* 2021]: (1) análise de imagens de produtos em uma linha de produção para classificação automática; (2) detecção de câncer através de imagens de tomografia; (3) comando de voz; (4) detecção de fraude em cartão de crédito; (5) recomendação de produtos baseado no histórico de compras do clientes e (6) plataforma de Internet das Coisas (IoT – *Internet of Things*), entre outros.

Dentre essas várias aplicações possíveis, a busca pela melhoria da eficiência energética tem recebido bastante atenção recentemente. A demanda por energia elétrica cresceu cerca de 4% em 2018 [Bousbiat *et al.* 2020] – considerado o maior crescimento naquela década (IEA 2018)¹ – por esse motivo tem-se almejado uma redução do consumo energético, bem como, os benefícios econômicos e ecológicos.

A Aprendizagem de máquina a partir de conjuntos de dados de séries temporais tem facilitado a vida das pessoas, seja ajudando a prever problemas cardíacos com a utilização de *smartwatches*, seja para a previsão do tempo. Neste caso, os dados a cerca do consumo de energia elétrica tem se tornado cada vez mais disponível graças às novas infraestruturas de *smart meters*, e a análise destes dados ganhou interesse significativo nos últimos anos por conta do aumento do consumo.

O monitoramento de cargas pode ser feito de duas formas, diretamente na carga de

¹<https://www.iea.org/geco/>

forma intrusiva realizando a medição individual de cada carga, ou de forma agregada quando várias cargas são medidas em um único ponto de medição. Desagregação destas cargas se faz importante quando há uma medição de várias cargas em um único ponto, sendo possível aplicar técnicas de separação de cargas para análise de consumo e de possíveis problemas.

O monitoramento intrusivo de cargas (ILM - *Intrusive Load Monitoring*) tem a aplicação como monitoramento local de energia, sendo que um aparelho ou setor da residência é monitorado por esta técnica, não a residência como um todo. A vantagem desta técnica de acordo com [Ridi, Gisler e Hennebert 2014] é o fornecimento de informações mais detalhadas sobre o comportamento e consumo individual de cada aparelho monitorado, e por monitorarem cargas menores, geralmente os sensores apresentam um custo reduzido. Por monitorarem aparelhos individualmente, para que seja monitorada uma residência completa o custo se torna mais elevado devido à quantidade de medidores necessários do que com aplicações com apenas um medidor geral, como as aplicações *NILM* propõem.

Informações detalhadas sobre utilização de energia elétrica permitem o emprego de técnicas para reduzir e otimizar a utilização de energia, previsão de demanda e controle de consumo. Para ter acesso a essas informações, coleta-se os dados agregados no ponto central de medição. Posteriormente com a aplicação de técnicas de aprendizagem de máquina serão desagregados, havendo um detalhamento das contribuições individuais de cargas através da técnica conhecida como NILM (Non Intrusive Load Monitoring ou monitoramento não intrusivo de cargas).

O monitoramento não intrusivo de cargas NILM (Non Intrusive Load Monitoring) é o processo de estimar o consumo de energia elétrica de cada aparelho individualmente a partir de medições realizadas em um ponto de entrada geral de energia. A aplicação inicial consiste em informar ao consumidor quanto cada aparelho consome individualmente. A partir dessa informação, o consumidor pode identificar potenciais economias sobre mudança de hábitos de consumo ou até mesmo aparelhos que estejam apresentando consumo superior ao esperado.

O problema foi formalizado nos anos 1990 por George Hart [Hart 1992] que propôs a utilização de aprendizagem de máquina. Em 2015, Jack Kelly [Kelly e Knottenbelt 2015] aperfeiçoou fazendo uso de técnicas de *deep learning*, alcançando um enorme avanço na área a partir de diferentes arquiteturas de redes neurais para séries temporais.

Com o aumento da demanda por energia elétrica, há uma maior preocupação com o uso consciente, bem como com a gestão do uso. Dessa forma, apresentar o consumo detalhado e possíveis pontos de economia podem ser muito importantes tanto para consumidores quanto para empresa de distribuição de energia. Atualmente, residências e comércios utilizam 40% da energia total no mundo, podendo chegar a 50% até 2030 [Zoha *et al.* 2012].

Atualmente no País ainda há dificuldade na realização da identificação do consumo de energia elétrica com a separação de cargas. Conforme observado na literatura, por não se tratar de uma solução trivial, formas que utilizam programação convencional não obtiveram resultados satisfatórios. Dessa forma, a solução proposta utiliza-se de redes neurais artificiais profundas para propor solucionar tal problema, pois esta abordagem que leva a uma melhor

acurácia sem necessidade de alto investimento em infraestrutura.

A obtenção de um conjunto de dados robusto composto somente de dados reais é muito dispendioso, pois há a necessidade de vários medidores para que muitas residências sejam coletadas simultaneamente e por longos períodos de tempo. Por este motivo o trabalho se propõe a analisar dois conjuntos de dados sintéticos, o *SmartSim* e o *SynD* juntamente com dados reais coletados, pois para que haja uma generalização nos dados a fim de obter-se uma desagregação de energia satisfatória é necessária uma grande quantidade de dados. Estes são utilizados no treinamento da rede neural convolucional causal dilatada *WaveNILM*, proposta para a desagregação de dados de energia elétrica pois é capaz de analisar grandes períodos de tempo mantendo bom desempenho. Tal treinamento tem a finalidade de avaliar qual impacto da utilização de conjuntos de dados sintéticos juntamente com os dados reais na desagregação de refrigeradores, pois este é o aparelho em comum entre os três *datasets*, tanto os simulados quanto o real, que possui impacto significativo na conta de energia elétrica [Batra *et al.* 2014].

1.1 Objetivos do Trabalho

Diante do cenário apresentado nas seção anterior, destaca-se como objetivo principal deste trabalho:

- Avaliar o uso de dados sintéticos para treinamento do *WaveNILM* com a finalidade de desagregar cargas elétricas referentes a refrigeradores domésticos.

Para alcançar os objetivos gerais deste trabalho, foram definidos os seguintes objetivos específicos:

- Gerar sinteticamente dados confiáveis de refrigeradores;
- Treinar o *WaveNILM* a partir de diferentes conjuntos de dados;
- Avaliar o desempenho do *WaveNILM* quando alimentado por diferentes conjuntos de dados sintéticos e reais;

1.2 Metodologia

Realizou-se revisão bibliográfica para verificar o estado da arte em relação a técnicas para desagregação de dados de diversos aparelhos eletrodomésticos a partir da medição da energia elétrica. Por se tratar de um problema complexo e ter disponibilidade limitada de dados reais, optou-se pela utilização de *datasets* sintéticos públicos para aumentar a quantidade de dados disponível.

Os *datasets* públicos utilizados foram *Synd* [C. Kovatsch C. 2020] e *Smartsim* [Chen, Irwin e Shenoy 2016]. Para realizar a desagregação de cargas, utilizou-se uma rede neural

convolucional causal dilatada *WaveNILM* [Harell, Makonin e Bajić 2019] em conjunto com a ferramenta *NILMTK* [Batra *et al.* 2014], treinada com dados sintéticos e reais, realizou-se a avaliação do impacto causado pela utilização dos conjuntos de dados sintéticos para o aprendizado da rede neural.

1.3 Contribuições

As contribuições deste trabalho são sumarizados nos seguintes pontos:

- Análise da semelhança entre os dados de aparelhos elétricos simulados e reais;
- Aumento na quantidade de dados disponível;
- Validar em cenários reais eficácia de dados sintéticos para futuros treinamentos a um baixo custo.

1.4 Organização do Documento

Este trabalho apresentará no Capítulo 2 os conhecimentos básicos necessários para compreender a rede neural utilizada baseada no *WaveNILM* e a ferramenta *NILMTK*. O atual estado da arte é apresentado no Capítulo 3, permitindo uma melhor compreensão da contribuição deste trabalho. Na sequência, o Capítulo 4 contemplará uma descrição detalhada da proposta dessa pesquisa, bem como uma análise de desempenho baseada nos resultados obtidos. Por fim, no Capítulo 5 são descritas as conclusões gerais dos estudos desenvolvidos e são sugeridos os trabalhos futuros que podem ser realizados no tema de estudo.

Capítulo 2

Fundamentação Teórica

Neste Capítulo apresenta-se uma descrição das tecnologias utilizadas, assim como uma explicação sobre as mesmas.

2.1 Tecnologias utilizadas

A partir dos requisitos levantados e da análise de tecnologias foram definidas as tecnologias a serem utilizadas. Linguagem de programação *python* [Rossum e Jr 1995], que surgiu no ano de 1991 desenvolvida pelo matemático Guido Von Rossum. É uma linguagem interpretada, de alto nível e fortemente tipada. Escolhida por sua forte integração com as demais ferramentas utilizadas, além de ser a mesma linguagem utilizada na ferramenta *NILMTK* [Batra *et al.* 2014]. A biblioteca *Numpy* [Harris *et al.* 2020] que consiste em uma ferramenta matemática e estatística que dá suporte a extração e conversão dos dados para o formato desejado.

A ferramenta utilizada para a construção da rede neural causal foi o *Tensorflow* [Abadi *et al.* 2015] que é um *framework* para aprendizado de máquina mais utilizado na indústria, criado pela empresa Google, mais especificamente a divisão Google Brain em 2015, lançado ao público sob a licença de código aberto.

2.2 *Datasets*

Os *datasets* podem ser classificados em 2 categorias de acordo com as suas características, estas características utilizadas para a classificação são a taxa de amostragem e tipo de dados.

Quanto a taxa de amostragem, podem ser de alta frequência como o *Cooll* [Picon *et al.* 2016] categoria esta atribuída a *datasets* que possuem taxa de amostragem superior a 1KHz, ou baixa frequência como o [Chen, Irwin e Shenoy 2016] que possui taxa de amostragem de 1Hz. Os *datasets* de alta frequência possuem a vantagem de capturar mais características a cerca dos aparelhos monitorados, no entanto pelo grande volume de dados que são necessários armazenar geralmente são coletados curtos períodos.

Quanto ao tipo de dados podem ser naturais ou sintéticos. Os sintéticos são gerados a partir de equações que parametrizam o funcionamento de aparelhos ou pela geração de dados semelhantes aos observados em coletas reais. Tem a vantagem de serem menos custosos, uma vez que não há a necessidade de instalação de medidores para coleta, no entanto ao realizar aproximações matemáticas um erro é inserido na geração de sinais do aparelho, o que pode resultar em uma generalização enviesada. Para os conjuntos de dados naturais não há a necessidade de verificação, uma vez que são formados a partir de dados reais.

2.3 WaveNILM

A rede neural utilizada foi proposta por [Oord *et al.* 2016] desenvolvida inicialmente para processamento de grande volume de dados de áudio, posteriormente utilizada para o processamento de dados de energia elétrica através do *WaveNILM* presente em [Harell, Makonin e Bajić 2019] uma vez que consiste de uma rede neural convolucional causal dilatada (*Dilated Causal Convolutional Neural Network*) adequada para tal finalidade pois este tipo de rede neural não permite que o modelo altere a ordem de entrada dos dados, além de relacioná-los temporalmente. Esta torna-se uma rede interessante para aplicação, pois consiste em uma técnica que utiliza apenas dados passados para desagregar o presente, enquanto que técnicas como redes neurais bidirecionais e convolucionais tradicionais precisam de amostras futuras para tal desagregação, gerando assim um atraso em sistemas de tempo real. Sendo de vital importância para tal tipo de dado, pois caso o consumo esteja elevado em um horário que as tarifas sejam mais elevadas, o usuário deverá ser avisado imediatamente.

Este tipo de rede se diferencia das redes convolucionais tradicionais por utilizar apenas dados das amostras anteriores para calcular a saída atual. Redes neurais causais dilatadas conseguem uma entrada significativamente maior que as tradicionais mantendo a mesma taxa de amostragem e mantendo a causalidade. No caso da utilização de várias camadas na rede neural sofrem menos com problemas de *vanishing/exploiting gradient descent* em comparação com redes neurais recorrentes, entretanto caso o fator de dilatação seja muito alto a rede por tornar-se muito esparsa, perdendo informações importantes. Para resolver tal problema, recomenda-se a utilização de fatores de dilatação diferentes para cada camada da rede neural.

Em uma rede neural causal com entrada $x[n]$, fator de dilatação M , comprimento N e c_k parâmetros tem a sua saída:

$$y[n] = \sum_{k=0}^{N-1} c_k \cdot x[n - M \cdot k] \quad (2.1)$$

note que a entrada de y é de tamanho $M \cdot (N - 1) + 1$ mesmo tendo apenas N parâmetros.

A figura 2.1 demonstra uma rede neural convolucional dilatada, com fator de dilatação 1, 2, 4, 8 em suas respectivas camadas em comparação com uma rede neural causal convo-

lucional padrão. O que significa que a cada nova camada da rede neural dilatada há uma redução na metade da quantidade de neurônios utilizados em relação à camada anterior.

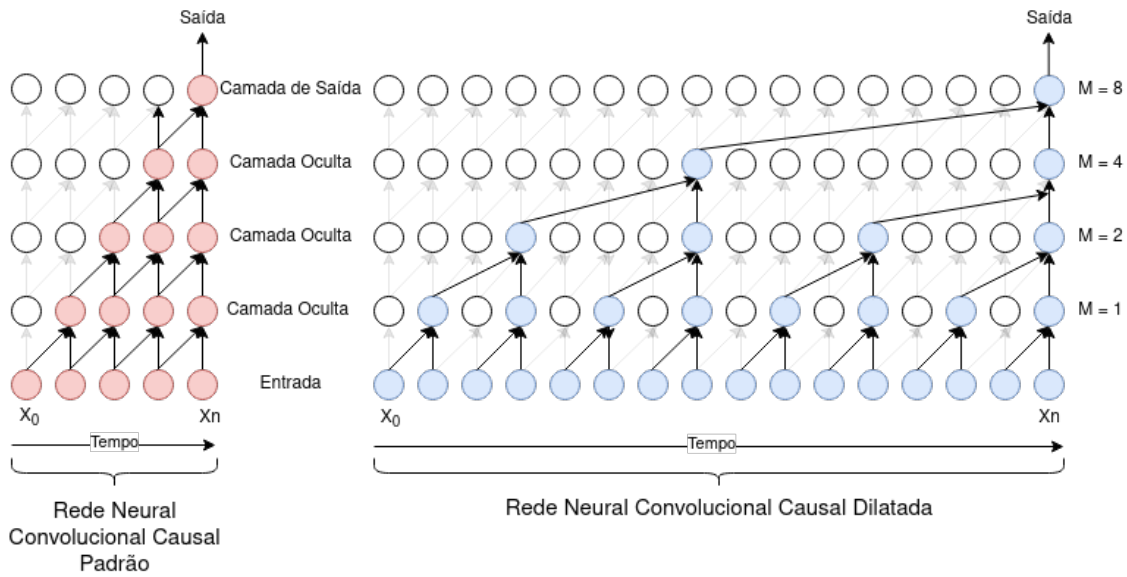


Figura 2.1: Comparativo de uma Rede Neural Causal padrão com uma Rede Neural Causal Dilatada. Fonte: Adaptado de [Oord et al. 2016]

Calcula-se o campo receptivo R com l camadas e *kernels* convolucionais de tamanho k como:

$$R = 2^l(k - 1) \quad (2.2)$$

com isto, a rede neural consegue ter na sua entrada $R - 1$ amostras do passado.

Adicionando mais camadas de redes convolucionais dilatadas com fator de dilatação maior aumenta-se exponencialmente o campo receptivo, mantendo a eficiência computacional [Yu e Koltun 2015]. Uma vez que a rede neural não possui *back propagation through time* possui algumas vantagens:

- Pode ser treinada em paralelo aproveitando a eficiência de GPUs para redes neurais;
- Não têm problemas de *vanishing/exploiting gradient descent*;
- Capaz de aprender padrões de passado distante sem problemas.

2.4 NILMTK

A ferramenta utilizada NILMTK [Batra et al. 2014] é o *toolkit* referência para utilização de redes neurais artificiais na análise de energia elétrica. Tanto por sua versatilidade, uma vez que foi construída de forma modular permitindo a modificação e adição de funcionalidades, quanto por ter sido construída especificamente para a desagregação de cargas. Teve como objetivo definir um padrão de utilização de vários *datasets*, bem como por possuir métricas

de avaliação e acurácia dos dados utilizados. Na figura 2.2 nota-se o fluxo de dados utilizado pela ferramenta.

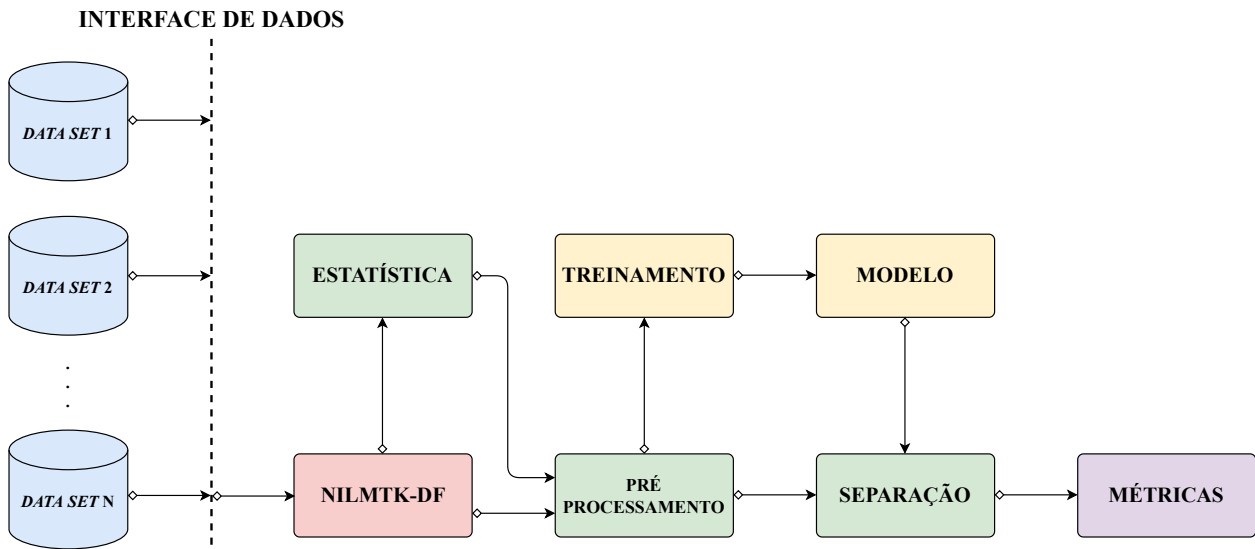


Figura 2.2: NILMTK pipeline. Em cada estágio da arquitetura de execução, resultados e dados são armazenados ou carregados de uma memória. Fonte: adaptado de [Batra et al. 2014].

A primeira etapa necessária para a utilização é a compatibilidade dos dados com o formato suportado pela ferramenta. Os formatos suportados pela mesma são *.csv*, um formato popular para armazenar dados estruturados e de fácil leitura, e *HDF5* que é um formato hierárquico binário que possui uma maior eficiência no armazenamento e utilização otimizada, uma vez que possibilita não só armazenar os dados que se precisa, bem como metadados e informações adicionais que podem ser relevantes para uma melhor previsão de consumo [Kelly e Knottenbelt 2014].

Com a etapa de importação do *dataset* concluída, possíveis problemas nos *datasets* são levantados pelo módulo de estatística, as seguintes análises são realizadas:

- **Detecção de lacunas:** Uma lacuna nos dados é detectado se o tempo entre duas amostras é maior que um limiar previamente definido;
- **Taxa de *dropout*:** O número total de amostras dividido pelo número de amostras esperadas;
- **Taxa de *dropout* (ignorando lacunas):** Ignora as perdas de dados provenientes do sensor ignorando os períodos em que esteve desligado, remove-se os períodos em que o sensor esteve desligado e calcula a taxa de *dropout* para o restante do período;
- ***Up-time*:** O tempo total que o sensor esteve medindo. Consiste no último *timestamp* menos o primeiro *timestamp* menos a duração de qualquer lacuna.

Após a realização da identificação de possíveis problemas no *dataset*, começa o pré-processamento do mesmo. Uma etapa de *downsampling* para que todos os *datasets* tenham

a mesma taxa de amostragem, uma vez que os *datasets* utilizados podem ter taxas de amostragem diferentes. Uma normalização na tensão de entrada pode ser feita a fim de diminuir o impacto na desagregação, uma vez que flutuação na tensão não tem relação com a característica do aparelho. Esta normalização ocorre segundo a seguinte equação:

$$Potencia_{normalizada} = \left(\frac{Tensao_{nominal}}{Tensao_{observada}} \right)^2 \times Potencia_{observada} \quad (2.3)$$

outras funções podem ser utilizadas nesta etapa:

- Interpolação de pequenos períodos de lacunas nos dados;
- Filtrar dados inconsistentes. Como medição de uma potência superior à que o sensor é capaz;
- Filtrar dados dos sensores secundários quando há dados faltando no medidor principal.

Para a etapa de treinamento a ferramenta duas opções, otimização combinatória (*Combinatorial Optimization - CO*) e Modelo Fatorial Oculto de Markov (*Factorial Hidden Markov Model - FHMM*). Esta etapa é opcional para a utilização da ferramenta, pois caso haja uma rede previamente treinada pode-se utilizar apenas o teste realizando-se a separação de cargas. No entanto para o presente trabalho esta consiste em uma etapa fundamental, uma vez que o objetivo consiste em avaliar o mesmo modelo sendo treinado em diferentes cenários. Após o treinamento, um modelo é gerado e testado na separação de cargas.

A ferramenta possui uma série de métricas que combinam métricas gerais e algumas específicas para separação de cargas. Tais métricas são:

- **Erro na energia total designada:** Consiste na diferença entre a energia total consumida pelo *n-ésimo* aparelho analisado e a energia real consumida.
- **Fração total da energia atribuída corretamente:** A sobreposição entre a fração da energia atribuída a cada *appliance* e a energia realmente consumida ao longo do *dataset*.
- **Erro normalizado na energia atribuída:** a soma das diferenças entre a potência atribuída e a potência real do aparelho *n* em cada intervalo de tempo *t* normalizado pelo consumo total da aplicação.
- **Erro RMS na potência atribuída:** O erro quadrático entre a potência atribuída e a potência real da *appliance n* em cada intervalo de tempo *t*
- **Matriz de confusão:** Número de intervalos de tempo nos quais o estado de cada aparelho foi confundido com outro estado ou classificado corretamente. *True Positive* (TP): intervalos de tempo nos quais o aparelho *n* foi corretamente classificado como ligado. *False Positive* (FP): Classificado com ligado quando estava desligado *False*

Negative (FN): Classificado como desligado quando estava ligado *True Negative* (TN): Classificado corretamente como desligado.

- **Precisão:** Intervalos de tempo nos quais o aparelho foi identificado como ligado e estava desligado.
- **Recall:** Intervalos de tempo nos quais o aparelho foi identificado como ligado e estava realmente ligado.
- **F-score:** Média harmônica entre precisão e *recall*.
- **Hamming Loss:** Total de informação perdida quando *appliances* são classificadas incorretamente no *dataset*.

2.5 Métrica de Avaliação

A métrica de avaliação utilizada não está presente na ferramenta *NILMTK*, mas foi sugerida pelo *WaveNILM*. A acurácia baseada na estimativa de potência precisa mostrar quão precisa é a estimativa da potência é consumida em relação ao consumo real. Por se tratar de um sistema que estima quanta potência está sendo consumida comparada ao consumo real, a utilização de métricas tradicionais como F1-Score e erro médio quadrático (*RMSE*) tornam difícil a comparação da desagregação de um aparelho com outro, pois tais métricas não são normalizadas. Portanto, como métrica de avaliação da rede neural utilizou-se a acurácia estimada de acordo com o [Harell, Makonin e Bajić 2019], pois por se tratar de uma rede neural de regressão, o erro é focado na potência desagregada ao invés do estado dos aparelhos. Tal métrica é definida por [Makonin e Popowich 2015] comum para avaliação de energia desagregada

$$AcuraciaEstimada = 1 - \frac{\sum_{t=1}^T \sum_{k=1}^K |\hat{s}_k(t) - s_k(t)|}{2 \sum_{t=1}^T \sum_{k=1}^K s_k(t)} \quad (2.4)$$

onde $\hat{s}_k(t)$ é a potência prevista e $s_k(t)$ é a potência real para o aparelho k em um instante t . O tempo total T e K é o número de *appliances* respectivamente. A equação leva a acurácia total estimada, como a rede neural é treinada especificamente para cada aparelho individualmente o somatório em relação à K pode ser removido, e como resultado tem-se a acurácia estimada por aparelho.

2.6 Considerações Finais

Neste capítulo as bases para o entendimento do trabalho foram explicadas, a classificação dos *datasets*, as redes neurais convolucionais causais dilatadas e a ferramenta *NILMTK*, bem como a aplicação vantagens e justificativas para a escolha de tais ferramentas.

Capítulo 3

Revisão Bibliográfica

Por se tratar de um problema complexo, há poucos conjuntos de dados sintéticos, foram encontrados 4 conjuntos de dados *SynD* presente em [C. Kovatsch C. 2020], o *SmartSim* presente em [Chen, Irwin e Shenoy 2016], o *AMBAL* [Buneeva e Reinhardt 2017] que não possui dados abertos para utilização e o *SHED* [Henriet *et al.* 2017] que trata-se de um conjunto de dados focados em ambientes comerciais, não sendo o foco desta pesquisa. Portanto os *datasets* que foram analisados em conjunto com os dados reais foram o *SynD* e *SmartSim*.

Como base tem-se a ferramenta *NILMTK* apresentada em et al. [Batra *et al.* 2014] utilizada para o estudo de separação de cargas demonstrada no capítulo 2.

3.1 Trabalhos Relacionados

Há poucos trabalhos que utilizam os conjuntos de dados selecionados para a desagregação de cargas, um destes é o [Bousbiat, Klemenjak e Elmenreich 2020], que utiliza o *dataset SynD* como um dos conjuntos de dados para validar a técnica de imagens de séries temporais para a desagregação e realização de *Data Augmentation* de cargas. Tal técnica baseia-se na conversão de de uma sequência de entrada para uma imagem, que serve como entrada para uma versão modificada de um *Autoencoder* eliminador de ruído utilizado para a desagregação de cargas. Principal contribuição do artigo é a comparação entre técnicas de análise de imagens Utilizando-se de um intervalo de 105 dias, onde 90 dias foram utilizados para treino e 15 dias para avaliação do algoritmo proposto. Técnica não recomendada, uma vez que o conjunto de treinamento é o mesmo utilizado para validação.

O trabalho de [Thorve *et al.*] propõe métricas para a validação de *datasets* residenciais sintéticos, propondo definições melhoradas de precisão, *recall* e cobertura na aplicação em larga escala para séries temporais, a fim de descrever a fidelidade de um conjunto de dados sintético em comparação com dados reais bem com a diversidade dos dados gerados. Com a utilização de técnicas de aprendizado não supervisionado.

O trabalho de [Harell 2020] propõe uma revisão comparando diversos *datasets* simulados entre si, e o proposto no trabalho que consiste em várias redes neurais adversariais para gerar

padrões de consumo. No entanto o comparativo realizado não é para desagregação de cargas, apenas a comparação de similaridade com dados reais e os dados sintéticos utilizados. Esta técnica proposta é focada em geração de sinais por aparelhos, não focado em geração de uma residência inteira a fim de desagregação de cargas.

O trabalho de [Luca *et al.* 2022] propõe uma análise da grande maioria dos *datasets* disponíveis simulados e reais. O trabalho visa superar as limitações existentes nos diversos conjuntos de dados, seja por possuírem poucos parâmetros elétricos, seja por períodos muito curtos de dados ou por hábitos de consumo muito padronizados. Para resolver estes problemas propões uma simulação baseada nos conjuntos de dados existentes, extraindo mais de 400 parâmetros elétricos discriminativos em 36 eletrodomésticos diferentes, discutindo configurações de aquisição, processo de simulação, parâmetros elétricos extraídos e exemplos de aplicabilidade. Para assim fornecer um índice de qualidade dos dados, um procedimento de validação foi utilizado mostrando como os dados simulados correspondem à aquisição real. O conjunto de dados é disponibilizado para download.

3.2 SmartSim

Gera dados por segundo dos aparelhos, levando em consideração o padrão de consumo do aparelho quando está ligado, a frequência de utilização do mesmo e o período que este passa ligado. Foi o primeiro simulador de dispositivos com resolução de 1Hz com intenção de dar suporte ao desenvolvimento e estudos de técnicas a cerca de energia elétrica. Possui integração com o formato do *NILMTK*, gerando arquivos em HDF5. Baseia os sinais dos aparelhos gerados comparando com os aparelhos presentes nos *datasets Smart** [Barker *et al.* 2012] e *REDD* [JZ 2011] a fim de verificar se a geração de sinais está próxima o suficiente dos aparelhos reais. A casa simulada é a soma dos aparelhos individuais escolhidos. Os modelos de dispositivos são baseados nos modelos descritos em [Barker *et al.* 2013], pois são formas de representar os principais dispositivos que causam impacto no consumo de energia. Os aparelhos são gerados de acordo com os seguintes comportamentos:

- **Modelos ON-OFF:** Geralmente aparelhos simples, que possuem um estado definido quando estão ligados, um exemplo pode ser um forno elétrico.
- **Modelos on-off crescimento/decaimento:** Aparelhos que possuem um pico inicial e após esse período tem um crescimento/decaimento do consumo até atingir um estado estável. Exemplo ar-condicionados e geladeiras.
- **Modelos mínimo/máximo estáveis:** Possuem um máximo ou mínimo estável mas possuem grandes variações a depender do uso. Exemplos podem ser computadores e TVs.
- **Modelos aleatórios:** Possuem um consumo aleatório quando estão em operação. Exemplo de aparelho é um micro-ondas.

- Categorias adicionais: **Modelos cíclicos:** Operam em intervalos regulares baseados em um temporizador ou termostato. **Modelos compostos:** Dispositivos que possuem múltiplas cargas. Secadoras incluem uma carga resistiva para o aquecimento e um motor com característica indutiva.

Para simular imprecisões causadas pelos medidores, o *SmartSim* adiciona um ruído de 1% uniformemente distribuído ao redor do valor correto gerado. Ao gerar o padrão de uso de energia de um dispositivo para um determinado dia, o *SmartSim* primeiro determina o número de vezes que o dispositivo é usado em um dia sorteando um número da frequência distribuição. O *SmartSim* determina então a duração do uso a partir da distribuição de duração.

3.3 SynD

O *SynD* apresenta um conjunto de dados sintéticos de consumo de energia com foco no setor residencial. *SynD* fornece 180 dias de uma residência simulada com 21 aparelhos domésticos. Os modelos de eletrodomésticos personalizados a partir do resultado de uma campanha de medição em duas famílias na Áustria e aplicando uma abordagem de modelagem semelhante a [Buneeva e Reinhardt 2017] e [Barker et al. 2013]. A casa da família simulada no *SynD* pode ser associado a um estilo de vida descontraído de uma pessoa solteira ou de um jovem casal. Implementa um gerador de dados que utiliza modelos de aparelhos personalizados para simular uma residência a partir de determinados parâmetros de entrada como taxa de amostragem e duração. Como conjuntos de dados tradicionais de consumo de energia, o *SynD* fornece dados agregados, bem como leituras de energia de eletrodomésticos individuais.

O *dataset* segue as sugestões indicadas por [Klemenjak et al. 2019] como requisitos para datasets de energia elétrica. Sendo compatível com a ferramenta *NILMTK* e os algoritmos estado da arte de acordo com [Batra et al. 2019] Tanto as leituras agregadas e submétricas são geradas em uma frequência de 5 Hz, consideradas adequadas para investigações NILM de baixa frequência segundo [Shin et al. 2019]. Além de dados gerados sobre a energia, no arquivo em formato *HDF5* também possui metadados que podem ser úteis para desagregação dos dados.

O padrão de consumo de cada aparelho é representado pela curva de consumo ao longo do tempo para uma única utilização. O medidor utilizado foi o *Rohde & Schwarz HMC8015 power analyzer* que possui uma acurácia de medição de 99,95% e uma taxa de amostragem de 100ms estando de acordo com o recomendado pelo [Klemenjak et al. 2019]

Foram modelados os seguintes comportamentos para os aparelhos:

- Sempre ligado: Aparelhos que consomem energia sempre que estão ligados e possuem um consumo constante. Exemplo é um roteador wifi. Ao simular este tipo de aparelho,

o simulador insere o padrão de consumo registrado durante a medição e o insere até que o consumo para 1 dia seja gerado.

- Periódico: Aparelhos que funcionam sem interferência e tem um padrão de consumo periódico e recorrente. Geladeiras se encaixam nesta categoria. Ao simular este tipo de aparelho, múltiplos ciclos foram gravados durante a medição, o simulador insere os dados gravados até que os dados para 1 dia sejam gerados.
- Padrão único: Aparelhos que precisam de um usuário para iniciar a operação, ou de uma programação específica. Aquecedor de água se encaixa nesta categoria. Ao simular este tipo de aparelho, o simulador seleciona um padrão gravado e insere nos dados.
- Multi-padrão: Aparelhos que oferecem muitos modos de operação e dentre esses modos há grandes diferenças no padrão de consumo. Aparelhos que se encaixam nesta categoria são lava louças, lava roupas. Ao simular este tipo de aparelho, o simulador escolhe aleatoriamente um dos padrões gravados e o insere.

O consumo simulado é gerado dia a dia de forma independente, não tendo relação o consumo de um dia com o anterior nem o posterior. Para cada um dos dias obtém-se o consumo dos aparelhos individualmente, a medição geral é obtida com a soma dos aparelhos individuais. Também considera-se que os aparelhos tem a utilização individual e que um determinado aparelho não afeta o comportamento de outro. As simulações de aparelhos têm as seguintes características que podem ser alteradas: intervalo de amostragem, duração e modelo de comportamento. Por padrão o simulador gera 180 dias com intervalo de amostragem de 0,2s. Por ser um simulador focado em consumo, a geração de energia ocorre apenas com a potência ativa, para a geração de dados de um novo dia os seguintes passos são seguidos:

1. Seleção de um padrão de consumo a partir dos templates.
2. Interpolação do padrão selecionado
3. Identificação do tempo de uso e inserção do padrão selecionado.

Depois de gerar os dados para aparelhos de padrão único e multi padrão uma variável aleatória e randômica com probabilidade uniforme decide se o padrão gerado será ignorado durante a geração daquele dia de consumo. Tal variável e modelo de decisão é baseado na utilização do [Monacchi *et al.* 2014]. Para que a simulação dos aparelhos não ocorra sempre nos mesmo horário, o que levaria a um erro, pois em um cenário real há variações nos horários de utilização dos aparelhos, por este motivo há desvios no horário simulado para utilização do aparelho. Um mecanismo que seleciona aleatoriamente o horário de utilização dos aparelhos dentro de uma janela de tempo pré-definida. Tais janelas de tempo pré-definidas são provenientes do estudo [Monacchi *et al.* 2017] e da campanha de medição real proveniente

de [Monacchi *et al.* 2014]. Para que as amostras sigam a distribuição normal, o gerador segue a seguinte função de densidade de probabilidade:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (3.1)$$

onde σ é o desvio padrão em minutos, μ é a média da distribuição, obtida calculando o tempo em minutos decorrido da meia noite até a hora atual e x é um inteiro selecionado aleatoriamente gerado pela biblioteca *Numpy*.

Para realizar a validação dos dados gerados foram utilizados os seguintes *datasets* reais *DRED* [Nambi, Lua e Prasad 2015], *ECO* [Beckel *et al.* 2014], *REFIT* [Murray *et al.* 2015] e *UK-DALE* [Kelly e Knottenbelt 2015] não só para validação do consumo agregado da residência, mas também para validação dos dados gerados por aparelho.

A validação do consumo geral da casa é feita seguindo os parâmetros citados em [Wang *et al.* 2018] que provê uma revisão sobre características de *smart meters*. Para objetivo de comparação entre os *datasets*, 40 dias de dados foram gerados, período presente em todos os *datasets* analisados, de forma que nenhum tivesse uma amostra maior que outro. Baseado na média de consumo diário os resultados indicaram que a casa sintética do *SynD* aparenta ser similar à casas reais. Uma segunda análise feita com relação aos horários de consumo, uma vez que a utilização de máquina de lavar roupas na madrugada não deve ser considerado um hábito normal. Para esta análise, o dia foi dividido em 48 intervalos de 30 minutos cada, a fim de comparação dos dados reais e simulados em relação aos horários de maior consumo. Notou-se uma diferença grande com relação ao consumo no horário da manhã, um dos possíveis motivos pode ser o fato de não ter nenhum aparelho que seja utilizado com maior frequência pela manhã. Um segundo motivo pode ser o hábito das famílias, uma vez que as famílias observadas são jovens e pode ter hábitos mais noturnos, horário no qual foi notado um maior consumo nos dados gerados.

A validação dos dados gerados por aparelho foi feita de duas formas: na primeira comparase o consumo dos aparelhos individualmente em uma janela de 40 dias, na segunda medidas estatísticas são aplicadas para examinar as similaridades entre os *datasets*. Analisando a média de consumo no período determinado, nota-se que apesar do consumo do *dataset SynD* ser próximo a outros utilizados na comparação, há uma grande variação com relação ao consumo de cada aparelho. Tanto por hábitos de consumo quanto por diferenças nos aparelhos. Na análise de refrigeradores os números de consumo variaram desde 5,49KWh à 30,85KWh na comparação entre os *datasets*, indicando que o impacto no consumo por aparelho depende mais do modelo e hábito de consumo do que o tipo de aparelho utilizado. Aparelhos diferem em padrões de consumo e níveis de consumo, para que possam ser comparados análises estatísticas foram aplicadas a fim de avaliar a similaridade dos dados sintéticos com os dados reais. Para que não houvessem diferenças entre os *datasets*, o período de 40 dias foi extraído da mesma época do ano para não apresentar sazonalidade nos dados, e a taxa de amostragem foi feita a cada 10 segundos, analisados com a seguinte função massa de probabilidade para

cada série temporal descrita em [Makonin *et al.* 2015]. A métrica escolhida foi a distância de Hellinger, que é definida como a norma Euclidiana da diferença dos quadrados de duas distribuições de probabilidade discretas P e Q :

$$D_H(P||Q) = \frac{1}{\sqrt{2}} \cdot \sqrt{\sum_{x \in X} \left(\sqrt{P(X)} - \sqrt{Q(X)} \right)^2} \equiv \frac{1}{\sqrt{2}} \cdot \|\sqrt{P} - \sqrt{Q}\|_2 \quad (3.2)$$

O valor da distância de Hellinger 0 indica uma similaridade total, onde a total discrepância possui valor 1. Como complemento da análise estatística, aplicou-se uma segunda medida estatística, a distância de *Jensen-Shannon* [Lin 1991] para avaliar a similaridade das funções massa de probabilidade, esta distância mede a similaridade entre duas distribuições de probabilidade P e Q , sendo definida como a raiz quadrada das divergências de *Kullback-Leibler* [MacKay, Kay *et al.* 2003].

$$D_{JS}(P||Q) = \sqrt{\frac{1}{2} \cdot (D_{KL}(P||M) + D_{KL}(Q||M))} \quad (3.3)$$

esta distância é simétrica e sempre retorna um valor finito, onde M é definido como o ponto médio entre P e Q :

$$M = \frac{1}{2} \cdot (P + Q) \quad (3.4)$$

A distância de *Kullback-Leibler* referida como entropia relativa, é a diferença logarítmica entre P e Q , onde a expectativa é em relação às probabilidades de P :

$$D_{KL}(P||Q) = \sum_{x \in X} P(x) \cdot \log \left(\frac{P(x)}{Q(x)} \right) \quad (3.5)$$

Ao finalizar a simulação os dados podem ser salvos de duas formas, a primeira delas em formato *HDF5* contendo os dados das medições armazenados como *DataFrame* a biblioteca *pandas* indexados pelo *timestamp* de cada medição. A segunda forma de salvar os dados é em um arquivo *ZIP* contendo os metadados e 22 arquivos no formato *.csv*, um para cada aparelho gerado e um para a potência agregada.

3.4 Considerações Finais

Neste capítulo foram apresentados os conjuntos de dados que serão analisados no 4. Apresentou-se os *datasets* que serão utilizados, bem como sua construção e métricas.

Capítulo 4

Desenvolvimento

4.1 Escolha dos Datasets

Dentre os *datasets* encontrados apenas dois tinham dados abertos, o *SynD* e *SmartSim*, ambos possuem a característica de serem com foco residencial, possuem diversos aparelhos e são compatíveis com o *NILMTK*, além de serem gerados sinteticamente.

Um *dataset* real previamente construído possui 23 refrigeradores reais coletados, de diferentes marcas e capacidades, que foram coletados em casas reais com diferenças de utilização. Como medidor utilizado com nome de *LiteMeter*, possui frequência de amostragem de 1Hz *true RMS* onde são coletadas diversas grandezas elétricas, no entanto, utiliza-se somente potência ativa e reativa para o presente trabalho.

Os conjuntos de dados possuem diferenças nos aparelhos coletados, apesar de vários aparelhos serem coletados, apenas micro-ondas, máquina de lavar roupas e refrigerador eram os aparelhos presentes em tanto nos conjuntos sintéticos quanto no conjunto de dados real. Apesar de consumo elevado, micro-ondas e máquina de lavar possuem impacto menor no consumo geral de uma residência, de acordo com a análise feita no *NILMTK* [Batra *et al.* 2014] aparelhos que representam mais de 5% no total de energia consumida são ar-condicionado, aquecedores e refrigeradores, por este motivo a análise se concentrou em refrigeradores. O que torna a análise deste aparelho com impacto real na rede neural.

Para a avaliação da rede neural treinada utilizou-se um outro conjunto de dados de residências reais coletadas especificamente para tal finalidade, uma vez que estes dados não são utilizados nas etapas de treino. Este conjunto de dados consiste de 4 residências reais onde os aparelhos são medidos individualmente, bem como a energia total da residência. Para avaliação há também a construção de cenários simulados a partir dos dados utilizados para treinamento, junto com outros aparelhos que não foram utilizados no treinamento como ar-condicionado, chuveiro elétrico, micro-ondas, máquinas de lavar roupas, televisores e ventiladores. A fim de notar quanto a rede treinada com dados sintéticos é eficiente no caso de uma aplicação com dados reais.

4.2 Análise Exploratória

Inicialmente ambos conjuntos de dados sintéticos foram analisados a fim de atestar a compatibilidade com dados reais e a capacidade do *WaveNILM* de aprender as características dos *datasets* sintéticos. Um comparativo dos dados foi realizado, na figura 4.1 onde podemos observar um gráfico com a potência ativa relativa a 1 ciclo de utilização de cada um dos conjuntos de dados.

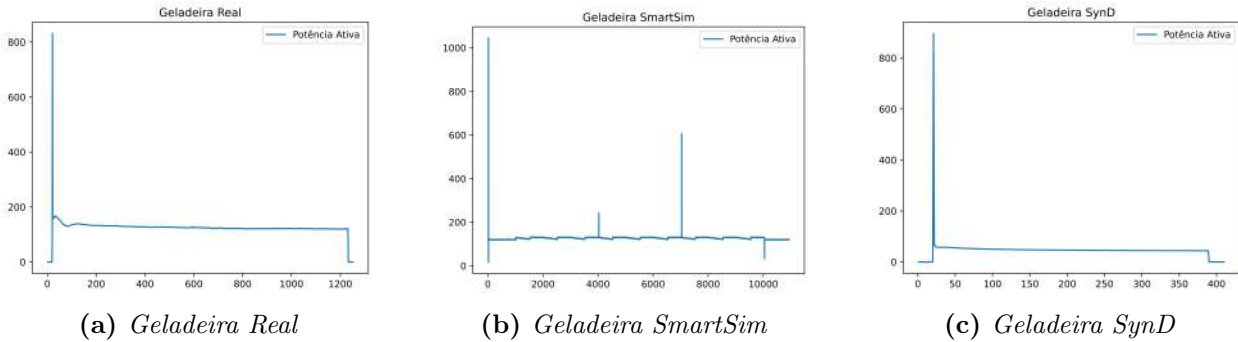


Figura 4.1: *Comparativo Refrigeradores*

Uma análise inicial mostra que o comportamento do refrigerador presente no *dataset SynD* 4.1c é semelhante ao comportamento de um dos refrigeradores presentes nos dados reais 4.1a, com ambas tendo um pico de ativação um pouco acima dos 800 watts e um comportamento decrescente até que o motor seja desligado e o consumo tenda a zero, embora o refrigerador real tenha um ciclo de aproximadamente 1200 segundos e o *SynD* tem ciclo mais curto de apenas 380 segundos. No entanto, o refrigerador presente no *SmartSim* possui um comportamento diferente dos demais, não apresentando um ciclo definido de utilização, iniciando com um pico de consumo superior a 1000 watts havendo apenas um pequeno desligamento após 10000 segundos, no entanto, instantaneamente é religada. A geladeira do *SynD* é do modelo *KB 289 Kühlbox mit Eisfach* fabricada pela *Bomann*, enquanto que a do exemplo real é do modelo *270* marca *Continental*, o *dataset SmartSim* não especifica qual modelo e fabricante foi utilizado para geração do modelo sintético.

4.2.1 Design de Experimento

Por se tratarem de *datasets* sintéticos não há necessidade de levantamento das estatísticas como detecção de lacunas, taxa de *dropout* e *up-time* citadas no capítulo 2. Uma vez que na geração dos mesmos, tais parâmetros podem ser setados. Portanto, podemos afirmar que ambos *datasets* utilizados possuem 100% de *up-time*, uma taxa de *dropout* de zero e 100% da energia coletada nos aparelhos também está presente na energia total.

Em se tratando dos dados reais coletados utilizados, por serem utilizados previamente em outras aplicações, um pré-processamento havia sido realizado. Pode-se fazer a mesma afirmação quanto às estatísticas do *dataset* real, possui 100% de *up-time*, uma taxa de

dropout de zero e 100% da energia coletada nos aparelhos também está presente na energia total.

Os dados reais utilizados e o *dataset SmartSim* possuem uma taxa de amostragem de 1Hz, então o *dataset SynD* foi gerado novamente utilizando-se esta taxa de amostragem, a fim de normalizar todos os conjuntos de dados utilizados. O tempo de coleta dos aparelhos reais é de aproximadamente 1 semana, o *dataset SmartSim* gera dados para 1 semana de análise, então ao gerar o *dataset SynD* com a nova taxa de amostragem, ao invés de gerar 180 dias apenas 7 foram gerados. Desta forma há um balanço nos dados, para que desta forma ao treinar a rede neural, esta não ficasse enviesada por nenhum conjunto de dados.

Durante o desenvolvimento observou-se a necessidade de adequação dos dados reais, pois os mesmos são coletados em formato *.csv*, onde cada arquivo contém os dados coletados de cada aparelho, incluindo a potência total coletada da residência, diferindo dos formatos aceitos pela ferramenta que são *.h5* ou *.hdf5*.

Ao converter os dados para o formato requerido, realiza-se a separação dos períodos de ativação dos aparelhos utilizando-se a ferramenta *NILMTK*. Pois a mesma possui uma funcionalidade que detecta eventos de ligar/desligar os aparelhos a partir dos dados obtidos. Esta etapa se faz necessária, pois o treinamento da rede neural utilizando-se os dados incluindo o período em que o aparelho não é utilizado leva a uma rede neural enviesada, levando em consideração que a frequência de utilização de cada aparelho reflete também o comportamento do usuário, não relevante para a pesquisa proposta. Uma vez detectados e registrados os eventos de cada aparelho, inicia-se o treinamento da rede neural.

A rede neural convolucional causal dilatada utilizada possui 5 camadas, com 512, 128, 128, 128 e 1 filtros em cada camada respectivamente totalizando 1.447.044 parâmetros treináveis. A entrada da rede neural recebe 512 amostras, a uma taxa de amostragem de 1 amostra por minuto que corresponde a aproximadamente 8,5 horas de dados. Esta entrada para a rede neural é sugerida por [Harell, Makonin e Bajić 2019] por possuir um campo receptivo grande suficiente para a detecção e desagregação dos aparelhos residenciais utilizados. Para o treinamento a rede recebe a potência total da residência e a potência do aparelho referentes ao mesmo instante. A saída possui um único neurônio e sua saída corresponde a potência ativa do aparelho analisado. Para o treinamento utilizou-se a validação cruzada de *10-fold* onde o dado total é dividido em 10 porções, 9 são utilizadas para o treinamento e 1 para teste. O *learning rate* inicial utilizado para todos os experimentos foi de 0,0001 mas não foi fixado para o experimento, o otimizador utilizado foi o *Adam* [Kingma e Ba 2014] que ajusta o *learning rate* automaticamente ao longo do treinamento.

A métrica utilizada para avaliação foi a acurácia estimada, conforme definição apresentada no capítulo 2.

As simulações foram feitas utilizando-se a ferramenta *NILMTK*, divididas em três experimentos:

- Experimento 1: Simulação dos dados sintéticos com a rede neural escolhida a fim de

validar a capacidade de aprendizado;

- Experimento 2: Simulação de cada um dos dados sintéticos com os dados reais a fim de testar o impacto com a adição de um *dataset* para treinamento em conjunto com os dados reais;
- Experimento 3: Simulação envolvendo ambos *datasets* sintéticos e os dados reais, a fim de testar o impacto com a adição de ambos *datasets* sintéticos para treinamento em conjunto com os dados reais.

No primeiro experimento os *datasets* sintéticos *SynD* e *SmartSim* foram separadamente treinados com a rede neural, sem a utilização de nenhum tipo de dado adicional, a fim de avaliar se os dados destes realmente podem ser aprendidos pela rede, bem como para mostrar que no caso de em um treinamento individual, ambos alcançam resultados satisfatórios e podem ser utilizados em conjunto com os dados reais, a fim de realizar aumento na quantidade de dados disponível. Nesta etapa, como os dados são insuficientes por não terem uma boa generalização por representarem apenas um único comportamento, não foi realizada a avaliação da rede neural após o treinamento.

No segundo experimento o *SynD* foi treinado juntamente com os dados reais e em um segundo treinamento o *SmartSim* foi treinado com juntamente os dados reais. Tal experimento foi realizado para analisar o impacto causado na desagregação do refrigerador quando os *datasets* são treinados juntamente com os dados reais.

No terceiro experimento tanto os dados dos *datasets* sintéticos quanto os dados reais foram utilizados para o treinamento final da rede. Podendo observar qual impacto da adição de mais dados teriam sobre os dados já utilizados, a fim de avaliar se com maior quantidade de dados misturando os sintéticos e reais a acurácia da rede seria melhor do que com os treinamentos individuais.

As etapas 2 e 3 são divididas em 3 tipos de simulações, a primeira com dados de 3 meses para o treinamento, a segunda com dados de 6 meses para o treinamento e a terceira conta com a simulação de 1 ano de dados para o treinamento. Para conseguir gerar tais dados, utiliza-se um "*simulador*" para gerar dados de casas sintéticas para o treinamento e avaliação da rede, o simulador é usado para pegar dados coletados de aparelhos individuais de forma aleatória e misturá-los para simular o consumo de uma casa durante o período desejado, esta junção é uma mera concatenação com uma mistura nos dados dos aparelhos utilizados.

Para que a comparação pudesse ser feita de forma igual, a quantidade de épocas foi a mesma em todas as três etapas de simulação. Foram executadas 100 épocas com a mesma rede neural em todas, para que a variável presente fosse a adição de novos dados. Para avaliar a rede neural após o treinamento, a mesma é submetida a cinco cenários simulados, os quais simulam casas com quantidades diferentes de aparelhos, cenários estes que são construídos de forma análoga à utilizada para gerar maior quantidade de dados de treinamento, e a quatro cenários reais diferentes com intervalo de avaliação de 1 mês tanto para cenários simulados

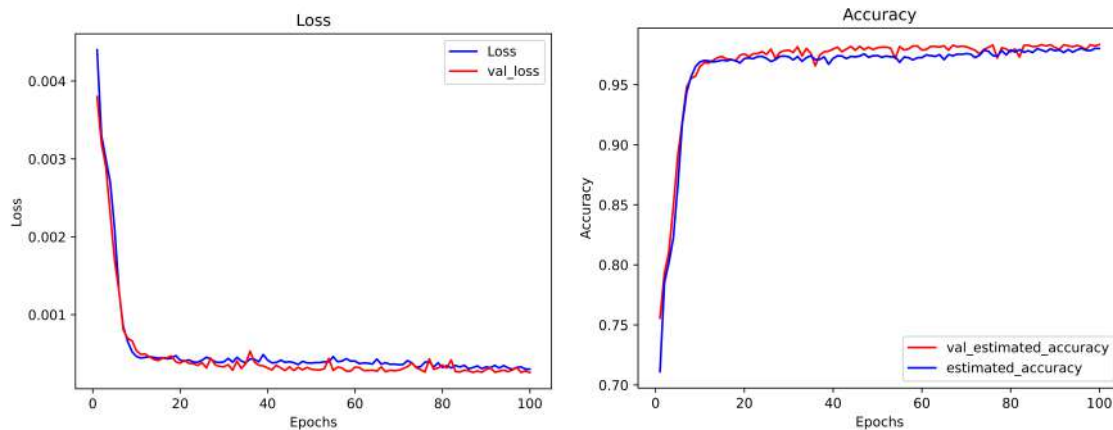
quanto reais. Os 4 cenários reais coletados com *full-submetering*, ou seja, todos os aparelhos das residências são coletados.

4.3 Resultados e Discussões

4.3.1 Simulações dados sintéticos individuais

Nesta primeira etapa, é realizado o treinamento dos conjuntos de dados individualmente para atestar a capacidade da rede neural de aprender os dados analisados. Nesta etapa foram utilizados apenas os 7 dias gerados para ambos *datasets*.

Com a análise do treinamento individual do refrigerador do *dataset SmartSim* na rede neural sem a utilização de nenhum outro dado, obteve-se o resultado presente na figura 4.2. Mostra que a rede neural utilizada é capaz de aprender o padrão, não sendo necessárias as 100 épocas utilizadas como padrão para os experimentos. O resultado do treinamento ocorre com uma perda baixa mostrado na figura 4.2a obtendo um resultado satisfatório antes de atingir 20 épocas. Como mostrado na figura 4.2b a acurácia chega perto de 100% rapidamente.



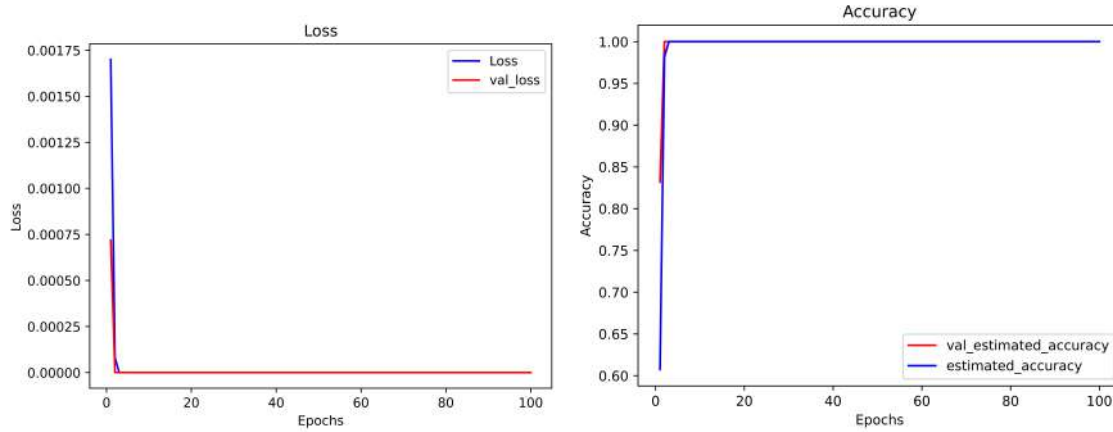
(a) Perda treinamento individual

(b) Acurácia treinamento individual

Figura 4.2: Resultados do treinamento individual refrigerador *SmartSim*

Após a análise do treinamento individual do refrigerador *SynD* foram gerados os gráficos da figura 4.3. Nota-se que o modelo chega ao resultado desejado ainda mais rápido que nos dados anteriores, conforme mostrado na figura 4.3a a perda chega a zero em menos de 10 épocas. Analisando a figura 4.3b, assim como na perda, ela chega no valor próximo de 1 muito rapidamente.

Após a realização de ambas simulações, conclui-se que a rede é capaz de aprender o padrão dos *datasets* após poucas épocas, demonstrando a possibilidade de serem utilizados para *data augmentation* com os dados reais, bem como a eficácia da rede neural utilizada para aprender tais padrões utilizados. Conforme observado, não há a necessidade de geração



(a) Perda treinamento individual

(b) Acurácia treinamento individual

Figura 4.3: Resultados do treinamento individual refrigerador *SynD*

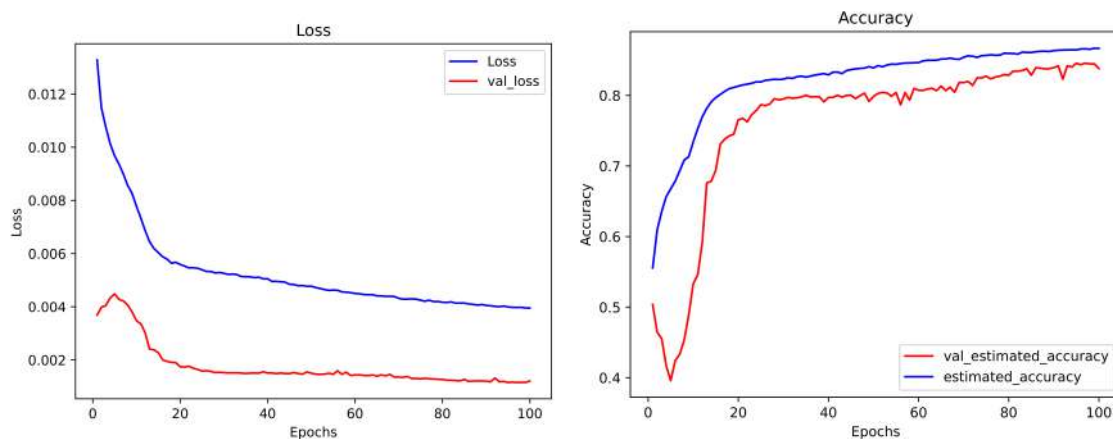
de mais dias, uma vez que a rede chegou a um resultado satisfatório antes mesmo de ter o experimento concluído.

4.3.2 Simulações dados reais e sintéticos

- *SmartSim*

Com *SmartSim* após as simulações geradas, utilizando 3 meses de dados para treinamento, o resultado do treinamento é mostrado na figura 4.4, chegando a uma perda de 0,0039 uma perda de validação de 0,0012 acurácia estimada de 0,8665 e uma acurácia estimada de validação de 0,8380.

O comparativo entre as 3 simulações está presente na tabela 4.1.

(a) Perda treinamento *SmartSim*(b) Acurácia treinamento *SmartSim***Figura 4.4:** Resultados do treinamento refrigerador *SmartSim* e dados reais 3 meses de dados

Depois de criar uma simulação usando 6 meses de dados de treinamento, os resultados obtidos são apresentados na figura 4.5, atingindo uma perda de 0,0035 uma perda de va-

lidaçãõ de 0,0008 acurácia estimada de 0,8835 e uma acurácia estimada de validaçãõ de 0,8887

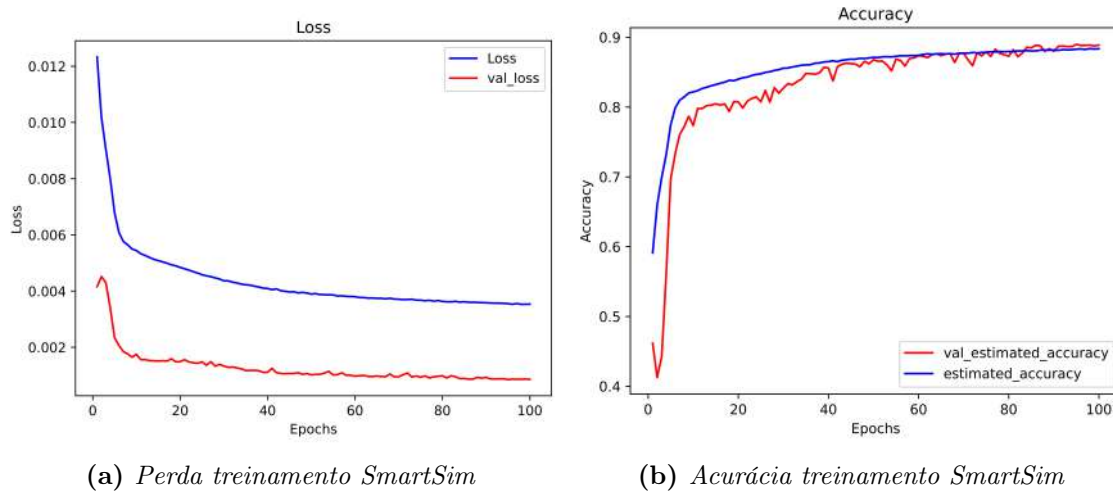


Figura 4.5: Resultados do treinamento refrigerador SmartSim e dados reais 6 meses de dados

Na última simulação desta etapa com os dados do *SmartSim* e dados reais por 1 ano alcançou-se o valor de perda de 0,0031 uma perda de validaçãõ de 0,0007 acurácia estimada de 0,8970 e uma acurácia estimada de validaçãõ de 0,9080. Não sendo muito superior aos treinamentos anteriores, no entanto, apresentando uma pequena melhora que se reflete na avaliação da rede neural presente na tabela 4.1. Os resultados do treinamento são mostrados na figura 4.6.

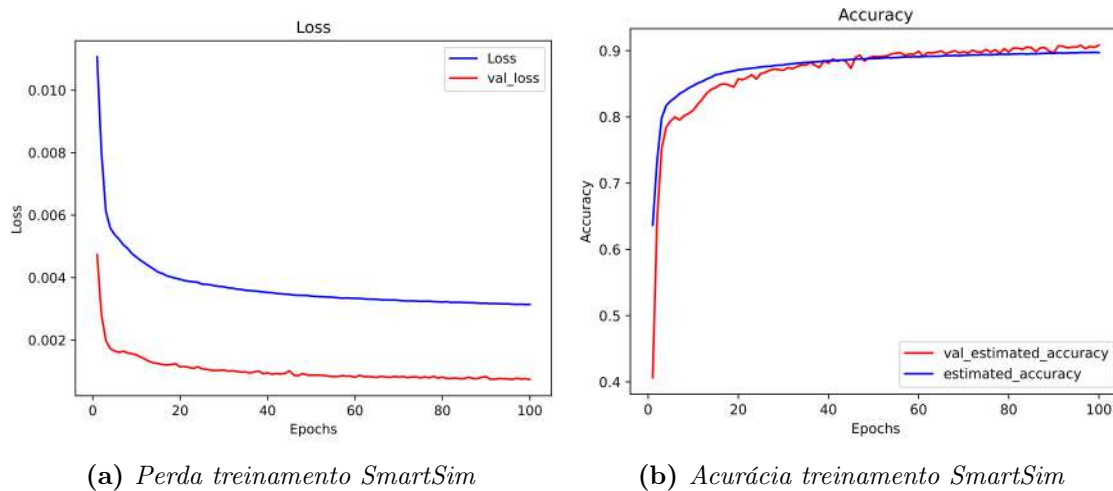


Figura 4.6: Resultados do treinamento refrigerador SmartSim e dados reais 1 ano de dados

A tabela mostra uma evolução na avaliação da rede neural treinada, quanto maior é o tempo de dados utilizado para a simulação. A rede treinada com 1 ano de dados se saiu melhor que as demais em todos os cenários simulados, no entanto, nos cenários reais não ocorre o mesmo comportamento, a rede treinada com apenas 3 meses de dados se saiu melhor, o que pode indicar que para os cenários reais a rede com mais dados pode ter chegado a

um sobreajuste (*overfit*) nos dados de treinamento, uma vez que os dados para as casas simuladas e os dados de treinamento pertencem ao mesmo conjunto de dados.

Cenários-Período	3 meses	6 meses	1 ano
Casas Simuladas	81,3%	83,8%	85,3%
Casas Reais	65,6%	58,9%	61,9%

Tabela 4.1: Tabela acurácia de avaliação *SmartSim*

- *Synd*

Gerando dados de 3 meses para a primeira simulação com o *Synd* e os dados reais, a rede alcançou uma perda de 0,0039 uma perda de validação de 0,0009 acurácia estimada de 0,8672 e uma acurácia estimada de validação de 0,8608. Estes resultados presentes na figura 4.7 são ligeiramente melhores que com o conjunto de dados anteriores, mas não significativos a ponto de ser uma melhora relevante.

Resumo da avaliação do *Synd* estão presentes na tabela 4.2

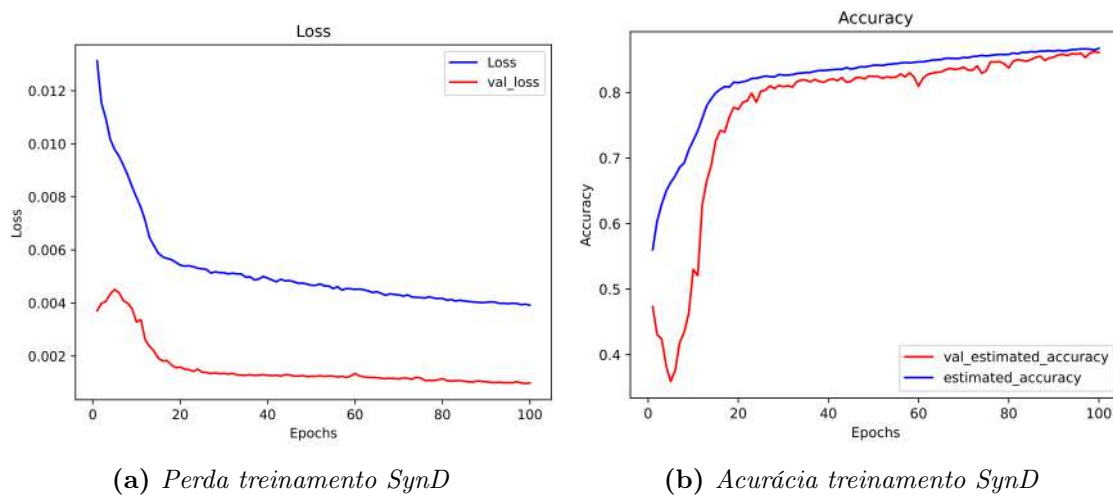


Figura 4.7: Resultados do treinamento refrigerador *Synd* e dados reais 3 meses de dados

Utilizando-se seis meses de dados para o treinamento a rede chegou a uma perda de 0,0033 uma perda de validação de 0,0007 acurácia estimada de 0,8877 e uma acurácia estimada de validação de 0,8914. Resultado mostrado na figura 4.8, similar ao obtido no conjunto de dados anterior para os dados de treinamento com 6 meses.

Treinar a rede com os dados de 1 ano chegando a uma perda de 0,0029 uma perda de validação de 0,0006 acurácia estimada de 0,9005 e uma acurácia estimada de validação de 0,9133, resultados mostrados na figura 4.9, são superiores aos obtidos com o *SmartSim* mas que não representam avanço significativo.

A tabela 4.2 mostra uma situação análoga a encontrada com a tabela 4.1, indicando que quando avaliada nas casas simuladas, a rede se mostra melhor quanto maior é quantidade de dados utilizados para o treinamento, no entanto, para os cenários reais acontece o inverso. Mostrando que há um provável sobreajuste *overfit* nos dados de treinamento.

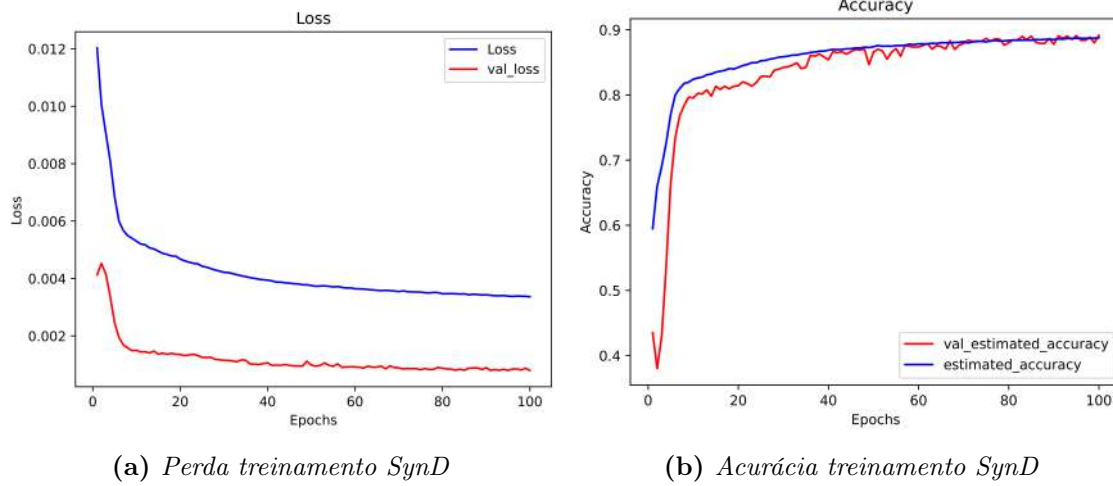


Figura 4.8: Resultados do treinamento refrigerador SynD e dados reais 6 meses de dados

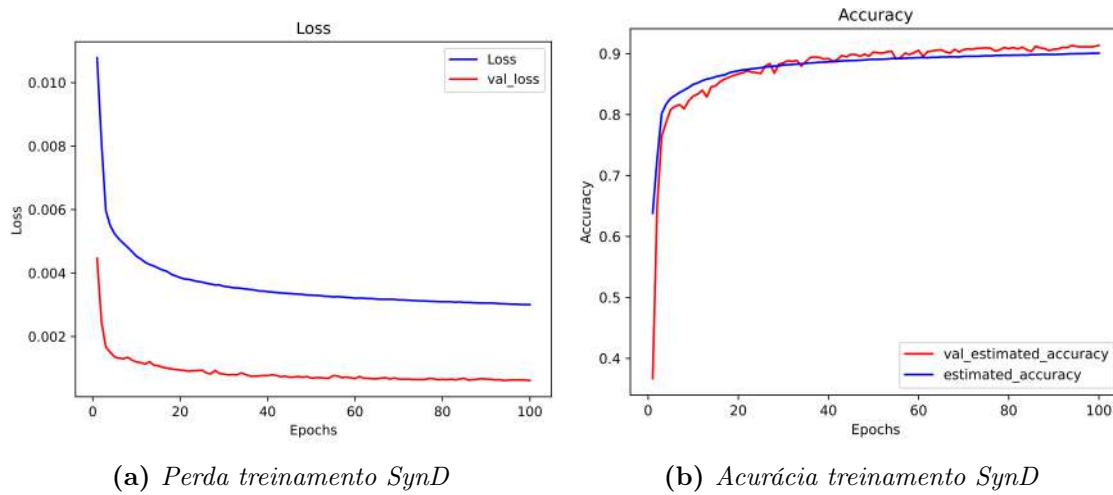


Figura 4.9: Resultados do treinamento refrigerador SynD e dados reais 1 ano de dados

Cenários-Período	3 meses	6 meses	1 ano
Casas Simuladas	79,2%	83,1%	85,4%
Casas Reais	57,4%	49,9%	55,3%

Tabela 4.2: Tabela acurácia de avaliação SynD

4.3.3 Todos os dados disponíveis

Ao utilizar todos os dados disponíveis para treinamento, com a simulação de 3 meses a rede tem um resultado com uma perda de 0,0039 uma perda de validação de 0,0011 acurácia estimada de 0,8640 e uma acurácia estimada de validação de 0,8484, mostradas na figura 4.10.

Após gerar 6 meses de dados com os dados disponíveis a rede tem um desempenho superior mostrado na figura 4.11, com uma perda de 0,0034 uma perda de validação de 0,0008 acurácia estimada de 0,8847 e uma acurácia estimada de validação de 0,8956.

Treinamento dados de 1 ano gerados com a utilização de todos os dados disponíveis obtém o melhor resultado dentre os experimentos realizados, sendo mostrado na figura 4.12,

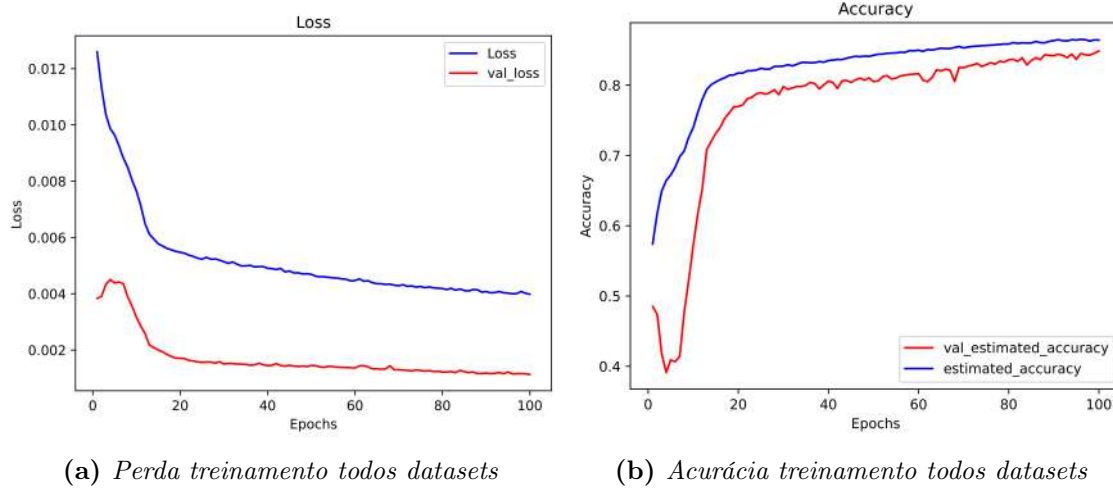


Figura 4.10: Resultados do treinamento refrigerador ambos datasets e dados reais 3 meses de dados

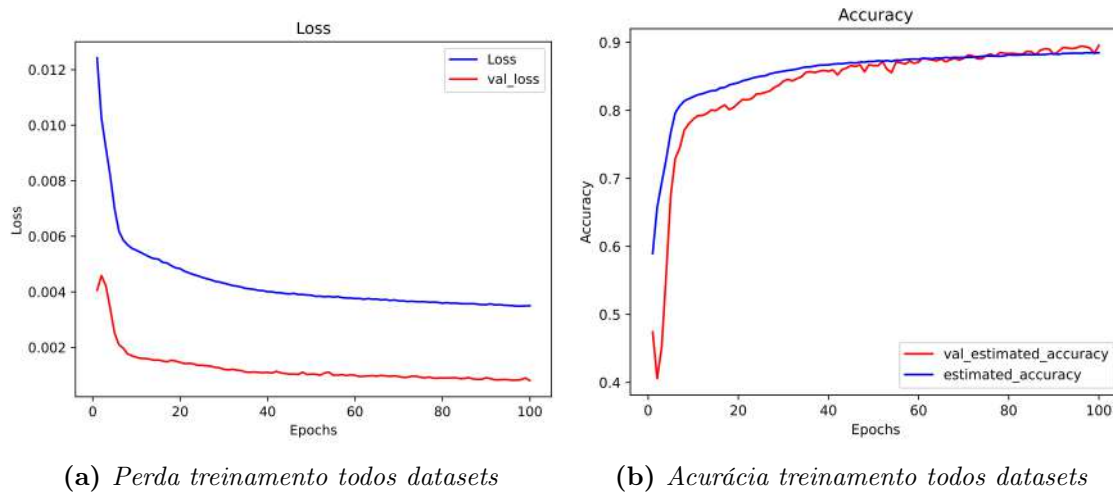


Figura 4.11: Resultados do treinamento refrigerador ambos datasets e dados reais 6 meses de dados

com métricas para a rede atingindo uma perda de 0,0031 uma perda de validação de 0,0007 acurácia estimada de 0,8981 e uma acurácia estimada de validação de 0,9099.

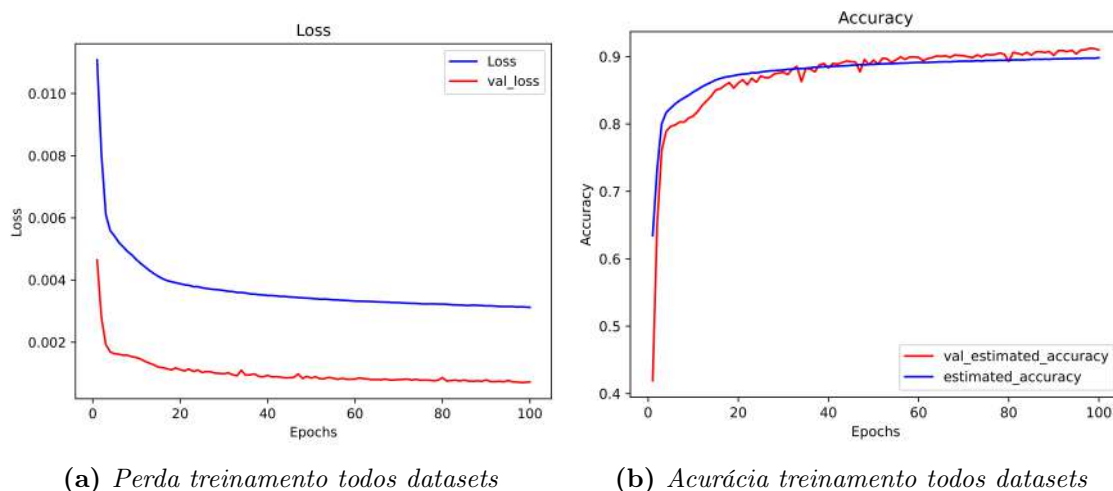


Figura 4.12: Resultados do treinamento refrigerador ambos datasets e dados reais 1 ano de dados

Ao analisarmos a tabela 4.3 de avaliação da rede neural treinada com todos os dados nos 3 períodos de tempo, nota-se que a rede é mais consistente, mas não sendo o melhor resultado em todos os cenários analisados.

Cenários-Período	3 meses	6 meses	1 ano
Casas Simuladas	80,7%	83,7%	85,0%
Casas Reais	63,0%	57,3%	61,4%

Tabela 4.3: Tabela acurácia de avaliação com todos os dados disponíveis

4.4 Considerações Finais

Tabela 4.4 apresenta o resumo das avaliações de todas as redes treinadas.

Pode-se observar um resultado superior quando as redes são treinadas com um intervalo de dados maior e avaliadas nos cenários simulados, uma possível explicação para este fato é que os dados utilizados para a geração de cenários simulados são os mesmos utilizados para treinamento adicionados de um ruído. Portanto, a rede consegue aprender melhor esses padrões, inclusive o melhor resultado é obtido pelo cenário onde ambos *datasets* sintéticos estão presentes no treinamento em conjunto com os dados reais.

Uma explicação para a baixa acurácia obtida com o *SynD* seja o fato de ser focado em consumo de energia, portanto, não há dados sobre potência reativa, apesar da geração de potência ativa se assemelhar com uma geladeira real coletada conforme observado na figura 4.1.

Ao analisar o cenário de avaliação com residências reais, os melhores resultados foram obtidos utilizando-se apenas 3 meses de dados para o treinamento, por ser um resultado recorrente leva a crer que há um *overfit* nos dados simulados, piorando a rede para um caso mais genérico quando a mesma é aplicada em dados desconhecidos.

O resultado da rede treinada apenas com os dados reais obteve nos melhores resultados uma acurácia de 64,9% nos cenários reais analisados e 84,9% nos cenários simulados. Sendo os resultados com a utilização do *dataset SmartSim* superiores com 65,6% nos cenários reais analisados e 81,3% nos cenários simulados. Portanto, conclui-se que a utilização do conjunto de dados deste *dataset* melhorou a acurácia da rede em 0,7%, não sendo uma melhora significativa, mas ocasionou um aumento da acurácia de avaliação.

	Casas Simuladas			Casas Reais		
	3 meses	6 meses	1 ano	3 meses	6 meses	1 ano
SmartSim	81,3%	83,8%	58,3%	65,6%	58,9%	61,9%
SynD	72,9%	83,1%	85,4%	57,4%	49,9%	55,3%
Ambos	80,7%	83,7%	85,0%	63,0%	57,3%	61,4%
Reais	79,1%	83,6%	84,9%	64,9%	58,4%	62,65%

Tabela 4.4: *Tabela resumo das avaliações das redes neurais*

Capítulo 5

Conclusão

Com a análise dos resultados obtidos no capítulo 4, tem-se que a utilização de um *dataset* sintético pode sim contribuir para a melhoria da acurácia de desagregação de cargas em cenários reais. No entanto dos cenários analisados, o *Synd* não melhorou a acurácia de desagregação da rede neural em cenários reais, enquanto o *SmartSim* apresentou uma melhoria de desagregação para cenários reais. Tendo uma melhoria de 0,7% em relação à utilização apenas dos dados coletados de aparelhos reais.

5.1 Sugestões para Trabalhos Futuros

Como visto nos treinamentos, uma acurácia de 65,6% nos cenários reais não é um valor suficiente para afirmar que a rede possui uma generalização satisfatória e funcionará em um número maior de residências reais, visto que a amostra de dados atual pode ser considerada apenas uma pequena amostra em um universo vasto de possíveis cenários. Portanto, extensões para o presente trabalho podem ser importantes. Como sugestões para trabalhos futuros, algumas ideias que podem melhorar mais a acurácia de redes neurais para desagregação são:

1. Geração de dados sintéticos de outros aparelhos, uma vez que a rede pode aprender não só os aparelhos que tenham grande impacto na conta de energia, mas a maioria dos aparelhos da residência de forma a identificar melhor cada carga. Cargas diferentes podem ter comportamento similar, como o comportamento de um refrigerador e um ar-condicionado, no entanto a faixa de consumo de ambos no geral é diferente, tendo este um consumo mais elevado. Além de tornar possível a identificação de mais cargas pode ser útil.
2. Novas formas de geração de dados, com uma variedade maior de dentre os sinais de um mesmo aparelho, como visto nos conjuntos de dados utilizados, a utilização dos aparelhos não sofre alterações de consumo, apenas o horário de utilização é modificado na geração dos dados sintéticos, para alcançar tal objetivo pode-se utilizar por exemplo redes neurais adversariais que geram uma variedade maior de cargas simulando

aparelhos diferentes para que a rede possua um grau maior de generalização quando avaliada em cenários reais.

3. Com a quantidade de dados reais limitados, este trabalho possui uma limitação, tendo em vista que com a utilização apenas dos dados reais a acurácia é de 64,9% o que ainda pode ser melhorado através da coleta de mais aparelhos reais, pois os *datasets* sintéticos conseguem aumentar a acurácia dos dados no entanto a diversidade de sinais é limitada e não há um ganho significativo em relação a diversidade dos dados que podem ser encontrados em cenários reais. Com a coleta de aparelhos reais de tipos e marcas diferentes há uma variedade maior de padrões de consumo o que tem impacto positivo para a desagregação de cargas.

Referências Bibliográficas

[Abadi *et al.* 2015] ABADI, M. *et al.* **TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems**. 2015. Software available from [tensorflow.org](https://www.tensorflow.org/). Disponível em: [<https://www.tensorflow.org/>](https://www.tensorflow.org/). 5

[Barker *et al.* 2013] BARKER, S. *et al.* Empirical characterization and modeling of electrical loads in smart homes. In: **IEEE. 2013 international green computing conference proceedings**. [S.l.], 2013. p. 1–10. 12, 13

[Barker *et al.* 2012] BARKER, S. *et al.* Smart*: An open data set and tools for enabling research in sustainable homes. **SustKDD, August**, v. 111, n. 112, p. 108, 2012. 12

[Batra *et al.* 2014] BATRA, N. *et al.* NILMTK: An Open Source Toolkit for Non-intrusive Load Monitoring. In: **Proceedings of the 5th international conference on Future energy systems**. ACM, 2014. Disponível em: <https://doi.org/10.1145/2F2602044.2602051>. xii, 3, 4, 5, 7, 8, 11, 17

[Batra *et al.* 2019] BATRA, N. *et al.* Towards reproducible state-of-the-art energy disaggregation. In: **Proceedings of the 6th ACM international conference on systems for energy-efficient buildings, cities, and transportation**. [S.l.: s.n.], 2019. p. 193–202. 13

[Beckel *et al.* 2014] BECKEL, C. *et al.* The eco data set and the performance of non-intrusive load monitoring algorithms. In: **Proceedings of the 1st ACM conference on embedded systems for energy-efficient buildings**. [S.l.: s.n.], 2014. p. 80–89. 15

[Bousbiat, Klemenjak e Elmenreich 2020] BOUSBIAT, H.; KLEMENJAK, C.; ELMENREICH, W. Exploring time series imaging for load disaggregation. In: **Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation**. [S.l.: s.n.], 2020. p. 254–257. 11

[Bousbiat *et al.* 2020] BOUSBIAT, H. *et al.* Augmenting an assisted living lab with non-intrusive load monitoring. In: **IEEE. 2020 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)**. [S.l.], 2020. p. 1–5. 1

[Buneeva e Reinhardt 2017] BUNEEVA, N.; REINHARDT, A. Ambal: Realistic load signature generation for load disaggregation performance evaluation. In: **IEEE. 2017 IEEE international conference on smart grid communications (smartgridcomm)**. [S.l.], 2017. p. 443–448. 11, 13

[C. Kovatsch C. 2020] C. KOVATSCH C., H. M. K. A synthetic energy dataset for non-intrusive load monitoring in households. Springer Nature, April 2020. Disponível em: <https://doi.org/10.1038/s41597-020-0434-6>. 3, 11

[Chen, Irwin e Shenoy 2016] CHEN, D.; IRWIN, D.; SHENOY, P. Smartsim: A device-accurate smart home simulator for energy analytics. In: **2016 IEEE International Conference on Smart Grid Communications (SmartGridComm)**. [S.l.]: IEEE, 2016. p. 686–692. 3, 5, 11

[Géron 2019] GÉRON, A. **Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems**. [S.l.]: "O'Reilly Media, Inc.", 2019. 1

[Harell 2020] HARELL, A. **Deep learning applications in non-intrusive load monitoring**. Tese (Doutorado) — Applied Sciences: School of Engineering Science, 2020. 11

[Harell, Makonin e Bajić 2019] HARELL, A.; MAKONIN, S.; BAJIĆ, I. V. **Wavenilm: A causal neural network for power disaggregation from the complex power signal**. arXiv, 2019. Disponível em: <<https://arxiv.org/abs/1902.08736>>. 4, 6, 10, 19

[Harris *et al.* 2020] HARRIS, C. R. *et al.* Array programming with NumPy. **Nature**, v. 585, p. 357–362, 2020. 5

[Hart 1992] HART, G. W. Nonintrusive appliance load monitoring. **Proceedings of the IEEE**, IEEE, v. 80, n. 12, p. 1870–1891, 1992. 2

[Henriet *et al.* 2017] HENRIET, S. *et al.* Synthetic dataset generation for non-intrusive load monitoring in commercial buildings. In: **Proceedings of the 4th ACM International Conference on Systems for Energy-Efficient Built Environments**. [S.l.: s.n.], 2017. p. 1–2. 11

[JZ 2011] JZ, K. Redd: a public data set for energy disaggregation research. In: **Proceedings of the 1st KDD workshop on data mining applications in sustainability (sustKDD)**, ACM. San Diego, CA. [S.l.: s.n.], 2011. 12

[Kelly e Knottenbelt 2014] KELLY, J.; KNOTTENBELT, W. Metadata for energy disaggregation. In: **2014 IEEE 38th International Computer Software and Applications Conference Workshops**. IEEE, 2014. Disponível em: <<https://doi.org/10.1109/2Fcompsacw.2014.97>>. 8

[Kelly e Knottenbelt 2015] KELLY, J.; KNOTTENBELT, W. Neural nilm: Deep neural networks applied to energy disaggregation. In: **Proceedings of the 2nd ACM international conference on embedded systems for energy-efficient built environments**. [S.l.: s.n.], 2015. p. 55–64. 2

[Kelly e Knottenbelt 2015] KELLY, J.; KNOTTENBELT, W. The uk-dale dataset, domestic appliance-level electricity demand and whole-house demand from five uk homes. **Scientific data**, Nature Publishing Group, v. 2, n. 1, p. 1–14, 2015. 15

[Kingma e Ba 2014] KINGMA, D. P.; BA, J. Adam: A method for stochastic optimization. **arXiv preprint arXiv:1412.6980**, 2014. 19

[Klemenjak *et al.* 2019] KLEMENJAK, C. *et al.* Electricity consumption data sets: Pitfalls and opportunities. In: **Proceedings of the 6th ACM international conference on systems for energy-efficient buildings, cities, and transportation**. [S.l.: s.n.], 2019. p. 159–162. 13

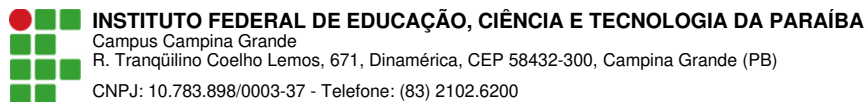
- [Lin 1991] LIN, J. Divergence measures based on the shannon entropy. **IEEE Transactions on Information theory**, IEEE, v. 37, n. 1, p. 145–151, 1991. 16
- [Luca *et al.* 2022] LUCA, T. *et al.* elami—an innovative simulated dataset of electrical loads for advanced smart energy applications. **IEEE Access**, IEEE, v. 10, p. 91177–91191, 2022. 12
- [MacKay, Kay *et al.* 2003] MACKAY, D. J.; KAY, D. J. M. *et al.* **Information theory, inference and learning algorithms**. [S.l.]: Cambridge university press, 2003. 16
- [Makonin e Popowich 2015] MAKONIN, S.; POPOWICH, F. Nonintrusive load monitoring (nilm) performance evaluation. **Energy Efficiency**, Springer, v. 8, n. 4, p. 809–814, 2015. 10
- [Makonin *et al.* 2015] MAKONIN, S. *et al.* Exploiting hmm sparsity to perform online real-time nonintrusive load monitoring. **IEEE Transactions on smart grid**, IEEE, v. 7, n. 6, p. 2575–2585, 2015. 16
- [Monacchi *et al.* 2014] MONACCHI, A. *et al.* Greend: An energy consumption dataset of households in italy and austria. In: IEEE. **2014 IEEE International Conference on Smart Grid Communications (SmartGridComm)**. [S.l.], 2014. p. 511–516. 14, 15
- [Monacchi *et al.* 2017] MONACCHI, A. *et al.* An open solution to provide personalized feedback for building energy management. **Journal of Ambient Intelligence and Smart Environments**, IOS Press, v. 9, n. 2, p. 147–162, 2017. 14
- [Murray *et al.* 2015] MURRAY, D. *et al.* A data management platform for personalised real-time energy feedback. IET, 2015. 15
- [Nambi, Lua e Prasad 2015] NAMBI, A. S. U.; LUA, A. R.; PRASAD, V. R. Locoed: Location-aware energy disaggregation framework. In: **Proceedings of the 2nd acm international conference on embedded systems for energy-efficient built environments**. [S.l.: s.n.], 2015. p. 45–54. 15
- [Oord *et al.* 2016] OORD, A. v. d. *et al.* Wavenet: A generative model for raw audio. **arXiv preprint arXiv:1609.03499**, 2016. xii, 6, 7
- [Picon *et al.* 2016] PICON, T. *et al.* Cooll: Controlled on/off loads library, a public dataset of high-sampled electrical signals for appliance identification. **arXiv preprint arXiv:1611.05803**, 2016. 5
- [Ridi, Gisler e Hennebert 2014] RIDI, A.; GISLER, C.; HENNEBERT, J. A survey on intrusive load monitoring for appliance recognition. In: IEEE. **2014 22nd international conference on pattern recognition**. [S.l.], 2014. p. 3702–3707. 2
- [Rossum e Jr 1995] ROSSUM, G. V.; JR, F. L. D. **Python reference manual**. [S.l.]: Centrum voor Wiskunde en Informatica Amsterdam, 1995. 5
- [Shin *et al.* 2019] SHIN, C. *et al.* Data requirements for applying machine learning to energy disaggregation. **Energies**, MDPI, v. 12, n. 9, p. 1696, 2019. 13
- [Thorve *et al.*] THORVE, S. *et al.* Fidelity and diversity metrics for validating hierarchical synthetic data: Application to residential energy demand. 11

[Tran *et al.* 2021] TRAN, M.-Q. *et al.* Experimental setup for online fault diagnosis of induction machines via promising iot and machine learning: Towards industry 4.0 empowerment. **IEEE access**, IEEE, v. 9, p. 115429–115441, 2021. [1](#)

[Wang *et al.* 2018] WANG, Y. *et al.* Review of smart meter data analytics: Applications, methodologies, and challenges. **IEEE Transactions on Smart Grid**, IEEE, v. 10, n. 3, p. 3125–3148, 2018. [15](#)

[Yu e Koltun 2015] YU, F.; KOLTUN, V. Multi-scale context aggregation by dilated convolutions. **arXiv preprint arXiv:1511.07122**, 2015. [7](#)

[Zoha *et al.* 2012] ZOHA, A. *et al.* Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey. **Sensors**, Molecular Diversity Preservation International (MDPI), v. 12, n. 12, p. 16838–16866, 2012. [2](#)



Documento Digitalizado Ostensivo (Público)

TCC Victor Andrade Perone

Assunto: TCC Victor Andrade Perone
Assinado por: Victor Perone
Tipo do Documento: Tese
Situação: Finalizado
Nível de Acesso: Ostensivo (Público)
Tipo do Conferência: Cópia Simples

Documento assinado eletronicamente por:

- **Victor Andrade Perone, ALUNO (201711250013) DE BACHARELADO EM ENGENHARIA DE COMPUTAÇÃO - CAMPINA GRANDE**, em 11/02/2023 11:28:00.

Este documento foi armazenado no SUAP em 11/02/2023. Para comprovar sua integridade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/verificar-documento-externo/> e forneça os dados abaixo:

Código Verificador: 743610
Código de Autenticação: 7850d187fd

