

Redes Neurais Artificiais Recorrentes no Auxílio do Controle de Consumo e Eficiência Energética

Mateus Dal-Bó Tonelli, Saulo Popov Zambiasi

Ciências da Computação – Universidade do Sul de Santa Catarina (UNISUL)
Caixa Postal 787 – 88704-900 – Tubarão – SC – Brazil

{mateustonelli4, saulopz}@gmail.com

Abstract. *The use of artificial intelligence is becoming more and more commercially common given the possibility of advances in computational power and also in the academic popularization of the subject. In this context, this paper presents an energy management system with the aid of neural networks, which is one of the areas that artificial intelligence addresses.*

Resumo. *O uso de inteligência artificial vem se tornando cada vez mais comum comercialmente dado a possibilidade pelo avanço no poder computacional e também na popularização acadêmica do assunto, com isso cada vez mais percebe-se termo inteligência artificial junto das mais diversas aplicações presentes no mercado. Nesse contexto, o presente artigo apresenta um sistema de gestão de energia com o auxílio de redes neurais que é uma das áreas que a inteligência artificial aborda.*

1. Introdução

A preocupação com o meio ambiente vem recebendo cada vez mais importância levando em consideração o custo ambiental na criação de hidrelétricas ou os efeitos nocivos ao meio ambiente da queima de combustíveis fósseis, criando a conscientização do uso excessivo de aparelhos eletrodomésticos e também gerando políticas de investimento em eficiência energética, cuja meta é melhorar a eficiência de aparelhos eletrodomésticos. Pode-se citar que uma maneira de minimizar a crescente demanda de energia é investir na eficiência energética. No Brasil, o Ministério de Minas e Energia criou o Programa Nacional de Conservação de Energia Elétrica (Procel) para este fim. Medidas já foram tomadas, como o lançamento do selo Procel, por exemplo. Com ele, o consumidor tem fácil acesso às informações relacionadas à eficiência energética de certos eletrodomésticos, o que incentiva as empresas a fabricarem produtos mais eficientes, diminuindo a energia consumida pela população. Além disso, a Procel disponibiliza um site com informações e orientações, tanto para a população quanto às indústrias, economizarem energia elétrica [Procel 2021].

A área da tecnologia da informação, então, pode contribuir para o apoio do controle de consumo utilizando de sistemas de gestão da quantidade do volume de energia em kilowatt-hora, ou kWh de cada aparelho que um consumidor possui em sua residência, dando assim autonomia para o consumidor poder tomar atitudes que contribuam para a redução do consumo excessivo. Além do sistema de gestão, também pode-se utilizar de técnicas de inteligência artificial para traçar comportamentos de consumo excessivo e sugerir melhores comportamentos para o usuário deste sistema. Para utilizar inteligência artificial na gestão de energia é necessário a utilização de aparelhos que conseguem medir seu consumo, estes dispositivos podem ser de utilização

manual onde o usuário mede e anota os dados obtidos ou conectados à rede permitindo automação da captação dos dados.

Nesse contexto, o presente trabalho propõe um protótipo de sistema de gestão de consumo de energia elétrica. Com facilitador no processo de análise para o gerenciamento do consumo é definido um módulo de inteligência artificial usando Redes Neurais Artificiais.

O artigo está organizado da seguinte forma. O capítulo 1 apresenta a introdução, problemática e proposta, o capítulo 2 apresenta uma breve passagem a evolução da inteligência artificial até o nascimento das redes neurais, e mostra a evolução das redes neurais com o surgimento das redes neurais recursivas dada a necessidade e as limitações das redes neurais artificiais, o capítulo 3 apresenta trabalhos que buscaram como este encontrar maneiras de auxiliar na economia de recursos elétricos utilizando de técnicas de inteligência artificial, no capítulo 4 é apresentado a fase de implementação, onde aparece todos os recursos utilizados para o desenvolvimento do projeto, no capítulo 5 é apresentado os testes e o resultado alcançado, e por fim no capítulo 6 é apresentado a conclusão do autor sobre os resultados alcançados com os recursos utilizados.

2. Redes Neurais Artificiais

É difícil definir o que é inteligência artificial, pois a mesma abrange muitas áreas de aplicação. Na tradução do livro Inteligência Artificial apresentado por [Russel e Norvig 2013], a IA pode ser dividida em quatro abordagens, sendo elas, pensando como um humano, pensando racionalmente, agindo como seres humanos e agindo racionalmente.

A inteligência artificial (IA) é um dos campos mais recentes em ciências e engenharia. O trabalho começou logo após a Segunda Guerra Mundial, e o próprio nome foi criado em 1956 [Russel e Norvig 2013]. Nessa década e na seguinte a IA obteve diversos sucessos, mas foi limitada pelos sistemas computacionais primitivos, os sistemas de Inteligência Artificial dependem diretamente de sistemas computacionais e tiveram como principal benefício a evolução destes sistemas nas últimas décadas, e utilizando deste poder computacional é possível o processamento uma quantidade de dados grandes em segundos, e quantas vezes forem necessárias, tornando mais fácil uma atividade que levaria horas ou dias de um, ou mais ser humano [Damaceno e Vasconcelos 2018].

Ainda no início da década de 50 a IA tinham suas limitações em aplicações, podendo apenas resolver quebra-cabeças, porém no ano de 1958 John McCarthy desenvolveu a primeira linguagem de alto nível “Lisp” que se tornou a linguagem de programação que predominou em estudos de IA nos 30 anos seguintes. Tendo isto em consideração, podemos considerar que a inteligência artificial também depende diretamente da área de ciência da computação, uma vez que a mesma forneceu recursos para sua existência, como os sistemas operacionais, linguagens de programação junto de frameworks modernos que auxiliam na resolução de problemas complexos sem ter que desenvolver todo o algoritmo de IA. Entre as décadas de 50 e 60 a IA passou por uma grande evolução dentro de diversos estudos que tinham como princípio tanto melhorar o desempenho, quanto buscar a autonomia da mesma para não apenas resolver problemas

específicos, porém apenas nos anos 80 surgiu então surgiu o primeiro sistema comercial de sucesso, o R1 como era chamado auxiliava na configuração de pedidos de novos sistemas de computadores, fazendo a empresa economizar cerca de 40 milhões de dólares por ano, tornando popular o uso de IA na maioria das empresas importantes dos Estados Unidos, que tinham seus próprios setores de uso ou estudo de sistemas especialistas [Russel e Norvig 2013].

Com a evolução da IA nascem as redes neurais que têm como teoria imitar o comportamento do cérebro humano, a ideia de uma rede neural é executar uma tarefa através do processo de aprendizagem. De forma simples, uma rede neural é um sistema projetado para agir da maneira que o cérebro de um ser humano executa uma tarefa específica [Haykin 2009].

Uma rede neural consiste em elementos processadores, comumente chamados de neurônios, células ou nodos, cada um deles conectados entre si com um peso determinado, o peso representa a informação utilizada pela rede para resolver problemas propostos (especialmente problemas não lineares) [Fausset 1993].

As redes neurais, porém apresentam limitações e desafios alguns deles são a necessidade de grandes fontes de dados de qualidade para treinamento, quando muito profundas exigem grande poder computacional para processamento e overfitting que é o termo utilizado quando uma rede neural fica viciada aos dados de treinamento e não traz bons resultados [Pessoa 2023].

Tendo como problema das redes neurais ser o processamento de dados sequenciais surge então as redes neurais recorrentes que resolvem este problema mudando a maneira como ela trafega as informações entre os neurônios, em uma rede neural simples os dados de um neurônio do ciclo anterior não são transmitidos para o neurônio seguinte, na rede neural recorrente, sim, aumentando sua capacidade de adaptação. Mesmo sendo uma evolução das redes neurais as redes neurais recorrentes ainda sofrem com a explosão e dissipação de gradientes, para solucionar este problema surge a arquitetura Long Short-Term Memory (LSTM), que permite que uma rede neural recorrente aprenda muito mais durante os ciclos, a evolução da arquitetura é o aumento da capacidade de armazenamento de informação por meio de portas, as LSTM tem 3 portas, uma porta de saída para controlar o que entra no neurônio, uma porta de entrada para gravar a informação no neurônio e uma porta de esquecimento para limpar a informação do neurônio, cada uma delas possui uma função [Schmidt 2019].

Na figura 1 é possível observar a representação das 3 portas explicados anteriormente que decidem o que pode entrar, o que pode ser gravado e o que deve ser esquecido.

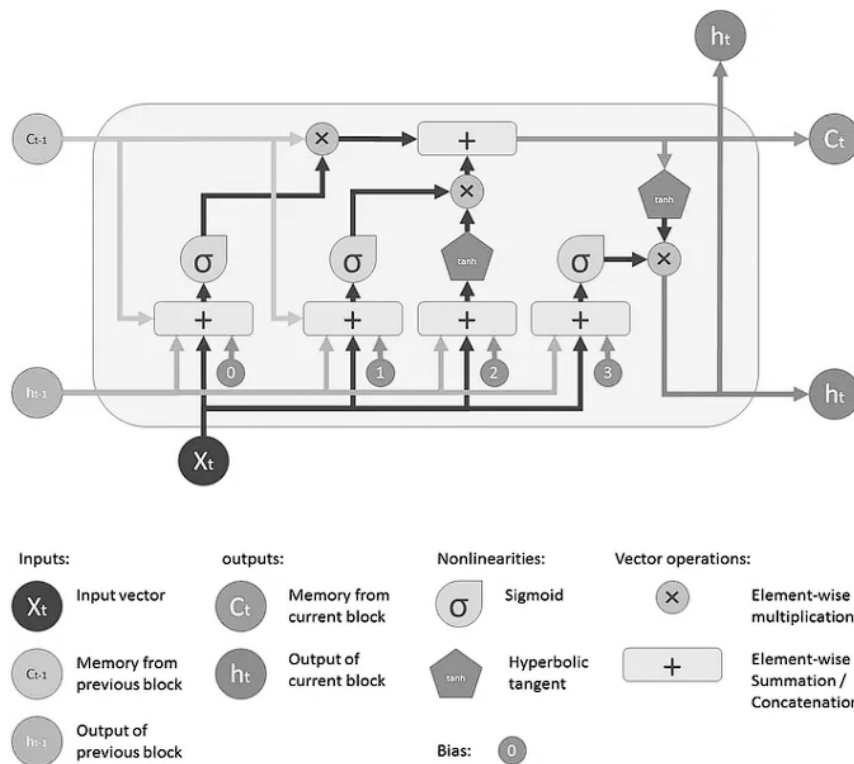


Figura 1: Representação de uma unidade de memória de uma LSTM. Fonte: (YAN, 2016)

Na figura 2 é C_{t-1} representa a memória da unidade de memória anterior, que passa pelo nodo “X” que representa a porta de esquecimento que é basicamente uma multiplicação de vetores, caso o vetor de esquecimento seja próximo a zero essa memória da unidade de memória antiga é esquecida, caso não o resultado passa pelo nodo “+” que representa a soma da memória da unidade anterior com uma memória nova adquirida, após estas duas operações a memória da unidade anterior representada por C_{t-1} vira uma nova memória representada por C_t [Yan 2016].

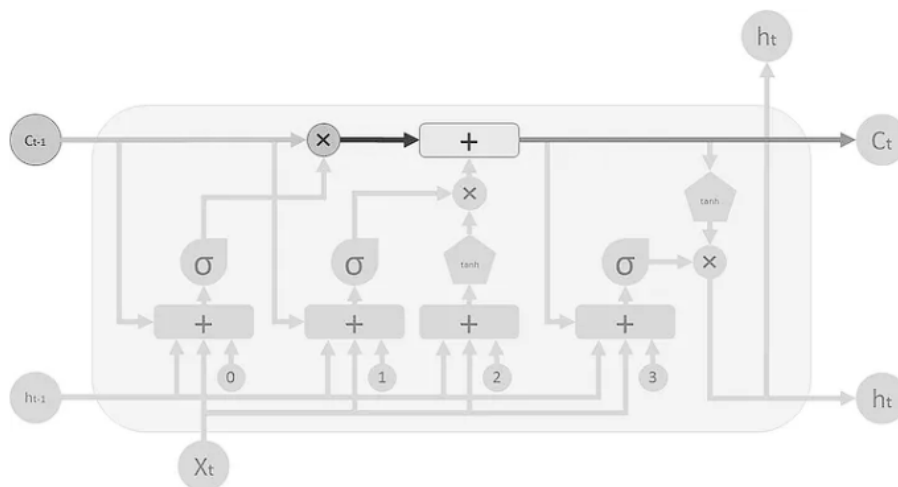


Figura 2: Representação da entrada da memória de uma unidade de memória antiga. Fonte: (YAN, 2016)

Na figura 3 é apresentado a função da porta de esquecimento que nada mais é que uma rede neural de uma camada a entrada desta rede é h_{t-1} sendo a saída da unidade de memória anterior, C_{t-1} sendo a entrada da unidade de memória atual, X_t sendo a entrada da unidade de memória atual e o peso, essa rede neural tem uma função de ativação sigmóide e o vetor resultado é a porta de esquecimento que será aplicada na memória anterior C_{t-1} através da multiplicação de elementos [Yan 2016].

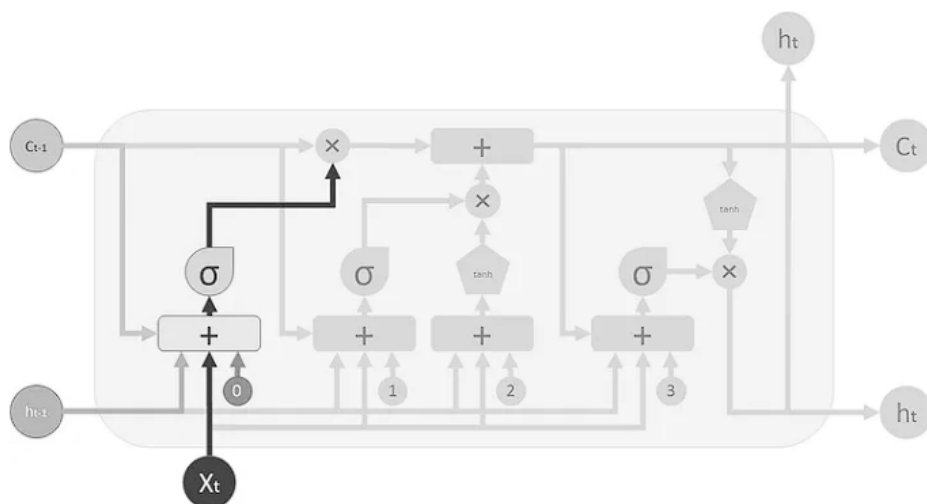


Figura 3: Representa o porta de esquecimento de uma unidade de memória.
Fonte: (YAN, 2016)

A figura 4 é apresentado a função da porta de entrada com dois caminhos o primeiro que controlará o quanto da memória antiga influenciará na memória nova, recebe basicamente as mesmas entradas da porta de esquecimento, e o segundo caminho que processa a memória nova por meio de uma função de ativação hiperbólica, o resultado dos dois caminho é unido por meio de uma multiplicação de elementos no final dos caminhos o resultado da multiplicação é somado a memória antiga para criação de uma nova memória [Yan 2016].

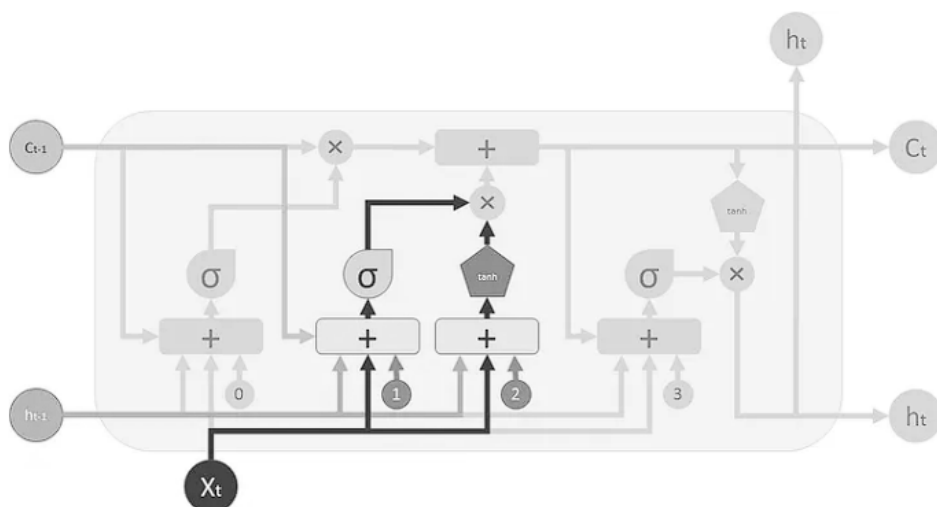


Figura 4: representação da porta de entrada de uma unidade de memória.
Fonte: (YAN, 2016)

A figura 5 é apresentado a função da porta de saída cuja função é controlar o quanto da nova memória deve ser replicado para a próxima unidade de memória, nesta etapa o valor da nova memória gerada pelas portas anteriores é processado com a entrada da unidade de memória anterior, com a entrada da unidade de memória atual e um peso para gerar uma saída para a próxima unidade de processamento (Yan 2016).

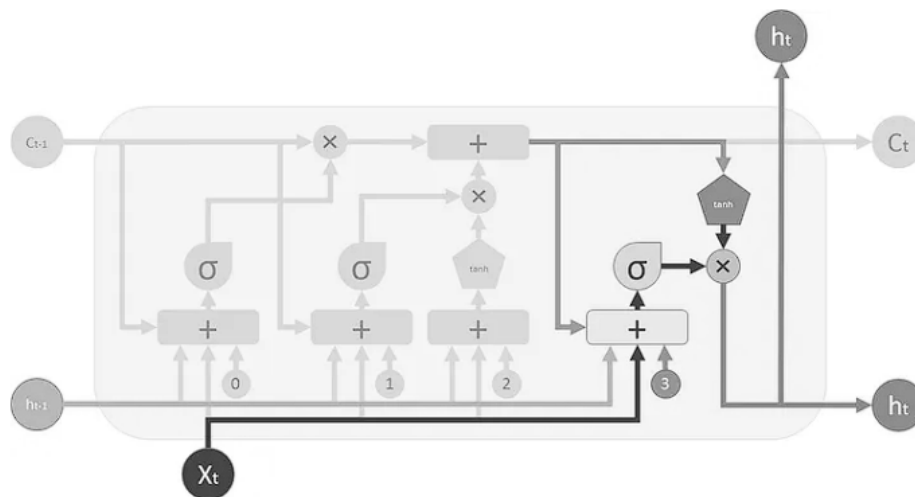


Figura 5: Representação da porta de saída de uma unidade de memória. Fonte: (YAN 2016)

3. Estado da Arte de Trabalhos de Economia de Energia Elétrica

O artigo de Luz et al. (2016) apresenta a utilização de Algoritmos Genéticos visando melhorar a operação de uma estação de tratamento de água. Os autores introduzem explicando que Estações de Tratamento de Água (ETA) consomem cerca de três por cento da energia nacional, e desses três por cento, noventa por cento são do consumo das bombas, e explica que se definindo regras de operação para estas bombas as mesmas podem economizar energia. O Algoritmo Genético aplicado no contexto deste projeto foi utilizado para reduzir o custo com energia do sistema de bombeamento de água, tendo o problema em vista, foi definido uma função objetivo, onde é calculado o consumo diário de cada motor em kWh como podemos ver a seguir:

$$Z = \sum_{t=1}^{24} [441 * T(t) * X_1(t) + 331 * T(t) * X_2(t) + 441 * T(t) * X_3(t)]$$

Então foi definido que o indivíduo é composto por um conjunto de genes de tamanho igual a 24, representando as horas do dia, o gene tem 3 alelos representando os motores conforme Figura 6:

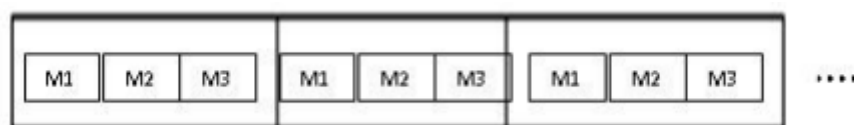


Figura 6: Cromossomo representando o indivíduo. Fonte: (Luz et al. 2016)

Tendo isso definido foi proposto o Algoritmo Genético a ser utilizado conforme o esquema representado na Figura 7.

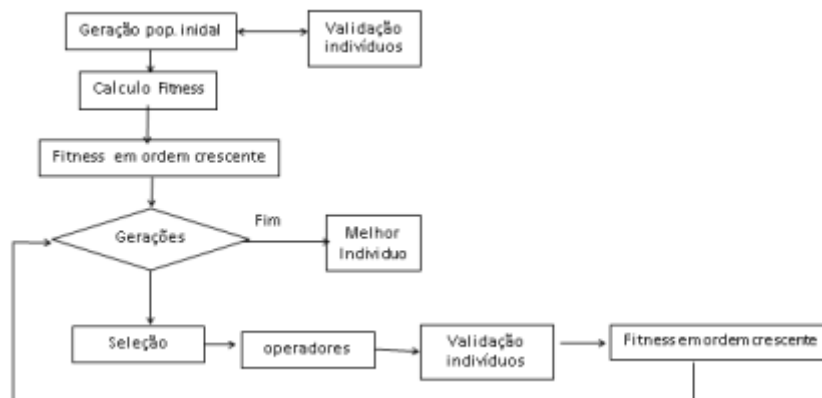


Figura 7: Esquema do algoritmo genético implementado. Fonte: (Luz et al. 2016)

Por fim, os autores afirmam que com o modelo proposto foi atingido um resultado de economia de aproximadamente 18% quando comparado com o resultado médio mensal informado pela companhia que foi analisada. Assim como no trabalho correlato apresentado o objetivo deste artigo é buscar economia através do auxílio de técnicas de inteligência artificial, mais precisamente algoritmos genéticos, no caso deste trabalho foram analisadas as bombas de uma estação de tratamento, neste artigo analisaremos o consumo somado de aparelhos e prever os consumos futuros [Luz et al. 2016].

O artigo [Nazário et al. 2017] propôs a redução do consumo de energia de uma residência utilizando a técnica de algoritmos genéticos. No artigo os autores utilizam recursos computacionais para simulação da oferta e da demanda de energia de uma residência, além disso, fizeram uma pesquisa de campo para obter o consumo de energia de aparelhos de refrigeração, iluminação e chuveiros em residências, considerando a quantidade e a estimativa de uso diário durante um mês. Com o estudo feito, os autores concluíram que o sistema resultou na probabilidade de dez por cento de economia na fatura.

Em Silva (2016) o autor buscou gerar comportamentos de economia de energia por meio de um sistema que gerencia o consumo e que alerta os usuários sobre o consumo inadequado da energia elétrica de sua residência utilizando o auxílio da técnica de redes neurais artificiais. Naquele trabalho, foi criado um hardware que recebe a informação dos aparelhos que serão controlados e dentro deste aparelho tem um software que já processa as entradas de informações pela rede neural artificial, o aparelho de monitoramento indicam a potência do aparelho que está sendo analisado por leds, e como explicado no artigo foi analisando um chuveiro elétrico o aparelho também emite sinais sonoros caso o uso passe de um tempo estipulado, a rede neural entra na parte de sugestão de uso ideal onde, por exemplo, se o usuário do chuveiro está em uma potência não ideal baseado na temperatura do ambiente, o aparelho notifica o usuário para usar uma potência que seria a ideal para ter a melhor eficiência no consumo de energia elétrica. Os autores concluíram que com o sistema foi possível uma economia

média de até trinta e seis por cento se o usuário seguir todas as recomendações do aparelho desenvolvido.

4. Implementação

A fonte de dados utilizada é um arquivo de texto contendo uma série temporal do consumo elétrico de uma residência, a base possuía valores vazios que precisavam ser substituídos pela média dos valores da propriedade que eles representavam para não afetar a predição.

Para implementação do algoritmo foi escolhida a linguagem de programação Python¹ devido a ampla disponibilidade de bibliotecas e frameworks que auxiliam no desenvolvimento quando o assunto envolve inteligência artificial, além das seguintes bibliotecas:

- **Keras**²: a biblioteca foi escolhida para facilitar a implementação e o treinamento da rede neural, se trata de uma biblioteca de alto nível que permite a criação de modelos de redes neurais de forma simples abstraindo qualquer detalhe de baixo nível, no projeto configuramos uma rede neural recorrente com o Keras iniciando um modelo sequencial utilizando a função de perda como erro quadrático médio adequada para problemas de regressão e utilizamos o otimizador adam que é amplamente utilizado em problemas de aprendizado profundo, ele possui uma taxa de aprendizado adaptativa e ajusta a sua taxa de aprendizado automaticamente durante a etapa de treino permitindo que a taxa de aprendizado seja maior para parâmetros comuns e menor para parâmetros menos frequentes, ele também possui um mecanismo de regularização que ajuda a evitar o overfitting;
- **NumPy**³: a biblioteca foi escolhida para parte de computação numérica permitindo a fácil manipulação de matrizes e também tem disponível funções matemáticas de alto desempenho abstraídas no projeto foi utilizada para a concatenação de matrizes na remoção das transformações de escala para processamento da rede neural, e também no cálculo do erro quadrático médio presente na figura 9;
- **Pandas**⁴: a biblioteca foi escolhida, pois trata a parte de manipulação de dados com alto desempenho, no algoritmo foi utilizada para o carregamento dos dados de um arquivo de texto onde continha a fonte de dados utilizada no treinamento e teste da rede neural para um DataFrame que é uma estrutura de dados multidimensional, a mesma também permite a manipulação de dados dentro do DataFrame de forma simples.

Na Figura 8 é possível observar uma pequena representação dos dados originais onde aparecem as informações vazias representadas pelo caractere de interrogação, no algoritmo utilizamos as funções manipulação de dados para

¹ <https://www.python.org/>

² <https://keras.io/>

³ <https://numpy.org/>

⁴ <https://pandas.pydata.org/>

tratar esses valores substituindo os mesmo pela soma da média de cada coluna respectiva, deixando o tratado para as fases seguintes.

```
21/12/2006;11:22:00;0.244;0.000;242.290;1.000;0.000;0.000;0.000
21/12/2006;11:23:00;?;?;?;?;?;?;?;
21/12/2006;11:24:00;?;?;?;?;?;?;?;
21/12/2006;11:25:00;0.246;0.000;241.740;1.000;0.000;0.000;0.000
21/12/2006;11:26:00;0.246;0.000;241.830;1.000;0.000;0.000;0.000
21/12/2006;11:27:00;0.244;0.000;240.960;1.000;0.000;0.000;0.000
```

Figura 8: Demonstração dos dados vazios em uma fração da fonte de dados. Fonte: Autor.

Na Figura 9 mostra a quantidade que haviam de valores vazios no DataFrame antes e depois tratamento feito;

```
Global_active_power      25979
Global_reactive_power    25979
Voltage                  25979
Global_intensity         25979
Sub_metering_1           25979
Sub_metering_2           25979
Sub_metering_3           25979
dtype: int64
Global_active_power      0
Global_reactive_power    0
Voltage                  0
Global_intensity         0
Sub_metering_1           0
Sub_metering_2           0
Sub_metering_3           0
dtype: int64
```

Figura 9: Representação de antes e depois da soma da quantidade de valores vazios na fonte de dados. Fonte: Autor.

- **Matplotlib⁵:** a biblioteca é uma biblioteca utilizada para visualização de dados por meio de funções simplificadas é possível personalizar e imprimir gráficos de alta qualidade, no projeto foi utilizada para representar a distribuição dos dados originais, a correlação das propriedades da fonte de dados que aparece na Figura 11, o gráfico de perdas presente na Figura 13 e a análise predição da rede neural presente na figura 10.

A Figura 10 é o resultado da impressão da fonte de dados utilizada no processamento da rede neural onde foi utilizado a biblioteca Matplotlib, com poucas linhas de código conseguimos gerar um gráfico de todas as propriedades da fonte de dados para uma análise de decisão;

⁵ <https://matplotlib.org/>

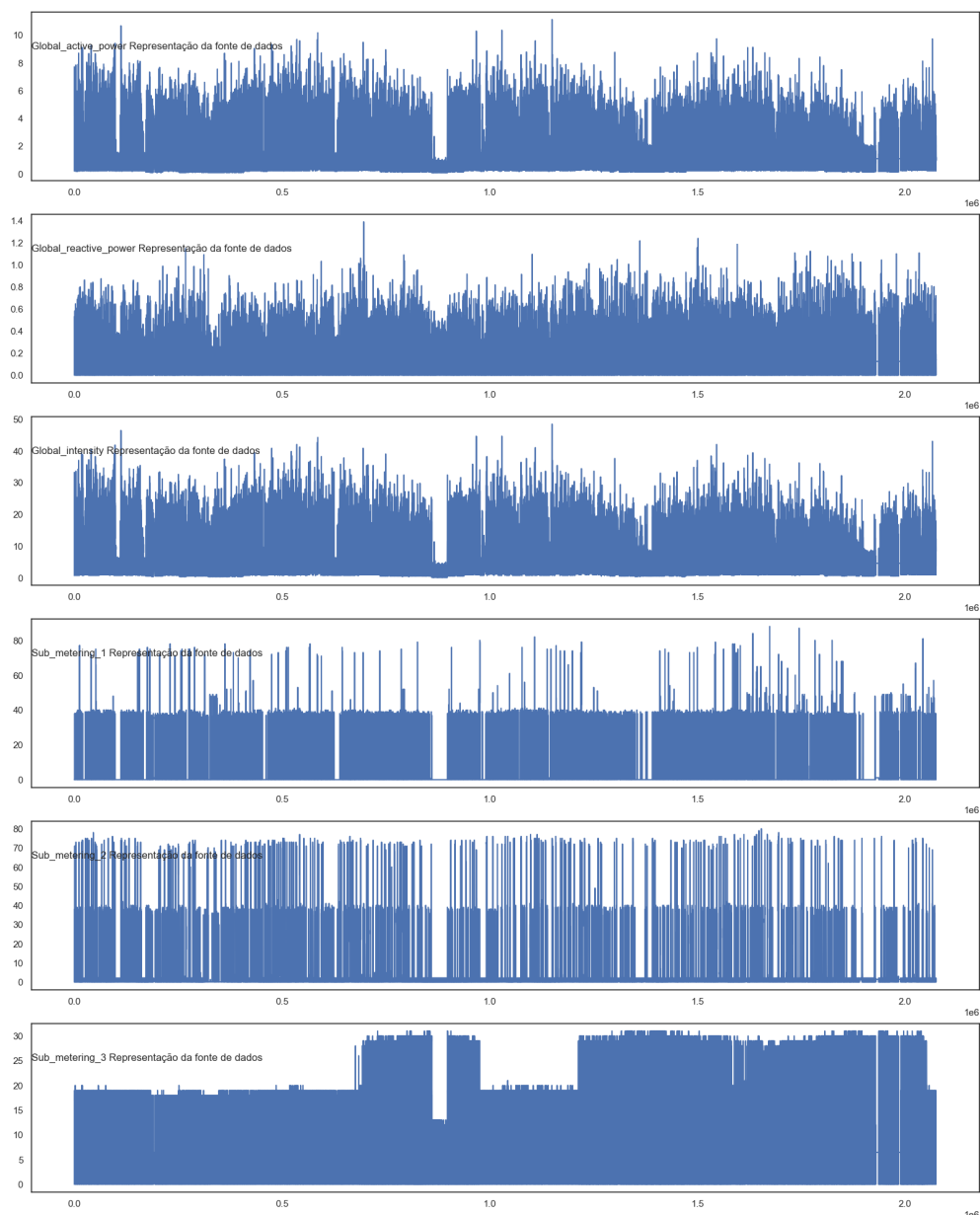


Figura 10: Gráfico apresentando a distribuição dos dados da fonte de dados.
Fonte: Autor.

- **scikit-learn**⁶: a biblioteca tem como função principal aprendizado de máquina e mineração de dados, mas também oferece uma grande quantidade de algoritmos para avaliação e construção de modelos, no projeto foi utilizada no cálculo do erro quadrático, e também para normalização das informações da fonte de dados;
- **seaborn**⁷: a biblioteca tem como função visualização de dados junto da biblioteca Matplotlib, no projeto foi utilizada junto da Matplotlib para geração da apresentação da correção dos parâmetros que aparece na Figura 11;

⁶ <https://scikit-learn.org/stable/index.html>

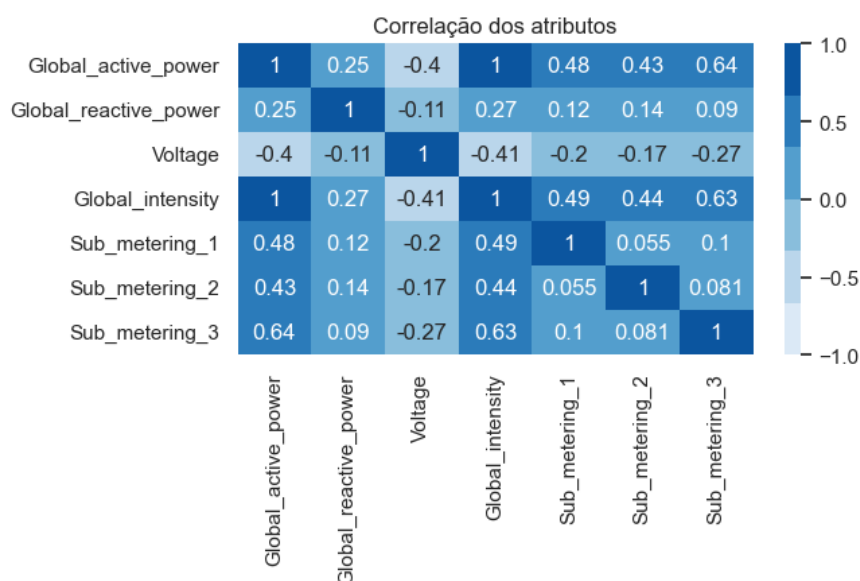


Figura 11: Tabela de correlação das propriedades da fonte de dados. Fonte: Autor.

5. Testes e Validação

Configuração do Computador utilizada na execução do algoritmo:

- Processador: Ryzen 7 5800X;
- Memória RAM: 16 GB DDR4;
- Placa de Vídeo: AMD RX6800;
- Sistema Operacional: Windows 10 Pro 64 bits;

Para o teste foi necessário transformar os dados que estão no formato de série temporal, para dados de aprendizado supervisionado na transformação também foi feita a normalização dos dados, analisando a correlação dos parâmetros presente na Figura 11 foi decidido remover a propriedade Voltage, pois a mesma apresenta pouca relação com o resto das outras propriedades, não sendo vantajoso utilizar ela, a rede neural recursiva foi configurada com uma camada LSTM, uma camada de dropout e uma camada densa de saída, a função de perda utilizada foi o erro quadrático médio e a otimizador utilizado foi o adam já explicados no capítulo de desenvolvimento.⁷

Na Figura 12 é possível observar as etapas impressas pelo Keras, a rede foi configurada para rodar 50 épocas conforme apresentado na figura, em cada etapa a biblioteca mostra a evolução na perda considerando a fração da base de treino e considerando a base de validação, o valor é representa o erro quadrático médio de cada umas das perdas respectivamente.

⁷ <https://seaborn.pydata.org/>

```

Epoch 1/50
72/72 - 18s - loss: 0.0152 - val_loss: 0.0041 - 18s/epoch - 247ms/step
Epoch 2/50
72/72 - 16s - loss: 0.0054 - val_loss: 0.0023 - 16s/epoch - 223ms/step
Epoch 3/50
72/72 - 16s - loss: 0.0041 - val_loss: 0.0015 - 16s/epoch - 227ms/step
Epoch 4/50
72/72 - 16s - loss: 0.0033 - val_loss: 0.0012 - 16s/epoch - 225ms/step
Epoch 5/50
72/72 - 16s - loss: 0.0029 - val_loss: 0.0010 - 16s/epoch - 227ms/step
Epoch 6/50
72/72 - 16s - loss: 0.0026 - val_loss: 8.9970e-04 - 16s/epoch - 229ms/step
Epoch 7/50
72/72 - 16s - loss: 0.0024 - val_loss: 8.0094e-04 - 16s/epoch - 226ms/step
Epoch 8/50
72/72 - 16s - loss: 0.0022 - val_loss: 7.2956e-04 - 16s/epoch - 227ms/step
Epoch 9/50
72/72 - 16s - loss: 0.0021 - val_loss: 6.8504e-04 - 16s/epoch - 228ms/step
Epoch 10/50
72/72 - 16s - loss: 0.0021 - val_loss: 6.5420e-04 - 16s/epoch - 222ms/step
Epoch 11/50
72/72 - 16s - loss: 0.0020 - val_loss: 6.3166e-04 - 16s/epoch - 228ms/step
Epoch 12/50
72/72 - 16s - loss: 0.0020 - val_loss: 6.1850e-04 - 16s/epoch - 229ms/step
Epoch 13/50
...
Epoch 49/50
72/72 - 16s - loss: 0.0020 - val_loss: 5.9764e-04 - 16s/epoch - 220ms/step
Epoch 50/50
72/72 - 16s - loss: 0.0020 - val_loss: 5.9355e-04 - 16s/epoch - 220ms/step

```

Figura 12: Saída do terminal com as etapas de execução do modelo sendo treinado. Fonte: Autor.

Na Figura 13 é possível observar a apresentação da evolução da perda com a fração da base de ter e a fração da base de validação.

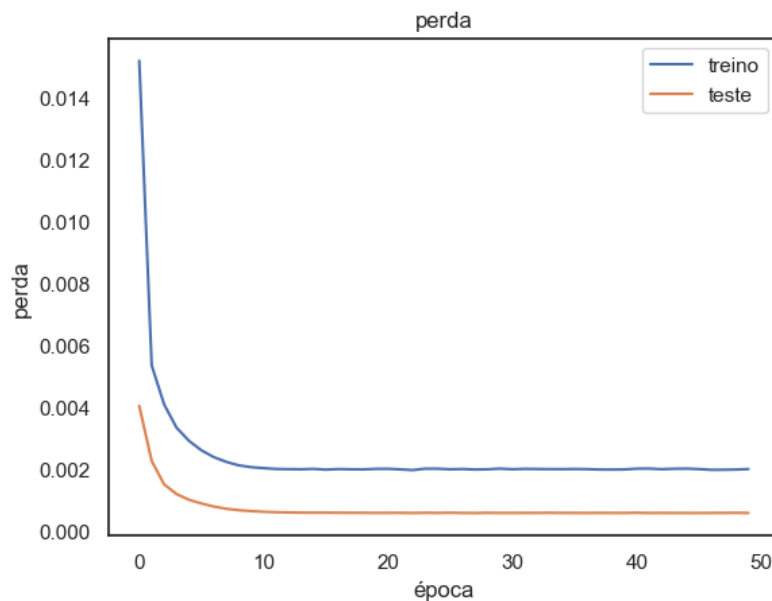


Figura 13: Gráfico apresentando a perda gradativamente diminuindo durante a execução do modelo LSTM. Fonte: Autor.

Na Figura 14 observa-se o erro quadrático médio entre a fração da base que foi utilizada para fazer o treino e a fração de dados da base que foi utilizada para validar a base.

```
64696/64696 [=====] - 34s 529us/step  
Teste RMSE: 0.269
```

Figura 14: Erro quadrático médio entre o treino e a validação. Fonte: Autor.

Após a execução do treino da rede neural foi executada a predição rede neural recursiva com os dados de toda a base para comparação dos dados originais da fonte de dados com os dados de predição o resultado pode ser visto na Figura 15.

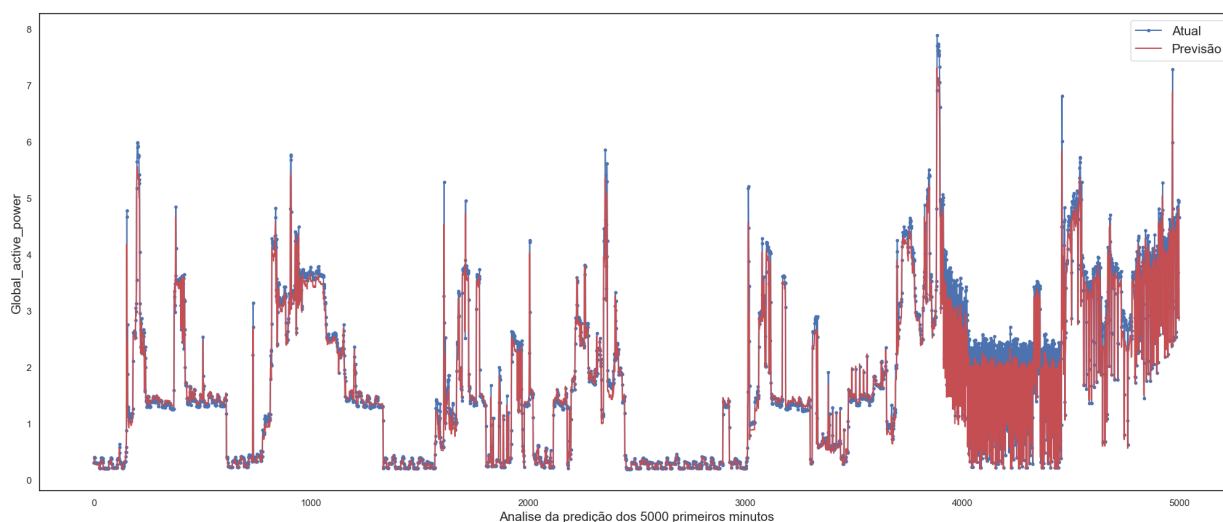


Figura 15: Gráfico comparando os dados reais em azul comparados com os dados previstos pela rede neural em vermelho. Fonte: Autor.

O maior desafio na implementação deste algoritmo descobrir os passos necessários para transformar os dados de série temporal em dados de aprendizado supervisionado, mas com poucas pesquisas foi encontraram funções prontas que já resolvem este problema, com a escolha da função foi estudado o objetivo da mesma para então utilizá-la no algoritmo, outro desafio encontrado foi a utilização da linguagem Python, mas também com algumas pesquisas em sites de ciência dos dados foi possível encontrar exemplos que pudessem contribuir para o aprendizado necessário para criação do algoritmo deste artigo.

6. Conclusão

Tendo em vista a busca pela redução do impacto ambiental causado pelo aumento no consumo elétrico este artigo propôs um software para auxiliar na redução do consumo de uma residência utilizando um algoritmo de predição do consumo ativo de aparelhos ligados a uma rede elétrica, no caso deste artigo foi analisado grupos de consumo onde cada um tinham um ou mais aparelhos ligados. O algoritmo é baseado na técnica de redes neurais recursivas, ideal para trabalhar com bancos de dados de série temporal.

Por meio de testes, o algoritmo mostrou-se consideravelmente eficaz, alcançando resultados muito próximos dos dados reais, tendo um valor de erro quadrático médio

entre a base de treino e teste relativamente baixo representando uma taxa baixa de diferença do modelo real para o previsto.

É possível também considerar a facilidade na obtenção do resultado considerando que foi utilizado um computador de uso doméstico na predição, mostrando que é possível encontrar respostas com algoritmos de inteligência artificial mesmo considerando cenários onde o(a) pesquisador(a) não tem acesso a recursos computacionais de alto valor.

Como próximos passos, propõe-se a criação de uma fonte de dados reais capturados por aparelhos eletrodomésticos que alimentam uma base constantemente, possibilitando que esses dados sejam utilizados por um programa de gerenciamento que um usuário comum consiga utilizar e o mesmo receba dicas de como usar os aparelhos de forma mais econômica.

Referencias

Procel. Programa Nacional de Conservação de Energia Elétrica - Procel. Disponível em: <<https://www.gov.br/mme/pt-br/assuntos/secretarias/spe/procel-programa-nacional-d-e-conservacao-de-energia-eletrica-1>>. Acesso em: Novembro de 2022.

Pessoa, E.. O que é Redes Neurais? Disponível em: <<https://pessoatech.com.br/o-que-e-redes-neurais/>>. Acesso em: Maio de 2023.

Yan, S.. Understanding LSTM and Its Diagrams. Disponível em: <<https://blog.mlreview.com/understanding-lstm-and-its-diagrams-37e2f46f1714>>. Acesso em: Maio de 2023.

Russel, S e Norvig, P... Inteligência Artificial. 3. ed. Tradução da Terceira Edição. São Paulo: Pearson Education do Brasil, 2013.

Damaceno, S. S. e Vasconcelos, R. O. Inteligência Artificial: Uma breve abordagem sobre seu conceito real e o conhecimento popular. Caderno de Graduação - Ciências Exatas e Tecnológicas - UNIT - SERGIPE, 2018.

Luz, E. C. et al.. Economia de Energia no Sistema de Abastecimento de Água: Uma Abordagem Usando Algoritmos Genéticos., 2016.

Nazário, A. G. et al. Automação Domótica Simulada Utilizando Algoritmo Genético Especializado na Redução do Consumo de Energia. Anais do Computer on the Beach, p. 180-189, 2017.

Silva, E. A. et al.. Controle e monitoramento para consumo eficiente de energia elétrica em uma smart home utilizando redes neurais artificiais. Trabalho de Conclusão de Curso. Engenharia da Computação. UFSC. 2017.

Fausett, L. V. et al. Fundamentals of Neural Networks: Architectures, Algorithms and Applications: United States Edition. 1st ed. United States: Pearson. 1993.

Haykin, S. et al. Neural Networks and Learning Machines. Pearson. 2009.

Schmidt, R. M.. Recurrent Neural Networks (RNNs):A gentle Introduction and Overview, Department of Computer Science Eberhard-Karls-University Tübingen, Tübingen, Germany, 2019.