

## **Predição de fluxo de potência usando redes neurais recorrentes com L.S.T.M. e algoritmo genético em um barramento IEEE 30**

**João Victor Cesario Fernandes**

Graduando em Engenharia Elétrica

Instituição: Instituto Federal do Triângulo Mineiro (IFTM)

E-mail: [d202210285@uftm.edu.br](mailto:d202210285@uftm.edu.br)

Orcid: <https://orcid.org/0009-0003-4515-4433>

Lattes: <http://lattes.cnpq.br/3845027875860177>

**Madeleine Rocío Medrano Castillo Albertini**

Pós-doutorado em Engenharia Elétrica

Instituição: Universidade Federal do Triângulo Mineiro (UFTM)

E-mail: [madeleine.albertini@uftm.edu.br](mailto:madeleine.albertini@uftm.edu.br)

Orcid: <https://orcid.org/0000-0003-4692-9205>

Lattes: <http://lattes.cnpq.br/0849523409183491>

**Jaqueline Oliveira Rezende**

Doutorado em Engenharia Elétrica

Instituição: Instituto Federal de Goiás (IFG)

E-mail: [jaqueline.rezende@ifg.edu.br](mailto:jaqueline.rezende@ifg.edu.br)

Orcid: <https://orcid.org/0000-0002-6697-1825>

Lattes: <http://lattes.cnpq.br/5450777657090154>

**Marcus Vinicius Borges Mendonça**

Doutorado em Engenharia Elétrica

Instituição: Universidade Federal do Triângulo Mineiro (UFTM)

E-mail: [marcus.mendonca@uftm.edu.br](mailto:marcus.mendonca@uftm.edu.br)

Orcid: <https://orcid.org/0000-0002-3185-0150>

Lattes: <http://lattes.cnpq.br/0172067411856936>

**Paulo Henrique Oliveira Rezende**

Doutorado em Engenharia Elétrica

Instituição: Universidade de Uberlândia (UFU)

E-mail: [paulohenrique16@gmail.com](mailto:paulohenrique16@gmail.com)

Orcid: <https://orcid.org/0000-0001-6156-7324>

Lattes: <http://lattes.cnpq.br/2338311941319569>

**Fabricio Augusto Matheus Moura**

Doutorado em Engenharia Elétrica

Instituição: Universidade de Uberlândia (UFU)

E-mail: [fabricio.moura@ufu.br](mailto:fabricio.moura@ufu.br)

Orcid: <https://orcid.org/0000-0002-7425-8018>

Lattes: <http://lattes.cnpq.br/8081704801120219>

**Arnaldo Jose Pereira Rosentino Junior**

Doutorado em Engenharia Elétrica

Instituição: Universidade Federal do Triângulo Mineiro (UFTM)

E-mail: [arnaldo.rosentino@uftm.edu.br](mailto:arnaldo.rosentino@uftm.edu.br)

Orcid: <https://orcid.org/0000-0001-6713-8395>

Lattes: <http://lattes.cnpq.br/5190070034823798>

**Antonio Cesar Costa Ferreira Rosa**

Mestrado

Instituição: Instituto Federal do Triângulo Mineiro (IFTM)

E-mail: [antoniorosa@iftm.edu.br](mailto:antoniorosa@iftm.edu.br)

Orcid: <https://orcid.org/0009-0008-5464-8984>

Lattes: <http://lattes.cnpq.br/3845027875860177>

## RESUMO

A previsão precisa das potências ativas (P) e reativas (Q) em redes elétricas é fundamental para aprimorar a confiabilidade operacional e o planejamento nos sistemas de potência modernos. Este artigo propõe um modelo de previsão baseado em redes neurais do tipo Long Short-Term Memory (LSTM) para estimar a demanda de carga em um sistema IEEE de 30 barras. O modelo considera como variáveis de entrada as potências ativas e reativas fornecidas pelos geradores, enquanto os componentes de potência das cargas são utilizados como variáveis-alvo de previsão. Para aprimorar o desempenho, foi empregado um Algoritmo Genético (GA) na otimização de hiperparâmetros, o que reduziu o Erro Absoluto Médio (MAE) e aumentou a precisão das previsões. Os resultados demonstram que a abordagem proposta fornece previsões estáveis do comportamento das cargas, evidenciando seu potencial de aplicação em smart grids e em sistemas de gerenciamento de microrredes.

**Palavras-chave:** Previsão de Fluxo de Potência. LSTM. Algoritmo Genético. Potência Ativa. Potência Reativa. Smart Grids.

## 1 INTRODUÇÃO

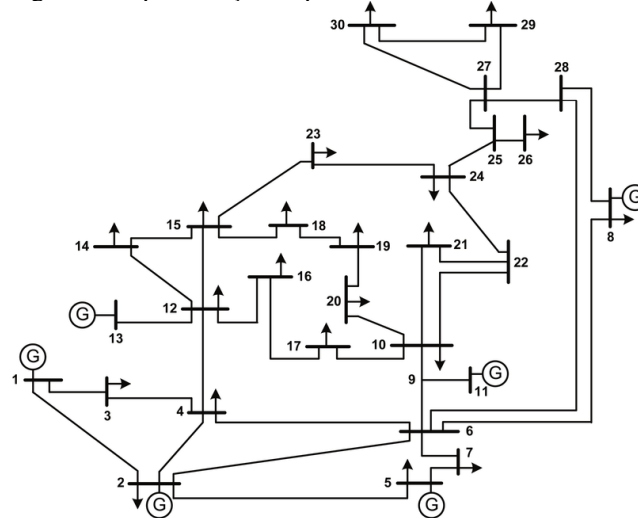
As crescente complexidade dos sistemas elétricos modernos, impulsionada pelo aumento da demanda e pela integração de fontes de energia renovável, intensificou a necessidade do uso de ferramentas computacionais avançadas para previsão de carga e análise do sistema. A estimativa precisa das potências ativa (P) e reativa (Q) é essencial para garantir a operação confiável, a estabilidade do sistema e a eficiência no planejamento energético.

As abordagens tradicionais de previsão, baseadas em métodos estatísticos ou lineares, apresentam limitações ao lidar com a natureza altamente não linear e dinâmica dos sistemas de potência. Nesse contexto, as Redes Neurais Recorrentes (RNNs), em especial as arquiteturas Long Short-Term Memory (LSTM), têm se destacado como alternativas eficazes devido à sua capacidade de capturar dependências temporais de longo prazo em dados históricos.

Neste trabalho, propõe-se o uso de uma rede LSTM para prever os componentes ativo e reativo de uma carga selecionada no sistema-teste IEEE de 30 barras, considerando como variáveis de entrada as

potências ativa e reativa fornecidas pelos geradores. Para melhorar ainda mais o desempenho do modelo, empregou-se um Algoritmo Genético (GA) para otimização dos hiperparâmetros da LSTM, possibilitando resultados mais consistentes com menores valores de Erro Absoluto Médio (MAE). A Figura 1 apresenta a representação esquemática do sistema IEEE de 30 barras utilizado como referência experimental. Esse sistema de teste é amplamente adotado na literatura como referência para a avaliação de metodologias de previsão e otimização em redes de energia elétrica.

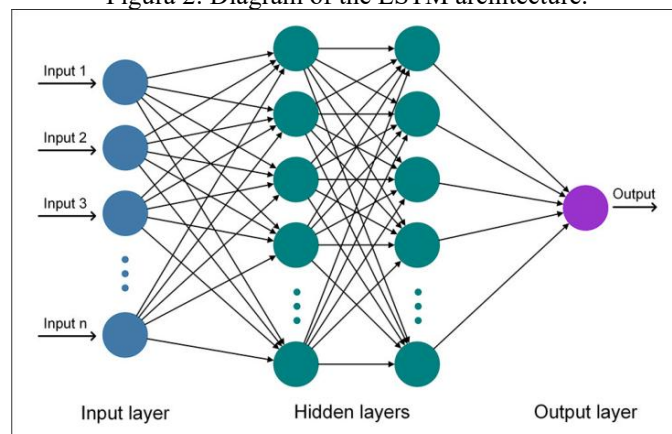
Figura 1: Representação esquemática do sistema IEEE de 30 barras



Fonte: Researchgate

A Figura 2 ilustra a arquitetura geral da rede LSTM proposta, destacando a sequência temporal das entradas (potências ativas e reativas dos geradores) e as saídas correspondentes às previsões das potências ativa (P) e reativa (Q) da carga selecionada.

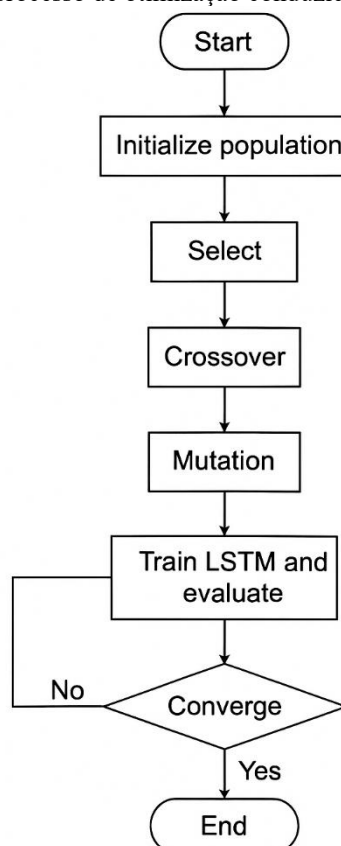
Figura 2: Diagram of the LSTM architecture.



Fonte: Researchgate

Por fim, a Figura 3 apresenta o fluxo do processo de otimização conduzido pelo Algoritmo Genético. O GA atua sobre os hiperparâmetros da LSTM (número de neurônios, taxa de aprendizado, número de camadas, entre outros), avaliando o desempenho com base no menor MAE obtido durante a validação.

Figura 3: fluxo do processo de otimização conduzido pelo Algoritmo Genético



Fonte: O autor

Neste cenário, o presente artigo propõe o desenvolvimento de um algoritmo baseado em redes neurais recorrentes (RNN) com unidades LSTM para a previsão do comportamento de cargas no sistema

IEEE de 30 barras, utilizando dados históricos de injeções de potência nas barras. As RNNs baseadas em LSTM se destacam pela capacidade de capturar padrões temporais complexos em dados sequenciais, aumentando a precisão das previsões mesmo em condições operacionais altamente dinâmicas. Trabalhos anteriores demonstraram que previsões precisas de carga impactam diretamente a eficiência da operação dos sistemas elétricos, especialmente quando integradas a estruturas de geração distribuída, permitindo a otimização de custos energéticos e o aumento da penetração de fontes renováveis na rede [3].

Para compreender o estado da arte e contextualizar os desafios técnicos envolvidos, realizou-se uma revisão de estudos prévios sobre previsão de séries temporais em sistemas elétricos e seus impactos no planejamento operacional. Em nossa análise, diferentes cenários de previsão foram testados utilizando dados reais de fluxo de potência combinados com abordagens de aprendizado de máquina. A capacidade de aprendizado das LSTM mostrou-se altamente eficaz na captura de dependências temporais, apresentando resultados promissores mesmo quando utilizado um conjunto reduzido de variáveis de entrada, como as potências ativa e reativa dos geradores, para prever a demanda ativa e reativa de cargas individuais. As previsões foram atualizadas a cada 10 instantes de tempo, simulando um horizonte operacional realista.

No contexto das aplicações industriais e comerciais, a previsão precisa de carga não apenas otimiza a operação de recursos distribuídos, mas também contribui para a estabilidade da rede elétrica em regiões com alta penetração de fontes renováveis. Estudos como [4] destacam o potencial das Redes Neurais Artificiais na previsão de perdas em redes de distribuição. De forma semelhante, a previsão de cargas com LSTM pode atuar como elemento central na redução de perdas técnicas em sistemas de distribuição quando integrada a estratégias avançadas de gerenciamento de carga e armazenamento de energia. Este trabalho avança em relação a modelos existentes ao introduzir abordagens inovadoras para lidar com a variabilidade e a complexidade dos dados de carga. Para isso, foi construído um conjunto de dados robusto, contemplando diversas condições operacionais do sistema IEEE de 30 barras. O conjunto de dados utilizado neste trabalho foi composto pelas condições operacionais de seus geradores e das 21 cargas conectadas. Especificamente, as variáveis de entrada do modelo corresponderam às potências ativa e reativa fornecidas pelos geradores, enquanto as variáveis de saída poderiam, em princípio, ser as potências ativa e reativa de cada uma das 21 cargas. Entretanto, para os experimentos conduzidos neste estudo, apenas a carga 2 foi selecionada como alvo de previsão. Essa escolha teve como objetivo reduzir a complexidade inicial do problema, permitindo validar a metodologia em um cenário controlado. Assim, ainda que o banco de dados completo inclua todas as cargas do sistema, o modelo de previsão foi treinado e avaliado considerando exclusivamente a demanda ativa e reativa da carga 2. Os dados foram pré-processados e normalizados para o treinamento e validação do modelo de previsão.

Outra novidade apresentada neste trabalho foi o uso de um Algoritmo Genético (GA) para otimizar os hiperparâmetros da rede LSTM, aumentando a velocidade de convergência e reduzindo o erro médio de previsão.

Em conclusão, este estudo representa uma contribuição significativa para o desenvolvimento de técnicas preditivas aplicadas à operação de sistemas elétricos, com implicações diretas para o setor elétrico e para o planejamento energético sustentável. Pesquisas futuras podem investigar a inclusão de variáveis operacionais adicionais, bem como o impacto de eventos extremos e contingências, visando aprimorar ainda mais os modelos preditivos propostos.

## 2 MATERIAL E MÉTODO

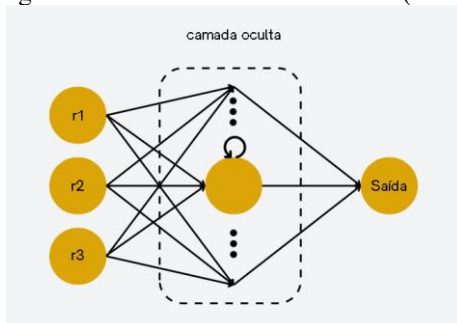
### 2.1 FUNDAMENTAÇÃO TEÓRICA

#### 2.1.1 RNN

As Redes Neurais Recorrentes (RNNs) constituem uma classe de redes neurais artificiais projetadas especificamente para lidar com dados sequenciais. Diferentemente das redes *feedforward* tradicionais, as RNNs incorporam laços de realimentação que permitem que informações de instantes anteriores influenciem a saída atual. Essa característica as torna particularmente adequadas para tarefas de previsão, nas quais dependências temporais e correlações desempenham papel fundamental [5].

Entretanto, as RNNs tradicionais frequentemente enfrentam limitações, como o desaparecimento (*vanishing*) e a explosão de gradientes (*exploding gradients*) quando treinadas em sequências longas. Esses problemas reduzem a capacidade da rede de capturar dependências de longo prazo, que são essenciais para previsões precisas em sistemas elétricos.

Figura 4: Redes Neurais Recorrentes (RNN)

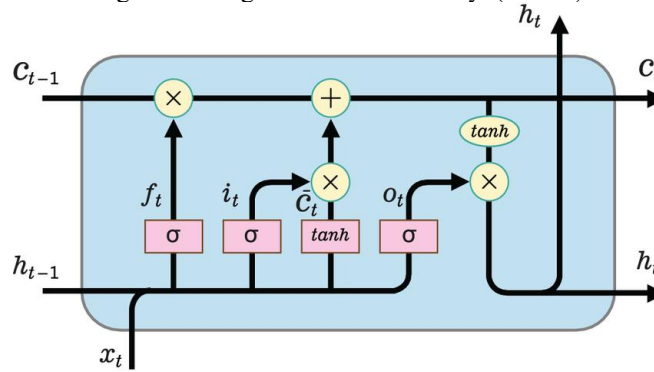


Fonte: O autor.

### 2.1.1.1 LSTM

Cada célula da camada oculta corresponde a uma unidade Long Short-Term Memory (LSTM), ilustrada na Figura 4. Essa é uma variação das redes neurais recorrentes projetada para lidar com dependências de longo prazo em sequências de dados. Seu principal objetivo é evitar problemas como o desaparecimento ou a explosão de gradientes, comuns em RNNs tradicionais. Para isso, a LSTM utiliza um mecanismo estruturado de portas que controlam o fluxo de informações ao longo do tempo, permitindo que a rede decida quais informações manter, descartar ou atualizar.

Figura 5: Long Short Term Memory (LSTM)



Fonte: Researchgate

De forma geral, a célula LSTM mantém dois estados principais: o estado da célula ( $C_t$ ) e o estado oculto ( $h_t$ ). O estado da célula é responsável por armazenar informações ao longo do tempo, enquanto o estado oculto serve como saída a cada instante. Esses estados são atualizados dinamicamente por meio de três portas principais: porta de esquecimento, porta de entrada e porta de saída.

#### 2.1.1.1.1 Porta de Esquecimento ( $f$ ):

Determina quais informações do estado anterior ( $C_{t-1}$ ) devem ser descartadas. Para isso, aplica-se uma função sigmoide, que gera valores entre 0 (descartar) e 1 (manter). Esses valores são calculados a partir da combinação do estado oculto anterior ( $h_{t-1}$ ) com a entrada atual ( $x_t$ ).

$$f_t = \sigma(W_f * [h_{t-1}, x_t] + b_f) \quad (1)$$

#### 2.1.1.1.2 Porta de Entrada ( $i$ ) e Informação Candidata ( $\tilde{C}$ ):

Define quais novas informações serão adicionadas ao estado da célula. Primeiramente, uma função sigmoide calcula os coeficientes de atualização. Em seguida, gera-se uma informação candidata ( $\tilde{C}_t$ ) por meio da função tangente hiperbólica ( $\tanh$ ), que produz valores entre -1 e 1. Esses dois passos asseguram que apenas informações relevantes sejam incorporadas.

$$i_t = \sigma(W_i * [h_{t-1}, x_t] + b_i) \quad (2)$$

$$\bar{C}_t = \tanh(W_c * [h_{t-1}, x_t] + b_c) \quad (3)$$

#### 2.1.1.1.3 Atualização do Estado da Célula (Ct):

O estado atual é atualizado combinando-se o estado anterior com a nova informação candidata. A porta de esquecimento regula quanto do estado anterior será mantido, enquanto a porta de entrada define a proporção da informação candidata que será incorporada.

$$C_t = f_t \odot C_{t-1} + i_t \odot \bar{C}_t \quad (4)$$

Aqui,  $\odot$  representa a multiplicação elemento a elemento, garantindo um ajuste dinâmico do fluxo de informações.

#### 2.1.1.1.4 Porta de Saída (o):

Define quais informações do estado atual (Ct) serão utilizadas para gerar o estado oculto (ht). A combinação do estado oculto anterior e da entrada atual passa por uma função sigmoide, que controla a fração liberada. Em seguida, o estado da célula é ativado com a função tanh e multiplicado pelos valores da porta de saída:

$$o_t = \sigma(W_o * [h_{t-1}, x_t] + b_o) \quad (5)$$

#### 2.1.1.1.5 Cálculo do Estado Oculto (ht):

Finalmente, o estado oculto é calculado como o produto elemento a elemento entre a saída da porta de saída (ot) e a versão ativada do estado da célula ( $\tanh(C_t)$ ).

$$h_t = o_t \odot \tanh(C_t) \quad (6)$$

### 2.1.1.2 HeatMap

O HeatMap é uma ferramenta de visualização de dados amplamente utilizada para representar os valores de uma matriz ou tabela por meio de variações de cores. Essa técnica facilita a identificação de padrões, tendências ou correlações entre variáveis em conjuntos de dados complexos. No contexto de séries temporais e aprendizado de máquina, os HeatMaps são frequentemente empregados para explorar relações entre diferentes variáveis ou para analisar o comportamento dos dados ao longo do tempo.



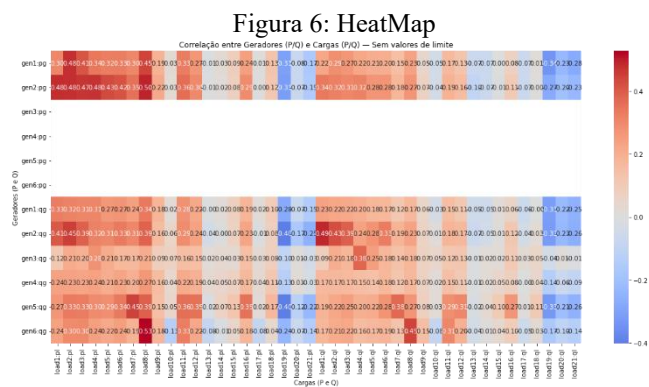
O funcionamento de um HeatMap baseia-se na atribuição de cores a valores numéricos, geralmente seguindo uma escala contínua, como gradientes que variam do azul (para valores baixos) ao vermelho (para valores altos). Cada célula do mapa representa a interseção entre duas variáveis, permitindo uma visualização intuitiva de suas interações. Por exemplo, em uma matriz de correlação, o HeatMap é usado para exibir a intensidade e a direção das relações lineares entre pares de variáveis.

Além disso, os HeatMaps são úteis para identificar anomalias ou padrões sazonais nos dados. Em aplicações práticas, como a previsão de séries temporais, eles podem ser utilizados para analisar a relevância das variáveis de entrada, identificar períodos de maior concentração de valores extremos ou compreender a influência de fatores externos sobre os dados analisados. Os HeatMaps podem ser implementados em ferramentas como *Seaborn* ou *Matplotlib*, que permitem a personalização de escalas de cor, rótulos e anotações, ampliando a capacidade de interpretação dos resultados.

Dessa forma, o uso de HeatMaps proporciona uma abordagem visual poderosa para compreender e comunicar informações complexas, sendo uma ferramenta essencial em estudos que exigem a análise de grandes volumes de dados e suas inter-relações.

A escolha dos dados de entrada na arquitetura da RNN, ilustrada na Figura 4, foi baseada na análise das correlações entre as variáveis do sistema. O HeatMap, apresentado na Figura 6, mostra as relações de correlação entre diferentes variáveis, possibilitando identificar aquelas com maior impacto no comportamento do sistema e, portanto, mais relevantes para o modelo.

Assim, as fontes de energia foram escolhidas como variáveis de entrada para o treinamento, e uma das cargas (neste caso, a Carga 2) foi selecionada como variável de saída da rede.



### 2.1.1.3 TensorFlow e Keras

O TensorFlow e o Keras são ferramentas essenciais para o desenvolvimento de modelos de *deep learning*, especialmente em tarefas que envolvem séries temporais, como a previsão de variáveis de fluxo de potência em redes elétricas.

O TensorFlow, desenvolvido pelo Google, é uma biblioteca de código aberto altamente escalável e eficiente, projetada para realizar operações matemáticas complexas e dar suporte ao treinamento de modelos em hardwares de alto desempenho, como GPUs e TPUs. O Keras, integrado como a API de alto nível do TensorFlow, fornece uma interface modular e de fácil utilização para a construção de redes neurais, permitindo que os pesquisadores foquem nos aspectos conceituais e experimentais da modelagem.

No modelo desenvolvido neste trabalho, diversas funcionalidades do TensorFlow e do Keras foram exploradas pela sua simplicidade e eficiência na implementação. A arquitetura foi construída utilizando a classe Sequential, que organiza as camadas de forma linear e é adequada para redes neurais que processam dados sequenciais, como as LSTMs empregadas neste estudo. As camadas LSTM formaram a espinha dorsal do modelo, projetadas para capturar padrões temporais e dependências de longo prazo nos dados de entrada. Essa capacidade é fundamental na análise de séries temporais em sistemas de potência, já que os valores de carga atuais são fortemente influenciados por variações passadas das potências ativa e reativa.

Para aprimorar o processo de aprendizado, foi adicionada uma camada Bidirectional, que permite à LSTM analisar os dados tanto no sentido direto quanto no inverso do tempo. Essa abordagem é particularmente vantajosa na previsão de fluxo de potência, onde as condições futuras de operação também podem ser influenciadas por interações passadas entre geração e demanda.

Além das camadas LSTM, foram incluídas camadas Dense, responsáveis por realizar transformações lineares e não lineares nas representações internas, conectando as saídas das LSTM às previsões finais. A camada de saída foi configurada com dois neurônios, correspondentes às potências ativa (P) e reativa (Q) da carga selecionada, garantindo que o modelo produzisse previsões alinhadas às grandezas físicas de interesse.

Para o treinamento, o modelo foi compilado utilizando o otimizador Adam, conhecido por sua eficiência em tarefas de *deep learning* devido ao ajuste adaptativo da taxa de aprendizado. A função de perda escolhida foi o Erro Absoluto Médio (MAE), adequada para problemas de regressão e que fornece uma interpretação direta da média das diferenças entre valores previstos e reais de potência.

Após o treinamento, as previsões foram geradas com a função predict do Keras, integrada ao fluxo de processamento. De forma geral, o TensorFlow e o Keras se destacaram não apenas pela facilidade de uso, mas também por oferecerem soluções avançadas para problemas complexos, como a captura de dependências temporais na operação de sistemas elétricos. Esses frameworks foram fundamentais para garantir a eficiência, flexibilidade e rigor científico do modelo de previsão proposto.

#### 2.1.1.4 Algoritmo Genético para Otimização de Hiperparâmetros

Os Algoritmos Genéticos (AGs) constituem uma classe de técnicas estocásticas de otimização inspiradas nos princípios da seleção natural e da genética. Inicialmente propostos por John Holland na década de 1970, os AGs são amplamente utilizados para resolver problemas de otimização em que o espaço de busca é complexo, não linear ou de difícil abordagem por métodos tradicionais baseados em gradiente. Sua robustez e adaptabilidade os tornam adequados para uma ampla variedade de aplicações em engenharia, incluindo aprendizado de máquina e análise de sistemas de potência.

Em sua essência, os AGs operam sobre uma população de soluções candidatas, chamadas de indivíduos, que são representados por cromossomos (frequentemente codificados como cadeias binárias, vetores de valores reais ou estruturas mais complexas, dependendo do problema). Cada cromossomo corresponde a uma solução potencial, cuja qualidade é avaliada por meio de uma função de aptidão (*fitness function*) que mede o quão bem ele atende ao objetivo de otimização.

O processo de um AG segue as seguintes etapas principais:

- **Inicialização:** Uma população de soluções candidatas é gerada aleatoriamente dentro do espaço de busca definido. No contexto da otimização de hiperparâmetros em redes neurais, esses candidatos podem representar configurações como número de células LSTM, taxas de *dropout*, taxa de aprendizado e número de épocas de treinamento.
- **Seleção:** Os indivíduos são selecionados com base em suas pontuações de aptidão, favorecendo aqueles com melhor desempenho. Estratégias como roleta, torneio ou seleção por ranking garantem que os indivíduos mais fortes tenham maior probabilidade de transmitir suas características à próxima geração.
- **Crossover (Recombinação):** Pares de indivíduos selecionados trocam partes de suas representações cromossômicas para gerar descendentes. Esse processo imita a reprodução biológica, promovendo a combinação de características vantajosas de diferentes soluções.
- **Mutação:** Alterações aleatórias são introduzidas nos cromossomos dos descendentes com uma pequena probabilidade. A mutação garante diversidade na população, evitando a convergência prematura para soluções subótimas.
- **Substituição:** Forma-se uma nova população, geralmente combinando os melhores indivíduos da geração anterior com os descendentes recém-gerados. O ciclo de avaliação, seleção, crossover e mutação é repetido por um número definido de gerações ou até que os critérios de convergência sejam atendidos.

#### 2.1.1.4.1 Aplicação neste trabalho

No presente estudo, o Algoritmo Genético foi empregado para otimizar os hiperparâmetros do modelo de previsão baseado em LSTM desenvolvido para o sistema IEEE de 30 barras. Em vez de selecionar os parâmetros manualmente por tentativa e erro, o AG realizou uma busca automática pela melhor combinação de hiperparâmetros que minimizasse o erro de previsão.

Os seguintes hiperparâmetros foram codificados nos cromossomos do AG:

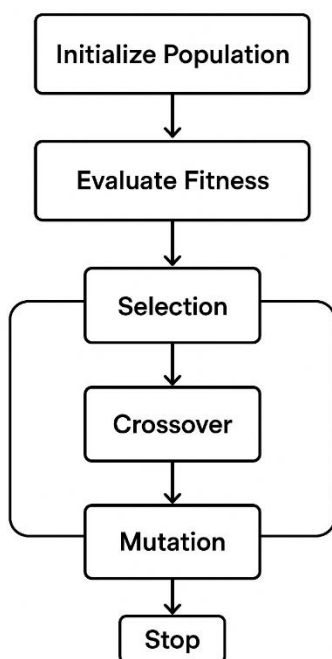
- Número de unidades LSTM por camada: o AG explorou diferentes configurações de neurônios ocultos, equilibrando a complexidade do modelo e sua capacidade de generalização.
- Número de épocas de treinamento: ao otimizar esse parâmetro, o AG garantiu uma convergência eficiente sem causar *overfitting* ou *underfitting*.
- Taxa de *dropout*: o AG ajustou o nível de regularização para evitar *overfitting*, aspecto particularmente importante em modelos sequenciais com grande capacidade de representação.
- Taxa de aprendizado: embora tenha sido fixada em parte dos experimentos, este parâmetro também pode ser otimizado pelo AG para refinar a velocidade de convergência e a estabilidade do treinamento.

A função de aptidão foi definida como o Erro Absoluto Médio (MAE) entre os valores previstos e reais de carga (potências ativa e reativa), após o treinamento do modelo com determinado conjunto de hiperparâmetros. Valores menores de MAE indicaram melhor desempenho, guiando a busca evolutiva em direção a configurações mais adequadas.

Com a integração do AG ao processo de modelagem, o modelo alcançou um MAE final de 0,0507, resultado superior ao de modelos de referência com hiperparâmetros ajustados manualmente. Esse desempenho destaca a eficiência dos AGs na exploração de grandes espaços de busca e sua capacidade de identificar, de forma adaptativa, soluções ótimas para problemas complexos, como a previsão de séries temporais em sistemas elétricos.

Figura 7: Esquema do Algoritmo Genético

## Genetic Algorithm



Fonte: os autores

### 3 ESTUDO DE CASO E RESULTADOS OBTIDOS

Uma vez estabelecida a metodologia para a implementação da previsão de fluxo de potência, sua validação necessariamente dependeu da comparação do desempenho do modelo em relação a um caso de referência derivado de um alimentador real de distribuição.

O código foi implementado em Python para prever as potências ativa (P) e reativa (Q) de uma carga selecionada, utilizando como entradas os dados de geração e variáveis auxiliares do sistema. Essa abordagem permite que o modelo capture padrões temporais e operacionais inerentes à dinâmica dos sistemas de potência, garantindo previsões precisas de curto prazo. Tais previsões são cruciais para o planejamento e a operação de sistemas elétricos modernos, especialmente em cenários com geração distribuída e demanda flutuante. A seguir, apresenta-se o procedimento adotado e as considerações sobre os resultados obtidos.

#### 3.1 DIVISÃO DO CONJUNTO DE DADOS

A divisão do conjunto de dados é uma etapa crítica para garantir que o modelo aprenda padrões confiáveis e seja capaz de generalizar para dados não vistos. Neste estudo, foi adotada uma divisão temporal, alocando 80% das amostras para treinamento e 20% para teste. Essa divisão preserva a ordem cronológica dos eventos, evitando o vazamento de informações (data leakage), aspecto fundamental em previsões de séries temporais. Diferentemente das divisões aleatórias, frequentemente utilizadas em outras tarefas de

aprendizado de máquina, a manutenção da sequência temporal reflete melhor as aplicações reais do modelo em cenários operacionais.

### 3.2 CONSTRUÇÃO DAS SEQUENCIAS

Como as redes LSTM são projetadas para processar dados sequenciais, o código empregou uma estratégia de janela deslizante (*sliding window*) para construir as sequências temporais. Especificamente, cada sequência de entrada foi composta por 20 instantes consecutivos de operação do sistema, enquanto a saída correspondeu aos valores de potência ativa e reativa da carga alvo no instante subsequente. Essa configuração fornece ao modelo contexto histórico suficiente para aprender as dependências temporais no comportamento da carga.

### 3.3 ARQUITETURA DA REDE NEURAL

A arquitetura da rede neural foi projetada para capturar tanto as dependências temporais quanto as não lineares presentes nos dados de fluxo de potência. Ela é composta por camadas empilhadas de LSTM Bidirecional, camadas densas totalmente conectadas e uma camada de saída linear. Diferente de arquiteturas fixas, a configuração deste modelo — como o número de unidades LSTM por camada e o número de épocas de treinamento — não foi escolhida arbitrariamente. Em vez disso, um Algoritmo Genético (AG) foi utilizado para otimizar os hiperparâmetros, buscando combinações que resultassem na melhor precisão de previsão.

**Camadas LSTM Bidirecionais:** foram testadas duas camadas bidirecionais com diferentes quantidades de unidades (ex.: 64, 128), permitindo que a rede aprendesse dependências temporais tanto no sentido direto quanto no inverso. Isso é especialmente relevante em sistemas elétricos, onde as condições atuais podem depender tanto de estados passados quanto de estados em evolução.

**Regularização por Dropout:** foi aplicada uma taxa de *dropout* entre 1% e 5%, a fim de mitigar *overfitting* e assegurar que o modelo generalizasse bem para condições operacionais não vistas.

**Camada Densa:** após as LSTM, foi incluída uma camada densa de tamanho variável (otimizada pelo AG) para refinar as representações aprendidas antes da saída final.

**Camada de Saída:** a camada final continha dois neurônios, correspondentes às previsões de potência ativa (P) e reativa (Q) da carga selecionada.

### 3.4 OTIMIZAÇÃO DE HIPERPARAMETROS COM ALGORITMO GENETICO

O Algoritmo Genético desempenhou papel central na escolha da configuração ótima do modelo. As soluções candidatas (cromossomos) codificaram hiperparâmetros como o número de células LSTM por camada e o número de épocas de treinamento. O AG evoluiu essas soluções ao longo das gerações, por meio

dos processos de seleção, *crossover* e mutação, convergindo para arquiteturas que minimizavam o erro de validação. Essa abordagem evolutiva mostrou-se eficiente no balanceamento entre a complexidade do modelo e sua precisão preditiva.

É importante destacar que os hiperparâmetros obtidos representam a melhor configuração para o conjunto de dados e condições operacionais deste estudo. No entanto, diferentes conjuntos de dados ou cenários de sistemas elétricos podem exigir configurações alternativas, reforçando a flexibilidade da combinação entre redes LSTM e técnicas de otimização evolutiva como os Algoritmos Genéticos.

### 3.5 COMPILAÇÃO E CONFIGURAÇÃO DO MODELO

Após definida a arquitetura, o modelo foi compilado com os seguintes elementos:

- Função de Perda: utilizou-se o Erro Absoluto Médio (MAE). Diferentemente do Erro Quadrático Médio (MSE), que penaliza mais severamente grandes desvios, o MAE fornece uma métrica de interpretação mais direta no contexto de fluxo de potência, representando a magnitude média dos erros de previsão nas potências ativa e reativa.
- Otimizador Adam: adotou-se o Adam (Adaptive Moment Estimation), que combina vantagens do RMSProp e do Gradiente Estocástico (SGD). O Adam ajusta dinamicamente a taxa de aprendizado de cada parâmetro com base em estimativas de primeira e segunda ordem (média e variância), acelerando a convergência e garantindo estabilidade no treinamento de dados sequenciais complexos.

### 3.6 TREINAMENTO DO MODELO

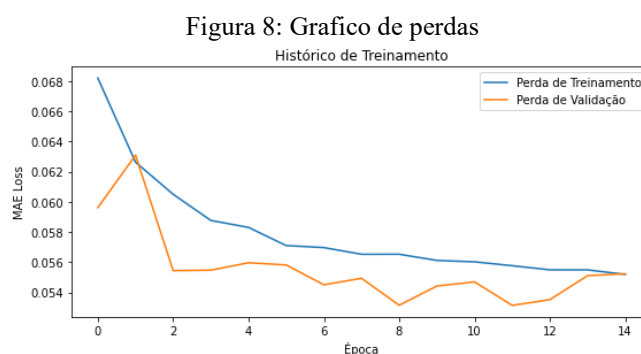
O treinamento foi realizado ao longo de 15 épocas, com tamanho de lote de 32 amostras. Durante o treinamento, aplicaram-se as seguintes estratégias:

- Taxa de Aprendizado Adaptativa: embora não tenha sido utilizado explicitamente o *callback* ReduceLROnPlateau, a taxa de aprendizado foi ajustada empiricamente e também pelo Algoritmo Genético, assegurando convergência sem oscilações significativas.
- Monitoramento da Função de Perda: as perdas de treinamento e validação foram acompanhadas a cada época, a fim de detectar sinais de *overfitting* ou *underfitting*. A evolução dessas perdas é mostrada na Figura 7 (Gráfico de Perda de Treinamento e Validação), confirmando o processo de aprendizado estável do modelo.

### 3.7 AVALIAÇÃO DO MODELO

Após o treinamento, o modelo foi avaliado no conjunto de teste para medir sua capacidade de generalização. A avaliação envolveu as seguintes etapas:

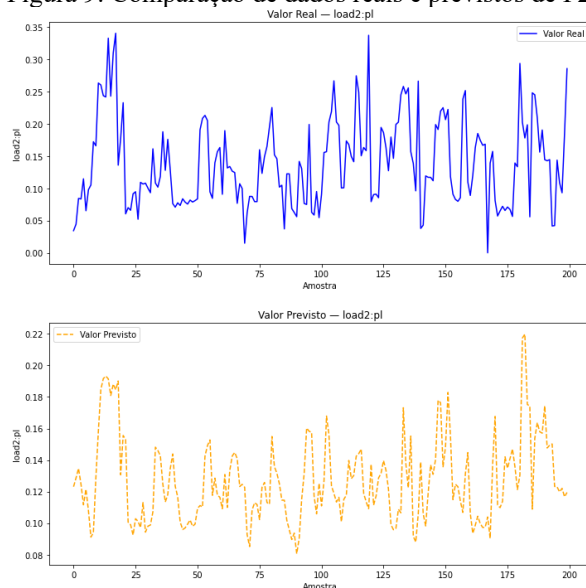
- Desnormalização: tanto os valores previstos quanto os reais foram reescalados para suas unidades originais utilizando os objetos *MinMaxScaler* ajustados. Isso assegurou que as métricas de avaliação fossem interpretáveis dentro do contexto físico do fluxo de potência.
- Erro Absoluto Médio (MAE): o MAE final obtido foi de 0,0507, refletindo um elevado grau de precisão considerando a escala das potências ativa e reativa no sistema IEEE de 30 barras. Esse baixo erro demonstra a robustez do modelo e sua capacidade de capturar relações temporais e não lineares complexas.
- Visualização das Previsões: foram gerados gráficos comparativos entre valores reais e previstos para as potências ativa e reativa da carga alvo. Esses gráficos, apresentados nas Figuras 8 (Potência Ativa – P2) e 9 (Potência Reativa – Q2), destacam a capacidade do modelo de acompanhar tendências e flutuações, confirmando sua eficácia na previsão de curto prazo do comportamento de cargas.



Fonte: os autores

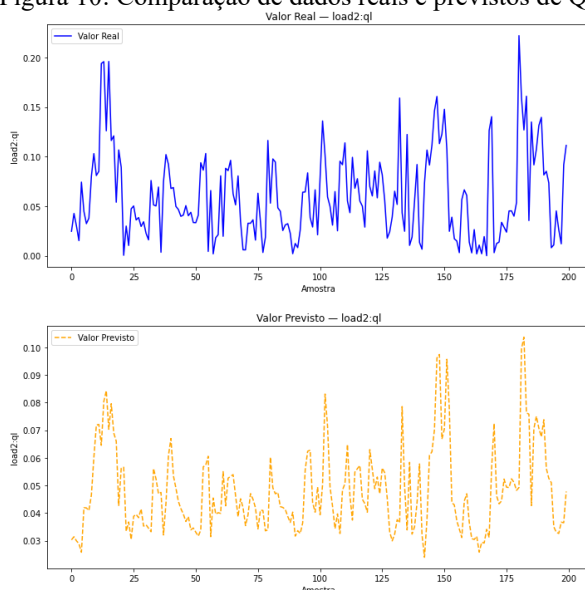


Figura 9: Comparação de dados reais e previstos de P2



Fonte: os autores

Figura 10: Comparação de dados reais e previstos de Q2



Fonte: os autores

## 4 FORÇAS E LIMITAÇÕES DO ESTUDO

A maioria dos estudos que analisam previsões em sistemas elétricos de potência não combina técnicas de redes neurais recorrentes com unidades de memória de longo prazo (LSTM) e Algoritmos Genéticos (AG) para otimização de hiperparâmetros em problemas de previsão de carga. Como ponto forte do nosso estudo, destacamos justamente a utilização dessa abordagem integrada, que permitiu uma adaptação automática da rede neural e reduziu a subjetividade na escolha dos parâmetros de treinamento. Além disso, enfatizamos que a análise foi realizada utilizando dados derivados de um sistema padrão amplamente aceito na literatura (IEEE 30 barras), garantindo a reprodutibilidade e comparabilidade dos resultados. Outro aspecto positivo é que os resultados alcançaram um erro médio absoluto (MAE)

relativamente baixo, o que reforça a robustez do modelo proposto e sua utilidade prática para aplicações de previsão no setor elétrico. Assim, acreditamos que os resultados deste estudo são de grande relevância, pois oferecem evidências iniciais sobre a aplicabilidade de técnicas de aprendizado profundo combinadas a algoritmos evolutivos em cenários de fluxo de potência.

Entretanto, reconhecemos que nosso estudo também apresenta limitações. Primeiramente, ele foi conduzido considerando apenas a previsão da potência ativa e reativa de uma carga específica, o que restringe a generalização dos resultados para todo o sistema. Além disso, embora os resultados tenham sido satisfatórios, a utilização de dados simulados limita a validade externa, uma vez que não foram considerados ruídos, falhas de medição ou contingências típicas de sistemas reais. Outro ponto limitante é o custo computacional associado ao Algoritmo Genético, que pode se tornar elevado em aplicações em tempo real ou em sistemas de maior porte. Outros fatores limitantes incluem a temporalidade reduzida da base de dados analisada e a ausência de testes sob cenários de eventos extremos ou condições de operação anômalas. Embora o modelo tenha mostrado capacidade preditiva em condições normais, ainda não foi validado em situações de maior variabilidade ou distúrbios elétricos, que são comuns em redes reais.

Portanto, pesquisas adicionais são sugeridas, especialmente aquelas que envolvam sistemas de maior porte e dados reais de operação, para avaliar a robustez do modelo em condições mais desafiadoras. Também é recomendada a investigação de outras metaheurísticas de otimização, bem como a expansão do escopo para múltiplas cargas simultâneas, de modo a aumentar a aplicabilidade e a generalização do método proposto.

## 5 CONCLUSÕES

Os resultados obtidos neste estudo demonstram a eficácia dos modelos de Redes Neurais Recorrentes (RNN), com unidades Long Short-Term Memory (LSTM), para a previsão da potência de carga (componentes ativa e reativa) no sistema IEEE de 30 barras. O modelo proposto, otimizado por meio de um Algoritmo Genético para a seleção de hiperparâmetros, alcançou um Erro Absoluto Médio (MAE) de 0,0507, o que indica um elevado nível de precisão na captura de padrões temporais a partir de dados históricos de fluxo de potência.

Embora o modelo tenha sido projetado para prever a potência ativa e reativa de uma única carga, a metodologia pode ser facilmente estendida para múltiplas cargas e configurações de rede mais complexas. O uso das potências ativa e reativa dos geradores como variáveis de entrada mostrou-se suficiente para modelar a dinâmica da carga selecionada, reduzindo a complexidade computacional e, ao mesmo tempo, mantendo um desempenho preditivo robusto.

A integração do Algoritmo Genético foi particularmente importante, pois possibilitou uma busca automática por hiperparâmetros ótimos — como o número de células LSTM por camada e o número de

épocas de treinamento — resultando em maior velocidade de convergência e menor erro de previsão em comparação a modelos ajustados manualmente.

Este trabalho representa uma contribuição significativa para o desenvolvimento de técnicas inteligentes de previsão aplicadas a sistemas de potência, oferecendo uma base sólida para aprimorar o planejamento operacional, a confiabilidade e a eficiência em smart grids. Pesquisas futuras podem explorar a aplicação desta abordagem a todas as cargas do sistema IEEE de 30 barras, a inclusão de fontes renováveis estocásticas e a utilização de técnicas híbridas de otimização, com o objetivo de refinar ainda mais a capacidade preditiva de modelos baseados em redes neurais em aplicações no setor elétrico.

## REFERÊNCIAS

- [1] Passini, A. F. C., Niz, I. V. G., Munaretto, L. F., de Borba, W. F., & Rodrigues, A. C. (2023). Energia solar no Brasil: Oportunidades e desafios. Instituto Brasileiro de Estudos Ambientais.
- [2] Passini, A. F. C., Niz, I. V. G., Munaretto, L. F., Borba, W. F., & Rodrigues, A. C. Energia Solar no Brasil: Oportunidades e Desafios. Instituto Brasileiro de Estudos Ambientais, 2023
- [3] Dronio. Solar Energy Dataset. Kaggle, 2019. Disponível em: <https://www.kaggle.com/datasets/dronio/SolarEnergy>.
- [4] Leal, A. G. (2006). Sistema para Determinação de Perdas em Redes de Distribuição de Energia Elétrica Utilizando Curvas de Demanda Típicas de Consumidores e Redes Neurais Artificiais. Tese de Doutorado, Escola Politécnica da Universidade de São Paulo, São Paulo
- [5] CORREIA, V. T. (2021). SISTEMA DE GERENCIAMENTO DE ENERGIA PARA MICRORREDES. CURITIBA: UNIVERSIDADE FEDERAL DO PARANÁ.
- [6] Cunha, M. V. da. (2016). Estratégias de gerenciamento pelo lado da demanda aplicadas aos consumidores de BT considerando a tarifa branca e a geração distribuída. Universidade Federal de Santa Maria.
- [7] Lara Filho, M. O. de. (2021). Modelo robusto orientado a dados para programação diária de operação de microrredes considerando recursos energéticos distribuídos sob incertezas.
- [8] Saúde, L. M. S. (2018). Análise comparativa entre os métodos autoregressivo, integrado de médias móveis e rede neural artificial para previsão de séries temporais.
- [9] Bissiriou, A. O. S. (2023). Contribuições para o gerenciamento de energia de microrredes CA monofásicas usadas em comunidades isoladas. Natal: Universidade Federal do Rio Grande do Norte.
- [10] Alves, F. A. Uma análise das smart grids aplicadas às microrredes inteligentes. Sobral: Universidade Federal do Ceará.
- [11] Rauber, T. W. Redes neurais artificiais. Universidade Federal do Espírito Santo.