

Chronos: módulo de predição de séries temporais com LSTM e GRU em ecossistemas IoT para cidades inteligentes

Marcelly Molinari Marsura, 081210020@faculdade.cefsa.edu.br, FESA
Nicolas Alexander da Silva Santos, 081210017@faculdade.cefsa.edu.br, FESA
Vinícius Benevides da Silva, 081210030@faculdade.cefsa.edu.br, FESA
Vitor Henrique Carvalho Silva, 081200030@faculdade.cefsa.edu.br, FESA
Dr. Fábio Henrique Cabrini, pro7799@cefsa.edu.br, FESA

Resumo

Com o avanço das cidades inteligentes, tecnologias emergentes como Internet das Coisas (IoT) e Inteligência Artificial (IA) têm sido aplicadas no monitoramento e otimização dos serviços urbanos. Nesse contexto, a análise de séries temporais provenientes de sensores conectados é essencial para antecipar eventos críticos e apoiar o planejamento de recursos. Apesar dos progressos, ainda há carência de soluções *open-source* que integrem, de forma nativa, modelos de aprendizado profundo em plataformas de IoT. Este trabalho apresenta o desenvolvimento em andamento do Chronos, um módulo de *Deep Learning* baseado nas arquiteturas *Long Short-Term Memory* (LSTM) e *Gated Recurrent Unit* (GRU), voltado à previsão de séries temporais no ecossistema FIWARE. A proposta inclui uma Interface de Programação de Aplicações (API) *RESTful* para treinamento e inferência, um *dashboard* interativo para acompanhamento em tempo de execução e integração com dados meteorológicos simulados. Resultados preliminares indicam que ambos os modelos alcançaram boa acurácia, com erros médios próximos (RMSE de 1,349 °C para o LSTM e 1,227 °C para o GRU). Entretanto, o GRU apresentou desempenho ligeiramente superior e tempo de processamento quase duas vezes menor, evidenciando-se como alternativa mais eficiente para cenários com restrições computacionais. O projeto encontra-se em fase de validação e integração ao FIWARE, com perspectivas de coleta de dados reais, gestão de modelos e consolidação da documentação.

Palavras-chave: LSTM. GRU. Séries Temporais. Aprendizado Profundo. Cidades Inteligentes.

Introdução

O conceito de cidades inteligentes tem se consolidado como uma alternativa estratégica para enfrentar desafios crescentes da urbanização. Mais do que a adoção de tecnologias, trata-se de um modelo de desenvolvimento urbano que busca alinhar inovação, sustentabilidade e planejamento social para construir ambientes resilientes, inclusivos e adaptáveis às necessidades da população (LEE *et al.*, 2023).

Nesse cenário, a Internet das Coisas (IoT) ocupa papel central ao viabilizar a coleta contínua de dados por meio de sensores distribuídos em infraestrutura urbana, orquestrada por soluções como o FIWARE. Esta é uma plataforma *open-source* voltada para gestão de dados contextuais amplamente utilizada em aplicações de cidades inteligentes por sua capacidade de integrar dispositivos IoT e serviços de análise em tempo de execução (FIWARE FOUNDATION, 2025; SANG *et al.*, 2020).

Esses dispositivos registram informações organizadas principalmente no formato de séries temporais. Elas correspondem a observações ordenadas cronologicamente que refletem o estado dinâmico do ambiente (PAPASTEFANOPOULOS *et al.*, 2023). A análise dessas séries permite tanto o monitoramento imediato quanto a antecipação de eventos críticos, como enchentes, sobrecarga energética

ou congestionamentos, conferindo maior capacidade de resposta e planejamento às cidades (SANTOS; RIBEIRO FILHO, 2024).

A efetividade dessas análises e previsões depende diretamente da aplicação de técnicas avançadas de Inteligência Artificial. Entre os métodos mais promissores destacam-se as redes neurais recorrentes profundas, em especial as arquiteturas *Long Short-Term Memory* (LSTM) e *Gated Recurrent Unit* (GRU), reconhecidas pela habilidade de capturar dependências de longo prazo, padrões sazonais e variações não lineares em dados sequenciais. Essas características tornam os modelos adequados para previsões em cenários urbanos, onde a dinâmica é marcada por múltiplos fatores interdependentes e pela necessidade de respostas rápidas e precisas (SHRESTHA; MAHMOOD, 2019).

Apesar do avanço no desenvolvimento de plataformas urbanas inteligentes, ainda há uma lacuna significativa em relação à disponibilidade de soluções *open-source* que integrem de forma nativa algoritmos de aprendizado profundo voltados à previsão de séries temporais. Na prática, pesquisadores e desenvolvedores enfrentam dificuldades em implementar tais modelos em sistemas de IoT, pois a ausência de componentes específicos resulta em processos manuais de integração, eleva custos de desenvolvimento e reduz a escalabilidade das aplicações (DUTTA BARUAH; MUÑOZ ORGANERO, 2024).

Diante desse contexto, este trabalho apresenta o Chronos, um módulo de aprendizado profundo para previsão de séries temporais em cidades inteligentes. A proposta busca disponibilizar um componente modular e escalável, baseado em LSTM e GRU, integrado ao ecossistema FIWARE. O sistema é exposto por meio de *endpoints* RESTful, permitindo sua utilização em diferentes aplicações, e acompanhado por uma interface gráfica voltada à gestão e visualização das previsões.

O objetivo central do projeto é desenvolver e validar esse componente, explorando sua viabilidade técnica e eficiência preditiva. Entre os objetivos específicos estão: (i) implementar as arquiteturas LSTM e GRU em Python com suporte a treinamento e previsão via Interface de Programação de Aplicações (API); (ii) aplicar o módulo em um estudo de caso com dados meteorológicos para avaliar sua performance; e (iii) documentar o sistema, incluindo aspectos arquiteturais, técnicos e de replicabilidade. A relevância da pesquisa reside tanto em seu impacto prático, ao apoiar a tomada de decisão em ambientes urbanos, quanto em sua contribuição científica, ao ampliar o acesso a ferramentas abertas e reprodutíveis de Inteligência Artificial aplicada às cidades inteligentes.

Metodologia

A pesquisa segue uma abordagem aplicada, de caráter exploratório e descritivo, conduzida de forma incremental. Essa estratégia possibilita que cada etapa, seja avaliada progressivamente, permitindo ajustes contínuos e assegurando a replicabilidade da solução em diferentes contextos urbanos.

Para a realização dos testes dos modelos, foi utilizada uma série temporal meteorológicas de temperatura máxima em graus Celsius, obtida junto ao Instituto Nacional de Meteorologia (INMET), com amostras horárias no período de 01/01/2022 a 31/10/2024. Esses dados foram transformados para o padrão NGSI (*Next Generation Service Interfaces*), de modo a garantir compatibilidade com o ecossistema FIWARE. Os experimentos foram conduzidos em uma máquina virtual com 4 vCPUs, 8 GB de memória RAM, 50 GB de armazenamento SSD e sistema operacional Ubuntu Server 22.04 LTS.

A base foi organizada de forma cronológica, destinando 80% dos registros ao treinamento e validação e 20% ao teste, conforme abordagem recomendada em aplicações urbanas similares (BARRETO; BERNARDINI; OLIVEIRA, 2024). O pré-processamento contemplou o tratamento de valores ausentes, com interpolação linear para manter a consistência temporal, detecção e correção de *outliers* por meio do

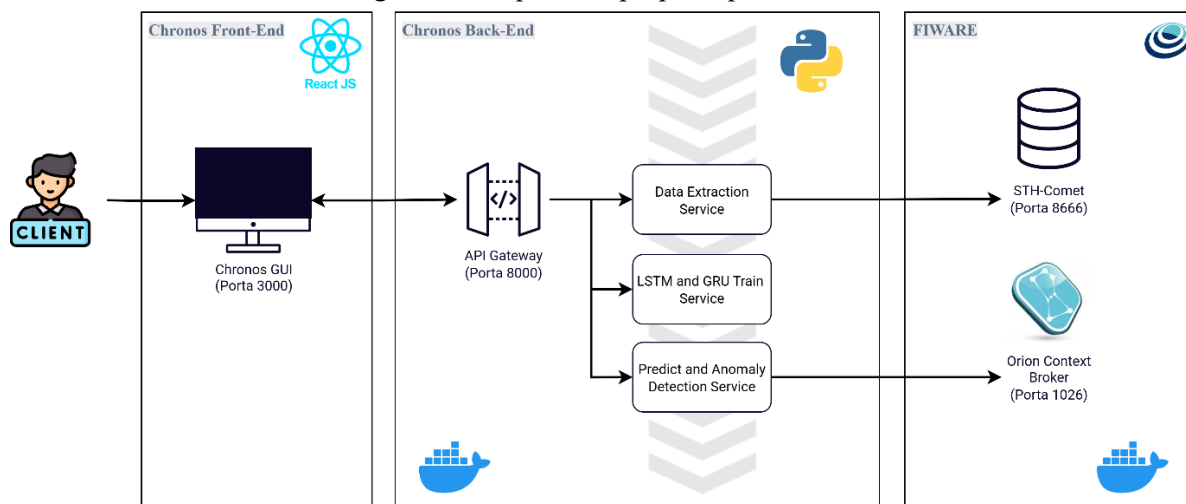
método do desvio interquartil (IQR), substituindo valores anômalos por suavizações locais, e aplicação de normalização *min-max*, realizada apenas sobre o conjunto de treino e replicada no conjunto de teste, a fim de reduzir riscos de *data leakage*, cenário em que informações do conjunto de teste influenciam no treinamento, comprometendo a validade da avaliação. Essas etapas asseguram a qualidade e consistência do fluxo de dados, sendo práticas recomendadas em análises de séries temporais (MIGLIATO; PONTI, 2021; SANG *et al.*, 2020).

Implementadas utilizando a biblioteca TensorFlow Keras, as arquiteturas LSTM e GRU foram selecionadas devido à sua capacidade de capturar dependências temporais de longo prazo e lidar com padrões sazonais. A configuração utilizada contemplou uma camada recorrente com 128 unidades, seguida de uma camada de dropout (0,2) para regularização, uma camada densa intermediária com 64 neurônios e ativação ReLU, além de uma camada de saída linear responsável pela previsão. O treinamento foi conduzido em 30 épocas, utilizando janelas deslizantes de 60 observações como entrada, horizonte de previsão de 5 passos e taxa de aprendizado de 0,001. As equações seguem as formulações originais propostas por Hochreiter e Schmidhuber (1997) para o LSTM e Cho *et al.* (2014) para o GRU.

A avaliação dos modelos baseou-se nas métricas recomendadas por Papastefanopoulos *et al.* (2023), em especial o Erro Médio Absoluto (MAE) e a Raiz do Erro Quadrático Médio (RMSE), por refletirem adequadamente o desempenho em séries temporais. Para mitigar riscos de sobreajuste (*overfitting*), foram aplicadas técnicas de regularização combinadas ao mecanismo de *early stopping*, que interrompe o treinamento em caso de estagnação na melhoria da perda de validação. O ajuste de hiperparâmetros (incluindo taxa de aprendizado, número de épocas, tamanho do *batch*, número de camadas e unidades ocultas) foi realizado de forma experimental, orientado pelos resultados obtidos nas métricas de avaliação.

A integração da solução ao FIWARE será realizada por meio de um módulo implementado em containers Docker, desenvolvido em Python, responsável pelo pré-processamento, treinamento e predição, exposto via API RESTful construída com Flask. Para interação e análise, foi projetada uma interface web em React, que permite a configuração de hiperparâmetros, o monitoramento de métricas de desempenho e a visualização das previsões em tempo de execução (Figura 1).

Figura 1 – Arquitetura proposta para o GE

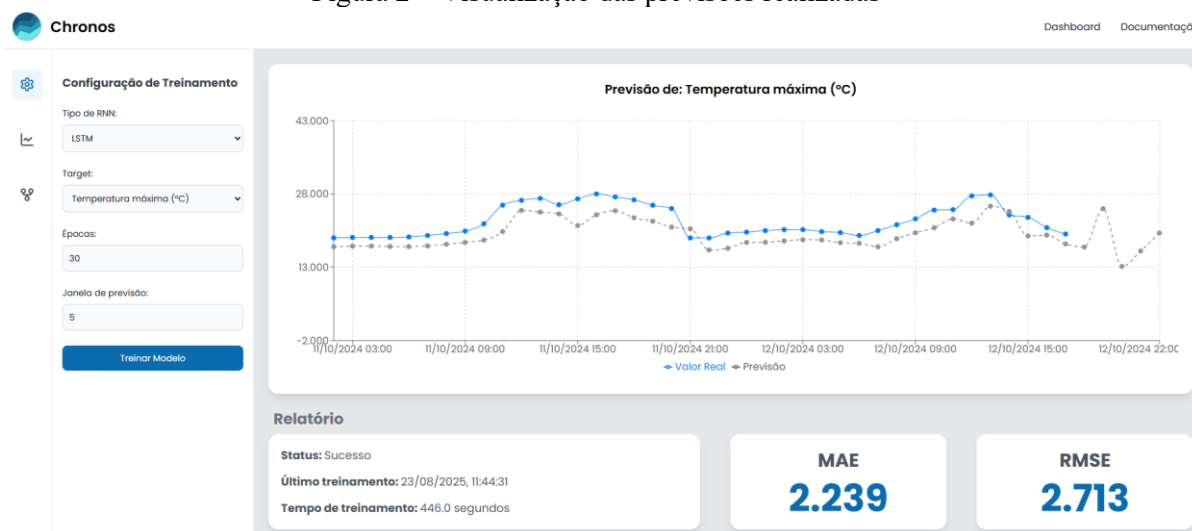


Fonte: Autoria própria (2025)

Resultados preliminares

O desenvolvimento do Chronos resultou em um protótipo funcional, integrando os módulos previstos. O sistema dispõe de uma *Web API* em Python, documentada em Swagger, que centraliza o treinamento, a inferência e a configuração dos modelos. Complementarmente, foi criada uma interface em React que permite ajustar hiperparâmetros, executar treinamentos e visualizar resultados gráficos e relatórios de desempenho (MAE, RMSE e tempo de processamento), ampliando o acesso mesmo para usuários não especializados, conforme Figura 2.

Figura 2 – Visualização das previsões realizadas



Fonte: Autoria própria (2025)

Os testes iniciais com dados meteorológicos simulados permitiram comparar o desempenho das arquiteturas LSTM e GRU em cenários controlados. A Tabela 1 apresenta os melhores resultados obtidos durante os treinamentos realizados.

Tabela 1 – Desempenho comparativo entre LSTM e GRU

Modelo	Tempo de processamento	RMSE	RMSE (%)	MAE	MAE (%)
LSTM	451,0 s	1,349 °C	7,843 %	1,111 °C	6,232 %
GRU	229,4 s	1,227 °C	6,616 %	0,972 °C	5,277 %

Fonte: Autoria própria (2025)

Os resultados revelam que ambos os modelos foram eficazes na previsão da temperatura máxima horária, com erros médios absolutos próximos de 1 °C (6,2% para o LSTM e 5,3% para o GRU). No entanto, o GRU mostrou-se superior ao alcançar menor RMSE e menor erro percentual, além de apresentar um processamento significativamente mais rápido. Dessa forma, o LSTM tende a ser mais adequado para aplicações centralizadas em servidores com maior capacidade de hardware, enquanto o GRU se mostra promissor para cenários de *edge computing*, em que dispositivos de menor potência processam dados localmente, como em estações meteorológicas ou sistemas de monitoramento energético.

As próximas etapas incluem a continuidade dos testes dos modelos e hiperparâmetros, a integração com dados de dispositivos IoT reais, a implementação de mecanismos de gestão de treinamentos e modelos para versionamento e reaproveitamento, a otimização dos fluxos de execução do código para reduzir tempos de resposta e, por fim, a revisão da documentação técnica e dos entregáveis, assegurando clareza, reprodutibilidade e suporte à continuidade da pesquisa em diferentes contextos urbanos.

Referências

BARRETO, T.; BERNARDINI, F.; OLIVEIRA, D. DE. Previsão de demanda de água potável em Cidades Inteligentes por meio do algoritmo de modelagem de séries temporais PROPHET. Anais do XII Workshop de Computação Aplicada em Governo Eletrônico (WCGE 2024). **Anais...**Sociedade Brasileira de Computação, 21 jul. 2024.

CHO, K. et al. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. **arXiv preprint**, arXiv:1406.1078, 2014, 3 set. 2014.

DUTTA BARUAH, R.; MUÑOZ ORGANERO, M. Explicit Context Integrated Recurrent Neural Network for applications in smart environments. **Expert Systems with Applications**, v. 255, p. 124752, dez. 2024.

FIWARE FOUNDATION. **FIWARE**. Disponível em: <https://fiware.org>. Acesso em: 18 ago. 2025.

HOCHREITER, S.; SCHMIDHUBER, J. Long Short-Term Memory. **Neural Computation**, v. 9, n. 8, p. 1735–1780, 1 nov. 1997.

LEE, J. et al. Smart city as a social transition towards inclusive development through technology: a tale of four smart cities. **International Journal of Urban Sciences**, v. 27, n. sup1, p. 75–100, 1 jan. 2023.

MIGLIATO, A. L.; PONTI, M. Estudo de predição em séries temporais sazonais comparando SARIMA e redes neurais LSTM e GRU. Anais do I Workshop de Matemática, Estatística e Computação Aplicadas à Indústria. **Anais...**Galoa, 2021.

PAPASTEFANOPOULOS, V. et al. Multivariate Time-Series Forecasting: A Review of Deep Learning Methods in Internet of Things Applications to Smart Cities. **Smart Cities**, v. 6, n. 5, p. 2519–2552, 23 set. 2023.

SANG, G. M. et al. **Towards Predictive Maintenance for Flexible Manufacturing Using FIWARE**. In: [s.l: s.n.]. p. 17–28.

SANTOS, A. S. DOS; RIBEIRO FILHO, N. P. O futuro das cidades: Desafios e oportunidades das soluções digitais em ambientes urbanos. **Revista ft**, v. 29, n. 140, p. 22–23, 12 nov. 2024.

SHRESTHA, A.; MAHMOOD, A. Review of Deep Learning Algorithms and Architectures. **IEEE Access**, v. 7, p. 53040–53065, 2019.