UFMG - Universidade Federal de Minas Gerais
PPGEE - Programa de Pós-Graduação em Engenharia Elétrica
MINDS - Machine Intelligence and Data Science Laboratory

Proposta de Dissertação de Mestrado: Detecção de Intrusão Baseada em Anomalias em Sistemas IoT com Clustering Evolutivo e Arquitetura de Alto Desempenho em Fluxos

Aluno: Augusto Custodio Vicente - 2024663200
Orientador: Frederico Gadelha Guimarães
Co-orientador: Renata Lopes Rosa

1 de junho de 2025

1 Introdução

O crescimento da Internet das Coisas (IoT) tem proporcionado novas oportunidades para aplicações em diferentes setores, como indústria, saúde e agricultura, ao mesmo tempo em que traz maiores desafios de segurança [Benkhelifa et al., 2018, Golestani and Makaroff, 2024]. A heterogeneidade dos dispositivos IoT, aliada aos grandes volumes de dados gerados em tempo real, faz com que sistemas de Detecção de Intrusão (Intrusion Detection Systems – IDS) baseados em assinaturas se tornem insuficientes para lidar com ataques novos ou pouco conhecidos [Inuwa and Das, 2024, Ahmad et al., 2021].
Nesse contexto, a detecção de anomalias em fluxos de dados (data streams) surge como alternativa promissora [Park, 2018], permitindo analisar comportamentos atípicos que fogem ao padrão normal de funcionamento da rede [Saied et al., 2024]. Entretanto, a alta velocidade de chegada dos pacotes e a grande variabilidade do tráfego impõem restrições

de tempo de resposta e de uso eficiente dos recursos computacionais [Chen et al., 2025, Ogobuchi, 2022].
Visando maior adaptabilidade a mudanças súbitas (conceitos mutáveis, também chamados de *concept drifts*), métodos de *clustering* evolutivos têm sido propostos [Maia et al., 2020], lidando com fluxos de dados em cenários reais. Além disso, há a necessidade de arquiteturas de alto desempenho para ingestão e processamento contínuo, como abordado em Surianarayanan et al. [2024], que sugerem uma solução de alto throughput com uso de sistemas de mensageria (por exemplo, Apache Kafka) e algoritmos de aprendizado de máquina em paralelo para detecção de anomalias.

1.1 Objetivos

O objetivo geral deste trabalho é desenvolver um sistema de detecção de intrusão baseado em anomalias para redes IoT, empregando técnicas de *clustering* evolutivo em fluxos de dados e uma arquitetura de alto desempenho para lidar com o volume e a velocidade característicos de ambientes IoT.
Como objetivos específicos, destacam-se:

- **Reduzir Falsos Positivos:** Investigar estratégias que diminuam o índice de alarmes indevidos, melhorando a confiabilidade do sistema [Sharma et al., 2024, Chen et al., 2025].
- **Reduzir Custo Computacional:** Propor algoritmos e estruturas de dados eficientes, viabilizando a execução em dispositivos IoT ou sistemas de borda [Olanrewaju-George and Pranggono, 2025, Cook et al., 2020].
- **Arquitetura em Duas Partes:** Inspirar-se em Park [2018] e Surianarayanan et al. [2024] para construir uma solução com ingestão e processamento alternados, onde um estágio realiza a identificação de tráfego anômalo e outro classifica ou investiga mais detalhadamente o tipo de anomalia, de forma a equilibrar latência e precisão.
- **Detecção em Tempo Real:** Validar o desempenho em cenários de streaming de dados, atendendo aos requisitos de baixa latência em aplicações IoT críticas [Nguyen et al., 2019, Surianarayanan et al., 2024].
- **Empregar Clustering Evolutivo:** Utilizar o método proposto em Maia et al. [2020] para lidar com mudanças nas distribuições dos dados (*concept drifts*) e possibilitar a adaptação dinâmica sem reconfigurações constantes.
- **Abordar Dispositivos Heterogêneos:** Incorporar a abordagem de modelos focados em dispositivos ou tipos de dispositivos [Golestani and Makaroff, 2024], permitindo maior granularidade e reduzindo possíveis imprecisões causadas por agregar dados muito diferentes em um único classificador.

- **Prevenir Sobrecarga em Fluxos de Dados**: Implementar mecanismos de alto throughput (Kafka ou similares) para ingestão dos dados de forma escalável, baseando-se na arquitetura descrita em Surianarayanan et al. [2024].
- **Tratar Balanceamento de Classes**: Analisar métodos de pré-processamento ou thresholds adaptativos para lidar com situações em que o tráfego benigno ou malicioso predomine significativamente [Ahmad et al., 2021, Ogobuchi, 2022].
- **Realizar Avaliação em Dados Reais ou Públicos**: Testar e comparar o sistema com abordagens existentes, avaliando métricas de acurácia, taxa de falsos positivos, tempo de resposta e escalabilidade [Sharma et al., 2024, Alqahtany et al., 2025].

2 Revisão Bibliográfica

A literatura sobre detecção de intrusão em sistemas IoT destaca diferentes abordagens para lidar com fluxos contínuos de dados, variando desde aprendizado de máquina supervisionado e não supervisionado até arquiteturas específicas para ingestão em alta velocidade.

(a) Arquiteturas de Alto Desempenho em Fluxos

Conforme Surianarayanan et al. [2024], uma arquitetura de alta taxa de transferência (*high-throughput*) é essencial para lidar com a chegada de dados em grandes volumes e velocidades. O modelo de publicação e assinatura (*publish-subscribe*), aliado à discretização inteligente do fluxo, permite que algoritmos de detecção de anomalias sejam executados em paralelo com menor latência. Tal abordagem mostra-se relevante para sistemas IoT, pois evita perda de dados e possibilita análises rápidas.

(b) Análise de Anomalias em Dados de Streaming

Estudos apontam a necessidade de lidar com cenários de *concept drift*, onde a distribuição de dados pode mudar ao longo do tempo. Maia et al. [2020] propuseram um algoritmo de *clustering* evolutivo capaz de ajustar parâmetros de forma autônoma, dispensando múltiplas reconfigurações e armazenamentos de alta escala. Paralelamente, Park [2018] destaca o desafio de detectar não apenas outliers pontuais, mas também padrões anômalos que ocorram em determinados intervalos de tempo ou mudanças de comportamento em janelas de dados.

(c) Modelos Focados em Dispositivos IoT

Outro fator importante está na diversidade de dispositivos IoT. Golestani and Makaroff [2024] investigaram a construção de modelos específicos para cada dispositivo ou tipo de dispositivo, evidenciando que essa segmentação pode melhorar a detecção de anomalias, principalmente em cenários onde o tráfego é dominado por um único tipo (malicioso ou

benigno). Em tais casos, algoritmos de uma classe, como *One-Class Classifiers*, podem se mostrar eficazes.

(d) Integração com Técnicas de Aprendizado de Máquina

Para a detecção de anomalias em tempo real, algoritmos como Random Forest, SVM e redes neurais (profundas ou não) aparecem como ferramentas comuns [Chen et al., 2025, Ahmad et al., 2021]. Contudo, os trabalhos de Maia et al. [2020] e Park [2018] ressaltam a importância de algoritmos mais adaptativos, que possam lidar com *drift* e grandes volumes de dados sem perda de desempenho.

(e) Desafios e Tendências

Há um consenso sobre a importância de:

- **Estruturas escaláveis** que suportem a alta velocidade de dados, evitando latências elevadas [Surianarayanan et al., 2024].
- **Adaptação a mudanças** (ou *drift*), o que requer métodos evolutivos e automatizados, como no caso de Maia et al. [2020].
- **Foco em dispositivos específicos**, especialmente em cenários com alto desequilíbrio de classes ou grande heterogeneidade de tráfego [Golestani and Makaroff, 2024].
- **Manutenção de baixas taxas de falsos positivos**, dada a criticidade de alarmes em ambientes IoT [Sharma et al., 2024, Chen et al., 2025].

3 Metodologia

A metodologia proposta para desenvolver um IDS baseado em anomalias em dispositivos IoT abrange os seguintes estágios:

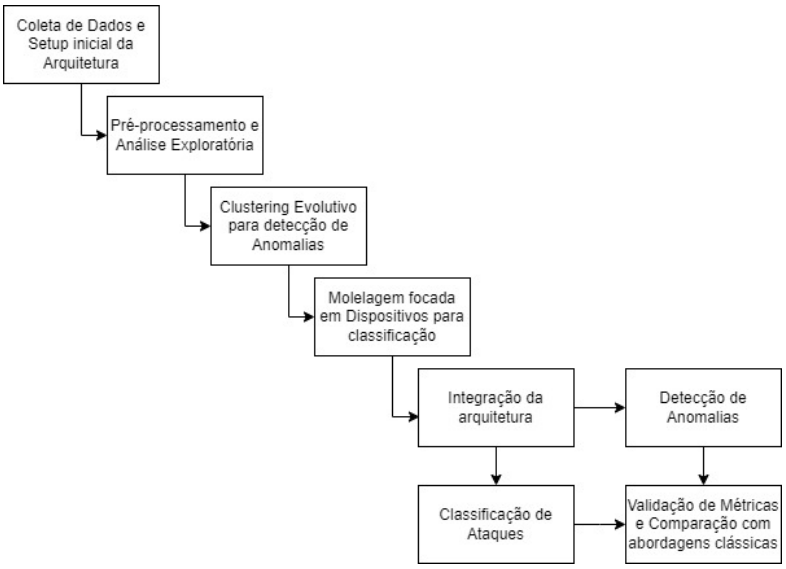

Figura 3.1: Metodologia Proposta. Fonte: Autor

1. **Coleta de Dados e Setup Inicial da Arquitetura:**
   - Utilizar sistemas de mensageria como o Apache Kafka, conforme sugerido por Surianarayanan et al. [2024], para receber e gerenciar fluxos contínuos de dados.
   - Implementar discretizações do fluxo (janelas de tempo ou contagem de pacotes) para permitir o processamento simultâneo em diferentes nós.
   - O estudo pretende usar os datasets presentes na literatura: CICIoT2023 [Neto et al., 2023], CICIDS2018 [for Cybersecurity, 2018], CICIDS2017 [for Cybersecurity, 2017]
2. **Pré-Processamento e Análise Exploratória:**
   - Aplicar estratégias de limpeza de dados, detecção e remoção de ruído excessivo ou dados faltantes.
   - Lidar com cenários de desequilíbrio de classes (ataques vs. tráfego benigno), possivelmente por meio de *thresholds* adaptativos [Ahmad et al., 2021, Ogobuchi, 2022].
3. **Clustering Evolutivo para Detecção de Anomalias:**

- Empregar abordagem inspirada no *Mixture of Typicalities*, proposta em Maia et al. [2020], segmentando os dados em micro e macro-clusters, de modo a lidar com a variabilidade do fluxo (concept drift) de forma autônoma.
- Adaptar o método para atualizar os clusters em tempo real, sem necessidade de reconfigurações manuais.

4. **Modelagem Focada em Dispositivos para classificação:**
   - Aplicar a abordagem de modelos específicos ou por tipo de dispositivo, conforme Golestani and Makaroff [2024], quando ocorrer grande disparidade entre padrões de tráfego.
   - Viabilizar detecções mais precisas em cenários com dispositivos homogêneos, porém usando fluxos distintos.
5. **Integração da Arquitetura:**
   - **Detecção de Anomalias:** Um subsistema avalia se cada lote (batch) de dados ou cada janela de fluxo corresponde ao comportamento normal ou fora do padrão, usando o *clustering* evolutivo, gerando um alerta em caso de anomalias no fluxo de dados.
   - **Detalhamento do Ataque (Opcional):** Caso haja suspeita de anomalia, outro subsistema aplica algoritmos mais específicos (por exemplo, Random Forest, CNN, OCC) para classificar ou detalhar o tipo de ataque [Surianarayanan et al., 2024, Ogobuchi, 2022], .

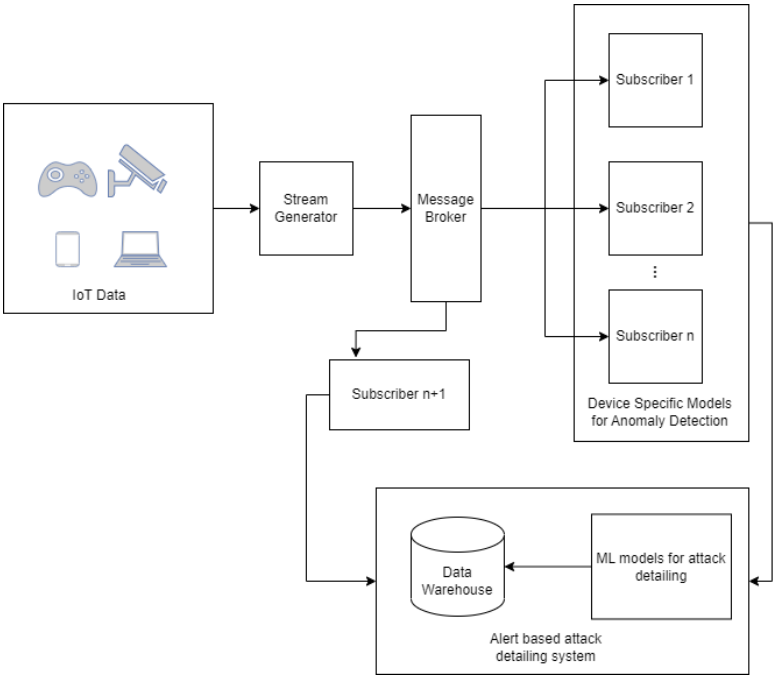

Figura 3.2: Arquitetura Proposta. Fonte: Adaptado de [Surianarayanan et al., 2024]

6. **Validação e Métricas e comparação com abordagens clássicas:**
    - Avaliar o sistema em termos de acurácia, taxa de falsos positivos, tempo de resposta (latência) e vazão suportada (throughput) e comparar com abordagens clássicas. [Sharma et al., 2024, Chen et al., 2025].
    - Realizar experimentos com configurações em nós únicos e distribuídos, medindo possíveis melhorias de escalabilidade, de maneira similar ao proposto em Surianarayanan et al. [2024].

Esse método visa compor uma solução escalável para detecção de anomalias em tempo real, capaz de se adaptar às mudanças no padrão de tráfego e com suporte a diferentes perfis de dispositivo, mantendo baixo custo computacional e baixa latência em ambientes IoT.

3.1 Ferramentas

1. **Ferramentas de hardware para execução da pesquisa:**
    - Central Processing Unit: Intel(R) Core(TM) i5-10210 CPU @ 1.60GHz, 2.11 GHz, 4 Core(s).
    - Random Access Memory: 16 GB (15.8 GB usable).
    - Operating System: Windows 11 64-bit.
    - Graphics Processing Unit: NVIDIA GeForce MX 110.
2. **Ferramentas de software para execução da pesquisa:**
    - Sklearn - Biblioteca de Machine Learning.
    - Pandas - Biblioteca de análise de dados.
    - Matplotlib - Biblioteca para plotagem de gráficose visualização de dados.
    - TensorFlow e Keras - Frameworks de Deep Learning.
    - Flask/Fastapi - Frameworks para apis REST.
    - Apache Kafka e Apache Kafka Streams- Software para arquitetura baseada em mensagens e geração de streams de dados respectivamente.
    - Docker e Kubernetes - Softwares para criação e orquestração de containers.

4 Cronograma

Tabela 4.1: Cronograma de Atividades (Estimativa para 12 Meses)

Atividade	**Meses**											
	1	2	3	4	5	6	7	8	9	10	11	12
Disciplinas	X	X	X	X								
Revisão Bibliográfica	X	X	X	X	X							
Coleta e Setup de Arquitetura			X	X	X							
Pré-Processamento e Anál. Expl.			X	X	X							
Desenvolvimento do Algoritmo Evolutivo			X	X	X	X						
Integrar Arquitetura (Detecção e Classificação)					X	X	X					
Modelos Focados em Dispositivos						X	X	X				
Testes e Avaliação de Métricas								X	X	X		
Documentação e Redação								X	X	X	X	
Depósito e Defesa										X	X	X

Referências

Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah, and F. Ahmad. Network intrusion detection system: A systematic study of machine learning and deep learning approaches. *Transactions on Emerging Telecommunications Technologies*, 32(1), 2021. doi: 10.1002/ett.4150.

S. S. Alqahtany, A. Shaikh, and A. Alqazzaz. Enhanced grey wolf optimization (egwo) and random forest based mechanism for intrusion detection in iot networks. *Scientific Reports*, 15(1):1916, 2025. doi: 10.1038/s41598-024-81147-x.

E. Benkhelifa, T. Welsh, and W. Hamouda. A critical review of practices and challenges in intrusion detection systems for iot: Toward universal and resilient systems. *IEEE Communications Surveys & Tutorials*, 20(4):3496–3509, 2018.

H. Chen, Z. Wang, S. Yang, X. Luo, D. He, and S. Chan. Intrusion detection using synaptic intelligent convolutional neural networks for dynamic internet of things environments. *Alexandria Engineering Journal*, 111:78–91, 2025. ISSN 1110-0168. doi: https://doi.org/10.1016/j.aej.2024.10.014.

A. A. Cook, G. Mısırlı, and Z. Fan. Anomaly detection for iot time-series data: A survey. *IEEE Internet of Things Journal*, 7(7):6481–6494, 2020. doi: 10.1109/JIOT.2019.2958185.

C. I. for Cybersecurity. CICIDS2017 Dataset. `https://www.unb.ca/cic/datasets/ids-2017.html`, 2017. Accessed on 20 February 2025.

C. I. for Cybersecurity. CICIDS2018 Dataset. https://www.unb.ca/cic/datasets/ids-2018.html, 2018. Accessed on 20 February 2025.

S. Golestani and D. Makaroff. Device-specific anomaly detection models for iot systems. In *2024 IEEE Conference on Communications and Network Security (CNS)*, pages 1–6, 2024. doi: 10.1109/CNS62487.2024.10735608.

M. M. Inuwa and R. Das. A comparative analysis of various machine learning methods for anomaly detection in cyber attacks on iot networks. *Internet of Things*, 26:101162, 2024. ISSN 2542-6605. doi: https://doi.org/10.1016/j.iot.2024.101162.

J. Maia, C. A. Severiano, F. G. Guimarães, C. L. de Castro, A. P. Lemos, J. C. Fonseca Galindo, and M. Weiss Cohen. Evolving clustering algorithm based on mixture of typicalities for stream data mining. *Future Generation Computer Systems*, 106:672–684, 2020. ISSN 0167-739X. doi: https://doi.org/10.1016/j.future.2020.01.017. URL https://www.sciencedirect.com/science/article/pii/S0167739X19312786.

E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani. Ciciot2023: A real-time dataset and benchmark for large-scale attacks in iot environment. *Sensors*, 23(13), 2023. ISSN 1424-8220. doi: 10.3390/s23135941. URL https://www.mdpi.com/1424-8220/23/13/5941.

T. D. Nguyen, S. Marchal, M. Miettinen, H. Fereidooni, N. Asokan, and A.-R. Sadeghi. DÏot: A federated self-learning anomaly detection system for iot. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pages 756–767, 2019. doi: 10.1109/ICDCS.2019.00080.

D. O. Ogobuchi. Multi-phase optimized intrusion detection system based on deep learning algorithms for computer networks. 2022. Master's Thesis.

B. Olanrewaju-George and B. Pranggono. Federated learning-based intrusion detection system for the internet of things using unsupervised and supervised deep learning models. *Cyber Security and Applications*, 3:100068, 2025. ISSN 2772-9184. doi: https://doi.org/10.1016/j.csa.2024.100068.

C. H. Park. Anomaly pattern detection on data streams. In *2018 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pages 689–692, 2018. doi: 10.1109/BigComp.2018.00127.

M. Saied, S. Guirguis, and M. Madbouly. Review of artificial intelligence for enhancing intrusion detection in the internet of things. *Engineering Applications of Artificial Intelligence*, 127:107231, 2024. ISSN 0952-1976. doi: https://doi.org/10.1016/j.engappai.2023.107231.

B. Sharma, L. Sharma, C. Lal, and S. Roy. Explainable artificial intelligence for intrusion detection in iot networks: A deep learning based approach. *Expert Systems with Applications*, 238:121751, 2024. ISSN 0957-4174. doi: https://doi.org/10.1016/j.eswa.2023.121751.

C. Surianarayanan, S. Kunasekaran, and P. R. Chelliah. A high-throughput architecture for anomaly detection in streaming data using machine learning algorithms. *International Journal of Information Technology*, 16(1):493–506, 2024. ISSN 2511-2112. doi: 10.1007/s41870-023-01585-0. URL `https://doi.org/10.1007/s41870-023-01585-0`.