

UNIVERSIDADE FEDERAL DO PARANÁ

ANGELITA RETTORE DE ARAUJO ZANELLA

DETECTOR HÍBRIDO DE ANOMALIAS PARA AGRICULTURA INTELIGENTE

CURITIBA PR

2022

ANGELITA RETTORE DE ARAUJO ZANELLA

DETECTOR HÍBRIDO DE ANOMALIAS PARA AGRICULTURA INTELIGENTE

Tese apresentada como requisito parcial à obtenção do grau de Doutor em Ciência da Computação no Programa de Pós-Graduação em Informática, Setor de Ciências Exatas, da Universidade Federal do Paraná.

Área de concentração: *Ciência da Computação*.

Orientador: Luiz Carlos Pessoa Albini.

Coorientador: Eduardo Silva.

CURITIBA PR

2022

DADOS INTERNACIONAIS DE CATALOGAÇÃO NA PUBLICAÇÃO (CIP)
UNIVERSIDADE FEDERAL DO PARANÁ
SISTEMA DE BIBLIOTECAS – BIBLIOTECA CIÊNCIA E TECNOLOGIA

Zanella, Angelita Rettore de Araujo
Detector híbrido de anomalias para agricultura inteligente. / Angelita
Rettore de Araujo Zanella. – Curitiba, 2022.
1 recurso on-line : PDF.

Tese (Doutorado) - Universidade Federal do Paraná, Setor de
Ciências Exatas, Programa de Pós-Graduação Informática.

Orientador: Luiz Carlos Pessoa Albini.
Coorientador: Eduardo Silva.

1. Agricultura. 2. Internet das coisas. 3. Informática (Confiabilidade).
4. Informática (Detecção de intrusões). I. Albini, Luiz Carlos Pessoa. II.
Silva, Eduardo. III. Universidade Federal do Paraná. Programa de Pós-
Graduação em Informática. IV. Título.

Bibliotecária: Roseny Rivelini Morciani CRB-9/1585

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação INFORMÁTICA da Universidade Federal do Paraná foram convocados para realizar a arguição da tese de Doutorado de **ANGELITA RETTORE DE ARAUJO ZANELLA** intitulada: **DETECTOR HÍBRIDO DE ANOMALIAS PARA AGRICULTURA INTELIGENTE**, sob orientação do Prof. Dr. LUIZ CARLOS PESSOA ALBINI, que após terem inquirido a aluna e realizada a avaliação do trabalho, são de parecer pela sua **APROVAÇÃO** no rito de defesa.

A outorga do título de doutora está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

CURITIBA, 22 de Setembro de 2022.

Assinatura Eletrônica

26/09/2022 17:49:03.0

LUIZ CARLOS PESSOA ALBINI

Presidente da Banca Examinadora

Assinatura Eletrônica

23/09/2022 15:54:11.0

LUCAS DIAS HIERA SAMPAIO

Avaliador Externo (UNIVERSIDADE TECNOLÓGICA FEDERAL DO
PARANÁ - UTFPR)

Assinatura Eletrônica

19/10/2022 14:56:43.0

ANDRÉ SANDMANN

Avaliador Externo (UNIVERSIDADE TECNOLÓGICA FEDERAL DO
PARANÁ - UTFPR)

Assinatura Eletrônica

23/09/2022 09:03:44.0

CARLOS ALBERTO MAZIERO

Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

Para Fabrício, meu esposo e fiel companheiro, que com muita paciência e carinho apoiou durante toda a jornada. Aos meus filhos, Pedro e Maria Luiza, que alegraram meus dias.

*A coragem é a primeira das qualidades
humanas porque garante todas as outras.*

Winston Churchill

AGRADECIMENTOS

Agradeço à Deus, por ter me guiado nesta caminhada e por me dar forças para seguir nos momentos difíceis. Ao meu querido esposo, Fabrício, pelo companheirismo e por estar sempre ao meu lado, incentivando a seguir em frente. Seu apoio incondicional, seu companheirismo, amizade, paciência e compreensão foram fundamentais para eu alcançar mais essa vitória. Aos meus pais, pelo exemplo de vida e contínuo apoio.

Ao amigo Tiago Possato, pelo convite para trabalhar com Internet das Coisas e pelos anos de trabalho no projeto voltado à Agricultura Inteligente. Obrigada pela ajuda na construção dos sensores e pelo desenvolvimento dos códigos integrados aos equipamentos. Agradeço aos professores Luiz Carlos Pessoa Albini e Eduardo Silva por acreditarem no projeto e por aceitarem me orientar nessa pesquisa. Obrigada por sua dedicação e preciosa ajuda durante esse período. Aos colegas e amigos, pelos conhecimentos, momentos e experiências compartilhadas. Aos funcionários do programa de pós-graduação, pela disponibilidade, simpatia e gentileza.

RESUMO

As mudanças climáticas, a crise da água e o crescimento populacional acrescentam novos desafios para a produção de alimentos. O aumento das taxas de produção e a preservação de recursos naturais dependem da modernização dos métodos agrícolas. A agricultura inteligente fornece recursos capazes de melhorar a atividade agrícola por meio do controle eficiente de atuadores, otimizando o consumo de recursos e o gerenciamento da produção, otimizando seus resultados. Para que estas tecnologias se tornem populares, elas devem ter um alto nível de confiabilidade e segurança, o que não é observado nos sistemas desenvolvidos até o momento. Para melhorar a confiabilidade na Agricultura Inteligente, este documento propõe o CEIFA, um detector de anomalias híbrido de baixo custo capaz de identificar falhas, erros e ataques que afetam estes sistemas. Diferentemente dos detectores já desenvolvidos, que tem seu escopo limitado à detecção de intrusões, o sistema proposto tem como alvo anomalias decorrentes de falhas e erros que acometem os dispositivos de coletas de dados e os ataques dentro do escopo do cibergroterrorismo, podendo detectar outros ataques que causem alterações nos dados. O CEIFA utiliza uma arquitetura híbrida, combinando operação na borda e na nuvem. Sua arquitetura modular permite que o sistema seja ajustado às necessidades e contextos do sistema agrícola, que pode utilizar dispositivos com recursos computacionais limitados, servidores locais ou na nuvem. A detecção é feita analisando os dados enviados pelos sensores do sistema agrícola. A identificação de anomalias é feita por meio de análise estatística, correlação de dados e aprendizagem de máquina. A análise estatística incorpora avaliação de tendências e dispersão de dados, em um modelo matemático capaz de identificar falhas e erros de forma eficiente e com baixo custo computacional. Para maximizar a precisão e possibilitar a detecção de intrusões, foram incluídas a correlação de dados e aprendizagem de máquina. A construção do sistema utilizou como base uma rede composta por cinco transdutores, equipados com sensores que coletaram dados climáticos e ambientais. Os transdutores foram instalados em ambiente agrícola em campo aberto, expostos aos eventos climáticos e ambientais, bem como ação de animais e humanos. Os sensores foram alvos de falhas e erros que permitiram mapear os padrões de dados. Ao final da coleta de dados, não havia dispositivos íntegros em quantidade suficiente para serem utilizados nas fases seguintes. Para validar o sistema, foi utilizado um protótipo, composto por 19 sensores virtuais, um dispositivo de borda e um servidor na nuvem. Os resultados mostram que o detector utiliza pouca memória, apresenta baixos níveis de processamento e ocupa pouco espaço em disco. Na borda os dados demoraram 355ms para serem processados, utilizando 111MB de memória (considerando todas as operações de leitura e escrita) e 262KB de espaço em disco. Na nuvem, os mesmos dados são processados em 4ms, consumindo 120MB de memória e ocupando 2MB de espaço em disco. Quanto à acurácia, o detector superou 95% de acertos na detecção de anomalias. A borda identificou 80% das anomalias alvo e a nuvem 93%. O CEIFA é, portanto, um detector de anomalias eficiente e de baixo custo, capaz de economizar recursos com alta eficiência e latência reduzida.

Palavras-chave: agricultura inteligente, Internet das Coisas, segurança, confiabilidade, detecção de anomalias, detecção de intrusões.

ABSTRACT

Climate change, the water crisis, and population growth include new challenges for food production. Increasing production rates while preserving natural resources requires the modernization of farming methods. Smart agriculture allows creating resources capable of improving agricultural activity. It permits the efficient control of actuators, optimizing resource consumption and production management, maximizing profit, and minimizing costs. For these technologies to become popular, they must have a high level of reliability and security, something lacking in the systems developed so far. This paper proposes CEIFA, a low-cost hybrid anomaly detector capable of identifying faults, errors, and attacks that affect these systems aiming to improve reliability in Smart Farming. Unlike existing anomaly detectors, which set out to detect intrusions, the CEIFA seeks to identify random failures, occurrences of saturation, degradation, damage, noise, and false data injection. It uses a hybrid architecture, combining operations at the edge and on the cloud. Its modular architecture permits adjusting the system to the needs and context of the farming system, which can use devices with limited computing resources, local servers, or the cloud. CEIFA performs detection by analyzing the data sent by the farming system's sensors. It integrates data analysis, trend analysis, and data scattering with machine learning to identify anomalies. The modeling of the data generated by agricultural systems allowed creating a model capable of recognizing some anomalies. By joining the mathematical model, machine learning, and data analysis, it was possible to maximize the precision of the detector. The system design uses a network comprising five transducers with sensors for climatic and environmental parameters. The transducers were installed in an open field agricultural environment and exposed to climatic and weather events, animal and human action. The sensors were subject to fails, faults and errors that allowed mapping the behavior of the actual data. After data collection, there were insufficient operational devices to continue using the system. Validation used a prototype comprising 19 virtual sensors, an edge device, and a virtual server in the cloud. The results show lower memory, processing, and disk space consumption. The edge processes data in 355ms (CPU time), using 400KB of memory, 262KB of disk space. In the cloud, processing usage 4ms of CPU time, 6MB of memory, and 2MB of disk space. With these capabilities, the edge can identify 80% of the target anomalies and the cloud 93%, which gives the detector efficiency of over 95% in anomaly detection. CEIFA is an efficient, low-cost anomaly detector capable of saving resources with high efficiency and reduced latency.

Keywords: smart agriculture, Internet of Things, security, reliability, anomaly detection, intrusion detection.

LISTA DE FIGURAS

1.1	Dezessete Objetivos de Desenvolvimento Sustentável	16
2.1	Revoluções Agrícolas	23
2.2	Estrutura dos Componentes de Agricultura Inteligente.	26
2.3	Riscos e ameaças à agricultura inteligente	28
2.4	Sensor de umidade do solo instalado dentro de uma estufa agrícola	42
2.5	Leituras de umidade relativa do ar	43
3.1	Fatores que influenciam a detecção de anomalias	45
3.2	Sistema de Detecção de Intrusão Simples	47
3.3	Classificação dos Sistemas de Detecção de Intrusão	48
3.4	Arquiteturas IDS	50
3.5	Tipos de Sistemas de Detecção de Intrusão.	52
4.1	Exemplo de falha aleatória de sensor	60
4.2	Exemplo de sensor saturado	61
4.3	Sensores de umidade do solo prontos para instalação	62
4.4	Sensores de umidade do solo degradados.	63
4.5	Arquitetura do detector de anomalias proposto	66
4.6	Componentes do detector de anomalias	67
4.7	Arquitetura do Edge Core.	68
4.8	Fluxo de Processamento do Edge Core	71
4.9	Arquitetura do Edge Data Analytics	72
4.10	Esquema de ações do módulo Decisões	76
5.1	Transdutor composto por sensores de parâmetros ambientais	80
5.2	Sensores utilizados no projeto	81
5.3	Precisão para detecção de falhas aleatórias	82
5.4	Falhas Aleatórias: dados classificados corretamente	83
5.5	Precisão para detecção de saturação	83
5.6	Saturação: dados classificados corretamente	84
5.7	Precisão para detecção de choques e manuseio não autorizado	84
5.8	Choques e manuseio não autorizado: dados classificados corretamente	85
5.9	Precisão para detecção de degradação	85
5.10	Degradação: dados classificados corretamente	86

5.11	Precisão para detecção de ruídos	86
5.12	Ruídos: dados classificados corretamente	87
5.13	Precisão para detecção de injeção de dados falsos	87
5.14	Erros e acertos durante a detecção de injeção de dados falsos	88
5.15	Pympler: Memória alocada pelos objetos	89
5.16	Guppy: alocação de memória na borda	90
5.17	Guppy: alocação de memória na nuvem	90
5.18	Processo: Memória alocada na borda.	91
5.19	Processo: Memória alocada na nuvem	92
5.20	Tempos de processamento dos filtros do <i>Edge Core</i>	92
5.21	Tempos de CPU gastos pelo <i>Edge Core</i> , <i>Edge Data Analytics</i> e <i>Naive Bayes</i> para Fluxos de Dados.	93
5.22	Tempos de CPU gastos pelo <i>Cloud Data Analytics</i> e Árvores de Hoeffding	94
5.23	Classificação dos dados Normais	96
5.24	Dados anômalos classificados como anomalia pelo CEIFA	97
5.25	Classificação das Falhas Aleatórias.	97
5.26	Classificação das Saturações	98
5.27	Classificação das Degradações	99
5.28	Classificação das Danificações	99
5.29	Classificação dos Ruídos	100
5.30	Classificação de Injeção de Dados Falsos - Cenário 1	100
5.31	Classificação de Injeção de Dados Falsos - Cenário 2	101
C.1	Memory-profiler: memória alocada pelos algoritmos de aprendizagem de máquina para série de dados	121
C.2	Guppy: memória alocada pelos algoritmos de aprendizagem de máquina para série de dados	122
C.3	Processamento consumido pelos algoritmos de aprendizagem de máquina para série de dados	123
C.4	Precisão alcançada pelos algoritmos de aprendizagem de máquina	123
C.5	Memory-profiler: memória alocada pelos algoritmos de aprendizagem de máquina para fluxos de dados.	124
C.6	Guppy: memória alocada pelos algoritmos de aprendizagem de máquina para fluxos de dados	125
C.7	Processamento consumido pelos algoritmos de aprendizagem de máquina para fluxos de dados	125
C.8	Memory-profiler: comparação do consumo de memória dos Shapelets e dos algoritmos para fluxos de dados	126
C.9	Guppy: comparação do consumo de memória dos Shapelets e dos algoritmos para fluxos de dados.	126

C.10	Comparação do processamento utilizado pelos Shapelets e pelos algoritmos para fluxos de dados	127
C.11	Memória alocada para classificação de valores únicos e múltiplos.	127
C.12	Memória total alocada para classificação de valores únicos e múltiplos	128
C.13	Pympier: memória alocada pelos objetos.	128
C.14	Processamento utilizado durante a classificação de valores únicos e múltiplos . .	128
C.15	Processamento total utilizado durante a classificação de valores únicos e múltiplos	129

LISTA DE TABELAS

2.1	Elementos de Sistemas Agrícolas inteligentes	27
2.2	Taxonomia de Segurança na Agricultura Inteligente	38
2.3	Recursos de segurança implementados em Sistemas Agrícolas	41
3.1	Vantagens e desvantagens dos métodos de detecção de anomalias.	49
3.2	Vantagens e desvantagens das arquiteturas dos sistemas de detecção de anomalias	52
3.3	Características dos tipos de sistemas de detecção de intrusão	53
5.1	Composição dos conjuntos de dados utilizados para análise de precisão	81
5.2	Índice de precisão dos algoritmos	88
5.3	Precisão na classificação de dados “anômalos”.	88
5.4	Precisão na classificação de dados “normais”	89
5.5	Mapeamentos realizados pelo smaps	91
5.6	Temperaturas (°C) registradas no dia 16 de fevereiro de 2021	95
5.7	Classificação dos dados anômalos pelos módulos do CEIFA	102
D.1	Orçamento Transdutores de Rede.	130

LISTA DE ACRÔNIMOS

APIDS	IDS baseado em protocolos de aplicações.
CDA	Cloud Data Analytics.
CEIFA	<i>Cloud-Edge Identifier of Farming Anomalies.</i>
CML	Cloud ML.
CVE	<i>Common Vulnerabilities and Exposures.</i>
DDoS	ataque distribuído de negação de serviço.
DoS	negação de serviço.
EC	Edge Core.
EDA	Edge Data Analytics.
EML	Edge ML.
FAO	Organização das Nações Unidas para a Alimentação e a Agricultura.
FSM	máquina de estados finitos.
HIDS	IDS baseado em <i>host</i> .
HT	Árvores de Hoeffding.
IDS	sistema de detecção de intrusão.
IoT	Internet das Coisas.
ISP	provedor de serviços de internet.
KNN	<i>K-Nearest Neighbor.</i>
KNN-DS	KNN para fluxos de dados.
LPWA	baixa potência e área ampla.
M2M	máquina-a-máquina.
ML	aprendizagem de máquina.
Naïve Bayes-DS	Naïve Bayes para fluxos de dados.
NIDS	IDS baseado em rede.
ONU	Organização das Nações Unidas.
OWASP	<i>Open Web Application Security Project.</i>
PIDS	IDS baseado em protocolos.
QoS	qualidade do serviço.
RFID	identificação por radiofrequência.

RSSF rede de sensores sem fio.
SVM Máquinas de Vetores de Suporte.
WSAN redes de sensores e atuadores sem fio.

SUMÁRIO

1	INTRODUÇÃO	16
1.1	METODOLOGIA	20
1.2	PRINCIPAIS CONTRIBUIÇÕES	21
1.3	ESTRUTURA DO TEXTO	21
2	AGRICULTURA INTELIGENTE: FORNECENDO FERRAMENTAS PARA MODIFICAR OS PROCESSOS PRODUTIVOS	23
2.1	A INTEGRAÇÃO DE TECNOLOGIAS PARA UMA AGRICULTURA SUSTENTÁVEL	24
2.2	AMEAÇAS À SEGURANÇA DA AGRICULTURA INTELIGENTE	27
2.2.1	Ameaças à segurança na camada de percepção	28
2.2.2	Problemas de segurança na camada de rede	31
2.2.3	Problemas de segurança na borda da rede	32
2.2.4	Problemas de segurança na camada de aplicação	34
2.3	CONFIABILIDADE EM SISTEMAS AGRÍCOLAS INTELIGENTES	35
2.4	A AMEAÇA DO AGROTERRORISMO	36
2.4.1	Ciberagroterrorismo e a Agricultura Inteligente	37
2.5	PANORAMA DA SEGURANÇA INCORPORADA PELOS SISTEMAS AGRÍCOLAS INTELIGENTES	37
2.6	FALHAS E ERROS EM DISPOSITIVOS DE COLETA DE DADOS	42
3	DETECÇÃO DE ANOMALIAS	44
3.1	PANORAMA DA DETECÇÃO DE ANOMALIAS	45
3.2	VISÃO GERAL DOS SISTEMAS DE DETECÇÃO DE INTRUSÃO	47
3.2.1	Métodos de Detecção de Intrusão	48
3.2.2	Arquiteturas dos Sistemas de Detecção de Intrusão	50
3.2.3	Tipos dos Sistemas de Detecção de Intrusão	52
3.2.4	Detecção Ativa e Passiva	54
3.3	ARQUITETURAS PARA DETECTORES VOLTADOS À AGRICULTURA 4.0	55
4	CEIFA: UM DETECTOR DE ANOMALIAS PARA SISTEMAS AGRÍCOLAS DIGITAIS	57
4.1	FALHAS QUE AFETAM DISPOSITIVOS DE COLETA DE DADOS	57
4.2	MODELO DE ANOMALIAS	59
4.2.1	Falha aleatória de sensor	59
4.2.2	Sensor saturado	59
4.2.3	Sensor degradado ou obstruído	60
4.2.4	Sensor Danificado	62

4.2.5	Ruídos	64
4.2.6	Injeção de Dados Falsos	64
4.3	VISÃO GERAL DO DETECTOR DE ANOMALIAS	65
4.4	ARQUITETURA DO DETECTOR DE ANOMALIAS	65
4.4.1	<i>Edge Core</i> : usando análise estatística para detectar anomalias	67
4.4.2	<i>Edge Data Analytics</i> : analisando dados para maximizar a eficácia	71
4.4.3	<i>Edge ML</i> : utilizando a aprendizagem de máquina para melhorar a precisão	74
4.4.4	Utilizando Aprendizagem de Máquina para detectar anomalias	75
4.4.5	Módulo de decisões na borda	76
4.4.6	<i>Cloud Data Analytics</i> : analisando os dados na nuvem	77
4.4.7	<i>Cloud ML</i>	78
4.4.8	Módulo de Decisões da Nuvem	79
5	AVALIAÇÃO E RESULTADOS	80
5.1	ANÁLISE DE PRECISÃO	81
5.2	CONSUMO DE RECURSOS COMPUTACIONAIS DO <i>EDGE CORE</i>	88
5.3	RESULTADOS	94
5.3.1	Análise dos resultados	102
6	CONCLUSÃO	103
6.1	CONTRIBUIÇÕES	105
6.2	LIMITAÇÕES	105
6.3	TRABALHOS FUTUROS	106
	REFERÊNCIAS	107
	APÊNDICE A – REDES LOW-POWER WIDE-AREA (LPWA)	118
	APÊNDICE B – RASPBERRY PI 2 MODEL B - HARDWARE GENERAL SPECIFICATIONS	120
	APÊNDICE C – ANÁLISE DOS ALGORITMOS DE APRENDIZAGEM DE MÁQUINA	121
	APÊNDICE D – TABELA DE GASTOS.	130

1 INTRODUÇÃO

A agricultura é a principal fonte de alimentos em todo o mundo, desempenhando papel fundamental para o desenvolvimento econômico de qualquer nação (Rajalakshmi e Devi Mahalakshmi, 2016). Em um relatório sobre os desafios da agricultura, publicado em 2012 (Alexandratos e Bruinsma, 2012), a Organização das Nações Unidas para a Alimentação e Agricultura (FAO, do inglês *Food and Agriculture Organization*) declara ser necessário ampliar a produção mundial de alimentos em cerca de 70% até 2050, para suprir a demanda. Isso envolve aumentar a produção de cereais em cerca de 3 bilhões de toneladas e a de carne em 200%. Esses dados estão embasados nas projeções de crescimento populacional, ratificados por relatórios recentes (ONU, 2019; UNDESA, 2019), que preveem um aumento de mais de 2 bilhões de pessoas nas próximas três décadas, e nas taxas de produção e desperdícios de alimentos, gerados nos processos de produção, transporte e consumo.

Em 2015 a Organização das Nações Unidas (ONU) estabeleceu 17 metas para o desenvolvimento sustentável (Figura 1.1), com o objetivo de acabar com a pobreza, proteger o planeta e melhorar a vida na Terra. Para alcançar essas metas, foi estabelecida uma agenda, chamada Agenda 2030, que fornece um plano e um conjunto de objetivos globais para um período de 15 anos (ONU, 2015; United Nation, 2017). Entre os indicadores-chave criados para o alcance da meta, estão aqueles ligados à produção de alimentos, como volumes de produção, renda média dos pequenos produtores, agricultura sustentável e preservação de recursos naturais.

Figura 1.1: Dezesete Objetivos de Desenvolvimento Sustentável



FONTE: Adaptado de Nações Unidas Brasil (<https://brasil.un.org/pt-br/sdgs>)

A erradicação da fome e da desnutrição é o segundo e, talvez, um dos mais importantes dentre os 17 Objetivos de Desenvolvimento Sustentável (United Nation, 2017), o que se apresenta como um desafio, considerando o aumento da população urbana mundial, projetado em 12%¹, a degradação de cerca de 20% das terras aráveis e a baixa probabilidade de que a demanda por água seja atendida em 2030 (Trendov et al., 2019, p. 01). Antes da pandemia de COVID-19 o progresso para erradicação da pobreza era lento. O período pré-pandêmico contava com cerca de

¹As previsões indicam que em 2050 cerca de 66% da população mundial viva em regiões urbanas, contrastando com o índice de 54% em 2014.

700 milhões de pessoas passando fome e 2 bilhões em estado de insegurança alimentar. Uma queda de 1,7% em relação a 2015. A pandemia agravou esse cenário, aumentando o número de pessoas famintas para 811 milhões, em 2021. No mesmo ano, cerca de 2.377 bilhões de pessoas estavam em situação de insegurança alimentar. Ao mesmo tempo, a porcentagem de alimentos perdidos após a colheita na fazenda e nas fases de transporte, armazenamento e processamento girou em torno de 13,8%, totalizando mais de 400 bilhões de dólares por ano (FAO, 2021).

A região que engloba a América Latina e o Caribe é onde a insegurança alimentar está aumentando mais rapidamente, subindo de 27,5% em 2015 para 48,8% em 2020, devido a um forte aumento na América do Sul. O Brasil está distante de alcançar a meta (FAO, 2021). Aqui, 70% dos alimentos consumidos são produzidos pela agricultura familiar. Essa produção é realizada em 77% das propriedades agrícolas do país, espalhadas ao longo de 80,9 milhões de hectares, o que representa 23% das fazendas brasileiras (Lima et al., 2019; Ministério da Agricultura, Pecuária e Abastecimento, 2019). De acordo com o último relatório da FAO (FAO, 2021), a produtividade dos pequenos produtores é menor do que a dos grandes produtores de alimentos. Possivelmente isso se deva ao fato de que a agricultura familiar desfruta de escassos recursos e investimentos, diferentemente da agricultura mecanizada, que conta com diferentes recursos financeiros e tecnológicos para realizar sua atividade (Lima et al., 2019).

A produção de alimentos exige mais recursos do que os atualmente disponíveis e o desenvolvimento de sistemas produtivos sustentáveis, de baixo custo financeiro e capazes de aumentar a produtividade ao mesmo tempo em que utilizam menos recursos naturais (Alexandratos e Bruinsma, 2012). Ainda que se acredite que a crescente demanda por alimentos possa ser atendida, não está claro como isso pode ser feito de uma forma sustentável e inclusiva. A única clareza que se tem é a urgente necessidade de transformar o sistema agropecuário, de modo a criar os recursos necessários para atender à nova demanda a médio e longo prazos (Trendov et al., 2019, p. 01).

Enquanto os desafios da agricultura e da produção de alimentos preocupam entidades e governos ao redor do mundo, a Quarta Revolução Industrial, também conhecida como Indústria 4.0, e a Internet das Coisas (IoT, do inglês *Internet of Things*) impulsionam o desenvolvimento de tecnologias e inovações nos mais diversos setores, incluindo a agricultura. Trendov et al. (2019, p. 02) chamam as inovações geradas pela Indústria 4.0 no campo agrícola de *agricultura digital* e afirmam que “esta é a mais disruptiva e transformadora de todas as indústrias, pois ela não só irá modificar como os agricultores cultivam, mas toda a cadeia produtiva”, desde o fornecimento de insumos até a comercialização dos alimentos. A integração dessas novas tecnologias com a IoT pode culminar na próxima revolução agrícola, chamada *Revolução Agrícola Digital*, resultando em sistemas mais produtivos, eficientes, inclusivos, transparentes e resilientes. Para que essa revolução aconteça, é necessário integrar recursos como dispositivos móveis, mídias sociais, agricultura de precisão, tecnologias de sensoriamento remoto, *big data*, computação em nuvem, ciência de dados, tecnologias de coordenação e integração, sistemas inteligentes e segurança cibernética (Trendov et al., 2019, p. 02).

A inclusão de recursos de segurança em áreas emergentes da IoT como cidades, casas e veículos inteligentes, tem sido amplamente pesquisada. Contudo, na agricultura, os recursos incluídos são escassos e não há pesquisas direcionadas especificamente para segurança em soluções voltadas a este setor (Gupta et al., 2020; Zanella et al., 2020; Window, 2019). Isso pode estar relacionado ao fato de que alguns ambientes rurais já são há muito tempo monitorados e que as soluções desenvolvidas para este setor operam de forma satisfatória (Peternela, 2019). Entretanto, as soluções tradicionais foram desenvolvidas para realizar controle local, sem qualquer conexão com redes externas, normalmente utilizando-se de equipamentos interconectados por cabos. O uso de cabeamento e a falta de comunicação com outras redes, cria um ambiente

isolado e pouco susceptível às interferências externas, dispensando uma série de recursos de segurança. No entanto, com o advento da IoT, esses sistemas tendem a ser substituídos por tecnologias conectadas à Internet, possivelmente integrando comunicação sem fio e tomada de decisão na nuvem, o que torna a segurança um requisito indispensável.

Nessa perspectiva, há uma série de questões preocupantes. A integração entre dispositivos e tecnologias, normalmente desenvolvidos por diferentes fornecedores, a preservação da privacidade e da confidencialidade, o gerenciamento e o armazenamento de informações, e a manutenção da confiança e da confiabilidade, podem ser cruciais para o desenvolvimento de soluções eficientes e confiáveis (Hassija et al., 2019). Essas questões são importantes, pois a IoT envolve a integração dos sistemas agrícolas com redes remotas e com a nuvem, incorporando à agricultura digital todos os desafios de segurança dessas redes, que podem incluir a Internet, as redes de sensores sem fio (RSSFs) e de celular, por exemplo. Outrossim, a precisão dos dados e a eficiência do sistema são fundamentais para evitar danos nas lavouras e perdas financeiras, requisitos importantes para toda cadeia agrícola e ainda mais significativos para agricultura familiar. Nesta, diferentemente da agricultura em larga escala, poucas variações na produção podem gerar grande impacto na renda dos agricultores. Uma agricultura provida de sistemas precisos e eficientes consegue não apenas reduzir as variações na produtividade, mas também melhorar a produtividade e aumentar a renda do produtor.

A precisão e a eficiência dos sistemas agrícolas podem ser prejudicadas por vulnerabilidades de segurança que permitam que agentes maliciosos, como criminosos e/ou terroristas virtuais, comprometam sistemas e atuem para danificar fazendas ou plantações, ou ainda, utilizem o sistema como parte de um ataque externo. Soluções com baixos níveis de segurança podem ser utilizadas por criminosos para gerar erros em leituras, mau funcionamento ou falhas, visando afetar o desempenho das aplicações e comprometer a lavoura (Boghossian et al., 2018; Demestichas et al., 2020; Window, 2019). Crimes cibernéticos orquestrados por grupos terroristas podem causar um impacto abrangente e enormes prejuízos a corporações ou a governos e suas economias (Olson, 2012). Por essas razões, a segurança desempenha um papel fundamental na operação eficiente e confiável da agricultura digital.

Além dos problemas de segurança herdados de outras redes e sistemas, a agricultura digital possui requisitos próprios e alguns deles são específicos ao objetivo e contexto em que é utilizada. Por exemplo, aplicações podem monitorar ou controlar ambientes protegidos, como estufas agrícolas, ou áreas abertas, como plantações de milho ou soja. Os ambientes protegidos tendem a ocupar regiões geográficas menores sendo utilizados para cultivar espécies que exigem maior controle climático e ambiental. Em áreas abertas são cultivadas espécies mais rústicas, que podem ocupar desde pequenas até grandes extensões geográficas. Portanto, as soluções para cada ambiente possuem atributos específicos.

Como as características de cada ambiente são distintas, a abordagem da segurança deve ser adequada, para maximizar a confiabilidade. Nas estufas agrícolas, por exemplo, sensores de umidade tendem a sofrer saturação quando expostos a altos índices de umidade, e sensores de velocidade do vento instalados em ambientes empoeirados podem acumular partículas, que prejudicam sua precisão (Paternela, 2019). Já em áreas abertas, os dispositivos podem estar expostos às mudanças climáticas, ações humanas e de animais, ou acidentes com equipamentos agrícolas. As interações entre esses elementos e os dispositivos da agricultura inteligente pode resultar em erros, falhas ou defeitos que prejudicam seu desempenho. Agentes maliciosos podem obter acesso ao sistema, comprometê-lo para manipulá-lo, corrompê-lo ou capturar informações. Muitas dessas questões passam despercebidas para os desenvolvedores, que começam a preocupar-se apenas quando ocorrem incidentes e são encontrados dados incompatíveis ou incoerentes. Em

ambos os casos há um sério comprometimento da confiabilidade do sistema, visto que ele deixa de operar da forma desejada.

Soluções de segurança desenvolvidas para outras áreas da IoT podem ser incorporadas à agricultura digital. O controle de acesso (Ali et al., 2019; He et al., 2018; Liu et al., 2020; Mandal et al., 2020), a autenticação (Ali et al., 2019; Almadhoun et al., 2018; Gope e Sikdar, 2019; Punithavathi et al., 2019), a troca segura de mensagens (Gündoğan et al., 2018; Naik, 2017), a segurança da borda e da nuvem (Chahid et al., 2017; Partra e Rao, 2016) e a proteção a ataques (Varga et al., 2017) são comuns às diferentes áreas e os recursos já desenvolvidos abarcam questões de segurança intrínsecas à agricultura digital. No entanto, questões relacionadas à coleta de dados, aos dados e aos dispositivos que operam na borda podem ser bastante diferentes de outros contextos. Por exemplo, alguns ambientes monitorados, como casas e veículos, não estão expostos a ações de animais, que podem danificar os equipamentos intencionalmente ou não. Os equipamentos que realizam monitoramento em cidades inteligentes são normalmente instalados em áreas de acesso restrito, enquanto os utilizados na agricultura podem estar posicionados em regiões frequentadas por pessoas e implementos agrícolas.

Adicionalmente, soluções buscam resolver questões relacionadas ao acesso malicioso, à intrusão ou às anomalias (Zarpelão et al., 2017; Santos et al., 2018; Nachan et al., 2021). Entretanto, uma questão em aberto está relacionada à qualidade dos dados recebidos de dispositivos como sensores e câmeras. A qualidade (ou acurácia) dos dados é um requisito muito importante, especialmente para sistemas que os utilizam para tomada de decisão, seja ela conduzida de forma automática ou manual². Dados incoerentes ou inconsistentes podem incorrer em falhas de sistemas, como aqueles que controlam temperatura, umidade ou luminosidade, ou resultar em decisões equivocadas, como aplicar ou não insumos para controle de pestes³ ou irrigar áreas desnecessariamente, resultando em desperdício de recursos ou até na perda de plantações inteiras. Da mesma forma que outros requisitos de segurança, a acurácia dos dados pode impactar fortemente na precisão de um sistema e ser decisivo para sua popularização.

Com o intuito de contribuir para a melhoria da confiabilidade de sistemas agrícolas digitais, especialmente aqueles voltados para a agricultura familiar, este trabalho integra a análise e o controle da qualidade dos dados à detecção de anomalias para propor uma solução que maximize a acurácia dos dados coletados e processados pela agricultura digital. A pergunta que norteou o desenvolvimento deste trabalho foi: é possível desenvolver um detector de anomalias que seja eficaz e de baixo custo? O escopo da pesquisa envolve sistemas desenvolvidos para atuarem em ambientes conhecidos, cujo comportamento dos recursos pode ser modelado. Nestes ambientes, espera-se que o comportamento permaneça nos padrões modelados e não ocorram mudanças além dos limiares predefinidos.

Como os dispositivos podem ser instalados em áreas com baixa conectividade com a Internet, como é o caso de muitas propriedades agrícolas, é vantajoso tomar decisões na borda da rede, antes que os dados sejam enviados para a nuvem. Tomar decisões na borda também pode reduzir custos financeiros ligados ao processamento na nuvem⁴ e aumentar a precisão

²Decisões “manuais” são aquelas tomada por um profissional.

³Um sistema para controle de pestes pode não detectar a necessidade de aplicar defensivos em uma área, expondo a plantação às pragas, por exemplo.

⁴O processamento na nuvem citado no texto se refere à *computação em nuvem*, que está associada ao uso de recursos computacionais (servidores, gerenciamento de banco de dados, armazenamento de dados, redes, aplicativos, poder computacional e capacidades especiais, tais como *blockchain* e inteligência artificial) sob demanda através da Internet, com preços pré-pagos e sem gerenciamento ativo direto por parte do usuário.

do sistema. Para isso, parte do processamento deve ser realizado na borda⁵, exigindo que os recursos sejam suficientemente leves para serem executados por dispositivos com capacidade de memória e processamento restritos. Todavia, realizar toda tarefa na borda pode não ser suficientemente eficaz, sendo necessário executar parte na nuvem, aproveitando a robustez dos recursos disponíveis.

Para responder à pergunta de pesquisa foi desenvolvido um detector híbrido de anomalias capaz de identificar falhas em equipamentos e erros de leitura. As indagações que conduziram a pesquisa foram as seguintes:

- É possível desenvolver um *framework* “genérico” para detectar anomalias em sistemas de IoT voltados para a agricultura digital?
- É possível detectar as anomalias com o mínimo de falsos positivos e falsos negativos?
- Qual algoritmo de Aprendizagem de Máquina alcança bom índice de acertos, com baixo consumo de recursos computacionais no cenário de agricultura inteligente com as seguintes limitações:
 - sensores instalados em campo aberto;
 - leitura de parâmetros em tempo real, sem armazenamento dos dados na camada de percepção;
 - processamento e memória limitados na borda.
- Qual o custo de desenvolvimento e implantação deste sistema?
- Para qual(is) tipo(s) de agricultura digital ele é recomendado?
- Quais falhas, erros e ataques relacionados à camada de percepção e da agricultura inteligente podem ser detectados a partir da análise dos dados?

1.1 METODOLOGIA

O desenvolvimento de um detector de anomalias demandou um conjunto de procedimentos metodológicos incluindo pesquisa bibliográfica e exploratória, modelagem de dados e prototipação do sistema. Inicialmente, foi realizada uma pesquisa bibliográfica para identificar o estado da arte dos sistemas agrícolas digitais. Em seguida, conduziu-se uma pesquisa exploratória para modelar os dados gerados por esses sistemas. Para isso, foi criado um ambiente de testes com dispositivos que coletam informações ambientais para gerar os conjuntos de dados.

Após a modelagem dos dados foi realizada uma pesquisa quantitativa para identificar as falhas que ocorrem nos dispositivos, correlacionar variáveis e analisar os conjuntos de dados para identificar atividade anormal. A pesquisa permitiu identificar modelos matemáticos (apresentados na seção 4.4) capazes de reconhecer algumas falhas e erros relacionadas aos dispositivos da camada de percepção e identificáveis a partir dos dados recebidos pela borda. Os modelos foram agregados à análise de dados e Inteligência Artificial para formar um sistema híbrido, que opera na borda da rede e na nuvem.

Por fim, foi desenvolvido um protótipo para verificar o desempenho de detector. O protótipo foi desenvolvido e testado com dados reais, gerados por cinco sensores instalados em

⁵O processamento na borda está relacionado à *Edge Computing*, que pode ser definida como um paradigma de computação distribuída que aproxima as aplicações das fontes de dados, tais como dispositivos IoT ou servidores locais.

ambiente agrícola aberto. A avaliação dos resultados utilizou 19 sensores virtuais, que geraram dados equivalentes aos sensores reais e dados anômalos correspondentes às anomalias alvo deste projeto. Os sensores virtuais foram necessários porque *i)* a quantidade de sensores reais foi insuficiente para testar o protótipo, *ii)* a classificação dos dados gerados por sensores reais é feita manualmente, ocasionando possível imprecisão, *iii)* não foi possível causar todas as falhas e erros em volume suficiente nos sensores reais. A avaliação do desempenho do detector considerou a precisão da classificação dos dados e consumo de recursos computacionais, como tempo de processamento, memória e armazenamento.

1.2 PRINCIPAIS CONTRIBUIÇÕES

Este trabalho apresenta a especificação de um detector de anomalias para sistemas de IoT voltados para a agricultura inteligente. Este detector integra a identificação de anomalias a partir de modelos matemáticos, análise de dados e Inteligência Artificial para localizar indícios de falhas, desgaste e mau funcionamento de equipamentos, ou erros de leitura. Essas são anomalias comuns em equipamentos digitais e que não podem ser prevenidas. Entretanto, é possível identificar sua ocorrência para evitar o uso dos dados e antecipar a correção de falhas ou o conserto dos equipamentos.

A especificação inclui dois módulos, um operando na borda e outro na nuvem. Os módulos podem operar em conjunto, como proposto neste trabalho, ou separadamente. A operação na borda permite a interceptação imediata de dados anômalos e a prevenção do seu uso em processos críticos. A retenção de dados anômalos na borda permite reduzir custos financeiros relacionados ao seu processamento na nuvem e à sua transmissão na rede. Por outro lado, a operação na nuvem aumenta a precisão e o alcance do detector. Além disso, o sistema proposto pode ser ajustado para operar apenas na borda, permitindo sua utilização por sistemas que não integram com a nuvem. Quando houver disponibilidade, o detector pode incorporar algoritmos especializados em ataques cibernéticos e filtrar dados originados na Internet.

O protótipo desenvolvido apresentou bons resultados. A detecção na borda foi capaz de identificar 80% das anomalias analisadas. Já a nuvem superou o índice de 93% de anomalias detectadas, o que demonstra a eficiência do sistema. O consumo de recursos computacionais pode ser considerado baixo, viabilizando a utilização em dispositivos com recursos restritos. Na borda, são consumidos pouco mais de 400KB de memória e 262 KB de espaço em disco⁶. Os dados foram processados em 355ms. Na nuvem, o tempo de processamento fica muito próximo a 4s⁷, com um consumo de memória em torno de 6MB e 2MB de armazenamento em disco.

1.3 ESTRUTURA DO TEXTO

O restante do texto está organizado da seguinte como descrito a seguir. O Capítulo 2 apresenta uma revisão de literatura sobre a agricultura digital, incluindo uma visão geral sobre as tecnologias desenvolvidas até o momento, os recursos de segurança integrados a esses sistemas e as ameaças à segurança desses sistemas. O Capítulo 3 discorre sobre a detecção de anomalias, apresentando um panorama sobre os sistemas de detecção de anomalias e uma visão geral sobre os sistemas de detecção de intrusão.

O Capítulo 4 descreve o detector de anomalias, as falhas que afetam os sistemas agrícolas digitais, o modelo de anomalias por ele tratado e detalha arquitetura do detector. O Capítulo 5

⁶Os valores de armazenamento não incluem bibliotecas e dados armazenados na base de dados.

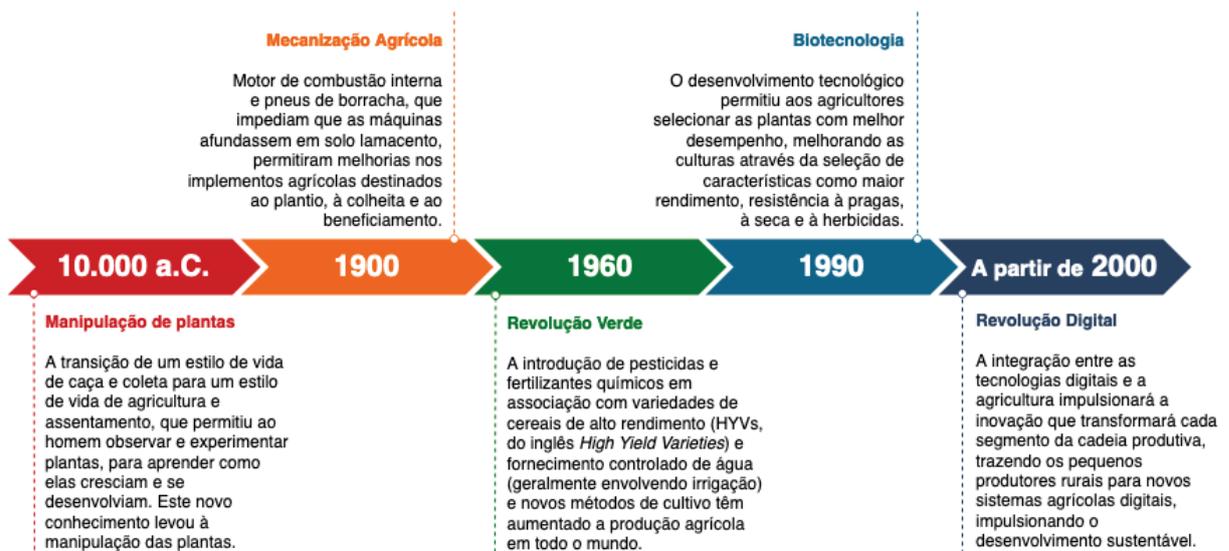
⁷Esses valores não incluem o tempo de processamento do algoritmo de aprendizagem de máquina, que pode variar dependendo do algoritmo escolhido.

aborda o protótipo construído para análise de resultados, incluindo o processo de escolha do algoritmo de aprendizagem de máquina. Este capítulo também mostra os resultados obtidos pelo detector, apresentando informações sobre a precisão e o consumo de recursos computacionais. Por fim, o Capítulo 6 apresenta as conclusões da pesquisa, suas limitações e trabalhos futuros.

2 AGRICULTURA INTELIGENTE: FORNECENDO FERRAMENTAS PARA MODIFICAR OS PROCESSOS PRODUTIVOS

A agricultura é uma das atividades humanas mais antigas e tem sido fundamental para a humanidade, desde a formação das primeiras civilizações (Ehlers, 2017). O seu desenvolvimento passou por diversas etapas e revoluções, como as apresentadas na Figura 2.1, acompanhando a evolução de outros setores como indústria, tecnologia e biotecnologia. Esse processo iniciou há mais de dez mil anos, quando alguns povos do norte da África e oeste da Ásia começaram a cultivar plantas e domesticar animais em substituição à caça e à coleta de alimentos (Ehlers, 2017; Trendov et al., 2019). Inicialmente esse processo era feito nas proximidades dos rios, locais onde a fertilidade do solo permitia a produção de grãos. Contudo, devido às técnicas precárias de produção e à crescente demanda por alimentos, a fome dizimou milhares de pessoas e tornou a produção de alimentos um dos maiores desafios ao longo de toda Antiguidade, Idade Média e Renascença (Ehlers, 2017).

Figura 2.1: Revoluções Agrícolas



FONTE: O Autor (2020)

Esse cenário modificou-se no início do século XVIII com o desenvolvimento de novos sistemas agrários e com a mecanização agrícola (Mazoyer e Roudart, 2006). Na primeira metade do século XX, as técnicas de cultivo se modificaram profundamente com o desenvolvimento de equipamentos e implementos agrícolas. Isso possibilitou produzir alimentos em maior escala, o que foi fundamental para atender à crescente demanda, suprimindo as necessidades de uma população cada vez mais urbana (Ehlers, 2017; Mazoyer e Roudart, 2006). A partir de 1960, o desenvolvimento da indústria química e da biologia permitiu a criação de novas variedades de algumas culturas, como arroz, milho, trigo e soja, a utilização de fertilizantes e pesticidas sintéticos, e a introdução de sistemas de irrigação e drenagem. Essas melhorias resultaram na chamada Revolução Verde, que permitiu aumentar substancialmente a produtividade agrícola (Mazoyer e Roudart, 2006). Já no final do século XX e início do século XXI, a biotecnologia permitiu a criação de plantas com características selecionadas, como maior produtividade, resistência às

pragas, às secas e aos herbicidas (Mazoyer e Roudart, 2006), características que permitiram o desenvolvimento da sociedade e a sobrevivência até os dias atuais.

No Brasil, as tecnologias agrícolas são acessíveis apenas para uma pequena parcela da população rural. Aqui, coexistem uma agricultura altamente mecanizada e tecnologicamente avançada e uma agricultura chamada “*familiar*”, que dispõe de poucos recursos financeiros e tecnológicos (Lima et al., 2019). Apesar de contar com poucos recursos financeiros e um baixo índice de assistência técnica, a agricultura familiar é responsável por uma produção superior a 70% dos alimentos consumidos no país (Lima et al., 2019; de Souza et al., 2019). Conforme o último censo agrícola, realizado em 2017, a agricultura familiar ocupa uma extensão de 80,9 milhões de hectares, o que equivale a 23% das fazendas brasileiras. Embora ocupe uma “pequena” faixa do território nacional, essas propriedades representam 77% das propriedades agrícolas do país (Ministério da Agricultura, Pecuária e Abastecimento, 2019).

Diante desse cenário e para que a sociedade continue a progredir e possa sobreviver nas próximas décadas, é preciso dar mais um passo, desenvolvendo técnicas mais eficientes, sustentáveis e de baixo custo financeiro, capazes de aumentar a produtividade enquanto preservam recursos naturais e econômicos. Essa é a missão da Revolução Agrícola Digital, capaz de unir as novas tecnologias fornecidas pela Indústria 4.0 e pela Internet das Coisas para dar suporte aos novos métodos de produção agrícola (Trendov et al., 2019).

2.1 A INTEGRAÇÃO DE TECNOLOGIAS PARA UMA AGRICULTURA SUSTENTÁVEL

Em um momento em que as mudanças climáticas estão em evidência e que as projeções indicam um aumento populacional substancial para as próximas décadas, torna-se imprescindível aperfeiçoar as formas de produção. Os novos métodos produtivos devem gerar menor impacto ambiental, consumindo menos recursos naturais, enquanto produzem mais em espaços geográficos menores. De acordo com Trendov et al. (2019), esse avanço pode ser obtido pela integração dos métodos de cultivo às novas tecnologias computacionais altamente interconectadas e com intenso uso de dados. Os novos sistemas agrícolas, aqui chamados também de Agricultura 4.0, agricultura inteligente ou agricultura digital, podem ser compostos por um conjunto de tecnologias, como dispositivos móveis, serviços de sensoriamento remoto, agricultura de precisão, *big data*, computação em nuvem e computação distribuída. Além disso, para serem considerados inteligentes, esses sistemas devem integrar recursos de inteligência artificial e aprendizagem de máquina.

O desenvolvimento da Agricultura 4.0 ainda está em suas fases iniciais e a Internet das Coisas pode contribuir significativamente, fornecendo equipamentos, tecnologias e protocolos capazes de coletar dados, interagir com o ambiente e interconectar recursos computacionais aos meios de produção. Embora os benefícios resultantes do uso de sistemas digitais na agricultura sejam evidentes, há muitos desafios a serem enfrentados para torná-la uma realidade (Trendov et al., 2019). No que compete à segurança, os principais desafios envolvem as definições sobre a propriedade dos dados, padronizações que permitam a comunicação entre dispositivos e soluções fornecidas por diferentes fabricantes, recursos para proteção dos dados nas fases de coleta, transporte, processamento e armazenamento.

Os primeiros passos para o desenvolvimento da agricultura inteligente se concentraram no aprimoramento de técnicas de automação com a inclusão de poucos recursos computacionais, como é o caso dos trabalhos desenvolvidos por Khelifa et al. (2015) e Sales et al. (2015). Esses sistemas foram equipados com sensores para coletar informações sobre a umidade do solo e enviá-las para um *gateway*, conectado a um computador através de uma rede local. O computador possui uma base de dados para armazenar os registros recebidos dos sensores e uma página *web*

simples para apresentar as informações ao usuário. Estes sistemas não se conectam a outras redes externas, como a Internet, e não atuam sobre o ambiente, limitando-se a coletar informações e mostrá-las ao usuário.

Recentemente o cenário tem sofrido mudanças impulsionadas principalmente pelas pesquisas em inteligência artificial e aprendizagem de máquina. Neste sentido, os maiores esforços têm se concentrado no desenvolvimento de soluções para irrigação (Goap et al., 2018; Mahalakshmi, 2018; Nageswara Rao e Sridhar, 2018; Navarro-Hellín et al., 2016; Rajalakshmi e Devi Mahalakshmi, 2016; Zhao et al., 2017). Tais soluções têm unido recursos para tomada de decisão a sistemas de monitoramento e controle, buscado reduzir o consumo de água e aumentar a produtividade. Também são encontradas soluções voltadas para hidroponia (Ruengittinun et al., 2017), horticultura (Lee et al., 2017), vinhedos (Oliver et al., 2018) e detecção de doenças foliares (Thorat et al., 2018). Alguns sistemas de propósito mais geral (Colezea et al., 2018; Minh et al., 2017; Musat et al., 2018; Raducu et al., 2015; Sales et al., 2015; Wongpatikaseree et al., 2018; Yoon et al., 2018) apenas implementam tecnologias e recursos de IoT, serviços *web*, serviços de alerta, recursos de rastreamento ou controles na nuvem. As soluções desenvolvidas para agricultura digital ainda são imaturas e possuem pouca inteligência.

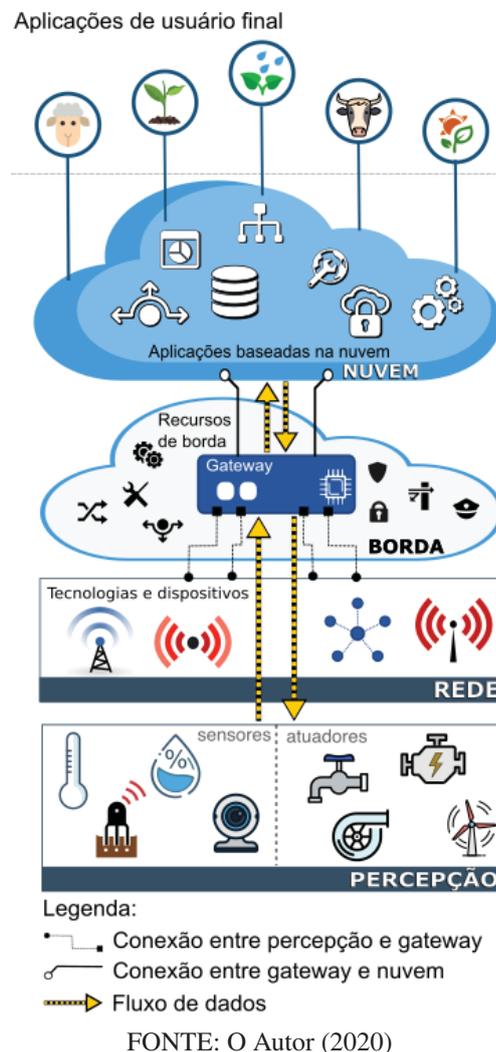
Algumas dessas propostas se restringem à automação, integrando sensores e atuadores a um *gateway*. Trabalhos como o de Navarro-Hellín et al. (2016), por exemplo, não apresentam informações sobre a utilização de recursos na nuvem ou conexão com a Internet, apesar de utilizar inteligência artificial para tomada de decisões. Outras propostas empregam sistemas integrados a Internet, para o envio e armazenamento de informações em uma base de dados (Rajalakshmi e Devi Mahalakshmi, 2016; Khelifa et al., 2015; Minh et al., 2017; Ruengittinun et al., 2017; Zhao et al., 2017; Wongpatikaseree et al., 2018; Nageswara Rao e Sridhar, 2018; Yoon et al., 2018). Entretanto, esses trabalhos usam apenas as capacidades de armazenamento da nuvem e a interação com o usuário, não avançando para o uso de outros recursos disponíveis na nuvem. Alguns poucos trabalhos utilizam *big data* ou técnicas de inteligência artificial para tomadas de decisão, fazendo, assim, um uso mais efetivo das funcionalidades da nuvem (Goap et al., 2018; Lee et al., 2017; Oliver et al., 2018; Raducu et al., 2015; Thorat et al., 2018).

Em geral, os sistemas mencionados foram desenvolvidos baseados na arquitetura mostrada na Figura 2.2, que consiste em dispositivos da camada de percepção, funcionalidades da camada de rede, recursos de borda, aplicações e serviços baseados na nuvem (Mekala e Viswanathan, 2017; Ray, 2017). A camada de percepção inclui sensores, etiquetas de identificação por radiofrequência (RFID, do inglês *Radio-Frequency IDentification*), câmeras, atuadores e outros dispositivos responsáveis por coletar dados no ambiente agrícola e atuar sobre o ambiente. Esses dispositivos não têm capacidade computacional para processar ou armazenar dados, o que pode ser feito na borda da rede ou na nuvem, por sistemas de banco de dados e soluções específicas para processamento e manipulação das informações. Para isso, essa camada precisa se comunicar com a borda para trocar informações sobre o estado atual do ambiente e as intervenções a serem efetivadas. A camada de percepção se conecta aos recursos de borda através de tecnologias de rede, como rede de sensores sem fio e Zigbee.

A borda pode conter diferentes recursos, como mecanismos de segurança, filtros de dados, ferramentas para tomada de decisões, *interfaces* de entrada e saída, e o *gateway*. Este, pode ser um dispositivo limitado⁸, como um Arduino, um ESP32 ou um Raspberry Pi. Alguns desses dispositivos podem apenas encaminhar os dados recebidos da percepção para a nuvem, enquanto outros também podem processá-los, tomar decisões e enviar comandos para atuadores. Na nuvem, os dados podem ser armazenados e processados para fornecer informações e serviços

⁸Dispositivos limitados são dispositivos pequenos com CPU, memória e recursos energéticos limitados (Bormann et al., 2014, RFC 7228)

Figura 2.2: Estrutura dos Componentes de Agricultura Inteligente



aos usuários finais. O processamento é um desafio, considerando a grande quantidade de dados produzida pelos subsistemas, que podem alcançar o mundo do *big data*. Processar tudo na nuvem, como proposto por muitas soluções, implica na necessidade de excessiva largura de banda, pode reduzir a vida útil da bateria dos dispositivos e pode custar muito caro.

Consequentemente, a computação na borda, ou *edge computing*, é um importante instrumento capaz de poupar recursos, melhorar a eficiência e a produtividade. Os dados consumidos ou pré-processados na borda economizam largura de banda, recursos computacionais e energéticos, protegem a privacidade e poupam custos de computação em nuvem (Zhang et al., 2016). A nuvem é responsável por armazenar e processar dados, tomar decisões e reportar informações importantes ao fazendeiro. O *gateway* e a nuvem são interconectados por um Provedor de Serviços de Internet (ISP, do inglês *Internet Service Provider*). Considerando que o processo de tomada de decisão deve basear-se em muitos dados (*big data*), ferramentas de inteligência artificial são um requisito importante.

A partir do aprimoramento dos dispositivos de coleta de dados e das tecnologias de comunicação, a tendência é integrar mais recursos computacionais aos sistemas. Tal integração visa atender às diferentes demandas da automação, de controle agrícola e da agricultura de

precisão (Ray, 2017; Sales et al., 2015). As soluções devem evoluir para sistemas de gerenciamento em vez de concentrarem-se apenas no monitoramento. Outra preocupação é a segurança dos dados. É fundamental assegurar a privacidade, a confiabilidade e a precisão dos dados, desde sua coleta, até o seu armazenamento e a tomada de decisão na nuvem.

2.2 AMEAÇAS À SEGURANÇA DA AGRICULTURA INTELIGENTE

Como discutido anteriormente, a agricultura digital pode ser dividida em quatro camadas: (i) percepção, (ii) rede, (iii) borda e (iv) aplicação. A tabela 2.1 analisa os recursos responsáveis pela coleta, transporte, processamento e armazenamento de dados em cada camada. Um conjunto de dispositivos, protocolos e tecnologias utiliza os dados para monitorar os ambientes e automatizar as atividades agrícolas (Mekala e Viswanathan, 2017). O armazenamento, o gerenciamento e o processamento de dados combinados à conectividade com a Internet trazem vários problemas e ameaças à segurança. A figura 2.3 resume os ataques à agricultura inteligente na perspectiva em camadas.

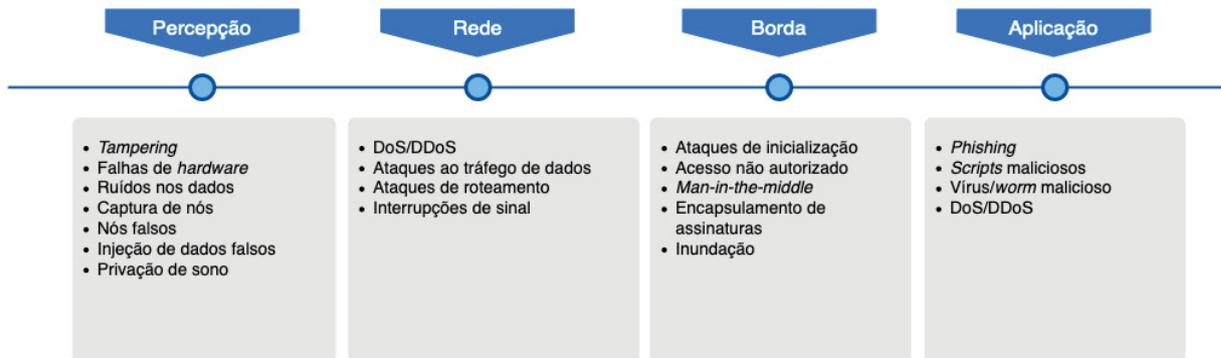
Tabela 2.1: Elementos de Sistemas Agrícolas inteligentes

Camada	Recurso	Descrição
Percepção	Sensor e Câmera	Pequenos dispositivos para coleta de dados ambientais, tais como umidade e temperatura.
	Atuador	Dispositivos ou sistemas usados para alterar o estado do ambiente. Exemplo: aspersor, dispositivos de ventilação e sistema de irrigação.
	Etiqueta RFID	Pequenos dispositivos capazes de armazenar dados, tais como número de identificação de um animal.
	GPS	Um sistema que fornece a geolocalização de um equipamentos e implementos agrícolas, por exemplo, e pode auxiliar a agricultura de precisão.
Rede	Tecnologias de Conexão	Equipamentos e tecnologias responsáveis por interconectar dispositivos remotos e transferir dados. Exemplo: roteador, pontos de acesso, protocolos.
Borda	Elementos de segurança	Protocolos e esquemas de segurança responsáveis por assegurar a disponibilidade, integridade e confidencialidade do sistema e dos dados.
	Interfaces de entrada e saída	Dispositivos <i>software</i> e <i>hardware</i> capazes de se comunicar com recursos localizados além da área local.
	Recursos diversos	Elementos de <i>software</i> empregados em tarefas como tomada de decisão e processamento de dados.
Aplicação	Gateway	Sistema situado na borda da rede, conectado a dispositivos agrícolas (camada de percepção) e à nuvem. Este sistema pode processar dados, armazenar pequenas quantidades de informações e comunicar com a nuvem.
	Sistemas de Banco de Dados	Sistema para armazenar os dados produzidos pelo sistema inteligente.
	Ferramenta <i>web</i>	Recursos para trocar dados entre aplicativos remotos e fornecer acesso a aplicativos de usuário final na Internet.
	Sistemas de Tomada de Decisão	Sistemas para tomada de decisões que alteram o estado do ambiente.
	Aplicativos de usuário final	<i>Software</i> para apresentar informações ao usuário e permitir que ele interaja com o ambiente.

FONTE: O Autor (2020)

Os incidentes de segurança podem ser acidentais ou intencionais. Animais, trabalhadores agrícolas e/ou máquinas agrícolas podem facilmente acessar ambientes de cultivo e causar incidentes. Além disso, os sistemas inteligentes compreendem dispositivos e *softwares*

Figura 2.3: Riscos e ameaças à agricultura inteligente



FONTE: O Autor (2020)

heterogêneos, de fabricantes distintos, instalados entre as lavouras e a nuvem. Estas características específicas podem criar várias brechas de segurança e resultar em incidentes que comprometem o sistema inteligente. No entanto, este tópico não foi estudado na maioria dos sistemas em uso até agora.

O projeto do sistema deve considerar a compatibilidade com dispositivos, protocolos, subsistemas distintos e métodos de multi-acesso. A agricultura inteligente utiliza comunicação máquina-a-máquina (M2M, do inglês *machine-to-machine*) e dispositivos fabricados por diferentes fornecedores. Entretanto, a maioria dos mecanismos de segurança foram desenvolvidos para o modelo de comunicação utilizado pelas redes TCP/IP. Esses mecanismos ignoram a existência de múltiplos dispositivos heterogêneos comunicando-se simultaneamente. Os recursos de segurança criados para redes TCP/IP podem dividir a conexão entre os dispositivos agrícolas inteligentes, reduzindo sua eficiência. Métodos de múltiplos acessos e heterogeneidade dificultam a segurança, a interoperabilidade e a coordenação da rede, aumentando as vulnerabilidades de segurança (Zhao e Ge, 2013).

A Agricultura 4.0 está exposta a um grande espectro de ciberataques e a preocupação com a segurança precisa fazer parte do sistema. Entre as questões de segurança a serem consideradas estão o controle de acesso, gerenciamento, armazenamento de informações, integridade de dados e confiabilidade. A maioria dos problemas que afetam a Agricultura 4.0 são bastante frequentes em outros sistemas de IoT, mas alguns, como oxidação, saturação, choques e danificação dos sensores devido à choques⁹, estão presentes apenas em contextos específicos, incluindo aqueles ligado à agricultura digital. Este trabalho abordará os requisitos de segurança independentemente do contexto. Portanto, é papel do desenvolvedor selecionar os requisitos relevantes para cada sistema. Aqui são introduzidas algumas das questões de segurança existentes na Agricultura 4.0, descrevendo as ameaças mais relevantes em cada camada separadamente.

2.2.1 Ameaças à segurança na camada de percepção

A camada de percepção trata principalmente de dispositivos físicos, tais como sensores e atuadores. Eles podem ser instalados em pequenas áreas agrícolas, como as encontradas na Europa e os sítios de agricultura familiar brasileiros, ou espalhados em grandes fazendas, comuns nos EUA, Austrália e regiões do Brasil. Os dispositivos dessa camada podem se

⁹Neste trabalho, choque está relacionado ao encontro violento, com impacto ou abalo brusco, entre dois corpos; colisão, concussão.

conectar a um *gateway* por cabos ou redes sem fio, sendo esta usualmente não estruturada. Essa comunicação permite que as camadas troquem dados e comandos para interagir com o ambiente. Os dispositivos físicos podem funcionar mal devido à ação humana acidental ou intencional, vírus, *malware* ou cibercriminosos. Há muitos modelos de sensores e tecnologias usadas por aplicações agrícolas inteligentes, e esta variedade permite várias ameaças à segurança, como as apresentadas a seguir:

Tampering - é uma modificação física deliberada ou não intencional de um dispositivo, ou enlace de comunicação (Varga et al., 2017). Um oponente pode modificar dispositivos ou redes de comunicação para desviá-los do seu funcionamento normal; Um funcionário insatisfeito ou concorrente comercial pode manipular recursos para gerar perdas ou obter vantagens econômicas; Pessoas ou animais podem colidir involuntariamente, mover ou remover os dispositivos da sua localização original, violando a integridade do sistema, a precisão e a disponibilidade dos dados; Modificações em um sensor ou atuador podem causar uma operação imprecisa do sistema. Alterações no enlace de comunicação podem causar corrupção de dados, resultando na incapacidade de transferí-los (negação de serviço). Mesmo que não seja viável evitar adulterações, é importante detectá-las para adotar medidas de recuperação e evitar o funcionamento não confiável do sistema.

Falhas de *hardware* - os dispositivos de percepção são propensos à falha, o que significa que eles podem parar de fazer o seu trabalho ou não fazê-lo corretamente (Di Modica et al., 2019). A degradação natural dos dispositivos ou o mau funcionamento pode danificá-los temporária ou permanentemente, causando falhas. A maioria dos sensores utilizados em áreas rurais são equipamentos relativamente simples, e estão expostos a eventos adversos (como sol, chuva, raios, neve e neblina), susceptíveis a danos físicos graduais (devido a variações frequentes ou repentinas), e são propensos a ações físico-químicas que podem danificá-los. Além disso, animais que transitam por áreas rurais, trabalhadores agrícolas e equipamentos agrícolas podem colidir ou danificar acidentalmente o dispositivo. Embora as falhas sejam comuns e relativamente previsíveis, elas são críticas porque podem fazer com que os sistemas inteligentes tomem decisões erradas ou funcionem incorretamente. Por exemplo, um sistema de irrigação inteligente pode começar a irrigar em uma área inundada depois de receber dados incorretos de sensores oxidados, ou nunca irrigar em virtude de sensores saturados. É comum que alguns sensores sejam trocados periodicamente, mas a ação de agentes externos¹⁰ e eventos meteorológicos (raios e granizos) podem acelerar o processo de degradação, causando falhas inesperadas ou danificando-os antecipadamente. Nessas condições, os dispositivos podem parar de funcionar ou enviar dados incorretos. Mesmo que não seja possível evitar essas falhas, é importante identificá-las para corrigi-las e evitar que dados imprecisos sejam usados para a tomada de decisões.

Ruídos nos dados - trata-se de uma alteração parcial ou completa dos dados, resultante de um fator externo (García-Gil et al., 2019). A agricultura digital pode ser afetada por ruídos causados por interferência eletromagnética gerada por motores, máquinas agrícolas ou por mau funcionamento de dispositivos. Em algumas regiões, especialmente em grandes fazendas, as redes de alta tensão podem passar sobre áreas agrícolas e gerando um campo eletromagnético que pode causar distorções ou corrupção de dados. Tal ruído pode resultar em informações incompletas ou mesmo falsas. Isso pode ser perigoso para

¹⁰Animais, trabalhadores agrícolas e máquinas agrícolas que podem colidir acidentalmente ou interagir com o dispositivo.

a tomada de decisões, resultar em análises de dados incorretas e reduzir a acuracidade das informações (Kumar et al., 2016; Raheem e Uwanthika, 2019).

Captura de nós - consiste na captura física ou lógica de um nó ou dispositivo. Esta operação pode ser realizada substituindo totalmente o dispositivo ou modificando os componentes de *hardware* ou *software* (Agrawal et al., 2019; Lin et al., 2017). A captura do nó pode não gerar um impacto significativo se realizada em um único nó e não desencadear outros ataques. Entretanto, após a captura de um dispositivo, o adversário pode modificar o *hardware* ou *software*, obter acesso ao sistema ou injetar dados falsos. A captura de um nó quebra a integridade do sistema, potencialmente interfere na tomada de decisões, pode danificar o cultivo e causar perdas financeiras. Por exemplo, um atuador hostil em um sistema de irrigação poderia nunca iniciar a irrigação ou inundar a cultura. Um funcionário insatisfeito ou um concorrente comercial que tenha acesso físico ou lógico ao sistema poderia realizar este ataque por várias razões.

Nós falsos - um adversário adiciona nós falsos ou maliciosos ao sistema para interromper o seu funcionamento (Zhao e Ge, 2013). A captura de um nó pode desencadear este ataque e levar à replicação do nó. Este tipo de ataque geralmente tende tanto a manipular dados quanto a desativar serviços e dispositivos. Em um sistema com controle de identidade insuficiente ou fraudulento, sensores maliciosos podem enviar dados errados interferindo na tomada de decisão, ou injetar vários pacotes na rede causando uma negação de serviço, ou privação de sono. Da mesma forma, atuadores podem agir maliciosamente, *gateways* hostis podem enviar comandos falsos para atuadores legítimos ou agir como buracos negros para causar danos.

Injeção de dados falsos - por meio de um hospedeiro malicioso, um adversário injeta dados falsos para causar erros, mau funcionamento ou interrupção de serviço (Hassija et al., 2019; Lin et al., 2017). Sensores maliciosos podem criar dados falsos ou alterar as informações coletadas para reduzir a precisão do sistema. Os dados falsos permitem a manipulação do sistema, comprometem a análise dos dados e produzem o mau comportamento (Mode et al., 2020; Bostami et al., 2019). Por exemplo, se a irrigação inteligente receber informações imprecisas sobre a umidade do solo, é provável que ela tome decisões erradas. Este ataque também pode ser realizado em outras camadas, como a borda ou a rede, explorando vulnerabilidades de protocolo, ou na camada de aplicação, abusando das fraquezas de segurança dos aplicativos. Nesses casos, um oponente manipulando os atuadores poderiam fazê-los funcionar de forma inadequada, ou modificar os dados trocados entre a borda e a nuvem para enganar o sistema.

Privação de sono - este ataque visa drenar a bateria do dispositivo até o seu esgotamento. Os sensores agrícolas inteligentes possuem energia limitada e normalmente usam baterias. Para reduzir o consumo de energia e prolongar a vida útil da bateria, os nós devem hibernar quando não estiverem ativos. O ataque de privação de sono envia conjuntos de pedidos aparentemente legítimos para que os dispositivos permaneçam acordados o máximo de tempo possível. Portanto, a bateria do dispositivo se esgotará e o nó se desligará. Se os sensores desligarem, os dados não serão mais coletados nem enviados, comprometendo a eficiência do sistema.

Como a Agricultura 4.0 é um sistema de campo aberto, eles são suscetíveis às condições ambientais, às flutuações climáticas e à ação humana. Medidas de segurança fracas podem afetar a confiabilidade e confiança do sistema e expô-los ao uso acidental de dados corrompidos,

controle remoto e danos físicos. Os sensores não possuem recursos computacionais que permitam a adoção de métodos tradicionais de segurança, como a criptografia, o que torna a segurança ainda mais desafiadora. Portanto, adicionar soluções de segurança inovadoras a esta camada é tão desafiador quanto necessário.

2.2.2 Problemas de segurança na camada de rede

A camada de rede transmite dados da camada de percepção para uma unidade computacional mais robusta, geralmente a nuvem. Esse encaminhamento é normalmente feito por redes que usam tecnologias Baixa Potência e Área Ampla (LPWA, do inglês *Low Power Wide Area*)¹¹, como as apresentadas no Apêndice A. A transmissão de grandes volumes de dados através de uma ampla área de transmissão torna a rede suscetível a ataques, que geralmente ameaçam a confidencialidade e a integridade (Kumar et al., 2016). Embora a rede de comunicação existente tenha medidas de proteção de segurança relativamente completas, existem algumas ameaças comuns que podem comprometer seus recursos (Chahid et al., 2017; Zhao e Ge, 2013). As principais questões de segurança nesta camada são as seguintes:

DoS/DDoS - é um ataque transversal que afeta todas as camadas. A Negação de Serviço (DoS, do inglês *Denial of Service*) visa impedir o acesso a serviços ou dispositivos, seja por sobrecarga da rede ou pela exploração de vulnerabilidades de protocolo que levam ao colapso de recursos, tais como CPU e memória (Vasques e Gondim, 2019). Este ataque pode ser realizado inundando servidores ou roteadores com inúmeros pedidos. Os ataques de inundação podem causar atrasos na rede, desativar dispositivos e tornar o serviço indisponível. Quando o atacante usa múltiplas fontes para inundar o alvo, então tal ataque é chamado ataque Distribuído de Negação de Serviço (DDoS, do inglês *Distributed Denial of Service*). Embora tais ataques não tenham sido projetados especificamente para sistemas inteligentes, a conectividade com a Internet, a pervasividade, a heterogeneidade e a alta vulnerabilidade desses sistemas os tornam propensos a tais ataques (Kolias et al., 2017). Em sistemas agrícolas, os ataques DoS podem impedir que os dados cheguem à borda ou à nuvem a tempo, atrasar os comandos para os atuadores e tornar os serviços indisponíveis.

Ataques ao tráfego de dados - alguns ataques pretendem interceptar os dados trocados entre os componentes da rede para encontrar informações sensíveis (Hassija et al., 2019). Diferentes tecnologias de conexão conectando pontos distintos na rede e redes sem fio transportando dados claros (não criptografados) tornam estes sistemas suscetíveis a violação de dados (Capellupo et al., 2017; Hassija et al., 2019). Um oponente poderia conduzir interceptação de tráfego através de pontos de acesso maliciosos ou por meio de ataques *man-in-the-middle* (Capellupo et al., 2017). Isto permite capturar informações sensíveis, como identificadores de dispositivos, credenciais de acesso ou chaves criptográficas, ou corromper o tráfego da rede, levando a um controle malicioso ou mesmo a um comprometimento total do sistema.

Ataques de roteamento - Ataques de roteamento alteram as rotas da rede para obter o controle do tráfego. As redes de IoT podem ter nós maliciosos que tentam redirecionar os caminhos de roteamento durante o processo de transmissão de dados. Ataques como *sinkhole* e *wormhole* podem subverter a rede e obter acesso não autorizado. O *sinkhole* é

¹¹Muitos trabalhos, assim como fornecedores brasileiros de soluções, usam tecnologias LPWA como LoRa, LoRaWan e Sigfox para comunicação com a Internet.

um ataque de roteamento em que um rival anuncia um caminho de roteamento mais curto e atrai nós para rotar o tráfego por meio dele. As rotas maliciosas permitem interromper o fluxo de tráfego (Hassija et al., 2019; Pundir et al., 2020). Em um *wormhole*, um oponente cria um túnel entre dois nós para transferência rápida de pacotes de modo a criar um atalho na rede e controlar o tráfego (Goyal e Dutta, 2018; Hassija et al., 2019). Durante estes ataques, o destinatário pode receber as informações atrasadas, receber informações parciais ou alteradas, ou até mesmo não recebê-las (Goyal e Dutta, 2018; Pundir et al., 2020).

Interrupções de sinal - Os campos eletromagnéticos inseridos na rede, deliberada ou inadvertidamente, podem causar perturbações que podem corromper os dados ou impossibilitar a comunicação. Os campos eletromagnéticos podem vir de motores e redes de energia operando perto de dispositivos agrícolas inteligentes, ou de ataques de interferência. Enquanto os ataques de interferência podem obstruir os canais de comunicação e drenar a energia dos dispositivos de percepção (Guo et al., 2017; Fadele et al., 2019), outros campos eletromagnéticos podem corromper pacotes aleatórios e reduzir a eficiência do sistema. Sinais corrompidos podem causar má representação de dados ou mesmo limitar a operação de protocolos de rede, impossibilitando a comunicação.

Os recursos da camada de rede na agricultura inteligente e na IoT têm algumas vulnerabilidades de segurança comuns. Entretanto, a Agricultura 4.0 pode consistir em múltiplos subsistemas e integrar tecnologias de diferentes fornecedores. Portanto, a integração de recursos requer cautela para evitar incompatibilidades. Da mesma forma, os mecanismos de segurança nativos e as tecnologias podem não ser totalmente confiáveis, pois podem conter vulnerabilidades embutidas no sistema ou geradas pelo processo de integração.

2.2.3 Problemas de segurança na borda da rede

A borda contém elementos críticos que monitoram e controlam subsistemas, comunicando-se com todas as camadas e acessando recursos estratégicos. O processamento dos dados gerados pela percepção pode ser local, reduzindo o consumo de energia e largura de banda proveniente da transferência dos dados para a infraestrutura centralizada. Devido à arquitetura distribuída da computação de borda, esta camada pode fornecer serviços com resposta mais rápida e maior qualidade, em contraste com a computação em nuvem (Lin et al., 2017). A conexão direta aos recursos da nuvem e da percepção faz da borda um ponto estratégico, tornando a segurança um requisito fundamental para garantir a confiabilidade do sistema. As principais questões de segurança da borda são as seguintes.

Ataques de inicialização - São ataques que exploram o processo de inicialização visando comprometer o sistema (Zhao et al., 2020). Por exemplo, cartões SD utilizados em dispositivos de borda podem conter *scripts* maliciosos que poderiam ser executados na inicialização (Schulz et al., 2017). Esses dispositivos normalmente têm poucos recursos de segurança e raramente incluem proteção de inicialização, deixando os dispositivos vulneráveis a ataques (Garcia et al., 2020). Processos de inicialização maliciosos podem desencadear uma série de ataques à borda com proteção insuficiente. Esses processos poderiam abrir *back doors*, ou permitir a elevação de privilégios. Recursos computacionais insuficientes e conexão direta com a percepção e a nuvem tornam imperativa a proteção do processo de inicialização.

Acesso não autorizado - Um adversário pode obter acesso indevido ao dispositivo de borda que não possua controles de acesso adequados. A borda é um elemento crítico, pois todos os dados passam por ela. Entretanto, vários sistemas agrícolas utilizam controles de acesso fracos ou insuficientes¹². Isso permite que um oponente acesse o dispositivo através da conexão com a nuvem. Os desenvolvedores de projetos para a agricultura inteligente não discutem o uso de recursos de controle de acesso, e as soluções comerciais existentes raramente mudam as credenciais após a implantação. Devido ao grande número de “coisas” que se comunicam com dispositivos ou serviços de borda, os métodos de autenticação precisam ser escaláveis, facilmente gerenciáveis e exigirem a menor intervenção humana possível (Lin et al., 2017; Ouaddah et al., 2017).

Man-in-the-middle - neste ataque, um oponente intercepta uma comunicação para coletar informações ou mesmo modificá-las. Sistemas com segurança limitada ou insuficiente são vulneráveis a ataques internos ou externos, que podem ser acionados por dispositivos comprometidos ou por nós maliciosos (Navas et al., 2018; Stojmenovic e Wen, 2014). Muitas soluções utilizam protocolos de comunicação que utilizam o modelo *publish-subscribe* com um corretor, que atua efetivamente como um *proxy*. Estes protocolos permitem dissociar os clientes e assinantes uns dos outros, permitindo o envio de mensagens sem o conhecimento do destinatário. Um atacante que obtém o controle do corretor e se torna um *man-in-the-middle* pode controlar a comunicação sem que nenhum dos clientes perceba (Hassija et al., 2019).

Encapsulamento de assinaturas - estes ataques modificam a mensagem original, injetando um elemento falso para realizar um pedido arbitrário de serviço *web*, enquanto se autentica como um usuário legítimo (Gajek et al., 2009). Os serviços *web* para comunicação da borda com a nuvem geralmente usam assinaturas XML. Um adversário que corrompe o algoritmo de assinatura, pode realizar operações ou alterar a mensagem capturada explorando vulnerabilidades de protocolo (Hassija et al., 2019). Um adversário pode controlar os atuadores ou manipular os sistemas de tomada de decisão através de mensagens maliciosas.

Inundação - este é um ataque DoS em que muitos pacotes são enviados para um sistema ou rede para sobrecarregá-lo. Em sistemas agrícolas, dispositivos infectados podem iniciar um ataque de inundação direcionado aos dispositivos de borda para comprometer a Qualidade do Serviço (QoS, do inglês *Quality of Service*) ou mesmo para interrompê-los. Um dispositivo hostil na camada de percepção ou um portal malicioso na nuvem poderia enviar várias requisições a um serviço até sua exaustão. Estes ataques poderiam ter um impacto severo sobre os sistemas, sobrecarregando a borda e resultando em uma negação de serviço. Este tipo de ataque também poderia ser realizado na camada de rede e na nuvem (Hassija et al., 2019; Varga et al., 2017).

Os elementos da borda fornecem serviços de computação para clientes ou aplicativos e podem se conectar com diferentes recursos de todas as camadas. Tanto os dados processados localmente como os enviados para recursos externos passam por esta camada. Proteger os dispositivos contra acesso remoto e usar recursos criptográficos apropriados são desafios-chave

¹²Controles de acesso realizados exclusivamente por autenticação simples (identificadores e senhas) são considerados fracos, pois é relativamente fácil descobrir a senha. Existem vários ataques voltados a esse tipo de autenticação. A autenticação por IP também é considerada insegura, pois um oponente pode forjar o endereço IP para obter acesso ao sistema.

de segurança. Portanto, é imperativo obter recursos de segurança para proteger os dados e os recursos de borda.

2.2.4 Problemas de segurança na camada de aplicação

A camada de aplicação visa fornecer serviços aos usuários finais, armazenar dados e tomar decisões sobre o sistema. As questões de segurança nesta camada concentram-se em prevenir o roubo de dados e na privacidade, e são específicas para diferentes aplicações. Algumas aplicações consistem numa subcamada, que suporta serviços e ajuda na alocação inteligente de recursos (Hassija et al., 2019; Lin et al., 2017). Cada aplicação tem suas características e é impossível prever todas as vulnerabilidades que possam afetá-las. Portanto, as questões de segurança abaixo são algumas das ameaças que podem potencialmente afetar aplicações e serviços baseados em nuvem.

Phishing - é uma praga virtual que visa obter fraudulentamente dados confidenciais do usuário, tais como identificador e senha. O *phishing* geralmente atinge o usuário final a partir de *e-mails* ou *websites* maliciosos (Benavides et al., 2020; Guarda et al., 2019). Um oponente que acesse o sistema com credenciais administrativas pode enviar comandos falsos para atuadores e alterar as configurações do sistema. Em casos críticos, o atacante pode interferir nos processos de tomada de decisão ou outros processos internos. É impossível evitar esses ataques, mas sistemas de controle de acesso seguro podem mitigá-lo. Entretanto, a proteção mais eficiente é fazer com que os próprios usuários se mantenham vigilantes enquanto navegam na rede (Lin et al., 2017).

Scripts maliciosos - a conectividade das soluções agrícolas com a Internet lhes permite interagir com outros serviços e usuários *on-line*. Esta interação as torna alvos de *scripts* maliciosos, tais como *applets* Java, *scripts* Active-X e *cross-site scripting* (XSS) (Kumar et al., 2016; Lin et al., 2017). Os *scripts* maliciosos podem enganar os clientes, injetar informações falsas, acessar informações sensíveis e quebrar os mecanismos de segurança. Os criminosos cibernéticos frequentemente utilizam este ataque para fins pessoais, financeiros e políticos. A partir de *scripts* maliciosos, eles podem danificar ou interromper a operação do serviço, exibindo anúncios indesejados e extorquindo dinheiro (Khan et al., 2017).

Negação de Serviços - este ataque causa interrupções de serviço por sobrecarga do tráfego da rede, ou por inundação do serviço com múltiplas solicitações (Shurman et al., 2019). Configurações fracas de segurança permitem iniciar este ataque a partir da Internet ou de um subsistema (Hassija et al., 2019). Tais ataques privam os usuários legítimos do uso dos serviços, impedem o processamento ou armazenamento adequado de informações não persistentes, reduzem a eficiência de sistemas críticos (como controles ambientais em estufas de cogumelos), e podem até causar um completo desligamento do sistema.

Essa camada é composta por aplicações e serviços baseados na nuvem, por isso contém todos os problemas de segurança relacionados à nuvem. Nela, as aplicações e os recursos estão expostos a ataques baseados na Internet tornando urgente a adoção de medidas preventivas. Geralmente, a principal preocupação é com a privacidade, visto que grandes volumes de dados sensíveis são armazenados e processados. No entanto, é importante ir além da privacidade e adotar medidas para garantir a disponibilidade e a integridade de todo o sistema.

Os sistemas agrícolas inteligentes incorporam um conjunto de dispositivos, com maiores ou menores níveis de limitações, que interagem uns com os outros. Muitos pontos fracos são

decorrentes das limitações dos dispositivos, que impossibilitam o uso de ferramentas e técnicas de segurança existentes. Tecnologias desenvolvidas para outros sistemas, como IoT ou Indústria 4.0, suportam a segurança, mas sua utilização requer recursos de processamento e memória que alguns dispositivos não possuem. Entretanto, é necessário conhecer as vulnerabilidades existentes e criar mecanismos para mitigar os efeitos dos incidentes. Então, as medidas de segurança podem ser implementadas nas camadas mais altas e em dispositivos equipados com os recursos necessários. As aplicações, a nuvem, a rede e a borda devem ser equipadas com recursos adequados, além de adotar medidas de segurança mais robustas para garantir a eficiência e a confiabilidade do sistema.

2.3 CONFIABILIDADE EM SISTEMAS AGRÍCOLAS INTELIGENTES

Os sistemas agrícolas inteligentes são uma espécie de sistema ciberfísico, visto que integram capacidades de computação, comunicação e armazenamento em conjunto com o monitoramento e controle de entidades do mundo físico. Esses são caracterizados por desempenharem funções de monitoramento e controle de forma correta, segura, eficiente e em tempo real. Assim como todos os sistemas de informação e comunicação, os sistemas agrícolas digitais possuem como propriedades fundamentais *i*) a funcionalidade; *ii*) o desempenho; *iii*) a dependabilidade e a segurança; e *iv*) o custo (Sanislav e Miclea, 2012).

A dependabilidade representa a capacidade de prestar um serviço em que se pode confiar, ou seja, que apresente falhas em uma frequência e gravidade em níveis aceitáveis. Usualmente é definida por atributos como confiabilidade, disponibilidade, segurança, integridade, confidencialidade e manutenibilidade (Avizienis et al., 2001; Sanislav e Miclea, 2012). Segundo Avizienis et al. (2001), a dependabilidade pode ser afetada por anomalias, falhas e erros que desviam o sistema do seu funcionamento correto. Em sistemas agrícolas isso pode ocorrer por falhas físicas ou lógicas em dispositivos, problemas na comunicação e transmissão, e pela ação de agentes externos, sejam eles maliciosos ou não. A violação dessa propriedade pode incorrer em falhas nas funções de monitoramento e controle.

Em muitos casos, um sistema pode deixar de funcionar corretamente devido à baixa qualidade dos dados. Este é um conceito amplo, relacionado a propriedades como precisão, pontualidade, completude, consistência, relevância e aptidão para uso. Essas propriedades afetam diretamente a qualidade da informação produzida e, conseqüentemente, a tomada de decisões, que é um elemento fundamental, seja ela realizada por um profissional especializado ou pelo próprio sistema (Janssen et al., 2017).

Como a agricultura digital gera dados em grande escala, com alta variabilidade na quantidade e velocidade, é difícil manusear conjuntos de dados através de ferramentas e técnicas tradicionais. Por isso, é importante utilizar recursos que melhorem a qualidade dos dados, identificando anomalias, falhas e erros que possam prejudicar a dependabilidade do sistema. Os sistemas de detecção de intrusão são recursos amplamente adotados para aumentar a segurança, especialmente no que tange à detecção de comportamentos anômalos ligados a ataques cibernéticos. No entanto, ataques são apenas parte das ameaças que podem afetar a confiabilidade de um sistema agrícola. A acurácia dos dados e, conseqüentemente, do próprio sistema pode ser afetada de forma ainda mais crítica pela ocorrência de falhas e erros em recursos de *hardware*, o que é bastante comum quando se trata de sensores. Contudo, não foram encontrados recursos de segurança capazes de identificar anomalias em dados manipulados por sistemas desenvolvidos para a IoT.

2.4 A AMEAÇA DO AGROTERRORISMO

O terrorismo é uma preocupação frequente entre os governantes de vários países, especialmente após os atentados de 11 de setembro de 2001. Os ataques terroristas mais conhecidos são aqueles que envolvem armas, porém os avanços em diversas áreas possibilitaram o desenvolvimento de outras formas de conflitos que podem causar grande destruição. Após a I Guerra Mundial foram estabelecidos acordos multilaterais que restringem o uso de armas de destruição em massa. Desde então, novos conflitos têm alcançado crescente importância, especialmente aqueles que envolvem guerra psicológica e geram alarmismo, destruição econômica e ecológica, e ameaças à população civil (ÓZSVÁRI et al., 2017). Entre esses novos conflitos está o agroterrorismo, que tem preocupado cada vez mais as autoridades, especialmente com a crescente tensão internacional ocorrida desde o início do século XXI (Monke, 2007).

O agroterrorismo é uma subárea do bioterrorismo, que visa a disseminação de doenças de origem animal e vegetal, causando medo, prejuízos financeiros e, eventualmente, instabilidade social (ÓZSVÁRI et al., 2017; Monke, 2007). Um exemplo, que poderia ser enquadrado como agroterrorismo, ocorreu recentemente e ficou conhecido como “*as sementes misteriosas da China*”. Este caso caracterizou-se pelo recebimento de pequenos pacotes contendo sementes oriundas da China e Malásia. Os pacotes foram destinados a pessoas comuns de países como Brasil, Estados Unidos, Portugal, Canadá e Reino Unido (Correio Brasiliense, 2020; UOL, 2020a; Coelho, 2020). Não se sabe ao certo o objetivo do envio das sementes. Uma das possibilidades consideradas é que se trata de um golpe, conhecido como “*brushing*”¹³ (UOL, 2020a,b). Apesar dessa possibilidade, o recebimento das sementes deixou autoridades em alerta e causaram certa agitação, uma vez que as sementes podem representar risco fitossanitário. Análises do Ministério da Agricultura do Brasil e outros laboratórios de referência em sanidade vegetal apontaram a presença de ácaros, fungos, bactérias e pragas quarentenárias, o que pode colocar a agricultura brasileira em risco (Natasha Werneck, 2020; Catraca Livre, 2020; UOL, 2020b). Embora o real objetivo do envio das sementes não esteja claro, é fato que elas poderão causar enormes prejuízos na agricultura e, conseqüentemente, na economia do país.

Normalmente, ataques como agroterrorismo não são a primeira opção dos criminosos, pois não causam o mesmo drama e espetáculo comuns a ataques violentos, como assassinatos e bombardeios. Todavia, eles podem ser utilizados para ampliar a agitação social causada por ataques menores e independentes (Olson, 2012). Em geral, o principal objetivo desses ataques não é destruir plantas e animais, mas causar crises agrícolas e na indústria alimentícia, gerando agitação social e perda de confiança no governo (Monke, 2007; Olson, 2012). Incidentes no setor alimentar podem promover compra desenfreada, devido ao pânico causado pelo provável desabastecimento, chamando a atenção da imprensa que alarma ainda mais a população e aumenta o pânico, realimentando esse círculo vicioso (Norton, 2016). Tudo isso gera um enorme impacto na cadeia produtiva. O agroterrorismo pode ter diferentes motivações, tais como (Olson, 2012):

- grupos terroristas podem querer causar danos econômicos a uma nação;
- oportunistas econômicos podem tentar manipular mercados;
- pessoas desequilibradas ou descontentes podem cometer ataques com motivações idiossincráticas, ou narcisistas;

¹³*Brushing* é o envio de mercadorias não solicitadas, por parte de um vendedor, para um grande número de indivíduos, permitindo a publicação de falsas avaliações positivas desse produto. Esta é uma técnica fraudulenta utilizada na Internet destinada a aumentar a confiança dos consumidores em um determinado produto.

O desenvolvimento de novas tecnologias, como sistemas inteligentes, pode contribuir para o surgimento de novas categorias de conflitos. A criação de sistemas agrícolas ligados à Internet, por exemplo, pode criar oportunidades para o agroterrorismo cibernético ou ciberagroterrorismo. Este pode utilizar sistemas computacionais de ambientes agrícolas para danificar lavouras, criações de animais e gerar prejuízos. Os sistemas agrícolas inteligentes são especialmente vulneráveis, visto que estão conectados à Internet e ainda possuem baixo nível de segurança.

2.4.1 Ciberagroterrorismo e a Agricultura Inteligente

Embora os riscos sejam grandes, ainda não há estudos sobre os ataques cibernéticos direcionados à agricultura digital. Os atacantes podem explorar diferentes vulnerabilidades presentes nos sistemas. Dispositivos da camada de percepção, como sensores e atuadores, podem estar acessíveis a agentes maliciosos capazes de adulterar o sistema, danificar os equipamentos ou inserir nós maliciosos. Equipamentos que usam tecnologia sem fio para se conectar à borda podem sofrer com ataques que injetem ruídos ou perturbem os sinais de radiofrequência. A comunicação entre os dispositivos da camada de percepção não pode ser criptografada, devido às limitações dos dispositivos, resultando em trocas de mensagens em texto limpo. Assim, um oponente que esteja no meio da comunicação consegue ler os dados transmitidos e forjar dados falsos.

Os dispositivos da borda da rede possuem mais recursos computacionais, possibilitando a inclusão de mecanismos de segurança. A literatura (revisão apresentada na seção 2.5) mostra que o controle de acesso por usuário e senha é um dos mecanismos comumente adicionados. Não foram encontradas evidências da inclusão de outros mecanismos importantes, como aqueles voltados para proteção dos serviços de rede, das *interfaces* de comunicação, da transferência de dados ou do gerenciamento de dispositivos. Entrevistas realizadas com fabricantes de sistemas para agricultura, mostrou que existe pouca segurança incluída. Vários entrevistados não responderam os questionamentos sobre segurança e atribuíram a responsabilidade para outras partes envolvidas. Enquanto desenvolvedores e prestadores de serviços se eximem da responsabilidade, a agricultura digital permanece vulnerável.

Os dispositivos de borda empregados na agricultura são bastante similares aos utilizados em outros contextos da Internet das Coisas. Isso significa que os riscos e vulnerabilidades também são similares. Uma pesquisa sobre as *Common Vulnerabilities and Exposures* (CVE) (Mitre Corporation, 2022) envolvendo dispositivos IoT mostra o registro de mais de mil vulnerabilidades até 2022. Os relatórios publicados pela *Open Web Application Security Project* (OWASP) (OWASP Foundation, 2018) apontam para a existência de senhas fracas e definidas em código. Essas senhas podem ser facilmente adivinhadas e permitir a obtenção de acesso não autorizado. Subsistemas inseguros (redes, componentes e *interfaces*) ampliam consideravelmente a gama de potenciais ataques.

2.5 PANORAMA DA SEGURANÇA INCORPORADA PELOS SISTEMAS AGRÍCOLAS INTELIGENTES

Nos últimos anos, houve um esforço crescente para desenvolver sistemas inteligentes para o aprimoramento das atividades agrícolas. Os agricultores geralmente conduzem essas atividades em campo aberto (áreas de lavoura) ou estufas. Este trabalho analisa projetos desenvolvidos para agricultura inteligente e explora informações sobre a segurança implementada por eles.

Neste escopo, a maioria dos esforços se concentra nos processos de irrigação, detecção de doenças, gerenciamento de lavouras e rastreabilidade. O controle pode ser automático ou

manual. Em ambos os casos, o sistema usa sensores para monitoramento e atuadores para modificar o ambiente. A decisão sobre as ações dos atuadores pode ser tomada automaticamente pelo sistema ou manualmente por um usuário. Alguns projetos só automatizam as fazendas, enquanto outros integram as tecnologias da Indústria 4.0 ou da IoT.

É importante salientar que a maioria dos projetos de Agricultura 4.0 se baseia em tecnologias de IoT e podem herdar suas vulnerabilidades de segurança. Alguns projetos utilizam protocolos com mecanismos de segurança. Outros não consideram a segurança de forma alguma. Protocolos como MQTT e CoAP¹⁴ desabilitam alguns mecanismos de segurança por padrão, e o desenvolvedor deve habilitá-los de acordo com as exigências de cada projeto. Como os pesquisadores não informam a ativação desses elementos, eles provavelmente permanecem desativados. A Tabela 2.2 apresenta uma taxonomia dos atuais recursos de segurança da agricultura inteligente.

Tabela 2.2: Taxonomia de Segurança na Agricultura Inteligente

Alvo de Segurança	Recursos de Segurança	Soluções
Sem alvo de segurança	Nenhum	Sales et al. (2015); Goap et al. (2018); Mahalakshmi (2018); Rajalakshmi e Devi Mahalakshmi (2016); Thorat et al. (2018); Yoon et al. (2018); Wongpatikaseree et al. (2018)
Troca de Dados	HTTPS, COAP, MQTT	Khelifa et al. (2015); Minh et al. (2017); Ruengittinun et al. (2017); Zhao et al. (2017)
Controle de Acesso	Autenticação IP	Nageswara Rao e Sridhar (2018)
	Gerência de Usuários e dispositivos	Oliver et al. (2018)

FONTE: O Autor (2020)

O artigo de Goap et al. (2018) apresenta um sistema de previsão das necessidades de irrigação baseado em informações climáticas e ambientais. O sistema utiliza dados coletados por sensores para prever a umidade do solo e fornece sugestões de irrigação. O usuário final interage com o sistema a partir de uma página *web*. Os autores não apresentam nenhuma característica de segurança, processos de validação ou verificações de falhas nas fases de coleta, transferência ou armazenamento de dados. A falta de segurança torna os sistemas vulneráveis a todos os ataques apresentados na Seção 2.2, ou seja, o sistema é altamente inseguro. Incidentes que levam ao corrompimento ou à imprecisão dos dados resultam em sugestões erradas e induzem às decisões equivocadas. Decisões incorretas podem danificar as plantações e reduzir a adoção do sistema.

Da mesma forma, Rajalakshmi e Devi Mahalakshmi (2016) desenvolveram um sistema para monitorar os campos de cultivo através da umidade do solo, temperatura, umidade relativa do ar e níveis de luminosidade. O controle da irrigação pode ser manual ou automático, realizado por aplicações *web* ou móveis¹⁵. Os autores não informam sobre recursos de segurança, o que expõe o sistema a toda a gama de ataques apresentados anteriormente. O controle de atuadores acionados por comandos de uma aplicação *web* ou móvel, sem recursos de segurança rígidos, é uma excelente oportunidade para adversários oportunistas, que podem usar *scripts* maliciosos e acesso não autorizado para manipular o sistema.

Zhao et al. (2017) propõem uma solução para irrigação inteligente que controla os dispositivos de irrigação. O controle é feito remotamente, por um servidor, e gerenciado por um profissional a partir de uma aplicação *web*. Não há detalhes sobre os recursos de segurança, o que

¹⁴MQTT e CoAP são protocolos para troca de mensagens desenvolvidos para comunicação máquina-a-máquina.

¹⁵Aqui o termo “aplicações móveis” se refere às aplicações desenvolvidas para dispositivos móveis, como *tablets* ou *smartphones*.

pode permitir que oponentes obtenham acesso indevido ao sistema, injetem dados maliciosos ou conduzam qualquer ataque anteriormente relatado para desviar o sistema de seu funcionamento normal.

Ruengittinun et al. (2017) introduzem um Ecosistema Agropecuário Hidropônico (HFE, do inglês *Hydroponic Farming Ecosystem*) para monitorar o ambiente de crescimento. O controle é automático e o usuário pode usar uma interface *web* para monitorar a agricultura. Intervenções automáticas requerem proteção rigorosa para evitar ruídos nos dados, injeção de dados falsos e outros ataques ou falhas que possam corromper os dados e perturbar a confiabilidade do sistema. No entanto, o HFE não fornece recursos de segurança para evitar os ataques apresentados na Seção 2.2.

Thorat et al. (2018) projetaram uma solução inteligente para a detecção de doenças foliares. O sistema processa informações provenientes de sensores e câmeras para prever doenças. O usuário final interage com o sistema através de uma aplicação móvel ou *web*. Os autores não discutem os detalhes de implementação ou segurança. Se este sistema for parte de um processo de controle de doenças e receber dados corrompidos ou maliciosos, ou se um oponente o comprometer, então os incidentes de segurança podem dificultar a detecção de doenças e causar o mau uso de recursos agrícolas. Em casos críticos, isto pode causar a perda de toda a produção.

Wongpatikaseree et al. (2018) apresentam o NETPIE, um sistema que fornece informações sobre produtos agrícolas. Usando um conjunto de dispositivos de percepção, o NETPIE controla e monitora o ambiente de cultivo. As informações de produção são resumidas e salvas em um código QR e disponibilizadas para o cliente. Assim como os outros sistemas apresentados, o NETPIE não discute os recursos de segurança. Qualquer um dos ataques que interrompam a precisão dos dados pode quebrar a confiabilidade das informações resumidas no código QR.

Sales et al. (2015) propõem um sistema de comunicação baseado em nuvem para Redes de Sensores e Atuadores Sem fio (WSAN, do inglês *Wireless Sensor and Actuator Network*) para monitorar e controlar dispositivos agrícolas. O sistema monitora as condições ambientais, prevê os requisitos de irrigação e age automaticamente sobre o meio ambiente. O documento descreve a arquitetura do sistema, incluindo dispositivos e protocolos, mas não apresenta qualquer recurso de segurança, permitindo que a WSAN permaneça vulnerável aos ataques mostrados na Seção 2.2.

Yoon et al. (2018) apresentam um sistema agrícola inteligente para o troca de dados entre o servidor, o *gateway* e os nós. O documento descreve a construção do sistema, não menciona nenhuma interação do usuário ou controle remoto e não demonstra nenhuma preocupação com a segurança. Por ser um sistema para troca de dados, os ataques mais críticos são aqueles que afetam a camada de rede, tais como DoS, interrupção de sinal e roteamento.

Mahalakshmi (2018) introduz um sistema de irrigação automatizado. O documento apresenta a construção passo a passo do sistema, que monitora e controla o fluxo de água remotamente. Embora o sistema controle e monitore os dispositivos de irrigação, não há evidência da adição de funcionalidades de segurança. Os ataques à camada de percepção, assim como os ataques que causam uma negação de serviço, podem danificar o funcionamento correto do sistema. Acesso não autorizado, *scripts* maliciosos e injeção de dados falsos podem manipular deliberadamente o sistema.

Algumas soluções acrescentam um pequeno nível de segurança. A estratégia de Khelifa et al. (2015), por exemplo, inclui a criptografia na comunicação entre as aplicações na nuvem e os usuários. Os autores propõem um sistema de irrigação inteligente controlado remotamente pelo usuário. Os agricultores controlam o processo de irrigação a partir de uma aplicação móvel. Esta estratégia usa HTTPS para criptografar a comunicação entre o servidor e o *smartphone*. O uso de criptografia protege os dados em trânsito, impedindo que um adversário intercepte a

comunicação, obtendo informações sensíveis e personificando a aplicação móvel. Entretanto, não há informações sobre outros recursos de segurança implantados pelo sistema, o que expõe o sistema aos outros ataques apresentados anteriormente.

Outra proposta que utiliza HTTPS é a de Minh et al. (2017), que desenvolveu um sistema inteligente para gerenciar e controlar fazendas de cogumelos e milho híbrido. Este sistema controla os ambientes de produção automaticamente e é gerenciado remotamente pelo usuário final. O servidor *web* usa HTTPS para a comunicação com o usuário, protegendo os dados em trânsito. Entretanto, este recurso de segurança é insuficiente, considerando que o sistema controla automaticamente as bombas de água, os níveis de luz e os ventiladores. O controle automático dos ambientes de produção, especialmente para culturas tão sensíveis quanto os cogumelos, requer o uso de mecanismos rigorosos de segurança para evitar que a injeção de dados falsos, o acesso não autorizado ou a negação de serviço desviem o sistema de seu funcionamento normal.

Oliver et al. (2018) introduzem um sistema chamado SEnviro. Este sistema é projetado para monitorar remotamente os vinhedos e prever algumas doenças. O documento apresenta a plataforma desenvolvida e não discute a predição. O sistema inclui um gerenciador de usuários e dispositivos, que permite controlar usuários e dispositivos autorizados para interagirem com o sistema. O controle de acesso impede que dispositivos ou usuários não autorizados tenham acesso ao sistema e atuem maliciosamente. Entretanto, este recurso é insuficiente para proteger uma plataforma projetada para prever doenças e monitoramento remoto, pois não impede e não detecta eventos que possam interferir na precisão dos dados ou que possam levar o sistema a um estado não confiável.

Da mesma forma, Nageswara Rao e Sridhar (2018) propõem um sistema de monitoramento remoto de campo de cultivo e de monitoramento automático de irrigação usando tecnologias IoT. O sistema utiliza dados coletados de sensores para estimar a quantidade de água necessária para irrigação. Assim como o controle de acesso apresentado por Oliver et al. (2018), o esquema de autenticação usado por Nageswara Rao e Sridhar (2018) evita o acesso não autorizado ao serviço, mas não protege a borda e outros subsistemas. A fraca proteção dos sistemas de controle automático é crítica, pois incidentes que afetam a precisão dos dados ou causam mau funcionamento do sistema podem resultar em perdas significativas para a plantação.

Resumindo os trabalhos relacionados, de uma perspectiva de segurança, eles usam sensores e atuadores sem nenhum mecanismo de segurança. Além disso, não há informações de segurança no *gateway*. Os sistemas desenvolvidos até o momento não apresentam informações sobre privacidade na transmissão de dados entre a percepção e a nuvem, ou autenticação de dispositivos. Recursos como controle de acesso, gerenciamento de identidade ou criptografia acrescentam um pouco de segurança à comunicação na Internet. A Tabela 2.3 mostra que a pouca segurança nos sistemas agrícolas está limitada à privacidade e transmissão de dados confiáveis entre o usuário e a nuvem ou entre o *gateway* e a nuvem.

Até o momento, os recursos de segurança são mencionados apenas na camada de aplicação. Enquanto Goap et al. (2018) e Sales et al. (2015) usam HTTPS para comunicação entre a nuvem e a aplicação do usuário final, a maioria dos sistemas usa os protocolos HTTP, CoAP e MQTT sem qualquer integração com os protocolos SSL ou TLS. O mesmo acontece com o controle de acesso, cuja implementação utiliza recursos limitados (Nageswara Rao e Sridhar, 2018; Oliver et al., 2018) ou não é relatada. A ausência de recursos de segurança robustos para a comunicação entre a nuvem e o usuário final abre várias brechas de segurança. Não há informações sobre configurações seguras em sistemas de gerenciamento de banco de dados ou o uso de técnicas de busca segura de dados em aplicações *web*. Logo, estas características provavelmente não estão incluídas.

Outros pesquisadores também analisaram a segurança na agricultura inteligente. Gupta et al. (2020) discute questões de segurança e privacidade no ambiente físico cibernético dinâmico e distribuído da agricultura de precisão. Os autores apresentam um conjunto de projetos e pesquisas que abordam a segurança cibernética na agricultura inteligente, apontando as contribuições e limitações de cada projeto. Há vários dos projetos analisados pelos autores que não implementam recursos de segurança. Os projetos que integram a segurança incorporam soluções limitadas ou insuficientes.

Tabela 2.3: Recursos de segurança implementados em Sistemas Agrícolas

Camada	Questões de Segurança	Recursos de Segurança	Soluções
Aplicação	Roubo de dados	HTTPS	Khelifa et al. (2015), Minh et al. (2017)
	<i>Sniffing</i>	HTTPS	Khelifa et al. (2015), Minh et al. (2017)
	Controle de Acesso	MQTT, COAP, Autenticação IP, Gerência de Controle de Usuários e Dispositivos	Khelifa et al. (2015), Zhao et al. (2017), Ruengittinun et al. (2017), Minh et al. (2017), Nageswara Rao e Sridhar (2018), Oliver et al. (2018)
	<i>Phishing</i>	Nenhum recurso informado	
	<i>Scripts</i> maliciosos	Nenhum recurso informado	
	Negação de Serviços	Nenhum recurso informado	
Borda	<i>Man-in-the-middle</i>	MQTT, COAP	Khelifa et al. (2015), Zhao et al. (2017), Ruengittinun et al. (2017), Minh et al. (2017)
	Ataques de inicialização	Nenhum recurso informado	
	Acesso não autorizado	Nenhum recurso informado	
	Encapsulamento de assinaturas	Nenhum recurso informado	
	Inundação	Nenhum recurso informado	
Rede	Interrupções de Sinal	Nenhum recurso informado	
	DoS/DDoS	Nenhum recurso informado	
	Ataques ao tráfego de dados	Nenhum recurso informado	
	Roteamento	Nenhum recurso informado	
Percepção	<i>Tampering</i>	Nenhum recurso informado	
	Falhas de <i>hardware</i>	Nenhum recurso informado	
	Ruídos nos dados	Nenhum recurso informado	
	Captura de nós	Nenhum recurso informado	
	Nós falsos	Nenhum recurso informado	
	Injeção de dados falsos	Nenhum recurso informado	
Privação de sono	Nenhum recurso informado		

FONTE: O Autor (2020)

Demestichas et al. (2020) fornece uma visão geral das principais ameaças existentes e potenciais para a agricultura. Os pesquisadores discutem a evolução das soluções TIC e seu impacto na agricultura. Eles apresentam ameaças e vulnerabilidades emergentes, assim como algumas medidas de mitigação e estratégias de segurança. Os autores afirmam que, apesar da crescente preocupação relacionada ao ciberterrorismo, ainda não há investimento suficiente para melhorar a proteção da segurança. Empresas e indústrias maiores já investem em sistemas e medidas de segurança seguros e eficientes, mas as empresas e fazendas menores frequentemente não dispõem de recursos financeiros, tempo e planos para projetar e implementar medidas adequadas contra possíveis ciberataques.

Atualmente, a agricultura digital é um alvo fácil para os agentes maliciosos. Os ataques podem ter várias motivações, tais como razões comerciais, ideológicas ou mesmo terroristas.

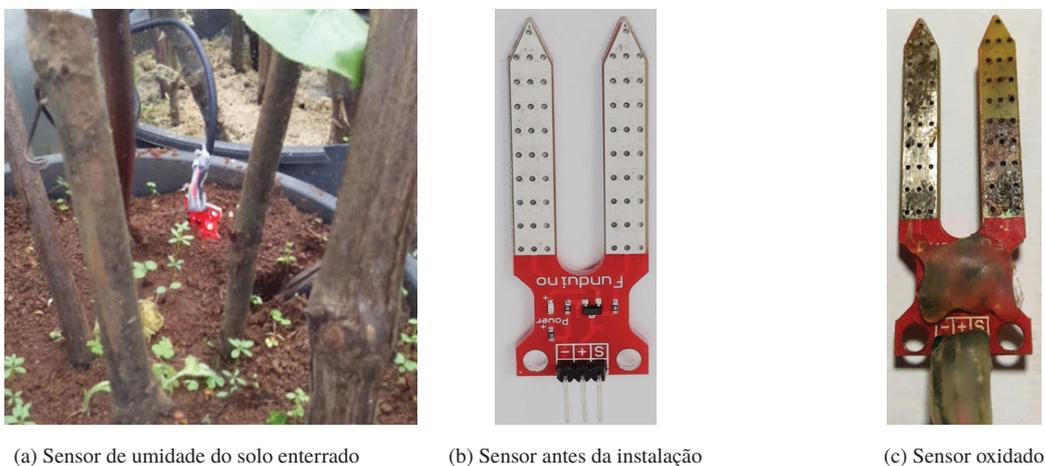
Por exemplo, grupos terroristas podem infligir danos econômicos a uma nação, oportunistas econômicos podem tentar manipular mercados e um funcionário pode prosseguir com um ataque por diversas razões (Olson, 2012). Portanto, é urgente acrescentar segurança como um recurso essencial para a agricultura inteligente, contribuindo para o desenvolvimento e a popularização de sistemas confiáveis e eficientes.

2.6 FALHAS E ERROS EM DISPOSITIVOS DE COLETA DE DADOS

Os sistemas agrícolas digitais são formados por uma série de dispositivos e subsistemas responsáveis por coletar dados, transferi-los, processá-los e armazená-los (Triantafyllou et al., 2019). A coleta de dados é geralmente realizada por sensores e transdutores, dispositivos físicos cuja dependabilidade está atrelada a diversos fatores, como os materiais utilizados para sua fabricação e detalhes de projeto. Equipamentos mais baratos tendem a ser fabricados com materiais de menor qualidade, o que pode resultar na sua degradação. Como consequência, os dados medidos não correspondem à realidade, o que caracteriza quebra da integridade.

Por exemplo, sensores de umidade do solo são geralmente instalados em contato com a terra e podem sofrer corrosão devido às propriedades físico-químicas do solo se não forem fabricados com materiais resistentes. Apesar disso ser bastante comum, muitas soluções para IoT têm utilizado sensores susceptíveis à corrosão, como o mostrado na Figura 2.4(a). O resultado são falhas nas leituras decorrentes do processo de deterioração. A Figura 2.4(b) mostra o sensor antes da sua instalação e a Figura 2.4(c) o mesmo sensor após 5 meses dentro de uma estufa agrícola.

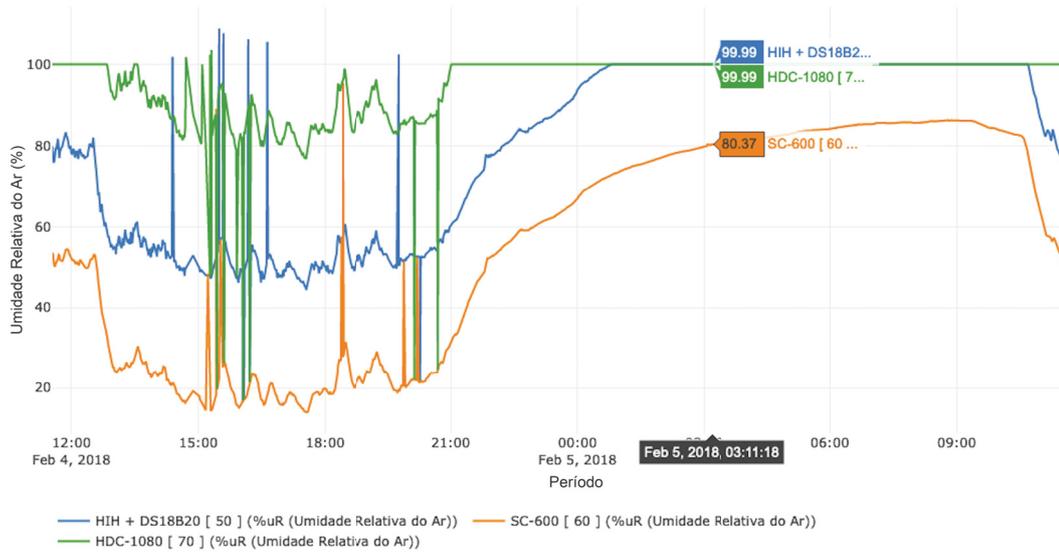
Figura 2.4: Sensor de umidade do solo instalado dentro de uma estufa agrícola



FONTE: O Autor (2019)

Outra situação é aquela onde os dispositivos são expostos a situações que resultam em erros de leitura, os quais podem ser resultado de distúrbios de carga ou falhas de determinado componente. Por exemplo, sensores de umidade relativa do ar podem sofrer saturação quando expostos a altos índices de umidade. Quando saturados, as medições ficam prejudicadas e o resultado pode ser observado na Figura 2.5. O gráfico apresenta as medições de umidade relativa do ar realizadas por três sensores instalados dentro de uma estufa agrícola, em que apenas o SC-600 não sofreu saturação. Os dispositivos em questão não foram expostos ao contato direto com água proveniente de chuva ou de irrigação, mas estiveram sujeitos ao contato com neblina e água proveniente de processos de evaporação.

Figura 2.5: Leituras de umidade relativa do ar



FONTE: O Autor (2019)

Há ainda os casos onde as condições inerentes ao ambiente em que o dispositivo está instalado podem resultar em falhas no funcionamento. É o caso dos sensores de velocidade do vento que sofrem com o acúmulo de partículas em suspensão no ar, quando instalados em ambientes empoeirados, como aviários. Nesses locais, o pó acumula sobre o dispositivo e as hastes têm dificuldades para girar, causando erros de medição. A solução para esse problema é a limpeza das hastes, que é normalmente executada por um trabalhador agrícola. No entanto, se o trabalhador não manusear o sensor com perícia, pode danificá-lo fisicamente ou desajustar suas hastes (Peternela, 2019).

Situações ainda mais graves podem ocorrer em áreas abertas, quando os dispositivos podem ser facilmente acessados por agentes externos, como pessoas, animais e equipamentos agrícolas. Estes podem deslocar ou remover o dispositivo, intencionalmente ou não, quando estiverem transitando nas proximidades. O resultado dessa ação pode ser a completa ou parcial danificação do dispositivo, ou alterações no seu local de instalação. Qualquer modificação não autorizada pode causar inconsistências nas medições e prejudicar a confiabilidade dos dados. Por exemplo, o sensor de umidade do solo pode informar aridez, quando suas hastes não estiverem totalmente em contato com a terra e um sensor de velocidade do vento pode deixar de registrar rajadas de vento após sofrer um choque que o danifique.

Falhas e erros são muito comuns em sensores e muitos exigem calibragem periódica, enquanto outros requerem a regular substituição ou manutenção. Esses processos são realizados por pessoas e não há como precisar se a manutenção foi realizada no tempo certo e com a perícia necessária. Além disso, podem ocorrer avarias nos intervalos entre manutenções, exigindo identificação da situação e sua pronta remediação.

Sendo assim, é fundamental que as soluções desenvolvidas para Agricultura 4.0 sejam providas de mecanismos, ferramentas e tecnologias capazes de mitigar ou evitar a ocorrência de falhas e erros. Em muitos casos, não é possível evitar que esses incidentes aconteçam, mas é factível evitar que sua ocorrência resulte em quebra da confiabilidade do sistema. Para isso, é preciso identificar a ocorrência de eventos que indiquem ou estejam relacionados a falhas e erros. Com base nesses eventos pode-se mitigar ou até mesmo prever incidentes, maximizando a acuracidade e a confiabilidade do sistema.

3 DETECÇÃO DE ANOMALIAS

Sistemas de detecção de anomalias podem ser aliados valiosos para aumentar a segurança da Agricultura 4.0. Anomalias são eventos importantes e a sua detecção tem sido investigada em diversos domínios. Detectar anomalias significa encontrar padrões que estejam em desconformidade com o comportamento esperado ou reconhecido como normal. Sua ação pode revelar informações significativas e até mesmo críticas, por isso a detecção de anomalias tem sido utilizada para diferentes finalidades, como detecção de fraudes em cartões de crédito, seguros, detecção de falhas em sistemas críticos e segurança de sistemas cibernéticos (Chandola et al., 2009). Formalmente pode-se definir uma anomalia como sendo (Donevski e Zia, 2018):

Definição 3.1 (Anomalia). *Algo que desvia do padrão, do normal ou do esperado, podendo também estar associada a um evento não compreendido.*

Pesquisadores de diferentes áreas vêm desenvolvendo metodologias para detecção de anomalias em seus domínios. Isso tem resultado no estabelecimento de técnicas por parte de centros de pesquisa e da indústria. Todavia, alguns fatores ainda tornam essa tarefa desafiadora. Primeiramente, não é trivial encontrar o ponto exato que separa um comportamento normal de um anormal para as diferentes áreas. Além disso, cada domínio possui suas próprias categorias de dados, tornando a atividade ainda mais complexa (Ariyaluran Habeeb et al., 2019). A IoT possui o desafio adicional de gerar maciças quantidades de dados, em alta velocidade e ampla variedade. Essas características inviabilizam a utilização de muitas técnicas de detecção de anomalias existentes (Ariyaluran Habeeb et al., 2019).

Visto que o principal objetivo é encontrar padrões desconformes em um conjunto de dados, então é necessário definir uma região que represente o comportamento normal. Em seguida bastaria, em uma visão simplista, considerar anômalos todos os dados não compatíveis com essa região. No entanto, definir o comportamento normal não é elementar. Chandola et al. (2009) apresentam seis fatores que tornam essa uma tarefa desafiadora:

1. É necessário definir uma fronteira que envolva todos os comportamentos normais e os separe, precisamente, dos comportamentos anômalos;
2. Domínios distintos possuem noções diferentes de anomalias, restringindo as aplicações de uma técnica;
3. Alguns domínios não possuem uma quantidade adequada de dados rotulados para treinamento e validação de modelos;
4. Em domínios dinâmicos um comportamento inicialmente considerado normal pode não ser representativo no futuro;
5. Adversários maliciosos tendem a adaptar suas ações para fazer com que resultem em observações parecidas com as normais;
6. Os dados podem possuir ruídos similares às anomalias, dificultando sua distinção e remoção.

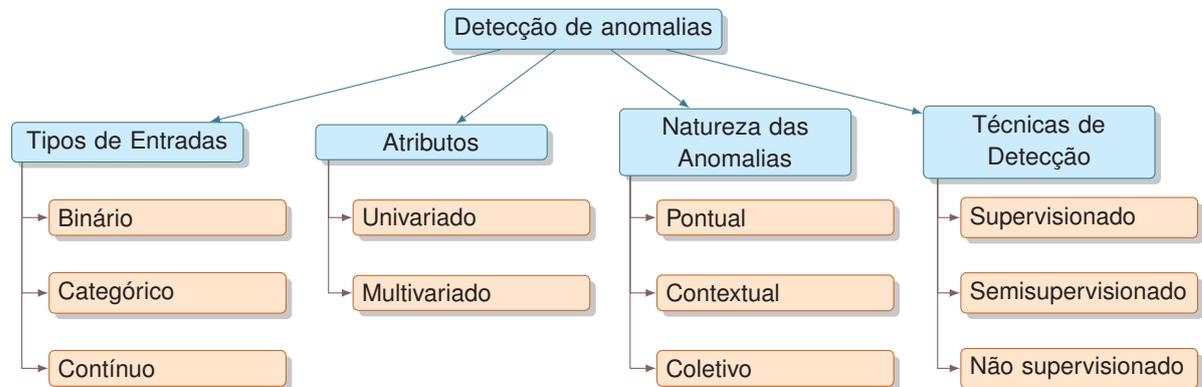
No campo da cibersegurança, Donevski e Zia (2018) afirmam que o maior desafio é definir uma fronteira que separe os comportamentos normais dos anômalos, em um ambiente

dinâmico. Além disso, modelar anomalias é uma tarefa difícil devido à quantidade reduzida de dados capazes de identificar irregularidades resultantes de atividades maliciosas. No âmbito da agricultura inteligente, ainda podem ser incluídos como desafios a alta velocidade e a enorme massa de dados concentrada na borda, aliada ao fato de que este ainda é um domínio onde as anomalias são pouco conhecidas, o que torna sua detecção uma tarefa complexa.

3.1 PANORAMA DA DETECÇÃO DE ANOMALIAS

O desenvolvimento de métodos para detecção de anomalias é determinado por fatores como tipos dos dados de entrada e seus atributos, natureza das anomalias inerentes ao domínio de aplicação e a disponibilidade de rótulos, que determina qual técnica de detecção pode ser utilizada. A Figura 3.1 apresenta uma visão geral desses fatores.

Figura 3.1: Fatores que influenciam a detecção de anomalias



FONTE: O Autor (2020)

Segundo Chandola et al. (2009), a natureza dos dados de entrada e das anomalias são elementos-chave para qualquer técnica de detecção de anomalias.

Definição 3.2 (Entradas). *São uma coleção de instâncias e dados, descritas através de um conjunto de atributos, que podem ser dos tipos binário, categórico ou contínuo.*

Cada instância pode consistir em um ou múltiplos atributos, denominados univariados, no primeiro caso, ou multivariados, no último caso. Instâncias multivariadas podem conter atributos de tipos diferentes ou de um único tipo. Com relação à natureza das anomalias, elas podem ser classificadas como pontual, contextual ou coletiva (Chandola et al., 2009; Zhou e Guo, 2018; Ariyaluran Habeeb et al., 2019).

Definição 3.3 (Anomalias pontuais). *Consistem em dados individuais considerados anômalos com relação aos demais.*

Definição 3.4 (Anomalias contextuais). *Compreendem os dados incompatíveis em relação a um contexto específico, mas não em outros contextos.*

O contexto é determinado pelo conjunto de dados e possui duas categorias de atributos: *a)* contextuais, usados para determinar o contexto ou vizinhança de uma instância, e *b)* comportamentais, que definem características não contextuais de uma instância. Por exemplo, um conjunto de dados espaciais que descrevem a precipitação média de uma região podem ter como atributos contextuais a latitude e a longitude, e como atributos comportamentais a quantidade de precipitação.

Definição 3.5 (Anomalias coletivas). *Constituem-se de uma coleção e dados relacionados classificados como anômalos em relação a todo o conjunto de dados.*

Nas anomalias coletivas os eventos individuais podem ser considerados normais, mas um conjunto maior dos mesmos dados pode ser considerado anomalia. Isso é comum em tráfego de rede, em que uma requisição TCP simples é considerada normal, mas múltiplas requisições podem estar relacionadas a um ataque DoS.

A criação de métodos de detecção eficientes depende da existência de dados rotulados capazes de representar o domínio correta e completamente. Os rótulos são associados às instâncias de dados para indicar se ela é *normal* ou *anômala*. No entanto, obter um conjunto representativo de dados precisamente rotulados pode ser excessivamente caro. Isso porque a rotulagem deve ser conduzida por especialistas humanos, que geralmente empregam um esforço significativo para marcar as instâncias. Normalmente é mais fácil obter conjuntos de dados que representem um comportamento normal do que conjuntos que incluam todos os tipos possíveis de comportamentos anômalos. Além disso, um comportamento anômalo pode sofrer mudanças e novos comportamentos podem surgir, dificultando ainda mais a obtenção de conjuntos de dados representativos.

Conforme a disponibilidade de dados rotulados, podem ser empregadas as seguintes técnicas de detecção de anomalias (Chandola et al., 2009):

Supervisionada assume a disponibilidade de um conjunto de dados representativo para treinamento do modelo, contendo classes normais e anômalas precisamente rotuladas. Durante o processo de detecção as instâncias de dados são comparadas com o modelo para determinar a qual classe elas pertencem.

Semisupervisionada exige que apenas os dados normais sejam marcados em um conjunto de dados de treinamento, o que amplia a aplicabilidade dessas técnicas.

Não supervisionada não requer dados de treinamento. As técnicas não supervisionadas supõem que as instâncias normais são muito mais comuns que as anômalas nos dados de teste.

Na detecção de anomalias supervisionada é comum construir um modelo preditivo para classes anômalas. Essa detecção não trabalha bem com distribuições desbalanceadas. Por isso, não é eficiente em domínios que não dispõem de conjuntos de dados de treinamento com instâncias anômalas em quantidades equivalentes às normais. Para mitigar o desequilíbrio entre classes, existem técnicas para injeção de anomalias artificiais para treinamento do modelo. Contudo, em alguns casos, modelar comportamentos anômalos pode ser demasiadamente complexo. Para esses casos, é mais interessante utilizar técnicas semisupervisionadas, pois elas modelam apenas comportamentos reconhecidos como normais. Essas são utilizadas especialmente em domínios em que é difícil modelar todos os possíveis comportamentos anômalos. Diversas técnicas semisupervisionadas podem ser adaptadas para operar em modo não supervisionado, introduzindo dados de testes com raras anomalias. Isso permite criar um modelo robusto para anomalias com poucas representações nas bases de testes.

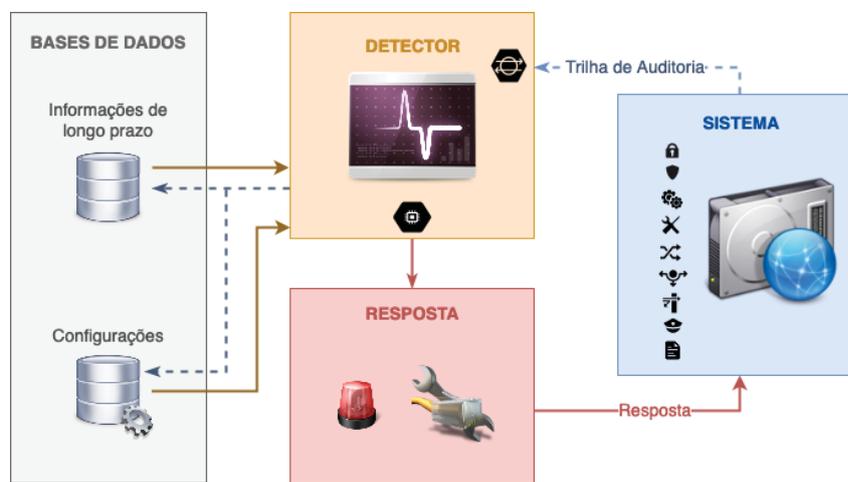
No domínio da segurança, a detecção de anomalias tem sido amplamente utilizada para detectar intrusões em redes e sistemas (Santos et al., 2018; Zarpelão et al., 2017). Sua principal aplicação é na construção de Sistemas de Detecção de Intrusão (IDS, do inglês *Intrusion Detection System*), desenvolvidos para identificar a ocorrência de ataques cibernéticos que possam prejudicar o desempenho de recursos computacionais ou de redes. Em sua maioria, os IDSs se limitam a identificar tipos específicos de ataques, sendo que alguns deles conseguem prevenir ou mitigar ataques. No contexto deste trabalho, as técnicas de detecção de intrusão podem ser úteis para a identificação de ataques cibernéticos, o que é parte da proposta aqui apresentada.

3.2 VISÃO GERAL DOS SISTEMAS DE DETECÇÃO DE INTRUSÃO

Os sistemas de detecção de intrusão são mecanismos de defesa utilizados para proteger redes e dispositivos de ataques cibernéticos (Iqbal et al., 2016; Khraisat et al., 2019). Esses mecanismos fortalecem a segurança de sistemas, especialmente daqueles que estão expostos a requisitos de segurança não mapeados previamente ou que possuem condições que os impedem de permanecer seguros (Debar et al., 1999; García-Teodoro et al., 2009). Na Agricultura 4.0 os IDSs podem ser eficientes para detectar anomalias decorrentes de ataques cibernéticos que visam manipular o sistema ou apenas desviá-lo de seu funcionamento normal.

Os IDSs podem ser estruturados de diversas maneiras, mas geralmente a detecção de intrusão envolve os elementos apresentados na Figura 3.2. O sistema é monitorado por sensores que coletam trilhas de auditoria.

Figura 3.2: Sistema de Detecção de Intrusão Simples



FONTE: Adaptado de Debar et al. (2000)

Definição 3.6 (Sistema). *Refere-se a um sistema de informação que está sendo monitorado pelo detector de intrusão.*

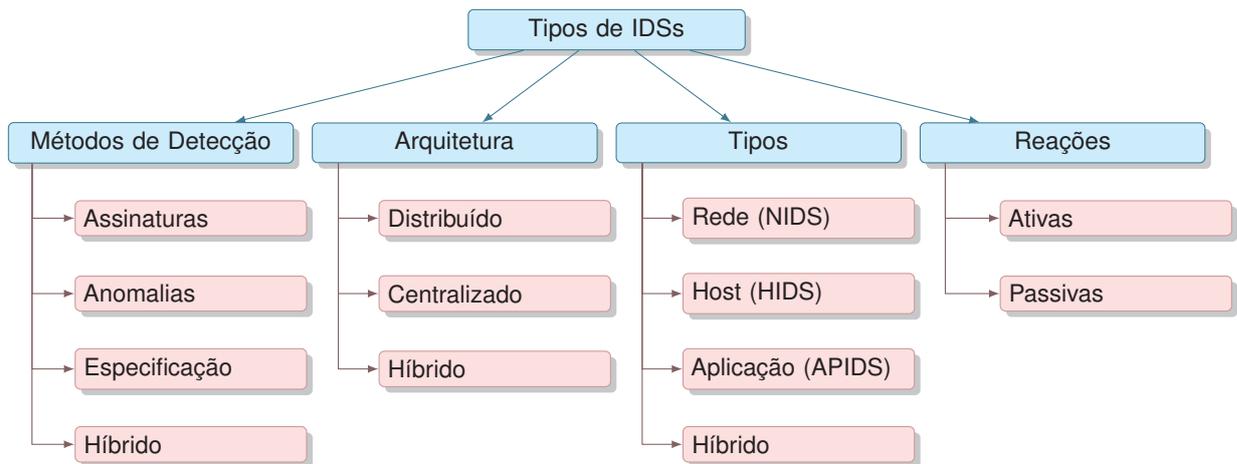
Definição 3.7 (Trilhas de auditoria). *São informações sobre os eventos de um sistema.*

As trilhas de auditoria podem conter registros de operações, arquivos, tráfego de rede, configurações de serviços ou do próprio sistema. O detector recebe as trilhas, elimina os dados desnecessários e armazena os demais em bases de dados. As trilhas de auditoria são analisadas para identificar o estado atual de segurança (Bhuyan et al., 2014; Debar et al., 2000; García-Teodoro et al., 2009).

O estado atual de segurança é obtido a partir da análise de três categorias de informações: *a)* de longo prazo, que contém dados relacionados ao método usado para detecção de intrusões, como uma base de conhecimentos sobre ataques, por exemplo; *b)* de configuração, que possui dados sobre o estado atual; e *c)* trilhas de auditoria, que contém registros sobre o evento. A partir dessas informações é tomada uma decisão que reflete a probabilidade de que os eventos auditados indiquem uma intrusão ou uma anomalia (Debar et al., 1999, 2000). Caso a probabilidade aponte para uma quebra da segurança, então será gerada uma resposta, que pode ser a emissão de alerta ou uma intervenção no sistema para mitigar o problema.

A intrusão pode ser detectada por diferentes métodos e estruturada de diversas formas. Os métodos de detecção podem utilizar estratégias baseadas em *assinaturas*, *anomalias*, *especificação* ou podem ser *híbridos*, combinando mais de um método. A detecção pode ser *centralizada* ou *distribuída*, operacionalizada pela análise dos registros de rede, de dispositivos individuais, protocolos de aplicações ou registros provenientes de diferentes origens de dados. A reação pode ser *ativa*, quando o IDS intervém no sistema, alterando sua configuração para mitigar o problema, ou *passiva*, quando emite alerta relacionado à intrusão. A Figura 3.3 apresenta uma visão geral das diferentes categorias de IDS.

Figura 3.3: Classificação dos Sistemas de Detecção de Intrusão



FONTE: O Autor (2020)

3.2.1 Métodos de Detecção de Intrusão

As intrusões são detectadas a partir da observação de eventos, analisando sua compatibilidade com operações típicas do sistema ou com ataques previamente conhecidos. De acordo com a estratégia utilizada, os IDS podem ser classificados em baseados em assinaturas e baseados em anomalias (Khraisat et al., 2019). Os detectores de intrusão baseados em anomalias utilizam técnicas de detecção de anomalias para detectar intrusões. Eles podem ser subdivididos em baseados em anomalias e baseados em especificação. A diferença entre eles é que o primeiro “aprende” o comportamento normal, enquanto no segundo um especialista humano define manualmente um conjunto de regras que caracteriza o comportamento esperado (Santos et al., 2018). Por outro lado, os métodos que identificam ataques cibernéticos analisam os eventos do sistema e os confrontam com padrões atribuídos a intrusões conhecidas, os quais ficam armazenados em bases de dados em forma de assinatura. Cada método possui vantagens e desvantagens, as quais estão resumidos na Tabela 3.1.

Definição 3.8 (Métodos baseados em anomalias). *Comparam as operações de um sistema em um dado instante com um perfil de comportamento considerado normal (Santos et al., 2018).*

Definição 3.9 (Métodos baseados em especificações). *Analisa o comportamento dos componentes de um sistema com um conjunto de regras e limites que definem um “comportamento esperado” (Santos et al., 2018).*

Definição 3.10 (Métodos baseados em assinaturas). *Confrontam os dados recebidos com um conjunto de padrões predefinidos (assinaturas) relacionados a ataques conhecidos (Santos et al., 2018).*

Tabela 3.1: Vantagens e desvantagens dos métodos de detecção de anomalias

	Métodos de Detecção		
	Anomalias	Especificação	Assinaturas
Vantagens	<ul style="list-style-type: none"> • Detecta comportamentos desconhecidos 	<ul style="list-style-type: none"> • Baixos índices de falsos positivos 	<ul style="list-style-type: none"> • Eficiente para detecção de ataques conhecidos
Desvantagens	<ul style="list-style-type: none"> • Altos índices de falsos positivos 	<ul style="list-style-type: none"> • Menor flexibilidade • Mais lento • Mais susceptível a erros 	Incapaz de identificar: <ul style="list-style-type: none"> • Variantes de ataques conhecidos • Novos ataques • Comportamentos desconhecidos

FONTE: O Autor (2020)

A detecção baseada em anomalias compara os eventos analisados com operações típicas do sistema e assinala como suspeito todo aquele que excede os limites predefinidos para comportamentos normais (Axelsson, 2015; García-Teodoro et al., 2009; Zarpelão et al., 2017). Os padrões de comportamentos são aprendidos durante a fase de treinamento e armazenados em uma base de conhecimentos. Na fase de detecção, o comportamento esperado é estimado e estabelece-se um limite que envolva comportamentos similares. A classificação entre os parâmetros normal e suspeito pode ser feita por modelos estatísticos, bases de conhecimentos ou aprendizagem de máquina (Axelsson, 2015). Essa abordagem é eficiente para detectar comportamentos desconhecidos pelo detector, porém pode apresentar índices de falso positivos superiores aos outros métodos (García-Teodoro et al., 2009; Zarpelão et al., 2017).

Similarmente, a detecção baseada em especificação busca por padrões que desviam de um conjunto de regras previamente definidas. Elas são modeladas manualmente por especialistas, descrevem explicitamente os comportamentos legítimos do sistema e os limites para ações aceitas. A modelagem pode ser feita por máquinas de estados finitos (FSM, do inglês *Finite State Machine*) ou modelos lógicos (Bhuyan et al., 2014; García-Teodoro et al., 2009; Santos et al., 2018; Zarpelão et al., 2017). Essa abordagem possui índices de falso positivos inferiores aos baseados em anomalias, pois evita identificar como intrusões atividades consideradas inofensivas. Por outro lado, é menos flexível, menos adaptável a ambientes distintos, pode ser mais demorada e propensa a erros (García-Teodoro et al., 2009; Santos et al., 2018; Zarpelão et al., 2017).

Já os detectores baseados em assinaturas buscam padrões predefinidos, chamados assinaturas, nos eventos analisados. Eventos que combinam com alguma assinatura são assinalados como intrusão. As assinaturas são armazenadas em bases de conhecimentos, contém informações sobre intrusões e seus vestígios e permitem distinguir entre operações normais e ataques, mesmo sem conhecer o comportamento do sistema (Axelsson, 2015; Bhuyan et al., 2014; García-Teodoro et al., 2009). Para isso, é preciso definir explicitamente regras de decisão capazes de identificar quais e como os rastros de intrusão podem ocorrer em operações normais. As intrusões podem ser modeladas como um conjunto de estados diferentes que possibilitem identificar sua ocorrência em um espaço de observação. Essa abordagem é bastante eficiente para detectar ataques conhecidos, porém é incapaz de identificar suas variantes, novos ataques ou

comportamentos desconhecidos, já que normalmente esses não combinam com as assinaturas armazenadas na base de dados (García-Teodoro et al., 2009; Zarpelão et al., 2017).

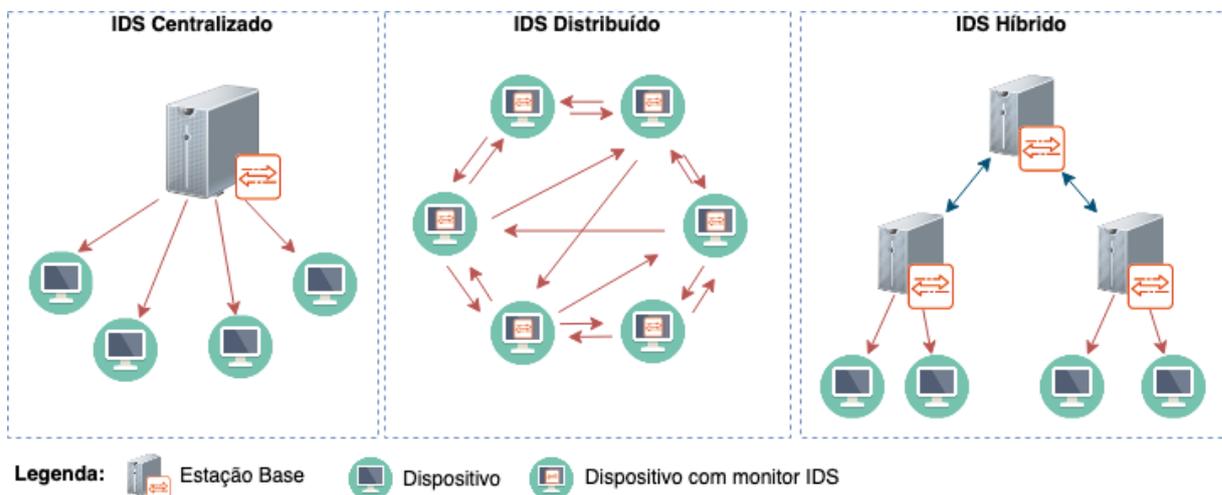
Para transpor as limitações dos métodos apresentados e aproveitar suas vantagens, pode ser utilizada uma abordagem híbrida. Há trabalhos que combinam detecção por assinaturas e por anomalias, aproveitando a simplicidade e eficiência da primeira e a flexibilidade da última (Krimmling e Peter, 2014; Zarpelão et al., 2017). Outros associam detectores baseados em anomalias e especificação, para maximizar a eficiência e a eficácia na detecção de intrusão (Cervantes et al., 2015; Zarpelão et al., 2017). A escolha dos métodos mais adequados varia conforme os objetivos e ambientes nos quais serão utilizados, mas utilizar abordagens híbridas pode maximizar o desempenho e a segurança dos sistemas.

Definição 3.11 (Métodos híbridos). *Combinam dois ou mais métodos de detecção de intrusão.*

3.2.2 Arquiteturas dos Sistemas de Detecção de Intrusão

Para coletar informações sobre anomalias ou ataques eficientemente, é importante que os IDSs sejam instalados em locais estratégicos. Em alguns casos, esses recursos têm sido localizados na borda da rede ou em dispositivos intermediários e/ou finais, dedicados ou não. O detector pode coletar dados somente localmente ou possuir sensores distribuídos em vários nós da rede, para coletar as trilhas de auditoria. A escolha de qual estrutura utilizar depende dos objetivos e do ambiente em que o IDS irá operar. A Figura 3.4 apresenta uma visão geral de cada uma das arquiteturas.

Figura 3.4: Arquiteturas IDS



FONTE: O Autor (2020)

Na arquitetura centralizada, os sistemas de detecção de intrusão são instalados em estações base, localizados na borda da rede ou em dispositivos dedicados (Farooqi e Khan, 2009; Zarpelão et al., 2017). O IDS deve estar localizado em um ponto estratégico da rede, pois irá coletar informações sobre os eventos que ocorrem nesse dispositivo. Esta é uma estratégia simples e pode ser eficiente para detectar ataques que passam por roteadores. Todavia, IDSs

centralizados são incapazes de detectar eventos que envolvem apenas os dispositivos finais ou não se propagam pela rede e podem possuir alto custo computacional local¹⁶ (Zarpelão et al., 2017).

Definição 3.12 (Arquitetura Centralizada). *O IDS é instalado na borda da rede ou em dispositivos dedicados (Farooqi e Khan, 2009).*

Em uma abordagem distribuída, os IDSs são instalados em todos os dispositivos que compõem o sistema (Farooqi e Khan, 2009; Zarpelão et al., 2017). As trilhas de auditoria podem ser coletadas no próprio dispositivo ou este pode atuar como um vigilante, monitorando o comportamento de seus vizinhos (Farooqi e Khan, 2009; Santos et al., 2018; Zarpelão et al., 2017). A classificação dos eventos pode ser feita individual ou cooperativamente, combinando decisões de vários vigilantes (Farooqi e Khan, 2009). Sistemas de detecção de intrusão distribuídos podem ser eficientes para identificar eventos suspeitos nas diversas partes da rede, reduzem o tráfego de dados e maximizam a capacidade de processamento. No entanto, essa é uma estrutura mais complexa e sua administração pode ser desafiadora (Zarpelão et al., 2017).

Definição 3.13 (Arquitetura Distribuída). *O IDS é instalado em todos os nós da rede (Farooqi e Khan, 2009).*

Uma estratégia intermediária é alcançada utilizando IDSs híbridos, que dividem a rede em regiões, cada qual com um vigilante, também chamado monitor (Farooqi e Khan, 2009). O monitor é o único nó da região que possui um IDS, sendo ele responsável por monitorar o comportamento de todos os dispositivos que operam na sua região (Zarpelão et al., 2017). Esses sistemas podem tomar decisões sobre comportamentos suspeitos em sua região ou apontar uma ação suspeita para um nó central, responsável pela classificação final (Farooqi e Khan, 2009; Zarpelão et al., 2017). Essa abordagem é mais simples que a distribuída, ao mesmo tempo em que reduz o custo computacional global, o consumo de recursos, permite a criação de diferentes regras para cada região e pode monitorar regiões inalcançáveis para IDSs centralizados. A Tabela 3.2 resume as vantagens e as desvantagens das arquiteturas aqui apresentadas.

Definição 3.14 (Arquitetura Híbrida). *A rede é dividida em regiões, cada uma com um nó monitor. O nó monitor possui um IDS instalado e é responsável por monitorar a sua região (Farooqi e Khan, 2009).*

Sistemas de detecção de intrusão centralizados são eficientes para detectar ataques em IDSs de rede, analisando o tráfego que passa através do roteador (Santos et al., 2018; Zarpelão et al., 2017). Nestes cenários os sistemas são eficazes do ponto de vista energético, pois operam em dispositivos robustos e concentram a maior parte do tráfego (Farooqi e Khan, 2009). Em contraposição, os sistemas distribuídos consomem muita energia e possuem alto custo computacional global, dado que trabalham em todos os nós (Farooqi e Khan, 2009; Zarpelão et al., 2017). No entanto, por atuarem em toda rede, conseguem identificar eventos suspeitos que não chegam até a borda. Finalmente, os sistemas híbridos equilibram melhor a relação entre complexidade, custo computacional e consumo de energia (Farooqi e Khan, 2009; Santos et al., 2018). No entanto, os sistemas híbridos são vulneráveis a ataques que afetam o desempenho geral da rede, o que pode resultar em erros de interpretação e na queda do desempenho dos IDSs (Farooqi e Khan, 2009).

¹⁶O alto custo computacional local se refere ao custo na estação base. O custo computacional global se refere ao custo de todo sistema, englobando todos os dispositivos que o compõem.

Tabela 3.2: Vantagens e desvantagens das arquiteturas dos sistemas de detecção de anomalias

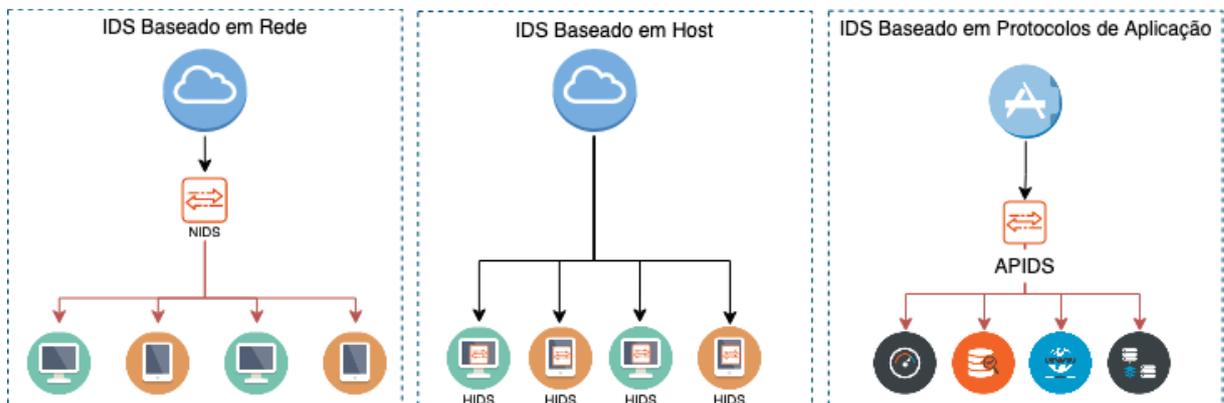
		Arquiteturas dos IDSs		
		Centralizada	Distribuída	Híbrida
Vantagens	<ul style="list-style-type: none"> • Simplicidade • Eficiência energética • Baixo custo computacional global • Eficiência na detecção de ataques que se propagam pela rede 	<ul style="list-style-type: none"> • Tráfego de rede reduzido • Capacidade de processamento maximizada • Menor custo computacional local • Eficiência na detecção ataques localizados 	<ul style="list-style-type: none"> • Equilíbrio entre complexidade, custo computacional e consumo de energia • Possibilidade de criação de regras locais 	
Desvantagens	<ul style="list-style-type: none"> • Incapacidade de detecção de ataques localizados • Baixo custo computacional local 	<ul style="list-style-type: none"> • Maior complexidade • Alto custo computacional global • Baixa eficiência energética 	<ul style="list-style-type: none"> • Vulnerabilidade a ataques que afetam o desempenho geral da rede 	

FONTE: O Autor (2020)

3.2.3 Tipos dos Sistemas de Detecção de Intrusão

Os detectores de intrusão podem ter vários objetivos, como identificar anomalias no tráfego de rede, na operação de serviços e aplicações ou comportamentos suspeitos de dispositivos, por exemplo. Os sistemas que monitoram tráfego de rede são chamados IDSs baseados em rede (NIDS, do inglês *Network-based IDS*) e inspecionam a transferência de dados em um ou mais segmentos de rede. Já os IDSs baseados em *host* (HIDS, do inglês *Host-based IDS*) são instalados em dispositivos individuais para monitorar um ou mais subsistemas. Ao contrário dos NIDSs, que apenas analisam o tráfego de rede, os HIDSs conseguem examinar também arquivos, processos, chamadas sistema, registros de aplicações e outros recursos (Santos et al., 2018; Zarpelão et al., 2017; Zuech et al., 2015). Os IDSs baseados em protocolos de aplicações (APIDS, do inglês *Application Protocol-based IDS*) monitoram os protocolos relacionados a uma aplicação específica filtrando eventos suspeitos (Nachan et al., 2021; Palekar, 2017). A Figura 3.5 apresenta a localização dos IDSs em um sistema computacional.

Figura 3.5: Tipos de Sistemas de Detecção de Intrusão



FONTE: O Autor (2020)

Os sistemas baseados em *rede* analisam os pacotes de dados em busca de indícios que apontem para a ocorrência de um ataque. Uma vez detectada uma atividade suspeita, o sistema pode disparar um alerta ou realizar uma ação que impeça seu prosseguimento (Iqbal et al., 2016; Zuech et al., 2015). Ao analisar o tráfego de dados, os NIDSs podem identificar e mitigar ataques que se propagam pela rede. Entretanto, esses sistemas não conseguem detectar aqueles que utilizam várias etapas, bem como inspecionar tráfego criptografado (Iqbal et al., 2016; Zuech et al., 2015).

Os sistemas baseados em *host* são instalados em dispositivos individuais, o que os habilita a identificar eventos que comprometam a segurança do dispositivo (Iqbal et al., 2016). Esses IDSs podem ser especialistas no monitoramento de aplicações ou protocolos específicos (Zuech et al., 2015). Eles operam monitorando um ou mais recursos e, ao identificar uma atividade suspeita, enviam alerta ao usuário ou administrador do dispositivo. Normalmente, os HIDSs não possuem recursos que os permitam mitigar ou impedir o prosseguimento do ataque.

Os IDSs baseados em *protocolos de aplicações* são sistemas que monitoram e analisam um ou múltiplos protocolos utilizados por uma aplicação, ou por um sistema computacional (SAKURAI e Kim, 2008). Eles são instalados dentro de um servidor ou em um grupo de servidores para monitorar e analisar os protocolos relacionados a uma aplicação específica (Nachan et al., 2021). Os APIDSs utilizam regras para verificar se um conjunto de eventos é compatível com o comportamento esperado (Palekar, 2017). Há, também, os IDSs baseados em protocolos (PIDS, do inglês *Protocol-based IDS*) que analisam os protocolos utilizados por um sistema. Os PIDSs são geralmente executados em servidores *web* para monitorar o protocolo HTTP. Devido à sua semelhança com os APIDSs, este trabalho considera os PIDSs no âmbito dos APIDSs. A Tabela 3.3 resume as características dos sistemas de detecção de intrusão apresentados nessa seção.

Tabela 3.3: Características dos tipos de sistemas de detecção de intrusão

Tipo IDS	Principais características
NIDS	Utilizados para inspecionar tráfego de rede
	Podem identificar e mitigar ataques que se propagam pela rede
	Não detectam ataques que utilizam múltiplas etapas
	Não inspecionam tráfego criptografado
HIDS	Utilizados para inspecionar eventos em dispositivos
	Podem inspecionar aplicações ou protocolos específicos
	São capazes de monitorar múltiplos recursos
	Normalmente não mitigam ou impedem o prosseguimento do ataque
APIDS	Utilizados para inspecionar o comportamento e os estados de protocolos
	Podem inspecionar múltiplos protocolos relacionados a uma aplicação específica
	São instalados entre processos ou um grupo de servidores
	Podem monitorar protocolos executados em dispositivos conectados

FONTE: O autor (2020)

Alguns sistemas mais complexos podem utilizar uma estratégia híbrida para detecção de intrusão. É o caso da nuvem, que possui uma estrutura mais complexa, exigindo recursos extras para permanecer segura. Neste ambiente, os NIDSs são bastante úteis para identificar uma série de ataques que podem comprometer a segurança de máquinas virtuais e *hypervisors*, porém são incapazes de detectar eventos que ocorrem dentro deles. No ambiente interno, podem ser utilizados HIDSs, de forma que o IDS da máquina virtual pode ser monitorado pelo usuário e o do *hypervisor* deve ser monitorado pelo administrador da nuvem (Iqbal et al., 2016). Aplicações críticas, como servidores *web* com banco de dados, podem utilizar um APIDS para monitorar como o protocolo SQL interage com o banco de dados.

3.2.4 Detecção Ativa e Passiva

Os detectores de intrusão também podem ser classificados de acordo com seu comportamento após a identificação de um evento suspeito: adotar medidas passivas ou ativas (Keegan et al., 2016). Os IDSs passivos se limitam a notificar uma autoridade apropriada, que deverá decidir sobre possíveis contramedidas. Já os ativos, podem controlar o sistema sob ataque, modificando seu estado para impedir ou mitigar os efeitos da intrusão, ou controlar o sistema de ataque na tentativa de impedir um incidente. Esta estratégia é muito difícil de implementar e pode incidir em quebras de disponibilidade ou ainda infringir legislações, ou regulações, por isso é pouco utilizada (Axelsson, 2015).

A adoção de medidas passivas pode ser interessante em alguns sistemas de detecção de anomalias, em cenários sujeitos a altos índices de falso positivo ou nos quais a adoção de contramedidas indevidas pode ser crítica. Na detecção de anomalias, pode não ser possível adotar uma contramedida efetiva, o que pode ocorrer em alguns sistemas que detectam falhas físicas em dispositivos, por exemplo. Em cenários sujeitos a muitos falsos positivos, respostas ativas podem gerar um impacto negativo ou comprometer a disponibilidade do sistema (Debar et al., 2000). Em sistemas críticos, a adoção de contramedidas pode exigir a aprovação de um especialista, de modo a evitar que as ações adotadas resultem em mais incidentes de segurança ou em prejuízos para uma terceira parte envolvida.

Em outros casos, especialmente naqueles sistemas que detectam ataques cibernéticos, é mais apropriado adotar uma reação ativa com o intuito de mitigar o ataque (Debar et al., 2000). Essas reações podem ser realizadas pelo próprio IDS ou por sistemas de apoio. Todavia, é preciso tomar cuidado para que as contramedidas executadas não interrompam ações devidas e não sejam exageradas (Halme e Bauer, 1995).

Os sistemas de detecção de intrusão podem ser elaborados de diversas formas e a definição sobre qual estrutura utilizar depende do contexto em que será aplicado. Quanto ao método de detecção, depende se o objetivo é analisar comportamentos ou ataques. No primeiro caso, métodos baseados em anomalias e especificação são mais indicados, no segundo, especificação e assinaturas são mais eficientes. Ambos têm suas vantagens e desvantagens, por isso métodos híbridos tendem a apresentar uma melhor relação entre custo e benefício.

A arquitetura distribuída é complexa, normalmente utilizada por HIDSs e indicada para monitorar eventos em dispositivos. Por outro lado, a arquitetura centralizada é mais simples, comumente empregada em NIDS e PIDS, onde são instalados em roteadores, servidores ou dispositivos dedicados conectados a um ponto central da rede. Em sistemas mais complexos, como é o caso da IoT, sistemas híbridos são mais adequados, pois permitem monitorar diferentes componentes, sem comprometer os recursos computacionais do sistema inteiro. Finalmente, utilizar respostas ativas pode ser interessante para evitar o prosseguimento de um ataque, mas é preciso adotar contramedidas que não comprometam ainda mais a segurança e o desempenho do sistema. Reações passivas são mais apropriadas para vários cenários, como detecção de anomalias em dispositivos físicos.

Apesar de existirem diversos detectores de intrusão e de anomalias desenvolvidos para IoT, os sistemas encontrados até o momento são voltados para a identificação e/ou mitigação de algumas categorias de ataques cibernéticos. Não foram encontrados trabalhos que busquem detectar falhas e erros em dispositivos. Essa é uma necessidade essencial das soluções voltadas para a agricultura, especialmente daquelas que dependem do bom funcionamento de dispositivos susceptíveis a falhas e/ou degradação, como é o caso de sensores. Assim como detectar ataques, identificar falhas e erros contribui substancialmente para a confiabilidade do sistema, requisito fundamental para a popularização de soluções criadas para a agricultura digital. Como os sistemas desenvolvidos até o momento não atendem a esses requisitos, torna-se necessária a

criação de novos recursos de segurança, que possibilitem a detecção de erros, falhas e ataques, maximizem a acuracidade e a confiabilidade dos dados e do sistema.

3.3 ARQUITETURAS PARA DETECTORES VOLTADOS À AGRICULTURA 4.0

Os recursos de segurança para agricultura inteligente podem incluir detectores de anomalias ou de intrusões. Enquanto os IDSs são artefatos desenvolvidos para detectar intrusões, os detectores de anomalias podem ir além e incluir entre suas funcionalidades a detecção de falhas e erros. As falhas e erros são comuns em dispositivos da percepção, especialmente aqueles expostos à condições climáticas ou ambientais extremas. Sensores instalados em ambientes muito úmidos ou expostos à temperaturas intensas, tendem a sofrer processos de saturação ou degradação gradual. Sensores expostos a intempéries podem sofrer danificação abrupta, devido à incidência de raios, por exemplo. Esses e outros eventos podem gerar alterações nos dados, que podem ser identificadas por um detector de anomalias. Entretanto, sensores e atuadores são dispositivos extremamente restritivos e incapazes de executar esses artefatos de segurança.

No contexto da agricultura inteligente, os detectores podem ser incluídos na borda ou na nuvem. A percepção é composta por dispositivos muito restritivos e incapazes de executar um monitor de anomalias/intrusões. Portanto, não é possível utilizar uma arquitetura totalmente distribuída. Uma arquitetura totalmente centralizada também não é viável, pois *i*) a nuvem não alcança a percepção, inviabilizando o monitoramento dos dispositivos dessa camada, e *ii*) o enlace entre a borda e a nuvem pode ser intermitente e lento, resultando em atrasos para uma resposta a ataques. Sendo assim, uma arquitetura híbrida é mais indicada.

A borda é composta por dispositivos computacionalmente limitados. Esses dispositivos geralmente possuem pouca memória e um processador capaz de executar instruções simples. Detectores que incluem algoritmos complexos não são adequados, pois exigem um processamento indisponível na borda, descartando a possibilidade de usar uma extensa quantidade de detectores considerados eficientes. Ademais, a percepção pode gerar grande quantidade de dados. Por isso, os sistemas desenvolvidos para agricultura precisam ser leves¹⁷ e rápidos, para não incorrer em latência excessiva, lentidão ou indisponibilidade do sistema.

É importante pontuar ainda que há poucas pesquisas voltadas para segurança em agricultura inteligente. Por isso, ainda não há uma taxonomia para ataques direcionados a este escopo. De forma geral, os ataques conhecidos para agricultura inteligente são aqueles que tem como alvo sistemas IoT.

A eficiência de detectores baseados exclusivamente em assinaturas é limitada, já que as intrusões em ambientes agrícolas são pouco conhecidas. Como esses sistemas possuem um escopo de atuação bem definido, métodos de detecção baseados em especificação apresentam uma boa relação e custo. Já a detecção baseada em anomalias pode ser eficiente para identificar falhas e erros e, em alguns casos, intrusões menos conhecidas.

Portanto, a agricultura inteligente carece de recursos de segurança capazes de identificar falhas, erros e ciberataques destinados ao contexto agrícola. Os detectores desenvolvidos com esse propósito podem utilizar uma arquitetura híbrida, com monitores espalhados na borda e na nuvem. Na borda, os artefatos precisam ser suficientemente leves para serem executados por dispositivos restritivos. A nuvem pode incluir algoritmos robustos para identificar ataques mais complexos. Os métodos de detecção (anomalias, especificação e assinaturas) podem ser combinados para reduzir a latência, o consumo de recursos computacionais e maximizar a eficiência. Detectores leves e eficientes reforçam a segurança dos sistemas agrícolas, potencializando sua adoção pela

¹⁷Que consomem pouca memória e processamento.

comunidade. A proposta deste trabalho considera todos esses quesitos para construir um detector de anomalias para a agricultura inteligente.

4 CEIFA: UM DETECTOR DE ANOMALIAS PARA SISTEMAS AGRÍCOLAS DIGITAIS

Os sistemas agrícolas são soluções que normalmente monitoram e/ou controlam atividades rurais, como processos de irrigação, adubação ou controle de estufas. Alguns são completamente automáticos, enquanto outros requerem interação humana para realizarem parte do processo. Em muitos casos, a decisão final é delegada a um profissional, que pode iniciar a irrigação pelo envio de um comando a partir de um dispositivo móvel ou fazê-lo manualmente. Esses sistemas são muito úteis e podem se tornar ferramentas importantes para o desenvolvimento da agricultura. Contudo, para serem incorporados aos processos agrícolas, eles precisam possuir um alto nível de confiabilidade.

A confiabilidade está atrelada ao fornecimento de informações corretas e completas sobre o estado atual do ambiente e a precisão das tomadas de decisões. Isso depende da qualidade dos dados recebidos, bem como da segurança do sistema. A qualidade dos dados pode ser prejudicada, por exemplo, por sensores saturados, descalibrados ou danificados, injeção de dados maliciosos e corrupção de pacotes durante sua transmissão. Ataques cibernéticos podem comprometer a confiabilidade, a disponibilidade e a integridade, afetando o sistema de diversas formas. A quebra de qualquer requisito de segurança pode culminar na desconfiança com relação ao sistema e sua rejeição por parte dos usuários. Sendo assim, a confiabilidade é fundamental para que a agricultura digital seja amplamente utilizada.

4.1 FALHAS QUE AFETAM DISPOSITIVOS DE COLETA DE DADOS

A análise dos dados processados pela agricultura inteligente pode dar bons indicadores sobre sua confiabilidade. A agricultura digital possui diversas fontes de dados, sendo os sensores a primeira delas. Os sensores são responsáveis pela coleta dos dados. Eles podem ser definidos como um “dispositivo, módulo ou subsistema que interage com o ambiente possibilitando a medição de algo variável através de respostas a estímulos físicos” (Aguirre, 2013).

Os estímulos físicos possuem grandezas irreconhecíveis pelos dispositivos digitais. Por isso, elas devem ser transformadas em medidas que possam ser processadas pelos sistemas. Os sensores transformam grandezas físicas, como temperatura, umidade e velocidade do vento, em sinais elétricos (Aguirre, 2013), mas os sistemas agrícolas manipulam valores digitais. Assim, o sensor precisa ser conectado a um equipamento capaz de transformar os estímulos elétricos em valores digitais.

Diversos sistemas são compostos por conjuntos de equipamentos que agregam um ou mais sensores. Esse equipamento recebe o sinal elétrico do sensor, o transforma em um valor numérico, enviado para a próxima camada do sistema, geralmente a borda. O valor numérico enviado pelo transdutor é chamado *valor*, aqui representado por v .

Definição 4.1 (Valor). *É a representação numérica de um fenômeno físico sentido por um sensor.*

Aguirre (2013) afirma que a saída de um sensor é um valor dentro de uma escala. O autor afirma ser preciso que “haja uma relação entre o sinal de entrada e sua representação”, que esta relação seja conhecida e não varie com o tempo (Aguirre, 2013, p.7). Como a maioria dos sensores utilizados em agricultura medem estímulos físicos ocorrendo em limites predefinidos, então os valores gerados por eles também podem variar dentro de um limite preestabelecido. Os valores que pertencem a este limite são considerados *válidos*.

Definição 4.2 (Valores Válidos). *Um sensor \mathcal{S} possui um conjunto de valores válidos \mathcal{V} , que variam entre um valor mínimo (v_{min}) e um valor máximo (v_{max}). Ou seja:*

$$\forall v \in \mathcal{V}, v_{min} \leq v \leq v_{max}.$$

Desta forma, qualquer valor $v \notin \mathcal{V}$ não representa um estímulo físico, portanto não é válido. Os valores enviados pelo sensor e que não pertencem ao conjunto \mathcal{V} são chamados Valores Fora de Escala.

Definição 4.3 (Valor Fora de Escala). *É todo valor enviado pelo transdutor que não pertence ao conjunto de valores válidos. Ou seja:*

$$\forall v, v \text{ é um Valor Fora de Escala} \iff v \notin \mathcal{V}.$$

Ainda que $v \in \mathcal{V}$, v pode não representar precisamente um estímulo físico devido à ocorrência de uma falha. Bezerra (2015) define falha como “um mau funcionamento de qualquer componente de um sistema, causando desde a sua perda de desempenho até a total parada da execução de suas funções”. As falhas podem ocorrer abrupta ou gradualmente e podem se apresentar de diversas formas, como, por exemplo: valor zero, deriva de valor de escala e valor de fundo de escala (Medeiros, 2009).

Definição 4.4 (Valor Zero). *Ocorre quando o sensor informa o valor constante zero (Medeiros, 2009).*

Valor Zero pode ser alcançado abruptamente, devido a algum evento que danifique o sensor e o impeça de perceber as alterações físicas para as quais foi construído. Também pode ocorrer gradualmente, sendo o ápice de um processo de degradação do sensor. Neste caso, antes de atingir o Valor Zero, o dispositivo passa pela Deriva de Valor de Escala, de forma que a escala se aproxima gradualmente do ponto zero.

Definição 4.5 (Deriva de Valor de Escala). *Ocorre quando os valores pertencentes à escala alteram ao longo do tempo (Medeiros, 2009).*

Outro processo comum em algumas categorias de sensores é o Valor de Fundo de Escala. Isso acontece quando sensores registram valores muito próximos ou iguais a v_{min} , ou v_{max} . Valores de Fundo de Escala são comuns em sensores de umidade relativa do ar que, quando expostos a altos índices de umidade, registram o valor máximo da escala. Em alguns casos o sensor continua registrando o Valor de Fundo de Escala mesmo após a umidade baixar.

Definição 4.6 (Valor de Fundo de Escala). *Ocorre quando o sensor informa um valor próximo aos limites da escala (Medeiros, 2009).*

Apesar de *valor fora de escala*, *valor zero*, *deriva de valor de escala* e *valor fundo de escala* serem falhas mapeadas em sensores industriais, elas também são identificadas em sensores utilizados pela agricultura inteligente. Neste caso, elas estão associadas a problemas como saturação de sensores e processos de degradação e obstrução, por exemplo.

4.2 MODELO DE ANOMALIAS

Este trabalho abarca um conjunto de anomalias que ocorrem na camada de percepção e afetam os dados enviados pelos transdutores para as camadas superiores. O escopo foi delimitado àquelas que afetam instrumentos¹⁸ de coleta ou medição de dados ambientais e podem ser detectadas a partir da análise dos dados coletados pelos sensores e enviados para a borda ou para a nuvem. O conjunto de anomalias aqui tratado envolve falhas aleatórias, saturação, degradação, obstrução, sensor danificado, ruídos e injeção de dados a partir da percepção.

4.2.1 Falha aleatória de sensor

Em geral, os sensores são acoplados a transdutores que solicitam informações aos seus componentes periodicamente. Ocasionalmente, pode ocorrer alguma falha que impeça o dispositivo de medir os estímulos do ambiente. Caso isso ocorra, o transdutor não recebe a informação solicitada, e envia um código predefinido. Essa anomalia não é disparada por um gatilho específico e ocorre esporadicamente, por isso é chamada falha aleatória. Sua ocorrência recorrente pode indicar o início de um processo de degradação. Isto posto, pode-se afirmar que:

Definição 4.7 (Falha aleatória). *Resulta da incapacidade do sensor de captar ou retornar dados sobre os estímulos do ambiente.*

Dado que esta anomalia não está diretamente relacionada a outras, ela é do tipo pontual e pode ser detectada a partir de um conjunto de especificações. Portanto, a categoria da anomalia e o modo de detecção são definidos da seguinte forma:

Tipo de anomalia: Pontual. Ocorre de forma aleatória e não intencional, sem um gatilho externo conhecido.

Modo de detecção: Especificação. Uma base de especificações contém as informações sobre a escala de valores aceita pelo sensor. Valores fora da escala são considerados anomalias.

Esta falha pode ocorrer nos mais diversos sensores, de diferentes formas. Entretanto, são considerados neste trabalho os equipamentos cujos valores variam em uma escala e que utilizam códigos fora da escala para indicar a ocorrência de falhas aleatórias.

Escopo: Sensores cujos valores válidos são conhecidos e variam dentro de uma escala predefinida.

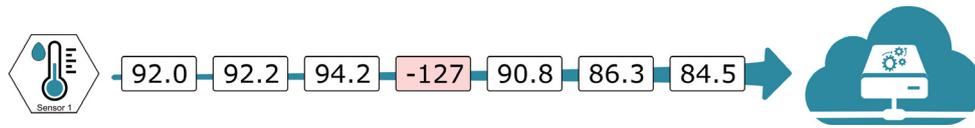
Exemplo: Considere o conjunto de dados da Figura 4.1, referentes às medições de umidade relativa do ar. O valor destacado em vermelho indica a ocorrência de falha e os marcados em preto apontam a umidade do ar, medida pelo sensor.

4.2.2 Sensor saturado

Equipamentos expostos a determinados estímulos ou condições ambientais podem ser levados a um estado extremo, ou de pico, a ponto de um novo aumento dos estímulos ou variações nas condições deixar de ser registrado. Este estado, chamado saturação, ocorre quando o dispositivo informa um Valor de Fundo de Escala ininterruptamente. Na maioria das vezes o instrumento sai do estado de saturação automaticamente. Em outros, ele permanece nesse estado por um longo período, mesmo que as condições ambientais tenham sofrido modificações.

¹⁸Nesta seção utilizamos os termos sensor, dispositivo, equipamento, instrumento e componente intercambiavelmente.

Figura 4.1: Exemplo de falha aleatória de sensor



FONTE: O Autor (2021)

Definição 4.8 (Sensor saturado). *Verifica-se quando o componente alcança um estado extremo ou de pico, do qual se torna incapaz de retornar.*

Visto que o Valor de Fundo de Escala pertence ao conjunto de valores válidos, é preciso analisar o contexto para identificar a ocorrência de saturação. Este estado é alcançado gradativamente, desencadeado por condições específicas. As condições que levam ao registro de valores extremos se modificam ao longo do tempo e espera-se que os sensores registrem essas variações. O registro ininterrupto desse valor por um longo tempo pode indicar saturação. Registros de sensores que medem estímulos similares ou correlacionados também podem ser utilizados para identificar essa anomalia.

Tipo de anomalia: Contextual. Ocorre gradativamente e de forma não intencional, tendo como gatilho condições ambientais específicas.

Modo de detecção: Anomalias. Pode-se detectar que um sensor está saturado quando ele passar um longo período informando um Valor de Fundo de Escala, sem variações. Divergências entre os valores registrados por dispositivos que medem os mesmos parâmetros também são relacionadas a esta anomalia.

Sensores de umidade relativa do ar podem entrar em estado de saturação. A umidade, assim como outros parâmetros agrícolas, variam ao longo de um período e não é admissível que o sensor permaneça muito tempo registrando um único valor. Para não restringir o escopo aos sensores de umidade relativa do ar, são considerados todos os dispositivos de coleta de dados que apresentam comportamento similar.

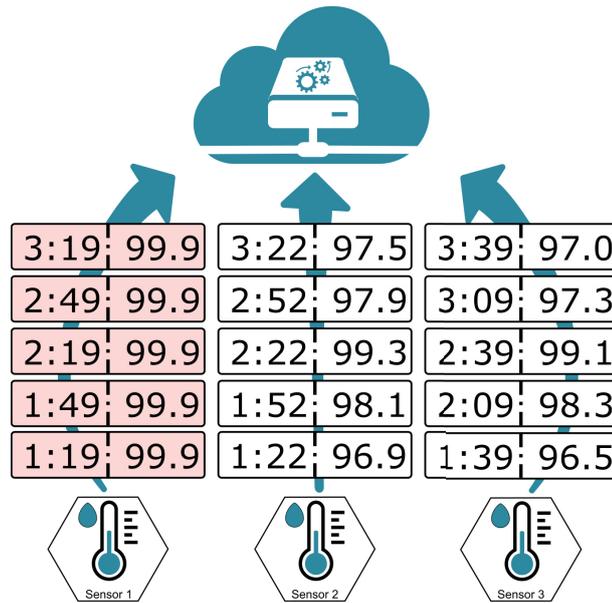
Escopo: Instrumentos que monitoram parâmetros que sofrem variações ao longo de um período e que, quando saturados, informam apenas um valor limite na escala.

Exemplo: Sensores de umidade relativa do ar podem sofrer saturação quando expostos a altos índices de umidade. Quando isso ocorre ele permanece um longo período estagnado, informando a mesma umidade, geralmente o valor máximo. No entanto, a umidade relativa do ar sofre pequenas variações durante o dia, influenciada por fatores climáticos. Por esta razão, não é aceitável que um sensor de umidade relativa do ar informe o mesmo valor por um longo tempo. Na Figura 4.2 o sensor 1 está saturado, enquanto os demais sensores estão funcionando normalmente. O primeiro parâmetro indica o horário em que a leitura foi realizada e o segundo, a umidade registrada.

4.2.3 Sensor degradado ou obstruído

Dispositivos expostos a determinadas condições físicas ou químicas podem sofrer obstrução, desgaste ou deterioração. Em decorrência do processo de degradação ou obstrução,

Figura 4.2: Exemplo de sensor saturado



FONTE: O Autor (2021)

eles tornam-se incapazes de coletar dados precisos sobre o ambiente. O processo de degradação faz com que o sensor registre valores compatíveis com Deriva de Valor de Escala. No ápice da deterioração, ele registra o Valor Zero constantemente.

Definição 4.9 (Sensor degradado). *É o efeito de um processo natural de desgaste ou deterioração, resultante da exposição (constante ou periódica) a determinadas condições.*

Definição 4.10 (Sensor obstruído). *É um bloqueio ou entupimento de alguns componentes que impedem ou dificultam o funcionamento do dispositivo.*

A degradação e a obstrução podem ser identificadas analisando o comportamento do próprio sensor e de outros dispositivos similares ou correlacionados. A análise permite verificar se as medições são compatíveis com o comportamento prévio do próprio equipamento e com os demais dispositivos.

Tipo de anomalia: Coletiva. Os dados informados pelo sensor degradado ou obstruído pertence à escala e, em geral, sofre pequenas variações. Entretanto, os valores coletados por esses equipamentos são incompatíveis com os dados apresentados por outros dispositivos no mesmo período ou com seus próprios dados históricos em período similar, ou compatível.

Modo de detecção: Anomalias. Para detectar essa anomalia é preciso comparar os dados do sensor com os dados de outros dispositivos ou com seus próprios valores históricos, de modo a identificar variações na escala ou registro indevido de valor zero.

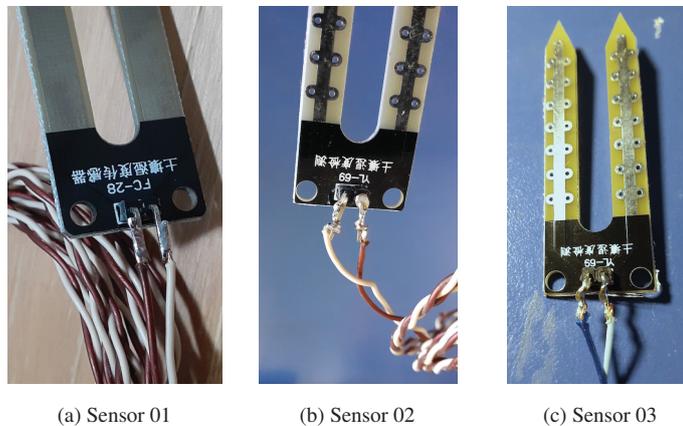
Os dispositivos mais propensos a sofrer degradação são aqueles cujos componentes entram em contato com elementos que podem causar reações químicas ou físicas. As hastes de sensores de umidade do solo, por exemplo, podem oxidar quando em contato com o ar, água e elementos presentes no solo. Já a obstrução pode ocorrer quando os componentes do sensor são fisicamente impedidos de operar corretamente.

Escopo: Sensores capazes de coletar parâmetros ambientais e aptos a sofrerem desgaste ou obstrução pela exposição a condições climáticas e ambientais.

Exemplo (1): Sensores de velocidade do vento instalados em locais empoeirados podem sofrer acúmulo de partículas que impedem suas hastes de girar. O acúmulo de poeira obstrui gradativamente o equipamento e os índices de velocidade do vento são inferiores aos valores reais. Em casos extremos, é informado o valor mínimo para a velocidade do vento.

Exemplo (2): Sensores de umidade do solo podem sofrer oxidação devido ao contato com elementos químicos presentes no solo. Equipamentos oxidados podem apresentar dados compatíveis com saturação. Outros registram valores discrepantes aos de dispositivos não degradados. Quando oxidados, eles são incapazes de medir a umidade do solo precisamente e informam valores incorretos sobre o estado do ambiente. A degradação ocorre lenta e gradativamente, dificultando sua detecção. Os sensores¹⁹ mostrados nas Figuras 4.3(a), 4.3(b) e 4.3(c) foram instalados em um ambiente de testes em maio de 2020 e permaneceram expostos às condições climáticas e ambientais por sete meses. Eles foram recolhidos em dezembro de 2020, pois alguns apresentaram falhas de leitura. O sensor mostrado na Figura 4.4(a) registrou valores discrepantes aos dos sensores não degradados. No entanto, foi possível identificar variações nos dados coletados por ele. Já o sensor mostrado na Figura 4.4(c), que teve o cabo que o conecta ao transdutor rompido, registrou o valor mínimo de umidade constantemente. O mesmo comportamento foi observado no dispositivo da Figura 4.4(b), cujo cabo não está roto.

Figura 4.3: Sensores de umidade do solo prontos para instalação



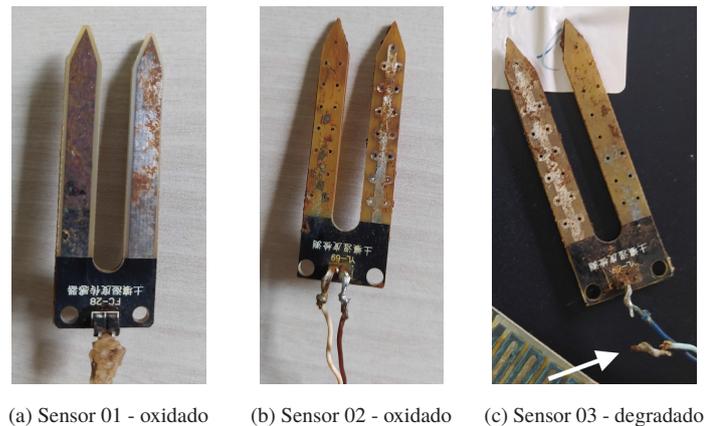
FONTE: O Autor (2021)

4.2.4 Sensor Danificado

Equipamentos utilizados em ambientes agrícolas estão normalmente expostos a eventos ou agentes que podem danificá-los ou desviá-los do seu funcionamento normal. Quando isso ocorre, o dispositivo registra valores incorretos, podendo, inclusive, parar de funcionar. Nesses casos pode ocorrer Deriva de Valor de Escala ou registro constante de Valor Zero.

¹⁹Esses sensores foram escolhidos por entrarem em processo de degradação rapidamente, o que permitiu a realização desta pesquisa. Tais dispositivos não são profissionais e não são utilizados pela agricultura inteligente. Entretanto, o uso de sensores profissionais inviabilizaria a realização desta pesquisa, por demorarem muito para entrar em processo de degradação.

Figura 4.4: Sensores de umidade do solo degradados



FONTE: O Autor (2021)

Definição 4.11 (Sensor Danificado). *Refere-se a sensores que sofreram dano ou estrago devido à ação de um agente externo (eventos naturais, animais, pessoas, etc.).*

A detecção de dano ou avaria é feita a partir da verificação do comportamento do sensor. Normalmente é esperado que o dispositivo registre variações no comportamento do ambiente. Todavia, quando danificado, ele deixa de registrar as variações. Alguns comportamentos observados em dispositivos danificados são: *i*) o registro recorrente de valores muito próximo aos Valores de Fundo de Escala; *ii*) registro ininterrupto de Valor Zero; *iii*) divergências significativas entre as leituras de sensores que medem os mesmos parâmetros.

Tipo de anomalia: Coletiva. Os dados informados pelo instrumento geralmente estão na escala e, em geral, sofrem pequenas variações. Contudo, quando ocorrem choques, os sensores são manuseados com imperícia ou são danificados propositadamente, registram dados incorretos.

Modo de detecção: Anomalias. Para detectar essa anomalia é preciso comparar os dados recentes do sensor com os dados de outros dispositivos ou com seus próprios registros históricos.

Diferentes dispositivos podem apresentar comportamento distintos quando danificados. O escopo deste trabalho inclui apenas os equipamentos que registram leituras distintas a de outros dispositivos similares²⁰, Valor de Fundo de Escala ou Valor Zero.

Escopo: Danificação ou estrago, parcial ou permanente, que leve o dispositivo a registrar dados distintos aos de outros sensores similares, Valor de Fundo de Escala ou Valor Zero.

Exemplo (1): Sensores de velocidade do vento instalados em locais empoeirados necessitam de manutenção periódica para que as partículas de pó sejam removidas. Essa atividade é geralmente realizada por um profissional, que deve retirar o componente e limpá-lo em água corrente. Se o profissional não realizar a manutenção com a devida perícia, pode danificá-lo.

Exemplo (2): Equipamentos instalados em ambientes externos estão expostos a choques com animais, pessoas ou equipamentos. Uma pessoa ou animal que colida com o sensor pode causar pequenos estragos, desviando o dispositivo do seu funcionamento normal. Caso o choque ocorra com um trator, é provável que o equipamento seja completamente danificado.

²⁰São considerados similares os dispositivos que medem os mesmos parâmetros.

Exemplo (3): Dispositivos instalados em ambientes externos podem ser facilmente acessados por pessoas que transitam no local. Eventualmente, uma pessoa mal-intencionada pode manusear o equipamento e corrompê-lo.

4.2.5 Ruídos

Transdutores podem sofrer interferências capazes de causar corrupção ou alteração nos dados. Quando tais interferências acontecem, os valores recebidos pela nuvem diferem daqueles enviados pelo transdutor. Em alguns casos, os dados podem ser modificados para um valor fora da escala. Em outros, os valores podem estar na escala, mas se distanciar do registrado pelo sensor. Existem várias fontes de ruídos, mas no contexto deste trabalho são considerados os:

Definição 4.12 (Ruídos). *Distúrbios eletromagnéticos que podem alcançar a rede de comunicação modificando os dados que estão sendo transmitidos e interferir no processo de medição do próprio sensor.*

Os ruídos podem ocorrer de forma não intencional ou serem injetados maliciosamente. Contudo, aqui são considerados apenas aqueles originados involuntária e aleatoriamente. Sua detecção é feita comparando-se as leituras com os registros recentes enviados pelo sensor. Desse modo, o tipo de anomalia e o modo de detecção são definidos como segue:

Tipo de anomalia: Pontual. Ocorre de forma aleatória, sem um gatilho predefinido.

Modo de detecção: Anomalias. Pode-se detectar dados corrompidos ao compará-los com um conjunto de dados recentes enviados pelo dispositivo.

Ademais, os ruídos inclusos no escopo alteram significativa e abruptamente um pequeno conjunto de dados, permitindo sua detecção pela análise dos registros recentes. Assim, o escopo fica restrito aos:

Escopo: Eventos esporádicos e temporários, que causam alterações abruptas em um pequeno conjunto de dados relacionados a parâmetros que normalmente não sofrem variações abruptas.

Exemplo: Alguns sistemas agrícolas podem utilizar redes sem fio para comunicação entre o sensor e a nuvem. Quando essa rede sofre uma interferência durante o processo de transmissão, os dados podem chegar à borda alterados ou completamente corrompidos.

4.2.6 Injeção de Dados Falsos

Um oponente malicioso²¹ pode injetar dados no sistema, levando-o para um estado de imprecisão. Os valores injetados comprometem a análise dos dados e impactam nos processos de tomada de decisão, sejam eles automáticos ou manuais. Hassija et al. (2019) e Lin et al. (2017) definem assim a injeção de dados falsos:

Definição 4.13 (Injeção de Dados Falsos). *São dados falsos enviados por um agente malicioso para um gateway.*

Esta anomalia é de difícil detecção, visto que os dados maliciosos não podem ser relacionados às falhas descritas na seção 4.1. Para ser possível identificar dados maliciosos, este trabalho compara as leituras de um sensor com os registros de outros sensores do mesmo tipo. Mas, para obter êxito, é imprescindível que a maioria dos dispositivos registre valores corretos. Isto posto, ficam assim definidos o tipo de anomalia, modo de detecção e escopo:

²¹Um oponente malicioso pode ser uma pessoa ou sistema que tente enganar o sistema.

Tipo de anomalia: Coletiva. Os dados enviados pelo hospedeiro malicioso geralmente estão na escala. Entretanto, eles diferem daqueles enviados por outros dispositivos do mesmo tipo.

Modo de detecção: Anomalias. Pode-se detectar dados corrompidos ao compará-los com um conjunto de dados enviados por outros sensores similares.

Escopo (1): Dados injetados por menos de 50% dos dispositivos de um sistema, que visem manipular o seu estado. A injeção de dados falsos por mais de 50% dos dispositivos inviabiliza a detecção, visto que o detector considera o padrão falso como sendo correto.

Escopo (2): Sistemas que possuem múltiplos sensores do mesmo tipo, coletando dados em ambientes com condições climáticas e ambientais similares.

Exemplo: Um sensor de umidade do solo comprometido ou um nó falso pode enviar informações que indiquem alto índice de umidade do solo, quando este está com umidade baixa, visando evitar que o sistema inicie a irrigação.

4.3 VISÃO GERAL DO DETECTOR DE ANOMALIAS

Este trabalho modela um detector de anomalias multinível denominado CEIFA (do inglês *Cloud-Edge Identifier of Farming Anomalies*), que tem como alvo as leituras dos sensores utilizados em sistemas agrícolas digitais. O CEIFA analisa os valores recebidos em busca de erros, falhas e outras anomalias que possam afetar a qualidade dos dados. Entre as anomalias alvo do detector estão falhas aleatórias, saturação, degradações decorrentes de obstrução e de processos naturais como oxidação, danificações parciais e permanentes, ruídos que causem corrupção ou alteração nos valores, e injeção de dados falsos. O detector foi estruturado como um APIDS, em uma arquitetura híbrida, com monitores na borda e na nuvem. O método de detecção é híbrido, combinando detecção por anomalias e especificações. As trilhas de auditoria são coletadas na borda e na nuvem.

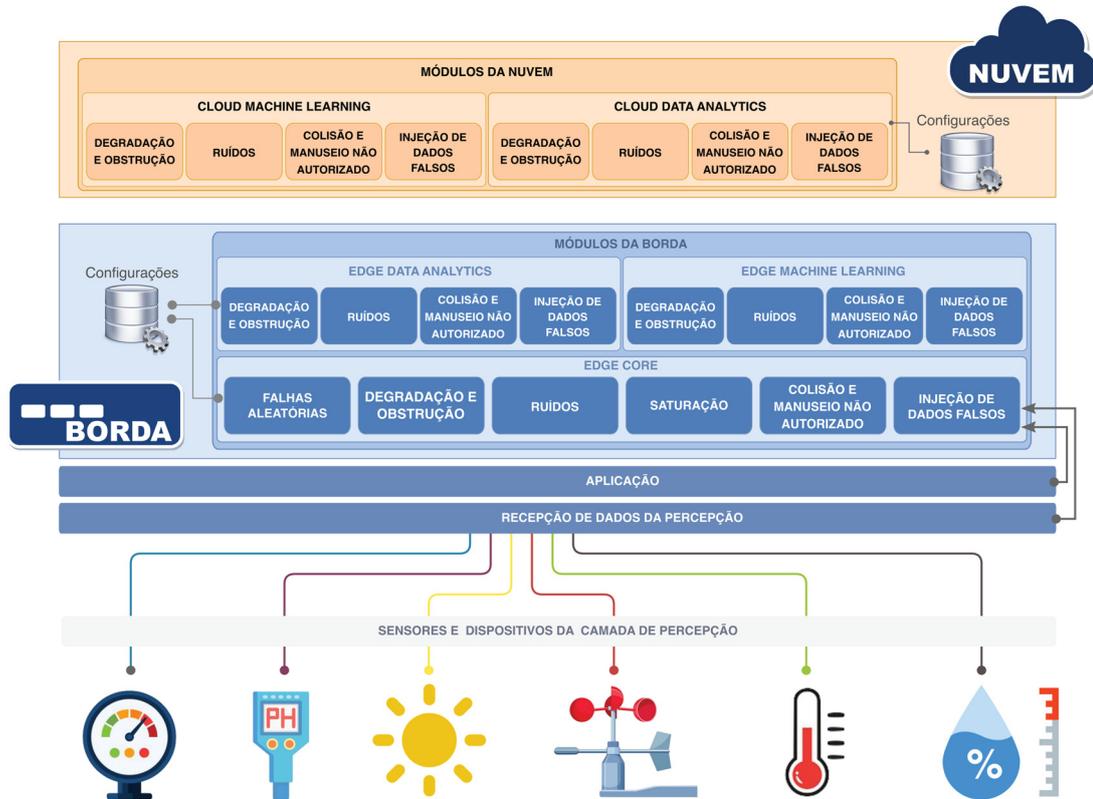
O sistema foi modelado para operar em vários níveis, conforme diagrama da Figura 4.5. A borda realiza a primeira análise dos dados. Ela pode ser composta por um ou mais módulos de classificação, configurados para trabalhar em conjunto ou separadamente. Esses módulos podem tomar sua classificação como definitiva ou solicitar verificação da nuvem. Caso o detector classifique os dados como *normais* na borda, o dado segue seu fluxo regular no sistema agrícola. Caso contrário, eles podem ser retidos para evitar sua utilização por processos de tomada de decisão. Os módulos da nuvem podem ser configurados para filtrar todos os registros do sistema agrícola ou para atender apenas as solicitações encaminhadas pelos módulos do detector.

O detector proposto neste trabalho é ajustável aos diferentes contextos e necessidades da agricultura inteligente. Cabe ao desenvolvedor identificar quais módulos são necessários e qual é a configuração mais adequada. As próximas seções detalham a solução proposta.

4.4 ARQUITETURA DO DETECTOR DE ANOMALIAS

O CEIFA foi estruturado para trabalhar em múltiplos níveis. Sua arquitetura permite que seja adaptado aos recursos computacionais disponíveis e às características do sistema agrícola que irá utilizá-lo. Ele recebe os registros diretamente do sensor (via módulo de rede) ou de uma aplicação. O detector é composto por módulos de classificação (os classificadores), responsáveis pela análise e classificação dos dados, e módulos de decisão (os decisores), que definem as ações a serem tomadas em decorrência da classificação. Ao concluir a classificação, os dados podem

Figura 4.5: Arquitetura do detector de anomalias proposto



FONTE: O Autor (2021)

ser retidos ou devolvidos/encaminhados para o sistema agrícola digital. O diagrama da Figura 4.6 mostra todos os módulos que constituem o detector bem como os fluxos de análise e decisão.

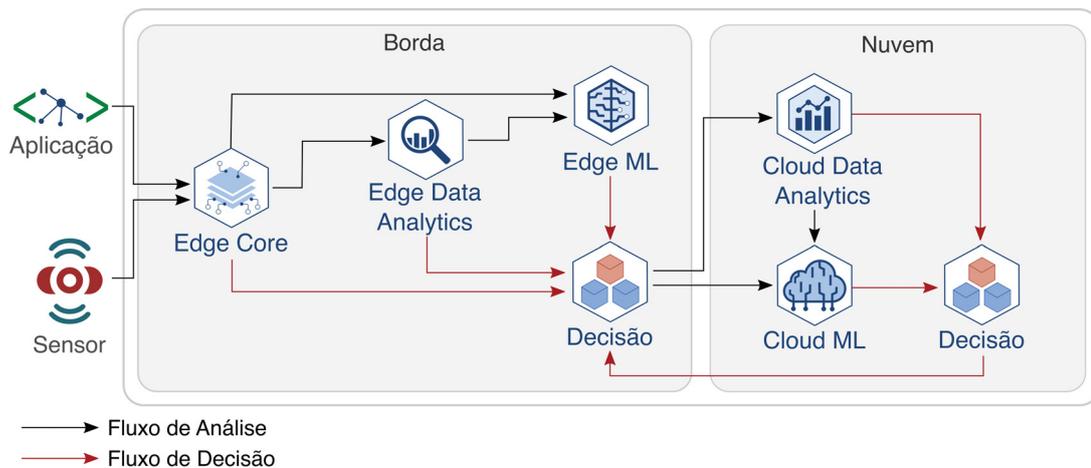
A borda pode possuir três classificadores: Edge Core (EC), Edge Data Analytics (EDA) e Edge ML (EML). O Edge Core é o primeiro nível de verificação, recebendo as leituras e classificando toda a massa de dados. Algumas classificações são tomadas como definitivas, enquanto outras podem ser retificadas por outros classificadores, estejam eles na borda ou na nuvem. A Seção 4.4.1 descreve o funcionamento deste classificador.

Dependendo das características dos dados e da classificação atribuída pelo EC, as leituras podem ser encaminhadas para o Edge Data Analytics. Este confronta-as com informações históricas e previsões, que podem estar armazenadas localmente, ser coletadas da nuvem ou da Internet. O EDA é eficiente para analisar dados que sofrem variações sazonais ou que tenham previsões disponíveis. Assim como o EC, as classificações atribuídas pelo EDA podem ser tomadas como definitivas ou encaminhadas para o EML ou para a nuvem. Mais informações sobre o funcionamento do Edge Data Analytics são encontradas na Seção 4.4.2.

Caso os recursos computacionais disponíveis permitam e as necessidades do sistema agrícola exijam, a borda pode agregar um terceiro classificador, o Edge ML. Este utiliza um algoritmo de aprendizagem de máquina para aprimorar a precisão da classificação, reduzir o tráfego na rede e o consumo de recursos computacionais na nuvem. A Seção 4.4.3 descreve este componente, apresenta suas vantagens e desvantagens, bem como os cenários em que é indicado.

O último componente da borda é o módulo de decisão, que determina uma ação conforme a classificação dos dados. Este módulo pode ser utilizado de diferentes formas. Sistemas agrícolas

Figura 4.6: Componentes do detector de anomalias



FONTE: O Autor (2021)

de monitoramento podem apenas marcar o conjunto de dados, indicando uma possível anomalia. Sistemas com mais recursos embarcados podem descartar valores classificados como anômalos, armazená-los em quarentena ou encaminhá-los para a nuvem. As diferentes ações e os cenários aplicáveis são descritos na Seção 4.4.5.

A nuvem pode possuir um ou dois classificadores: Cloud Data Analytics (CDA) e Cloud ML (CML). O primeiro trabalha similarmente ao EDA, confrontando os dados com informações históricas e/ou de previsões. Suas classificações podem ser encaminhadas para o módulo de decisão ou enviadas para o CML. O Cloud ML utiliza um ou mais algoritmos de aprendizagem de máquina para filtrar os dados e fornecer um veredicto final. As Seções 4.4.6 e 4.4.7 descrevem os módulos Cloud Data Analytics e Cloud ML, respectivamente. O módulo Decisões da nuvem, descrito na Seção 4.4.8, contém as ações a serem tomadas pela nuvem após a classificação dos dados. Entre as ações possíveis estão registrar a ocorrência de anomalias, enviar notificações aos administradores e intervir no sistema para isolar falhas.

4.4.1 *Edge Core*: usando análise estatística para detectar anomalias

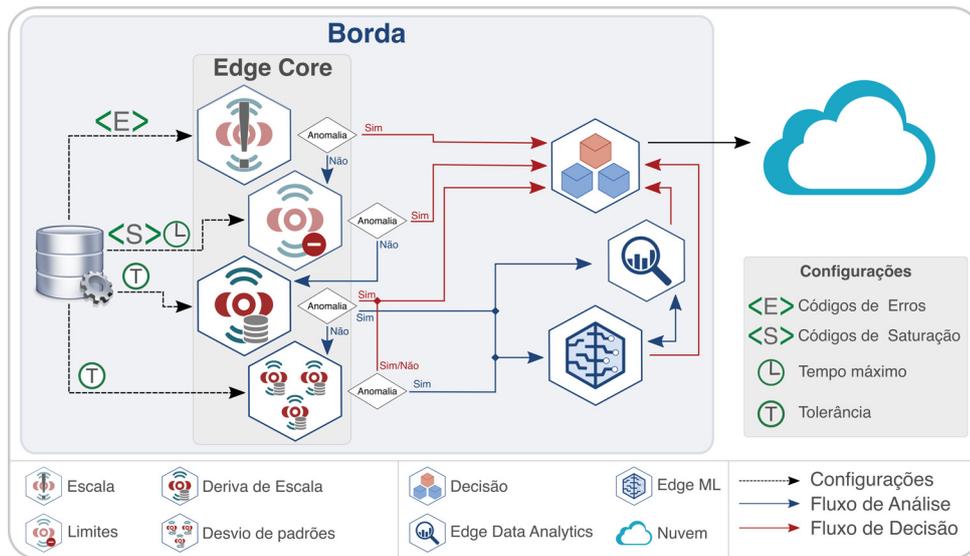
O Edge Core é o módulo que recebe os dados e faz a primeira análise. Ele é o primeiro elemento do CEIFA, responsável por verificar a validade dos dados processados pela borda. É um classificador baseado em especificações, que classifica os registros como *normais* ou *anômalos* por meio da análise estatística apresentada na Seção 4.4.1.1. O detector pode tomar a classificação atribuída pelo EC como definitiva ou encaminhar os valores para verificação de outros módulos. A ação adotada varia conforme a classificação final e com os componentes do detector²². A Figura 4.7 apresenta uma visão geral deste módulo. A Seção 4.4.1.2 descreve o fluxo de processamento do EC.

4.4.1.1 *Análise estatística no Edge Core*

A maior parte das anomalias descritas na Seção 4.2 podem ser identificadas por análise de tendências e probabilidade. Outras requerem o auxílio de estratégias adicionais, como análise

²²Um detector pode não possuir todos os componentes propostos e a decisão sobre quais incluir é tomada pelos desenvolvedores.

Figura 4.7: Arquitetura do Edge Core



FONTE: O Autor (2021)

de dados e aprendizagem de máquina, que serão descritas nas seções subsequentes. A análise estatística proposta utiliza operações matemáticas que podem ser executadas pelos dispositivos que possuem capacidade computacional (memória e processamento) suficiente para executá-las. O Capítulo 5 apresenta mais detalhes sobre os recursos utilizados pelo detector, permitindo identificar quais dispositivos possuem os pré-requisitos para executar essa estratégia.

Segundo a descrição contida na subseção 4.2.1, as falhas aleatórias resultam no envio de valores fora da escala ou valor *inválido*. Por isso, elas podem ser detectadas pela identificação de Valor Fora de Escala. Conforme a Definição 4.2, um valor válido é todo aquele que pertence a uma escala. Dado que os limites da escala sejam conhecidos, é possível verificar se o valor $v \in \mathcal{V}$ a partir da Equação 4.1, onde v_{min} e v_{max} representam, respectivamente, os limites inferior e superior da escala.

$$(v \geq v_{min}) \wedge (v < v_{max}) = normal \quad (4.1)$$

Algumas categorias de ruídos também podem ser detectadas pela Equação 4.1. Entre elas estão aquelas causadas por distúrbios eletromagnéticos que alteram os dados de modo a desviá-los da escala. Outras, afetam os dados de forma menos perceptível, mantendo-os entre os valores reconhecidos como válidos. Neste caso, pode-se utilizar probabilidade estatística para verificar qual a confiabilidade de um determinado valor. A Desigualdade de Hoeffding é bastante útil para esta finalidade.

Na teoria da probabilidade, a Desigualdade de Hoeffding fornece um limite superior para que a soma de variáveis aleatórias independentes se desviem de um valor esperado mais do que uma certa quantidade. Na detecção de anomalias, o valor esperado para v não pode se desviar mais que a tolerância (ϵ) da média de seus valores históricos (\overline{H}_s). Sendo assim, a tolerância (ϵ) é subtraída do valor absoluto da diferença entre v e \overline{H}_s . Para que o registro seja

considerado *normal*, o resultado deve ser inferior ao desvio padrão dos valores históricos do sensor²³ (σ_s), conforme mostra a Equação 4.2.

$$|v - \overline{H_s}| - \varepsilon < \sigma_s \quad (4.2)$$

A título de exemplo, considere um sensor de umidade relativa do ar que vem registrando valores entre 80 e 92. Esses valores estão registrados no histórico. Considere que a média dos valores históricos é 86.5, ou seja, $\overline{H_s} = 86.5$. O desvio padrão para os valores contidos no histórico é 3.89 ($\sigma_s = 3.89$). Assumindo que ε foi definido como 5 ($\varepsilon = 5$) e que o valor mais recente registrado pelo sensor é 79, pode-se afirmar que este é “normal”, pois está entre os valores esperados para o sensor, como mostra a Equação 4.3:

$$\begin{aligned} |79 - 86.5| - 5 &< 3.89 \\ 7.5 - 5 &< 3.89 \\ 2.5 &< 3.89 \end{aligned} \quad (4.3)$$

A Equação 4.2 consegue identificar dados incompatíveis com os esperados, mas que permanecem na escala. Além dos ruídos, esses dados incompatíveis também podem ter sua origem em anomalias dos sensores ou ataques externos. A análise sobre essas anomalias ou ataques que possam ser identificados por esta equação extrapola o escopo desta pesquisa.

Conforme discutido na subseção 4.2.2, um sensor saturado é caracterizado pelo registro recorrente de Valor Fundo de Escala. Existe uma quantidade considerada “aceitável” para se registrar esses valores ininterruptamente. Um registro excessivo de Valor Fundo de Escala pode sinalizar um estado anômalo, ligado à saturação. A Equação 4.4 pode ser utilizada para identificar esta anomalia. Ela considera as informações sobre os limites inferior e superior da escala, a média do histórico de leituras recentes do sensor ($\overline{H_s}$) e sua última leitura (v). A quantidade de valores armazenados no histórico deve ser equivalente à quantidade aceitável de registros para valor fundo de escala.

$$(v = v_{min} \wedge \overline{H_s} = v_{min}) \vee (v = v_{max} \wedge \overline{H_s} = v_{max}) = anomalia \quad (4.4)$$

Por exemplo, considere um sensor de umidade relativa do ar que registra valores entre 0 e 99.9. Um especialista definiu que se esse sensor registrar mais que cem vezes o valor 99.9, então ele estará saturado. Neste caso, o histórico deve armazenar as cem leituras mais recentes. Considerando que v_{max} é igual a 99.9 e o último (mais recente) valor recebido do sensor é 99.9, ou seja, $v = v_{max}$, o sensor será considerado saturado se todos os valores contidos no histórico forem 99.9. Enquanto houver um único valor diferente deste, o sensor não será considerado saturado, pois a média dos valores históricos ($\overline{H_s}$) diferirá de v_{max} .

Estados de danificação ou avaria também podem levar ao registro de Valor Fora da Escala, Valor Fundo de Escala ou valores muito próximos. Esse estado pode ser alcançado abrupta ou gradativamente. Quando o sensor foi completamente danificado, ele se torna incapaz de realizar a medição dos seus parâmetros e, em casos extremos, ele se torna inoperante. No primeiro contexto, é comum ocorrer o registro recorrente de Valor Zero, o que pode ser detectado pela Equação 4.4. No último, o transdutor não recebe uma resposta do sensor e pode enviar um código de erro. Então, a Equação 4.1 pode detectar essa anomalia. Entretanto, a danificação pode ser gradativa, gerando alterações na escala de valores registrados.

²³Os valores históricos recentes de todos os sensores devem estar armazenados em memória para que esta operação seja possível.

Esse comportamento é bastante similar aos associados a processos de degradação e obstrução. Em geral, a degradação e a obstrução são gradativas e causam alterações na escala. Essas alterações estão relacionadas à Deriva de Valor de Escala e podem ser detectadas pela Equação 4.5. É esperado que a média dos valores históricos do sensor (\overline{H}_s) somado ao desvio padrão dos valores históricos dos demais sensores (σ_t) seja muito próxima à média dos valores históricos dos demais sensores (\overline{H}_t). Entretanto, quando os sensores entram em processo de degradação, obstrução ou danificação, eles registram valores inferiores aos dos demais. Assim, se a soma de \overline{H}_s e σ_t , mais um valor de tolerância ε for menor que \overline{H}_t , então está ocorrendo alteração na escala do sensor.

$$\overline{H}_s + \sigma_t + \varepsilon < \overline{H}_t \quad (4.5)$$

Para exemplificar, tome um conjunto de 5 sensores de umidade do solo cuja escala inicial varia entre 0 e 100. Todos os sensores registram valores 0 quando o solo está muito seco e 100 quando está encharcado. Como cada sensor está instalado em um local geográfico diferente, porém próximo, é natural que seus registros diverjam. Considere que o sensor 1 entra em processo de degradação e, gradualmente, seu limite superior é reduzido. As médias dos últimos registros de cada sensor são as seguintes: $s_1 = 48.9$, $s_2 = 59.8$, $s_3 = 60.8$, $s_4 = 59.6$ e $s_5 = 59.8$. Para verificar o estado do sensor 1, tome sua média $s_1 = 48.9$, portanto $\overline{H}_s = 48.9$. A média dos demais sensores é 60, resultando em $\overline{H}_t = 60$. Calculando o desvio padrão das leituras dos sensores 2, 3, 4 e 5 chega-se ao valor 6.1 ($\sigma_t = 6.1$). Assumindo uma tolerância igual a 3 ($\varepsilon = 3$) e tomando os cálculos apresentados na Equação 4.6, pode-se concluir que o sensor 1 está, provavelmente, em um estado de degradação.

$$\begin{aligned} 48.9 + 6.1 + 3 &< 60.0 \\ 58.0 &< 60.0 \end{aligned} \quad (4.6)$$

Os cálculos estatísticos apresentados são bastante úteis para identificar alterações no estado físico dos dispositivos, desde que essas alterações afetem os dados. Para a Equação 4.6, também é importante que as anomalias afetem menos de 50% dos dispositivos. Caso contrário o comportamento anômalo poderá ser considerado normal. A grande vantagem da estatística é não depender de dados externos e utilizar poucos recursos computacionais, sendo estes ligados ao processamento e armazenamento dos dados históricos. A quantidade de dados armazenados é predefinida, não aumenta com o tempo e a análise acompanha o comportamento dos dados, ajustando-se às mudanças climáticas e ambientais que ocorrem gradualmente. No entanto, a estatística não consegue detectar ataques, como injeção de dados falsos e não detecta as anomalias apresentadas quando existirem muitos dispositivos comprometidos.

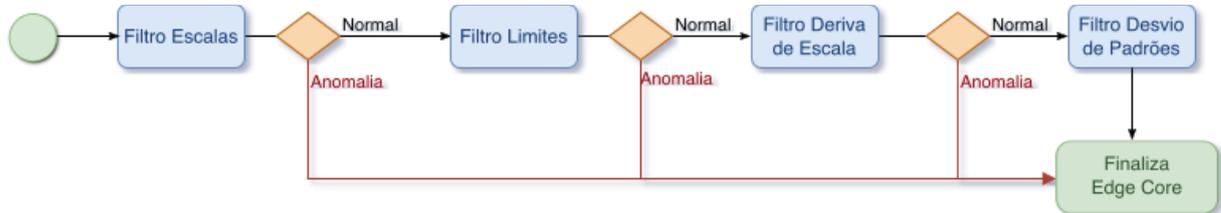
4.4.1.2 Fluxo de processamento do Edge Core

O fluxo de processamento segue o diagrama apresentado na Figura 4.8. Os dados recebidos pelo Edge Core são encaminhados para o filtro de *Escalas*, que utiliza a Equação 4.1 para identificar valores fora da escala. A classificação positiva²⁴ atribuída por este filtro pode ser considerada correta, sem a necessidade de seguir para outros filtros. Desta forma, evita-se o desperdício de recursos computacionais. Neste caso a classificação é informada para o módulo *Decisões* e o processo no EC é concluído. Caso a classificação atribuída seja negativa, o valor passa para o filtro de *Limites*. Este utiliza a Equação 4.4 para checar se é um valor fundo de escala. Caso as configurações estejam corretas, a classificação atribuída por esse filtro também

²⁴A classificação positiva indica que o valor está fora da escala do sensor, portanto, é uma falha/erro.

pode ser tomada como definitiva. Uma classificação positiva nesse módulo deve ser informada para a nuvem, dado que valor fundo de escala indica saturação ou degradação, o que pode ser corrigido a partir de intervenção humana. Em caso de classificação negativa, os dados passam para o próximo filtro.

Figura 4.8: Fluxo de Processamento do Edge Core



FONTE: O Autor (2021)

O filtro *Deriva de Escala* consegue identificar diferentes problemas expostos na Seção 4.4.1.1. Ele utiliza dados históricos do próprio sensor e realiza o processo apresentado na Equação 4.5. O filtro calcula a tendência e a dispersão dos dados²⁵ para identificar se eles combinam com os registros recentes dos sensores. Em caso de classificação negativa, o dado passa para o próximo nível.

O último filtro desse classificador é o *Desvio de Padrões*. Este filtro utiliza a Equação 4.2 e dados de equipamentos similares para verificar a ocorrência de anomalias. Da mesma forma que o filtro anterior, este utiliza valores históricos recentes para verificar se a medição registrada é confiável. A classificação atribuída pelos dois últimos filtros pode ser verificada em outros módulos, seja na borda ou na nuvem. Na borda podem ser utilizados os módulos EDA e/ou EML. Quais usar e em qual ordem é uma escolha do desenvolvedor e envolve questões como recursos disponíveis e características dos dados.

4.4.2 *Edge Data Analytics*: analisando dados para maximizar a eficácia

O Edge Data Analytics utiliza análise de dados para verificar a confiabilidade das leituras. Ele recebe os dados do Edge Core e os compara com dois possíveis filtros: *Previsões* e *Históricos*. O EDA pode ser utilizado sozinho ou em conjunto com os demais classificadores. Seus filtros são opcionais e podem ser utilizados segundo as necessidades do sistema. A Figura 4.9 apresenta uma visão geral deste módulo e suas interações com outros componentes do detector.

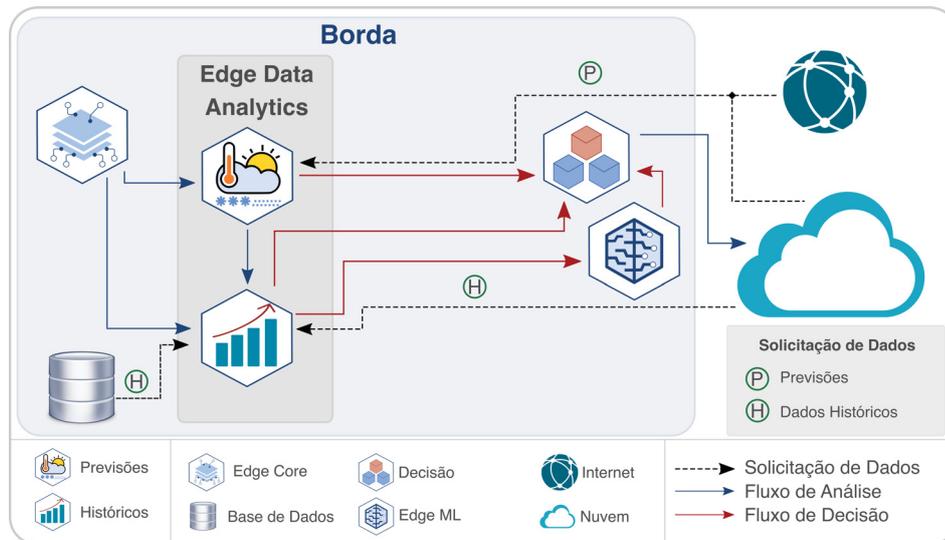
A análise de dados é utilizada para apoiar o processo de detecção de anomalias nos casos em que a análise estatística pura é incapaz de determinar a confiabilidade dos dados. Muitas vezes, mudanças climáticas e ambientais ocorrem de forma não regular. Nestes casos, a análise estatística pode falhar ao apontar para uma anomalia. Já a análise de dados históricos ou previsões futuras podem assinalar o comportamento válido, apesar de eventual. Este módulo usa dados históricos e previsões futuras para determinar um comportamento considerado “normal”.

4.4.2.1 *Análise de Dados*

A análise de dados é o processo de analisar informações com um propósito específico ou para responder a uma pergunta (Hoppen e Santos, 2018). É bastante usada em alguns setores

²⁵Aqui são utilizados a média e o desvio padrão.

Figura 4.9: Arquitetura do Edge Data Analytics



FONTE: O Autor (2021)

para identificar oportunidades, reduzir custos e agilizar processos. Na agricultura, ela pode ser utilizada para verificar a confiabilidade dos dados gerados pelo sistema. A comparação com registros históricos e/ou previsões futuras pode indicar se uma leitura é confiável ou potencialmente anômala.

A agricultura inteligente gera dados periodicamente e em quantidade relevante. Esses dados podem ser armazenados em uma base para comporem um conjunto de dados históricos. Os dados coletados e processados pelo sistema podem ser analisados e resultar em indicadores. Estes, são usados pelo detector de anomalias para identificar a confiabilidade das leituras. Em alguns casos, eles também podem indicar a ocorrência de eventos inesperados, resultando em uma anomalia.

Para exemplificar, considere um sistema agrícola que conte com processos de irrigação. Assuma que a irrigação ocorre sempre nos mesmos períodos do dia. A análise de dados pode apresentar os indicadores de umidade por período, permitindo evidenciar em quais períodos o solo está mais seco ou úmido. Assuma que a leitura de um sensor assinala uma umidade baixa e os indicadores informem que, no mesmo período, a umidade do solo é normalmente alta. A divergência entre as leituras do sensor e os indicadores do sistema pode apontar para a ocorrência de uma anomalia. Essa anomalia pode estar relacionada a uma falha no sensor ou à não realização da irrigação, eventos aqui considerados como inesperados.

Os indicadores podem ser obtidos a partir de dados do próprio sistema ou serem adquiridos de fontes externas. Eles podem estar relacionados a ocorrências passadas ou projeções futuras. Empresas especializadas podem fornecer indicadores específicos, como previsões do tempo e projeções de chuva. A obtenção das informações e a análise dos dados excedem o escopo deste trabalho, que está limitado ao seu uso.

Para ilustrar, considere o cenário em que se deseja analisar a confiabilidade de uma leitura de temperatura. O sistema dispõe de previsões meteorológicas que apontam para uma temperatura aproximada de 30°C em um determinado período. O sensor está registrando uma temperatura de 15°C. Apesar de ser esperada uma variação entre a leitura registrada e a prevista, não é admissível que a diferença seja tão grande. Assim, pode-se concluir que o registro é

anômalo, pois diferencia muito da realidade esperada. Por outro lado, é incomum o registro de temperaturas em torno de -4°C em Pinheiro Preto (SC), apenas seis dias após ocorrerem registros de temperaturas próximas dos 26°C . Mas, tanto a meteorologia previu a ocorrência dessa temperatura²⁶, quanto os dados históricos mostram a ocorrência eventual de temperaturas muito baixas²⁷ (para a região) ao longo das últimas décadas. Neste caso, as leituras dos sensores estão corretas e as possíveis anomalias estão relacionadas ao clima, fugindo da alçada do detector.

Visto que os indicadores estão disponíveis, o custo computacional engloba os requisitos para obtenção e troca segura das informações (infraestrutura de rede e segurança), seu armazenamento seguro (memória e processamento) e o processamento envolvido na checagem da confiabilidade das leituras. Contudo, não é possível precisar quantos e quais recursos serão utilizados, porque isso varia conforme as informações disponíveis, sua complexidade, custos computacional e financeiro para sua obtenção, transporte e armazenamento. Esses custos variam de acordo com os provedores de serviços e com os requisitos do sistema.

4.4.2.2 Fluxo de processamento do Edge Data Analytics

Os dados recebidos pelo EDA podem ser encaminhados para o filtro *Históricos* ou *Previsões*. Aqui não há um fluxo predefinido. A decisão sobre como os filtros estão organizados é tomada durante a construção do detector. Os desenvolvedores podem decidir, por exemplo, que as leituras relacionadas à temperatura passarão apenas pelo filtro *Previsões*, as relacionadas à umidade do solo, pelo filtro *Históricos*, e as ligadas à umidade relativa do ar e velocidade do vento passarão por ambos filtros. Como cada sistema agrícola possui características e conjuntos de dados específicos, cabe aos especialistas decidirem quais filtros usar e em qual ordem.

O filtro *Históricos* utiliza indicadores históricos do próprio sensor e/ou de outros dispositivos e os compara com os dados em análise. Por exemplo, é possível verificar se a temperatura registrada pelo sensor é compatível com temperaturas históricas no mesmo período do ano. Essas médias podem ser registradas pelo próprio sistema ou obtidas de fontes externas. Informações originadas no próprio sistema representam com maior fidelidade o comportamento do ambiente, desde que estejam corretas. Caso essas informações não estejam disponíveis, pode-se utilizar dados meteorológicos. Fornecedores de dados meteorológicos, como AccuWeather²⁸ e Climatempo²⁹ fornecem históricos que podem ser utilizados por esse filtro. Os históricos podem ser armazenados na borda, caso o sistema suporte, ou podem ser obtidos da nuvem periodicamente.

Comparar os registros dos sensores com dados históricos permite identificar comportamentos anormais relacionados a defeitos ou danos nos dispositivos. Esses registros são ainda mais relevantes quando se tratam de dados climáticos, que sofrem variações sazonais. Apesar dos registros históricos não refletirem com exatidão o comportamento atual, eles certamente fornecem bons parâmetros para o comportamento esperado. Por exemplo, considerando que as temperaturas do mês de janeiro em Curitiba (PR)³⁰ variam em torno de 18° e 25° , o registro de uma temperatura muito inferior ou superior pode indicar uma anomalia. Apesar de não ser

²⁶O comportamento pode ser verificado em: <https://www.accuweather.com/en/br/pinheiro-preto/41234/july-weather/41234?year=2021>

²⁷O registro de temperaturas negativas foi previsto e amplamente divulgado pela mídia: <https://g1.globo.com/natureza/noticia/2020/08/18/a-onda-historica-de-frio-que-fara-as-temperaturas-desabarem-do-sul-ao-norte-do-brasil.ghtml>

²⁸Site oficial: <https://www.accuweather.com/>.

²⁹Site oficial: <https://www.climatempo.com.br/>.

³⁰Fonte: <https://www.climatempo.com.br/climatologia/271/curitiba-pr>

“comum”, essa possibilidade existe, por isso o valor informado pelo sensor pode passar por outro filtro, que indica a temperatura esperada para o período.

O filtro *Previsões* faz uma análise dos dados recebidos da EC com informações de previsão, que podem ser meteorológicas, por exemplo. Previsões oficiais estão disponíveis em várias plataformas acessíveis pela *Internet* e podem ser obtidas gratuitamente ou a um custo muito baixo. A obtenção das informações pode ser conduzida na borda ou na nuvem. A periodicidade depende da fonte dos dados. O Climatempo, por exemplo, oferece dados em tempo real, previsões para as próximas 72 horas e para os próximos 15 dias. A periodicidade com que os dados são obtidos interfere diretamente na eficiência e custo do sistema. Uma atualização diária oferece uma boa relação entre confiabilidade e consumo de recursos.

É importante destacar que previsões climáticas podem não refletir com precisão o comportamento do clima em um dado instante e local. Microrregiões em uma mesma cidade apresentam microclimas diferentes e os dados meteorológicos podem divergir dos registros dos sensores. Entretanto, eles são um bom indicador sobre o comportamento esperado pelo sistema.

Cada um dos filtros pode ser utilizado individualmente ou combinados para oferecer uma classificação mais precisa. Detalhes de sua construção e fontes de dados dependem da necessidade e dos recursos disponíveis. Em ambientes protegidos, como estufas agrícolas, as informações meteorológicas não são úteis, visto que umidade, temperatura e precipitação dos ambientes interno e externo diferem sensivelmente. Neste caso, existe um conjunto de valores esperados e essas informações podem ser utilizadas para compor os filtros. Já as lavouras em campo aberto podem se beneficiar dessas informações.

Ao utilizar dados obtidos remotamente, é importante tomar cuidado com a sua segurança. Um agente malicioso poderia forjar dados falsos e enviar para o EDA de modo a manipular o sistema. Assim, é importante autenticar a origem das informações e verificar a sua autenticidade. É recomendável criptografar os dados e utilizar canais de comunicação seguros.

4.4.3 *Edge ML*: utilizando a aprendizagem de máquina para melhorar a precisão

O *Edge ML* é um classificador baseado em anomalias que utiliza aprendizagem de máquina para determinar um comportamento normal. Ele possui um único filtro, que pode empregar um ou múltiplos algoritmos em conjunto para analisar os dados. A escolha do(s) algoritmo(s) fica a critério do desenvolvedor e deve considerar, entre outras questões, os requisitos computacionais necessários para ser executado e a eficiência do algoritmo. Alguns, utilizam muitos recursos computacionais durante o treinamento da base, inviabilizando sua execução na borda. Uma alternativa é realizar o treinamento na nuvem e transferir o classificador para a borda, deixando para esta apenas a classificação.

Quando existir troca de informações entre a borda e a nuvem, é importante adotar medidas de segurança para proteger tanto os dados quanto os classificadores. Medidas importantes para esse processo são a autenticação mútua das partes comunicantes, autenticação das aplicações na borda e na nuvem, utilização de um canal seguro de comunicação e criptografia dos dados.

Vários algoritmos de aprendizagem de máquina podem ser utilizados por este módulo, sendo preciso avaliar qual se adapta melhor ao padrão de dados. Pesquisas sobre sistemas de detecção de intrusão para IoT tem usado principalmente Máquinas de Vetores de Suporte (SVM, do inglês *Support Vector Machine*), Naïve Bayes, Árvores de Decisão, Florestas Aleatórias e *K-Nearest Neighbor* (KNN) (Mahdavinejad et al., 2018; Keegan et al., 2016). Alguns trabalhos também estudam o uso de Árvores de Hoeffding (HT, do inglês *Hoeffding Tree*), KNN para fluxos de dados (KNN-DS) e Naïve Bayes para fluxos de dados (Naïve Bayes-DS) para detecção de instrução (Alaei e Noorbehbahani, 2017; Dave e Vashishtha, 2013; Keegan et al., 2016; Khan e Jain, 2016).

Os algoritmos mais promissores para operarem na borda são aqueles que combinam pouco consumo de memória e processamento com alto índice de precisão. Algoritmos para dados em lote, como Árvores de Decisões, KNN e Naïve Bayes são bastante promissores para classificar parâmetros em que não são esperadas variações importantes, como os dados gerados em estufas agrícolas. Já os algoritmos desenvolvidos para fluxos prometem ser mais ajustáveis aos valores gerados em campo aberto, especialmente aqueles que sofrem variações sazonais, como temperatura. A maior adaptabilidade se deve à capacidade desses algoritmos de “aprenderem” com os novos registros.

4.4.4 Utilizando Aprendizagem de Máquina para detectar anomalias

A aprendizagem de máquina (ML, do inglês *machine learning*) é um ramo da Inteligência Artificial que utiliza técnicas e algoritmos para criação de modelos analíticos (Sarkar et al., 2018; Silva, 2019). Os modelos conseguem identificar padrões em conjuntos de dados, que podem ser utilizados para diversas finalidades. Na agricultura inteligente, ela pode auxiliar processos de detecção de anomalias e verificação da confiabilidade das informações. Pode ser utilizada como ferramenta principal ou auxiliar, dependendo das características do sistema agrícola e da disponibilidade de recursos. Os requisitos computacionais variam conforme o algoritmo e cada um se ajusta distintamente a diferentes conjuntos de valores. Por isso, o algoritmo de aprendizagem de máquina deve ser escolhido com rigor, considerando o ajuste aos dados e o custo computacional.

Existem vários algoritmos e técnicas de aprendizagem de máquina, que podem ser classificados de diferentes formas (Silva, 2019). Sarkar et al. (2018) classificam os algoritmos quanto ao grau de supervisão humana no processo de treinamento, quanto à forma de generalização e quanto à capacidade de aprendizagem. Por conveniência, este trabalho utiliza a última classificação.

Conforme a capacidade de aprendizagem, os algoritmos são classificados em (i) aprendizagem em lote ou *offline* e (ii) aprendizagem continuada ou *online*. O primeiro utiliza dados previamente disponíveis para treinar o algoritmo e criar o modelo. Uma vez criado o modelo, não são realizados novos treinamentos. O último também realiza um treinamento inicial baseado em dados previamente disponíveis. Entretanto, o modelo continua a aprender com aqueles que chegam ao sistema (Sarkar et al., 2018).

A aprendizagem em lote é eficiente para identificar padrões em conjuntos de valores que não sofrem variações, pois o modelo não assimila novas informações. Se o modelo se tornar ineficiente, é preciso realizar um novo treinamento com um novo conjunto composto pelos dados iniciais e os novos, disponíveis desde a primeira classificação. Isso pode gerar um conjunto muito grande e inviabilizar a abordagem. Os algoritmos dessa categoria podem ser utilizados em sistemas que operam em ambientes controlados, como as estufas, uma vez que as condições tendem a ser constantes e os dados não sofrem alterações significativas.

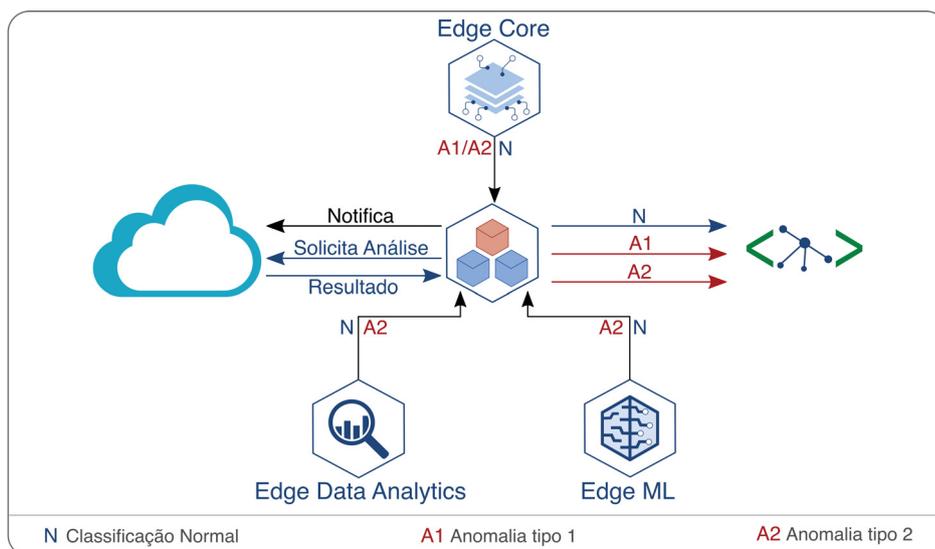
A aprendizagem continuada é utilizada especialmente em algoritmos desenvolvidos para processar grandes fluxos de dados. Algumas aplicações podem gerar valores em uma quantidade e velocidade que não permitem seu armazenamento. Neste caso, a aprendizagem continuada é a mais indicada, visto que ela aprende com os novos registros sem executar todo treinamento com os valores históricos. Contudo, o treinamento contínuo está susceptível às amostras de dados ruins, o que compromete a qualidade da classificação, e gera um consumo extra de recursos computacionais. Os algoritmos para fluxos de dados, como são chamados, são indicados para sistemas operando em campo aberto, cujos registros sofrem variações, como aqueles relacionados à temperatura e umidade relativa do ar.

O algoritmo utilizado pelo detector deve ser escolhido pelo desenvolvedor, considerando as características do conjunto de informações e os requisitos do sistema. Ele pode ser utilizado na borda ou na nuvem, mas para ser executado na borda precisa utilizar memória e processamento compatíveis com os disponíveis. A nuvem geralmente dispõe de recursos mais robustos. Entretanto, há um custo financeiro por processamento o que exige cautela na escolha do algoritmo, visto que ele será executado constantemente para processar os dados gerados pelo sistema. A escolha de um algoritmo com alto custo financeiro ou computacional pode inviabilizar sua utilização.

4.4.5 Módulo de decisões na borda

O módulo de *decisões* determina as ações a serem tomadas diante das classificações geradas por cada classificador. A Figura 4.10 apresenta o esquema de ações desse módulo, sendo elas: notificar a aplicação sobre a ocorrência ou não de anomalias, informar a nuvem sobre a ocorrência atípica de valores fora da escala e valor fundo de escala, solicitar análise do dado pela nuvem ou informar a aplicação sobre dados anômalos.

Figura 4.10: Esquema de ações do módulo Decisões



FONTE: O Autor (2021)

Caso nenhum dos detectores tenham classificado os dados como anomalia, então o módulo *Decisões* receberá o código *N* (classificação normal) e o repassará para a aplicação. Por outro lado, se os filtros de *Escala* ou de *Limites* do Edge Core classificarem a leitura como anomalia, o código recebido será *A1*. Quando os demais filtros do EC, do EDA ou o EML classificarem como anomalia, será enviado o código *A2*. O código *A1* indica para a aplicação que os sensores estão saturados ou apresentam muitos erros. Esses casos exigem intervenção humana para correção e o detector enviará um alarme para notificar o sistema. A classificação pode ser tomada como definitiva, dispensando a análise de outros classificadores. Normalmente, esses registros não indicam o estado do ambiente e os registros podem ser descartados ou armazenados em quarentena e o sistema deve ser notificado.

Já o código *A2* aponta para possíveis falhas e início de processos de degradação. Essas são anomalias “menores”, pois os dados se afastam ligeiramente do estado do ambiente.

Para anomalias A2, o detector aciona um alarme para notificar o sistema e pode adotar diferentes medidas relacionadas aos dados como: *i*) devolver a informação para prosseguir seu processamento³¹, *ii*) descartar o valor, *iii*) marcar como anomalia e devolver para a aplicação, *iv*) marcar como anomalia e encaminhar para a nuvem, ou *v*) reter em quarentena e enviar para análise de outros módulos do sistema (em geral, na nuvem). Os valores descartados não devem ser utilizados para tomadas de decisões para evitar o funcionamento incorreto do sistema. Valores marcados como anomalia e devolvidos para a aplicação apenas recebem a marcação. A aplicação que solicitou a análise deve definir como tratar esse valor. Valores marcados como anomalia e encaminhados para a nuvem são devolvidos para a aplicação com a marcação. Na nuvem o valor pode compor uma base de dados utilizada pelos módulos da nuvem. Caso a ação definida seja reter em quarentena e encaminhar para a nuvem, o decisor encaminha o valor para os filtros da nuvem e aguarda a classificação gerada nesse nível. Valores marcados como anomalia A decisão final é, então, aquela informada pelos classificadores da nuvem. Outras decisões podem ser definidas pelo desenvolvedor da solução.

4.4.6 *Cloud Data Analytics*: analisando os dados na nuvem

Assim como o EDA, o *Cloud Data Analytics* utiliza dados históricos e/ou previsões meteorológicas para identificar comportamentos anômalos. É um módulo executado na nuvem e conta com a disponibilidade de recursos mais robustos, tais como capacidade computacional, largura de banda e conectividade permanente. Na nuvem, podem ser utilizados conjuntos de dados mais complexos e previsões meteorológicas atualizadas com maior frequência ou obtidas de múltiplas fontes. Toda análise do CDA é feita pelo filtro *Analytics*, que processa as informações combinadas ou individualmente.

O filtro *Analytics* utiliza análise de tendências e probabilidade para identificar comportamentos anômalos. Suas análises baseiam-se em informações de longo prazo (meses ou anos) e previsões meteorológicas. As informações de longo prazo podem incluir dados históricos, obtidos de fontes de dados externas ou originados no próprio sistema. Este filtro pode correlacionar diferentes parâmetros, como temperatura, umidade e chuvas, para apresentar tendências mais precisas que as fornecidas pelo EDA. Por exemplo, a umidade do ar influencia a temperatura e as probabilidades de chuvas. Assim, a análise destas informações pode apontar tendências relacionadas à precipitação e temperatura. A correlação dos resultados da análise com registros recentes dos mesmos parâmetros pode apontar para comportamentos anômalos nos dispositivos de percepção.

O *Analytics* pode incluir previsões meteorológicas em suas análises. Na borda, recomenda-se atualizar as previsões em períodos mais esparsos, enquanto a nuvem pode usar as previsões atuais. Alguns provedores disponibilizam informações em tempo real, que tendem a estar mais próximas da realidade. Também é possível utilizar diferentes fontes de dados, especialmente se houver estações meteorológicas próximas. Desta forma, as probabilidades geradas pelo sistema tornam-se mais próximas da realidade. É importante pontuar que as previsões do tempo apresentam dados ligeiramente diferentes dos registradas pelos sensores. Entretanto, essa variação é relativamente constante. Alterações sensíveis nesta variância podem indicar processos de degradação e leituras com uma variância incomum pode assinalar uma anomalia. Sendo assim, pode-se utilizar as Equações 4.2 e 4.5 para detectar desvios de comportamento na nuvem.

O CDA pode ser estruturado de diferentes maneiras: *i*) atuando como um filtro independente na nuvem; *ii*) apoiando o EDA na realização de processamentos; ou *iii*) substituindo

³¹Os dados voltam a seguir o seu processamento/fluxo normal no sistema.

o EDA quando a borda não possui recursos computacionais suficientes. Como um filtro independente, os dados seguem o fluxo normal no CEIFA até chegar ao CDA. Este atribui sua classificação e o detector adota uma ação correspondente. Como suporte ao EDA, o Edge Data Analytics pode realizar consultas diretamente ao CDA, permitindo que a borda considere as tendências e probabilidades do *Analytics*. Ao substituir o EDA, a borda não implanta o EDA e a análise de dados é realizada somente na nuvem. A decisão sobre como estruturar é definida no projeto e deve considerar os custos computacionais, a disponibilidade de recursos e as necessidades do sistema.

Este classificador exige um cuidado adicional com a segurança. A coleta de dados externos, como é o caso das previsões meteorológicas, exige precauções para evitar dados corrompidos ou maliciosos. Esses dados poderiam impactar severamente na eficácia do detector. Alguns requisitos importantes de segurança são: autenticação da fonte de dados, verificação de integridade dos dados recebidos e utilização de canais seguros para transferência de dados. Atender a esses requisitos reduz a probabilidade do sistema ser manipulado por agentes maliciosos, aumentando sua segurança e eficiência.

4.4.7 *Cloud ML*

O Cloud ML (CML) é um módulo hospedado na nuvem que utiliza um ou mais algoritmos de aprendizagem de máquina para detectar anomalias e ataques. Ele funciona de forma semelhante ao EML. Entretanto, sua localização na nuvem permite empregar algoritmos mais robustos, que podem operar individualmente ou combinados. Enquanto o EML deve usar algoritmos compatíveis com dispositivos restritivos, o CML pode implementar algoritmos para fluxos de dados, tais como Árvores de Hoeffding e KNN-DS, SVM, ou mesmo usar aprendizagem profunda (*Deep Learning*). Um ou mais algoritmos podem ser usados, dispostos em conjunto ou em votação. A escolha do algoritmo envolve questões como o custo de processamento e o ajuste aos dados.

O CML pode ser utilizado como apoio aos módulos da borda, atuando como mais um nível de filtragem, ou em substituição ao EML. No primeiro caso, a aprendizagem de máquina (ML, do inglês *machine learning*) pode ser treinada com modelos capazes de reconhecer ataques originados na Internet, na borda ou camada de percepção e destinados à nuvem. Também podem ser treinados para reconhecer anomalias nos registros encaminhados pela borda de modo a identificar aquelas que as camadas inferiores não conseguem detectar. No último, age como um apoio, analisando os valores e devolvendo uma classificação para a borda.

Quando atuar como mais um nível de filtragem, a CML receberá todos os registros processados pelo detector. Os dados podem ser (i) aqueles originados na nuvem ou na Internet e destinados à borda, (ii) originados na borda ou na percepção e destinado à nuvem, ou ambos. O classificador pode ser treinado para identificar ataques como os descritos em Xiao et al. (2018), Coulter e Pan (2018) e Mamdouh et al. (2018), ou para detectar as anomalias apresentadas na Seção 4.2. É recomendável que esse classificador seja treinado para identificar falhas que os outros classificadores não conseguem detectar.

Quando funcionar como um classificador de apoio, a CML recebe apenas os dados em que há “dúvidas” sobre a classificação atribuída por outros classificadores. Por exemplo, o EDA classificou um valor como *anomalia*, mas a diferença entre o valor esperado e a leitura do sensor está dentro de uma tolerância que pode “indicar” um registro normal. Neste caso, a leitura pode ser submetida à análise do CML. O classificador pode ser treinado para correlacionar diferentes parâmetros (umidade relativa do ar, temperatura e pressão atmosférica, por exemplo) e, considerando a correlação entre eles, verificar se a leitura é, potencialmente, uma anomalia.

Novamente, quais algoritmos usar e como combiná-los depende dos requisitos do sistema. Um ponto a ser considerado são os custos de computação em nuvem. Existem vários provedores de serviços em nuvem, sendo os mais conhecidos: Amazon AWS, Microsoft Azure, Google Cloud Platform e Oracle Cloud.

4.4.8 Módulo de Decisões da Nuvem

O módulo *Decisões* da nuvem encarrega-se das operações realizadas pela nuvem, decorrentes da classificação dos dados. Entre as ações definidas estão: registrar informações sobre a classificação, notificar os administradores, notificar uma aplicação e acionar um alarme. O registro sobre a classificação pode ser efetuado somente quando ocorrer uma anomalia, somente quando o valor não for anômalo ou ambos casos. Os registros podem ser utilizados para auditorias, processos de análises de dados ou pelo próprio sistema agrícola. A notificação aos administradores é útil especialmente em casos que exigem atenção humana. Alguns casos típicos são aqueles em que um sensor apresenta muitas falhas recorrentes ou entra em processo de saturação. A notificação a uma aplicação pode ser interessante, por exemplo, quando existir um processo capaz de intervir no sistema para isolar uma falha.

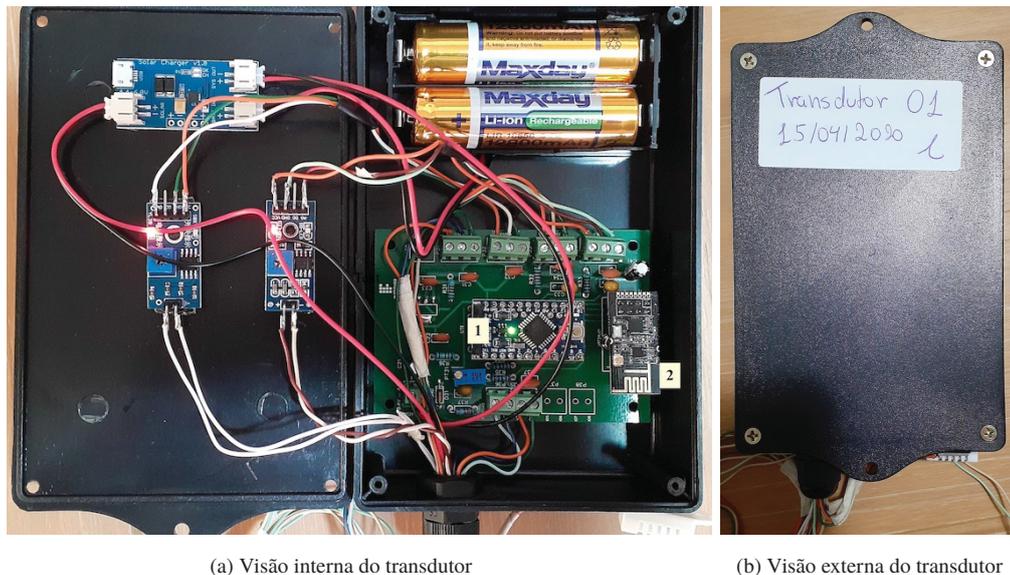
Este módulo não interage com os módulos da borda. Sua função é executar as tarefas finais e encerrar o processo na nuvem. A interação com os módulos da borda é feita diretamente pelos classificadores, quando for o caso. A partir do trabalho deste decisor, outros subsistemas podem ser notificados e entrar em ação. Este decisor é, portanto, o elemento final do detector de anomalias.

Embora o CEIFA esteja estruturado para trabalhar na borda e na nuvem, ele tem apenas um módulo indispensável, o EC. No entanto, o EC pode não detectar todas as anomalias, exigindo o suporte de outros módulos. O CEIFA pode empregar os módulos de análise de dados e aprendizagem de máquinas na borda, na nuvem, ou ambos. A vantagem de incluir estes classificadores na borda é a economia nos custos financeiros ligados à transmissão de dados e à computação em nuvem e a baixa latência. Entretanto, as técnicas na borda precisam ser compatíveis com as restrições dos dispositivos que tipicamente têm baixa capacidade computacional. Por outro lado, a nuvem tem alta capacidade computacional, permitindo o uso de técnicas mais eficientes. Uma melhor relação entre custo e desempenho pode ser obtida usando todos os módulos da arquitetura proposta.

5 AVALIAÇÃO E RESULTADOS

Para trabalhar com dados realistas criou-se um ambiente de testes. Este ambiente foi constituído por um conjunto de sensores e dispositivos normalmente utilizados em projetos de eletrônica voltados para a IoT. Optou-se por utilizar sensores comuns e de baixo custo, que sofrem processos de degradação, para possibilitar a validação do detector de anomalias em tempo hábil. Foram instalados cinco transdutores³², iguais aos mostrados na Figura 5.1 e um minicomputador.

Figura 5.1: Transdutor composto por sensores de parâmetros ambientais



(a) Visão interna do transdutor

(b) Visão externa do transdutor

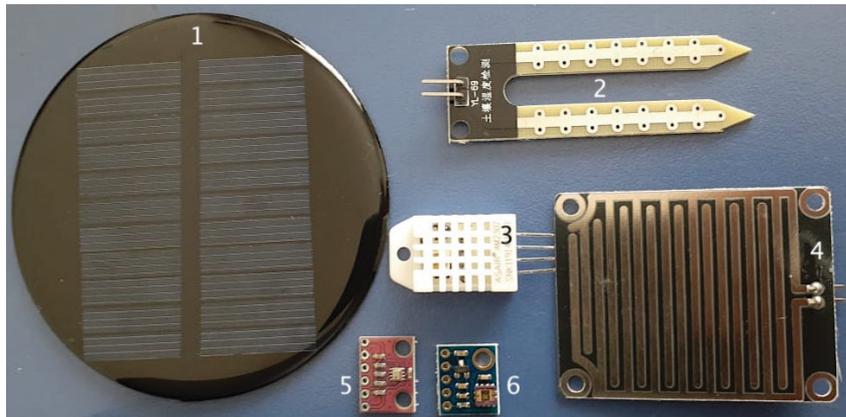
FONTE: O Autor (2021)

Cada transdutor foi equipado com um Arduino Pro Mini, com processador ATmega328P, uma placa de rede sem fio GT-24 nRF24L01, cinco sensores, uma pequena placa solar e um conjunto de baterias para armazenar energia. O Arduino pode ser observado na Figura 5.1(a), marcado com o número 1, e a placa de rede está na mesma figura, marcada com o número 2. A placa solar e os sensores são os mostrados na Figura 5.2, sendo: (1) placa solar, sensores (2) de umidade do solo, (3) DHT22, que mede a umidade relativa do ar e a temperatura, (4) precipitação (chuva), (5) pressão atmosférica e (6) raios UV. O minicomputador, um Raspberry Pi descrito no Apêndice B, foi utilizado como borda da rede. Esse dispositivo foi equipado com uma placa de rede sem fio e cartão de memória. A Tabela D.1 presente no Apêndice D apresenta os gastos com os equipamentos utilizados neste projeto.

Inicialmente os dispositivos foram instalados em um ambiente externo, de modo a coletar dados para utilizar nos testes. Os dispositivos permaneceram instalados de abril de 2020, a abril de 2021. Os dados coletados pelos sensores foram enviados para o *gateway*, que os encaminhou para a nuvem. Como nuvem, foram utilizados dois servidores: (i) um servidor local, equipado com um processador Intel Core i3-9100 3.60GHz e 4GB de memória e (ii) um servidor virtual, com um vCPU e 1GB de memória, alocado na Digital Ocean. Os dados coletados pelos sensores foram utilizados na fase de desenvolvimento e nos primeiros testes dos algoritmos.

³²As fotos dos transdutores estão disponíveis em um repositório público (Gthub)

Figura 5.2: Sensores utilizados no projeto



FONTE: O Autor (2021)

5.1 ANÁLISE DE PRECISÃO

Antes de construir o detector proposto, investigou-se o consumo de recursos computacionais por parte dos algoritmos de aprendizagem de máquina. Esse estudo, detalhado no Apêndice C, pautou a escolha dos algoritmos para análise de precisão apresentada nesta seção. A partir dos resultados, foram escolhidas as Árvores de Decisões, as Árvores de Hoeffding e Naïve Bayes para fluxos de dados. Os resultados de cada algoritmo foram comparados com uma simulação do Edge Core, que implementa as equações propostas na Seção 4.4.1.1. Esta análise avaliou o desempenho dos algoritmos para identificar as anomalias apresentadas na Seção 4.2.

A avaliação da precisão utilizou os dados coletados previamente e classificados por um especialista. É importante ressaltar que estes dados foram gerados por sensores instalados em ambiente de testes, sem terem suas escalas calibradas. Isso implica na existência de diferenças entre as escalas dos sensores instalados. As bases de dados foram classificadas conforme a anomalia alvo e as demais anomalias não foram retiradas da base. Por isso, uma base de dados que contenha degradação, também pode conter outras falhas. Todavia, se o alvo era degradação, apenas essa foi marcada como anomalia e os demais dados foram marcados como normais. A Tabela 5.1 apresenta a quantidade de dados normais e anômalos para cada base de dados utilizada nessa fase.

Tabela 5.1: Composição dos conjuntos de dados utilizados para análise de precisão

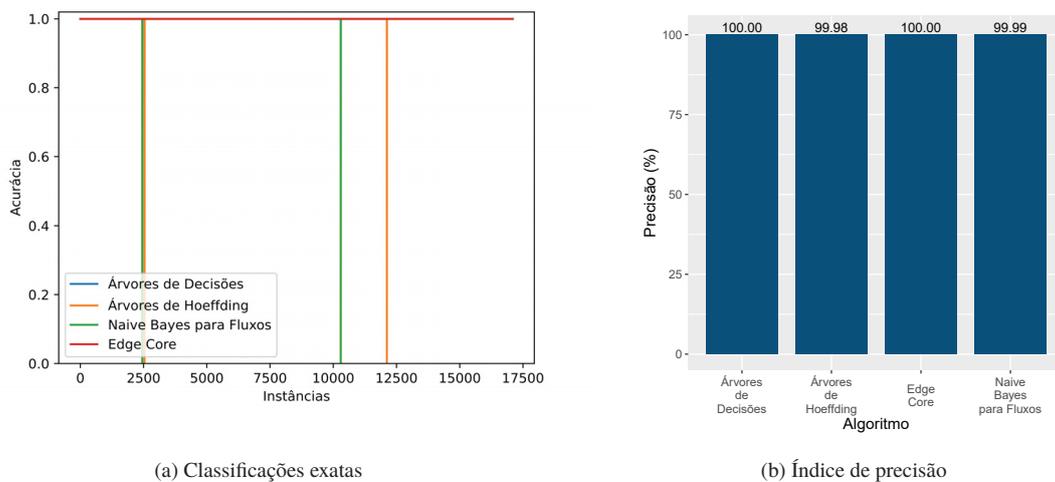
Conjunto de Dados	Dados Normais*	Dados Anômalos*
Falhas aleatórias	1.704.300	5.300
Saturação	1.439.200	270.400
Degradação	231.700	684.700
Ruídos	582.100	15.400
Colisão	752.800	78.100
Injeção de Dados Falsos	995.7900	34.100

*Quantidade de registros que compõem a base de dados

FONTE: O Autor (2020)

Primeiro analisou-se a capacidade de identificação de Falhas Aleatórias. A análise utilizou uma base de dados³³ composta por 1.709.600 registros, dos quais 5.300 eram dados anômalos. Geralmente os sensores registram poucas ou nenhuma falha, por isso a ocorrência dessas anomalias é bastante baixa em relação à quantidade total de dados. O resultado é apresentado na Figura 5.3. A Figura 5.3(a) mostra a precisão de cada algoritmo durante a classificação. O Edge Core e as Árvores de decisões alcançaram 100% de precisão, por isso há uma linha contínua percorrendo o topo do gráfico. O Árvores de Hoeffding e Naive Bayes erraram algumas classificações, o que é demonstrado pela linha que retorna à base do gráfico. A Figura 5.3(b) sumariza os resultados. Todos os algoritmos obtiveram bom desempenho.

Figura 5.3: Precisão para detecção de falhas aleatórias



FONTE: O Autor (2021)

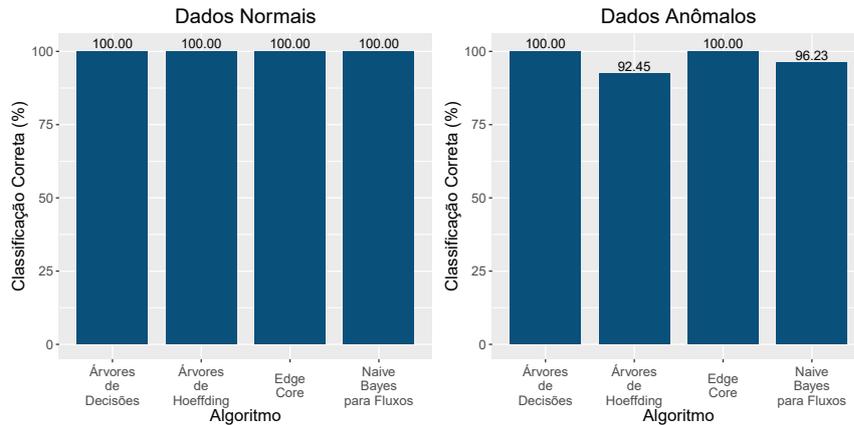
Os indicadores sobre a classificação de dados normais e anômalos foram separados de modo a obter informações mais precisas sobre os erros e acertos dos algoritmos. Essas informações são apresentadas na Figura 5.4. A figura mostra que todos os algoritmos classificaram os dados normais corretamente. As Árvores de Decisões e o Edge Core também foram precisos na identificação de dados anômalos. Já os algoritmos para fluxos de dados classificaram algumas falhas como dados válidos. Naive Bayes classificou 200 registros anômalos como normais e as Árvores de Hoeffding o dobro desse valor.

Em seguida, verificou-se a capacidade de detecção de saturações. Para isso foram utilizados dados de umidade relativa do ar. Alguns sensores ficaram longo período registrando um alto índice de umidade do ar, resultando em um conjunto de 270.400 registros de saturação. A classificação foi realizada manualmente por um especialista, o que não descarta a possibilidade de erros. Os resultados são apresentados nas Figuras 5.5 e 5.6. O Edge Core obteve o melhor desempenho, seguido das Árvores de Hoeffding e Naive Bayes. O pior desempenho foi alcançado pelas Árvores de Decisões.

Analisando os dados sintetizados na Figura 5.6, percebe-se que o Edge Core cometeu poucos erros tanto na classificação de dados normais quanto com os dados anômalos. Dos 1.709.600 registros, apenas 1.300 foram classificados incorretamente. As Árvores de Hoeffding também alcançaram bom desempenho na classificação de dados normais, mas falhou na

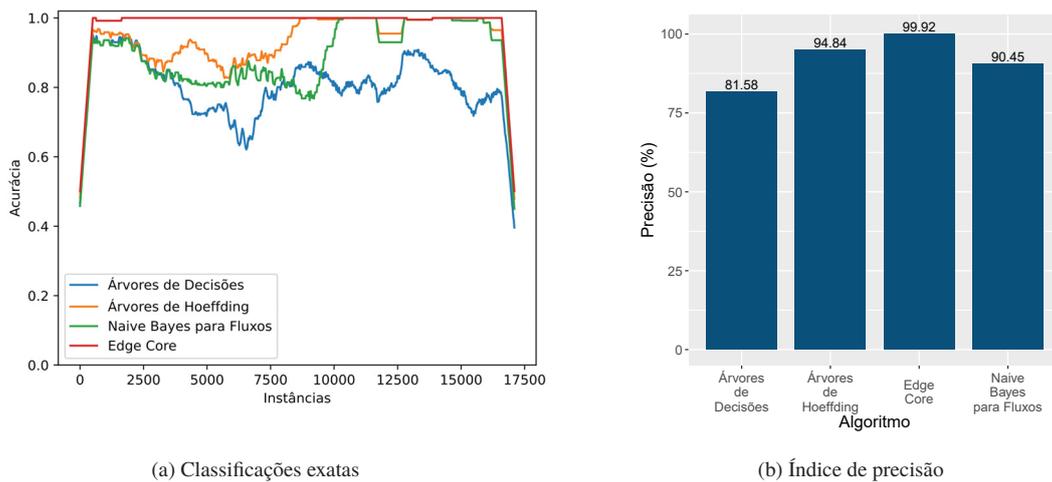
³³O conjunto é formado por dados de umidade relativa do ar. Este sensor retornou códigos de erros em alguns períodos.

Figura 5.4: Falhas Aleatórias: dados classificados corretamente



FONTE: O Autor (2021)

Figura 5.5: Precisão para detecção de saturação



(a) Classificações exatas

(b) Índice de precisão

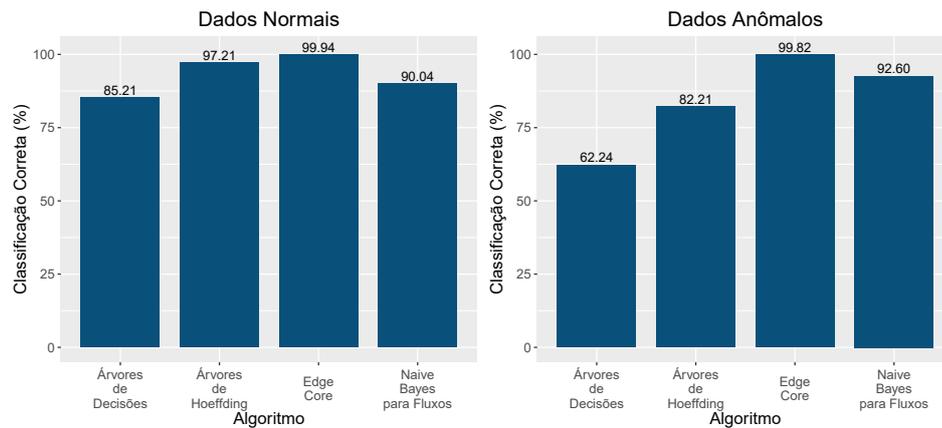
FONTE: O Autor (2021)

classificação de 88.300 registros. Naive Bayes, por sua vez, cometeu 163.300 erros, porém foi muito mais preciso que as Árvores de Decisões, que classificaram 314.900 registros erroneamente.

Na sequência verificou-se a capacidade de identificar choques em sensores e manuseio não autorizado. Para avaliar esta anomalia, foi simulada uma situação em que um animal ou pessoa colide com o sensor de umidade do solo, retirando-o da sua instalação original, sem danificá-lo. Este choque impede o sensor de registrar a umidade corretamente. Os dispositivos utilizados para gerar os dados não foram calibrados, o que significa que os registros podem conter erros de leitura. Alguns dos sensores utilizados para gerar esses dados já haviam entrado em processo de degradação no momento da coleta. Os resultados (Figura 5.7) mostram que os algoritmos para fluxos são os mais precisos.

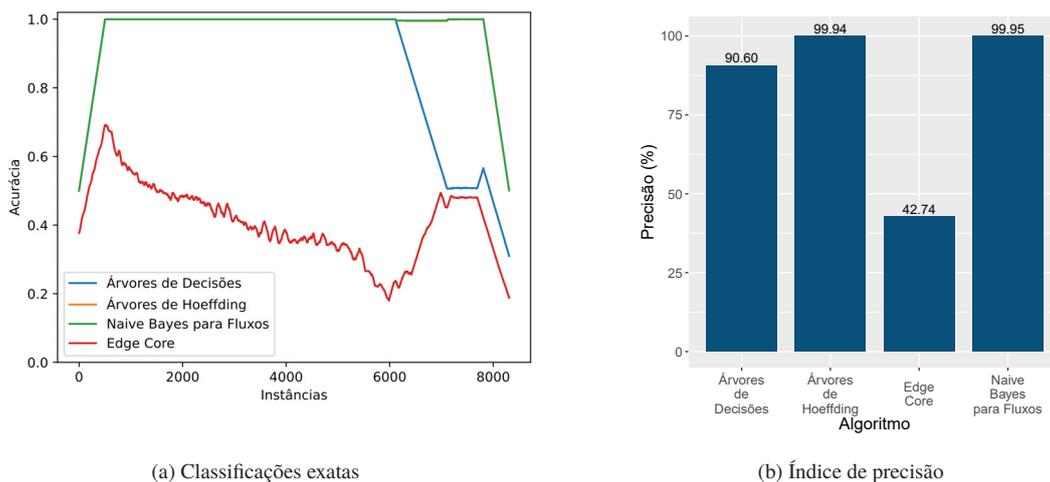
Ao observar os gráficos de acertos (Figura 5.8), percebe-se que os algoritmos de aprendizagem de máquina foram precisos na classificação de todos os dados normais. Entretanto, as Árvores de Decisões não classificaram nenhum registro como anomalia, mostrando-se incapaz

Figura 5.6: Saturação: dados classificados corretamente



FONTE: O Autor (2021)

Figura 5.7: Precisão para detecção de choques e manuseio não autorizado



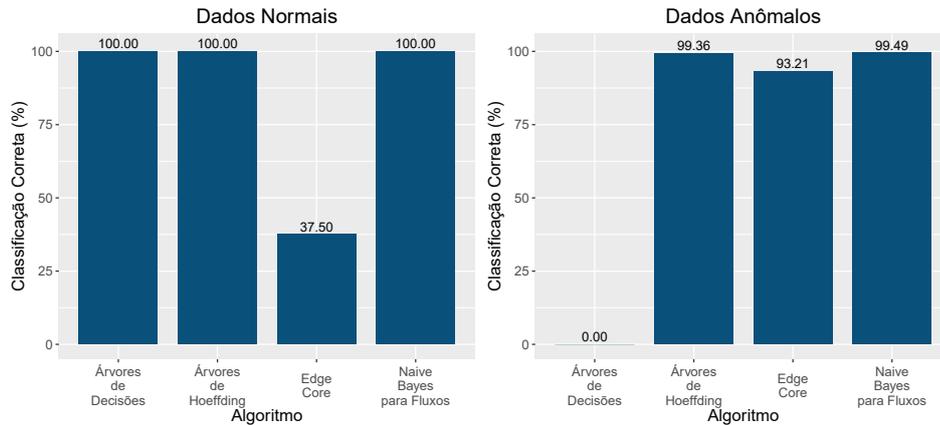
FONTE: O Autor (2021)

de detectar esse tipo de evento. O Edge Core falhou na classificação de 470.500 registros normais e 5.300 registros anômalos, apresentando o pior desempenho entre todos os algoritmos testados.

Para identificar processos de degradação foram utilizados dados dos sensores de umidade do solo. Eles permaneceram um longo período expostos ao ambiente externo e em contato com o solo. Não demorou para que começassem a registrar informações diferentes de sensores não degradados. Como degradação ocorreu muito rapidamente, a quantidade de dados normais (Tabela 5.1) passou a ser inferior à quantidade de dados anômalos. Quando esse evento foi observado, foram adicionados novos sensores não degradados à rede. Ainda assim, a quantidade de dados anômalos foi superior à quantidade de dados normais. O Edge Core alcançou um bom desempenho na detecção desta anomalia, apesar da precisão reduzir conforme a quantidade de sensores degradados aumentava. A Figura 5.9 apresenta os dados de precisão.

Os algoritmos de aprendizagem de máquina foram muito precisos. As Árvores de Hoeffding e Árvores de Decisões classificaram todos os dados corretamente. Naive Bayes classificou 400 registros normais como anômalos. Já o Edge Core identificou todos os registros

Figura 5.8: Choques e manuseio não autorizado: dados classificados corretamente

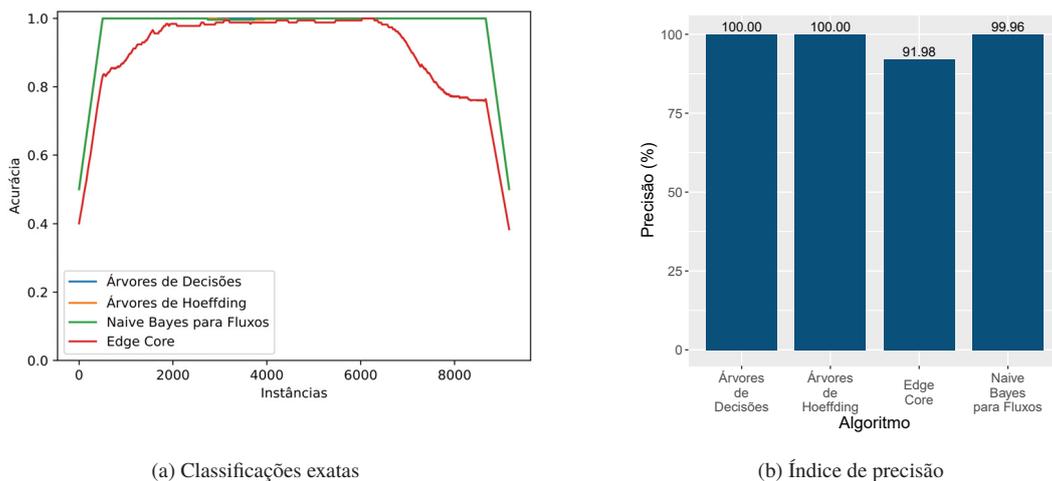


FONTE: O Autor (2021)

de sensores não degradados, mas falhou em 73.500 dos 684.700 registros oriundos de sensores degradados. Os resultados, apresentados na Figura 5.10, mostram que o índice é baixo, apesar de superior aos dos demais algoritmos.

Para testar a detecção de ruídos, foi criado um dispositivo equipado com um sensor de umidade relativa do ar. Em períodos aleatórios, após coletar a informação sobre a umidade, o valor em binário sofria uma alteração de um bit, aleatoriamente. Esse cenário simula ruídos capazes de alterar bits de dados durante sua transmissão. Os dados produzidos pelo sensor foram enviados para borda. Observando a precisão geral (Figura 5.11), os algoritmos para fluxos de dados e o Edge Core alcançaram excelente desempenho.

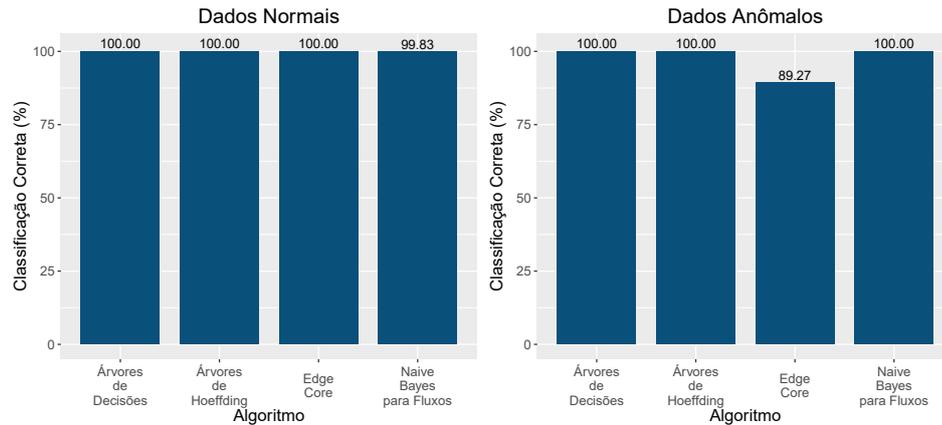
Figura 5.9: Precisão para detecção de degradação



FONTE: O Autor (2021)

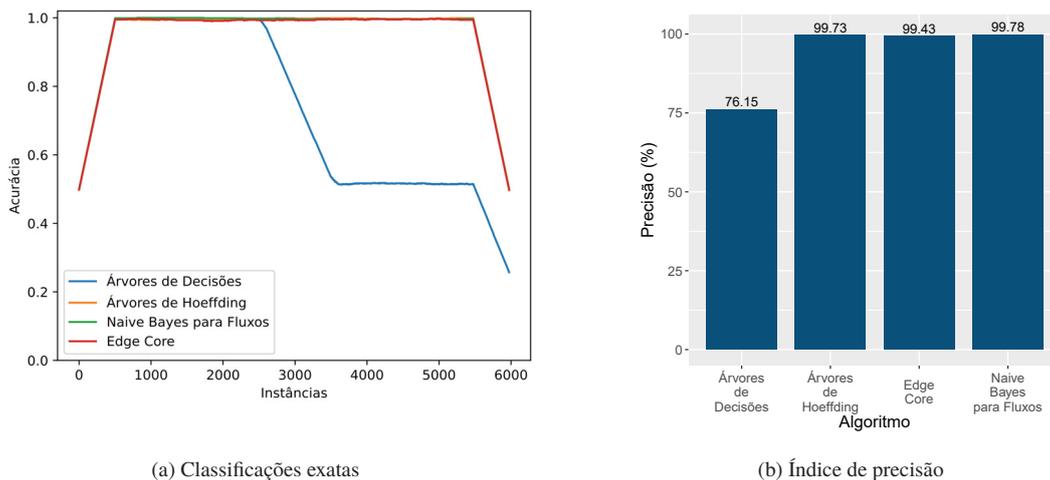
O gráfico de acertos (Figura 5.12), mostra que o Naive Bayes para fluxos de dados é o mais preciso na detecção de ruídos. Ele classificou corretamente todos os dados normais, mas falhou em 1.300 registros anômalos. As Árvores de Hoeffding falharam um pouco mais, classificando incorretamente 1.600 registros anômalos. O Edge Core cometeu mais erros, falhando em 3.400

Figura 5.10: Degradação: dados classificados corretamente



FONTE: O Autor (2021)

Figura 5.11: Precisão para detecção de ruídos



(a) Classificações exatas

(b) Índice de precisão

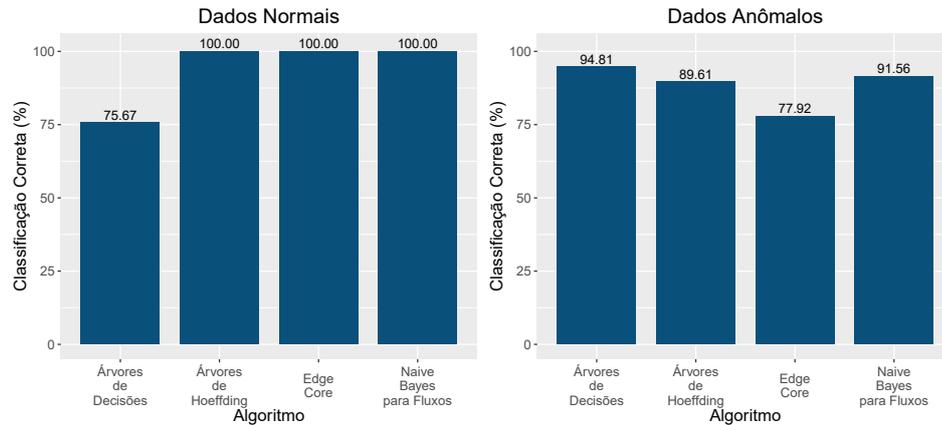
FONTE: O Autor (2021)

registros anômalos. As Árvores de Decisões foram mais precisas na classificação de dados anômalos, mas falharam na identificação de 141.600 registros normais.

Por fim, verificou-se a injeção de dados falsos. Para esse processo, foi criado um nó malicioso que enviou alguns dados que divergem levemente dos valores reais. O processo rodou por cinco dias consecutivos. O objetivo era identificar uma tentativa de manipular o sistema, alterando as leituras para valores superiores e inferiores. O Edge Core não consegue detectar essa anomalia, por isso foi suprimido dos gráficos. A Figura 5.13 mostra que o pior desempenho foi alcançado pelas Árvores de Decisões, enquanto as Árvores de Hoeffding são as mais precisas.

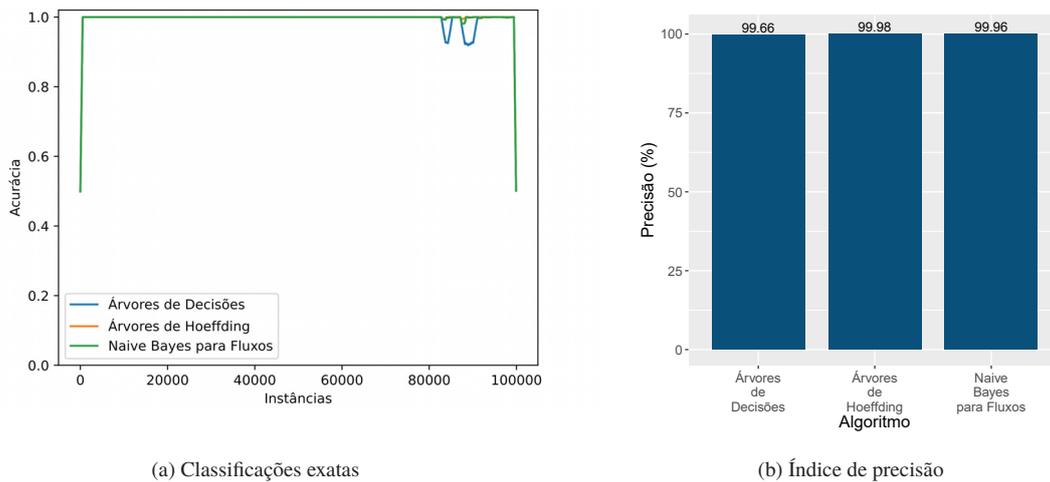
O gráfico de acertos (Figura 5.14) mostra que o algoritmo para séries de dados foi incapaz de identificar os dados injetados e classificaram todos como normais. As Árvores de Hoeffding foram as mais precisas, classificando alguns poucos dados anômalos (1.500) como corretos. Naive Bayes-DS falhou ao classificar 1.200 registros corretos e 2.500 dados injetados, o que é uma taxa bastante baixa diante dos quase 1 milhão de registros.

Figura 5.12: Ruídos: dados classificados corretamente



FONTE: O Autor (2021)

Figura 5.13: Precisão para detecção de injeção de dados falsos



(a) Classificações exatas

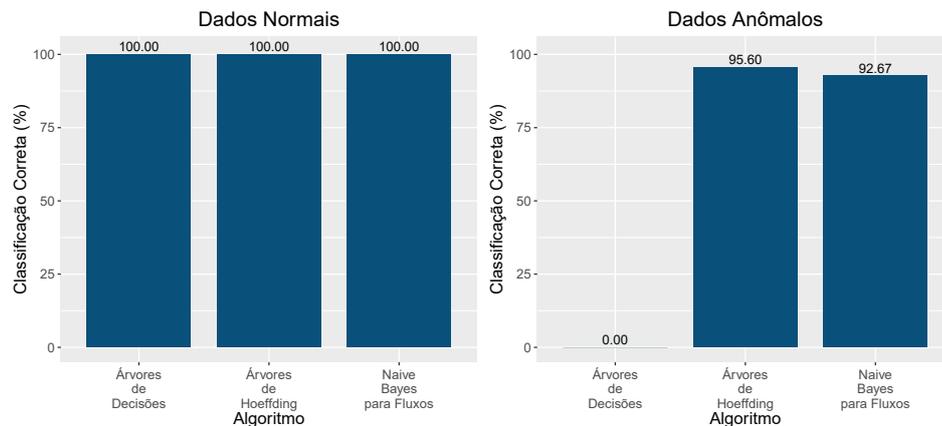
(b) Índice de precisão

FONTE: O Autor (2021)

O resultado geral, apresentado na Tabela 5.2, mostra que o Edge Core alcança bom desempenho para detecção da maioria das anomalias avaliadas, mas precisa do apoio de outros algoritmos, especialmente para detecção de choques e injeção de dados falsos. Os algoritmos para fluxos de dados são os mais precisos e alcançam melhor desempenho especialmente quando o Edge Core falha em maior frequência.

Observando o índice de acertos para identificação de dados normais (Tabela 5.4) e anômalos (Tabela 5.3), observa-se que a taxa total de acertos do Árvores de Hoeffding é superior à do Naive Bayes para fluxos de dados. Entretanto, considerando apenas as anomalias em que o Edge Core é inferior aos demais, o índice de acertos dos dois algoritmos é equivalente. Sabendo que o consumo de recursos computacionais do Naive Bayes-DS é menor (vide Apêndice C), este algoritmo tende a ser o mais adequado para executar na borda. Portanto, o Naive Bayes-DS foi selecionado para compor o Edge ML. Para a nuvem, o algoritmo escolhido foi o Árvores de Hoeffding, que apresenta maior índice geral de acertos.

Figura 5.14: Erros e acertos durante a detecção de injeção de dados falsos



FONTE: O Autor (2021)

Tabela 5.2: Índice de precisão dos algoritmos

Anomalia	Edge Core	Árvores de Decisões	Árvores de Hoeffding	Naive Bayes para Fluxos
Falhas Aleatórias	100,00%	100,00%	99,98%	99,99%
Saturação	99,92%	81,58%	94,83%	90,45%
Choques	42,73%	90,60%	99,94%	99,95%
Degradação	91,98%	100,00%	100,00%	99,96%
Ruídos	99,43%	76,15%	99,73%	99,78%
Injeção de Dados Falsos	-	99,65%	99,98%	99,96%
Taxa de Acertos	86,81%	91,33%	99,08%	98,35%

Tabela 5.3: Precisão na classificação de dados “anômalos”

Anomalia	Edge Core	Árvores de Decisões	Árvores de Hoeffding	Naive Bayes para Fluxos
Falhas Aleatórias	100,00%	100,00%	92,45%	96,23%
Saturação	99,82%	62,24%	82,21%	92,60%
Choques	93,21%	0,00%	99,36%	99,49%
Degradação	89,27%	100,00%	100,00%	100,00%
Ruídos	77,92%	94,81%	89,61%	91,56%
Injeção de Dados Falsos	-	0,00%	95,60%	92,67%
Taxa de Acertos	92,04%	59,51%	93,21%	95,42%

5.2 CONSUMO DE RECURSOS COMPUTACIONAIS DO *EDGE CORE*

Dada a boa precisão alcançada pela simulação do Edge Core, foi realizada sua implementação e a do Edge Data Analytics a fim de verificar o consumo de recursos na borda. Os resultados dos dois módulos foram comparados aos alcançados pelo Naive Bayes para fluxos de dados. Na nuvem, analisou-se o consumo de recursos por parte do Cloud Data Analytics. Na borda, o EDA consulta informações sobre a previsão do tempo uma vez ao dia. Na nuvem, o EC consulta dos dados de duas fontes: um provedor de previsões e uma estação meteorológica

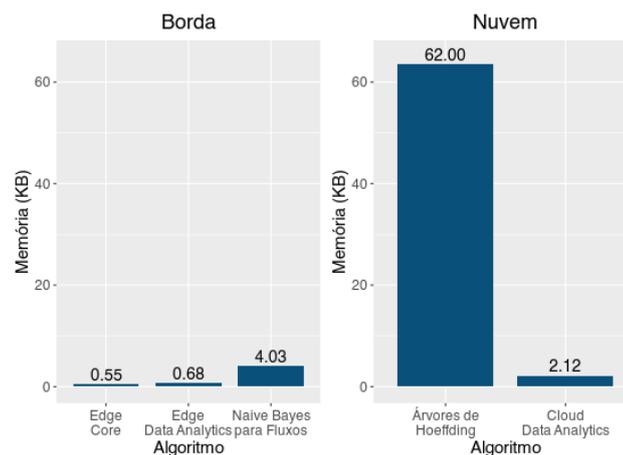
Tabela 5.4: Precisão na classificação de dados “normais”

Anomalia	Edge Core	Árvores de Decisões	Árvores de Hoeffding	Naïve Bayes para Fluxos
Falhas Aleatórias	100,00%	100,00%	100,00%	100,00%
Saturação	99,94%	85,21%	97,21%	90,04%
Choque	37,50%	100,00%	100,00%	100,00%
Degradação	100,00%	100,00%	100,00%	99,83%
Ruídos	100,00%	75,67%	100,00%	100,00%
Injeção de Dados Falsos		100,00%	100,00%	100,00%
Taxa de Acertos	87,49%	93,48%	99,53%	98,31%

instalada na cidade vizinha. Os resultados do CDA foram comparados aos alcançados pelas Árvores de Hoeffding. Foram investigados o consumo de memória e o processamento para a classificação de conjuntos de um milhão de valores, além do espaço ocupado em disco por arquivos do sistema. A análise não incluiu o espaço ocupado pelo banco de dados e pelas bibliotecas.

A análise da memória utilizou as bibliotecas Guppy e Pympler, conforme descrito no Anexo C. Pympler é uma ferramenta para medir e analisar o comportamento da memória de objetos Python, em uma aplicação Python em execução. Guppy fornece um conjunto de ferramentas de análise da memória. O Pympler (Figura 5.15) informa que os objetos Edge Core e o EDA alocam 568 e 696 *bytes* de memória, respectivamente. Isso representa cerca de um sexto da memória alocada pelo Naïve Bayes para fluxos de dados (4.127 *bytes*). Ao todo, os objetos CEIFA ocupam 5KB de memória na borda. O consumo de memória na nuvem é maior. Os objetos do CDA ocupam cerca de 2KB e os do EML aproximadamente 62KB de memória.

Figura 5.15: Pympler: Memória alocada pelos objetos

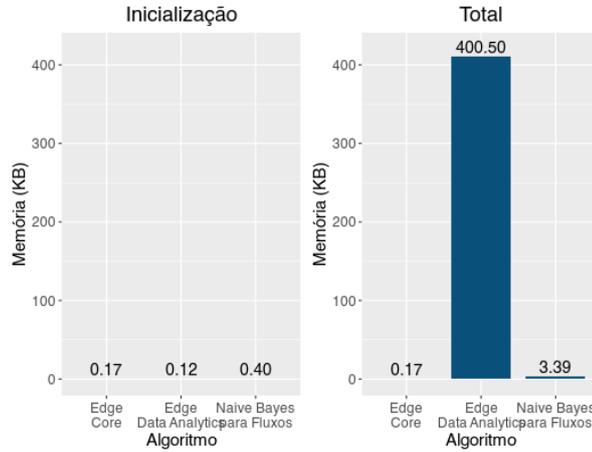


FONTE: O Autor (2021)

Guppy fornece dados sobre o consumo de memória durante a execução. Na inicialização (Figura 5.16), Edge Core aloca 176 *bytes*, o Edge Data Analytics, 120 *bytes*, e Naïve Bayes para fluxos de dados, 3428 *bytes*. Quando os dados são classificados, o EDA usa 400KB de memória adicional. Esta memória está relacionada à recuperação de dados de previsão do tempo. As Árvores de Hoeffding alocaram aproximadamente 3KB durante o treinamento do algoritmo e cerca de 44 *bytes* adicionais na classificação dos dados. No total, o Edge Core ocupou 176 *bytes*,

o EDA 401KB e o EML 3KB de memória RAM. O Edge Core não utiliza memória adicional para classificar os dados.

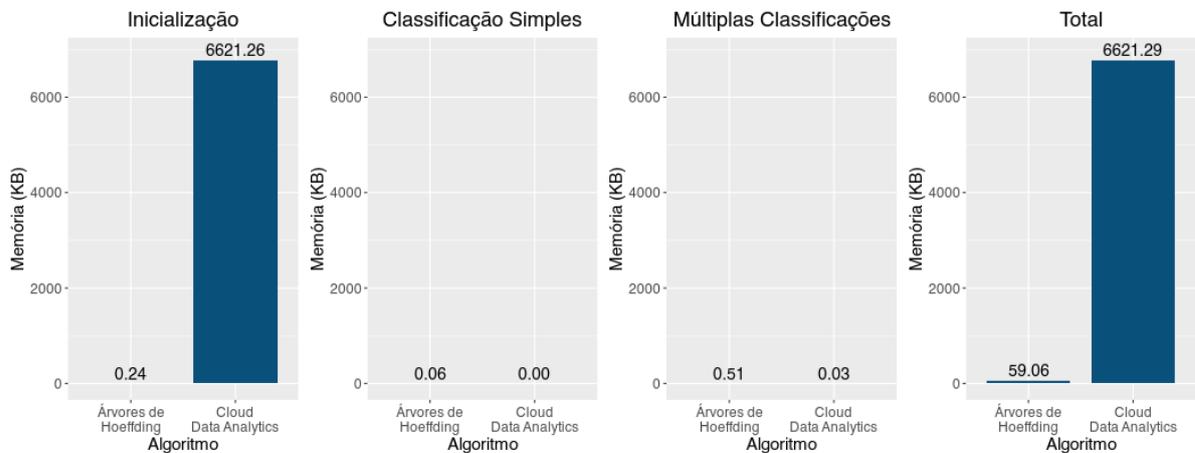
Figura 5.16: Guppy: alocação de memória na borda



FONTE: O Autor (2021)

Na nuvem (Figura 5.17), o CDA requer 6MB de memória na inicialização. Neste ponto, o módulo consulta as informações meteorológicas externas e armazena os dados na memória. Em geral, os provedores fornecem mais parâmetros do que o necessário e é preciso baixar todos os dados e depois filtrá-los. O CDA consultou uma estação meteorológica local, que fornece uma extensa base de dados, resultando em um alto consumo de memória. As Árvores de Hoeffding são bastante econômicas, alocando 28 bytes na inicialização. Para classificar um único registro, o CDA não utiliza memória adicional e as Árvores de Hoeffding utilizam 57 bytes. Quando se trata da classificação de múltiplos registros (mil registros), o CDA aloca 28 bytes de memória adicional e as Árvores de Hoeffding 524 bytes. No total, o CDA usa 6MB de memória, enquanto as Árvores de Hoeffding usam 59KB.

Figura 5.17: Guppy: alocação de memória na nuvem



FONTE: O Autor (2021)

Como as bibliotecas Guppy e Pympler não fornecem informações sobre toda memória consumida pelo processo, foi realizada uma terceira avaliação considerando os dados do sistema

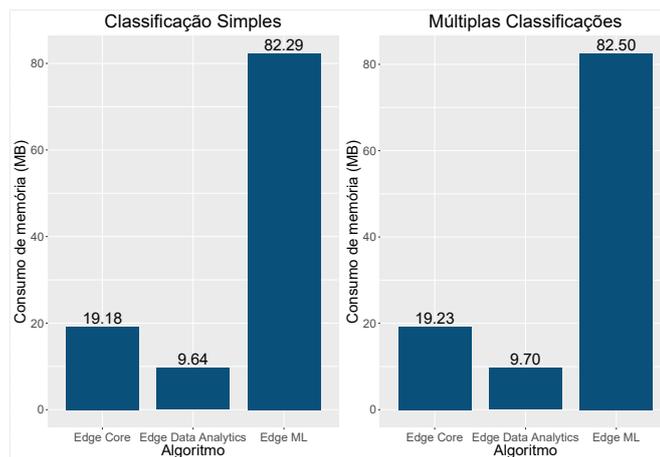
operacional. Foram consideradas as informações registradas no arquivo `smaps`, contidas no diretório `/proc/pid`³⁴. O arquivo `smaps` contém informações detalhadas sobre todos os acessos às memórias físicas e virtuais utilizadas durante a execução do processo. Essas informações são resultantes de mapeamentos das áreas de memória, realizados pelo sistema operacional. Os resultados apresentam a soma dos registros de `PSS`, que representa a contagem de páginas que um processo possui na memória (Kernel development community, 2022). Durante a execução do CEIFA são realizados diversos mapeamentos, cujas quantidades são apresentadas na Tabela 5.5.

Tabela 5.5: Mapeamentos realizados pelo `smaps`

Módulo do CEIFA	Quantidade de mapeamentos
Edge Core	846
Edge Data Analytics	844
Edge Machine Learning	2056
Cloud Data Analytics	669
Cloud Machine Learning	3016

De acordo com os dados apresentados na Figura 5.18, o Edge Core consome 19,18MB de memória para classificar um único registro e 19,23 para classificar múltiplos registros. Isso representa um incremento de pouco mais de 49KB de memória. Diferentemente do que apontam o Guppy e o Pympler, os dados registrados no `smaps` apontam que o Edge Data Analytics é mais econômico, consumindo 9,64MB de memória para classificar um registro. A classificação de múltiplos registros adiciona 61KB à essa memória, totalizando 9,70MB. O maior consumo de memória é feito pelo Edge ML, que utiliza mais de 82MB. O processamento de múltiplos registros por esse classificador adiciona mais de 217KB de memória.

Figura 5.18: Processo: Memória alocada na borda



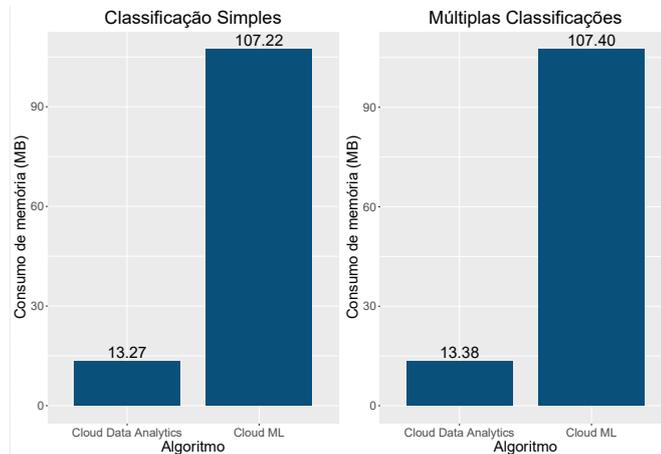
FONTE: O Autor (2022)

Na nuvem, o Cloud Data Analytics consome 13,27KB de memória para classificar um registro e 13,38 para classificar múltiplos registros, um acréscimo de 113KB. Novamente a aprendizagem de máquina consome mais recursos, adicionando 107,22MB de memória para processamento simples e 183KB para classificar múltiplos registros, totalizando 107,40MB na

³⁴`pid` é o identificador do processo.

classificação de múltiplos registros. Ao todo, o CEIFA consome aproximadamente 111MB de memória para classificação de registros na borda e 120MB na nuvem, com uma variação aproximadamente 300KB entre a classificação de um único registro e múltiplos registros.

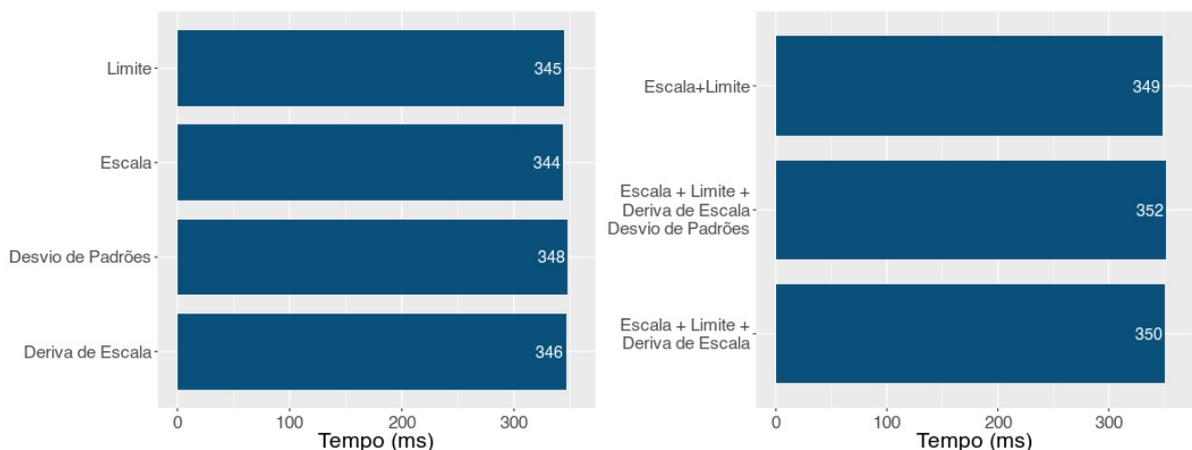
Figura 5.19: Processo: Memória alocada na nuvem



FONTE: O Autor (2022)

A avaliação do processamento utilizou a biblioteca Time. Para obter uma visão completa, foram coletados dados sobre os tempos de processamento de cada filtro e os tempos agrupando os filtros. Isso porque alguns valores podem passar por todos os filtros, enquanto outros podem ser classificados como anômalos no início do processo, economizando recursos dos filtros seguintes. A Figura 5.20 apresenta os tempos de processamento dos filtros do Edge Core. No primeiro momento, foi verificado quanto tempo cada filtro demora para processar os dados, individualmente. Posteriormente, foi conduzido um novo teste³⁵ para averiguar o tempo de processamento quando um valor passa por mais de um filtro.

Figura 5.20: Tempos de processamento dos filtros do *Edge Core*



FONTE: O Autor (2021)

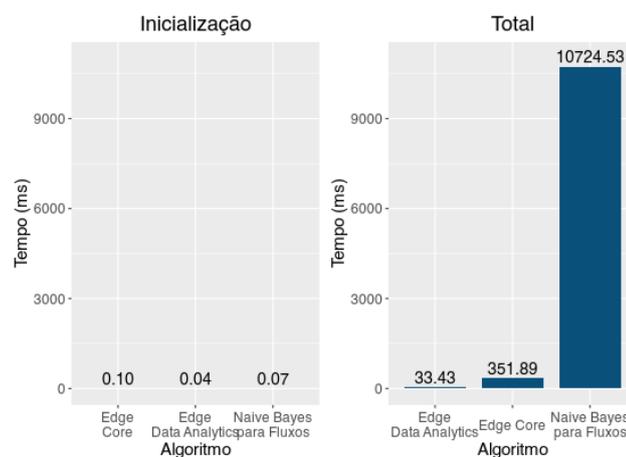
³⁵Esse teste foi conduzido porque alguns valores passam por todos os filtros, enquanto outros são marcados como anômalos antes de serem processados por todos os filtros do Edge Core

A hipótese inicial era de que os tempos seriam somados. Porém, parte do tempo de processamento se deve à recuperação dos dados armazenados em memória. Quando o processamento é feito por um único filtro, cada valor é recuperado da memória, analisado pelo filtro, marcado como normal ou anômalo e devolvido para o sistema. Isso significa que cada filtro irá recuperar o valor da memória no momento do processamento. Por outro lado, quando vários filtros processam o mesmo valor, este é recuperado e devolvido para a memória apenas uma vez, reduzindo o tempo total de processamento.

O filtro Escalas processa todos os registros que chegam ao detector de anomalias. Seu trabalho utiliza 344ms de CPU. Em seguida, os registros pré-classificados como “normal” são transferidos para o Limites. Enquanto o processamento através do Limites custa 345ms, o agrupamento Escalas e Limites custa 349ms. Em seguida, os dados são passados para os filtros de Deriva de Escala e Deriva de Padrões. O Deriva de Escala sozinho consome 346ms, mas quando agrupado aos filtros anteriores, ele acrescenta apenas 1ms ao tempo de processamento. O Desvio de padrões é o mais lento de todos os filtros individuais, demorando 348ms para processar, apenas 4ms a menos que o processamento completo. O processo inteiro em conjunto custa 352ms de processamento, justificando a organização dos filtros em conjunto.

A comparação entre os tempos de CPU usados pelo Edge Core, Edge Data Analytics e Naïve Bayes para fluxos de dados inclui o processamento por todos os filtros no EC. Na borda, o tempo total de classificação para um registro passando por todos os filtros é de cerca de 355ms. A Figura 5.21 resume os resultados. Na inicialização, o Edge Core utiliza mais ciclos de CPU, permanecendo cerca de 0,1ms no processador. Entretanto, a classificação é muito mais rápida do que a do Naive Bayes para Fluxos. Enquanto o Naive Bayes leva mais de dois segundos para classificar um milhão de valores, o Edge Core gasta um pouco mais de 35ms. Este desempenho torna o Edge Core viável para funcionar na maioria dos dispositivos de borda. O Edge Data Analytics realiza pouco processamento em todos os estágios, consumindo 0,044ms de CPU na inicialização e 33ms de tempo total. Isso se deve ao fato de que o EDA realiza comparações com dados pré-existentes sem realizar cálculos matemáticos.

Figura 5.21: Tempos de CPU gastos pelo *Edge Core*, *Edge Data Analytics* e *Naive Bayes* para Fluxos de Dados

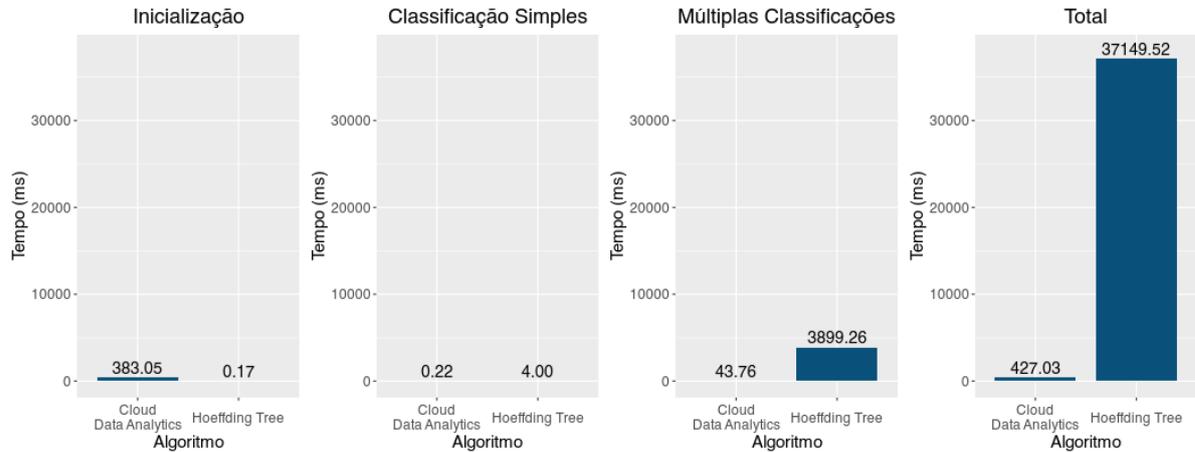


FONTE: O Autor (2021)

Na nuvem (Figura 5.22), o Cloud Data Analytics gasta 383ms de CPU na inicialização, enquanto as Árvore de Hoeffding gastam 0,165ms. Embora mais lento na inicialização, o CDA é mais rápido na classificação. O CDA demora 0,2ms para processar um único registro e 42ms para processar um conjunto de mil valores. As Árvore de Hoeffding demoram em média 4ms

para processar um único e 3,9s para mil registros. Ao todo, o CDA gasta 427ms de CPU e as Árvores de Hoeffding 37,1s.

Figura 5.22: Tempos de CPU gastos pelo *Cloud Data Analytics* e Árvores de Hoeffding



FONTE: O Autor (2021)

Quanto ao armazenamento em disco, o detector ocupa aproximadamente 262KB na borda, sendo 40KB relativos ao código fonte e 222KB aos arquivos de dados. Os valores não incluem arquivos de biblioteca e bancos de dados SQLite. Os baixos requisitos de armazenamento permitem a execução do CEIFA em dispositivos com baixa disponibilidade de armazenamento. Na nuvem, o CEIFA usa cerca de 13KB de arquivos executáveis e 2MB em outros arquivos.

A diferença no consumo de recursos do CDA e do CML é notável. A implementação do CDA usada para análise de recursos consome mais memória, mas economiza processamento. Portanto, é fundamental analisar rigorosamente a disponibilidade de recursos computacionais e os custos de processamento, armazenamento e memória para definir a melhor maneira de estruturar o detector de anomalias.

5.3 RESULTADOS

O desempenho do detector foi avaliado considerando o padrão de dados gerados pelos sensores instalados em ambiente de testes. Esses dados foram comparados com informações meteorológicas fornecidas pelo Climatempo e AccuWeather e observou-se que os valores registrados pelos sensores variaram em até 10% com relação às previsões meteorológicas. A Tabela 5.6 apresenta alguns registros dos sensores no dia 16 de fevereiro de 2021. Neste dia, os provedores registraram temperaturas entre 15°C e 25°C.

Parte da diferença entre os registros e as previsões se devem ao fato de que alguns dos dispositivos (sensores de umidade e temperatura) não foram corretamente calibrados antes de serem instalados. Outros sensores (umidade do solo) entraram em estado precoce de degradação. O local de instalação do sensor também contribui para as diferenças. Regiões distintas de uma mesma cidade apresentam temperaturas diferentes, conforme a quantidade de vegetação, construções, presença de rios ou lagos. Dependendo do horário do dia, um sensor pode estar mais exposto ao vento, ao sol ou outras condições climáticas e ambientais. Sensores de umidade do solo

³⁵Registro histórico disponível em: <https://www.accuweather.com/en/br/pinheiro-preto/41234/february-weather/41234?year=2021>.

Tabela 5.6: Temperaturas (°C) registradas no dia 16 de fevereiro de 2021

Horário	Sensor 1	Sensor 2	Sensor 3	Sensor 4	Sensor 5
10:00	23,9	24,3	24,1	23,7	24,2
12:00	25,9	24,4	26,2	25,9	23,1
15:00	27,4	27,2	27,5	27,6	25,1
18:00	26,6	27,0	26,9	26,9	25,0
23:00	24,6	24,7	24,8	24,6	23,6

também podem apresentar dados distintos devido às variações das características físico-químicas do terreno.

Os sensores instalados em 2020 sofreram processo de degradação e passaram por uma revisão técnica para corrigir as falhas e substituir alguns componentes. Antes de serem reinstalados, todos os sensores foram calibrados. Apesar de gerarem dados equivalentes em laboratório, no ambiente de testes os registros dos sensores variaram em até 2%. Comparando com as previsões meteorológicas do Climatempo, a variação ficou em torno dos 3%.

Durante o período de coleta de dados, foram geradas quantidades significativas de informações. Foram mais de mil registros de temperatura em um único dia. Considerando a coleta de 4 parâmetros diferentes (umidade relativa do ar, temperatura, pressão atmosférica e umidade do solo), são mais de 120 mil registros em um mês. A classificação manual dessas informações é morosa e propensa a erros. Além disso, não ocorrem falhas em quantidade suficiente para os testes e algumas falhas, como danificação, não puderam ser observadas. Adicionalmente, alguns sensores entraram em processo de degradação precoce, pois alguns componentes não foram substituídos, o que reduziu sensivelmente a quantidade de dispositivos disponíveis e inviabilizou o progresso dos testes com dispositivos reais. Por isso, decidiu-se criar sensores virtuais que se comportam de forma semelhante aos reais.

A utilização de sensores virtuais permitiu ampliar a quantidade de dispositivos e de dados, reduzir o tempo e aumentar a precisão da classificação. Foram modelados 17 sensores. Como o sistema precisa que pelo menos 50% dos dispositivos estejam funcionando corretamente, 10 desses sensores foram modelados para enviar dados iguais ou com uma variação de até 3% com relação às previsões do Climatempo. Esse valor foi escolhido por ser a variação observada nos sensores reais. Para viabilizar a ocorrência de anomalias, foi modelado um sensor para cada anomalia, sendo:

- 1 sensor que apresenta falhas aleatórias;
- 1 sensor que entra em estado de saturação;
- 1 sensor que entra em estado de degradação;
- 2 sensores danificados, sendo que um simula uma danificação abrupta e um gradual;
- 1 sensor que, periodicamente, envia ruídos;
- 1 sensor que envia dados falsos;

As falhas aleatórias, saturação e degradação foram modeladas baseadas nos dados dos sensores reais. As falhas aleatórias são eventos esporádicos nos sensores reais, nos virtuais, esse evento ocorre em tempos aleatórios. A saturação ocorre quando a umidade relativa do ar alcança 99.9%. Para isso, o tempo deve estar chuvoso. Como no período dos testes ocorreram poucas chuvas e a umidade raramente alcançava os 99.9%, optou-se por simular a saturação quando

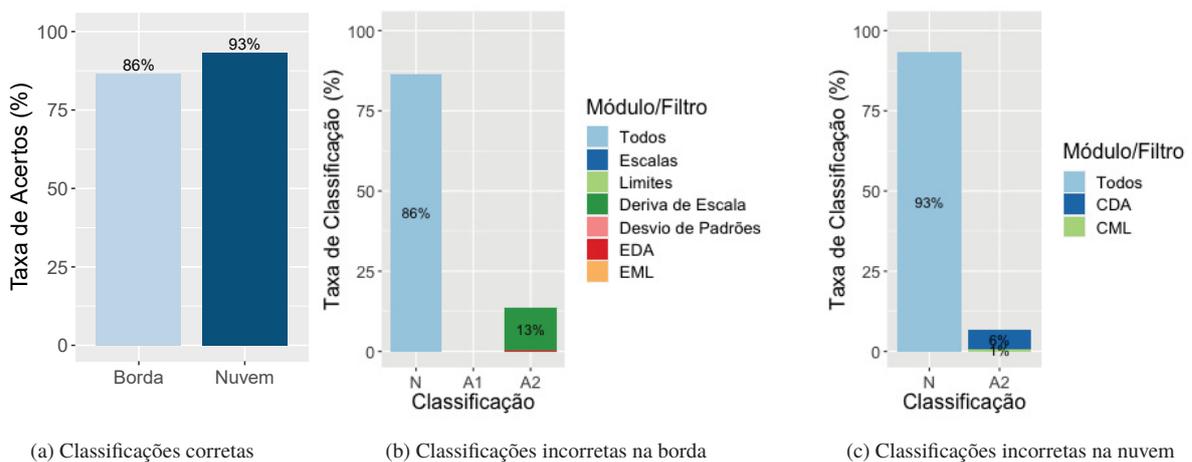
a umidade chegava a 99,4%³⁶. A degradação foi alcançada de forma gradual a partir de uma alteração lenta e progressiva na escala.

Para a danificação foram utilizados dois padrões: uma abrupta e outra gradual. A danificação gradual foi simulada alterando a escala a partir do centésimo registro. Esse valor foi escolhido para que o sensor tenha vários registros relativos ao seu funcionamento correto e depois inicie um processo gradual de degradação. Já o dispositivo danificado abruptamente passou a enviar o valor zero constantemente a partir do centésimo registro. Para os ruídos optou-se por fazer alterações nos *bits*, como se os dados enviados pela rede fossem perturbados. A quantidade de *bits* perturbados foi aleatória (de 1 a 5 *bits*), assim como a escolha de quais *bits* foram perturbados.

Para os dados falsos, foi simulada uma manipulação intencional em que um agente malicioso diminui em 25% a temperatura e mantém a umidade relativa do ar alta, quando a temperatura for superior a 15°C. O objetivo foi simular a situação em que o agente tenta impedir o acionamento de dispositivos de controle de temperatura. Nesse cenário, quando a temperatura excede os 15°C, todos os registros de temperatura são reduzidos. A partir da modelagem e implementação dos sensores, iniciaram os testes finais.

A Figura 5.23(a) mostra o resultado da classificação dos dados normais pelo CEIFA. De todos os dados normais enviados pelos sensores, a borda classificou corretamente 86,5% e a nuvem 93,3%. Na borda (Figura 5.23(b)), 13,1% dos registros foram classificados incorretamente como *deriva de escala* e 0,04% como *deriva de padrão*. O EDA classificou erroneamente 0,39% dos registros como uma anomalia. O EML e os filtros *Limites* e *Escalas* não classificaram nenhum registro incorretamente. Na nuvem (Figura 5.23(c)), o CDA classificou 5,92% dos dados corretos como uma anomalia e o CML falhou na classificação de 0,75% dos registros.

Figura 5.23: Classificação dos dados Normais

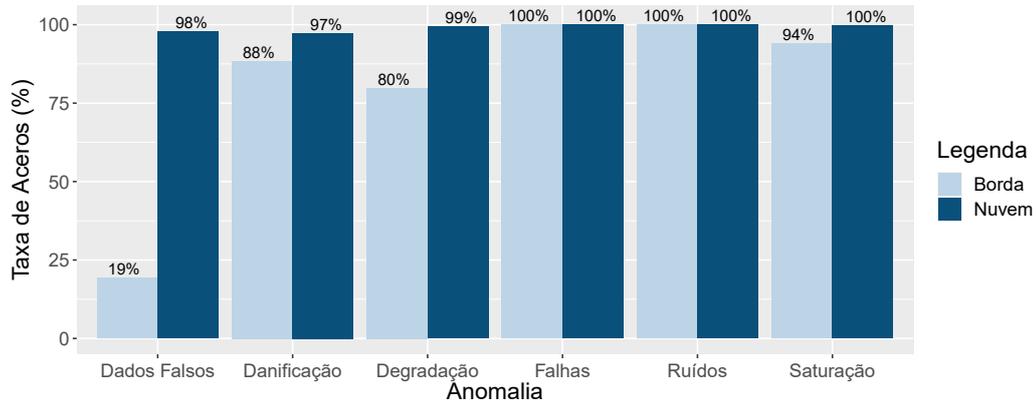


FONTE: O Autor (2021)

A Figura 5.24 apresenta os índices de detecção para cada anomalia. O CEIFA detectou todas as falhas aleatórias e ruídos enviados pelos sensores virtuais. A borda detectou 94% dos registros de saturação, 88% dos registros de danificação, 74% das degradações e 17% dos dados falsos. A nuvem, identificou 99% dos registros de saturação, 99% das degradações, 98% dos dados falsos e 97% das danificações.

³⁶Esse valor foi escolhido para ampliar as possibilidades de que a falha ocorreria.

Figura 5.24: Dados anômalos classificados como anomalia pelo CEIFA

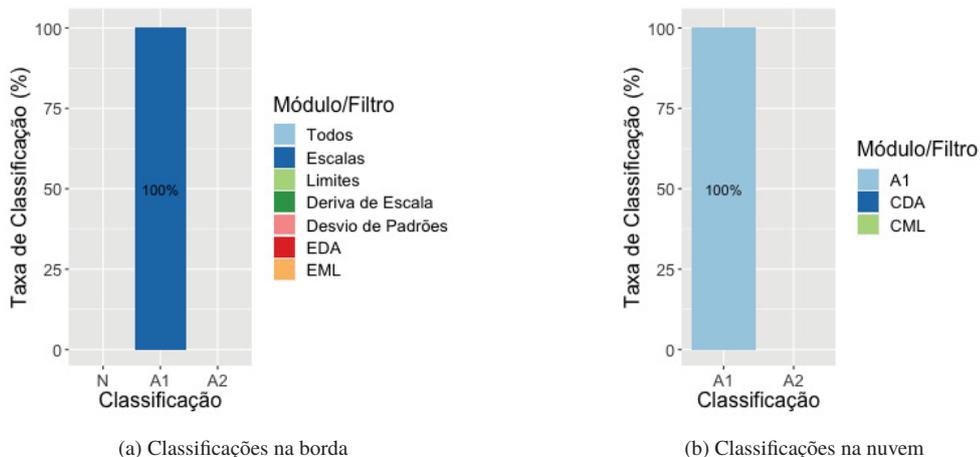


FONTE: O Autor (2021)

O conjunto de dados continha 74% de dados normais e 26% de anomalias. A borda classificou 70,1% de todos os dados como normais, 13% como anomalias do tipo “A1”, e 17% como anomalias do tipo “A2”. O detector retém anomalias do tipo A1 e aciona um alarme de notificação. Do total de 1950KB em dados anômalos, 1048KB foram retidos na borda, economizando 46% na transmissão de dados anômalos. Além da economia de largura de banda, a retenção de dados na borda economiza processamento e armazenamento na nuvem.

A figura anterior mostra que o CEIFA foi eficaz na identificação das falhas aleatórias, detectando todas as ocorrências. O primeiro filtro do Edge Core (*Escalas*), foi o responsável pela identificação da anomalia (Figura 5.25(a)). Esse filtro marcou todas as ocorrências da anomalia como “A1”. Ao receber essa marcação, a nuvem (Figura 5.25(b)) considera os registros como anomalia e não realiza nenhum processamento adicional. Os módulos da nuvem processam apenas os dados recebidos com a marcação “A2”. Desta forma, o CEIFA reduz o consumo de recursos computacionais.

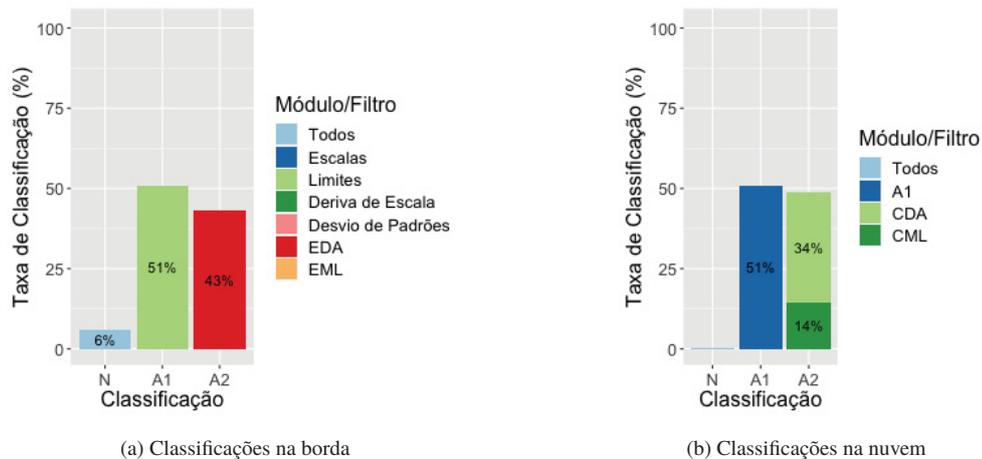
Figura 5.25: Classificação das Falhas Aleatórias



FONTE: O Autor (2021)

O CEIFA também foi bastante eficiente na detecção de eventos de saturação. A borda (Figura 5.26(a)) identificou 94,1% dessas falhas, sendo 51% detectadas pelo filtro *Limites* e 43% pelo EDA. O filtro *Limites* começa a identificar as anomalias a partir de uma certa quantidade de registros recorrentes³⁷, portanto, não marca como anomalia as primeiras ocorrências de saturação. O EDA tem uma tolerância³⁸ para medições que divergem da previsão do tempo. Esta característica faz com que ele marque a saturação como normal quando a umidade está próxima ao valor registrado pelo sensor. Na nuvem (Figura 5.26(b)), o CDA detectou 54,1% de saturação. O CML detectou 44,8% das anomalias restantes, marcando apenas 1,01% dos registros de saturação como normal.

Figura 5.26: Classificação das Saturações



FONTE: O Autor (2021)

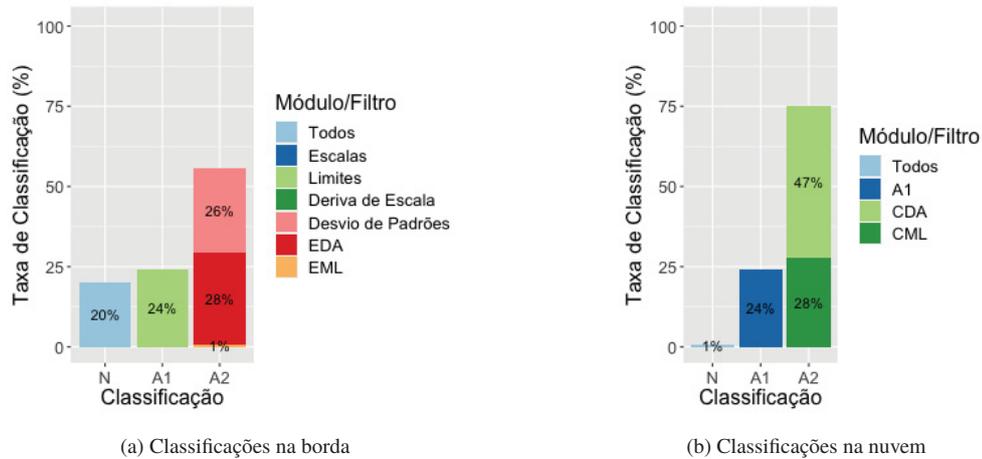
Os eventos de degradação causam falhas no sensor de modo que os valores registrados pelo dispositivo desviam da escala inicial. Pequenas alterações na escala não são incomuns. A documentação dos sensores utilizados informa que os dispositivos podem registrar valores com até 0,3% de erro (Adafruit, 2016). Considerando esse percentual, o CEIFA aponta como anomalia os dados divergem mais que 0,3% do valor considerado correto pelo sistema. De todos os eventos registrados no sistema, a borda (Figura 5.27(a)) detectou quase 80% das degradações. Dessas, 24,1% foram identificadas pelo filtro *Limites*, 26,5% pelo filtro *Desvio de Padrões*, 28,5% pelo EDA e 0,8% pelo EML. Os eventos não detectados pela borda são aqueles registrados no início do processo de degradação, em que os dados estão muito próximos dos valores registrados por sensores não degradados. Na nuvem (Figura 5.27(b)), 24,1% dos registros chegaram marcados como anomalia “A1” e não foram filtrados novamente. O CDA foi preciso em 47,4% dos registros. Isso se deve ao uso de previsões em tempo real. O CML capturou corretamente outros 27,8%. Assim, o CEIFA falhou em apenas 0,67% dos casos.

Quanto à verificação de danificação (Figura 5.28), o EC detectou 74,1% dessas anomalias (16,3% identificadas pelo filtro *Escala*, 50% pelo *Limites* e 10,8% pelo *Deriva de Escala*). O EDA indicou quase 10,4% registros como anômalos, e o EML 0,93%. A borda marcou 66,3% desses registros como “A1”. Esses registros não são processados pelos classificadores da nuvem. O CDA detectou 22,4% dos casos de danificação e o CML 8,63%. O CEIFA não identificou 2,74% dos registros de danos.

³⁷A quantidade pode ser configurada. Para os testes, foram considerados 15 registros recorrentes.

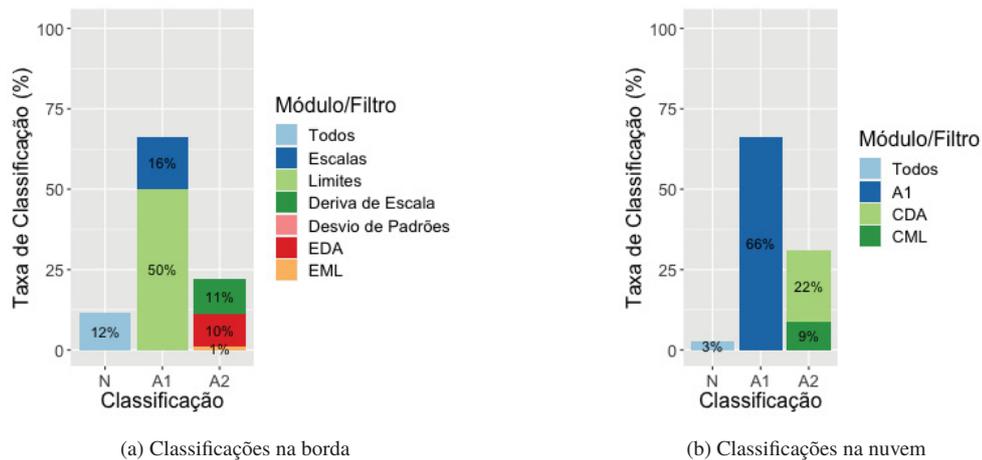
³⁸Os testes consideraram até 5° de tolerância para a temperatura e até 7% para a umidade relativa do ar.

Figura 5.27: Classificação das Degradações



FONTE: O Autor (2021)

Figura 5.28: Classificação das Danificações

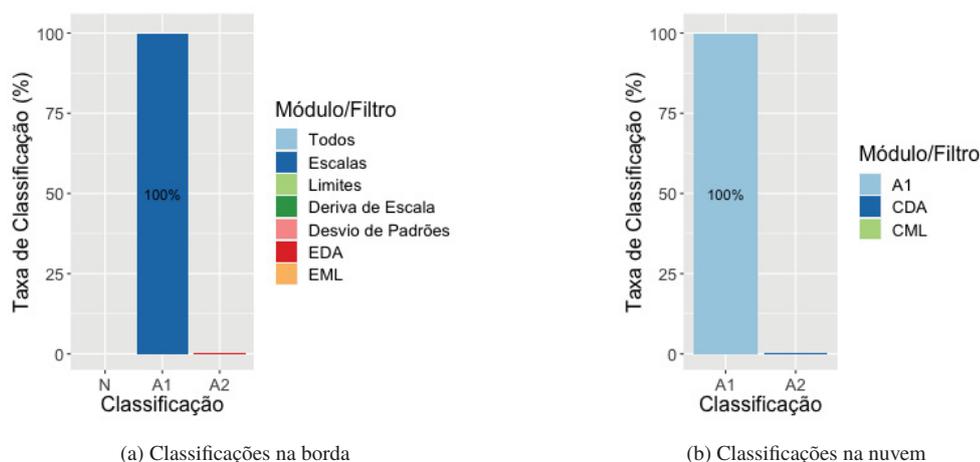


FONTE: O Autor (2021)

Os ruídos são eventos que perturbam os dados enquanto a transmissão está em andamento. Nos testes de desempenho, o distúrbio alterou um conjunto aleatório de *bits*, cuja quantidade variou entre 1 e 5 *bits*. A alteração considerou apenas os dados, visto que o CEIFA não analisa cabeçalhos de rede. A perturbação gera uma mudança significativa nos dados, tornando-os compatíveis com valores fora de escala. Por exemplo, um valor 250 em binário equivale a 11111010. Se esse conjunto sofrer a alteração de um *bit* (10111010), o valor em binário passa a ser 186. Quando são alterados 4 *bits* (10000010), o valor em decimal é 130. Por esta razão, o filtro *Escalas* (Figura 5.29(a)) foi eficiente na detecção de 99,8% das ocorrências de ruído e o *Deriva de Escala* em 0,2%. Na nuvem (Figura 5.29(b)), o CDA identificou outros 0,23% dos eventos.

A injeção de dados falsos ocorre de várias maneiras. A avaliação de desempenho considerou dois cenários: tentativas de *i*) evitar a ativação de atuadores para resfriar o ambiente e *ii*) manter o sistema de irrigação constantemente inundando a cultura. O primeiro cenário assume a ativação de atuadores em altas temperaturas e baixa umidade relativa do ar. Um oponente (ou

Figura 5.29: Classificação dos Ruídos

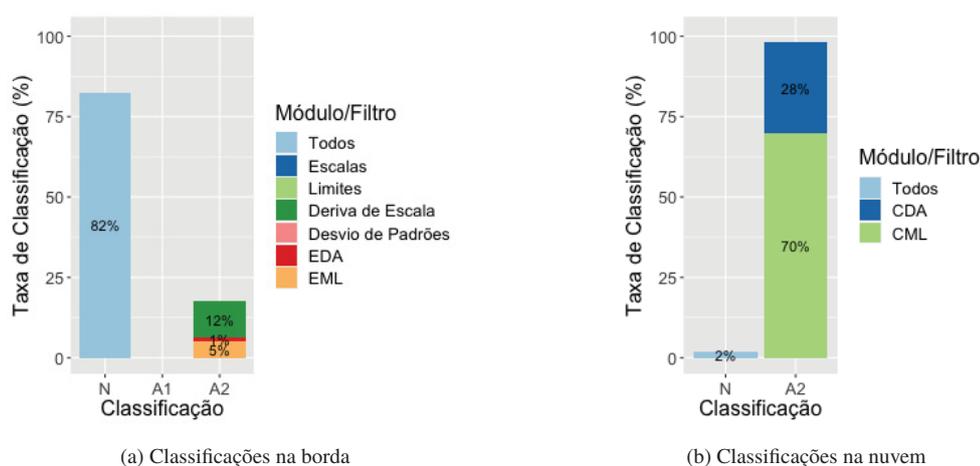


FONTE: O Autor (2021)

ciberterrorista) compromete um sensor e o reconfigura para enviar valores abaixo de 15°C de temperatura e acima de 50% de umidade. Quando a umidade e a temperatura são compatíveis com as definidas pelo terrorista, os dados permanecem inalterados. Se a temperatura exceder 15°C e a umidade for inferior a 50%, o nó malicioso reduz a temperatura em 10% e mantém a umidade próxima a 50%³⁹.

Os resultados mostram que CEIFA pode detectar cerca de 17,7% das ocorrências na borda e 98,1% na nuvem. Na borda, Figura 5.30(a), o filtro *Deriva de Escala* conseguiu identificar 11,5% dos dados falsos e o EDA 0,9% e o EML 5,28%. A nuvem, Figura 5.30(b), foi mais eficaz na detecção de dados falsos. O CDA encontrou 28,5% das ocorrências, e o CML identificou 69,6%. Apenas 1,85% das ocorrências foram classificadas como dados normais.

Figura 5.30: Classificação de Injeção de Dados Falsos - Cenário 1



FONTE: O Autor (2021)

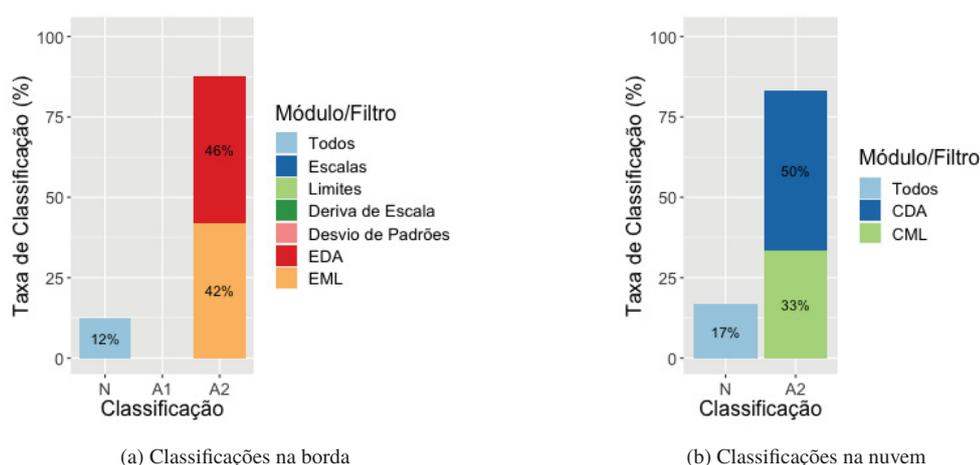
No segundo cenário, o objetivo do ciberagroterrorista é manter o sistema de irrigação inundando a plantação. Assume-se que o sistema funciona corretamente até que os dispositivos

³⁹O valor é aleatório, variando entre 50% e 60%.

sejam comprometidos. Portanto, há muitos dados corretos antes que o ataque inicie. Os sensores são comprometidos, um por dia, até a metade menos um dos sensores⁴⁰. Este teste utilizou 15 sensores. Os dispositivos comprometidos enviam valores de temperatura variando entre 25°C e 30°C, dados de umidade do ar abaixo de 50%, a precipitação a 0 (sem chuva) e a pressão atmosférica entre 1000hPa e 1030hPa⁴¹

Os resultados (Figura 5.31) mostram que a borda detecta com sucesso 87,8% dos dados falsos, enquanto a nuvem detecta 83,3%. Isso significa que aproximadamente 4,5% dos dados falsos identificados pela borda são classificados como normais na nuvem. A análise dos dados e os recursos de aprendizagem da máquina são responsáveis pela detecção. O módulo de análise de dados da nuvem é mais eficiente que o da borda. O EDA identificou 45,9% das anomalias e o CDA 50%. Por outro lado, o algoritmo de aprendizagem de máquina da borda identificou mais anomalias (45,9%) que o da nuvem (33%). Deve-se observar que o teste começou durante um período quente e seco, mas o tempo mudou muito rapidamente. No período final, o tempo estava frio e chuvoso. Esse comportamento resultou em erros de classificação por parte dos algoritmos de fluxo de dados usado na nuvem.

Figura 5.31: Classificação de Injeção de Dados Falsos - Cenário 2



FONTE: O Autor (2021)

Os dois experimentos de injeção de dados falsos têm características diferentes. No primeiro cenário, a manipulação de dados utiliza dados corretos como base. O experimento realizada quando as temperaturas estavam em torno de 12°C e 25°C, e a umidade geralmente se manteve acima de 50%. O segundo foi realizado em um período em que as temperaturas variaram entre 25°C e 5°C. A faixa de temperatura da região onde os testes foram realizados é grande. No mesmo dia, é possível registrar temperaturas próximas a 10°C e 25°C. Portanto, os nós maliciosos utilizaram valores aleatórios, variando entre 25°C e 30°C, de acordo com a temperatura registrada no momento do ataque. A umidade do ar foi reduzida em 50% e a pressão atmosférica foi mantida entre 1000hPa e 1030hPa. Estas características climáticas são importantes e interferem no desempenho do detector de anomalias. As diferenças substanciais nos resultados apontam para a necessidade de testar diferentes formas de injeção de dados falsos sob condições climáticas variáveis.

⁴⁰No primeiro dia, o sistema tem um sensor comprometido, dois dispositivos comprometidos no segundo dia, e assim por diante, aumentando consecutivamente, até que no sétimo dia haja sete sensores comprometidos.

⁴¹A pressão atmosférica está relacionada à umidade do ar. Quanto mais seco o ar estiver, maior será a pressão atmosférica. Uma redução neste valor indica um aumento na umidade do ar, sinalizando a possibilidade de chuva.

5.3.1 Análise dos resultados

O resultado geral, apresentado na Tabela 5.7, mostra que o CEIFA consegue detectar 98,31% das anomalias. O Edge Core detectou 65% dos dados anômalos enviados para o sistema. Aproximadamente 52% desses registros foram detectados pelos filtros Limites e Escalas, evitando processamentos futuros, indicando economia de recursos computacionais. A borda detectou 80% dos dados anômalos, permitindo uma resposta rápida ao sistema. Cerca de 13% dos dados normais foram classificados como anômalos e 20% das anomalias foram classificadas como registros normais. Na nuvem, a detecção de quase 55% das anomalias pelo Cloud Data Analytics permite economizar recursos computacionais. Ao todo, a nuvem detectou 98% das anomalias, falhou na classificação de 1,6% dos dados anômalos e em 6,6% dos dados normais.

Tabela 5.7: Classificação dos dados anômalos pelos módulos do CEIFA

Anomalias	Borda				Nuvem		
	Edge Core	EDA	EML	Total	CDA	CML	Total
Falhas Aleatórias	100,00%	-	-	100,00%	-	-	100,00% ¹
Saturação	51,00%	43,00%	-	94,00%	54,10%	44,80%	98,90%
Degradação	50,60%	29,10%	0,20%	79,90%	71,50%	27,80%	99,33%
Danificação	77,10%	10,80%	0,50%	88,40%	72,40%	8,63%	97,30% ²
Ruídos	99,80%	0,20%	-	100,00%	0,20%	-	100,00% ³
Injeção de Dados Falsos	11,31%	7,60%	1,01%	19,37%	28,90%	68,90%	97,80%
Geral	65,17%	13,94%	0,69%	79,60%	54,74%	20,91%	98,31%⁴

¹ A nuvem recebeu 100% das falhas com a marca "A1", por isso elas foram marcadas como anomalia sem passar pelos classificadores da nuvem.

² A borda enviou 16,27% das anomalias com a marca "A1".

³ A nuvem recebeu 99,8% dos registros de ruídos com a marca "A1".

⁴ A nuvem recebeu 22,70% das anomalias com a marca "A1".

Apesar dos bons resultados no panorama, os resultados individuais não atingiram todas as expectativas. Esperava-se que o Edge Core detectasse mais dados de saturação e degradação, o que não foi observado nos testes e precisa ser investigado. Entretanto, os índices de detecção de ruídos e danificação superaram o esperado. Ademais, não havia perspectiva de que esse módulo identificasse injeção de dados falsos. O módulo que ficou aquém das expectativas foi o EML. Esperava-se que o Naïve Bayes-DS detectasse mais degradações, danificações e, especialmente, dados falsos, o que não ocorreu. Foram efetuados diversos ajustes visando melhorar o desempenho do algoritmo, porém o resultado não alcançou as expectativas. Novas investigações serão realizadas futuramente para otimizar o desempenho do classificador.

O CEIFA é um detector de anomalias adequado para a detecção das anomalias propostas, já que consegue detectar cerca de 98% das ocorrências. Ele pode economizar recursos através da filtragem dos dados na borda. A detecção realizada nos filtros Limites e Escalas, do Edge Core reduzem sensivelmente o consumo de recursos, já que eles consomem pouca memória e processamento. Juntos, os dois módulos conseguem detectar aproximadamente 22,7% das anomalias com baixo consumo de recursos. O EDA, apesar de consumir memória ligada à recuperação de informações meteorológicas, previne o consumo de processamento. A detecção realizada pelo CDA reduz o consumo de recursos significativamente. Se mais de 77% das anomalias são detectadas pelos módulos anteriores, apenas uma pequena parcela precisa ser processada pela aprendizagem de máquina. Desta forma, o CEIFA pode ser utilizado tanto por sistemas que dispõem de poucos recursos, como é o caso da agricultura familiar, quanto por sistemas com mais capacidades.

6 CONCLUSÃO

Atender aos 17 Objetivos para o Desenvolvimento Sustentável, estabelecidos pela ONU, e cumprir a Agenda 2030 tem se mostrado um desafio cada vez mais difícil de alcançar. Os percentuais de fome e subnutrição, que já eram preocupantes no período pré-pandêmico, pioraram durante a pandemia de COVID-19. Entre 2019 e 2020, cerca de 132 milhões de pessoas ingressaram na linha da fome (FAO, 2021). A agricultura sustentável está no centro da Agenda 2030 e é o primeiro passo para reverter esse cenário (ONU, 2015). Entretanto, os relatórios recentes publicados pela FAO mostram que o progresso no domínio da agricultura tem sido insuficiente. O mundo ainda conta com altos índices de desperdício de alimentos (FAO, 2019), enormes disparidades no fornecimento de recursos, tecnologias e na renda dos trabalhadores rurais (FAO, 2021, 2020). A solução para este problema passa por um melhor acesso a novas tecnologias agrícolas e recursos capazes de aumentar a produtividade e a renda do agricultor, especialmente nas áreas rurais mais vulneráveis economicamente.

Diante deste cenário, pesquisadores e a indústria ao redor do mundo tem se esforçado para criar tecnologias capazes de aumentar a produção e reduzir o consumo de recursos naturais. Essas tecnologias incorporam inovações da Indústria 4.0 e da IoT, e criam o que vem sendo chamado de *agricultura inteligente*. As inovações nesta área podem criar sistemas que ajudem a atender as demandas globais. Para isso, é preciso que as novas tecnologias agrícolas integrem recursos como agricultura de precisão, tecnologias de sensoriamento remoto, *big data*, computação em nuvem, ciência de dados, sistemas inteligentes e segurança cibernética (Trendov et al., 2019). Entretanto, a agricultura inteligente ainda inclui escassos recursos, especialmente no campo da segurança (Gupta et al., 2020; Window, 2019; Zanella et al., 2020).

A Agricultura 4.0 integra equipamentos e subsistemas de diferentes fornecedores. Muitos deles são de baixo custo e computacionalmente limitados. Atualmente, percebe-se a predominância dos sistemas de monitoramento, que incorporam poucas capacidades de gerenciamento. Mas a tendência é que esses sistemas sejam cada vez mais “inteligentes” e realizem processos automáticos, com pouca intervenção humana. A falta de características de segurança adequadas pode resultar em sistemas ineficientes, vulneráveis a anomalias e ciberataques, prejudicando a produtividade e causando perdas financeiras. Portanto, as soluções para este setor precisam ser seguras e confiáveis, sob pena de serem descartadas pelo mercado. Para maximizar a segurança, precisam haver controles de acesso, gerenciamento, armazenamento de informações, integridade de dados e confiabilidade. No que tange à confiabilidade, é preciso que o sistema funcione de forma correta, segura, eficiente e em tempo real.

No campo da cibersegurança, poucos recursos têm sido incorporados, sendo a maioria direcionado para troca de dados e controle de acesso. Além dos recursos serem insuficientes, os que são incluídos não têm todas as suas funcionalidades habilitadas, limitando as capacidades defensivas. Talvez isso se deva ao fato de que essas tecnologias ainda estão em suas fases iniciais de desenvolvimento e às restrições computacionais. Ademais, é preciso atentar para o agroterrorismo. Em tempos de grande tensão mundial e disputas financeiras, a agricultura pode ser utilizada para causar perturbações e manipular mercados. O que outrora era feito através de estratégias pouco automatizadas, agora pode utilizar sistemas digitais, ampliando o espectro de possibilidades e criando o ciberagroterrorismo. Este, pode ser potencialmente mais danoso, abrangente e difícil de detectar, tornando os controles de segurança ainda mais importantes.

No que se refere à confiabilidade, questões de dependabilidade podem desviar o sistema do funcionamento correto e eficiente. Falhas físicas ou lógicas, problemas na comunicação

e transmissão, e a ação de agentes externos podem violar esse requisito. A geração de dados em grande escala, com alta variabilidade na quantidade e velocidade, dificulta o manuseio de conjuntos de dados através de ferramentas e técnicas tradicionais, exigindo o uso de recursos que melhorem a qualidade das informações. A acuracidade dos dados e do próprio sistema pode ser afetada de forma ainda mais crítica pela ocorrência de falhas e erros em equipamentos físicos, o que é bastante comum quando se trata de sensores. Essas questões podem ser mitigadas pelo uso de sistemas de detecção de anomalias, que podem ser úteis especialmente para a identificação de comportamentos atípicos. Apesar desses comportamentos estarem associados a diversos fatores, a maioria dos detectores são especializados em ataques cibernéticos, abarcando apenas parte das ameaças que podem afetar a confiabilidade de um sistema agrícola.

Para contribuir para a melhoria da confiabilidade da agricultura digital, esta pesquisa propõe o CEIFA, um detector de anomalias projetado para identificar comportamentos atípicos e aumentar a confiabilidade. O CEIFA integra análise de dados, análise de tendências e dispersão com aprendizagem de máquinas para identificar falhas, erros e alguns ataques nos sensores utilizados pela agricultura inteligente. O detector utiliza uma arquitetura híbrida, incluindo operações na borda e na nuvem, para economizar recursos computacionais e financeiros. Ele detecta falhas aleatórias, saturação, degradação, danificação, alguns tipos de ruído e ataques de injeção de dados a partir da percepção.

O CEIFA foi estruturado em módulos distribuídos na borda e na nuvem. A borda possui três classificadores e um decisor. Apesar da estrutura prever três classificadores, apenas o primeiro é imprescindível, tornando o CEIFA um detector de anomalias genérico que pode ser utilizado em diferentes contextos a agricultura digital e nos sistemas agrícolas existentes. Em alguns casos não há recursos computacionais ou informações disponíveis para executar os processos. Os classificadores da borda podem operar isoladamente ou em conjunto com a nuvem, para otimizar o trabalho. O módulo decisor determina as ações adotadas pelo detector a partir da classificação atribuída.

A nuvem possui dois classificadores e um decisor. Os classificadores empregam análise de dados e aprendizagem de máquina, sendo similares aos da borda. Entretanto, a nuvem dispõe de fartos recursos computacionais, permitindo o uso de algoritmos e operações mais complexas. A análise de dados, por exemplo, pode utilizar informações mais atualizadas e consultar diferentes fontes de dados, e a aprendizagem de máquina pode executar algoritmos mais robustos. A decisão sobre quais classificadores incluir e sua organização é tomada durante o projeto do sistema e deve considerar questões como contexto de aplicação, tipos de dados, disponibilidade de informações e custos financeiros.

Os classificadores da borda possuem baixa latência e economizam recursos financeiros ligados à transmissão de dados e à computação em nuvem. Como a borda é o centro e processamento mais próximo à percepção, a detecção neste dispositivo permite uma resposta rápida e evita a utilização de registros anômalos. Os registros que forem retidos pelo detector não passam pelas linhas de transmissão, economizando banda e recursos ligados ao processamento e armazenamento dos dados. Entretanto, as técnicas empregadas na borda precisam ser compatíveis com as restrições dos dispositivos, que têm tipicamente baixa capacidade computacional. Por outro lado, a nuvem possui alta capacidade computacional, permitindo o uso de técnicas mais eficientes. O acesso a informações atualizadas e a possibilidade de correlacionar diferentes parâmetros amplia a precisão da análise e o desempenho do detector. Uma melhor relação entre custo e desempenho pode ser obtida usando todos os módulos da arquitetura proposta.

O detector proposto utiliza poucos recursos computacionais, o que lhe permite trabalhar com dispositivos computacionalmente limitados. O módulo Edge Core consome pouca memória e processamento, sendo compatível com a maioria dos equipamentos utilizados na borda. O

Edge Data Analytics, que realiza análise de dados, consome mais memória e processamento, mas agrega em precisão e eficiência. Para analisar o desempenho do Edge ML, foram testados vários algoritmos de aprendizagem de máquina e o Naïve Bayes para fluxos de dados foi o que apresentou melhor relação entre eficiência e custo computacional. Seu consumo de memória foi muito inferior ao do EDA, porém houve um consumo bastante superior de processamento. Ao todo, os módulos de borda consomem cerca de 400KB de memória, 262KB de espaço em disco, e demoram 355ms (tempo de CPU) para verificar a anomalia. Considerando todos os acessos à memória, incluindo operações de leitura e escrita nas memórias física e virtual, a borda consome 111MB de memória. Com esses recursos a borda conseguiu identificar 80% das anomalias alvo e mais de 86% dos dados normais foram classificados corretamente.

Na nuvem, o CEIFA utilizou mais recursos. O classificador Cloud Data Analytics consultou duas fontes de dados externas e coletou extensos conjuntos de dados. Isso resultou em um alto consumo de memória. Por outro lado, seu custo de processamento foi bastante baixo. O Cloud ML utilizou as Árvores de Hoeffding, o algoritmo com melhor desempenho entre todos os analisados. Esse algoritmo possui um consumo relativamente alto de processamento, quando comparado aos outros classificadores, mas apresenta altos índices de precisão. Ao todo, os objetos da nuvem consomem de 6MB de memória, sendo 120MB quando computadas todas as operações de leitura e escrita, e 2MB de espaço em disco. O tempo de processamento dos registros é de cerca de 4ms, mostrando um tempo de resposta baixo. O detector proposto apresentou um índice de erros inferior a 2%. A nuvem detectou mais de 98% das anomalias alvo e classificou sem erros 93% dos dados normais.

Portanto, o CEIFA é um detector de anomalias eficiente e de baixo custo. Sua operação na borda permite economizar recursos com alta eficiência e latência reduzida. A nuvem alcançou alto desempenho, sem um incremento importante no processamento. Isso permite que os módulos da nuvem sejam executados tanto em provedores de serviços de computação em nuvem quanto em dispositivos menos equipados, como é o caso de alguns dispositivos proprietários construídos especificamente para a agricultura inteligente. O detector é recomendado tanto para sistemas que dispõem de poucos recursos, como é o caso da agricultura familiar, como por aqueles que dispõem de mais recursos, como os utilizados pela agricultura em larga escala.

6.1 CONTRIBUIÇÕES

Este trabalho de doutorado contribuiu com o levantamento de informações sobre segurança na agricultura inteligente e com a especificação de um detector de anomalias. A seguir, são descritas as publicações realizadas nesse contexto:

- *Security challenges to smart agriculture: Current state, key issues, and future directions* (Zanella et al., 2020): esta pesquisa mostra o estado atual de segurança na agricultura inteligente e aponta oportunidades para melhoria da segurança nesse escopo.
- *CEIFA: a Multi-Level Anomaly Detector for Smart Farming* (Zanella et al., 2022): este trabalho apresenta a especificação do detector de anomalias híbrido denominado CEIFA.

6.2 LIMITAÇÕES

A pesquisa realizada apresenta limitações quanto à detecção de ataques e consumo de recursos. Os esforços se concentraram em criar uma estratégia para detecção de falhas e erros, pois era uma questão em aberto sobre a qual não foram encontrados trabalhos relacionados. Nos testes realizados, foi incluída a injeção de dados falsos para verificar a resposta do sistema a

ataques. Mas ataques como esse são complexos e precisam ser investigados com mais rigor. As ações maliciosas ligadas ao agroterrorismo não foram investigadas até o momento.

Embora o CEIFA tenha baixo consumo de recursos, os módulos que utilizam previsões meteorológicas utilizam muita memória. Isso resulta em gastos financeiros desnecessários. Os algoritmos de aprendizagem de máquinas utilizam muitos ciclos de CPU para treinamento. Isto pode inviabilizar o uso destes algoritmos por alguns sistemas. O sistema não foi testado em ambiente de produção, devido às limitações impostas pela pandemia. Restrições de acesso aos campos em que seriam realizados os testes, degradação de equipamentos e falta de equipamentos no mercado impediram a realização de alguns testes previstos anteriormente.

6.3 TRABALHOS FUTUROS

Os trabalhos futuros podem se concentrar em otimizar desempenho do CEIFA, reduzir o custo computacional e ampliar seu alcance. Uma possível estratégia é a inversão do processo de detecção. Até o momento, o objetivo foi identificar as anomalias no conjunto de dados. Ao inverter a estratégia, a meta será identificar os dados normais nos primeiros passos, reduzindo a latência do sistema e o consumo de recursos, uma vez que há significativamente mais dados normais do que dados anômalos.

O custo computacional pode ser otimizado pela redução do consumo de memória dos módulos que executam análise de dados (EDA e CDA). Atualmente eles carregam um extenso conjunto de dados em memória, resultando em desperdício. É preciso construir uma estratégia que carregue apenas os dados necessários para a execução das análises. O EML utiliza uma quantidade de processamento que pode não estar disponível na borda. É preciso criar estratégias que permitam o uso desses algoritmos por sistemas ainda mais restritivos.

O alcance do detector será incrementado incluindo a detecção de ataques maliciosos direcionados à agricultura inteligente (agroterrorismo) e ciberataques usando a agricultura inteligente como recurso intermediário. É preciso, também, pesquisar quais ações maliciosas surgem a partir da digitalização dos sistemas agrícolas e identificar potenciais contramedidas. Pesquisas sobre ciberagroterrorismo podem resultar em uma taxonomia para ataques que visam a Agricultura Inteligente (ciberataques). Um estudo futuro avaliará a escalabilidade de detecção e seu impacto sobre o consumo de recursos.

A análise de desempenho utilizou a precisão dos algoritmos como métrica. Futuramente pretende-se realizar um estudo utilizando a curva rock para verificar a eficácia do algoritmo. Além disso, pretende-se organizar e disponibilizar publicamente todos os dados utilizados nesta pesquisa. Atualmente não existem bases de dados públicas para agricultura inteligente, o que dificulta a pesquisa nessa área. Ao disponibilizar os dados, nós contribuimos com a comunidade científica e com o desenvolvimento de novas pesquisas.

REFERÊNCIAS

- Adafruit (2016). *Digital-output relative humidity and temperature sensor/module AM3202/DHT22*. Adafruit.
- Agrawal, S., Das, M. L. e Lopez, J. (2019). Detection of node capture attack in wireless sensor networks. *IEEE Systems Journal*, 13(1):238–247.
- Aguirre, L. A. (2013). *Fundamentos de Instrumentação*. Pearson Education do Brasil, São Paulo, 1 edition.
- Alaei, P. e Noorbehbahani, F. (2017). Incremental anomaly-based intrusion detection system using limited labeled data. Em *2017 3th International Conference on Web Research (ICWR)*, páginas 178–184.
- Alexandratos, N. e Bruinsma, J. (2012). World agriculture towards 2030/2050: the 2012 revision. Em *ESA Working Paper n. 12-03*. FAO, Roma.
- Ali, I., Sabir, S. e Ullah, Z. (2019). Internet of things security, device authentication and access control: A review. *CoRR*, abs/1901.07309.
- Almadhoun, R., Kadadha, M., Alhemeiri, M., Alshehhi, M. e Salah, K. (2018). A user authentication scheme of iot devices using blockchain-enabled fog nodes. Em *2018 IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA)*, páginas 1–8.
- Ariyaluran Habeeb, R. A., Nasaruddin, F., Gani, A., Targio Hashem, I. A., Ahmed, E. e Imran, M. (2019). Real-time big data processing for anomaly detection: A Survey. *International Journal of Information Management*, 45(February):289–307.
- Avizienis, A., Avizienis, A., Claude Laprie, J. e Randell, B. (2001). Fundamental concepts of dependability.
- Axelsson, S. (2015). *Intrusion Detection Systems: A Survey and Taxonomy*. Relatório Técnico April 2000, Department of Computer Engineering Chalmers University of Technology Göteborg.
- Benavides, E., Fuertes, W., Sanchez, S. e Sanchez, M. (2020). Classification of Phishing Attack Solutions by Employing Deep Learning Techniques: A Systematic Literature Review. Em *Developments and Advances in Defense and Security*, páginas 51–64, Singapore. Springer Singapore.
- Bezerra, L. G. S. (2015). Detecção, diagnóstico e correção de falha em sensores industriais foundation fieldbus utilizando agentes inteligentes. Dissertação de Mestrado, Universidade Federal do Rio Grande do Norte.
- Bhuyan, M. H., Bhattacharyya, D. K. e Kalita, J. K. (2014). Network anomaly detection: Methods, systems and tools. *IEEE Communications Surveys and Tutorials*, 16(1):303–336.

- Boghossian, A., Linksy, S., Brown, A., Mutschler, P., Ulicny, B., Barrett, L., Bethel, G., Matson, M., Strang, T. e Ramsdell, K. W. (2018). Threats to precision agriculture—homeland security. *A study supported by the United States Office of the Director of National Intelligence (ODNI) and the Department of Homeland Security (DHS) October*, 3:2018.
- Bormann, C., Ersue, M. e Keranen, A. (2014). Terminology for constrained-node networks. RFC 7228, RFC Editor. <http://www.rfc-editor.org/rfc/rfc7228.txt>.
- Bostami, B., Ahmed, M. e Choudhury, S. (2019). False Data Injection Attacks in Internet of Things. Em *EAI/Springer Innovations in Communication and Computing*, capítulo: Performabi, páginas 47–58. Springer, Cham, eai/springer edition.
- Capellupo, M., Liranzo, J., Bhuiyan, M. Z. A., Hayajneh, T. e Wang, G. (2017). Security and Attack Vector Analysis of IoT Devices. Em *Security, Privacy, and Anonymity in Computation, Communication, and Storage*, páginas 593–606, Cham. Springer, Cham.
- Catraca Livre (2020). Sementes misteriosas da China possuem ácaro, bactérias e pragas. Disponível em: <https://catracalivre.com.br/cidadania/sementes-misteriosas-da-china-possuem-acaro-bacterias-e-pragas/>. Acessado em 25/05/2021.
- Cervantes, C., Poplade, D., Nogueira, M. e Santos, A. (2015). Detection of sinkhole attacks for supporting secure routing on 6LoWPAN for Internet of Things. Em *Proceedings of the 2015 IFIP/IEEE International Symposium on Integrated Network Management, IM 2015*, páginas 606–611. IEEE.
- Chahid, Y., Benabdellah, M. e Azizi, A. (2017). Internet of things security. Em *2017 International Conference on Wireless Technologies, Embedded and Intelligent Systems, WITS 2017*, páginas 1–6, Fez, Morocco. IEEE.
- Chandola, V., Banerjee, A. e Kumar, V. (2009). Anomaly Detection: A survey. *ACM Computing Survey*, 41(3):1–58.
- Coelho, P. (2020). Sementes misteriosas: O que foi divulgado sobre o envio de pacotes da Ásia. *Aventuras na História*. Disponível em: <https://aventurasnahistoria.uol.com.br/noticias/almanaque/sementes-misteriosas-o-que-foi-divulgado-sobre-o-envio-de-pacotes-da-asia.phtml>. Acessado em 25/05/2021.
- Colezea, M., Musat, G., Pop, F., Negru, C., Dumitrascu, A. e Mocanu, M. (2018). CLUEFARM: Integrated web-service platform for smart farms. *Computers and Electronics in Agriculture*, 154:134–154.
- Correio Brasiliense (2020). Sementes misteriosas da China chegam ao Brasil e autoridades emitem alerta. Disponível em: <https://www.correiobrasiliense.com.br/brasil/2020/09/4876023-sementes-misteriosas-da-china-chegam-ao-brasil-e-autoridades-emitem-alerta.html>. Acessado em 25/05/2021.
- Coulter, R. e Pan, L. (2018). Intelligent agents defending for an IoT world: A review. *Computers and Security*, 73:439–458.
- Dave, D. e Vashishtha, S. (2013). Efficient intrusion detection with knn classification and ds theory. Em *Proceedings of All India Seminar on Biomedical Engineering 2012 (AISOB 2012)*, páginas 173–188. Springer.

- de Souza, P. M., Fornazier, A., de Souza, H. M. e Ponciano, N. J. (2019). Diferenças regionais de tecnologia na agricultura familiar no Brasil. *Revista de Economia e Sociologia Rural*, 57(4):594–617.
- Debar, H., Dacier, M. e Wespi, A. (1999). Towards a taxonomy of intrusion-detection systems. *Computer Networks*, 31(8):805–822.
- Debar, H., Dacier, M. e Wespi, A. (2000). A revised taxonomy for intrusion-detection systems. *Annals of Telecommunications*, 55(7-8):361–378.
- Demestichas, K., Peppes, N. e Alexakis, T. (2020). Survey on Security Threats in Agricultural IoT and Smart Farming. *Sensors*, 20(22).
- Di Modica, G., Gulino, S. e Tomarchio, O. (2019). IoT fault management in cloud/fog environments. Em *ACM International Conference on the Internet of Things*, páginas 1–4, New York, USA. ACM.
- Donevski, M. e Zia, T. (2018). A survey of anomaly and automation from a cybersecurity perspective. Em *2018 IEEE Globecom Workshops (GC Wkshps)*, páginas 1–6, Abu Dhabi, United Arab Emirates. IEEE.
- Ehlers, E. (2017). *O que é agricultura sustentável*. editora Brasiliense, São Paulo, 1 edition.
- Fadele, A. A., Othman, M., Hashem, I. A. T., Yaqoob, I., Imran, M. e Shoaib, M. (2019). A novel countermeasure technique for reactive jamming attack in internet of things. *Multimedia Tools and Applications*, 78(21):29899–29920.
- FAO (2019). The State of Food and Agriculture: moving forward on food loss and waste reduction. Relatório técnico, FAO, Rome, Italy.
- FAO (2020). Agricultural Commodity Markets and Sustainable Development. Relatório técnico, FAO, Rome, Italy.
- FAO (2021). Tracking progress on food and agriculture-related SDG indicators 2021: A report on the indicators under FAO custodianship. Relatório técnico, FAO, Rome.
- Farooqi, A. H. e Khan, F. A. (2009). A survey of intrusion detection systems for wireless sensor networks. *International Journal of Ad Hoc and Ubiquitous Computing*, 56:234–241.
- Finnegan, J. e Brown, S. (2018). A comparative survey of LPWA networking. *CoRR*, abs/1802.04222.
- Gajek, S., Jensen, M., Liao, L. e Schwenk, J. (2009). Analysis of signature wrapping attacks and countermeasures. Em *2009 IEEE International Conference on Web Services, ICWS 2009*, páginas 575–582, Los Angeles, USA. IEEE.
- Garcia, L., Parra, L., Jimenez, J. M. e Lloret, J. (2020). IoT-Based Smart Irrigation Systems: An Overview on the Recent Trends on Sensors and IoT Systems for Irrigation in Precision Agriculture. *Sensors*, 20(4):1–48.
- García-Gil, D., Luengo, J., García, S. e Herrera, F. (2019). Enabling Smart Data: Noise filtering in Big Data classification. *Information Sciences*, 479(2019):135–152.

- García-Teodoro, P., Díaz-Verdejo, J., Maciá-Fernández, G. e Vázquez, E. (2009). Anomaly-based network intrusion detection: Techniques, systems and challenges. *Computers and Security*, 28(1-2):18–28.
- Goap, A., Sharma, D., Shukla, A. K. e Rama Krishna, C. (2018). An IoT based smart irrigation management system using Machine learning and open source technologies. *Computers and Electronics in Agriculture*, 155:41–49.
- Gope, P. e Sikdar, B. (2019). Lightweight and privacy-preserving two-factor authentication scheme for iot devices. *IEEE Internet of Things Journal*, 6(1):580–589.
- Goyal, M. e Dutta, M. (2018). Intrusion Detection of Wormhole Attack in IoT: A Review. Em *2018 International Conference on Circuits and Systems in Digital Enterprise Technology, ICCSDET 2018*, páginas 1–5, Kottayam, India. IEEE.
- Guarda, T., Augusto, M. F. e Lopes, I. (2019). The Art of Phishing. Em *Advances in Intelligent Systems and Computing*, páginas 683–690. Springer, Cham.
- Gündoğan, C., Kietzmann, P., Lenders, M., Petersen, H., Schmidt, T. C. e Wählisch, M. (2018). Ndn, coap, and mqtt: A comparative measurement study in the iot. Em *Proceedings of the 5th ACM Conference on Information-Centric Networking, ICN '18*, página 159–171, New York, NY, USA. ACM.
- Guo, J., Zhao, N., Yu, F. R., Zhang, S., Yang, Z. e Leung, V. C. M. (2017). Disrupting Anti-Jamming Interference Alignment Sensor Networks with Optimal Signal Design. *IEEE Sensors Letters*, 1(3):1–4.
- Gupta, M., Abdelsalam, M., Khorsandroo, S. e Mittal, S. (2020). Security and privacy in smart farming: Challenges and opportunities. *IEEE Access*, 8:34564–34584.
- Halme, L. R. e Bauer, R. K. (1995). Aint misbehaving: a taxonomy of anti-intrusion techniques. Em *Proceedings of the 18th national information systems security conference*, San Jose, USA.
- Hassija, V., Chamola, V., Saxena, V., Jain, D., Goyal, P. e Sikdar, B. (2019). A Survey on IoT Security: Application Areas, Security Threats, and Solution Architectures. *IEEE Access*, 7:82721–82743.
- He, W., Golla, M., Padhi, R., Ofek, J., Dürmuth, M., Fernandes, E. e Ur, B. (2018). Rethinking access control and authentication for the home internet of things (iot). Em *27th USENIX Security Symposium (USENIX Security 18)*, páginas 255–272, Baltimore, MD. USENIX Association.
- Hoppen, J. e Santos, M. (2018). O que é Data Analytics. Disponível em: <https://www.aquare.la/o-que-e-data-analytics/>. Acessado em 06/08/2021.
- Iqbal, S., Mat Kiah, M. L., Dhaghighi, B., Hussain, M., Khan, S., Khan, M. K. e Raymond Choo, K. K. (2016). On cloud security attacks: A taxonomy and intrusion detection and prevention as a service. *Journal of Network and Computer Applications*, 74:98–120.
- Janssen, M., van der Voort, H. e Wahyudi, A. (2017). Factors influencing big data decision-making quality. *Journal of Business Research*, 70:338–345.

- Keegan, N., Ji, S. Y., Chaudhary, A., Concolato, C., Yu, B. e Jeong, D. H. (2016). A survey of cloud-based network intrusion detection analysis. *Human-centric Computing and Information Sciences*, 6(1).
- Kernel development community (2022). The linux kernel documentation. Disponível em: <https://www.kernel.org/doc/html/latest/index.html>. Acessado em 10/10/2022.
- Khan, J. A. e Jain, N. (2016). Improving intrusion detection system based on knn and knn-ds with detection of u2r, r2l attack for network probe attack detection. *International journal of scientific research in science, engineering and technology*, 2(5):209–212.
- Khan, N., Abdullah, J. e Khan, A. S. (2017). Defending Malicious Script Attacks Using Machine Learning Classifiers. *Wireless Communications and Mobile Computing*, 2017:1–9.
- Khelifa, B., Amel, D., Amel, B., Mohamed, C. e Tarek, B. (2015). Smart irrigation using internet of things. Em *4th International Conference on Future Generation Communication Technology, FGCT 2015*, páginas 91–96, Luton, UK. IEEE.
- Khraisat, A., Gondal, I., Vamplew, P. e Kamruzzaman, J. (2019). Survey of intrusion detection systems: techniques, datasets and challenges. *Cybersecurity*, 2(1).
- Kolias, C., Kambourakis, G., Stavrou, A. e Jeffrey Voas (2017). DDoS in the IoT: Mirai and Other Botnets. *Computer*, 50(7):80–84.
- Krimmling, J. e Peter, S. (2014). Integration and evaluation of intrusion detection for coap in smart city applications. Em *2014 IEEE Conference on Communications and Network Security*, páginas 73–78, San Francisco, USA. IEEE.
- Kumar, S. A., Vealey, T. e Srivastava, H. (2016). Security in internet of things: Challenges, solutions and future directions. Em *Proceedings of the Annual Hawaii International Conference on System Sciences*, páginas 5772–5781, Koloa, HI, USA. IEEE.
- Lavric, A., Petrariu, A. I. e Popa, V. (2019). Sigfox communication protocol: The new era of iot? Em *2019 International Conference on Sensing and Instrumentation in IoT Era (ISSI)*, páginas 1–4.
- Lee, M., Kim, H. e Yoe, H. (2017). Intelligent environment management system for controlled horticulture. Em *4th NAFOSTED Conference on Information and Computer Science, NICS 2017 - Proceedings*, páginas 116–119, Hanoi; Vietnam. IEEE Inc.
- Lima, A. F., Silva, E. G. d. A. e Iwata, B. D. F. (2019). Agriculturas e agricultura familiar no Brasil: uma revisão de literatura. *Retratos de Assentamentos*, 22(1):50.
- Lin, J., Yu, W., Zhang, N., Yang, X., Zhang, H. e Zhao, W. (2017). A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications. *IEEE Internet of Things Journal*, 4(5):1125–1142.
- Liu, H., Han, D. e Li, D. (2020). Fabric-iot: A blockchain-based access control system in iot. *IEEE Access*, 8:18207–18218.
- Mahalakshmi, M. (2018). Distant Monitoring and Controlling of Solar Driven Irrigation System Through IoT. Em *National Power Engineering Conference (NPEC)*, páginas 1–5, Madurai, India. IEEE.

- Mahdavinejad, M. S., Rezvan, M., Barekatin, M., Adibi, P., Barnaghi, P. e Sheth, A. P. (2018). Machine learning for internet of things data analysis: a survey. *Digital Communications and Networks*, 4(3):161–175.
- Mamdouh, M., Elrukhsi, M. A. e Khattab, A. (2018). Securing the Internet of Things and Wireless Sensor Networks via Machine Learning: A Survey. Em *2018 International Conference on Computer and Applications, ICCA 2018*, número Section II, páginas 215–218, Beirut, Lebanon. IEEE.
- Mandal, S., Bera, B., Sutrala, A. K., Das, A. K., Choo, K.-K. R. e Park, Y. (2020). Certificateless-signcryption-based three-factor user access control scheme for iot environment. *IEEE Internet of Things Journal*, 7(4):3184–3197.
- Mazoyer, M. e Roudart, L. (2006). *A History of World Agriculture*. Earthscan, London, UK.
- Medeiros, J. P. d. (2009). Estudo e implementação de algoritmos inteligentes para detecção e classificação de falhas na medição de gás natural. Dissertação de Mestrado, Universidade Federal do Rio Grande do Norte.
- Mekala, M. S. e Viswanathan, P. (2017). A Survey: Smart agriculture IoT with cloud computing. Em *International Conference on Microelectronic Devices, Circuits and Systems, ICMDCS 2017*, páginas 1–7, Vellore, India. IEEE.
- Mekki, K., Bajic, E., Chaxel, F. e Meyer, F. (2018). Overview of cellular lpwan technologies for iot deployment: Sigfox, lorawan, and nb-iot. Em *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, páginas 197–202.
- Minh, Q. T., Phan, T. N., Takahashi, A., Thanh, T. T., Duy, S. N., Thanh, M. N. e Hong, C. N. (2017). A Cost-effective Smart Farming System with Knowledge Base. Em *8th International Symposium on Information and Communication Technology - SoICT 2017*, páginas 309–316, Nha Trang; Vietnam. ACM New York.
- Ministério da Agricultura, Pecuária e Abastecimento (2019). Agricultura Familiar. Disponível em: <https://www.gov.br/agricultura/pt-br/assuntos/agricultura-familiar/agricultura-familiar-1>. Acessado em 20/07/2019.
- Mitre Corporation (2022). Common vulnerabilities and exposures. Disponível em: <https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=iot>. Acessado em 04/08/2022.
- Mode, G. R., Calyam, P. e Hoque, K. A. (2020). False Data Injection Attacks in Internet of Things and Deep Learning enabled Predictive Analytics. Em *EEE/IFIP Network Operations and Management Symposium*, páginas 1–11, Budapest, Hungary. IEEE.
- Monke, J. (2007). Agroterrorism: Threats and Preparedness. Relatório técnico, Congressional Research Service.
- Musat, G. A., Colezea, M., Pop, F., Negru, C., Mocanu, M., Esposito, C. e Castiglione, A. (2018). Advanced services for efficient management of smart farms. *Journal of Parallel and Distributed Computing*, 116:3–17.

- Muteba, F., Djouani, K. e Olwal, T. (2019). A comparative Survey Study on LPWA IoT Technologies: Design, considerations, challenges and solutions. *Procedia Computer Science*, 155:636–641.
- Nachan, H., Poddar, D., Sarode, S., Kumhar, P. e Birla, S. (2021). Intrusion detection system: A survey. *International Journal of Engineering Research and Technology*, 10(5):1036–1047.
- Nageswara Rao, R. e Sridhar, B. (2018). IoT based smart crop-field monitoring and automation irrigation system. Em *2nd International Conference on Inventive Systems and Control, ICISC 2018*, páginas 478–483, Coimbatore, India. IEEE.
- Naik, N. (2017). Choice of effective messaging protocols for iot systems: Mqtt, coap, amqp and http. Em *2017 IEEE International Systems Engineering Symposium (ISSE)*, páginas 1–7.
- Natasha Werneck (2020). Sementes 'misteriosas' recebidas da China contêm fungos e pragas que não existem no Brasil. *Estado de Minas*. Disponível em: https://www.em.com.br/app/noticia/nacional/2020/11/27/interna_nacional,1214992/sementes-misteriosas-china-contem-fungos-pragas-nao-existem-no-brasil.shtml. Acessado em 25/05/2021.
- Navarro-Hellín, H., Martínez-del Rincon, J., Domingo-Miguel, R., Soto-Valles, F. e Torres-Sánchez, R. (2016). A decision support system for managing irrigation in agriculture. *Computers and Electronics in Agriculture*, 124:121–131.
- Navas, R. E., Boudier, H. L., Cuppens, N., Cuppens, F. e Papadopoulos, G. Z. (2018). Demo: Do Not Trust Your Neighbors! A Small IoT Platform Illustrating a Man-in-the-Middle Attack. Em *DHOC-NOW: International Conference on Ad Hoc Networks and Wireless*, número September, páginas 1–6, Saint-Malo, France. Springer, Cham.
- Norton, R. A. (2016). *Is Terrorism a Threat to the U. S. Food Industry?*
- Oliver, S. T., González-Pérez, A. e Guijarro, J. H. (2018). An IoT proposal for monitoring vineyards called SEnviro for agriculture. Em *Proceedings of the 8th International Conference on the Internet of Things - IOT '18*, páginas 1–4, Santa Barbara; United States. ACM New York.
- Olson, D. (2012). Agroterrorism: Threats to America's Economy and Food Supply. <https://www.food-safety.com/articles/5201-is-terrorism-a-threat-to-the-us-food-industry>.
- ONU (2015). United Nations General Assembly Resolutions. Relatório Técnico September, ONU, New York, USA.
- ONU (2019). *World population prospects 2019*. Número 141 em 1. United Nations, New York, 2019 edition.
- Ouaddah, A., Mousannif, H., Abou Elkalam, A. e Ait Ouahman, A. (2017). Access control in the Internet of Things: Big challenges and new opportunities. *Computer Networks*, 112:237–262.
- OWASP Foundation (2018). Owasp internet of things project. Disponível em: https://wiki.owasp.org/index.php/OWASP_Internet_of_Things_Project. Acessado em 04/08/2022.
- ÓZSVÁRI, L., KASZA, G. e LAKNER, Z. (2017). Historical And Economic Aspects Of Bioterrorism. Em *Management, organizations and society*, capítulo: 18, páginas 179–186. Agroinform.

- Palekar, A. B. (2017). acknowledge the importance for network intrusion detection. *International Journal of Innovative Research and Advanced Studies (IJIRAS)*, 4(2):173–177.
- Partra, L. e Rao, U. P. (2016). Internet of Things — Architecture, applications, security and other major challenges. Em *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, páginas 1201–1206, New Delhi, India. IEEE.
- Peternela, A. (2019). Segurança em sistemas agrícolas digitais. Informação verbal, 21 aug. 2019.
- Pundir, S., Wazid, M., Singh, D. P., Das, A. K., Rodrigues, J. J. e Park, Y. (2020). Designing efficient sinkhole attack detection mechanism in edge-based IoT deployment. *Sensors*, 20(5):1–27.
- Punithavathi, P., Geetha, S., Karuppiah, M., Islam, S. K. H., Hassan, M. M. e Choo, K.-K. R. (2019). A lightweight machine learning-based authentication framework for smart IoT devices. *Information Sciences*, 484:255–268.
- Raducu, I. G., Bojan, V. C., Pop, F., Mocanu, M. e Cristea, V. (2015). Real-Time Alert Service for Cyber-Infrastructure Environments. Em *10th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, 3PGCIC 2015*, páginas 296–303, Krakow; Poland. IEEE.
- Raheem, F. e Uwanthika, I. (2019). Big data analytics: Key technologies and challenges. Em *Annual International Research Conference*, páginas 157–164. Faculty of Management and Commerce, South Eastern University of Sri Lanka.
- Rajalakshmi, P. e Devi Mahalakshmi, S. (2016). IOT based crop-field monitoring and irrigation automation. Em *10th International Conference on Intelligent Systems and Control, ISCO 2016*, páginas 1–6, Coimbatore; India. IEEE.
- Ray, P. P. (2017). Internet of things for smart agriculture: Technologies, practices and future direction. *Journal of Ambient Intelligence and Smart Environments*, 9(4):395–420.
- Reynders, B. e Pollin, S. (2016). Chirp spread spectrum as a modulation technique for long range communication. Em *2016 Symposium on Communications and Vehicular Technologies (SCVT)*, páginas 1–5.
- Ruengtinnun, S., Phongsamsuan, S. e Sureeratanakorn, P. (2017). Applied internet of thing for smart hydroponic farming ecosystem (HFE). Em *10th International Conference on Ubi-Media Computing and Workshops with the 4th International Workshop on Advanced E-Learning and the 1st International Workshop on Multimedia and IoT: Networks, Systems and Applications (Ubi-Media 2017)*, páginas 1–4, Beach Road Pattaya; Thailand. IEEE Inc.
- Saari, M., bin Baharudin, A. M., Sillberg, P., Hyrynsalmi, S. e Yan, W. (2018). Lora — a survey of recent research trends. Em *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, páginas 0872–0877.
- SAKURAI, K. e Kim, T. (2008). A trend in ids researches. *Journal of Security Engineering*, 5(4):8.
- Sales, N., Remedios, O. e Arsenio, A. (2015). Wireless sensor and actuator system for smart irrigation on the cloud. Em *IEEE World Forum on Internet of Things, WF-IoT*, páginas 693–698, Milan; Italy. IEEE.

- Sanislav, T. e Miclea, L. (2012). Cyber-physical systems - Concept, challenges and research areas. *Control Engineering and Applied Informatics*, 14(2):28–33.
- Santos, L., Rabadao, C. e Goncalves, R. (2018). Intrusion detection systems in Internet of Things: A literature review. Em *13th Iberian Conference on Information Systems and Technologies (CISTI)*, páginas 1–7, Cáceres. IEEE.
- Sarkar, D., Bali, R. e Sharma, T. (2018). *Machine Learning Basics*, páginas 3–65. Apress, Berkeley, CA.
- Schulz, S., Schaller, A., Kohnhäuser, F. e Katzenbeisser, S. (2017). Boot Attestation: Secure Remote Reporting with Off-The-Shelf IoT Sensors. Em *Computer Security – ESORICS 2017*, páginas 437–455, Oslo, Norway. Springer, Cham.
- Semtech Corporation (2015). LoRa Modulation Basics. Relatório Técnico May, Semtech, Camarillo, CA.
- Semtech Corporation (2021). What Is LoRaWAN. Disponível em: <https://www.semtech.com/lora/lorawan-standard>. Acessado em 11/11/2021.
- Shurman, M. M., Khrais, R. M. e Yateem, A. A. (2019). IoT Denial-of-Service Attack Detection And Prevention Using Hybrid IDS. Em *International Arab Conference on Information Technology (ACIT)*, Al Ain, United Arab Emirates. IEEE.
- Silva, L. C. e. (2019). Aprendizado de máquina com treinamento continuado aplicado à previsão de demanda de curto prazo: o caso do restaurante universitário da universidade federal de uberlândia. Dissertação de Mestrado, Universidade Federal de Uberlândia.
- Sinha, R. S., Wei, Y. e Hwang, S.-H. (2017). A survey on LPWA technology: LoRa and NB-IoT. *ICT Express*, 3(1):14–21.
- Stojmenovic, I. e Wen, S. (2014). The Fog computing paradigm: Scenarios and security issues. Em *2014 Federated Conference on Computer Science and Information Systems, FedCSIS 2014*, páginas 1–8, Warsaw, Poland.
- Thorat, A., Kumari, S. e Valakunde, N. D. (2018). An IoT based smart solution for leaf disease detection. Em *International Conference on Big Data, IoT and Data Science, BID 2017*, páginas 193–198, Pune, India. IEEE.
- Trendov, N. M., Varas, S. e Zeng, M. (2019). Digital technologies in agriculture and rural areas - Status report. Relatório técnico, Nations, Food and Agriculture Organization of the United, Rome, Italy.
- Triantafyllou, A., Tsouros, D. C., Sarigiannidis, P. e Bibi, S. (2019). An architecture model for smart farming. Em *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, páginas 385–392.
- UNDESA (2019). World Population Prospects 2019: Highlights.
- United Nation (2017). Sustainable Development Goals. Acessado em 20/07/2019.

- UOL (2020a). Sementes misteriosas da China provocam alertas de autoridades americanas. Disponível em: <https://noticias.uol.com.br/internacional/ultimas-noticias/2020/07/27/sementes-misteriosas-da-china-provocam-alertas-de-autoridades-americanas.htm>. Acessado em 25/05/2021.
- UOL (2020b). Sementes recebidas por brasileiros contêm pragas inexistentes no país. Disponível em: <https://economia.uol.com.br/noticias/redacao/2020/11/26/sementes-recebidas-por-brasileiros-contem-pragas-inexistentes-no-pais.htm>. Acessado em 25/05/2021.
- Varga, P., Plosz, S., Soos, G. e Hegedus, C. (2017). Security threats and issues in automation IoT. Em *IEEE International Workshop on Factory Communication Systems - Proceedings, WFCS*, páginas 1–6, Trondheim, Norway. IEEE.
- Vasques, A. T. e Gondim, J. J. (2019). Amplified reflection DDoS attacks over iot mirrors: A saturation analysis. Em *WCNPS 2019 - Workshop on Communication Networks and Power Systems*, páginas 1–6, Brasília, Brasil. IEEE.
- Window, M. (2019). *Security in Precision Agriculture: Vulnerabilities and risks of agricultural systems*. Dissertation, Luleå University of Technology.
- Wongpatikaseree, K., Kanka, P. e Ratikan, A. (2018). Developing Smart Farm and Traceability System for Agricultural Products using IoT Technology. Em *IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS)*, páginas 180–184, Singapore, Singapore. IEEE.
- Xiao, L., Wan, X., Lu, X., Zhang, Y. e Wu, D. (2018). IoT Security Techniques Based on Machine Learning: How Do IoT Devices Use AI to Enhance Security? *IEEE Signal Processing Magazine*, 35(5):41–49.
- Yoon, C., Huh, M., Kang, S.-G., Park, J. e Lee, C. (2018). Implement Smart Farm with IoT Technology. Em *20th International Conference on Advanced Communication Technology (ICACT)*, páginas 749–752, Chuncheon-si Gangwon-do; Korea (South). IEEE.
- Zanella, A. R. d. A., da Silva, E. e Albin, L. C. P. (2020). Security challenges to smart agriculture: Current state, key issues, and future directions. *Array*, 8:100048.
- Zanella, A. R. d. A., da Silva, E. e Albin, L. C. P. (2022). CEIFA: A multi-level anomaly detector for smart farming. *Computers and Electronics in Agriculture*, 202:107279.
- Zarpelão, B. B., Miani, R. S., Kawakani, C. T. e de Alvarenga, S. C. (2017). A survey of intrusion detection in Internet of Things. *Journal of Network and Computer Applications*, 84:25–37.
- Zayas, A. D. e Merino, P. (2017). The 3gpp nb-iot system architecture for the internet of things. Em *2017 IEEE International Conference on Communications Workshops (ICC Workshops)*, páginas 277–282.
- Zhang, Q., Xu, L., Li, Y., Shi, W. e Cao, J. (2016). Edge Computing: Vision and Challenges. *IEEE Internet of Things Journal*, 3(5):637–646.
- Zhao, K. e Ge, L. (2013). A survey on the internet of things security. *2013 Ninth International Conference on Computational Intelligence and Security*, páginas 663–667.

- Zhao, W., Lin, S., Han, J., Xu, R. e Hou, L. (2017). Design and Implementation of a Smart Irrigation System for Improved Water-Energy Efficiency. Em *IEEE Globecom Workshops*, páginas 1–6, Singapore, Singapore. IEEE.
- Zhao, W., Yang, S. e Luo, X. (2020). On threat analysis of iot-based systems: A survey. Em *2020 IEEE International Conference on Smart Internet of Things (SmartIoT)*, páginas 205–212.
- Zhou, L. e Guo, H. (2018). Anomaly Detection Methods for IIoT Networks. Em *2018 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI)*, páginas 214–219, Singapore, Singapore. IEEE.
- Zuech, R., Khoshgoftaar, T. M. e Wald, R. (2015). Intrusion detection and Big Heterogeneous Data: a Survey. *Journal of Big Data*, 2(1):1–41.

APÊNDICE A – REDES LOW-POWER WIDE-AREA (LPWA)

Os recentes avanços promovidos pela Internet das Coisas e o crescente desenvolvimento de aplicações para este setor criaram a demanda por redes de comunicação sem fio escalonáveis e de longo alcance (Finnegan e Brown, 2018). Várias tecnologias podem ser utilizadas para estabelecer a comunicação entre dispositivos espalhados por uma extensa área geográfica (Muteba et al., 2019). Entre as mais populares estão as redes de Baixa Potência e Área Ampla (LPWA, do inglês *Low Power Wide Area*), termo que agrupa um conjunto de tecnologias que permitem comunicações em longas distâncias, com menor custo e melhor consumo de energia (Finnegan e Brown, 2018; Sinha et al., 2017).

A LPWA oferece melhor propagação de sinal para locais fechados e conexão entre dispositivos instalados a uma distância que pode superar dez quilômetros em alguns cenários. A maioria das tecnologias usa a banda sub-GHz, que oferece comunicação robusta e confiável com baixo consumo de energia (Muteba et al., 2019). Essas tecnologias são adequadas para aplicações que se comunicam em longas distâncias, transmitem pequenas quantidades de informação e possuem necessidades limitadas de latência (Muteba et al., 2019; Sinha et al., 2017). Diversas tecnologias LPWA tem sido desenvolvidas, tanto nos espectros licenciados quanto nos não licenciados. Entre as mais populares estão LoRA/LoRaWAN, NB-IoT e SigFox.

Long Range (LoRa) é uma técnica de modulação proprietária de amplo espectro derivada da tecnologia *Chirp Spread Spectrum* (CSS) (Semtech Corporation, 2015; Reynders e Pollin, 2016). Possui modulação escalável tanto em frequência quanto em largura de banda. É resistente a interferências, possui imunidade a múltiplos caminhos (*multipath*) e desvanecimento (*fading*), o que a torna adequada para áreas urbanas (Semtech Corporation, 2015; Sinha et al., 2017; Finnegan e Brown, 2018). A taxa máxima de transmissão de dados pode atingir os 50 kbps, dependendo do fator de propagação e da largura de banda do canal (Sinha et al., 2017). Mensagens transmitidas usando diferentes fatores de espalhamento podem ser recebidas ao mesmo tempo pela estação base (Finnegan e Brown, 2018).

Enquanto LoRa opera na camada física, LoRaWAN foi projetado para trabalhar na camada de rede, atuando como um protocolo de roteamento (Sinha et al., 2017; Muteba et al., 2019). Trata-se de um protocolo de comunicação aberto, baseado em LoRa e padronizado pela LoRa-Alliance em 2015. Projetado para conectar dispositivos sem fio alimentados por bateria à Internet, esse padrão atende a requisitos como comunicação bidirecional, segurança ponta a ponta, mobilidade e serviços de geolocalização. LoRaWAN opera no espectro de rádio não licenciado na banda Industrial, Científica e Médica (ISM, do inglês *Industrial, Scientific and Medical*) (Semtech Corporation, 2021). Esse protocolo gerencia a comunicação entre os dispositivos finais e a estação base da rede. Toda mensagem transmitida por um dispositivo final é recebida por todas as estações base que operam na mesma faixa (Muteba et al., 2019).

Para atender aos diversos requisitos das aplicações, LoRaWAN define três classes de dispositivos: *classe A*, que compreende aqueles alimentados por bateria e projetados para eficiência energética; *classe B*, englobando os equipamentos alimentados por bateria e projetados principalmente como atuadores; e *classe C*, que envolve os atuadores alimentados por uma fonte permanente. Os dispositivos de classe C usam mais energia para operar do que os de classe A e B, mas oferecem a menor latência para a comunicação do dispositivo final, e podem ouvir a rede continuamente (Muteba et al., 2019; Saari et al., 2018; Sinha et al., 2017).

NarrowBand-IoT (NB-IoT) é um protocolo de comunicação lançado pelo *Third Generation Partnership Project* (3GPP) em junho de 2016 para aquisição de dados por aplicações

IoT com baixa taxa de transmissão (Zayas e Merino, 2017; Muteba et al., 2019). Embora tenha sido construída a partir do padrão LTE, possui simplificações em diversos requisitos como processo de aquisição, largura de banda e acesso aleatório, o que permite reduzir os custos do dispositivo e minimizar o consumo da bateria (Finnegan e Brown, 2018; Sinha et al., 2017). O NB-IoT pode operar de três modos: (i) dentro da banda LTE (modo *in-band*), compartilhando a potência de transmissão com a banda larga LTE; (ii) de modo autônomo (*stand-alone*), geralmente utilizando frequências GSM; e (iii) na banda de guarda (*guard-band*), compartilhando uma célula LTE (Finnegan e Brown, 2018; Muteba et al., 2019). No primeiro modo, a largura de banda é compartilhada entre a LTE e o NB-IoT e ambos podem ser suportados pela mesma estação base, sem que o desempenho de qualquer uma delas seja comprometido. No segundo modo, utiliza toda potência de transmissão da estação base, o que aumenta a cobertura. Já na banda de guarda, ele compartilha o mesmo amplificador de potência do canal LTE e alcança a mesma potência de transmissão que a banda larga.

Sigfox é uma tecnologia proprietária que opera nas bandas ISM não licenciadas abaixo de 1 GHz e utiliza banda ultra estreita para transmitir pequenas quantidades de dados (Lavric et al., 2019). Utiliza o modelo de operadora em que os usuários compram o dispositivo final e adquirem assinaturas de um provedor de serviços, que gerencia a rede de *gateways* (Finnegan e Brown, 2018). Os dispositivos finais se conectam às estações base utilizando a modulação BPSK em uma faixa ultra estreita de 100Hz a uma taxa máxima transmissão de 100 bps. A banda ultra estreita no espectro sub-GHz possibilita um uso eficiente da banda de frequência, com níveis de ruído e consumo de energia muito baixos, alta sensibilidade do receptor e projeto de antena de baixo custo. (Mekki et al., 2018). A principal vantagem dessa tecnologia é sua resistência a interferências e colisões, que permite a cada sensor enviar pacotes de dados em três canais de comunicação em espaços de tempo selecionados aleatoriamente (Lavric et al., 2019).

Cada uma dessas tecnologias atende a diferentes requisitos e possui um lugar no mercado de IoT. SigFox e LoRA/LoRaWAN atendem a um mercado que exige menor custo do dispositivo e longo alcance do sinal. LoRaWAN também permite a implementação em redes locais e uma comunicação confiável entre dispositivos que se movimentam em alta velocidade (Mekki et al., 2018). Por outro lado, NB-IoT atende aplicações que exigem latência muito baixa e alta qualidade de serviço.

APÊNDICE B – RASPBERRY PI 2 MODEL B - HARDWARE GENERAL SPECIFICATIONS

Os dados foram obtidos em <https://raspberrypi.org/hardware/raspberry-pi-2-model-b/rpi2-model-b-hardware-general-specifications>.

Processador

Original (pre V1.1): Processador Broadcom BCM2836 ARMv7 Quad Core 32bit, rodando a 900MHz.

V1.2 PCB: Processador Broadcom BCM2837 Quad Core ARM Cortex-A53.

GPU é o mesmo que para o Raspberry Pi original:

- Capacidade de reprodução com qualidade BluRay, usando H.264 a 40MBits/s.
- Possui um núcleo 3D rápido, acessível usando as bibliotecas OpenGL ES2.0 e OpenVG disponíveis.
- Disponibiliza Open GL ES 2.0, OpenVG acelerado por hardware e decodificação de alto perfil 1080p30 H.264.
- Compatível com processadores de 1Gpixel/s, 1.5Gtexel/s ou 24 GFLOPs para computação de uso geral e possui um conjunto de filtragem de textura e infra-estrutura DMA.
- Os recursos gráficos são equivalentes ao nível de desempenho do Xbox 1.

Memória

1GB RAM

Conexões

4 portas USB

Porta Ethernet

Conector de 3,5mm para saída de áudio e vídeo combinado

HDMI

Energia

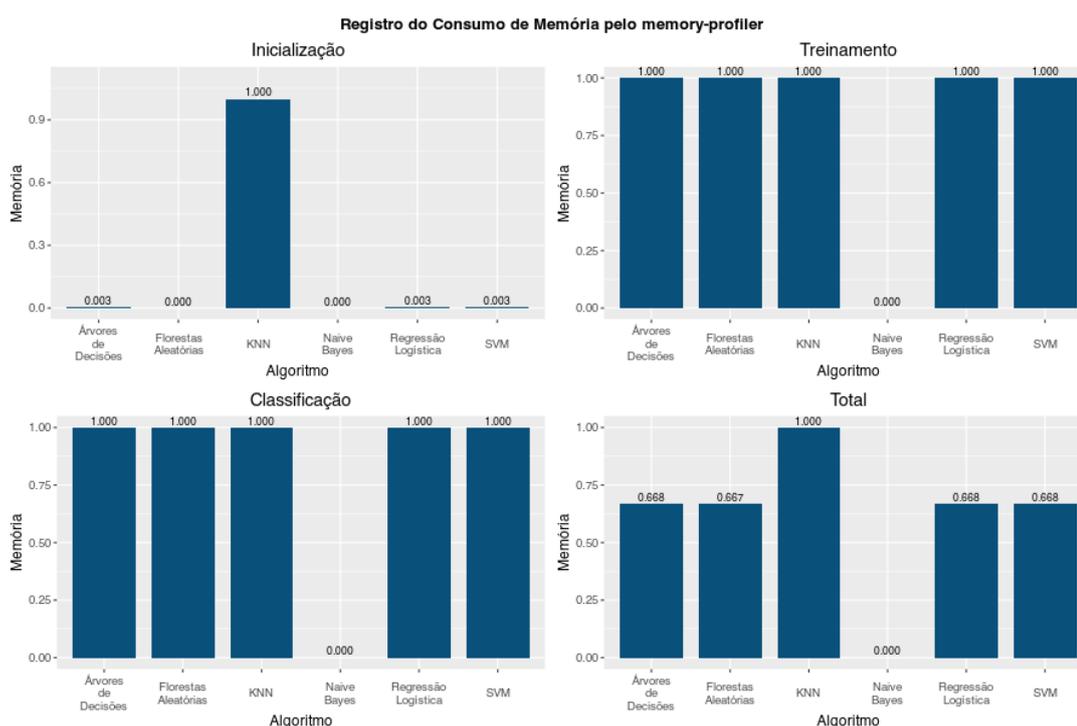
Mais poder de processamento = mais consumo de energia.

- Consumo máximo de potência original do Raspberry Pi Modelo B: aproximadamente 3 watts.
- Pico de consumo de energia da framboesa Pi Modelo B+: aprox. 2 watts.
- Pico de consumo de energia do Raspberry Pi 2 Modelo B: aprox. 3 watts (processador quad core com um aumento de 6x na potência de processamento).
- Raspberry Pi 2 Modelo B consumo de energia ociosa: aprox. o mesmo modelo original.

APÊNDICE C – ANÁLISE DOS ALGORITMOS DE APRENDIZAGEM DE MÁQUINA

A análise dos algoritmos de aprendizagem de máquina envolveu a verificação do consumo de recursos computacionais (memória e processamento) e da precisão para classificação dos dados. O consumo de memória foi medido utilizando as bibliotecas *memory-profiler*⁴² e *Guppy*⁴³. Visto que o objetivo era comparar o consumo de memória entre os algoritmos, foi realizada a normalização dos dados. Para registrar o total de memória alocada por cada algoritmo, o *memory-profiler* foi configurado para apresentar uma precisão de 15 casas decimais. Essa precisão foi escolhida porque alguns algoritmos apresentaram consumo de memória muito similar, havendo diferenças apenas em alguns dígitos finais. Analisando os registros, foram encontrados alguns dados discrepantes, por isso foi realizada a remoção de *outliers*⁴⁴. Os dados são resultantes de mais 100 execuções realizadas com intervalo mínimo de 10 minutos. Cada algoritmo foi testado individualmente, treinando e classificando os mesmos dados. O resultado foi sintetizado na Figura C.1.

Figura C.1: Memory-profiler: memória alocada pelos algoritmos de aprendizagem de máquina para série de dados



FONTE: O Autor (2021)

O *memory-profiler* registrou o consumo de memória *i)* no momento da inicialização do objeto, *ii)* durante o treinamento e *iii)* durante a classificação. A biblioteca indica que toda memória é alocada no momento da inicialização. Enquanto o KNN apresenta um consumo de

⁴²A descrição da biblioteca *memory-profiler* está disponível em: <https://pypi.org/project/memory-profiler/>.

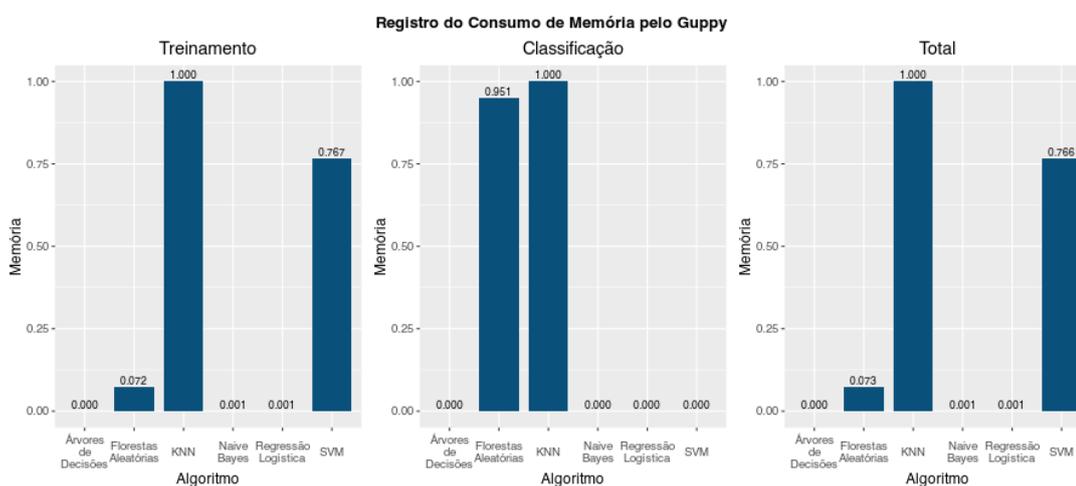
⁴³A descrição da biblioteca *Guppy* está disponível em: <https://pypi.org/project/guppy/>.

⁴⁴Foram considerados *outliers* o conjunto de até seis registros não compatíveis com a média.

memória excessivo, os demais algoritmos alocam uma quantidade similar memória⁴⁵. Durante a classificação e o treinamento não houve registro do consumo de memória adicional. Em alguns casos raros, houve o registro de liberação de memória. Esse comportamento foi mais intenso no algoritmo Naïve Bayes, que registrou liberação de memória em todas as execuções. Em algumas, a memória foi liberada durante o treinamento, em outras durante o teste.

Para checar possíveis divergências na alocação de memória, também foi utilizada a biblioteca Guppy. Os dados deste analisador foram obtidos medindo a memória alocada antes da inicialização do objeto, depois do treinamento e após a classificação dos dados, e subtraindo os valores obtidos. A memória alocada durante o treinamento e teste, bem como o total são apresentados na Figura C.2. Novamente o KNN é campeão na alocação de memória. No entanto, diferentemente da biblioteca anterior, o Guppy aponta um importante consumo por parte do algoritmo SVM. Durante a classificação dos dados, o KNN e as Florestas Aleatórias foram os que mais alocaram memória. No registro total, KNN e SVM são os maiores consumidores deste recurso.

Figura C.2: Guppy: memória alocada pelos algoritmos de aprendizagem de máquina para série de dados



FONTE: O Autor (2021)

Para verificar o processamento, foi utilizada a biblioteca Time⁴⁶. Antes de realizar os testes, os dados gerados pelo Time foram comparados com os da biblioteca cProfile. Os resultados são muito próximos, com a diferença de que o Time retorna mais casas decimais, o que motivou sua escolha. Os registros mostram o tempo de processamento, em segundos. Foram analisados o processamento decorrente dos processos de treinamento dos algoritmos e classificação dos dados, bem como o tempo total. Durante o treinamento o SVM registrou um consumo de memória muito superior aos demais. Durante a classificação, Florestas Aleatórias e SVM foram os que mais consumiram tempo de processador. Apesar desses dados serem importantes separadamente, eles não apresentam grande impacto no tempo de processamento total, cujos dados são influenciados principalmente pelo treinamento, devido ao volume de dados.

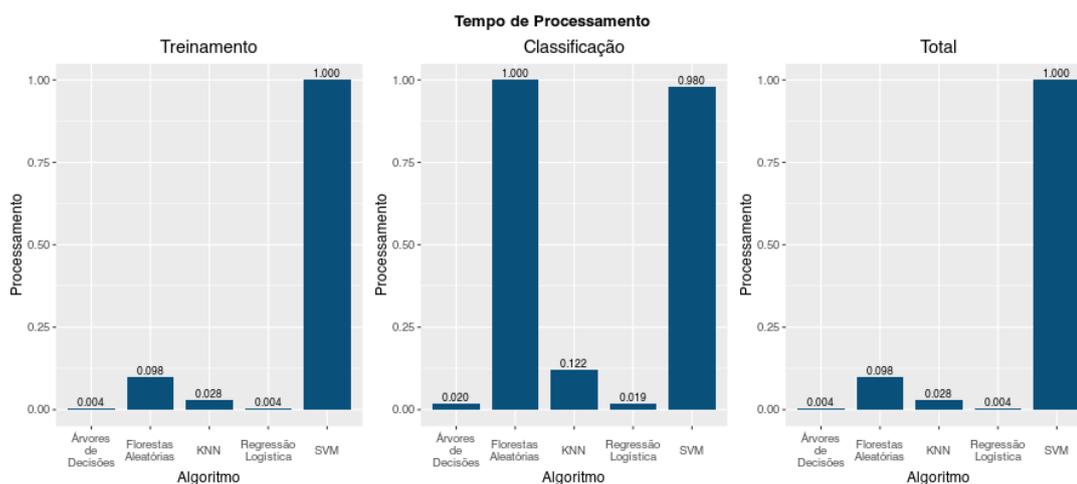
Para completar a análise, foi verificada a precisão dos algoritmos. Para isso, foram classificados 2274 dados coletados em tempo real e gerados em ambiente controlado. Foram analisados o percentual de acertos de cada algoritmo e o índice de falsos positivo⁴⁷ e falso

⁴⁵Os valores idênticos resultam do processo de arredondamento.

⁴⁶Descrição disponível em: <https://docs.python.org/3/library/time.html>.

⁴⁷Quantidade de amostras normais classificadas como anômalas.

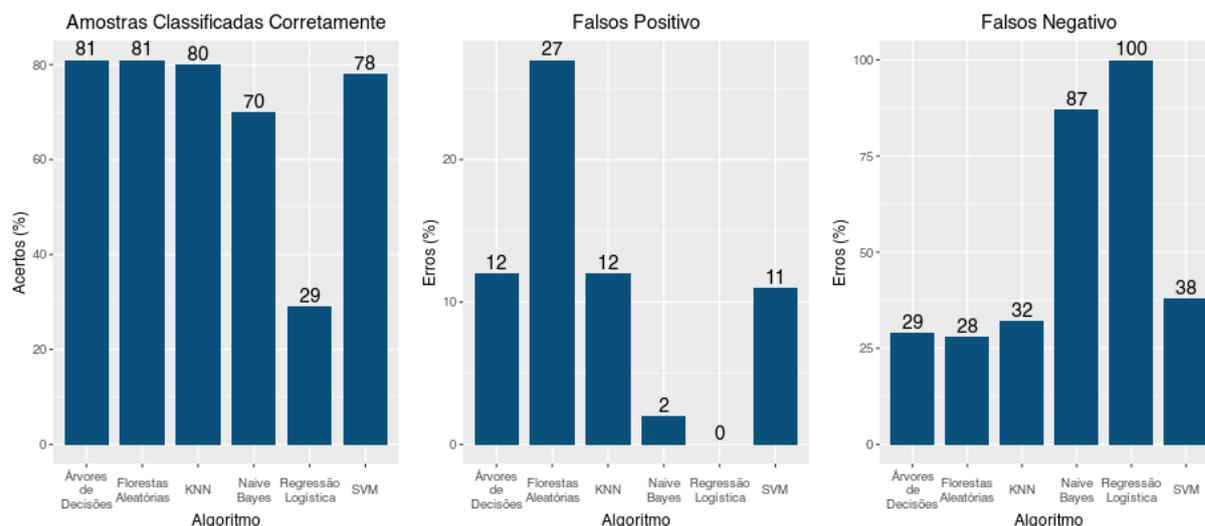
Figura C.3: Processamento consumido pelos algoritmos de aprendizagem de máquina para série de dados



FONTE: O Autor (2021)

negativo⁴⁸. A precisão pode ser observada nos gráficos da Figura C.4. Árvores de Decisões e Florestas Aleatórias foram os algoritmos que classificaram mais amostras corretamente, 1824 das 2274. Já KNN acertou a classificação de 1799 amostras. Apesar de ter alcançado a maior precisão, o algoritmo Florestas Aleatórias classificou incorretamente o maior número de amostras normais, 332. Já Naïve Bayes classificou 910 anômalas como corretas e Regressão Logística classificou todas as amostras como anômalas.

Figura C.4: Precisão alcançada pelos algoritmos de aprendizagem de máquina



FONTE: O Autor (2021)

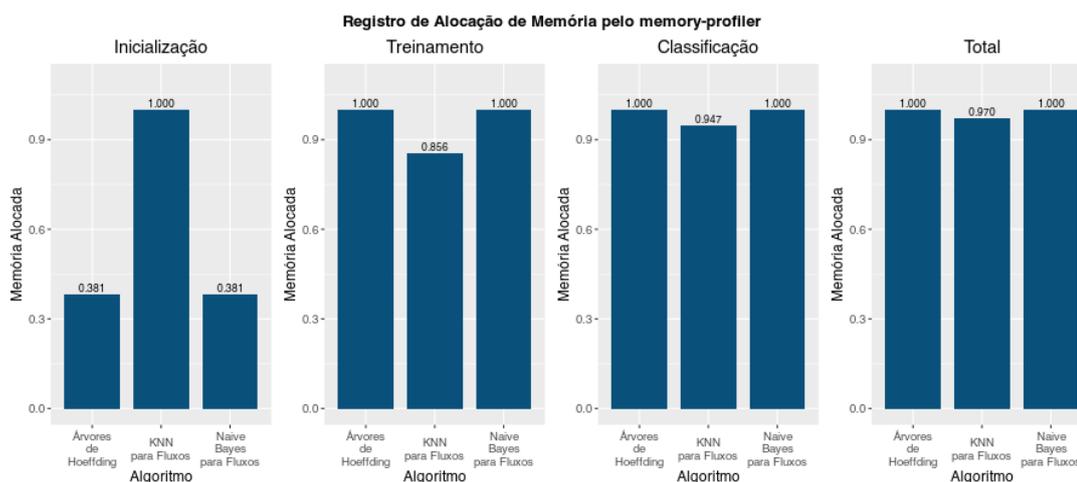
Observando o cenário geral, Florestas Aleatórias e Árvores de Decisões são os que obtêm melhor acurácia geral. No entanto, Florestas Aleatórias alcançam o dobro falsos positivo e consomem muito mais recursos computacionais. Por essa razão, optou-se por excluí-las dos

⁴⁸Quantidade de amostras anômalas classificadas como normais.

próximos testes. KNN, SVM e Naïve Bayes não foram descartados completamente, mas foram preteridos devido ao elevado consumo de recursos ou baixa precisão.

Além dos algoritmos para série de dados, foram testados aqueles especializados em fluxos de informações. Foram escolhidas três variações para fluxos de dados: Árvores de Hoeffding, baseado em Árvores de Decisões, KNN para fluxos e Naïve Bayes para fluxos. Todos eles classificaram todas as amostras corretamente. As Figuras C.5, C.6 e C.7 apresentam os resultados dos testes.

Figura C.5: Memory-profiler: memória alocada pelos algoritmos de aprendizagem de máquina para fluxos de dados



FONTE: O Autor (2021)

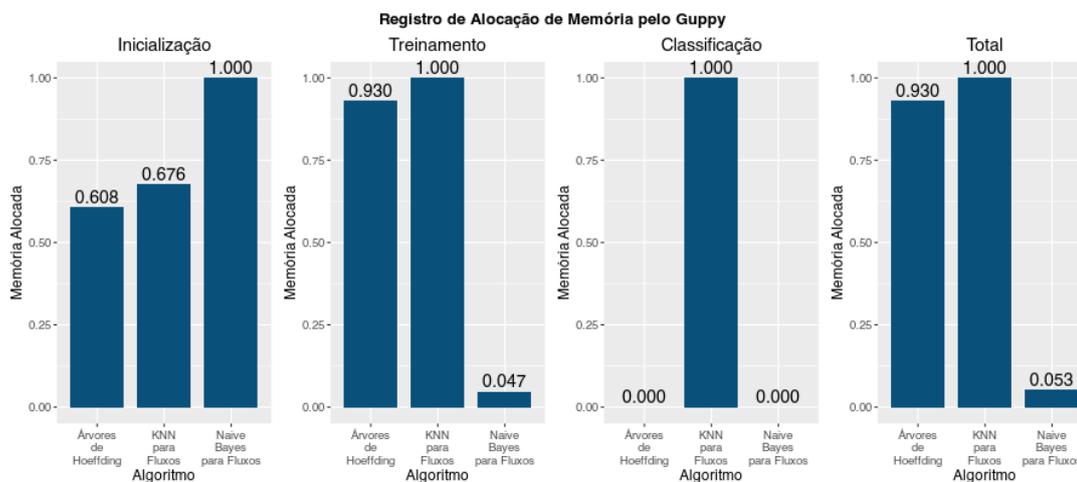
Com relação à memória, o memory-profiler aponta maior consumo por parte do KNN na inicialização. Durante o treinamento e a classificação, as Árvores de Hoeffding consomem mais. Ambos consomem quantidade de memória similar, com poucos *bytes* de diferença. De modo geral, o memory-profiler aponta o KNN como o mais econômico, apesar de todos alocarem quantidades muito próximas. Por outro lado, o Guppy registra um comportamento bem diferente. O Naïve Bayes aloca mais memória na inicialização, mas é o mais econômico nos processos seguintes. O KNN é o mais guloso em todas as fases seguintes e as Árvores de Hoeffding apresentam desempenho intermediário.

Com relação ao processamento, as Árvores de Hoeffding usam o processador por mais tempo, tanto na inicialização, quanto no treinamento. O KNN é o que mais processa durante a classificação dos dados, porém seu processamento é intermediário no contexto geral. O Naïve Bayes foi o mais econômico em todas as fases, caracterizando-se como o algoritmo que menos consome recursos computacionais.

Adicionalmente, também foi analisada a possibilidade de se usar Shapelets, um algoritmo para mineração de séries temporais. Os resultados de precisão foram excelentes, pois o algoritmo alcançou 100% de acurácia. No entanto, esse algoritmo consome muitos recursos computacionais. As Figuras C.8, C.9 e C.10 apresentam uma comparação dos resultados alcançados pelo Shapelets e os algoritmos para fluxos de dados.

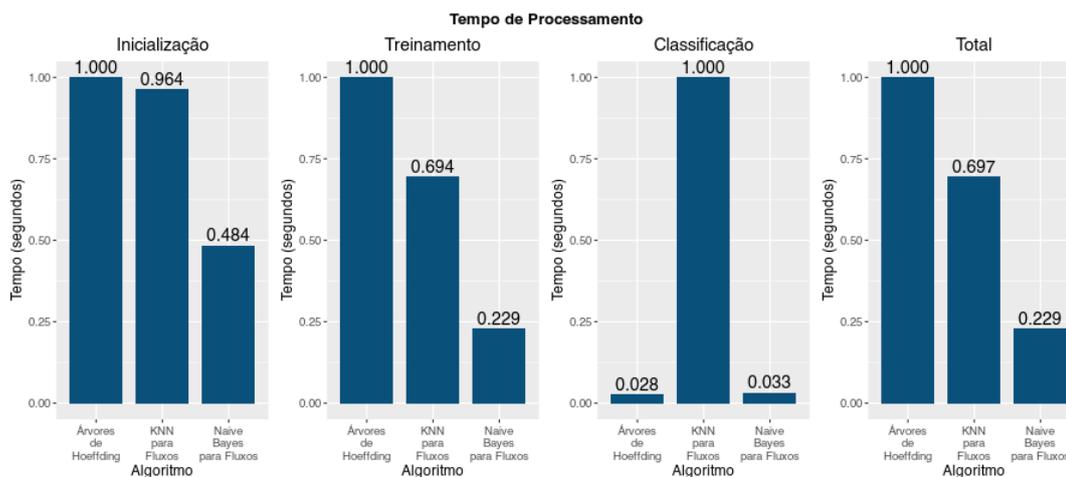
O consumo de memória registrado pelo memory-profiler ultrapassa é muito superior ao dos demais algoritmos durante a inicialização e treinamento dos dados. Durante a classificação, essa biblioteca não registrou consumo de memória adicional. Contudo, esse comportamento é bastante distinto do registrado pelo Guppy. Segundo este, o consumo de memória por parte dos

Figura C.6: Guppy: memória alocada pelos algoritmos de aprendizagem de máquina para fluxos de dados



FONTE: O Autor (2021)

Figura C.7: Processamento consumido pelos algoritmos de aprendizagem de máquina para fluxos de dados



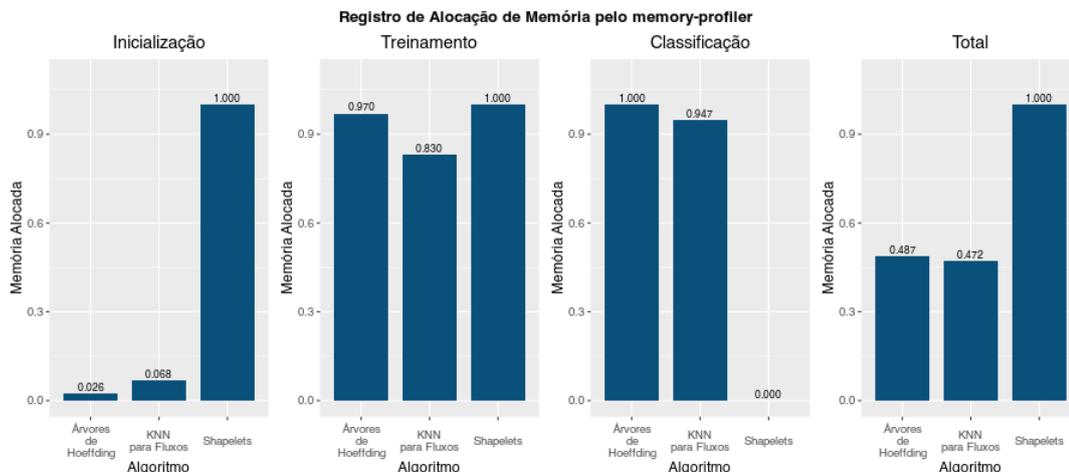
FONTE: O Autor (2021)

Shapelets é quase cinco vezes maior que os demais algoritmos durante a inicialização e mais de 75 vezes superior nas demais fases.

O que mais chamou a atenção foi o consumo de processamento. Os Shapelets utilizam uma quantidade muito alta de processamento, o que inviabiliza sua utilização. O tempo de execução do algoritmo é muito alto, o que gera aumento significativo na fila de dados a serem processados. Dependendo da massa de dados e da velocidade com que eles chegam à borda, pode haver perda de informações e o sistema pode ser levado a um estado de negação de serviço.

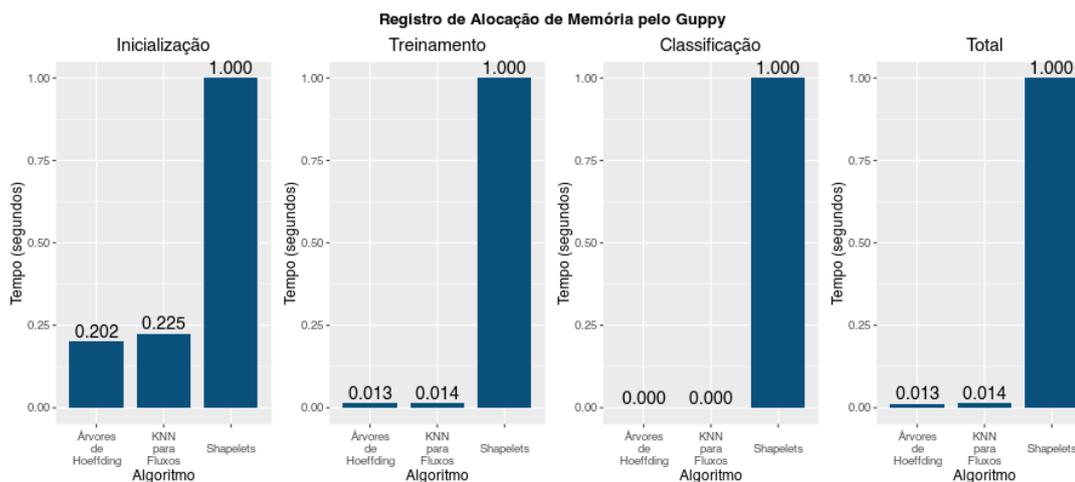
Por fim, foi realizada uma comparação entre as Árvores de Decisões, Árvores de Hoeffding e Naive Bayes para fluxos de modo a verificar o desempenho desses algoritmos ao classificarem múltiplos dados. Nesse teste foi incluída a verificação do desempenho dos algoritmos para classificar dados individuais e grandes conjuntos de dados para verificar as diferenças entre a classificação de dados individuais e grandes conjuntos. Para o teste dos

Figura C.8: Memory-profiler: comparação do consumo de memória dos Shapelets e dos algoritmos para fluxos de dados



FONTE: O Autor (2021)

Figura C.9: Guppy: comparação do consumo de memória dos Shapelets e dos algoritmos para fluxos de dados

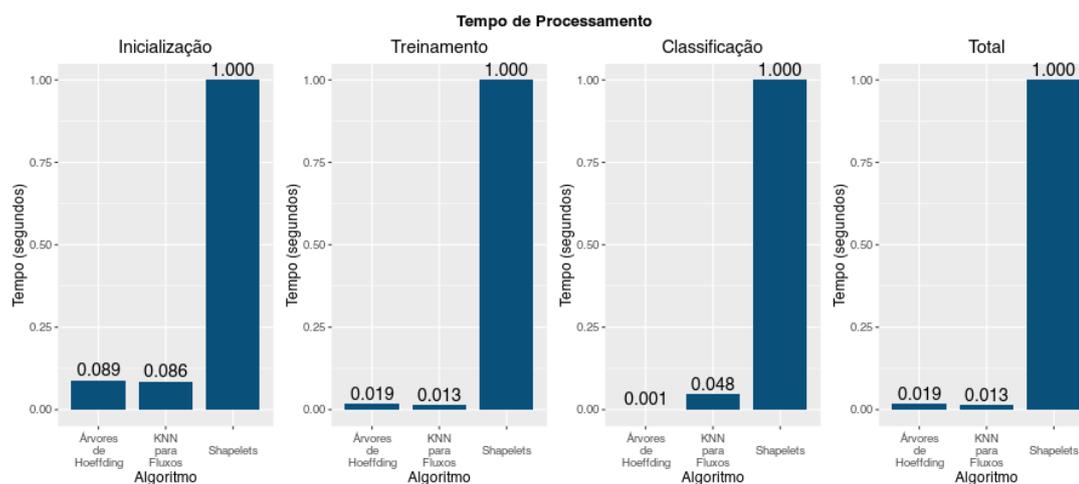


FONTE: O Autor (2021)

conjuntos, foram gerados um milhão valores aleatórios dentro compreendendo aqueles que estão na escala válida para o sensor, valores de falha e saturação.

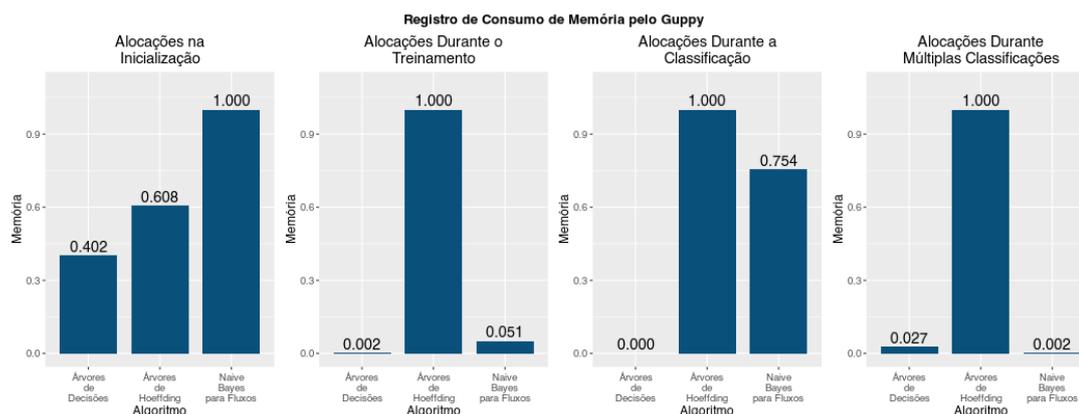
As Figuras C.11 e C.12 mostram que, apesar de Naïve Bayes alocar mais memória da inicialização, é o algoritmo mais econômico na classificação de múltiplos valores. Durante a classificação, as Árvores de Decisões trabalham com uma quantidade mínima de memória, tal que os valores são perceptíveis apenas quando são classificados milhares de registros. Com relação ao Naïve Bayes, o que se percebe é que, conforme as classificações acontecem, ocorrem alocações e liberações de memória, sendo que as liberações ocorrem com maior frequência. Observando os totais (Figura C.12), percebe-se que as Árvores de Decisões são bastante econômicas e o Naïve Bayes apresenta um desempenho bastante reduzido, considerando que realiza o treinamento parcial a cada nova classificação.

Figura C.10: Comparação do processamento utilizado pelos Shapelets e pelos algoritmos para fluxos de dados



FONTE: O Autor (2021)

Figura C.11: Memória alocada para classificação de valores únicos e múltiplos



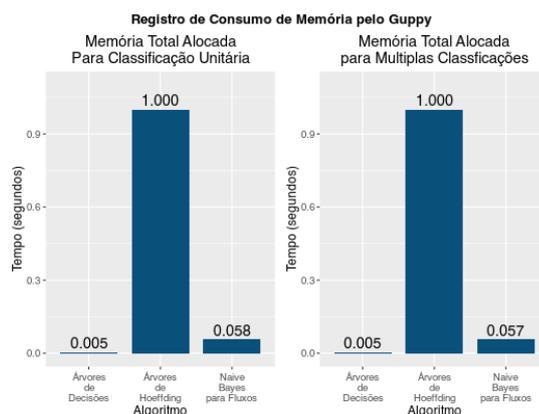
FONTE: O Autor (2021)

Nesta etapa, foi utilizado um novo analisador de memória, o Pympler⁴⁹, que informa a quantidade de memória alocada pelo objeto. O objetivo era agregar mais um elemento para a escolha do algoritmo usado no detector de anomalias. Segundo os dados desse analisador, as Árvores de Decisões alocam 15 vezes mais memória que Naive Bayes e 46 vezes mais que as Árvores de Decisões.

Os dados sobre o processamento (Figuras C.14 e C.15) mostram que, apesar das Árvores de Decisões utilizarem o processador durante mais tempo na inicialização, elas são as mais econômicas nos processos seguintes. As Árvores de Hoeffding são as que mais consomem recursos, consumindo quatro vezes mais processamento que Naive Bayes e quase 50 vezes mais que as Árvores de Decisões.

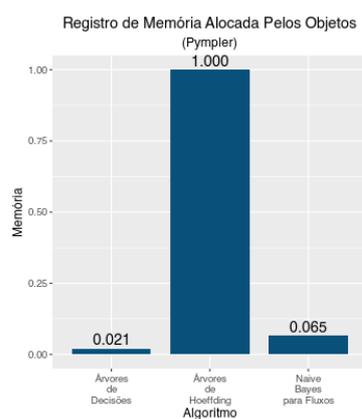
⁴⁹Descrição da biblioteca disponível em: <https://pythonhosted.org/Pympler/>

Figura C.12: Memória total alocada para classificação de valores únicos e múltiplos



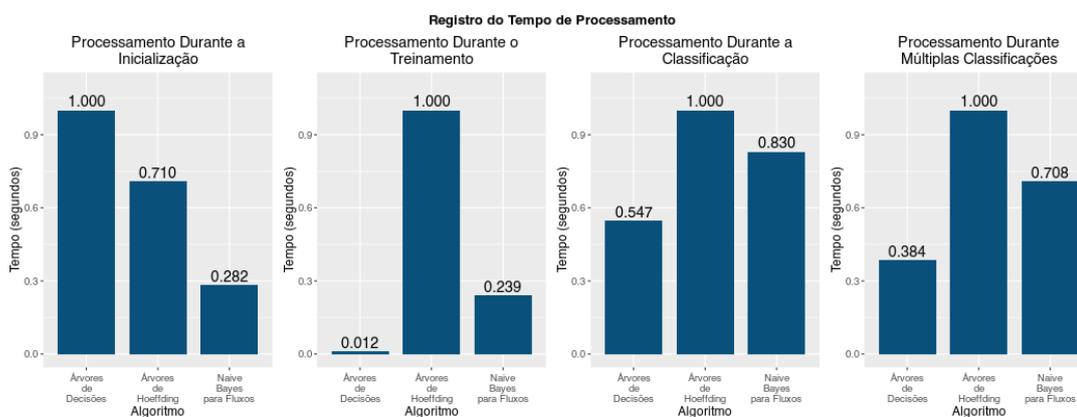
FONTE: O Autor (2021)

Figura C.13: Pympler: memória alocada pelos objetos



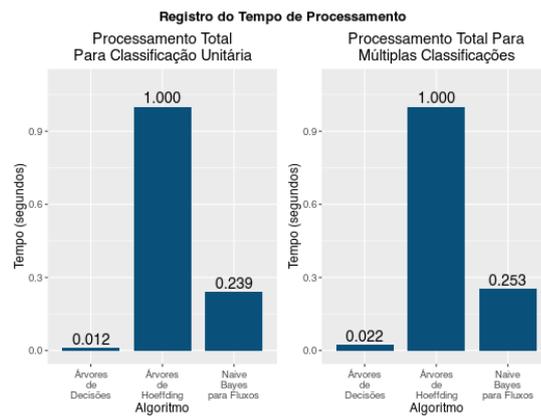
FONTE: O Autor (2021)

Figura C.14: Processamento utilizado durante a classificação de valores únicos e múltiplos



FONTE: O Autor (2021)

Figura C.15: Processamento total utilizado durante a classificação de valores únicos e múltiplos



FONTE: O Autor (2021)

APÊNDICE D – TABELA DE GASTOS

Equipamentos e suprimentos adquiridos até dezembro de 2020.

Tabela D.1: Orçamento Transdutores de Rede

Descrição	Quantidade	Valor unitário	Frete	Valor total
Raios UV	11	R\$ 10,14	R\$ 15,64	R\$ 127,18
Suporte de bateria	11	R\$ 0,77	R\$ 34,02	R\$ 42,49
Arduino	10	R\$ 7,84	R\$ 24,42	R\$ 102,82
Placa sem fio	10	R\$ 7,17	R\$ 15,19	R\$ 86,89
Sensor de pressão	11	R\$ 2,59	R\$ 25,93	R\$ 54,42
Conector 2 vias (20 por lote)	2	R\$ 6,78	R\$ 28,06	R\$ 41,62
Conector 3 vias (20 por lote)	2	R\$ 9,83		R\$ 19,66
Diodo 1N4148	11	R\$ 0,05		R\$ 0,55
Capacitor 100 nF	70	R\$ 0,05	R\$ 17,24	R\$ 20,74
Capacitor 100 uF	22	R\$ 0,14		R\$ 3,08
Capacitor 220 nF	33	R\$ 0,21		R\$ 6,93
Resistor 390k	11	R\$ 0,13		R\$ 1,43
Resistor 4k7	11	R\$ 0,13		R\$ 1,43
Resistor 10k	22	R\$ 0,13		R\$ 2,86
Resistor 1M	11	R\$ 0,13		R\$ 1,43
Resistor 2k2	22	R\$ 0,13		R\$ 2,86
Regulador de tensão	11	R\$ 2,65		R\$ 29,15
Trimpot 50 k ohm	11	R\$ 1,11		R\$ 12,21
Carregador	11	R\$ 7,12	R\$ 43,52	R\$ 121,84
Sensor de precipitação	5	R\$ 2,19	R\$ 18,07	R\$ 29,02
Placa solar	11	R\$ 6,45	R\$ 14,96	R\$ 85,91
Caixa plástica para montagem do sensor	10	R\$ 18,50		R\$ 185,00
Barra de pino fêmea - 40 pinos	10	R\$ 2,49	R\$ 18,90	R\$ 43,80
Prensa cabos para montagem do sensor	10	R\$ 2,49		R\$ 24,90
Bateria	22	R\$ 11,99	R\$ 35,60	R\$ 299,38
DHT22	10	R\$ 11,56	R\$ 0,00	R\$ 115,55
Fabricação da placa (lote com 10)	1	R\$ 51,00		R\$ 51,00
Abraçadeira de nylon para montagem (pct c/ 100)	1	14,6		14,6
Raspberry Pi 2 Model B	1	346,00		346,00
Cartão de memória micro SD 15 GB	1	35,00		35,00
Total				R\$1.909,75