



**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
UNIDADE ACADÊMICA DE SISTEMAS E COMPUTAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO**

ARAMIS SALES ARAUJO

**AVALIAÇÃO DA INTEGRAÇÃO DE MÉTRICAS DE REDE COM
DADOS DE EQUIPAMENTOS INDUSTRIAIS PARA USO EM
REDES MÓVEIS 5G DE FÁBRICAS INTELIGENTES**

**CAMPINA GRANDE - PB
2024**

Universidade Federal de Campina Grande
Centro de Engenharia Elétrica e Informática
Coordenação de Pós-Graduação em Ciência da Computação

Avaliação da Integração de Métricas de Rede com
Dados de Equipamentos Industriais para Uso em
Redes Móveis 5G de Fábricas Inteligentes

Aramis Sales Araujo

Dissertação submetida à Coordenação do Curso de Pós-Graduação em
Ciência da Computação da Universidade Federal de Campina Grande -
Campus I como parte dos requisitos necessários para obtenção do grau
de Mestre em Ciência da Computação.

Área de Concentração: Ciência da Computação

Linha de Pesquisa: Engenharia de Software

Danilo Freire de Souza Santos

(Orientador)

Campina Grande, Paraíba, Brasil

©Aramis Sales Araujo, 24/01/2024

A663a

Araujo, Aramis Sales.

Avaliação da integração de métricas de rede com dados de equipamentos industriais para uso em redes móveis 5G de fábricas inteligentes / Aramis Sales Araujo. – Campina Grande, 2024.

79 f. : il. color.

Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Campina Grande, Centro de Engenharia Elétrica e Informática, 2024.

"Orientação: Prof. Dr. Danilo Freire de Souza Santos".

Referências.

1. Redes Móveis 5G – Fábricas Inteligentes. 2. Intent-Based Networks. 3. Aprendizado de Máquina. 4. Internet das Coisas. 5. Engenharia de Software. I. Santos, Danilo Freire de Souza. II. Título.

CDU 004.7(043)



MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE CAMPINA GRANDE

POS-GRADUACAO EM CIENCIA DA COMPUTACAO

Rua Aprígio Veloso, 882, Edifício Telmo Silva de Araújo, Bloco CG1, - Bairro Universitário, Campina Grande/PB, CEP 58429-900

Telefone: 2101-1122 - (83) 2101-1123 - (83) 2101-1124

Site: <http://computacao.ufcg.edu.br> - E-mail: secretaria-copin@computacao.ufcg.edu.br / copin@copin.ufcg.edu.br

FOLHA DE ASSINATURA PARA TESES E DISSERTAÇÕES

ARAMIS SALES ARAÚJO

AVALIAÇÃO DA INTEGRAÇÃO DE MÉTRICAS DE REDE COM DADOS DE EQUIPAMENTOS INDUSTRIAIS PARA USO EM REDES MÓVEIS 5G DE FÁBRICAS INTELIGENTES

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação como pré-requisito para obtenção do título de Mestre em Ciência da Computação.

Aprovada em: 16/02/2024

Prof. Dr. DANILO FREIRE DE SOUZA SANTOS, UFCG, Orientador

Prof. Dr. DALTON CÉZANE GOMES VALADARES, IFPE, Examinador Interno

Prof. Dr. ÁLVARO ALVARES DE CARVALHO CÉSAR SOBRINHO, UFAPE, Examinador Externo



Documento assinado eletronicamente por **DANILO FREIRE DE SOUZA SANTOS, PROFESSOR(A) DO MAGISTERIO SUPERIOR**, em 16/05/2024, às 10:30, conforme horário oficial de Brasília, com fundamento no art. 8º, caput, da [Portaria SEI nº 002, de 25 de outubro de 2018](#).



Documento assinado eletronicamente por **Dalton Cézane Gomes Valadares, Usuário Externo**, em 16/05/2024, às 10:51, conforme horário oficial de Brasília, com fundamento no art. 8º, caput, da [Portaria SEI nº 002, de 25 de outubro de 2018](#).



Documento assinado eletronicamente por **Álvaro Alvares de Carvalho César Sobrinho, Usuário Externo**, em 16/05/2024, às 13:32, conforme horário oficial de Brasília, com fundamento no art. 8º, caput, da [Portaria SEI nº 002, de 25 de outubro de 2018](#).



A autenticidade deste documento pode ser conferida no site <https://sei.ufcg.edu.br/autenticidade>, informando o código verificador **4433678** e o código CRC **C016D50E**.

Abstract

5G mobile networks play a fundamental role in connected smart factories, providing significant improvements in security and Quality of Service (QoS) factors compared to previous generations. The ability to transmit large amounts of data with low latency enables remote control and monitoring in critical scenarios, involving the control of machines, industrial devices, and the monitoring of smart sensors. Considering the scenario in which 5G mobile networks are applied to industry, the opportunity to prioritize data transmission flows and network services using network orchestration tools becomes evident. For this purpose, Intent-Based Network Systems (IBNs) are tools that combine artificial intelligence techniques and network orchestration capabilities to comply with high-level defined business intents, improving communication effectiveness and promoting the adequate functioning of the systems involved. In this context, this work aims to evaluate the integration between the data and statistics from Internet of Things (IoT) devices, observed within a connected industrial environment, and the metrics provided by the 5G network utilized in smart factories, in the use of neural network models for the task of classifying network problem scenarios like the increase of communication latency. Therefore, the quantitative evaluation of these models' performance highlights the potential of IoT devices' data in the task of network problem scenario classification. Bringing out the potential to automate and reconfigure network systems in a fast, secure, and autonomous way, promoting compliance with the rules defined in the business model of the specific industrial environment.

Keywords: 5G Mobile Networks, Smart Factories, Intent-Based Network, Machine Learning, Internet of Things.

Resumo

As redes de comunicação 5G desempenham um papel fundamental nas fábricas inteligentes, proporcionando melhorias significativas nos fatores de segurança e Qualidade do Serviço (em inglês *Quality of Service*, QoS) em comparação com as gerações anteriores. A capacidade de transmitir grandes volumes de dados com baixa latência possibilita o controle e monitoramento remoto em cenários críticos, envolvendo o controle de máquinas, dispositivos industriais e o monitoramento de sensores inteligentes. Observando este cenário de redes 5G aplicadas à indústria, nota-se a oportunidade de realizar a priorização de fluxos e serviços, a partir de ferramentas autônomas de gestão de rede. Sistemas de redes baseadas em intenção (em inglês, *Intent-Based Network Systems*) são ferramentas que combinam técnicas de inteligência artificial e orquestração de rede para atender à objetivos definidos em linguagem de alto nível, aprimorando eficácia da comunicação e fomentando o bom funcionamento dos sistemas de rede de forma autônoma. Nesse contexto, este trabalho objetiva integrar os dados e estatísticas provenientes de dispositivos de internet das coisas (em inglês *Internet of Things*, *IoT*) presentes em um ambiente industrial conectado, com as métricas providas pela rede 5G utilizada na indústria, para realizar a comparação quantitativa entre modelos de rede neural capazes de distinguir eventuais cenários de problemas na rede e na comunicação entre os dispositivos *IoT*, como o aumento da latência na comunicação. Portanto, a avaliação quantitativa do desempenho desses modelos ressalta a importância dos dados advindos dos dispositivos do ambiente industrial nesta tarefa de classificação de cenários de problemas de rede. Evidenciando o potencial de realizar a automação da gestão e aprimorar o processo de reconfiguração de rede de forma rápida e segura, assegurando o cumprimento objetivos definidos no modelo de negócio do ambiente industrial.

Palavras-Chave: Redes Móveis 5G, Fábricas Inteligentes, *Intent-Based Networks*, Aprendizado de Máquina, Internet das Coisas.

Agradecimentos

Gostaria de agradecer a Deus, por diversas graças e coragem que me possibilitou atingir meus objetivos. Agradeço imensamente à minha família, em especial e primeiramente às minhas irmãs, que me deram todo amor e suporte na minha jornada, além de me inspirar a buscar grandes conquistas. Agradeço aos meus pais por todo o amor e educação que me deram, em especial à minha mãe que me proporcionou sonhar mais alto. Não haveriam páginas suficientes para agradecer aos meus amigos, que estiveram presentes em cada momento de alegria e de desafios da minha jornada. Obrigado por tudo. Gostaria de agradecer ao meu orientador Danilo, por tamanha paciência, dicas, ensinamentos e encorajamento. Por fim, agradeço aos colegas do Virtus@UFCG pelo companheirismo no dia a dia de trabalho.

Conteúdo

1	Introdução	9
1.1	Problemática	12
1.2	Objetivo	14
1.3	Metodologia e Contribuições	15
1.4	Estrutura do documento	17
2	Fundamentação Teórica	18
2.1	Serviços e Funções de uma Rede Móvel 5G	18
2.2	Métricas de Rede 5G e de Comunicação dos Dispositivos IoT	22
2.3	Redes Neurais e Aprendizado de Máquina	23
2.3.1	Redes Neurais RNN e LSTM	23
2.3.2	Treinamento e avaliação de Modelos de Rede Neural	25
2.4	Considerações Finais	26
3	Trabalhos Relacionados	27
3.1	Redes Privadas 5G em Fábricas Inteligentes	27
3.2	Implementações de Código Aberto para Núcleo 5G	28
3.3	<i>Robot Operating System 2</i> para Dispositivos IoT	28
3.4	Aprendizado de Máquina para Análise de Dados de Rede 5G	29
3.5	Considerações Finais	31
4	Metodologia	32
4.1	Experimento	35
4.2	Coleta das Métricas	38
4.3	Cenários de Problemas de Rede	38

4.4	Métricas Coletadas	40
4.4.1	Métricas do Núcleo 5G	40
4.4.2	Métricas da Comunicação dos Dispositivos	41
4.5	Conjunto de Dados	43
4.5.1	Amostras Coletadas	44
4.5.2	Divisão para Treinamento, Teste e Validação	46
4.6	Problema de Classificação	46
4.6.1	Modelo de Rede Neural	46
4.6.2	Metodologia para avaliação dos modelos	47
4.7	Considerações Finais	48
5	Implementação e Resultados	49
5.1	Arquitetura dos Componentes do Experimento	49
5.2	Técnicas utilizadas para treinamento dos modelos	55
5.3	Resultados	59
5.3.1	Avaliação dos modelos	59
5.3.2	Discussão	67
5.3.3	Desafios e limitações	68
5.4	Considerações Finais	69
6	Conclusão	71
6.1	Resumo dos Resultados	71
6.2	Trabalhos Futuros	72

Lista de Símbolos

3GPP - *3rd Generation Partnership Project*

DDS - *Data Distribution Service*

DN - *Data Network*

GPRS - *General Packet Radio Service*

gNB - *Next Generation Node B (gNodeB)*

GTP - *GPRS Tunneling Protocol*

IBN - *Intent Based Network*

IoT - *Internet of Things*

LSTM - *Long Short-Term Memory*

LTE - *Long-Term Evolution*

LTS - *Lont-Term Support*

LXD - *Linux Containers Daemon*

NWDAF - *Network Data Analytics Function*

ONF - *Open Network Foundation*

PDU - *Packet Data Unit*

QoS - *Quality of Service*

RAN - *Radio Access Network*

ROS2 - *Robot Operating System 2*

SMF - *Session Management Function*

TUN - *Network Tunnel*

UDP - *User Datagram Protocol*

UE - *User Equipament*

UPF - *User Plane Field*

Lista de Figuras

1.1	<i>Network Slicing</i> para fábricas inteligentes [41]	10
2.1	Arquitetura simplificada de serviços 5G e suas interfaces [1]	22
2.2	Arquitetura da célula LSTM [10]	24
4.1	Fases da metodologia utilizada neste trabalho	34
4.2	Amostra de uma única sequência dos dados de rede e da comunicação dos dispositivos IoT	45
5.1	Arquitetura simplificada dos serviços e aplicações utilizados no experimento	52
5.2	Arquitetura das interfaces de rede utilizadas no experimento	54
5.3	Arquitetura simplificada definida para os modelos	56
5.4	Decaimento da perda dos modelos durante treinamento (<i>training loss</i>) do modelo com subconjunto de dados do núcleo 5G	63
5.5	Decaimento da perda dos modelos durante treinamento (<i>training loss</i>) do modelo com dados do núcleo 5G e dispositivos IoT	64
5.6	Curva da acurácia do modelo com dados do núcleo 5G	64
5.7	Curva da acurácia do modelo com dados do ecossistema completo	65
5.8	Curva ROC para o modelo com dados do núcleo 5G ^l	66
5.9	Curva ROC para o modelo com dados do ecossistema completo	66
5.10	Importância dos atributos para o modelo com dados do núcleo 5G	68
5.11	Importância dos atributos para o modelo com dados do ecossistema completo	68

Lista de Tabelas

2.1	Componentes do Núcleo 5G	19
4.1	Cenários de Problemas de Rede do Experimento	39
4.2	Métricas Coletadas do Núcleo 5G e Serviços	41
4.3	Métricas Coletadas da Comunicação dos Dispositivos IoT	41
5.1	Caracterização das Máquinas Virtuais do Experimento	50
5.2	Hiperparâmetros Modelo Core 5G	58
5.3	Hiperparâmetros Modelo Core 5G + IoT	59
5.4	Resultados obtidos para o modelo treinado apenas com dados do Núcleo 5G	61
5.5	Resultados obtidos para o modelo com dados do núcleo e dos dispositivos IoT	62

Capítulo 1

Introdução

A revolução industrial 4.0 desencadeou várias inovações e impulsionou a interconexão de conceitos-chave como Internet das Coisas (em inglês *Internet of Things*, *IoT*), redes de comunicação móvel 5G [39] e a análise de dados em larga escala (em inglês, *Big Data Analysis*) [22]. Esta sinergia não apenas transforma o cenário produtivo, como promoveu a evolução do próprio conceito de fábrica. As *Smart Factories*, ou fábricas inteligentes, surgem como protagonistas nesse novo cenário conectado, prometendo ganhos significativos em produtividade, flexibilidade e autonomia [15].

Nas *Smart Factories*, sensores inteligentes e outros dispositivos *IoT* são usados para coletar dados em tempo real sobre o desempenho dos processos produtivos. A coleta destes dados num cenário industrial resulta em volumes massivos de informação que torna necessária a utilização técnicas de análise de *Big Data* para identificar oportunidades de otimização de processos de produção [22]. As *Smart Factories* também usam artifícios de inteligência artificial para automatizar tarefas e tomar decisões sem a intervenção humana com base nestas análises [36].

Para realizar a comunicação entre os dispositivos nesse cenário, o uso de redes móveis privadas 5G oferece vantagens expressivas quando comparado às gerações anteriores por proporcionarem conexão de alta velocidade e baixa latência, que são características essenciais para aplicações *IoT* específicas do cenário industrial, como a transferência de dados de sensores em tempo real, coordenação de tarefas ou tomada de decisões [42]. Além destas vantagens, as redes 5G apresentam escalabilidade aprimorada para suportar a conexão de aglomerados de dispositivos e também funcionalidades de isolamento de serviços nati-

vamente, permitindo a partição e priorização de seções da rede para garantir características distintas de latência, largura de banda e segurança a estas seções, adaptando a rede para atender a requisitos específicos de cada fluxo operacional[30]. Outra vantagem importante das redes móveis 5G é a provisão de ferramentas de orquestração e gerenciamento que permitem o monitoramento de métricas da rede e também o controle para reconfiguração das características da rede, como o fatiamento de rede (em inglês *network slicing*), Figura 1.1 [48, 41].

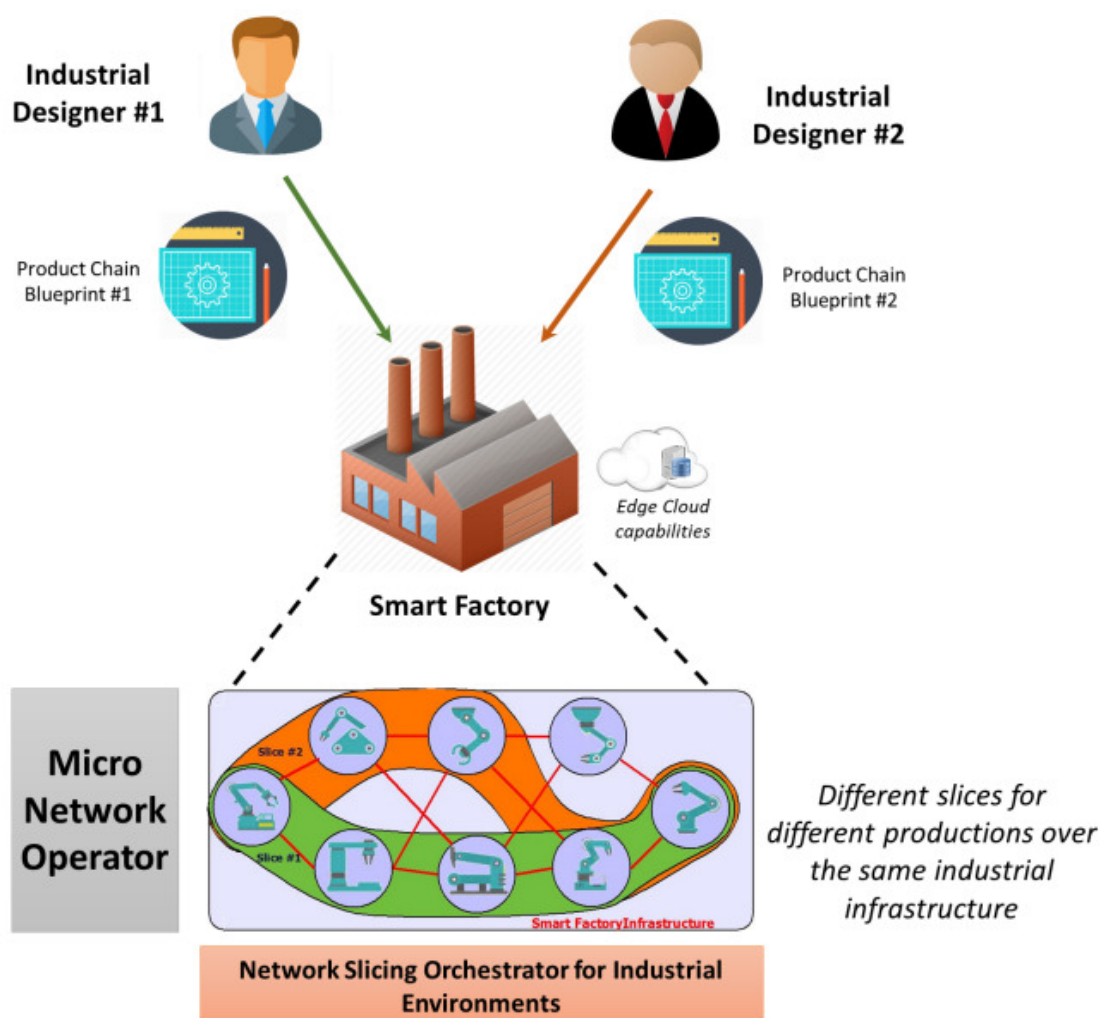


Figura 1.1: *Network Slicing* para fábricas inteligentes [41]

É a partir das funcionalidades mencionadas anteriormente que se torna viável a implantação de redes baseadas em intenções (em inglês *Intent-based networks, IBNs*) [35]. As *IBNs* são uma abordagem de gerenciamento de redes que combina artifícios de inteligência artificial, aprendizado de máquina e orquestração de rede, para atender às intenções ou objetivos de desempenho definidos em linguagem de alto nível para a rede.

Esses objetivos são definidos por um administrador de rede, com o intuito de garantir recursos e características específicas para cada segmento da rede, priorizando conexões de baixa latência para aplicações críticas, como o controle de robôs, ou garantindo alta largura de banda para aplicações de coletas de dados de diversos sensores presentes no cenário de *smart factory*.

Em detalhes, *IBNs* utilizam algoritmos de aprendizado de máquina para analisar dados históricos da rede, identificando padrões e tendências que permitem prever ou identificar com maior precisão o comportamento futuro do ecossistema. Estes algoritmos alimentam mecanismos de detecção de problemas de rede que permitem a identificação automática destes problemas, facilitando a resposta imediata à ameaças de segurança e problemas de desempenho em tempo real. Além disso, as técnicas de aprendizado de máquina possibilitam que os sistemas *IBN* aprimorem-se de forma contínua e otimizem seus processos de tomada de decisão, a partir do aprendizado com ações e resultados anteriores, promovendo operações de rede mais eficazes e alinhadas com as intenções definidas.

À medida que as fábricas adicionam novos dispositivos e aplicações à sua rede, as *IBNs* podem ser utilizadas para garantir a flexibilidade necessária para adaptar-se ao cenário e reagir a eventos inesperados como falhas de *hardware* ou ataques[35]. Isso permite a automatização de tarefas de provisionamento de recursos, monitoramento e resolução de problemas de rede, reduzindo também a necessidade de intervenção humana, o que pode levar a economia de custos.

Há, porém, alguns desafios que vão além de custos operacionais e de *hardware* para a implantação de *IBNs*. Um dos principais desafios é a demanda por profissionais especialistas para o planejamento, implantação e manutenção destes sistemas. Isto ocorre por que as *IBNs* são tecnologias complexas que exigem conhecimento profundo de redes, automação e segurança[47]. Além disso, *IBNs* requerem a implantação de monitoramento e ferramentas de orquestração mais proativas do que redes tradicionais devido à sua necessidade de informações da rede para garantir a capacidade dinamicidade. É importante também ressaltar a alta responsabilidade ao suportar aplicações críticas como automação de processos e a coleta dos dados que são essenciais para realimentação dos mecanismos de aprendizado inerentes à *IBN*. Estes requisitos demandam alta disponibilidade e confiabilidade do sistema de rede.

Neste sentido, a inclusão de métricas advindas do ecossistema *IoT*, mais especificamente

do cenário industrial (em inglês *Industrial Internet of Things*, IIoT) [6], pode otimizar o processo de gestão e tomada de decisão neste âmbito, introduzindo informações que podem dar margem a visões de cenários de problemas de conectividade, congestionamento e consumo de banda que não estariam em evidência considerando apenas as métricas advindas da rede. Com isto, viabilizando a atuação precisa e com medidas preventivas para alocação de recursos da rede, otimização do tráfego e redução do tempo de inatividade.

Observando esta hipótese, é neste contexto de gestão inteligente e autônoma de redes móveis 5G, num cenário de fábricas inteligentes que se insere este trabalho. Mais especificamente, o foco é avaliar um ambiente industrial conectado, no qual dispositivos *IoT* e outros equipamentos conectados comunicam-se a partir de uma rede móvel 5G, e realizar a coleta das métricas desta rede em conjunto com as métricas do ecossistema *IIoT*. O objetivo é observar se a combinação destas informações pode prover melhorias a um modelo de inteligência artificial que classifica diferentes cenários de problemas de rede em comparação ao modelo que utiliza apenas as métricas de rede, como de uma rede 5G.

1.1 Problemática

A implantação de *IBNs* apoia-se na análise inteligente de métricas da rede para tomar decisões e adaptar a alocação de recursos a fim de atender aos requisitos definidos pelo administrador da rede de forma autônoma. Num cenário de *Smart Factory*, além das métricas inerentes à rede, é possível utilizar as métricas referentes ao ecossistema *IIoT* para ampliar a base de conhecimento do sistema de rede.

Essas métricas podem fornecer informações valiosas sobre o uso da rede e dos dispositivos, permitindo que o sistema inteligente tome decisões mais precisas e oportunas para aprimorar a autonomia geral do sistema de rede, em especial nos casos de falhas ou inconsistências de comunicação. Por exemplo, a análise destas pode indicar problemas como aumento de latência, perda de pacotes ou congestionamento causado por insuficiência de banda.

Observando a questão específica de identificação de problemas na conexão entre os dispositivos, reitera-se a importância de considerar as informações providas pelo sistema utilizado para comunicação entre eles, pois a não inclusão destas informações na análise pode

encobrir falhas como aumento de latência, perda de pacotes e congestionamento de banda que podem levar a consequências graves, como:

- Inoperabilidade dos dispositivos, devido à sobrecarga ou a problemas de conectividade.
- A interrupção da comunicação entre serviços e aplicações, que pode afetar o desempenho de processos críticos da fábrica.
- Perda de dados, o que pode prejudicar a tomada de decisão e, conseqüentemente, a produtividade.

Entretanto, a coleta destes dados é uma tarefa que requer o esforço de sincronizar todas as aplicações, serviços e sistemas envolvidos, para que se possa associar cada instante de tempo ao cenário atual do ecossistema construído. A natureza sequencial e sensível ao tempo destes dados tem sido um problema desafiador para a área de inteligência artificial no âmbito de classificação [16]. Modelos de rede neural, porém, têm se destacado em eficácia para estas tarefas de classificação com dados de séries temporais [46], portanto, são oportunos candidatos para complementar sistemas que agem de forma autônoma para alcançar objetivos declarados em alto nível. Porém, há uma lacuna na observação quantitativa de melhorias provenientes da utilização diferentes subconjuntos de dados da rede e seus componentes para treinamento destes modelos, objetivando a tarefa específica de classificação de cenários de problemas de rede.

A partir da problemática evidenciada anteriormente, observa-se que na literatura há poucos trabalhos que discorrem sobre a combinação dos subconjuntos de dados da rede e dos sistemas *IIoT*, como o trabalho de Zhong et al. em [50] com foco nos aspectos qualitativos desta combinação, e Wang et al. em [45], que demonstra um experimento realizado com foco na otimização da taxa de utilização da rede. Nota-se, porém, a escassez de trabalhos que evidenciam quantitativamente as melhorias providas pela inclusão das métricas do ecossistema *IIoT* na tarefa de identificação de problemas de comunicação na rede.

Com base nos trabalhos citados e na temática específica inexplorada, elaborou-se a problemática de pesquisa deste trabalho. Esta consiste na exploração da combinação de métricas da rede 5G privativa utilizada num cenário de fábricas inteligentes e dos sistemas *IIoT*,

também presentes neste cenário, para avaliar quantitativamente as melhorias que esta combinação pode oferecer para a identificação e classificação de cenários de problemas de rede.

Com o intuito de explorar efetivamente a problemática mencionada, foram elencadas as seguintes questões de pesquisa (QP):

- **QP1:** É possível realizar a coleta e consolidação dos conjuntos de dados de um núcleo 5G e ecossistema *IIoT* de forma sincronizada sem que haja sobrecarga no funcionamento destes sistemas?
- **QP2:** Um modelo treinado com dados que incorporam informações advindas do núcleo 5G e do ecossistema *IIoT* é capaz prover melhor desempenho na tarefa de classificação de cenários de problemas de rede, quando comparado a outro treinado com o subconjunto de dados provenientes apenas do núcleo 5G?
- **QP3:** Dado o cenário atual de ferramentas de código aberto e visando promover a reprodutibilidade deste trabalho, é possível a implantação de um ecossistema conjunto de rede 5G e aplicações *IoT* no âmbito de fábricas inteligentes?

Diante disto, temos dois principais problemas abordados neste trabalho. O primeiro trata da exploração da coleta de métricas da rede 5G e do ecossistema *IIoT* em diferentes cenários de problemas de comunicação para construção de um conjunto de dados destes cenários. E o segundo trata da avaliação quantitativa das melhorias em métricas como acurácia e precisão, na utilização destes subconjuntos de dados no treinamento de modelos de rede neural capazes de classificar problemas de rede.

1.2 Objetivo

O objetivo geral deste trabalho é a avaliação da viabilidade do uso combinado de métricas coletadas por dispositivos *IIoT* com as métricas obtidas a partir do núcleo 5G e serviços num contexto de fábrica inteligente, para auxiliar na tarefa de classificação de cenários de problemas de comunicação. Desta forma, provendo uma visão quantitativa que sustenta a hipótese de que as métricas coletadas a partir do núcleo 5G são importantes, mas que combiná-las com as informações do ecossistema *IIoT* das fábricas inteligentes provém resultados ainda

melhores de precisão, acurácia e sensibilidade (em inglês *recall*) para um modelo de classificação destes cenários de problema de comunicação.

Com base neste objetivo geral, foram definidos os seguintes objetivos específicos:

- Projetar e implantar um ambiente simulado de rede 5G a partir do qual aplicações de fábrica inteligente realizam comunicação entre si;
- Implantar um sistema para coleta das métricas da rede 5G e do ecossistema *IIoT*;
- Definir e simular cenários de problemas de conexão de rede para posterior treinamento de modelos de *Deep Learning* objetivando a classificação destes cenários;
- Avaliar quantitativamente a melhora das métricas de precisão, acurácia e sensibilidade (*recall*) do modelo a partir da fusão dos conjuntos de dados coletados.

1.3 Metodologia e Contribuições

Para atingir os objetivos listados anteriormente, foi adotada uma metodologia de engenharia [18], que possui como objetivo a proposição, o desenvolvimento, a avaliação e o aprimoramento de uma solução, como descrito a seguir.

Foi proposto analisar um problema de classificação multiclasse, no âmbito de cenários de problemas de rede que podem gerar grande impacto no funcionamento de uma fábrica inteligente. Dada esta problemática de classificação, foram utilizados dois modelos de inteligência artificial e a avaliação comparativa entre eles foi feita quantitativamente nos termos de precisão, acurácia e sensibilidade (*recall*).

Em detalhes, foram utilizadas as métricas obtidas da rede 5G para o treinamento de dois modelos na tarefa de classificação de cenários de problemas de conexão dos dispositivos, diferenciando-se um dos modelos que recebeu também as métricas relacionadas ao ecossistema *IIoT* durante as etapas de treinamento, teste e validação.

A implantação da rede 5G foi feita utilizando um núcleo 5G (*Core 5G*) simulado de código aberto [26], a partir do qual foram conectadas estações radio-base simuladas *gNodeB* (*gNB*) para interconexão dos dispositivos.

Os dispositivos *IoT* também foram implantados de forma simulada a partir do *ROS2* [20], com o uso de atuadores que realizaram tarefas arbitrárias, gerando carga na rede. Estes dispositivos se comunicaram a partir das *gNBs* mencionadas anteriormente.

A partir dos componentes de rede 5G e dispositivos *IoT*, foram coletadas métricas de forma sincronizada, com o intuito de assegurar a característica de sensibilidade temporal dos dados para cada cenário de problema de rede explorado. Esta característica é de grande importância para este problema proposto de classificação, pois cada amostra do conjunto de dados coletados refere-se a uma sequência de observações das métricas da rede e do sistema de comunicação dos dispositivos. Denotando o estado do ecossistema de comunicação em cada instante.

As métricas que indicam tal estado do ecossistema, foram consolidadas em um conjunto de dados unificado, de forma tabular e devidamente estratificada para cada cenário de problema de rede, detalhado no [Seção 4.1](#). A partir deste, foi particionado um subconjunto no qual foram excluídos as informações relativas ao sistema de comunicação dos dispositivos que se comunicaram a partir da rede, portanto informações como latência da comunicação não estão disponíveis neste subconjunto. Estes dois conjuntos de dados foram definidos para que se pudesse realizar a comparação entre os modelos de rede neural na tarefa de classificação.

Objetivando classificar os cenários de problema de rede, foram implementados dois modelos de rede neural com a utilização das bibliotecas Tensorflow [2], Keras [11] e Scikit-learn [27], disponíveis na linguagem de programação Python. Os detalhes da estratificação dos conjuntos de dados, treinamento, teste e validação desses modelos estão descritos na [Seção 5.2](#). A comparação de caráter quantitativo desses modelos na tarefa de classificação se deu a partir da observação das métricas de precisão, acurácia, sensibilidade e das taxas de falso positivo e positivo verdadeiro, observados individualmente para cada classe que representa um cenário de problema de rede.

A principal contribuição deste trabalho é a avaliação quantitativa das métricas mencionadas no parágrafo anterior, para cada modelo de classificação de cenários de problema de rede, os quais foram treinados a partir de dois conjuntos de dados distintos. O primeiro contendo todas as métricas coletadas da rede 5G e dos dispositivos *IoT* e o segundo como um subconjunto do primeiro, contendo apenas as métricas relacionadas à rede 5G. Espera-se

observar que o modelo treinado com o conjunto completo dos dados apresente melhorias nesta tarefa de classificação dos cenários, em comparação ao outro que foi treinado com o subconjunto dos dados (apenas métricas da rede 5G).

Neste contexto, pode-se também citar outras contribuições complementares:

- A demonstração da implantação de uma rede 5G e de gNodeB simulados;
- A demonstração da conexão de atores simulados, especificamente dispositivos IoT, a partir de gNodeB;
- A demonstração de um método integrado de coleta de métricas da rede 5G e do ecossistema *IIoT*;
- Uma visão do cenário atual de ferramentas de código aberto para implantação de um ecossistema conjunto envolvendo núcleo 5G e dispositivos *IoT* de uma fábrica inteligente.

1.4 Estrutura do documento

O restante deste documento está organizado da seguinte forma:

- No Capítulo 2, são apresentados conceitos-chave para a fundamentação teórica deste trabalho.
- No Capítulo 3 os trabalhos relacionados à temática apresentada neste trabalho, seus destaques e uma comparação entre eles.
- No Capítulo 4, a metodologia proposta para este trabalho é apresentada.
- No Capítulo 5, são detalhadas as ferramentas utilizadas para a composição do experimento, bem como os detalhes para implantação dos componentes e as validações do método.
- No Capítulo 6, as conclusões do presente trabalho são apresentadas.

Capítulo 2

Fundamentação Teórica

Este capítulo é composto por duas seções, a primeira trata dos serviços e funções de uma rede móvel 5G, medições de desempenho e gerenciamento de redes 5G. São incluídos detalhes também sobre o desempenho da comunicação entre dispositivos IoT. Nesta seção são descritos os serviços e funções que compõem uma rede móvel 5G e as métricas que são comumente utilizadas para avaliação do estado de uma rede móvel 5G. São descritos também os parâmetros utilizados para medição do desempenho da comunicação entre dispositivos IoT.

A seção subsequente trata de conceitos de redes neurais e aprendizado de máquina no contexto de classificação multiclasse em dados de séries temporais. São discutidas as características específicas de redes RNN e LSTM, além do resgate de algumas técnicas de modelagem de redes neurais e avaliação destes modelos.

2.1 Serviços e Funções de uma Rede Móvel 5G

Em uma rede móvel 5G, o Núcleo 5G refere-se a um conjunto de funções e serviços essenciais para a gestão da rede [7]. Estas funções incluem a autenticação de usuários e a garantia de conectividade contínua e de alta velocidade, ao passo em que os usuários reconectam entre as diferentes células da rede. Outras funções também envolvem o gerenciamento eficiente de sessões de dados, permitindo a comunicação sem interrupções e a transferência de grandes volumes de dados. Além disso, o Núcleo 5G e seus componentes desempenham um papel crucial na segmentação de rede, proporcionando a capacidade de criar e gerenciar múltiplas

redes virtuais para atender a diferentes requisitos de serviço e cliente. Os componentes e serviços mais importantes e suas funções estão destacados na [Tabela 2.1](#).

Tabela 2.1: Componentes do Núcleo 5G

Sigla	Componente	Função
AMF	Access and Mobility Management Function	Gestão de acesso e mobilidade dos usuários na rede, incluindo autenticação, autorização e rastreamento da localização dos dispositivos.
SMF	Session Management Function	Gestão de sessões. Estabelece, modifica e encerra sessões de rede para os usuários.
UPF	User Plane Function	Realiza encaminhamento de dados do usuário, função essencial para o processamento de tráfego de dados e roteamento para diferentes destinos na rede.
NSSF	Network Slice Selection Function	Gestão das fatias de rede, ou seja, conjuntos de recursos de rede configurados para atender às necessidades específicas de um determinado tipo de serviço ou cliente.
UDM	Unified Data Management	Gerencia os dados de assinantes e autenticação. Substitui o HSS (<i>Home Subscriber Server</i>) usado nas redes 4G/LTE.
AF	Application Function	Realiza interação com o SMF fornecendo informações sobre aplicativos para influenciar o comportamento da sessão.
NEF	Network Exposure Function	Expõe as capacidades da rede para aplicativos de terceiros.

Continua na próxima página

Tabela 2.1: Componentes do Núcleo 5G (Continuação)

NRF	Network Repository Function	Funciona como um registro de serviço para os componentes do Núcleo 5G. Mantém um registro da disponibilidade dos serviços da rede e forma de acesso.
AUSF	Authentication Server Function	Responsável pela autenticação dos usuários na rede.
PCF	Policy Control Function	Define e gerencia as políticas de controle de tráfego e uso da rede.
DN	Data Network	Representa a rede externa, como a Internet ou redes corporativas, a qual os usuários desejam se conectar.
SEPP	Security Edge Protection Proxy	Protege a troca de informações entre as redes, assegurando a segurança e a privacidade dos dados.
NWDAF	Network Data Analytics Function	Coleta e analisa dados para otimização e perspectivas sobre a rede.

Entre estes componentes do Núcleo 5G, pode-se destacar para este trabalho: *Access and Mobility Management Function (AMF)*, *Session Management Function (SMF)*, *User Plane Function (UPF)* e *Network Data Analytics Function (NWDAF)*. O *AMF* é o serviço responsável por realizar a gestão da conectividade e acessibilidade dos dispositivos de usuário (em inglês *User Equipment, UE*), em questões específicas como a autenticação e autorização. A gestão da sessão de comunicação fica a cargo da *SMF*. Por sua vez, o *UPF* é responsável pelo encaminhamento e tratamento dos dados do usuário, atuando como ponto de ligação para o tráfego de dados, conectando a rede 5G às redes de dados externas.

Externo ao Núcleo 5G, a estação de base *gNodeB* (em inglês *Next Generation Node B, gNB*) é uma parte principal da implementação do componente *Radio Access Network (RAN)*, este comunica-se com o conjunto Núcleo 5G por meio de interfaces definidas para prover

acesso aos dispositivos móveis (UE), conforme visualizado na [Figura 2.1](#). A comunicação entre *gNB* e *AMF* se dá através da interface *N2*, a partir da qual ocorre o processo para sinalização de controle e gestão de sessão (envolvendo *SMF* a partir da interface *N4*). Estabelecida a sessão, *gNB* conecta-se ao *UPF* através da interface *N3* para realizar o tráfego de dados do usuário, consolidando a sessão para o dispositivo (*PDU Session*). Esta separação de responsabilidades com a criação de um plano de dados (em inglês *data plane*) e um plano de controle (em inglês *control plane*) é uma estratégia eficiente para otimizar a gestão de tráfego de dados e alocação de recursos da rede.

Para dar suporte à análise contínua dos serviços que compõem o Núcleo 5G e também do estado da rede, a função *NWDAF* comunica-se com diversos desses serviços, coletando dados e provendo análises precisas. *NWDAF* interage em especial com *Policy Control Function (PCF)* e *Network Slice Selection Function (NSSF)*, que lidam respectivamente com o controle de políticas e as partições de rede. Esta interação é crucial para a integração inteligente destes componentes e para promover a autonomia e adaptabilidade da rede 5G. A flexibilidade da *NWDAF* no processamento e tratamento dos dados permite a agregação e análise de informações de uma variedade de serviços e aplicações, inclusive aqueles que estão fora do âmbito tradicional do Núcleo 5G. Isso abre possibilidades para a integração de métricas de sistemas de comunicação utilizados pelos dispositivos da rede para auxiliar na gestão autônoma desta, provendo informações adicionais que podem ter grande importância neste âmbito, objetivo explorado neste trabalho.

Além disso, é importante salientar que as definições e os padrões relacionados à *NWDAF*, fornecidos pela 3GPP, estão em constante evolução. Essa dinâmica reflete o compromisso contínuo com a inovação e a melhoria, garantindo que a *NWDAF* permaneça alinhada com as necessidades emergentes e as tecnologias em desenvolvimento no ecossistema 5G. Portanto, a *NWDAF* é mais do que um serviço de análise de dados; ela é um componente vital para a escalabilidade, flexibilidade e sustentabilidade da rede 5G, auxiliando na adaptação contínua para atender às demandas futuras, em especial no cenário industrial.

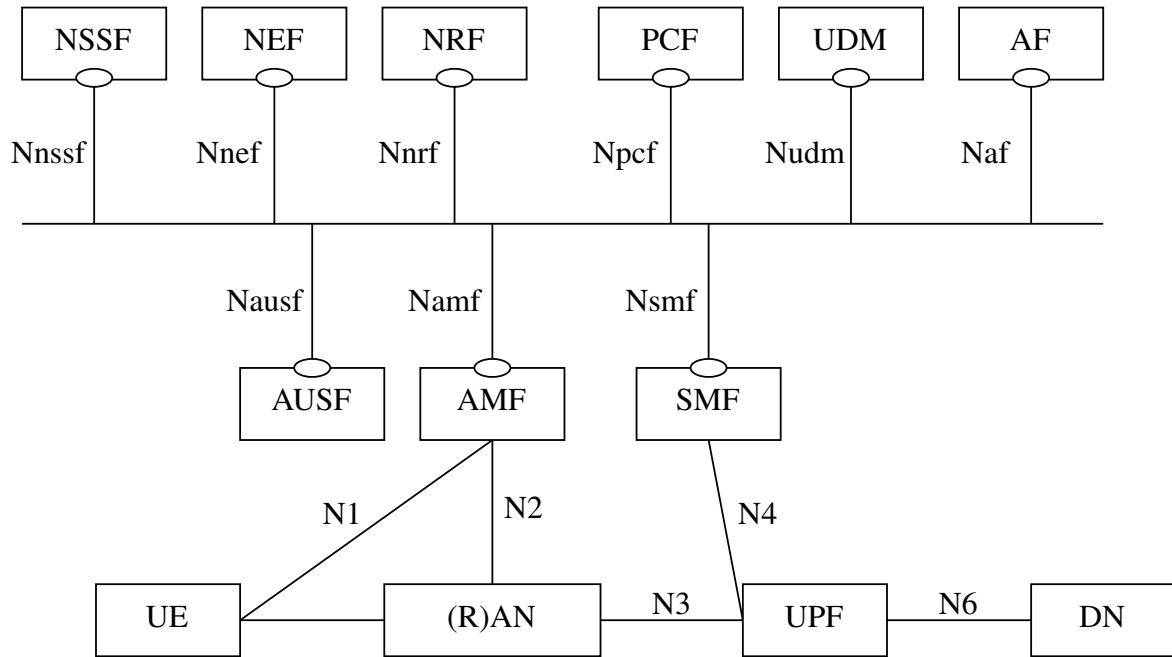


Figura 2.1: Arquitetura simplificada de serviços 5G e suas interfaces [1]

2.2 Métricas de Rede 5G e de Comunicação dos Dispositivos IoT

As métricas de avaliação de desempenho de uma rede 5G e também do ecossistema *IIoT* são fundamentais para compreender e otimizar o funcionamento destes. As métricas de destaque para observação neste trabalho são:

- A latência, que mede o tempo de transporte de um pacote de dados de um ponto a outro na rede, sendo crucial para aplicações que requerem respostas em tempo real;
- A largura de banda, métrica vital que se refere à capacidade máxima de transferência de dados de uma rede, indicando a quantidade de dados que pode ser transmitida em um período específico;
- A perda de pacotes, que denota a quantidade de pacotes de dados que não alcançam seu destino, pode significativamente degradar a qualidade da comunicação, sendo um indicativo de problemas como congestionamento na rede.

Além destas, é importante considerar as métricas relacionadas às aplicações e processos que implementam estes serviços, portanto o uso de memória e *CPU* são importantes para

considerar a carga total deste sistema.

2.3 Redes Neurais e Aprendizado de Máquina

Dado o principal objetivo proposto neste trabalho que é a observação quantitativa que a combinação de dados do ecossistema IoT e métricas da rede, na tarefa de classificação de problemas de rede, faz-se necessário ressaltar a natureza temporal e sequencial destes conjuntos de dados.

Portanto, trata-se de um problema de classificação multiclasse em séries temporais, o qual tem sido considerado pela literatura um desafio nos últimos anos na área de aprendizado de máquina [16]. Entre os motivos pelos quais este tipo de problema de classificação é considerado desafiador, destaca-se que as características de interesse nos dados só podem ser percebidas na observação contínua das amostras. Ademais, em geral há interdependência entre cada parâmetro de cada amostra pontual, pelo do caráter inerentemente temporal deste tipo de dado. Por estes motivos os dados de séries temporais devem ser divididos em trechos com amostras contínuas para que se possa observar e preservar estas características.

As redes neurais, porém, têm-se mostrado eficazes nesta tarefa, mesmo considerando dados com o caráter temporal e sequencial [33], portanto serviram de inspiração para a proposição dos modelos utilizados no experimento deste trabalho. Estes modelos foram treinados com os diferentes conjuntos de dados para observar a validade das hipóteses estabelecidas anteriormente. Em resumo, a intenção é proporcionar uma comparação entre modelos que classificam cenários de problemas de rede, para indicar a importância dos dados coletados do ecossistema IoT neste problema de classificação.

2.3.1 Redes Neurais RNN e LSTM

Ao lidar com dados de séries temporais, as redes neurais dos tipos *Recurrent Neural Network* (RNNs) [37] e *Long Short-Term Memory* (LSTM) [14] oferecem vantagens significativas para tarefas de classificação [16]. Diferentemente das redes neurais que processam os dados apenas na direção dos nós de entrada para os nós de saída (redes *feed-forward*), em que as amostras são processadas de forma independente, as RNN e LSTM foram arquitetadas para a capturar padrões e dependências temporais contidas nas sequências das amostras de dados.

Esta característica permite modelar relações de caráter temporal complexas, tornando-as ferramentas adequadas para tarefas como a classificação de cenários de problema de rede, como proposto no experimento deste trabalho. Ademais, a capacidade de aprender dependências de longo termo e eliminar informações com pouca relevância, torna as LSTM mais robustas para lidar com ruído e irregularidades presentes nos dados analisados, em comparação com as RNN. Em resumo, as RNN e LSTM foram arquitetadas para lidar com entradas de sequências com tamanho variável, e ainda manter informação contextual das sequências de tempo, resultando em alto desempenho para tarefas que tratam de modelagem que requer dinâmica temporal.

Observando em detalhes a estrutura única dos LSTM, destacam-se as portas de esquecimento (em inglês, *forget gate*), entrada e saída, que possibilitam a manutenção e regulação eficaz de informações ao longo de períodos extensos (Figura 2.2). A introdução destes componentes na arquitetura do LSTM permite reter informações relevantes e descartar as não essenciais a partir da combinação de uma memória de curto prazo que influencia na ativação de outra memória de longo prazo, propagando as características relevantes das sequências das amostras de forma eficaz. Portanto, este tipo de modelo é considerado excepcionalmente apto para a tarefa de classificação em séries temporais, já que a relevância de uma amostra pode ser influenciada por eventos anteriores e distantes na sequência dos dados.

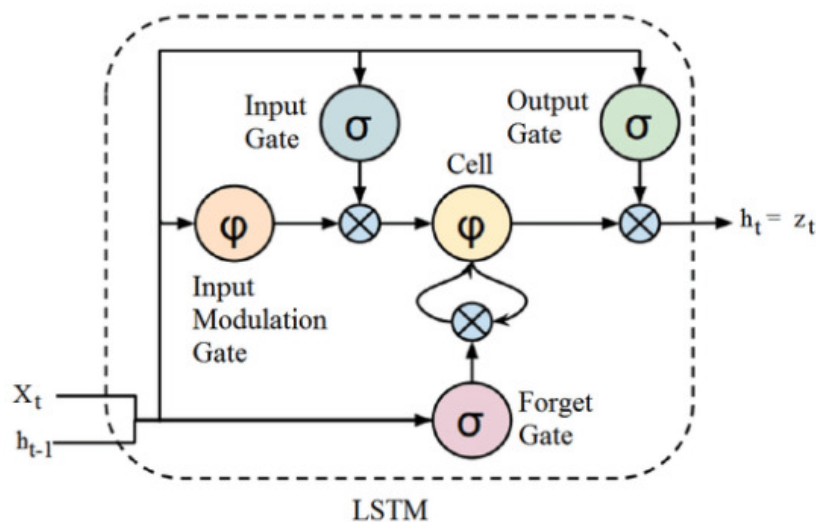


Figura 2.2: Arquitetura da célula LSTM [10]

2.3.2 Treinamento e avaliação de Modelos de Rede Neural

Nesta subseção apresenta-se um breve resumo sobre as técnicas utilizadas para arquitetar as camadas do modelo de rede neural proposto no experimento deste trabalho, cada item da lista representa um tipo de camada e seu propósito:

- **Regularização L2:** Adiciona um termo de penalidade ao custo de treinamento, baseado na magnitude dos pesos dos neurônios, penalizando pesos grandes e promovendo um modelo mais simples e menos propenso ao sobreajuste. Esse termo de regularização tende a dispersar o erro sobre todos os neurônios, favorecendo pesos menores e mais distribuídos;
- **Normalização em Lote:** Normaliza as entradas de cada camada para ter uma média de zero e um desvio padrão de um, reduzindo o deslocamento interno da covariância (em inglês *Internal Covariate Shift*) e permitindo o uso de taxas de aprendizagem mais elevadas, além de ser menos sensível à inicialização dos pesos;
- **Dropout:** Ajuda a prevenir o sobreajuste (overfitting) ao descartar aleatoriamente unidades (neurônios) durante o treinamento, contribuindo para uma aprendizagem mais robusta e menos dependente de padrões específicos nos dados de treinamento.

O treinamento de modelos com grande número de parâmetros e camadas, como os propostos no experimento deste trabalho, demanda grandes recursos computacionais e de tempo. Para otimizar este processo e atender às restrições de recursos, destacam-se as técnicas utilizadas realizar o treinamento dos modelos do experimento:

- **Early Stopping:** Técnica que interrompe o treinamento assim que o desempenho do modelo no conjunto de validação começa a piorar, evitando o sobreajuste e garantindo que os modelos mantenham a capacidade de generalização;
- **Hyperbanding:** Consiste em definir um limiar para cada hiperparâmetro do modelo de rede neural e utilizar uma distribuição bayesiana para amostrar diferentes combinações dos hiperparâmetros. Avaliando o desempenho do modelo em cada uma dessas configurações. Esta técnica escolhe automaticamente o conjunto de hiperparâmetros que resulta em melhor desempenho, otimizando o treinamento do modelo e reduzindo a necessidade de experimentos manuais.

Para a avaliação de modelos de aprendizado de máquina, no contexto de classificação multiclasse como no experimento proposto neste trabalho, faz-se necessária a observação de métricas como acurácia, precisão e sensibilidade de forma individual para cada classe. Além disso, deve-se analisar as curvas de decaimento da perda do modelo e do aumento da acurácia durante o treinamento a fim de identificar a possibilidade da ocorrência de sobreajuste (em inglês, *overfitting*) ou subajuste (em inglês, *underfitting*), aspectos que interferem na confiabilidade do treinamento destes modelos.

Outra abordagem que pode ajudar na avaliação do desempenho destes modelos na tarefa de classificação multiclasse é a observação da curva característica de operação do receptor (em inglês *Receiver Operating Characteristic, ROC*). A ideia é considerar cada classe como um problema de classificação binária (um contra todos), no qual o modelo precisa distinguir entre a classe alvo e todas as outras classes. A curva *ROC* é então calculada para cada um desses subproblemas, comparando a sensibilidade com a taxa de falsos positivos. Essa abordagem permite avaliar como o modelo se comporta em relação à cada classe individualmente, facilitando a comparação entre a capacidade de classificação para cada classe individualmente.

2.4 Considerações Finais

Neste capítulo foram revisados conceitos importantes da arquitetura de uma rede 5G, com foco nos serviços que a compõem. Foram destacados conceitos sobre as métricas apresentadas para avaliação do desempenho da rede e também do sistema de comunicação de dispositivos IoT. Foram destacados também conceitos sobre redes neurais e aprendizado de máquina voltados para o problema de classificação multiclasse em dados de séries temporais.

Foram abordados conceitos sobre redes neurais específicas (RNN e LSTM) na tarefa de classificação citada. Além disso, foram explorados conceitos sobre arquitetura desses modelos, treinamento e avaliação do desempenho para esta tarefa. Estes conceitos são essenciais para a plena compreensão do desenvolvimento deste trabalho e do experimento proposto para observação da importância dos dados coletados na tarefa de classificação de cenários de problemas de rede.

Capítulo 3

Trabalhos Relacionados

Neste capítulo são abordados trabalhos com proposta ou temática tangentes àquela proposta neste trabalho. São apresentados para cada trabalho um breve resumo com foco no tema de interesse.

3.1 Redes Privadas 5G em Fábricas Inteligentes

Com o intuito de observar a motivação para a adoção específica de redes 5G no cenário de fábricas inteligentes e as funcionalidades que estas podem prover no âmbito de monitoramento e gestão da rede, foram revisados os trabalhos a seguir.

[Maman et al. \[21\]](#), reforçam o papel crucial das redes privadas 5G para a consolidação das fábricas inteligentes. São elencadas tecnologias indispensáveis para atender a exigências rigorosas e heterogêneas em termos de confiabilidade, latência e flexibilidade, providas por este cenário industrial. Entre estas tecnologias, estão partição de rede (em inglês, *Network Slicing*) e estratégias inteligentes de orquestração da rede a partir do monitoramento contínuo da rede associado à técnicas de inteligência artificial. [Brown](#), em [\[8\]](#), reitera o desempenho superior das redes móveis 5G em comparação a tecnologias anteriores como *LTE* e *WI-FI*, nos termos de confiabilidade, baixa latência, suporte a quantidades massivas de dispositivos e banda de rede. Neste trabalho, destaca a flexibilidade e confiabilidade como pontos importantes para abarcar as aplicações industriais das fábricas inteligentes. [Sachs e Landernäs](#), em [\[34\]](#), ilustram os casos de uso de redes móveis 5G privadas no cenário industrial, destacando sua importância em possibilitar a comunicação entre dispositivos e humanos em

baixa latência neste contexto, em especial nas tarefas de controle e monitoramento remotos, possibilitando maiores graus de automação.

3.2 Implementações de Código Aberto para Núcleo 5G

A realização de estudos que envolvem um Núcleo 5G pode eventualmente deparar-se com barreiras onerosas como a aquisição e configuração de equipamentos sofisticados, além dos trâmites legais para obtenção das permissões necessárias para o uso da faixa de radiofrequência para estes equipamentos. Portanto, o uso de Núcleo 5G simulado é uma solução que permite contornar esses requisitos em troca de alguns comprometimentos.

Em [31], Reddy et al. discorrem sobre as alternativas de código aberto para implementação do Núcleo 5G, destacando os projetos *Free5GC*[13], *OAI 5G CN*[17] e *Open5GS*[26]. Associados a estes Núcleos 5G, os autores citam também implementações para *RAN* e *UE* como *UERANSIM*[43], *My5g-RANTester*[40] e *omec-gnbsim*[24], para realizar a conexão com os serviços das redes 5G simuladas que foram implantadas. Como resultado da análise quantitativa e qualitativa, observaram que as implementações *Free5GC* e *Open5GS* se assemelham em quesitos de desempenho e características de implantação, superando o projeto *OAI 5G CN*. Em relação aos simuladores *RAN* e *UE*, os autores optaram pelo *UERANSIM*, a alternativa mais flexível entre os projetos citados, por prover uma interface de rede para fazer o papel do *UE* a partir da qual se pode conectar qualquer tipo de aplicação para realizar testes na rede.

Os resultados obtidos por este estudo inspiraram as escolhas para implantação dos componentes do experimento descrito nos [Capítulo 4](#) e [Capítulo 5](#), as limitações encontradas estão descritas na [Subseção 5.3.3](#).

3.3 Robot Operating System 2 para Dispositivos IoT

Dado que o experimento proposto neste trabalho objetiva observar a combinação de dados de rede 5G e da comunicação de dispositivos IoT, notou-se a necessidade de explorar implementações capaz de simular dispositivos IoT e também os protocolos utilizados por estes para comunicação, objetivando suprir a necessidade de construir um ambiente simulado para

o experimento. Portanto, foram revisadas as ferramentas e trabalhos a seguir.

O *ROS2* (*Robot Operating System 2*) é uma conjunto de ferramentas avançado e de código aberto, destinado ao desenvolvimento e à simulação de sistemas robóticos e dispositivos *IoT*. O *ROS2* se destaca na simulação e integração de dispositivos *IoT*, especialmente em ambientes de fábricas inteligentes, através do emprego de *middlewares* de comunicação. Estes *middlewares* possibilitam a interoperabilidade entre uma variedade de dispositivos e sistemas, facilitando o fluxo de dados e comandos.

Urrea e Kern, Abdelatti e Sodhi demonstram em seus respectivos trabalhos, [44] e [3], a eficácia do *ROS2* para a implementação de ambientes simulados de dispositivos *IoT*. Em [3], a implantação de uma fábrica inteligente em escala laboratorial se destacou, não só como uma experiência educacional valiosa, mas também como um ambiente de testes inovador para aprimorar técnicas de manufatura inteligente.

Ademais, o uso do *ROS2* para a implantação de um laboratório virtual para engenharia de sistemas de controle ilustra como esta plataforma pode facilitar a integração entre componentes físicos e digitais, provendo a capacidade de aquisição e visualização de dados em tempo real. Estas características realçam o *ROS2* como uma ferramenta versátil e robusta para atender às necessidades de simulação e desenvolvimento em ambientes industriais *IoT*.

3.4 Aprendizado de Máquina para Análise de Dados de Rede 5G

Existem diversos exemplares na literatura que tratam da detecção ou classificação de problemas de rede em cenários de grande escala, *big data*. Nesta seção foram explorados trabalhos que lidaram com algum problema de classificação de dados que possuem caráter temporal utilizando diversas abordagens de redes neurais.

Ahmed et al. em [4] discorrem sobre a detecção de problemas de rede e a revisão de métodos utilizados para predição e classificação destes. Porém no contexto de redes neurais, trata apenas de redes neurais convolucionais (em inglês *Convolutional Neural Networks*, CNN) que são do tipo *feed forward*. Como explicitado no Capítulo 2, este tipo de rede neural não apresenta grande capacidade de generalização para padrões temporais pela desvantagem de não possuir o caráter de memória, quando comparado com outros tipos de modelo como

RNN e LSTM. É revisado ainda no trabalho em questão que técnicas de sumarização em *big data*, como é o caso dos dados de rede, são complexas e não promovem grandes melhorias para generalização de padrões neste tipo de dado temporal.

[Ren et al.] em [32], tratam de uma abordagem com a combinação de redes neurais *CNN* e um algoritmo de resíduo espectral (em inglês Spectral Residual, *SR*) na tentativa de aplicar transformações e melhorar os dados de rede, com o intuito de obter melhor desempenho no modelo de rede neural *CNN*. Os resultados deste trabalho são promissores, porém foi explorada a problemática de classificação binária, na ocorrência ou não do problema de rede. Que difere do cenário proposto neste trabalho que é a classificação multiclasse envolvendo estes dados de série temporal no âmbito de rede.

Aproximando-se da abordagem utilizada neste trabalho, [Zhang et al.] em [49], discorrem sobre um modelo com caráter recorrente para a resolução de problemas de detecção e classificação de anomalias em dados de uma usina elétrica. O modelo proposto no trabalho possui uma arquitetura complexa e robusta, objetivando lidar com ruídos inerentes a estes dados e o processo de coleta deles. Este modelo foi capaz de generalizar correlações entre os dados coletados dos sensores na usina e o caráter temporal inerente a estes. Em resumo, o trabalho observado trata de um conjunto dados diferente do proposto para o experimento deste. Porém, pode-se notar características similares ao objetivo proposto deste trabalho observando a similar dependência temporal dos dados e o problema de classificação elencado.

Recapitulando o que foi mencionado anteriormente, o serviço *NWDAF* do Núcleo 5G desempenha papel importante para o quesito de automação e reconfiguração automática da rede por ser responsável pela coleta e análise de dados de desempenho. No contexto desta função do Núcleo 5G, [Sevgican et al.] em [38] realizaram experimentos sob dados sintéticos de um Núcleo 5G, a partir dos quais comparam diferentes modelos de rede neural na tarefa de predição da carga de rede, elegendo o modelo *Long Short-Term Memory (LSTM)* como aquele com melhor desempenho. Ainda neste trabalho, os autores realizam comparação entre modelos de regressão linear e *XGBoost* para a tarefa de detecção de anomalias, ou seja, uma tarefa de classificação semelhante à proposta por este trabalho. Apontando o modelo *XGBoost* como candidato com melhor acurácia para esta tarefa.

Ressalta-se, porém, que os dados utilizados em [38] são sintéticos e não incluem informações sobre a comunicação dos dispositivos, apenas dados de fluxo da rede. Indo além,

acredita-se que um modelo *LSTM*, por sua característica inerente de observar padrões em dados de caráter temporal, como aqueles relacionados ao estado da rede, pode realizar a tarefa de classificação destes cenários de anomalia de rede com ganhos em acurácia, comparação que não foi explorada no trabalho citado anteriormente.

3.5 Considerações Finais

A revisão destes trabalhos relacionados inspirou a escolha dos componentes a serem utilizados no experimento proposto no trabalho. Além disso, notou-se a oportunidade de explorar diferentes abordagens para resolução do problema de classificação previamente descrito, especialmente no âmbito da aplicação de técnicas de aprendizado de máquina. A análise detalhada destes estudos reforça a ideia de que a combinação de dados de rede 5G e dispositivos *IoT* pode ser benéfica em análises mais precisas e eficientes que viabilizam a alocação eficiente e automática de recursos da rede, otimizando o tráfego de dados e reduzindo do tempo de inatividade em ambientes industriais complexos, nos quais a precisão e a eficácia na identificação de cenários de problemas de rede são críticas para a tomada de decisão.

Capítulo 4

Metodologia

Com base nas questões de pesquisa (QP) elencadas anteriormente e na revisão bibliográfica realizada (Capítulo 3), foram elencadas as seguintes hipóteses:

- **Hipótese 1 (H1):** A partir de um serviço containerizado na infraestrutura de funções de rede (*network functions*), é possível realizar a coleta de métricas do núcleo 5G e do ecossistema *IIoT* de forma sincronizada e sem gerar sobrecarga na comunicação destes;
- **Hipótese 2 (H2):** A utilização dados integrados de métricas da rede 5G e de um ecossistema *IIoT*, presente em fábricas inteligentes, é capaz de aprimorar a capacidade de classificação de cenários de problemas de rede utilizando modelos de rede neural. Deve-se observar melhoria em métricas como acurácia e precisão, na comparação entre um modelo treinado o subconjunto dos dados da rede e um modelo treinado com o conjunto completo de dados da rede e do ecossistema *IIoT*.

Portanto, com intuito de avaliar as hipóteses elencadas, foi aplicada uma metodologia de engenharia proposta por Kontio et al. em [18], que possui como objetivo o desenvolvimento, avaliação e aprimoramento de uma solução. Essa solução foi testada e incrementada até que nenhuma outra melhoria pudesse ser viável de acordo com as limitações de tempo e outros recursos, nas implementações do ecossistema de rede e dispositivos e no treinamento e avaliação dos modelos a serem comparados. Os detalhes de implantação e limitações estão explicitados no Subseção 5.3.3. Esta metodologia de engenharia está subdividida em cinco fases sequenciais (Figura 4.1) :

1. **Observação** : Nesta fase foi observado o estado da arte e da prática com o objetivo de identificar oportunidades de atuação e possíveis soluções. A revisão dos trabalhos relacionados pode ser observada no [Capítulo 3](#) inspirou os objetivos declarados na [Seção 1.2](#). Foram exploradas algumas ferramentas de código aberto, como descrito na [Seção 4.1](#);
2. **Proposição** : Fase em que houve a formulação de um problema a ser resolvido, consolidando um conjunto de teorias, questões de pesquisa e hipóteses. A partir desta problemática, foi proposta a arquitetura de um experimento, detalhado na [Seção 4.1](#);
3. **Análise** : Fase em que foi posto em prática o experimento que foi definido a partir das fases de observação e proposição;
4. **Avaliação** : Nesta fase foram realizados testes e avaliação do experimento com os modelos através de medições e análises dos resultados obtidos que podem ser observados na [Seção 5.3](#). Nesta, também estão inclusas as etapas de exploração, experimentos, e revisão adicional do experimento proposto e dos modelos avaliados;
5. **Implantação** : Fase em que ocorreu a organização do experimento de forma implantável e reproduzível. Junto ao experimento foram definidos mecanismos para coleta e análise das estatísticas de uso, a fim de promover a reprodutibilidade¹.

¹github.com/AramisAraujo/5GIoTMetrics

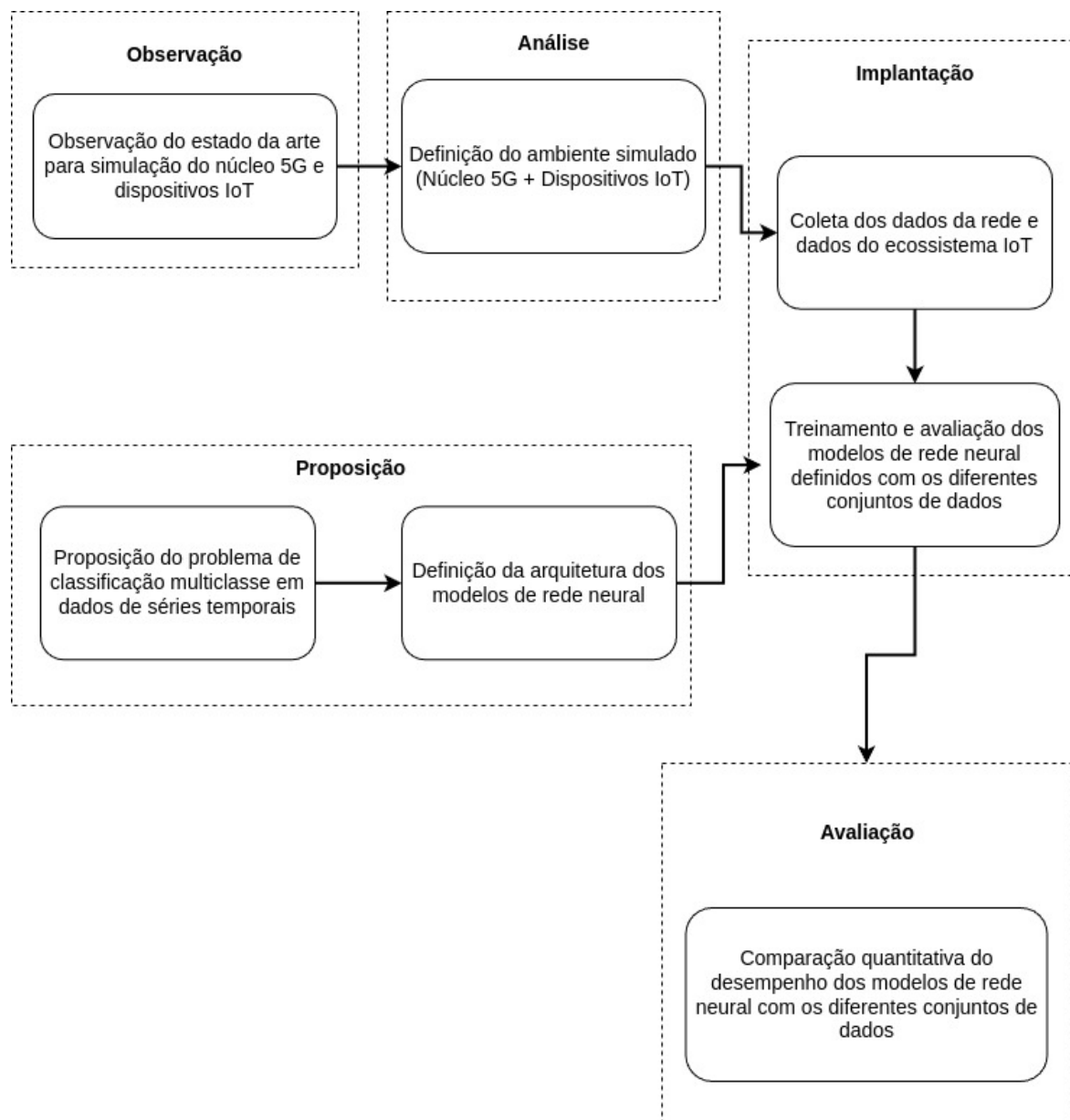


Figura 4.1: Fases da metodologia utilizada neste trabalho

Revisitando o objetivo deste trabalho e as questões de pesquisa **QP1**, **QP2** e **QP3**, que envolvem a avaliação do impacto da integração de dados advindos do ecossistema *IoT* em uma *Smart Factory*, com os dados de monitoramento coletados do Núcleo 5G, para a identificação de cenários de problemas de rede, neste capítulo apresenta-se a caracterização do experimento realizado para avaliação das hipóteses **H1** e **H2**, qual consistiu nas etapas de:

1. Implantação de um Núcleo 5G simulado para realizar a comunicação entre os dispositivos.

2. Implantação de gNodeB simulados para conectar os dispositivos IoT à rede 5G.
3. Implantação de um cenário de testes a partir do qual os dispositivos IoT simulados comunicam-se através da rede 5G.
4. Implantação de um sistema para coleta integrada de métricas da rede 5G e da comunicação dos dispositivos (**H1**).
5. Treinamento de modelos de classificação para cenários de problemas de rede (**H2**), detalhados na [Seção 4.3](#).
6. Avaliação e comparação do desempenho entre esses modelos na tarefa de classificação (**H2**).

Estas etapas são detalhadas nas seções subsequentes.

Objetivando implementar um sistema capaz de satisfazer as hipóteses **H1** e **H2**, foram definidos os seguintes requisitos:

- **Requisito 1 (R1)**: Os componentes do sistema devem ser projetos de código aberto, a fim de promover a reprodutibilidade do experimento;
- **Requisito 2 (R2)**: A comunicação entre os componentes do sistema deve ocorrer estritamente a partir da rede 5G, para que os dados coletados possam refletir o comportamento do experimento nesta rede;
- **Requisito 3 (R3)**: Os dados devem ser coletados a partir do núcleo 5G e ecossistema *IoT* de forma que não gere sobrecarga na comunicação destes sistemas;
- **Requisito 4 (R4)**: Os modelos devem ser treinados em condições equivalentes, com conjuntos de dados balanceados e estratificados.

4.1 Experimento

Para atender ao objetivo de avaliar o impacto da integração destas informações, notou-se a necessidade de simular um cenário de rede 5G aplicada ao âmbito industrial. Foram consideradas algumas opções de implementações de código aberto para os serviços do Núcleo

5G (**R1**). Breves revisões dos projetos *Aether* [25], *Open5GS* [26] e *Free5GC* [13] levaram a observar que ainda há lacunas nas implementações em relação às especificações, ressaltado pelo fato de que, até o momento, alguns serviços do Núcleo 5G não estão presentes ou maduros o suficiente para utilização em nenhum dos candidatos.

Ressalta-se também que algumas funcionalidades relacionadas à qualidade do serviço (em inglês *Quality of Service*, QoS) não estão implementadas em sua totalidade em nenhum dos projetos mencionados, portanto não puderam ser explorados neste trabalho. Outro ponto notoriamente importante é a ausência da implementação do serviço NWDAF (*Network Data Analytics Function*), responsável pela coleta e análise dos dados dos serviços do Núcleo 5G. Logo, o passo de coleta e análise dos dados precisou ser implementado para execução deste experimento.

Apontadas as observações, foi escolhido o projeto *Open5GS* para condução do experimento deste trabalho, por prover uma implementação mais completa dos serviços em comparação aos outros, além das métricas de rede disponíveis por meio de uma interface para comunicação com um *scraper*, no caso o *Prometheus* [29] (alinhado com o requisito **R3**). Os serviços disponíveis nesta implementação de Núcleo 5G são: **MME**, **SGW-C**, **SMF**, **AMF**, **SGW-U**, **UPF**, **HSS**, **PCRF**, **NRF**, **SCP**, **AUSF**, **UDM**, **PCF**, **NSSF**, **BSF** e **UDR**. Destes, as métricas dos seguintes serviços estavam disponíveis para coleta: **AMF**, **MME**, **PCF**, **SMF** e **UPF**.

Após consideradas as alternativas para implementação do núcleo 5G, foram observadas as opções para implementação da estação base (*gNodeB*), fazendo o papel de *RAN* na arquitetura da rede. Existem poucas opções de projetos de código aberto, ainda mais considerando a necessidade de conectar dispositivos e aplicações arbitrárias à rede 5G para realização dos experimentos de comunicação.

O *UERANSIM* [43] destaca-se como projeto de código aberto (**R1**) amplamente utilizado em experimentos que necessitam de conexão entre *RAN* e *UPF*, visto que simula a *gNodeB* e também um *UE*, com a provisão de uma interface de tunelamento *TUN* que direciona os pacotes a partir do protocolo de transporte *UDP*. A partir desta interface é possível ter acesso à internet pública (neste contexto, definida como *DN*) por meio da rede 5G. Há, porém, algumas funcionalidades que não estão disponíveis no *UERANSIM* em relação à implementação das funções de *QoS*, que também impediram de explorar outros cenários no

experimento.

De forma a implantar os atores do experimento, ou seja, aqueles responsáveis por prover os dados dos dispositivos *IoT* e gerar carga na rede, foi utilizado o *ROS2*[20] (*Robot Operating System 2*). O *ROS2* é um conjunto flexível de ferramentas de código aberto (em alusão ao requisito **R1**) destinado a facilitar o desenvolvimento de software para robótica. Além de possibilitar a simulação de dispositivos *IoT*, outra de suas principais funções é a provisão de um sistema robusto de comunicação entre estes.

Tal sistema de comunicação garante a interação eficiente entre diferentes dispositivos, sejam eles reais ou simulados. Um componente essencial neste sistema é o *FastDDS*[12], um *middleware* de comunicação de alto desempenho que utiliza o protocolo *DDS* (*Data Distribution Service*), implementação padrão nas versões mais recentes do *ROS2*. Este componente otimiza a transmissão de dados em tempo real, característica que é essencial visando o cenário industrial de dispositivos *IoT*. Ele garante a troca eficiente de mensagens mesmo em redes complexas e distribuídas.

Para realizar um *benchmark* deste sistema de comunicação e coletar métricas para a análise, foi utilizado o *ApexAI performance_test*[28]. Esta ferramenta foi projetada para testar e avaliar o desempenho de sistemas de *middleware* de comunicação utilizados pelo *ROS2*. O foco principal da ferramenta *performance_test* é avaliar o desempenho em aspectos críticos como latência, vazão (em inglês *throughput*), uso de CPU e memória em diferentes condições de rede e carga de trabalho.

Dentre as outras ferramentas disponíveis para avaliação desses *middleware* de comunicação, esta se destacou ao oferecer um teste de comunicação entre processos distribuídos, visto que as outras ferramentas disponíveis só realizavam comunicação entre processos no mesmo sistema. Por isto, esta ferramenta adequou-se à proposta do experimento da comunicação envolvendo dois atores (*UE*), implantados em máquinas virtuais distintas e realizando comunicação através da rede 5G. Além disso, na arquitetura da ferramenta o processo de comunicação está desacoplado do processo de coleta e análise dos resultados, de forma que não gera sobrecarga nos sistemas durante a comunicação, permitindo preservar as características definidas anteriormente para o experimento (**R3**).

Esta ferramenta possibilitou simular a comunicação entre dois dispositivos *IoT*, realizando troca de mensagens de tamanho estipulado e em intervalos específicos, gerando es-

tatísticas importantes como latência observada, banda utilizada, perda de pacotes e outras. Neste teste de desempenho, dois processos instanciados em diferentes máquinas virtuais realizaram troca de mensagens a partir da interface de tunelamento provida para cada um, simulando *UE*. O processo de comunicação se desenvolveu em esquema de *round-trip*, ou seja, cada mensagem foi transmitida do emissor ao receptor e em seguida devolvida ao emissor para cálculo das métricas de latência, perda de pacotes e outros.

A estrutura das mensagens consistiu em 1024 bytes, dos quais 8 foram utilizados para um identificador, 8 para registro do tempo de emissão da mensagem e o restante para uma mensagem arbitrária. Os processos se comunicaram através do protocolo *FastDDS*[\[12\]](#), utilizando um servidor de descoberta (*discovery server*) que foi instanciado numa terceira máquina virtual e conectado também através de uma *gNB* e interface de tunelamento.

4.2 Coleta das Métricas

A coleta das métricas do Núcleo 5G (*Open5GS*) foi realizada a partir das interfaces providas pela implementação de cada serviço. Foi realizada a coleta periódica dos dados a partir da ferramenta de *scraping Prometheus* a cada 1 segundo, os dados foram salvos em um banco de dados.

A coleta das métricas do experimento de comunicação dos dispositivos foi feita a partir da ferramenta de *benchmark ApexAI performance_test*, que provê as estatísticas de forma tabular, com cada entrada representando 1 segundo do processo de comunicação.

Os conjuntos de dados gerados pela coleta periódica do núcleo 5G e do *benchmark* foram posteriormente agregados para realizar o treinamento dos modelos.

4.3 Cenários de Problemas de Rede

Para caracterização dos possíveis cenários de problemas de rede, foi utilizada a ferramenta *traffic control*[\[19\]](#) que utiliza um emulador de rede (em inglês *Network Emulator, netem*) provendo uma regra de enfileiramento para controle do tráfego. A partir desta ferramenta, foram implementados quatro cenários de comunicação entre os dispositivos, sendo eles:

- Cenário típico de comunicação;

- Cenário no qual é introduzida latência;
- Cenário no qual é introduzida perda de pacotes;
- e um cenário no qual é limitada a banda de rede de forma intermitente.

Os cenários detalhados na [Tabela 4.1](#) foram escolhidos com base nas características imprescindíveis de uma rede 5G no âmbito de fábricas inteligentes, que são a alta velocidade, disponibilidade e baixa latência. Estes aspectos foram explorados nos cenários pela aplicação de restrições intermitentes simuladas na banda de rede, introdução de latência na comunicação e pela introdução da política que ocasiona perda de pacotes, respectivamente. Os valores específicos para as restrições aplicadas a cada cenário simulado foram escolhidos de acordo com as características das mensagens trocadas pelos dispositivos simulados emissores e receptores, detalhados na subseção anterior.

Tabela 4.1: Cenários de Problemas de Rede do Experimento

Cenário	Descrição	Implementação
Típico	O cenário comum de comunicação entre emissor e receptor, sem restrições adicionadas à comunicação.	Transferência de 1000 mensagens por segundo, de 1024 bytes cada.
Alta Latência	Introduzida variação de latência de 350 a 500ms nas interfaces de rede providas pelo gNodeB simulado para os dispositivos emissor e receptor.	Variação do cenário Típico, na qual foi introduzida latência de 350ms a 500ms nas interfaces de rede.
Perda de Pacotes	Introduzida perda de 25% de pacotes nas interfaces de rede providas pelo gNodeB simulado para os dispositivos emissor e receptor.	Variação do cenário Típico, na qual foi introduzida perda de pacotes de 25% nas interfaces de rede.

Continua na próxima página

Tabela 4.1: Cenários de Problemas de Rede do Experimento (Continuação)

Limitação de Banda	Aplicada restrição intermitente de 1Mbps na transferência de informações entre os dispositivos emissor e receptor.	Varição do cenário Típico, na qual foi introduzida limitação de banda de rede em 1 Mbps de forma intermitente, variando entre 10s e 1min.
--------------------	--	---

Cada um dos 4 cenários foram simulados durante um período de 5 horas contínuas. Durante as simulações foi realizada a coleta das métricas como descrito anteriormente, para obter conjuntos estatisticamente relevantes após a partição em sequências. Cada conjunto de dados de cada cenário contém 18000 entradas representando o estado da rede e do *middleware* de comunicação dos dispositivos naquele segundo.

Os dados referentes a cada experimento foram particionados em sequências de 2 minutos (120 entradas), resultando em 150 amostras de trechos de execução contínuos para cada cenário. Esta partição foi determinada a partir da característica pseudorrandômica das disciplinas de rede introduzidas em cada cenário.

Para compor um conjunto de dados balanceado (mesmo número de amostras para cada cenário, como definido no requisito **R4**) a ser utilizado no treinamento e validação dos modelos na tarefa de classificação, esta proporção balanceada foi mantida durante a seleção dos conjuntos de treino, teste e validação utilizando estratificação com base no cenário (atendendo ao requisito **R4**).

4.4 Métricas Coletadas

Nesta seção são apresentadas as métricas coletadas para os conjuntos de serviços que compõem o núcleo 5G e para os serviços relacionados ao ecossistema *IoT* simulado a partir do *benchmark* de comunicação destes dispositivos.

4.4.1 Métricas do Núcleo 5G

Dentre as métricas disponíveis para coleta nos serviços do Núcleo 5G simulado a partir do *Open5GS*, foram selecionadas aquelas relacionadas ao tráfego na rede, uso de CPU e gestão

de sessão de dispositivos na interface *N4*. Estas escolhas são justificadas pelo fato de que os serviços que compõem o núcleo 5G, em especial o *UPF*, têm sua eficiência atrelada à disponibilidade desses recursos de CPU. As métricas estão detalhadas na [Tabela 4.2](#).

Tabela 4.2: Métricas Coletadas do Núcleo 5G e Serviços

Identificador	Descrição
inDataPktN3	Número de pacotes GTP recebidos na interface N3.
outDataPktN3	Número de pacotes GTP partindo da interface N3.
inDataVolumeN3	Volume de dados (bytes) dos pacotes GTP recebidos na interface N3.
outDataVolumeN3	Volume de dados (bytes) dos pacotes GTP partindo da interface N3.
sessionReportN4	Número de sessões requisitadas na interface N4.
cpuUsagePercent	Uso da CPU realizado pelos processos do Núcleo 5G, em porcentagem.

4.4.2 Métricas da Comunicação dos Dispositivos

Revisando as métricas providas pela ferramenta *performance_test*, foram selecionadas aquelas relacionadas à troca de mensagem entre os dispositivos e aquelas relacionadas ao uso de CPU e memória. Escolha justificada pelo fato de que os processos de comunicação têm sua eficiência atrelada à disponibilidade e velocidade de CPU e memória. Além disso, foram escolhidas aquelas com semântica alinhada com os cenários de problemas de rede. Estas métricas estão detalhadas na [Tabela 4.3](#).

Tabela 4.3: Métricas Coletadas da Comunicação dos Dispositivos IoT

Identificador	Descrição	Unidade
received	Número de mensagens recebidas.	Quantidade Observada

Continua na próxima página

Tabela 4.3: Métricas Coletadas da Comunicação dos Dispositivos IoT (Continuação)

sent	Número de mensagens enviadas.	Quantidade Observada
lost	Número de mensagens perdidas.	Quantidade Observada
relativeLoss	Relação entre número de mensagens perdidas e enviadas.	Porcentagem
dataReceived	Volume de dados recebidos	Bytes
latencyMin	Latência mínima.	Milissegundos
latencyMax	Latência máxima.	Milissegundos
latencyMean	Média de latência.	Milissegundos
latencyVariance	Variância da latência.	Milissegundos
publishLoopResMin	Tempo mínimo de execução da tarefa de publicação de mensagens.	Milissegundos
publishLoopResMax	Tempo máximo de execução da tarefa de publicação de mensagens.	Milissegundos
publishLoopResMean	Tempo médio de execução da tarefa de publicação de mensagens.	Milissegundos
publishLoopResVariance	Variância de tempo de execução da tarefa de publicação de mensagens.	Milissegundos
ruTime	Tempo total consumido pelos processos na CPU.	Milissegundos
ruMaxRss	Memória utilizada pelos processos.	Bytes
ruMinflt	Quantidade de requisições à memória utilizada pelo processo.	Quantidade observada
ruNvesw	Número de trocas voluntárias de contexto (CPU).	Quantidade observada

Continua na próxima página

Tabela 4.3: Métricas Coletadas da Comunicação dos Dispositivos IoT (Continuação)

ruNivcsw	Número de trocas involuntárias de contexto (CPU).	Quantidade observada
cpuUsagePercent	Uso da CPU realizado pelo processo, em porcentagem.	Porcentagem

4.5 Conjunto de Dados

Como detalhado na [Seção 4.3](#), o experimento foi realizado de forma a coletar os dados de métricas de rede e da comunicação de dispositivos *IoT* em de 4 cenários distintos de problemas de rede (incluindo o cenário típico em que não havia introdução de restrições na rede).

A partir da coleta destes dados advindos dos serviços do núcleo 5G e do *middleware* de comunicação dos dispositivos *IoT* simulados, foi necessário realizar o passo de agregar as métricas definidas na seção anterior ([Tabela 4.2](#) e [Tabela 4.3](#)). Esta agregação se deu a partir da variável em comum que é o horário anotado em cada amostra (em inglês, *timestamp*).

Agregados os dados do experimento, foi consolidado um único conjunto de dados, contendo informações dos serviços do núcleo 5G e também das métricas da comunicação entre os dispositivos *IoT*. Resgatando o objetivo principal deste trabalho, que é a observação quantitativa da comparação de dois modelos de rede neural na tarefa de classificação de cenários de problemas de rede. Observa-se o objetivo final de quantificar a melhoria provida por essa agregação destes conjuntos de métricas. Para isto foi realizada a partição do conjunto mencionado anteriormente para consolidação de um subconjunto contendo apenas as métricas da rede e serviços do núcleo 5G (como descrito na [Tabela 4.2](#)).

Desta forma, foram explorados neste trabalho dois conjuntos de dados: O primeiro, como demonstrado na amostra da [Figura 4.2](#), contém os dados da rede e serviços do núcleo 5G, agregados aos dados referentes à comunicação dos dispositivos *IoT* realizada nesta mesma rede, durante a execução dos cenários. E o segundo conjunto de dados, definido como um subconjunto do primeiro, em que foram selecionadas apenas as características referentes à rede e serviços do núcleo 5G, com o intuito de prover a comparação entre dois modelos que foram treinados com cada um destes dois conjuntos de dados.

4.5.1 Amostras Coletadas

Detalhando o que foi descrito na [Seção 4.3](#), as amostras foram coletadas com granularidade de 1 segundo, totalizando 18000 observações individuais do estado do sistema, referente à execução de 5 horas contínuas para cada um dos 4 cenários definidos anteriormente. Sendo assim, cada conjunto de dados final contém 72000 observações individuais.

Dada a natureza temporal destes dados, foi necessário definir um valor para realizar a amostragem das observações levando em conta a característica temporal, organizando estas observações em sequências de comprimento fixo. Como explicitado na [Seção 4.3](#), foi decidido particionar as observações em sequências contínuas de 120 segundos, resultando assim em 150 sequências equivalentes a 2 minutos de observação do estado da rede, com granularidade de 1 segundo. Logo, para cada um dos 4 cenários, em ambos os dois conjuntos de dados, foram definidas 150 sequências.

Ressalta-se que os conjuntos de dados finais estão balanceados entre as 4 classes, pois foram executados os cenários do experimento pela mesma quantidade de tempo. Esta característica foi verificada meticulosamente para garantir atender ao requisito **R4**, tratando do balanceamento dos dados contidos em cada conjunto.

A [Figura 4.2](#) demonstra o resumo de uma amostra do conjunto de dados completo, contendo 120 observações das métricas da rede e da comunicação dos dispositivos *IoT*.

timestamp	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	120
inDataPktN3	2600.0	2600.0	2600.0	2600.0	2600.0	2600.0	2600.0	2600.0	2600.0	2600.0	2600.0	6500.0	6500.0	6500.0	6500.0	109400.0
inDataVolumeN3	2915544.0	2915544.0	2915544.0	2915544.0	2915544.0	2915544.0	2915544.0	2915544.0	2915544.0	2915544.0	7407660.0	7407660.0	7407660.0	7407660.0	7407660.0	125601248.0
outDataPktN3	2500.0	2500.0	2500.0	2500.0	2500.0	2500.0	2500.0	2500.0	2500.0	2500.0	6500.0	6500.0	6500.0	6500.0	6500.0	109400.0
outDataVolumeN3	2852380.0	2852380.0	2852380.0	2852380.0	2852380.0	2852380.0	2852380.0	2852380.0	2852380.0	2852380.0	7460192.0	7460192.0	7460192.0	7460192.0	7460192.0	125600788.0
sessionReportN4	3.0	3.0	3.0	3.0	3.0	3.0	3.0	3.0	3.0	3.0	3.0	3.0	3.0	3.0	3.0	3.0
received	93	999	999	551	107	107	107	107	107	107	107	107	107	107	107	999
sent	1001	999	999	999	999	999	1000	999	999	999	999	999	999	1000	999	999
lost	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
dataReceived	97730	1039870	1039877	574007	112304	112303	111633	112303	112306	112305	112304	112301	112302	112303	112304	1039847
latencyMin	0.1379	0.1298	0.1361	0.1233	455.9	1349	2241	3133	4026	4919	5812	6704	7597	8489	9264	0.1331
latencyMax	16.89	29.34	46.17	447.6	1340	2233	3125	4018	4911	5803	6696	7588	8481	9275	9266	3065
latencyMean	1808.0	0.9428	0.8474	22.28	898.0	1791.0	2684.0	3576.0	4468.0	5363.0	6254.0	7147.0	8039.0	8925.0	9265.0	0.2124
latencyVariance	0.01589	0.01089	0.007421	6262	66.38	66.38	66.48	66.53	66.52	66.79	66.4	66.62	66.38	61.02	0.0001434	0.01653
publishLoopResMin	548.0	0.8721	44.16	0.8618	0.2944	0.2009	0.2727	15.21	0.2578	4812.0	5.76	15.15	0.5206	11.24	0.1158	12.09
publishLoopResMax	0.9448	0.9404	0.9392	0.9402	0.9782	0.9403	0.9848	0.9674	0.9542	0.9624	0.9744	0.9788	0.9672	0.9602	0.9408	0.9784
publishLoopResMean	0.9253	0.9351	0.9381	0.9315	0.9249	0.9251	0.9225	0.5583	925.0	0.9066	0.8948	0.6882	0.9255	826.0	0.9218	0.7564
ruTime	0.02837	0.05994	0.1105	152.0	0.1824	0.2141	0.2471	0.3049	0.3338	0.3605	0.3884	0.4296	0.4557	0.4811	509.0	4913.0
ruMaxRss	26564	26564	26564	26564	26564	26564	26564	26564	26564	26564	26564	26564	26564	26564	26564	26564
ruMinflt	3912	3912	3924	3924	3924	3924	3924	3924	3924	3924	3924	3924	3924	3924	3924	3924
ruNvcsw	1468	6377	10969	14686	17046	19423	21795	24065	26365	28565	30771	32934	35157	37348	39569	437464
ruNvcsw	7	14	23	23	23	25	26	30	30	31	31	32	32	32	32	327
cpuUsagePercent	0	2.75	3.5	2.25	2	1.75	1.98	3.25	2	1.75	1.5	2.75	1.75	2	1.5	3

Figura 4.2: Amostra de uma única sequência dos dados de rede e da comunicação dos dispositivos IoT

4.5.2 Divisão para Treinamento, Teste e Validação

As sequências descritas nas subseções anteriores foram divididas na proporção 70:20:10 para treino, teste e validação, respectivamente. Esta proporção é comumente utilizada na prática de aprendizado de máquina, pois proporciona uma divisão balanceada dos dados, provendo uma quantidade significativa de exemplos para treinamento (70%) com poucos compromettimentos à capacidade de avaliação do modelo com dados de teste (20%) ou validação (10%). Além disso, essa distribuição é amplamente reconhecida como um compromisso eficaz entre utilizar a maior quantidade possível de dados para treinamento e manter conjuntos de teste e validação suficientemente grandes para fornecer estimativas confiáveis do desempenho do modelos.

Portanto, para cada classe há 104 sequências para treino, 31 para teste e 15 para validação, em cada um dos dois conjuntos de dados. Foi utilizada a ferramenta *Scikit-learn* [27] para realizar a partição destas amostras.

4.6 Problema de Classificação

Dado o experimento no qual os dispositivos comunicam-se a partir da rede 5G em cenários predefinidos, nos quais são induzidas situações de problemas da rede específicos, nota-se que a identificação destes cenários pode ser tratada como um problema de classificação. Considerando que as informações disponíveis para caracterização destes cenários são relacionadas ao estado da rede e da eficiência da comunicação dos dispositivos a cada instante de tempo de forma sequencial, destaca-se a sensibilidade destes dados à passagem de tempo. Logo, trata-se de um problema de classificação de dados que se organizam em séries temporais, que é considerado complexo pela literatura [33].

As subseções a seguir detalham a escolha dos modelos utilizados para resolução deste problema de classificação e a forma como os resultados obtidos foram avaliados.

4.6.1 Modelo de Rede Neural

A utilização de modelos de rede neural na resolução deste problema de classificação é justificada pela capacidade destes modelos de identificar padrões complexos e não-lineares nos da-

dos. Redes neurais, especialmente as *Recurrent Neural Networks (RNNs)* e *Long Short-Term Memory networks (LSTMs)*, são particularmente adequadas para lidar com dados sequenciais e de caráter temporal, como é o caso das séries temporais.

Como comentado no [Capítulo 2](#) e também revisado nos trabalhos relacionados ([Capítulo 3](#)), a adoção de um modelo LSTM para solução do problema de classificação multiclasse em séries temporais, proposto neste trabalho, foi essencial para prover bons resultados nesta tarefa. A característica inerente a este modelo de combinar memórias de curto e longo prazo se mostrou eficaz para generalizar padrões que só estão em evidência ao observar as sequências das amostras dos dados de rede, por exemplo.

4.6.2 Metodologia para avaliação dos modelos

Para avaliar modelos que realizam a tarefa de classificação, o método amplamente utilizado consiste em observar as métricas de acurácia ([Equação 4.1](#)), precisão ([Equação 4.2](#)), recall (sensibilidade, [Equação 4.3](#)) e F1-Score ([Equação 4.4](#)). A precisão é uma métrica importante, pois denota a proporção das previsões positivas corretas em relação ao total de previsões positivas (TP), caso em que o cenário de problema de rede é classificado corretamente. Nesse contexto previsões de falsos positivos (FP) e falsos negativos (FN) são aquelas em que a amostra de série temporal é classificada na categoria incorreta, observados a depender da classe em questão, visto que se trata de um problema de classificação em múltiplas classes (cenários).

Neste contexto, outras métricas importantes de avaliar são *recall* (sensibilidade), que denota a proporção de verdadeiros positivos em relação ao total de casos positivos reais (verdadeiros positivos mais falsos negativos, [Equação 4.3](#)) e o *F1-Score* que representa uma média harmônica de precisão e recall, especialmente útil para avaliar conjuntos de dados que não estão precisamente balanceados, como pode ser o eventual caso de observar problemas na rede 5G implementada num ambiente industrial.

Além destas métricas, observar a curva *ROC* (em inglês, *Receiver Operating Characteristic*) em um problema de classificação multiclasse pode trazer visões importantes sobre o desempenho dos modelos. Visto que a esta oferece uma visualização abrangente da capacidade do modelo de distinguir as diferentes classes, demonstrando uma visão na qual este problema de classificação é tratado como um problema de classificação binária, em que o

modelo é avaliado na sua capacidade de distinguir a classe alvo dentre as outras, uma a uma, a partir da taxa de falso positivos (FP) e positivos verdadeiros (TP).

Por fim, a acurácia é diretamente ligada à taxa de acerto total do modelo, levando em conta todas as classes neste problema de classificação de cenários de problemas de rede. Ressalta-se, que por se tratar de um problema de classificação multiclasse, é importante realizar a avaliação das métricas mencionadas anteriormente para cada classe de forma individual, pois a média destas métricas pode encobrir o desempenho de classificação para o modelo. Além disso, é importante também observar o valor obtido para a métrica de *recall*, pois esta irá indicar a taxa acerto na classificação dos cenários de problema de rede para cada classe pelo modelo.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4.1)$$

$$Precision = \frac{TP}{TP + FP} \quad (4.2)$$

$$Recall = \frac{TP}{TP + FN} \quad (4.3)$$

$$F1-Score = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * TP}{2 * TP + FP + FN} \quad (4.4)$$

4.7 Considerações Finais

Em resumo, a coleta de métricas do núcleo 5G e do ecossistema de dispositivos *IIoT*, juntamente com a análise de cenários de problemas de rede, proporciona uma visão dos impactos da combinação destas informações para análises de dados nas fábricas inteligentes. A aplicação de modelos de rede neural na análise de dados, como modelos *LSTM*, destaca-se como uma evolução importante na classificação e predição de problemas de rede, crucial para a eficiência e previsibilidade no cenário de fábricas inteligentes.

A abordagem metodológica utilizada neste capítulo não só orienta a execução do presente trabalho, mas também traça um caminho para futuras investigações neste campo, de acordo com as hipóteses e requisitos alcançados.

Capítulo 5

Implementação e Resultados

Neste capítulo são apresentados os detalhes específicos de implementação de cada componente do experimento, levando em conta os requisitos elencados no capítulo anterior. Na [Seção 5.1](#) são detalhados os componentes do experimento e sua implementação, na [Seção 5.2](#) são detalhadas as técnicas utilizadas para o treinamento dos modelos, na [Seção 5.3](#) são explorados os resultados obtidos neste trabalho, bem como a descrição dos desafios e limitações observadas durante a construção deste trabalho.

5.1 Arquitetura dos Componentes do Experimento

Para a arquitetura da solução proposta, foram levados em conta os requisitos elencados no [Capítulo 4](#), que são:

- **Requisito 1 (R1):** Os componentes do sistema devem ser projetos de código aberto;
- **Requisito 2 (R2):** A comunicação entre os componentes do sistema deve ocorrer estritamente a partir da rede 5G;
- **Requisito 3 (R3):** Os dados devem ser coletados a partir do núcleo 5G e ecossistema *IIoT* de forma que não interfira na comunicação destes sistemas;
- **Requisito 4 (R4):** Os modelos devem ser treinados em parâmetros equivalentes, com conjuntos de dados balanceados e estratificados.

Tabela 5.1: Caracterização das Máquinas Virtuais do Experimento

Máquina Virtual	Sistema Operacional	vCPU	RAM	Função
Core 5G Simulado	Ubuntu Server 22.04.3 LTS	8	8 GB	Implantar o Core 5G Simulado (Open5GS) incluindo os componentes principais como AMF, UPF e SMF
gNodeB Simulado #1 e #3		4	8 GB	Implantar gNodeB simulado (UERanSim) e os UE Simulados (Nós ROS2 para simulação dos dispositivos emissor e receptor)
gNodeB Simulado #2		4	8 GB	Implantar gNodeB simulado (UERanSim) e o UE Simulado (Servidor de descoberta FastDDS)
Sistema agregador de métricas		4	4 GB	Implantar componentes de monitoramento e consolidação de métricas (Prometheus e Grafana)

Com isto, foram instanciadas 5 máquinas virtuais para serem distribuídas entre os serviços destacados na [Tabela 5.1](#). Os recursos disponíveis para cada máquina virtual foram distribuídos de acordo com a disponibilidade do ambiente de testes e também observando os requisitos de *hardware* de cada serviço. Desta forma foram alocados mais recursos para a máquina virtual na qual foi implantado o núcleo 5G para evitar sobrecarga durante a execução dos cenários do experimento.

Todas as máquinas virtuais foram instanciadas com o sistema operacional *Ubuntu Server 22.04.3 LTS*, através do software *Multipass* [9] com o uso do *Hypervisor LXD* para o suporte a pontes para conexão com as interfaces de rede (*bridge network*), característica que foi essencial para efetivação da comunicação entre os componentes do experimento (atendendo ao requisito **R2**).

A comunicação entre as aplicações que compõem o experimento tem como elemento central o núcleo 5G, implantado a partir do *Open5GS*, componente que proveu uma partição

de rede (em inglês, *network slice*) utilizando o serviço de *UPF*. A partir deste serviço e partição, foram conectados os *gNodeB* simulados a partir da ferramenta *UERANSIM* que disponibiliza interfaces de rede para conexão das aplicações da fábrica inteligente (atendendo ao requisito **R2**). Foram utilizadas 3 instâncias de máquina virtual para implantação dos agentes emissor, receptor das mensagens e para o serviço de descoberta (*discovery server*) inerente ao protocolo de comunicação (*FastDDS*).

O serviço de coleta de métricas utiliza o *Prometheus* para comunicar-se diretamente com os serviços do núcleo 5G através de interfaces já implementadas pelo *Open5GS*. A coleta das métricas por este serviço se deu a partir dos dados tabulares salvos em um arquivo após a consolidação do experimento de cada cenário. A representação visual desta arquitetura pode ser observada no diagrama da [Figura 5.1](#) (destacando o objetivo do requisito **R3**).

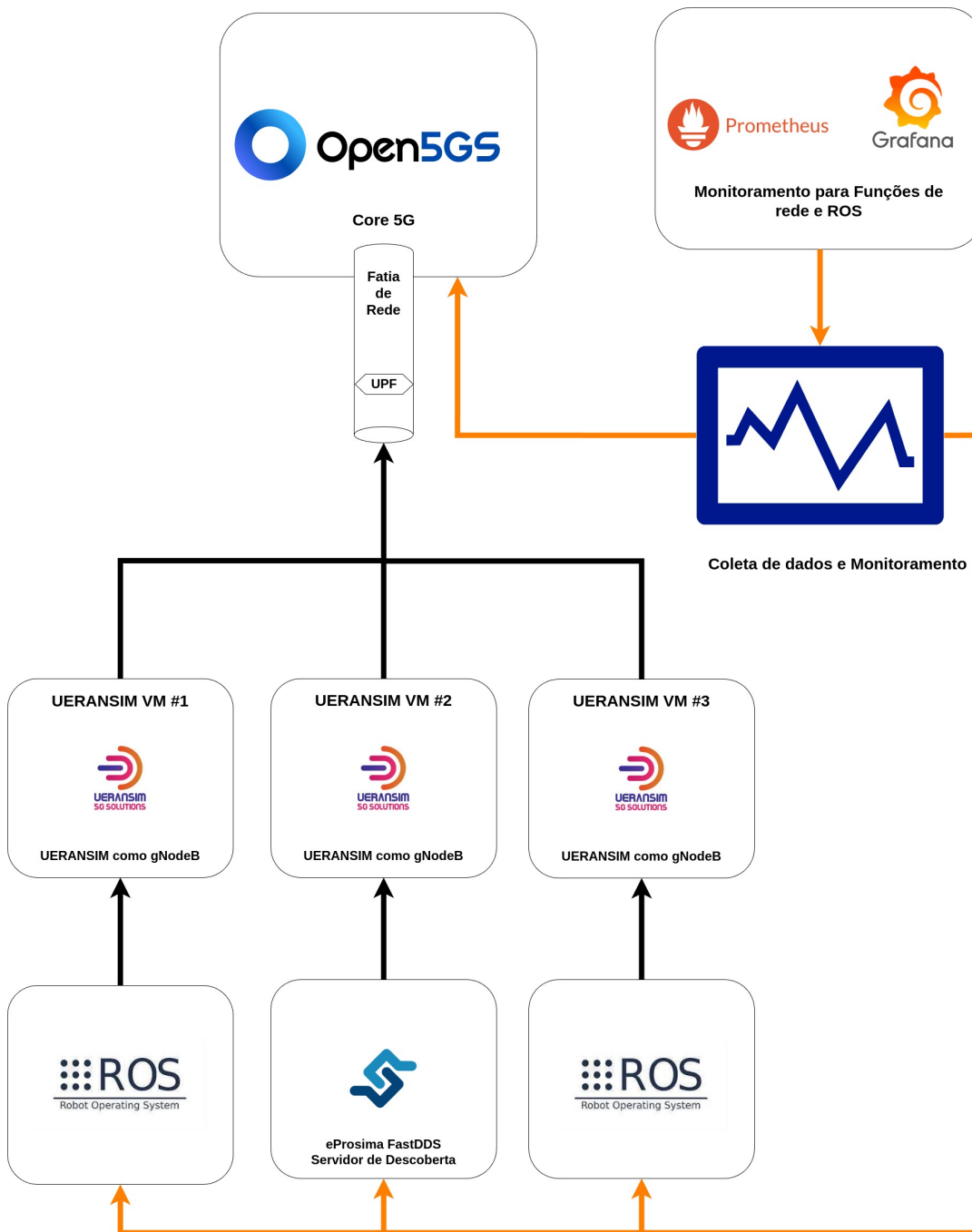


Figura 5.1: Arquitetura simplificada dos serviços e aplicações utilizados no experimento

As aplicações utilizadas no experimento foram isoladas utilizando restrições de rota de rede para que a comunicação fosse efetivada estritamente a partir das interfaces de rede providas pelo *UERANSIM*, fazendo o papel de *gNodeB* (**R2**). O serviço de descoberta utilizado pelo protocolo de comunicação permitiu a comunicação entre os processos distribuídos de forma direta, evitando a propagação das mensagens em *broadcast*. Todas as mensagens tro-

cadadas entre os processos simulando *UE* e o serviço de descoberta foram encapsuladas em pacotes *GTP*, característica inerente ao protocolo *GPRS* utilizado pelo núcleo 5G. A representação visual desta arquitetura está descrita no diagrama da [Figura 5.2](#).

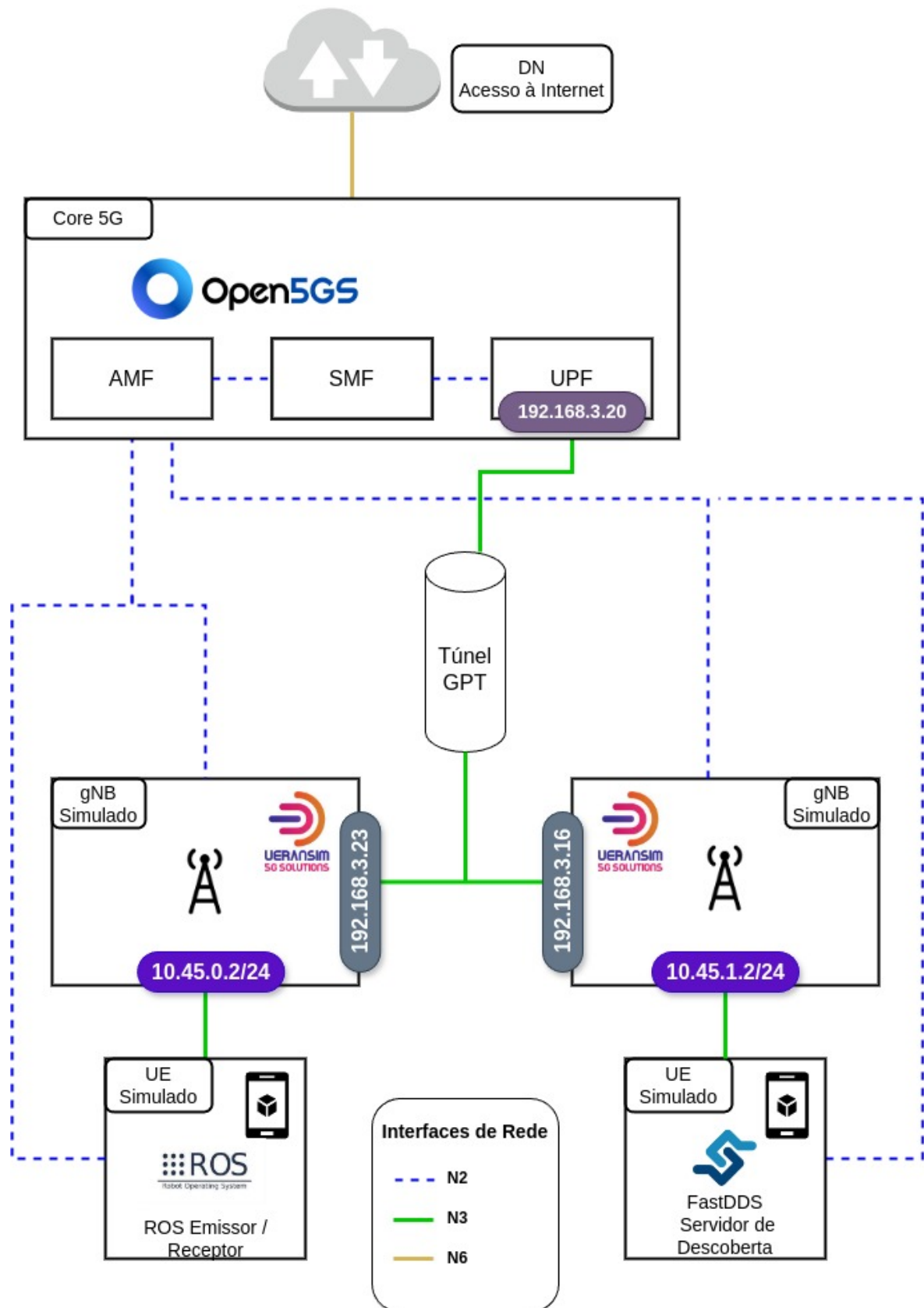


Figura 5.2: Arquitetura das interfaces de rede utilizadas no experimento

5.2 Técnicas utilizadas para treinamento dos modelos

Para obter modelos com o melhor desempenho possível em cada conjunto de dados (para a comparação da introdução das métricas da comunicação dos dispositivos), foi utilizada a mesma estratégia em ambos os modelos a fim de promover uma comparação justa de desempenho (**R4**).

Recapitulando os detalhes fornecidos na [Seção 4.5](#), foi realizada a partição dos dados em amostras de 2 minutos, totalizando 150 amostras obtidas a partir do experimento, para cada um dos 4 cenários de problema de rede. As amostras foram divididas em seções de treino, teste e validação na proporção *70:20:10*, respectivamente.

Para otimizar o treinamento dos modelos *LSTM* voltados para classificação multiclasse de séries temporais, foram adotadas técnicas como *early stopping* e *hyperband*, cada uma com um papel específico no aprimoramento da aprendizagem e generalização dos modelos. Inicialmente, utilizou-se a técnica de *Dropout*, que ajuda a prevenir o sobreajuste (*overfitting*) ao descartar aleatoriamente unidades (neurônios) durante o treinamento, contribuindo para uma aprendizagem mais robusta e menos dependente de padrões específicos nos dados de treinamento.

A *regularização L2*, por outro lado, adiciona um termo de penalidade ao custo de treinamento, baseado na magnitude dos pesos dos neurônios, penalizando pesos grandes e promovendo um modelo mais simples e menos propenso ao sobreajuste. Esse termo de regularização tende a dispersar o erro sobre todos os neurônios, favorecendo pesos menores e mais distribuídos.

A normalização em lote (*Batch Normalization*) foi empregada para estabilizar e acelerar o treinamento. Ela normaliza as entradas de cada camada para ter uma média de zero e um desvio padrão de um, reduzindo o deslocamento interno da covariância (*Internal Covariate Shift*) e permitindo o uso de taxas de aprendizagem mais elevadas, além de ser menos sensível à inicialização dos pesos.

Além disso, foi realizada normalização dos dados antes do treinamento. Esta normalização nos dados de entrada, de modo que tenham uma escala comum, é crucial para evitar que características com magnitudes maiores dominem o processo de aprendizagem e garanta que o gradiente descendente funcione de forma eficiente.

Por fim, o *Early Stopping* foi utilizado como uma forma de interromper o treinamento assim que o desempenho do modelo no conjunto de validação começasse a piorar, evitando assim o sobreajuste e garantindo que os modelos mantenham a capacidade de generalização.

Essas técnicas, implementadas com o auxílio dos *frameworks TensorFlow*[2] e *Keras*[11], contribuíram significativamente para a eficácia e eficiência do treinamento dos modelos *LSTM*. Os hiperparâmetros para estas técnicas, bem como outros aspectos da arquitetura do modelo, foram refinados utilizando o *keras-tuner*[23], uma ferramenta que executa testes extensivos com o objetivo de alcançar a melhor acurácia através da técnica de *hyperband*. Com os hiperparâmetros otimizados identificados, os modelos foram avaliados utilizando o conjunto de teste, permitindo uma avaliação precisa do desempenho de classificação. A arquitetura simplificada para os modelos *LSTM* está descrita na [Figura 5.3](#).

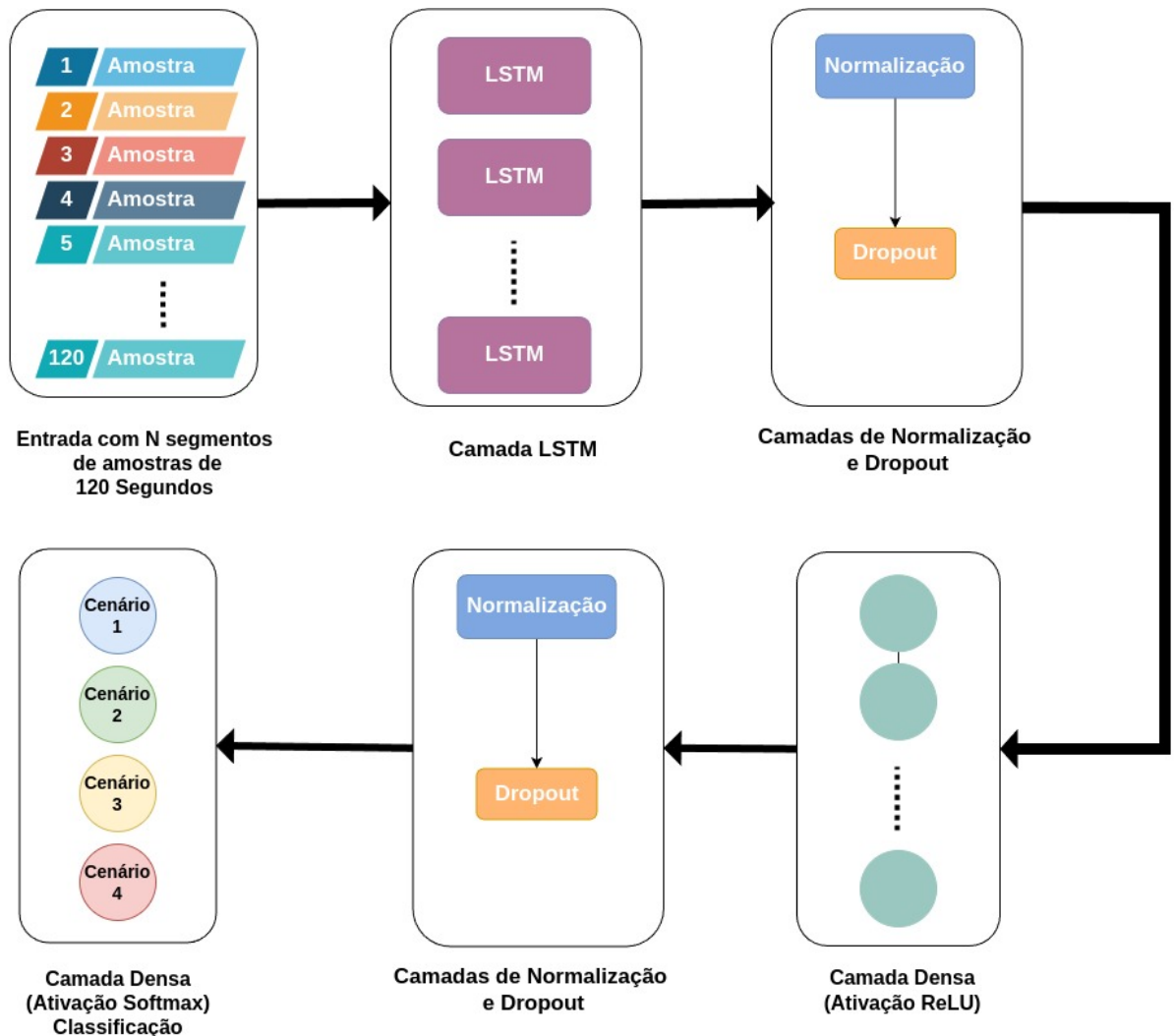


Figura 5.3: Arquitetura simplificada definida para os modelos

Uma prática comum para o treinamento de modelos de rede neural é utilizar técnicas de validação cruzada, (em inglês, *K-Fold Cross Validation*). Porém, ao se tratar de dados sequenciais e que possuem dependência temporal nesta sequência, este método não é indicado para treinamento de modelos visando resolver um problema de classificação de dados de série temporal. Bergmeir e Benítez em [5], evidenciam que a utilização desta técnica descaracteriza o aspecto sequencial dos dados e pode trazer perdas de desempenho ainda maiores no âmbito da classificação multiclasse. Portanto, neste trabalho foi utilizado o método de *holdout*, no qual os conjuntos de dados foram divididos entre treino, teste e validação, abordagem adequada para tratar das amostras de dados de séries temporais.

Por fim, após as etapas de testes utilizando *hyperband*, foram definidos os hiperparâmetros mais promissores para cada modelo, observando a métrica de acurácia. Os hiperparâmetros definidos para o treinamento do modelo que recebeu o conjunto de dados contendo apenas as informações do núcleo 5G estão resumidos na Tabela 5.2.

Camada	Formato de saída	Parâmetro(s)	Ativação
LSTM	480	Unidades: 240	LSTM (sigmoid e tanh)
		Reg. L2: 0.013	
		Reg. L2 recorrente: 0.005	
Normalização # 1	480	-	-
Dropout # 1	480	Dropout: 0.36	-
Dense # 1	130	Unidades: 130	ReLU
Normalização # 2	130	-	-
Dropout # 2	130	Dropout: 0.36	-
Dense #2	4	Unidades: 4	Softmax (Classificação)
Taxa de Aprendizado	0.001		

Tabela 5.2: Hiperparâmetros Modelo Core 5G

Já os hiperparâmetros definidos para o treinamento do modelo que recebeu o conjunto de dados contendo as métricas relativas à comunicação dos dispositivos *IoT* em conjunto com as informações do núcleo 5G estão resumidos na [Tabela 5.3](#).

Camada	Formato de saída	Parâmetro(s)	Ativação
LSTM	848	Unidades: 424	LSTM (sigmoid e tanh)
		Reg. L2: 0.03	
		Reg. L2 recorrente: 0.02	
Normalização # 1	848	-	-
Dropout # 1	848	Dropout: 0.1	-
Dense # 1	243	Unidades: 243	ReLU
Normalização # 2	243	-	-
Dropout # 2	243	Dropout: 0.13	-
Dense #2	4	Unidades: 4	Softmax (Classificação)
Taxa de Aprendizado	0.0001		

Tabela 5.3: Hiperparâmetros Modelo Core 5G + IoT

5.3 Resultados

Nesta seção são apresentados os resultados da avaliação dos modelos, bem como a discussão do impacto destes em relação ao objetivo proposto. Em seguida são apresentados os desafios e limitações observados durante a construção deste trabalho.

5.3.1 Avaliação dos modelos

A partir do treinamento e avaliação dos modelos com os diferentes conjuntos de dados, sendo um deles incluindo os dados da comunicação dos dispositivos *IoT* simulados, pode-se observar melhoria de 12% na acurácia média e 15% no *recall* médio. Ressalta-se que,

para este problema de classificação multiclasse, é importante observar os valores de acurácia e *recall* obtidos para cada classe individualmente. Além disso, o *recall* é responsável por indicar se cada ocorrência do cenário de problema foi identificado corretamente (caso em que o valor se aproxima de 100%). Na questão de acurácia individual, houve incremento de 19% para o cenário de perda de pacotes e 43% para o cenário de congestionamento de banda. No quesito de *recall*, foi obtida melhoria no cenário de perda de pacotes de 72%. As métricas finais para o modelo treinado com o subconjunto contendo apenas os dados do núcleo 5G estão detalhadas na [Tabela 5.4](#), e para o modelo treinado com o conjunto completo dos dados, estão detalhadas na [Tabela 5.5](#).

Tabela 5.4: Resultados obtidos para o modelo treinado apenas com dados do Núcleo 5G

Cenário	Precisão (%)	Recall (%) (Sensibilidade)	F1-Score (%)	Amostras
Típico	100	100	100	26
Alta Latência	100	100	100	26
Perda de Pacotes	64	28	39	26
Congestionamento de Banda	57	86	59	26
Métricas Finais (%)		Amostras		
Acurácia Precisão	81	104		
Média Macro Precisão	80			
Média Macro Recall	78			
Média Macro F1- Score	77			
Média Ponderada Precisão	82			
Média Ponderada Recall	81			
Média Ponderada F1-Score	80			

Tabela 5.5: Resultados obtidos para o modelo com dados do núcleo e dos dispositivos IoT

Cenário	Precisão (%)	Recall (%) (Sensibilidade)	F1-Score (%)	Amostras
Típico	100	100	100	26
Alta Latência	100	100	100	26
Perda de Pacotes	83	100	91	26
Congestionamento de Banda	100	82	90	26
Métricas Finais (%)			Amostras	
Acurácia Precisão	93	104		
Média Macro Precisão	93			
Média Macro Recall	93			
Média Macro F1- Score	92			
Média Ponderada Precisão	94			
Média Ponderada Recall	93			
Média Ponderada F1-Score	93			

A análise das curvas de taxa de perda durante o treinamento, [Figura 5.4](#) e [Figura 5.5](#), não indicam fortes evidências de *underfitting* (má generalização do modelo durante o treinamento) ou *overfitting* (boa taxa de acertos durante o treinamento, porém má taxa de acertos

durante a validação). A evolução da acurácia dos modelos ao longo do treinamento pode ser observada nas [Figura 5.6](#) e [Figura 5.7](#). Uma visão mais precisa destas figuras permite notar uma maior regularização no decaimento da perda durante o treinamento e validação para o modelo que recebeu o conjunto de dados completo.

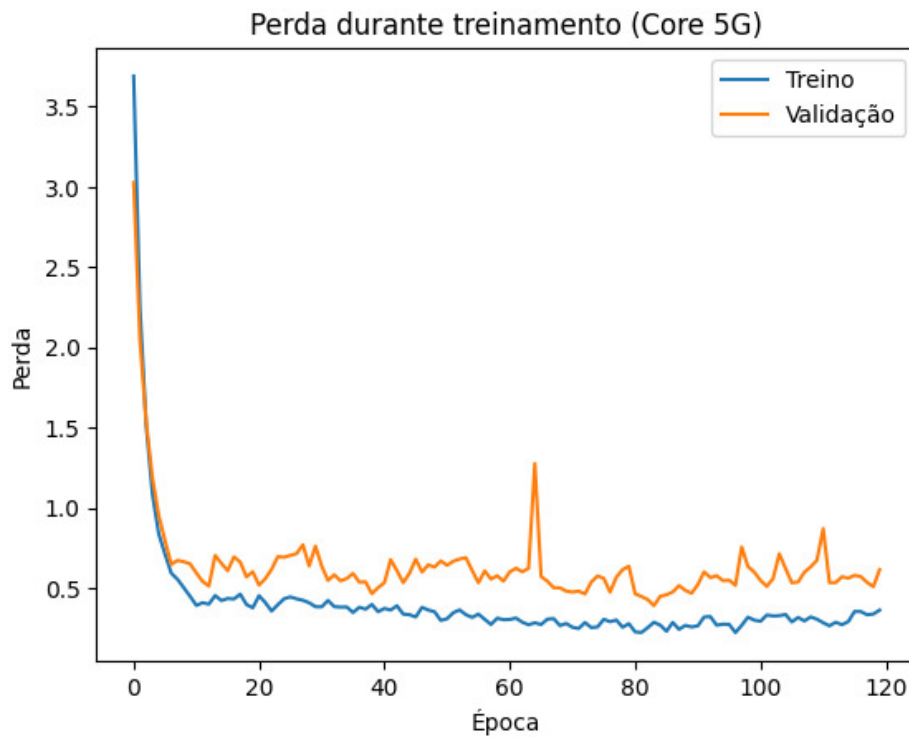


Figura 5.4: Decaimento da perda dos modelos durante treinamento (*training loss*) do modelo com subconjunto de dados do núcleo 5G

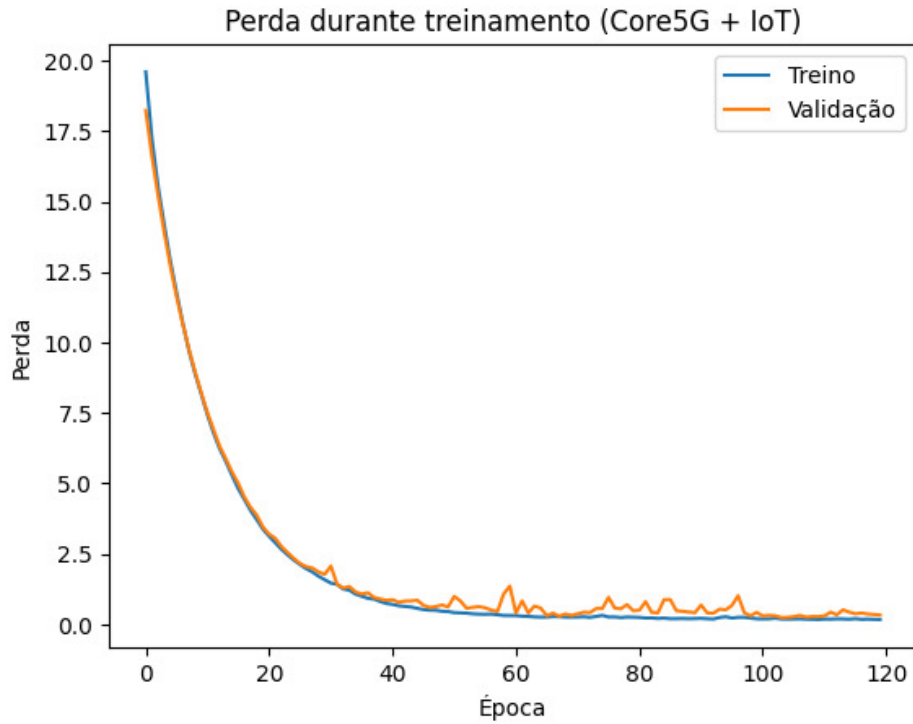


Figura 5.5: Decaimento da perda dos modelos durante treinamento (*training loss*) do modelo com dados do núcleo 5G e dispositivos IoT

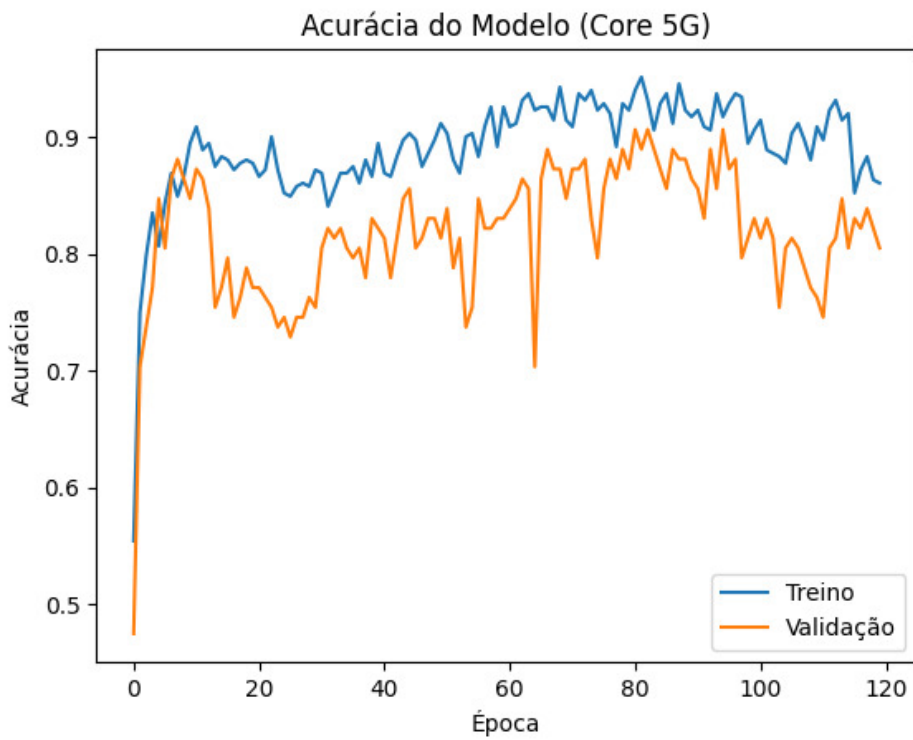


Figura 5.6: Curva da acurácia do modelo com dados do núcleo 5G

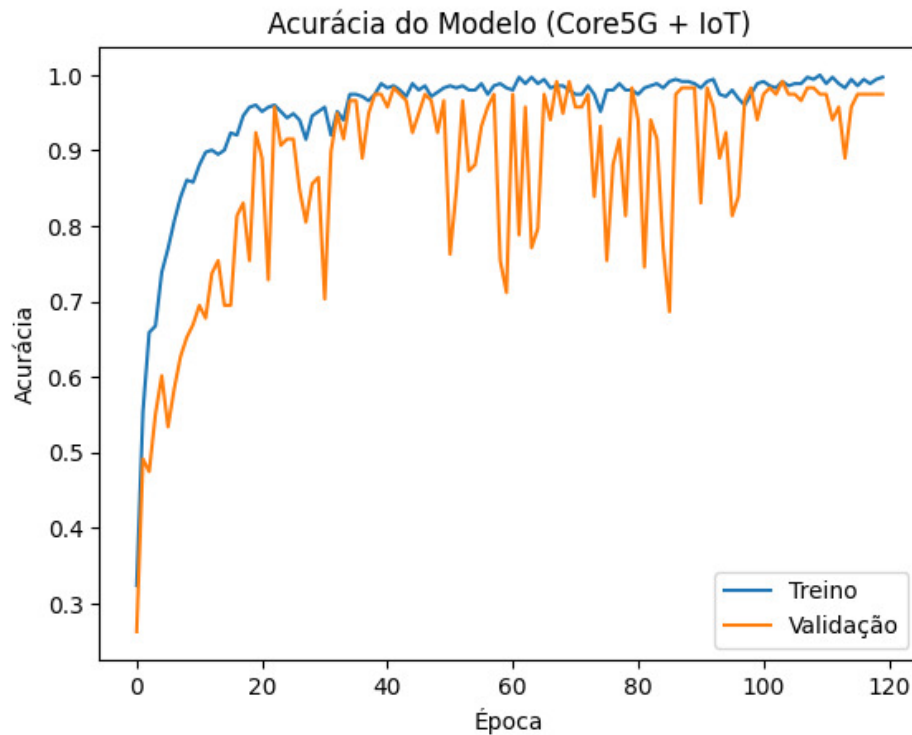


Figura 5.7: Curva da acurácia do modelo com dados do ecossistema completo

A partir da observação das curvas ROC calculadas para ambos os modelos, observa-se que há ganhos de desempenho no modelo que foi treinado com o conjunto completo dos dados (Figura 5.9). Nota-se também que o desempenho do modelo treinado apenas com dados da rede 5G é significativo, mas há falhas em distinguir os cenários em que há perda de pacotes e limitação de banda (Figura 5.8), em comparação com o outro modelo.

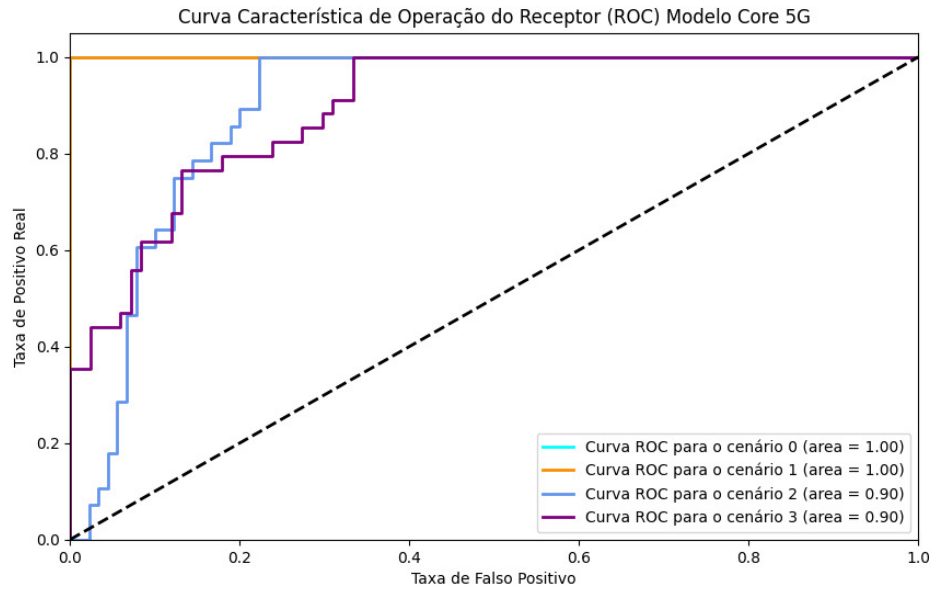


Figura 5.8: Curva ROC para o modelo com dados do núcleo 5G¹

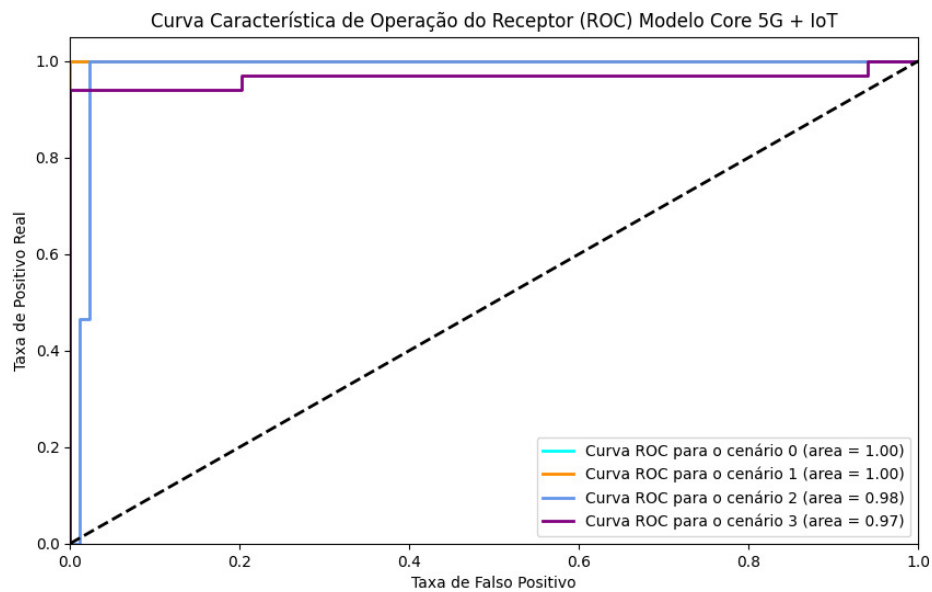


Figura 5.9: Curva ROC para o modelo com dados do ecossistema completo

¹Para as Figuras [Figura 5.8](#) e [Figura 5.9](#), os cenários de 0 a 3 representam os cenários: Típico, Alta Latência, Perda de Pacotes e Limitação de Banda, respectivamente.

5.3.2 Discussão

A partir dos ganhos observados na acurácia e *recall* do modelo que foi treinado com os dados agregados da rede 5G e dos dispositivos *IoT*, obtidos durante o experimento, observa-se a expressividade e impacto que estes dados puderam gerar numa tarefa complexa que é a de classificação multiclasse em séries temporais. Estes aspectos também foram evidenciados na observação das curvas *ROC* e nas figuras que demonstram a importância dos atributos para cada modelo, discutidas a seguir. Estas melhorias obtidas sinalizam que a integração de dados gerados por dispositivos operando em uma rede 5G pode ser benéfica para otimizar os processos de gestão e decisão no contexto industrial, visto que agregaram valor na tarefa de classificação de problemas de rede proposta neste trabalho.

Logo, tal integração propicia uma compreensão mais aprofundada de potenciais problemas relacionados à conectividade, ao congestionamento de rede e ao consumo de banda, questões estas que poderiam não ser identificadas mediante a análise exclusiva das métricas convencionais de rede. Portanto, a análise metódica desses dados específicos pode ser instrumental na implementação de estratégias preventivas para uma gestão mais eficiente dos recursos da rede, para a priorização de fluxo de tráfego e para a redução do tempo de inatividade, impulsionando assim a eficácia operacional.

Com o do treinamento dos modelos na tarefa de classificação dos cenários de problemas de rede, foi possível observar a importância atribuída pelo modelo a cada atributo disponível nos conjuntos de dados. A medida de importância de cada atributo está diretamente ligada ao parâmetro de peso que a camada de entrada associa a estes atributos, denotando seu valor para a tarefa de classificação.

Nas figuras [Figura 5.10](#) e [Figura 5.11](#) que demonstram os atributos considerados mais importantes para o modelo treinado com os dados da rede e com os dados do ecossistema completo, respectivamente, nota-se que os atributos considerados mais relevantes se repetem. Porém, pode-se observar que o modelo que atingiu maior desempenho (treinado com o conjunto de dados completo), associa maior importância aos atributos de pacotes enviados durante o experimento e à variável tempo. Para esta última foi atribuído um menor valor de importância no modelo com desempenho inferior.

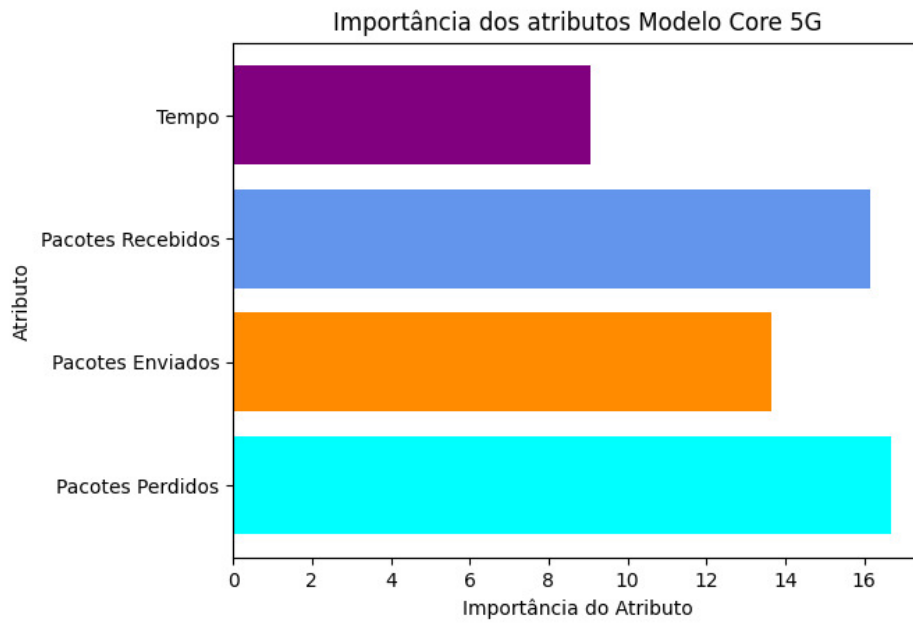


Figura 5.10: Importância dos atributos para o modelo com dados do núcleo 5G

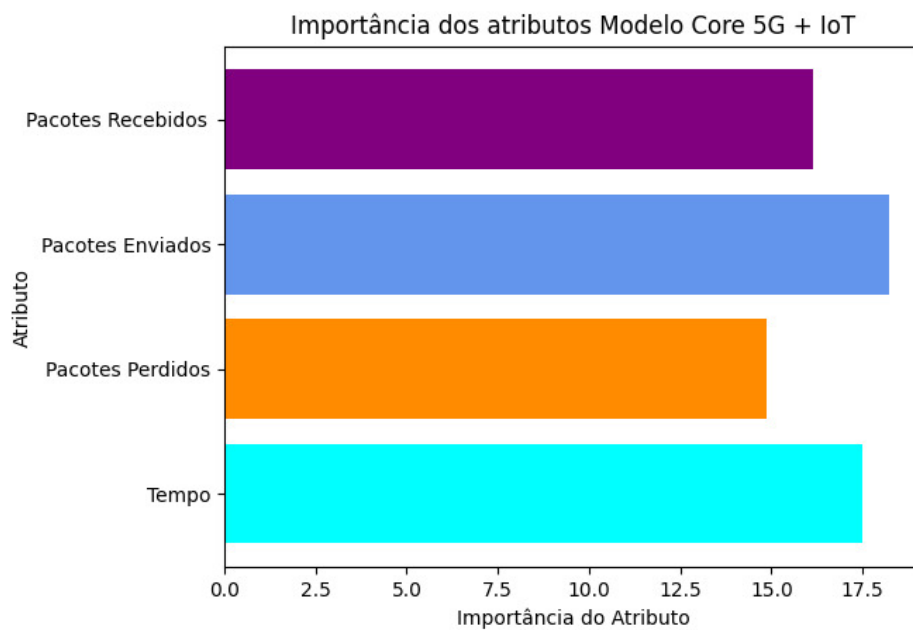


Figura 5.11: Importância dos atributos para o modelo com dados do ecossistema completo

5.3.3 Desafios e limitações

Como descrito em capítulos anteriores, os projetos de código aberto que realizam a implementação de núcleo 5G ainda não estão maduros no âmbito da coleta e análise de dados.

O fato de que o serviço *NWDAF* (em inglês, *Network Data Analytics Function*) não está presente nativamente em nenhum dos projetos analisados, até o presente momento, foi de grande impacto para guiar a definição do experimento realizado e proporcionou limitação quanto aos cenários explorados e métricas coletadas.

Além disso, a complexidade dos sistemas implementados e configuração de roteamento da rede que os comunica limitou testes em cenários com maior número de dispositivos e *UPF*. A ausência de implementações definitivas para as regras de asseguramento de *QoS* (*Quality of Service*) por parte dos projetos *Open5GS* e *UERANSIM*, fazendo o papel de núcleo 5G e *gNodeB* respectivamente, também foi aspecto limitante quanto aos cenários presentes no experimento.

Para a implementação dos modelos de rede neural utilizados neste trabalho, foi necessário considerar uma arquitetura simplificada de modelos que são conhecidos na literatura por ter grande desempenho na tarefa de classificação proposta. Esta simplificação está ligada aos limites de recursos computacionais e de tempo necessários para a experimentação dos hiperparâmetros (*hyperband*), treinamento e avaliação destes modelos. Portanto, modelos como observados no trabalho de Zhang et al. [49] estavam distantes dos recursos disponíveis para execução deste experimento. Ademais, a implantação e execução dos cenários de problemas de rede propostos no experimento realizado também foram limitados pela disponibilidade destes recursos. Portanto, além da impossibilidade de explorar os cenários que envolvem *QoS* nesse âmbito (pela lacuna na implementação), foi necessário limitar a combinação dos cenários de problemas de rede e também o tempo de execução de cada cenário no experimento.

5.4 Considerações Finais

Em resumo, a análise dos resultados evidencia não só a viabilidade da inclusão das métricas relacionadas ao ecossistema *IoT*, como sua importância para identificação de problemas de rede no âmbito de fábricas inteligentes. Além disso, ressalta-se que a integração destes dados no sistema de análise é transparente à arquitetura de rede 5G já existente.

Outra contribuição importante é a arquitetura utilizada para implantação e execução do experimento, que pode servir como base para trabalhos futuros, não só no âmbito de fábricas

inteligentes, como no cenário de análise de dados envolvendo núcleo 5G.

Ressalta-se, porém, que o amadurecimento dos projetos de código aberto, em especial nos quesitos de *Data Analytics*, pode fomentar trabalhos mais complexos no âmbito de gestão inteligente de rede e priorização automática de fluxos.

Observa-se o cumprimento dos requisitos elencados **R1** a **R4** a partir do sistema, modelo e experimento propostos e implantados. A partir destes, foi possível validar as hipóteses **H1** e **H2** que tratam da coleta de métricas do ecossistema e construção do modelo aprimorado na tarefa de classificação dos cenários de problemas de rede. Visto que os sistemas propostos foram implantados e sincronizados para a coleta dos dados, que foi feita de forma a não gerar sobrecarga nestes, validando a hipótese **H1**. A hipótese **H2** também foi validada a partir da observação dos resultados discutidos anteriormente, evidenciando a importância dos dados dos dispositivos *IoT* no âmbito de fábricas inteligentes e redes de comunicação presentes neste cenário. Foi também possível observar a viabilidade da implantação destes sistemas utilizando ferramentas de código aberto, com as ressalvas mencionadas da necessidade de amadurecimento destes projetos para trabalhos futuros mais complexos.

Capítulo 6

Conclusão

Observando os resultados obtidos, pode-se concluir que a integração entre as métricas dos componentes da rede 5G (núcleo 5G e seus serviços) e as métricas coletadas dos dispositivos que realizam comunicação entre si a partir desta rede, resulta em informações que são de grande valor para a tarefa de identificação de problemas da rede de comunicação destes dispositivos. Em outras palavras, a integração destas informações é relevante para a tomada de decisões no âmbito de reconfiguração autônoma da rede.

6.1 Resumo dos Resultados

Foi realizado um experimento no qual serviços de um núcleo 5G foram implantados para simulação de uma rede 5G, num cenário em que dispositivos *IIoT*, também simulados, realizaram comunicação entre si em diferentes contextos de problemas introduzidos nesta rede de comunicação. A partir deste experimento foram coletadas métricas dos serviços que compõem o núcleo 5G e também as estatísticas advindas do ecossistema *IIoT*. Estes dados foram utilizados no treinamento de modelos de rede neural na tarefa de classificação dos cenários de problemas de rede.

Dois modelos de rede neural do tipo LSTM (*Long Short-Term Memory*), capazes de identificar padrões em séries temporais, foram treinados para realizar a tarefa de classificação de cenários de problemas de rede. Apenas um dos modelos foi treinado utilizando a integração entre os dados da rede e da comunicação dos dispositivos, apresentando ganho de acurácia de 12% na acurácia média e 15% no *recall* médio. Em termos individuais, houve incremento

na acurácia de 19% para o cenário de perda de pacotes e 43% para o cenário de congestionamento de banda. Observando a métrica de *recall*, foi obtida melhoria no cenário de perda de pacotes de 72%, em comparação ao outro que foi treinado apenas com os dados do núcleo 5G.

A partir destes resultados, ressalta-se a relevância da inclusão das métricas do ecossistema *IIoT* na tarefa de identificação de problemas de rede, que pode levar a viabilizar priorização automática de fluxos e possivelmente outros cenários.

Em resgate às questões de pesquisa (**QP**) observa-se a comprovação de que é viável realizar a coleta e integração dos dados do ecossistema *IIoT* e do núcleo 5G sem que haja interferência na comunicação destes sistemas (**QP1**). Além disso, os dados coletados foram importantes para promover melhorias nos indicadores de acurácia, precisão e sensibilidade na tarefa de classificação de cenários de problemas de rede (**QP2**), que é definido como um problema complexo de classificação multiclasse em dados de série temporal. Por fim, embora ressaltadas as limitações encontradas no cenário atual de projetos de código aberto para implementações do núcleo 5G e *gNodeB*, principalmente no âmbito de coleta e análise de métricas, foi possível atingir o objetivo proposto na questão de pesquisa **QP3** com a implantação efetiva do ecossistema conjunto de rede 5G e aplicações de fábrica inteligente.

6.2 Trabalhos Futuros

Como discutido anteriormente, a observação do presente estado dos projetos de código aberto relacionados à implementação de núcleo 5G levou a notar a carência de implementação de serviços que lidam com a geração, coleta e análise de estatísticas da rede, como é o caso da ausência de implementações para o serviço NWDAF (*Network Data Analytics Function*), que atua diretamente com a análise de dados para promover otimizações na rede. Outro aspecto que ainda segue em desenvolvimento são as funcionalidades relacionadas às definições de *QoS* (*Quality of Service*) que não puderam ser exploradas no experimento deste trabalho, e que enriqueceria os cenários de problema definidos. Ademais, a disponibilidade de outras estatísticas de rede e um maior tempo de coleta para os dados seria de grande valor para ampliar a complexidade dos modelos e melhorar sua acurácia.

Além disso, dada a completa implementação dos serviços do núcleo 5G, poderiam ser

explorados os gatilhos presentes nestes para reconfiguração automática da rede em aspectos como o controle de fluxos e realocamento de *UE (User Equipment)* em diferentes *UPF* de acordo com definições de intenções ou objetivos de negócios.

Em resumo, observou-se a partir deste trabalho o grande potencial desta área de pesquisa para atingir objetivos de automação e gestão inteligente de redes, porém, o cenário de projetos de código aberto ainda precisa de contribuições relevantes para fomentar esta evolução.

Bibliografia

- [1] 3GPP. System architecture for the 5G System (5GS). Technical Specification (TS) 23.501, 3rd Generation Partnership Project (3GPP), 2023. URL <http://www.3gpp.org/DynaReport/23501.htm>.
- [2] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., e Zheng, X. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL <https://www.tensorflow.org/>. Software available from tensorflow.org.
- [3] Abdelatti, M. e Sodhi, M. Lab-scale smart factory implementation using ros. In *Robot Operating System (ROS) The Complete Reference (Volume 7)*, pages 119–143. Springer, 2023.
- [4] Ahmed, M., Naser Mahmood, A., e Hu, J. A survey of network anomaly detection techniques. *Journal of Network and Computer Applications*, 60:19–31, 2016. ISSN 1084-8045. doi: <https://doi.org/10.1016/j.jnca.2015.11.016>. URL <https://www.sciencedirect.com/science/article/pii/S1084804515002891>.
- [5] Bergmeir, C. e Benítez, J. M. On the use of cross-validation for time series predictor evaluation. *Information Sciences*, 191:192–213, 2012. ISSN 0020-0255. doi: <https://doi.org/10.1016/j.ins.2011.12.028>. URL <https://www.sciencedirect.com/>

- [science/article/pii/S0020025511006773](https://doi.org/10.1016/j.dajour.2021.100015). Data Mining for Software Trustworthiness.
- [6] Boyes, H., Hallaq, B., Cunningham, J., e Watson, T. The industrial internet of things (iiot): An analysis framework. *Computers in industry*, 101:1–12, 2018.
- [7] Brown, G. Service-based architecture for 5g core networks. *Huawei White Paper*, 1, 2017.
- [8] Brown, G. Private 5g mobile networks for industrial iot. *Heavy Reading, White Paper, Qualcomm Inc*, 2019.
- [9] Canonical. Multipass. <https://github.com/canonical/multipass>, 2024. Acesso em 18 Janeiro 2024.
- [10] Chhajer, P., Shah, M., e Kshirsagar, A. The applications of artificial neural networks, support vector machines, and long–short term memory for stock market prediction. *Decision Analytics Journal*, 2:100015, 2022. ISSN 2772-6622. doi: <https://doi.org/10.1016/j.dajour.2021.100015>. URL <https://www.sciencedirect.com/science/article/pii/S2772662221000102>.
- [11] Chollet, F. et al. Keras. <https://github.com/fchollet/keras>, 2015.
- [12] eProxima. Fastdds: High-performance, open-source dds implementation. <https://fast-dds.docs.eprosima.com/en/latest/>, 2024. Acesso em 18 Janeiro 2024.
- [13] Free5GC. Free5gc: Open source 5g core network. <https://www.free5gc.org/>, 2024. Acesso em 18 Janeiro 2024.
- [14] Hochreiter, S. e Schmidhuber, J. Long Short-Term Memory. *Neural Computation*, 9 (8):1735–1780, 11 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735. URL <https://doi.org/10.1162/neco.1997.9.8.1735>.
- [15] Hozdić, E. Smart factory for industry 4.0: A review. *International Journal of Modern Manufacturing Technologies*, 7(1):28–35, 2015.

- [16] Ismail Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., e Muller, P.-A. Deep learning for time series classification: a review. *Data mining and knowledge discovery*, 33(4): 917–963, 2019.
- [17] Kaltenberger, F., Souza, G. d., Knopp, R., e Wang, H. The openairinterface 5g new radio implementation: Current status and roadmap. In *WSA 2019; 23rd International ITG Workshop on Smart Antennas*, pages 1–5, 2019.
- [18] Kontio, J. et al. *Software engineering risk management: a method, improvement framework, and empirical evaluation*. Helsinki University of Technology, 2001.
- [19] Kuznetsov, A. N. *tc - show / manipulate traffic control settings*, Accessed 2024. URL <https://www.man7.org/linux/man-pages/man8/tc.8.html>, Linux manual page for tc command.
- [20] Macenski, S., Foote, T., Gerkey, B., Lalancette, C., e Woodall, W. Robot operating system 2: Design, architecture, and uses in the wild. *Science Robotics*, 7(66):eabm6074, 2022. doi: 10.1126/scirobotics.abm6074. URL <https://www.science.org/doi/abs/10.1126/scirobotics.abm6074>.
- [21] Maman, M., Calvanese-Strinati, E., Dinh, L. N., Haustein, T., Keusgen, W., Wittig, S., Schmieder, M., Barbarossa, S., Merluzzi, M., Costanzo, F., et al. Beyond private 5g networks: applications, architectures, operator models and technological enablers. *EURASIP Journal on Wireless Communications and Networking*, 2021(1):1–46, 2021.
- [22] Nagorny, K., Lima-Monteiro, P., Barata, J., e Colombo, A. W. Big data analysis in smart manufacturing: A review. *International Journal of Communications, Network and System Sciences*, 10(3):31–58, 2017.
- [23] O’Malley, T., Bursztein, E., Long, J., Chollet, F., Jin, H., Invernizzi, L., et al. Keras Tuner. <https://github.com/keras-team/keras-tuner>, 2019.
- [24] ONF project. 5G Test tool - gNBSim. <https://github.com/omec-project/gnbsim>, 2022. Acesso em 18 Janeiro 2024.
- [25] Open Networking Foundation. Aether project. <https://opennetworking.org/aether/>, 2023. Acesso em 18 Janeiro 2024.

- [26] Open5GS. Open5gs: A 5gc and epc open source software. <https://open5gs.org/>, 2024. Acesso em 18 Janeiro 2024.
- [27] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., e Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [28] Pemmaiah, A., Pangercic, D., Aggarwal, D., Neumann, K., e Marcey, K. Performance testing in ros 2. Technical report, Apex.AI Inc., Palo Alto, CA, USA, 2020. URL <https://www.apex.ai/post/performance-testing-in-ros-2>.
- [29] Prometheus. Prometheus: An open-source systems monitoring and alerting toolkit. <https://prometheus.io/>, 2024. Acesso em 18 Janeiro 2024.
- [30] Rao, S. K. e Prasad, R. Impact of 5g technologies on industry 4.0. *Wireless personal communications*, 100:145–159, 2018.
- [31] Reddy, R., Gundall, M., Lipps, C., e Schotten, H. D. Open source 5g core network implementations: A qualitative and quantitative analysis. In *2023 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom)*, pages 253–258. IEEE, 2023.
- [32] Ren, H., Xu, B., Wang, Y., Yi, C., Huang, C., Kou, X., Xing, T., Yang, M., Tong, J., e Zhang, Q. Time-series anomaly detection service at microsoft. *CoRR*, abs/1906.03821, 2019. URL <http://arxiv.org/abs/1906.03821>.
- [33] Ruiz, A. P., Flynn, M., Large, J., Middlehurst, M., e Bagnall, A. The great multivariate time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data Mining and Knowledge Discovery*, 35(2):401–449, 2021.
- [34] Sachs, J. e Landernäs, K. Review of 5g capabilities for smart manufacturing. In *2021 17th International Symposium on Wireless Communication Systems (ISWCS)*, pages 1–6. IEEE, 2021.

- [35] Saha, B. K., Tandur, D., Haab, L., e Podleski, L. Intent-based networks: An industrial perspective. In *Proceedings of the 1st International Workshop on Future Industrial Communication Networks*, pages 35–40, 2018.
- [36] Sahoo, S. e Lo, C.-Y. Smart manufacturing powered by recent technological advancements: A review. *Journal of Manufacturing Systems*, 64:236–250, 2022.
- [37] Schmidt, R. M. Recurrent neural networks (rnns): A gentle introduction and overview, 2019.
- [38] Sevgican, S., Turan, M., Gökarslan, K., Yilmaz, H. B., e Tugcu, T. Intelligent network data analytics function in 5g cellular networks using machine learning. *Journal of Communications and Networks*, 22(3):269–280, 2020. doi: 10.1109/JCN.2020.000019.
- [39] Shafique, K., Khawaja, B. A., Sabir, F., Qazi, S., e Mustaqim, M. Internet of things (iot) for next-generation smart systems: a review of current challenges, future trends and prospects for emerging 5g-iot scenarios. *IEEE Access*, 8:23022–23040, 2020. doi: 10.1109/access.2020.2970118.
- [40] Silveira, L. B., de Resende, H. C., Both, C. B., Marquez-Barja, J. M., Silvestre, B., e Cardoso, K. V. Tutorial on communication between access networks and the 5g core. *Computer Networks*, 216:109301, 2022. ISSN 1389-1286. doi: <https://doi.org/10.1016/j.comnet.2022.109301>. URL <https://www.sciencedirect.com/science/article/pii/S1389128622003528>.
- [41] Taleb, T., Afolabi, I., e Bagaa, M. Orchestrating 5g network slices to support industrial internet and to shape next-generation smart factories. *Ieee Network*, 33(4):146–154, 2019.
- [42] Temesvári, Z. M., Maros, D., e Kádár, P. Review of mobile communication and the 5g in manufacturing. *Procedia Manufacturing*, 32:600–612, 2019.
- [43] UERANSIM. Ueransim: Ue and ran simulator for 5g. <https://github.com/aligungr/UERANSIM>, 2024. Acesso em 18 Janeiro 2024.
- [44] Urrea, C. e Kern, J. Integrating ros and iot in a virtual laboratory for control system engineering. *Journal of Applied Mathematics*, 2020:1–7, 2020.

- [45] Wang, D., Chen, D., Song, B., Guizani, N., Yu, X., e Du, X. From iot to 5g i-iot: The next generation iot-based intelligent algorithms and 5g technologies. *IEEE Communications Magazine*, 56(10):114–120, 2018. doi: 10.1109/MCOM.2018.1701310.
- [46] Wang, Z., Yan, W., e Oates, T. Time series classification from scratch with deep neural networks: A strong baseline. In *2017 International joint conference on neural networks (IJCNN)*, pages 1578–1585. IEEE, 2017.
- [47] Wei, Y., Peng, M., e Liu, Y. Intent-based networks for 6g: Insights and challenges. *Digital Communications and Networks*, 6(3):270–280, 2020.
- [48] Wijethilaka, S. e Liyanage, M. Survey on network slicing for internet of things realization in 5g networks. *IEEE Communications Surveys & Tutorials*, 23(2):957–994, 2021.
- [49] Zhang, C., Song, D., Chen, Y., Feng, X., Lumezanu, C., Cheng, W., Ni, J., Zong, B., Chen, H., e Chawla, N. V. A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):1409–1416, Jul. 2019. doi: 10.1609/aaai.v33i01.33011409. URL <https://ojs.aaai.org/index.php/AAAI/article/view/3942>.
- [50] Zhong, M., Yang, Y., Yao, H., Fu, X., Dobre, O. A., e Postolache, O. 5g and iot: Towards a new era of communications and measurements. *IEEE Instrumentation & Measurement Magazine*, 22(6):18–26, 2019.