UNIVERSIDADE ESTADUAL PAULISTA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO



CÉSAR ROMANO DELGADO MARÍN

REDES NEURAIS BASEADAS EM GRAFOS ESPAÇOTEMPORAIS PARA A PREVISÃO DE AGENTES INFECCIOSOS EM AMBIENTES HOSPITALARES

INSTITUTO DE GEOCIÊNCIAS E CIÊNCIAS EXATAS

RIO CLARO - (SP)

2026

UNIVERSIDADE ESTADUAL PAULISTA
"Júlio de Mesquita Filho"
Instituto de Geociências e Ciências Exatas
Câmpus de Rio Claro

César Romano Delgado Marín

Redes Neurais Baseadas em Grafos Espaçotemporais para a Previsão de Agentes Infecciosos em Ambientes Hospitalares

Dissertação de Mestrado apresentada ao Instituto de Geociências e Ciências Exatas do Câmpus de Rio Claro, da Universidade Estadual Paulista "Júlio de Mesquita Filho", como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação.

Orientador: Prof. Dr. Denis Henrique Pinheiro Salvadeo

Rio Claro - SP
2026

M337r

Marin, Cesar Romano Delgado

Redes Neurais baseadas em grafos espaçotemporais para a previsão de agentes infecciosos em ambientes hospitalares / Cesar Romano Delgado Marin. -- Rio Claro, 2026

143 p.

Dissertação (mestrado) - Universidade Estadual Paulista (UNESP), Instituto de Geociências e Ciências Exatas, Rio Claro

Orientador: Denis Henrique Pinheiro Salvadeo

1. Aprendizagem profunda. 2. Teoria dos grafos. 3. Vigilância epidemiológica. 4. Agente Infeccioso. I. Título.

Sistema de geração automática de fichas catalográficas da Unesp. Dados fornecidos pelo autor(a).

IMPACTO POTENCIAL DESTA PESQUISA

Esta pesquisa pode motivar o uso de técnicas de aprendizado profundo baseadas em grafos para modelar dinâmicas espaçotemporais complexas em ambientes hospitalares como a vigilância bacteriana hospitalar, ao mesmo tempo, seus resultados podem orientar estudos futuros e auxiliar a priorização de ações de prevenção e controle de infecções.

POTENTIAL IMPACT OF THIS RESEARCH

This research may motivate the use of graph-based deep learning techniques to model complex spatiotemporal dynamics in hospital environments such as hospital bacterial surveillance, while its results may guide future studies and assist in prioritizing infection prevention and control actions.

UNIVERSIDADE ESTADUAL PAULISTA
"Júlio de Mesquita Filho"
Instituto de Geociências e Ciências Exatas
Câmpus de Rio Claro

César Romano Delgado Marín

Redes Neurais Baseadas em Grafos Espaçotemporais para a Previsão de Agentes Infecciosos em Ambientes Hospitalares

Dissertação de Mestrado apresentada ao Instituto de Geociências e Ciências Exatas do Câmpus de Rio Claro, da Universidade Estadual Paulista "Júlio de Mesquita Filho", como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação.

Comissão Examinadora

Prof. Dr. DANIEL CARLOS GUIMARÃES PEDRONETTE
IGCE/UNESP/Rio Claro (SP)

Prof. Dra. MARIANA RECAMONDE MENDOZA
INF/UFRGS/Porto Alegre (RS)

Prof. Dr. VANDER LUIS DE SOUZA FREITAS
DECOM/UFOP/Ouro Preto (MG)

Conceito: Aprovado.

Rio Claro (SP), 28 de Janeiro de 2026.

Agradecimentos

Agradeço ao Professor Doutor Julio César Delgado Correal, à Professora Doutora Marisa Zenaide Ribeiro Gomes, ao Professor Doutor Eduardo Almeida Ribeiro de Castro e a todo o time e colaboradores da Comissão de Controle de Infecção Hospitalar (CCIH) que participaram do desenvolvimento deste trabalho.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

Declaração de uso de inteligência artificial

Declaro que foi utilizada a inteligência artificial CahtGPT da OpenAi, versão GPT-4 como apoio durante a elaboração deste trabalho, especificamente para revisão linguística, organização textual, sugestões de redação acadêmica, auxílio na estruturação de pseudocódigos, geração de scripts auxiliares e apoio à interpretação e documentação do código desenvolvido. O conteúdo final, as decisões metodológicas, a análise dos resultados, a verificação das informações e a responsabilidade acadêmica pelo trabalho permanecem integralmente sob responsabilidade do autor.

Resumo

O aumento da resistência bacteriana é um problema endêmico em ambientes hospitalares, afetando principalmente pacientes imunossuprimidos, e que tem se intensificado nas últimas décadas. Durante a pandemia da COVID-19, esse fenômeno foi agravado, ao mesmo tempo em que surgiram novos modelos de previsão de epidemias baseados em redes neurais artificiais. Em termos gerais, tais modelos apresentam desempenho superior aos métodos estatísticos tradicionais, e sua aplicação em ambientes fechados e monitorados, como hospitais, pode contribuir para o controle de agentes infecciosos. Dentre essas abordagens, as redes neurais baseadas em grafos têm ganhado destaque por sua capacidade de representar dinâmicas complexas do mundo real. Nesse contexto, este trabalho investiga modelos de aprendizado profundo sobre grafos dinâmicos discretos para a previsão e o monitoramento da presença de espécies bacterianas em enfermarias de um hospital do Rio de Janeiro. As contribuições deste trabalho são: (i) formular a vigilância intra-hospitalar de espécies bacterianas como uma tarefa de previsão espaço-temporal multirrótulo em nível de enfermaria, sobre uma sequência de grafos construída a partir de dados rotineiros de microbiologia, ocupação e transferências de pacientes; (ii) comparar uma linha de base por média histórica com arquiteturas espaciais, temporais e espaço-temporais baseadas em GraphSAGE, GRU, atenção temporal e GAT, considerando diferentes horizontes de previsão, definições de vizinhança e representações de atributos clínico-microbiológicos; e (iii) analisar o impacto de inconsistências nas movimentações de pacientes sobre a coerência da sequência temporal e o desempenho preditivo. Os resultados indicam que a dinâmica temporal das enfermarias é o principal sinal preditivo, com melhor desempenho obtido pelo modelo GRU nos horizontes de três, sete e quinze dias. A estrutura espacial do hospital e os atributos microbiológicos enriquecidos contribuem de forma complementar, mas os modelos baseados em grafos não superaram a arquitetura temporal pura nas configurações avaliadas.

Palavras-chave: Redes neurais baseadas em grafos, GNN, Redes dinâmicas discretas, Agentes infecciosos, Espécies bacterianas, Resistência antimicrobiana, Ambiente hospitalar, Previsão multirrótulo, Aprendizado profundo

Abstract

The increase in bacterial resistance is an endemic problem in hospital settings, primarily affecting immunocompromised patients, and has intensified over the past decades. During the COVID-19 pandemic, this phenomenon was further aggravated, while new epidemic forecasting models based on artificial neural networks were developed. In general, such models achieve better performance than traditional statistical approaches, and their application in closed and monitored environments, such as hospitals, may support the control of infectious agents. Among these approaches, graph-based neural networks have gained prominence due to their ability to represent complex real-world dynamics. In this context, this work investigates deep learning models over discrete dynamic graphs for forecasting and monitoring the presence of bacterial species across wards in a hospital in Rio de Janeiro. The contributions of this work are: (i) we formulate intra-hospital bacterial surveillance as a ward-level spatiotemporal multilabel forecasting task over a sequence of graphs constructed from routinely collected microbiology, occupancy, and patient transfer data; (ii) we compare a historical-mean baseline with spatial, temporal, and spatiotemporal architectures based on GraphSAGE, GRU, temporal attention, and GAT, considering different forecasting horizons, neighborhood definitions, and clinical-microbiological attribute representations; and (iii) we analyze the impact of inconsistencies in patient movement records on the coherence of the temporal sequence and on predictive performance. The results indicate that ward-level temporal dynamics are the dominant predictive signal, with the GRU model achieving the best performance for three-, seven-, and fifteen-day horizons. The hospital spatial structure and enriched microbiological attributes provide complementary information, but graph-based models did not outperform the purely temporal architecture in the evaluated configurations.

Keywords: Graph Neural Networks, Discrete Dynamic Networks, Infectious agents, Bacterial species, Antimicrobial resistance, Hospital environment, Multilabel forecasting, Deep Learning

Lista de ilustrações

Figura 1 – Categorias de bactérias multi-resistentes 21
Figura 2 – Fluxo local GLASS 23
Figura 3 – Sarcoma em New York 26
Figura 4 – Modelos de previsão por compartimentos 29
Figura 5 – Grafo não dirigido 31
Figura 6 – Aresta 32
Figura 7 – Grafo dirigido 32
Figura 8 – Passeio 33
Figura 9 – Trilha 33
Figura 10 – Caminho 33
Figura 11 – Subgrafo 34
Figura 12 – Grafo dinâmico discreto 36
Figura 13 – Perceptron 39
Figura 14 – Rede neural multicamada 41
Figura 15 – Operação de convolução 42
Figura 16 – Conexões esparsas em uma CNN 42
Figura 17 – Janela *kernel* 43
Figura 18 – *Max pooling* 43
Figura 19 – Rede neural convolucional 44
Figura 20 – Rede neural recorrente 45
Figura 21 – Arquitetura RNN 46
Figura 22 – Desaparecimento dos gradientes 46
Figura 23 – Célula LSTM 47
Figura 24 – Fluxo de informação na LSTM 49
Figura 25 – Unidade oculta GRU 50
Figura 26 – Modelo *Transformer* 52
Figura 27 – Atenção de várias cabeças 53
Figura 28 – Embeddings de grafos 56
Figura 29 – Arquitetura de embeddings de grafos 56
Figura 30 – Reconstrução de vizinhança 57
Figura 31 – Fatoração de matrizes 58
Figura 32 – *DeepWalk* 60
Figura 33 – Amostragem BFS e DFS 61
Figura 34 – Controle do node2vec 61
Figura 35 – *embeddings* vs MPNN 66
Figura 36 – GNN recorrente vs. ConvGNN 68

Figura 37 – Sinal de um grafo 69
Figura 38 – Convolução em imagem e grafo 69
Figura 39 – Passagem de mensagens 70
Figura 40 – GAT mecanismo de atenção 73
Figura 41 – GAT de várias cabeças 73
Figura 42 – Arquitetura Cola-GNN 75
Figura 43 – Arestas espaciais e temporais 77
Figura 44 – Passagem de nó semente 78
Figura 45 – Reconstrução do estado populacional 79
Figura 46 – Modelo de propagação com GNN 79
Figura 47 – Georreferenciamento hospitalar 81
Figura 48 – Grafo fixo do hospital 84
Figura 49 – Snapshot com movimentações 87
Figura 50 – Embedding bacteriano 90
Figura 51 – Agregação de perfis por enfermaria 91
Figura 52 – Codificador espacial 92
Figura 53 – Codificador temporal 92
Figura 54 – Decodificador temporal 93
Figura 55 – Desempenho por arquitetura 98
Figura 56 – Diagrama completo do modelo 128

Lista de abreviaturas e siglas

ADAM	Método de otimização por estimação adaptativa de momentos
AMR	Resistência Antimicrobiana
AR	Autorregressivo
ARIMA	Modelo Autorregressivo Integrado de Médias Móveis
ARMA	Média Móvel Autorregressiva
AST	Teste de Suscetibilidade Antimicrobiana
AUPRC	Área sob a Curva Precisão-Revocação
BCE	Entropia Cruzada Binária
BFS	Amostragem em Largura
BiNE	*Embedding* de Rede Bipartita
BRCAST	Comitê Brasileiro de Testes de Suscetibilidade aos Antimicrobianos
CAAE	Certificado de Apresentação de Apreciação Ética
CAD	Desenho Assistido por Computador
CAPES	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
CCIH	Comissão de Controle de Infecção Hospitalar
CLSI	Instituto de Padrões Clínicos e Laboratoriais
CNN	Rede Neural Convolucional
COVID-19	Doença causada pelo coronavírus SARS-CoV-2
CSV	Valores Separados por Vírgula
CT	Tomografia Computadorizada
DFS	Amostragem em Profundidade
DHNE	*Deep Hyper-Network Embedding*
EHRs	Registros Eletrônicos de Saúde

EUCAST	Comitê Europeu de Testes de Suscetibilidade Antimicrobiana
FNN	Rede Neural de Avanço
fMRI	Ressonância Magnética funcional
GAR	Autorregressão Global
GAT	Rede de Atenção em Grafos
GCN	Rede Neural Convolucional em Grafos
GCNNs	Redes Neurais Convolucionais em Grafos
GIS	Sistema de Informação Geográfico
GLASS	Sistema Global de Vigilância do Uso e Resistência de Antimicrobianos
GNB	Bactéria Gram-negativa
GNN	Rede Neural Baseada em Grafos
GraphSAGE	Modelo de Amostra e Agregação
GRU	Unidade Recorrente Fechada
HNE	*Embedding* de Rede Heterogênea
IA	Inteligência Artificial
LSTM	Memória de Longo-Curto Prazo
MAE	Erro Médio Absoluto
MDR	Resistente a Múltiplas Drogas
MDR-GNB	Bactéria Gram-negativa multirresistente
MDROs	Organismos Multirresistentes
MLP	Rede Neural Multicamada
MPNN	Rede Neural de Passagem de Mensagens
MPGNN	Rede Neural de Passagem de Mensagens em Grafos
MRI	Imageamento por Ressonância Magnética
MS	Multissensível
MSE	Erro Quadrático Médio

MSLE	Erro Quadrático Logarítmico Médio
NPI	Intervenção Não Farmacêutica
PCC	Correlação de Pearson
PDR	Bactérias Pan-Resistentes
QGIS	Sistema de Informação Geográfica de Código Aberto
RMSE	Raiz do Erro Médio Quadrático
RNN	Rede Neural Recorrente
ReLU	Unidade Linear Retificada
SATSCAN	Software para análise espacial, temporal e espaço-temporal
SCNN	Rede Neural Convolucional Espectral
SEIR	Suscetível-Exposto-Infectado-Recuperado
SEIRD	Suscetível-Exposto-Infectado-Recuperado-Morto
SIR	Suscetível-Infectado-Recuperado
SIRS	Suscetível-Infectado-Recuperado-Suscetível
SIS	Suscetível-Infectado-Suscetível
SSL	Aprendizado Fracamente Supervisionado
SSE	Soma dos Erros Quadráticos
SVM	Máquina de Vetores de Suporte
US	Ultrassom
UTI	Unidade de Terapia Intensiva
VAR	Autorregressão Vetorizada
WHONET	Sistema para gerenciamento e análise de dados microbiológicos
WHO	Organização Mundial da Saúde
XDR	Bactérias Amplamente Resistentes

Sumário

1 INTRODUÇÃO 16

2 RESISTÊNCIA ANTIMICROBIANA E PREVISÃO DE EPIDEMIAS 20
2.1 Resistência antimicrobiana (AMR) 20
2.2 Epidemias 26
2.3 Modelos tradicionais de previsão de epidemias 28

3 ASPECTOS TEÓRICOS 31
3.1 Teoria de grafos 31
3.1.1 Definições, Propriedades e Medidas 31
3.1.2 Representações 33
3.1.3 Grafos Dinâmicos Discretos 36
3.2 Redes neurais artificiais 37
3.2.1 Tipos de Aprendizado 37
3.2.2 O Perceptron e a Rede Neural Multicamada (MLP) 39
3.2.3 Rede Neural Convolucional (CNN) 40
3.2.4 Redes Neurais Recorrentes (RNNs) 44
3.2.5 Modelo *Transformer* e mecanismos de atenção 50
3.3 *Embeddings* de Grafos 55
3.3.1 Técnicas de *Embeddings* de Grafos 57
3.3.2 Aplicações de *Embeddings* de Grafos 63
3.3.3 Limitações 64

4 REDES NEURAIS BASEADAS EM GRAFOS (GNNS) E PREVISÃO DE EPIDEMIAS 66
4.1 Redes Neurais Convolucionais Baseadas em Grafos 67
4.2 Redes Neurais Baseadas em Grafos e mecanismos de atenção . . . 72
4.3 Modelos de previsão de epidemias baseados em GNNs 74

5 JUSTIFICATIVA, HIPÓTESES E METODOLOGIA 81
5.1 Conjunto de dados e configuração do hospital 82
5.2 Construção do grafo 83
5.3 Definição do problema de previsão 87
5.4 Modelo espaço-temporal baseado em grafos 88
5.5 Considerações éticas 91

6 EXPERIMENTOS E RESULTADOS 94

6.1 Configuração do experimento 94
6.2 Resultados da previsão 97
6.3 Inconsistências nas movimentações de pacientes 99
6.4 Discussão dos resultados 100

7 CONCLUSÕES, LIMITAÇÕES E TRABALHOS FUTUROS 102
7.1 Limitações 103
7.2 Trabalhos futuros 104

REFERÊNCIAS 106

APÊNDICE A – PSEUDOCÓDIGOS DOS ALGORITMOS 118

APÊNDICE B – VOCABULÁRIO BACTERIANO 124

APÊNDICE C – DIAGRAMA COMPLETO DO MODELO 127

ANEXO A – PARECER CONSUBSTANCIADO DO COMITÊ DE ÉTICA 129

ANEXO B – RESULTADOS COMPLETOS DAS EXPERIMENTAÇÕES 136

DADOS CURRICULARES 142

1 Introdução

Desde o nascimento da computação é latente o desejo de dotar os computadores com capacidades cognitivas. O cálculo de operações leva em si essa influência básica, mas é a partir das redes neurais artificiais que essa ideia é expressa com maior sentido. Apesar de que a idealização do primeiro neurônio artificial, o Perceptron, tenha acontecido há muito tempo (ROSENBLATT, 1962), é só a partir de avanços recentes na computação e do rápido aumento no fluxo de informação que as redes neurais artificiais se desdobraram nos mais variados cenários do mundo, apresentando habilidades para lidar com estruturas de informação como imagens ou textos (domínio no espaço Euclidiano).

Porém, este tipo de estrutura de informação não é o único que ocorre no mundo. Ele pode ser composto, em grande parte, por relações intricadas, algumas delas tão complexas e intercambiáveis que precisam de outras formas de representação (domínio no espaço não-Euclidiano). Os grafos, como estruturas capazes de modelar conjuntos de objetos (vértices) e suas relações (arestas), surgem como uma opção com suficiente poder de representação para sua modelagem (LIANG *et al.*, 2022). Mas para gerar aprendizados a partir destas estruturas, ainda é necessário levá-las para representações de menor dimensão. Os *embeddings* cumprem esta função, mapeando informações do grafo para um espaço vetorial que as redes neurais artificiais possam manipular, enquanto cuidam de preservar suas características e topologia.

Com o sucesso das técnicas de aprendizado de representações, as redes neurais profundas ganharam a habilidade de extrair representações de forma hierárquica, trazendo vantagens em termos de eficiência e generalização (CUN *et al.*, 1989; GOODFELLOW; BENGIO; COURVILLE, 2016). Rapidamente foram reunidos esforços com o propósito de aplicar a convolução em grafos. Como resultado foram desenvolvidos estudos a partir do domínio espectral e espacial dos grafos (BRUNA *et al.*, 2013; DEFFERRARD; BRESSON; VANDERGHEYNST, 2016; KIPF; WELLING, 2016; LEVIE *et al.*, 2018; ATWOOD; TOWSLEY, 2016; NIEPERT; AHMED; KUTZKOV, 2016; HAMILTON; YING; LESKOVEC, 2017), permitindo o surgimento de poderosos modelos de redes neurais baseados em grafos que apresentam aplicações importantes em diversas áreas (WAIKHOM; PATGIRI, 2021; ZHOU *et al.*, 2020).

Um exemplo recente de aplicação foi o observado durante a pandemia da COVID-19 em que vários destes modelos foram desenvolvidos com o objetivo de entender e prever sua propagação. A partir de informações de contágio e deslocamentos populacionais foram construídos grafos representando regiões como nós, deslocamentos como arestas e registros de contágio como características (KAPOOR *et al.*, 2020).

Atualmente, existe outro problema que pode se beneficiar de soluções baseadas em grafos. Há algumas décadas vem se apresentando uma emergência hospitalar devido ao aumento na resistência antimicrobiana (AMR) (LIVERMORE, 2012), um problema que a Organização Mundial da Saúde (WHO) classifica como uma das 10 ameaças mais sérias para a saúde pública que a humanidade atualmente enfrenta (ECDC, 2022), e que durante a pandemia da COVID-19 se intensificou, levando para um incremento de casos de patógenos como *Klebsiella pneumoniae* especialmente entre os anos 2020 e 2021 (CHAGAS *et al.*, 2024; GASPAR *et al.*, 2021). No Brasil, o número de casos foi triplicado entre os anos 2019 e 2023 (MENEZES, 2021).

A criação de novos antibióticos baseados na pesquisa dos genes AMR é um dos enfoques mais comuns no tratamento do problema. Vários autores (KATTIS *et al.*, 2016) têm aproveitado as recentes contribuições de métodos de aprendizado profundo para o desenvolvimento e descoberta de drogas (por exemplo, a criação da molécula *Halicin* usando GNNs) (STOKES *et al.*, 2020). Porém, dado que o problema é causado em parte por um uso e prescrição de antimicrobianos inadequados, intervenções não farmacêuticas (NPI) como a vigilância e previsão destes organismos são recomendadas (WHO, 2014).

Tradicionalmente, a vigilância e previsão da AMR é feita de forma retrospectiva usando análises estatísticas baseadas em séries temporais (LÓPEZ-LOZANO *et al.*, 2019; GHARBI *et al.*, 2015). Do outro lado, técnicas de IA como regressão linear e logística, árvores de decisão, comitês classificadores, florestas aleatórias e, em menor medida, redes neurais artificiais (MLP e CNN) têm sido utilizadas para atacar o problema da AMR, principalmente identificando fatores de risco e recomendando tratamentos adequados (GIACOBBE *et al.*, 2020). Até onde foi consultado, não há registro do uso de redes neurais baseadas em grafos para este problema.

No contexto hospitalar investigado, a motivação para uma abordagem baseada em grafos vem de estudo anterior realizado em um hospital público, federal e terciário do Rio de Janeiro, no qual a circulação intra-hospitalar do complexo *Klebsiella pneumoniae* foi analisada por meio de georreferenciamento e estatística espacial (SILVA *et al.*, 2021). Esse estudo mostrou que a combinação entre localização das enfermarias, movimentação de pacientes e ocorrência de perfis de susceptibilidade antimicrobiana permitia representar espacialmente agrupamentos infecciosos e sugeria o potencial de sistemas de vigilância como ferramentas de aviso prévio. A evolução posterior do agente para perfis XDR/PDR, associada a focos de infecção de difícil tratamento (GOMES *et al.*, 2023), reforça a necessidade de modelos capazes de explorar simultaneamente a estrutura física do hospital, a dinâmica temporal das movimentações e a presença microbiológica observada.

Com base nessa motivação, este trabalho parte da hipótese de que técnicas de aprendizado profundo baseadas em grafos podem capturar padrões espaço-temporais de circulação de espécies bacterianas no ambiente intra-hospitalar. Também se investiga se

modelos baseados em grafos dinâmicos podem alcançar desempenho preditivo competitivo em relação a modelos tradicionais de previsão e a modelos neurais que não exploram explicitamente a estrutura espacial do hospital.

Assim, esta pesquisa investigou modelos de aprendizado profundo sobre grafos dinâmicos para a previsão e o monitoramento da presença de espécies bacterianas em um hospital do Rio de Janeiro, Brasil. Para isso, a estrutura física e operacional do hospital foi representada como uma sequência temporal de grafos, em que os nós correspondem às enfermarias, as arestas fixas representam relações de vizinhança espacial e de serviço, e as movimentações intra-hospitalares dos pacientes atualizam os estados das enfermarias ao longo do tempo. A partir de dados rotineiros de microbiologia, admissões, transferências, altas e óbitos, o problema foi formulado como uma tarefa de classificação multirrótulo em nível de nó, visando estimar a presença futura de espécies bacterianas em cada enfermaria para horizontes de três, sete e quinze dias. Nesta primeira abordagem, em função do quantitativo de dados disponível, a análise concentrou-se na distribuição de espécies bacterianas, sem restringir a previsão a um perfil específico de susceptibilidade antimicrobiana.

Foram comparadas diferentes abordagens de modelagem, incluindo uma linha de base por média histórica e arquiteturas neurais espaciais, temporais e espaço-temporais. O protocolo experimental avaliou 120 configurações neurais baseadas em GraphSAGE, GRU, atenção temporal e GAT, combinando duas definições de vizinhança, três horizontes de previsão, quatro configurações de atributos e cinco arquiteturas preditivas, além da linha de base por média histórica. A avaliação seguiu uma divisão cronológica dos dados em conjuntos de treinamento, validação e teste, de modo a preservar a causalidade temporal.

Além da análise de desempenho preditivo, o trabalho investigou o impacto da qualidade dos dados na aprendizagem em grafos dinâmicos, em particular os efeitos de inconsistências nos registros de transferência de pacientes. Os resultados indicam que a dinâmica temporal observada em cada enfermaria constitui o sinal preditivo dominante, com melhor desempenho do modelo GRU nos três horizontes avaliados. A estrutura espacial do hospital e a representação enriquecida dos atributos microbiológicos contribuíram de forma complementar, mas os modelos espaço-temporais mais complexos não superaram a arquitetura temporal nas configurações avaliadas. A remoção de movimentações duplicadas reduziu violações estruturais na ocupação hospitalar, embora seu impacto nas métricas de previsão tenha variado entre as arquiteturas. Esses achados reforçam tanto o potencial de modelos temporais e baseados em grafos para a vigilância epidemiológica hospitalar quanto a importância da consistência dos dados em aplicações reais.

O restante do trabalho está organizado da seguinte forma:

- o **Capítulo 2** aprofunda o problema da resistência antimicrobiana e apresenta os

métodos tradicionais para sua vigilância e previsão;

- o **Capítulo 3** apresenta aspectos teóricos relativos aos grafos, às redes neurais artificiais e *embeddings*;
- o **Capítulo 4** apresenta as intuições e formalismos das redes neurais baseadas em grafos e suas variações, como sua aplicação em casos relacionados com o problema de pesquisa;
- o **Capítulo 5** apresenta a justificativa do estudo, o problema de pesquisa, as hipóteses investigadas, a construção dos dados, a formulação da tarefa de previsão baseada em grafos dinâmicos, os componentes do modelo proposto e as considerações éticas;
- o **Capítulo 6** descreve a configuração experimental adotada, os conjuntos de dados utilizados, os modelos avaliados e os critérios de comparação, bem como apresenta, analisa e discute os resultados obtidos nos experimentos de previsão;
- por fim, o **Capítulo 7** sintetiza as principais contribuições do trabalho, apresenta as conclusões alcançadas, discute as principais limitações metodológicas e operacionais do estudo e aponta possíveis direções para trabalhos futuros.

2 Resistência Antimicrobiana e Previsão de Epidemias

Neste capítulo é abordado o problema da AMR, os organismos internacionais e as estratégias envolvidas na sua vigilância e monitoramento. Também são abordados alguns dos métodos frequentes de análise e previsão e estudos relacionados com IA. Dado que este é um problema epidemiológico, conceitos básicos de epidemias também são revisados. Finalmente, é feita uma menção sobre modelos epidêmicos tradicionais e algumas das suas características.

2.1 Resistência antimicrobiana (AMR)

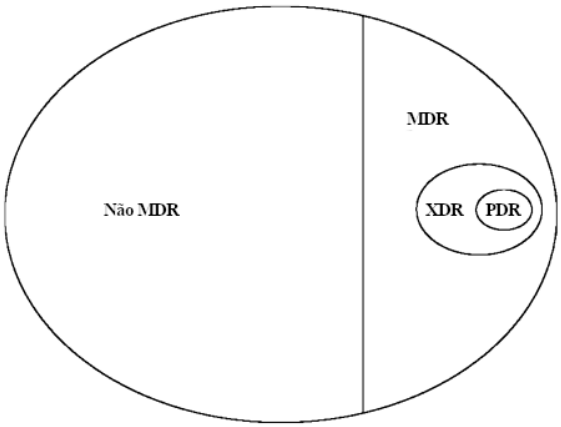
Segundo a Organização Mundial da Saúde (*World Health Organization - WHO*), AMR é uma das 10 ameaças mais sérias para a saúde pública que a humanidade atualmente enfrenta. Dada sua difícil erradicação, a WHO destaca a necessidade de melhorar esforços para prevenir e detectar a resistência, assim como o papel que a vigilância e o monitoramento podem desempenhar no fortalecimento da resiliência e preparação do sistema de saúde (ECDC, 2022). É importante estabelecer uma diferença entre os termos vigilância e monitoramento, dado que às vezes podem ser usados de forma homóloga, porém representam processos distintos. De acordo com Declich e Carter (1994), a vigilância epidemiológica está voltada para a observação contínua de eventos relacionados à saúde em populações, permitindo identificar mudanças na ocorrência de doenças e avaliar seus impactos ao longo do tempo. Por exemplo, ela pode ser utilizada para analisar o comportamento de uma doença infecciosa antes e depois da introdução de uma vacina. Enquanto o monitoramento está associado ao acompanhamento contínuo dos efeitos de uma intervenção, de um programa, de um serviço ou do desempenho de profissionais de saúde. No caso da vacinação, o monitoramento permite medir os resultados do programa implementado e realizar ajustes conforme o desempenho observado. Embora ambos utilizem a coleta rotineira e contínua de dados, geralmente por meio de métodos rápidos e pragmáticos, a vigilância tem como foco principal as populações, enquanto o monitoramento pode ser aplicado a grupos específicos, serviços, programas ou indivíduos, funcionando também como uma importante ferramenta de gestão.

Em termos gerais, a resistência a antibióticos é desenvolvida quando uma bactéria se adapta e cresce na presença de antibióticos, podendo ser espalhada no intercâmbio de material genético com outras bactérias. Isto em consequência afeta o tratamento por meio de antimicrobianos de várias infecções e doenças. Uma bactéria resistente pode circular

em populações de seres humanos e animais através da água e o ambiente, influenciada pelas interações entre humanos e animais e suas viagens e migrações. Também pode ser encontrada em comida animal e humana (WHO, 2015a).

Amostras de bactérias são submetidas a testes de suscetibilidade antimicrobiana (*Antimicrobial Susceptibility Test - AST*) para a identificação de AMR. Desta forma, uma bactéria ou grupo bacteriano é MDR quando apresenta não-suscetibilidade pelo menos a um agente em três ou mais categorias de antimicrobianos para os quais seriam usualmente susceptíveis. Bactérias com resistência estendida (*Extended Drug Resistance - XDR*) apresentam não-suscetibilidade pelo menos a um agente em todas as categorias de antimicrobianos, exceto duas ou menos. Finalmente, as bactérias pan-resistentes (PDR) apresentam não-suscetibilidade a todos os agentes em todas as categorias de antimicrobianos. Assim, XDR é um subconjunto de MDR e PDR é um subconjunto de XDR. A Figura 1 apresenta esta ideia (MAGIORAKOS *et al.*, 2012).

Figura 1 – Uma amostra bacteriana que é identificada como XDR também será considerada como MDR. Da mesma forma, uma amostra bacteriana tem que ser XDR antes de ser considerada PDR



Fonte: Traduzido de Magiorakos *et al.* (2012).

Em 2015, a WHO publicou o Plano Global contra a AMR (GAP-AMR) (WHO, 2015a), que abrange a participação de países membros e de indústrias, associações e fundações das áreas médica, farmacêutica e agrônoma. E que tem como objetivo geral "*garantir, pelo maior tempo possível, a continuidade da capacidade de tratar e prevenir doenças infecciosas com medicamentos eficazes, seguros, de qualidade, usados de forma responsável e acessíveis a todos aqueles que necessitam*". O plano é projetado a partir de cinco princípios: engajamento de toda a sociedade, prevenção em primeiro lugar, acesso a tratamentos existentes e novos e a serviços de saúde, sustentabilidade e metas incrementais para sua implementação.

Entre os cinco objetivos estratégicos que compõem o objetivo geral, o objetivo

estratégico número dois destaca a necessidade de fortalecer o conhecimento base da AMR através de vigilância e pesquisa, i.e., a disponibilização de informações sobre a incidência, prevalência, variação entre patógenos e padrões geográficos da AMR em tempo real. Assim como uma melhor compreensão sobre seu desenvolvimento, espalhamento e circulação no ambiente. A garantia desses conhecimentos pode ajudar na orientação do tratamento de pacientes e monitoramento da eficácia das intervenções, assim como no desenvolvimento de novas ferramentas, políticas e regulamentos para combater a AMR.

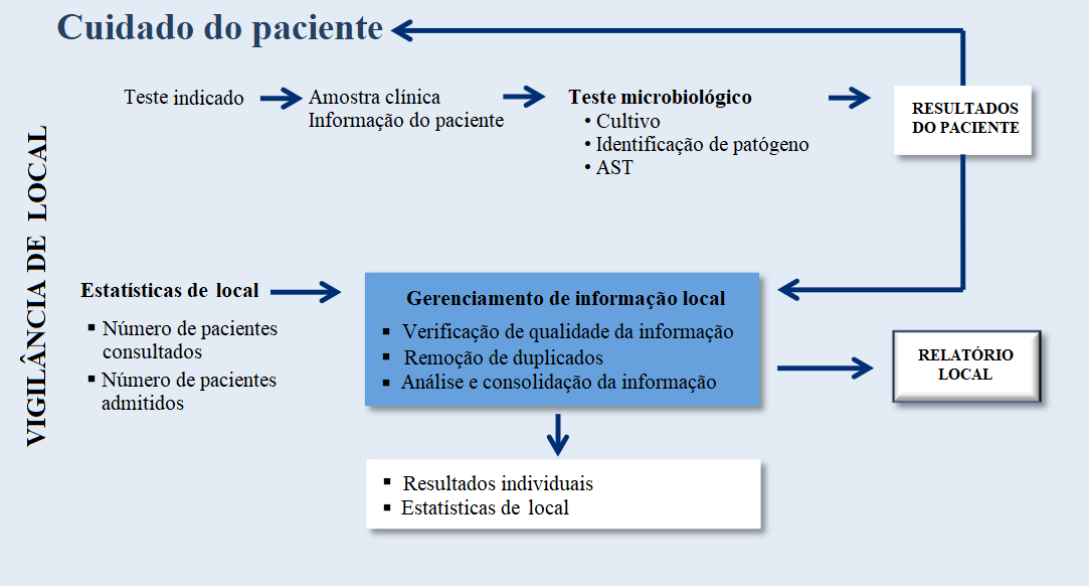
No ano anterior, a WHO lançou o Sistema Global de Vigilância do Uso e Resistência de Antimicrobianos (GLASS) (WHO, 2014), visando a padronização da vigilância da AMR. No seu manual para a implementação antecipada, GLASS fornece métodos e ferramentas para a coleta, análise, interpretação e compartilhamento da informação dos países membros (WHO, 2015b).

Os métodos estabelecem que a procura de casos parte de uma lista prioritária de bactérias e é feita a partir de testes AST aplicados em amostras clínicas de rotina (por exemplo, sangue, urina, fezes, esfregaços uretrais e cervicais) extraídas de pacientes selecionados nos locais de vigilância. Também, é coletada informação demográfica e epidemiológica dos pacientes e da população do local para identificar as suas características. Assim, os resultados dos testes AST são combinados com a informação do paciente e relacionados com a informação da população do local de vigilância. A informação mínima demográfica e epidemiológica que deve acompanhar a amostra inclui: idade, gênero, tipo de amostra, data de amostragem, e o local de amostragem. A informação logo é inserida num software de gestão de informação como WHONET, um software livre fornecido pela WHO para gerenciar, preparar e analisar a informação (AGARWAL; KAPILA; KUMAR, 2009).

WHONET permite a conversão de informação microbiológica para informação padronizada por meio da utilidade *BacLink* (WHO Collaborating Centre for Surveillance of Antimicrobial Resistance, 2026). Também permite exportar estatísticas de resistência e é integrado com SaTScan™(KULLDORFF, 2006), um software livre que analisa informação espacial, temporal e espaço-temporal usando estatística espacial (KULLDORFF; NAGARWALLA, 1995), temporal (NAGARWALLA, 1996) e espaço-temporal (KULLDORFF, 2001), modelos estatísticos baseados em probabilidade de *Poisson* e *Bernoulli*. A integração permite a vigilância geográfica de doenças, para detectar grupos de doenças espaciais ou espaço-temporais e para ver se são estatisticamente significativos; testar se uma doença está distribuída aleatoriamente no espaço, tempo ou espaço-tempo; e realizar vigilância prospectiva de doenças em tempo real ou de forma periódica para a detecção precoce de surtos de doenças.

Finalmente, a informação é compartilhada com o gerenciador nacional da AMR, que agrega as informações dos diferentes locais. Este fluxo é ilustrado na Figura 2.

Figura 2 – Esquema do fluxo de informação no nível local padronizado segundo GLASS



Fonte: Traduzido de WHO (2015b).

O controle efetivo de surtos requer uma compreensão detalhada de como a transmissão aconteceu (SNITKIN *et al.*, 2012). Por isso, a detecção de agrupamentos infecciosos é comumente investigada. Tradicionalmente a detecção é baseada em regras simples (por exemplo, a ocorrência de três novos casos de um patógeno na mesma sala durante duas semanas). Porém, esta abordagem baseada em regras depende de habilidade humana para detectar agrupamentos infecciosos a partir de informação coletada de amostras de pacientes, geralmente focada em poucos patógenos (HUANG *et al.*, 2010).

Vários estudos em vigilância da AMR se apoiam em ferramentas como WHONET para a detecção de agrupamentos de perfis infecciosos. Por exemplo, em Snitkin *et al.* (2012) a transmissão de um surto de *K. pneumoniae* é reconstruída por meio do sequenciamento do genoma a partir de amostras, e de informação epidemiológica como o rastreamento da localização de pacientes durante sua estadia no hospital.

Em Huang *et al.* (2010) foram investigados e classificados agrupamentos de bactérias em tempo e espaço usando WHONET-SaTScan™. Os agrupamentos foram depois comparados com agrupamentos encontrados pelo programa de controle de infecções. Foram encontrados 59 agrupamentos usando WHONET-SaTScan™, dos quais 95% foram classificados como de importância segundo a opinião de dois epidemiologistas do hospital. Apenas 3 agrupamentos tinham sido identificados pelo programa de controle de infecções.

Diferentemente dos últimos dois estudos, Ryu *et al.* (2019) realiza a vigilância de forma prospectiva, i.e., vigilância contínua de pacientes em admissão e durante hospitalização. Desta forma é determinada a prevalência de colonização da AMR em pacientes novos

hospitalizados e a aquisição de novas bactérias AMR durante a hospitalização. Análises logísticas multivariadas foram realizadas para identificar fatores de risco para colonização por bactérias MDR e aquisição de novas bactérias MDR (RATTANAUMPAWAN *et al.*, 2018).

Outros estudos de vigilância da AMR vão além, tentando prever a aparição de surtos apoiados em abordagens estatísticas baseadas em séries temporais (LÓPEZ-LOZANO *et al.*, 2019). Por exemplo, em Gharbi *et al.* (2015) é avaliada a confiabilidade e precisão de um modelo automatizado de vigilância da AMR para predizer surtos potenciais usando dados de consumo de antimicrobianos obtidos em farmácias. A previsão é realizada usando um modelo autorregressivo integrado de média móvel (ARIMA) (BOX *et al.*, 2015) para múltiplas séries temporais. Como resultado, o modelo apresenta uma diminuição na métrica raiz do erro quadrático médio (RMSE). Por sua vez, Silva *et al.* (2021) apresenta o uso da estatística SATSCAN aplicada a um sistema de georreferenciamento por meio de AutoCAD e software QGIS, onde foram mapeadas as ocorrências, permitindo a visualização espacial e temporal, e formação de *clusters* em mapas de calor, de pacientes com presença de K. pneumoniae multissensível (MS), MDR, XDR e PDR em áreas críticas e não críticas de um hospital terciário, federal, do Rio de Janeiro, durante três anos consecutivos (2014 a 2016). Os autores sugerem que o sistema de vigilância pode ser usado como um sistema de aviso prévio para predizer a emergência de organismos multirresistentes (MDROs).

Porém, em Pei *et al.* (2023) são apresentadas quatro dificuldades para a aplicação de modelos preditivos na AMR. Primeiro, a falta de compreensão científica sobre certos fatores que afetam a propagação da AMR. Temas como o impacto de diferentes drogas antimicrobianas na emergência da AMR, a transmissão e espalhamento da AMR através dos locais físicos, e o papel que cumprem as redes de contato, ainda não são bem compreendidos.

A segunda dificuldade está relacionada com uma limitada disponibilidade de informação de alta qualidade, especialmente registros de longo prazo de perfis infecciosos da AMR em países de baixa e média renda, e a necessidade de coletar informação de processos não biológicos relacionados com a transmissão de MDROs e o comportamento do paciente.

Em terceiro lugar, dado que a estrutura de modelos explicativos da AMR é tão complexa, isto dificulta sua calibração, dependendo da estrutura do modelo específico e os tipos de observação disponíveis, os quais podem variar em escalas diferentes. A quantificação da incerteza é também um desafio, especialmente para modelos que rastreiam pessoas e seus contatos de forma individual. Uma última dificuldade vem da implementação e avaliação prospectiva de modelos preditivos da AMR.

Não existem guias sobre como devem ser implementados estes modelos, nem da coleta de informação apropriada ou objetivos de previsão. Modelos padronizados de informação, objetivos e horizontes de tempo de previsão, e regras para a avaliação do

desempenho da previsão poderiam trazer avanços na previsão da AMR.

Apesar das dificuldades, Pei *et al.* (2023) encorajam o desenvolvimento da pesquisa através dos dados e recursos disponíveis, argumentando que estudos podem ser levados para testar a previsão em tempo real da AMR em nível de população usando informação de vigilância existente. Finalmente, mencionam que iniciativas como o aumento da disponibilidade da informação incluindo registros eletrônicos de saúde (EHRs) ao mesmo tempo em que é preservada a privacidade dos pacientes pode acelerar o progresso da previsão da AMR.

Por outra parte, pesquisas sobre vigilância da AMR usando IA vêm se desenvolvendo em anos recentes. Especificamente, técnicas de aprendizado de máquina têm sido utilizadas no estudo do problema da resistência bacteriana em três linhas principais (GIACOBBE *et al.*, 2020). A primeira, na identificação de pacientes com risco de adquirir infecções MDR-GNB, isto é, a identificação de fatores de risco nos quais intervir para reduzir o risco de colonização e prevenir a infecção. Por exemplo, Tacconelli *et al.* (2020) e Garcia-Vidal *et al.* (2021) combinam técnicas estatísticas clássicas e algoritmos de aprendizado de máquina, para obter informações associadas a fatores de risco.

A segunda, no reconhecimento da etiologia de MDR-GNB em pacientes infectados, que permita a identificação de intervenções (tratamentos) para melhorar o prognóstico de infecções (GOODMAN *et al.*, 2019; FERETZAKIS *et al.*, 2021). Vários modelos de aprendizado de máquina e aprendizado profundo (uma de suas técnicas principais) têm sido propostos na identificação e predição de resistência antimicrobiana, implementando dados de AST (XUE *et al.*, 2018; MACESIC *et al.*, 2020; LU *et al.*, 2021; FOOMANI *et al.*, 2022; Söylemez *et al.*, 2022).

E a terceira, evitando o aumento da resistência de MDROs a múltiplas drogas por meio da identificação de intervenções adequadas como medidas de controle de infecção e administração de antimicrobianos. Levando em conta que, em hospitais, o impacto das intervenções de administração de antibióticos pode ser algo complexo de avaliar, em Beaudoin *et al.* (2016) foi criado um sistema classificador baseado em regras para suporte clínico, capaz de aprender regras de classificação para a rotulagem de prescrições de antibióticos inapropriadas, baseando-se em um conjunto de dados de recomendações de administração passadas.

Finalmente, o projeto OUTBREAK fundado no ano 2018 pelo Fundo de Pesquisa Médica para o Futuro e liderado pela Universidade Tecnológica de Sydney, em parceria com outras organizações ao redor de Austrália, UK e Nova Zelândia vem trabalhando num sistema de vigilância que visa coletar e analisar grandes volumes de informação sobre AMR usando inteligência artificial para a exploração de diferentes estratégias de intervenção (Australian Government, 2022). O projeto é baseado na abordagem "*One Health*" proposta por GLASS e que visa a vigilância da AMR e sua relação ser humano - animais - meio

ambiente (ADISASMITO *et al.*, 2022).

Até onde foi consultado, não há registro de estudos ou modelos de redes neurais baseadas em grafos voltados à vigilância, monitoramento ou previsão da AMR.

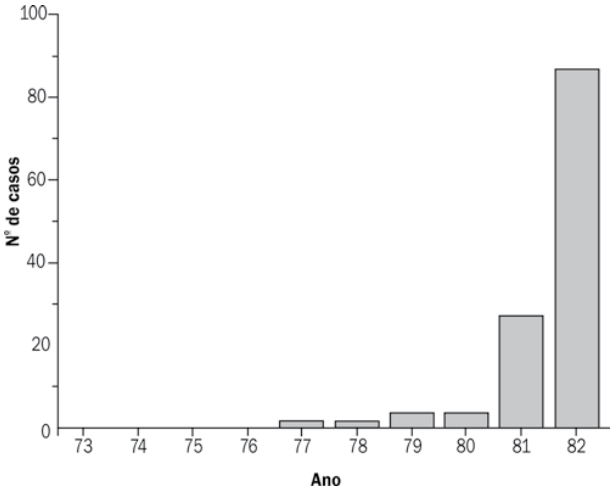
2.2 Epidemias

Dado que a vigilância e o monitoramento da AMR é um problema estreitamente ligado à análise de propagação de epidemias, é preciso entender o que é uma epidemia e quais são suas características para o desenvolvimento do trabalho.

Segundo Bonita *et al.* (2006), uma epidemia é definida como "a ocorrência em uma região ou comunidade de um número de casos em excesso, em relação ao que normalmente seria esperado."

Na identificação de uma epidemia devem ser considerados o período, região geográfica, entre outras particularidades da população em que os casos ocorreram. Essas informações também devem ser comparadas com a frequência usual da doença na região, no mesmo grupo populacional, durante a mesma estação do ano. Um pequeno número de casos de uma doença que não ocorreu previamente na região pode ser o suficiente para constituir a ocorrência de uma epidemia. Isto está ilustrado na Figura 3.

Figura 3 – Sarcoma de Kaposi em Nova York. Dois casos ocorreram em 1977 e 1978, enquanto em 1982 houve 88 casos



Fonte: Extraído de Bonita *et al.* (2006).

Especificamente, uma **epidemia por contágio** se dá quando uma doença é transmitida de uma pessoa para outra. A velocidade e extensão da transmissão depende principalmente de fatores como o número de indivíduos suscetíveis, as fontes potenciais de infecção, bem como do microrganismo e vias de transmissão. Outros fatores que contribuem

incluem a duração do período infeccioso, os comportamentos do hospedeiro (por exemplo, práticas de higiene) e as mudanças ecológicas (MORSE, 1995).

As **doenças transmissíveis endêmicas** são doenças que apresentam um padrão de ocorrência estável em uma área geográfica específica ou grupo populacional, com uma prevalência relativamente estável, previsível ou "esperada", podendo variar de níveis de ocorrência baixos a altos. A mudança deste tipo de doenças endêmicas para epidêmicas é comum diante de mudanças no hospedeiro, agente ou condições ambientais. (MORSE, 1995)

A **cadeia de infecção** de uma doença transmissível é dada pela interação de 4 elementos (BONITA *et al.*, 2006):

- O **Agente infeccioso** que é o microrganismo causante da infecção, cuja infecção se dá quando o agente infeccioso entra num hospedeiro e se desenvolve ou multiplica;
- O **Processo de transmissão** que consiste na transmissão ou difusão do agente infeccioso para o ambiente ou para outra pessoa e que pode ser feito de forma direta (por exemplo, mãos, beijo, relação sexual, contato com gotículas, secreções de tosse e espirro, sangue, ou transmissão vertical via placenta) ou indireta, i.e., por meio de veículos (materiais contaminados), vetores (insetos ou animais) ou aérea (por poeira ou pequenas partículas respiratórias especializadas - núcleos de gotículas- que permanecem suspensas no ar por longos períodos);
- O **Hospedeiro** ou o local natural, que proporciona as condições adequadas para que o agente infeccioso possa crescer e se multiplicar (pessoa ou animal). A reação do hospedeiro à infecção pode variar desde a infecção inaparente (sem sinais ou sintomas) até reações clínicas severas. O período desde a entrada do agente infeccioso até o aparecimento dos primeiros sintomas é denominado incubação. O grau de resistência de um hospedeiro pode dificultar o crescimento do agente infeccioso e pode ser aumentado naturalmente ou por meio de imunização (por exemplo, vacinação);
- E, finalmente, o **Ambiente**, que abrange as condições sanitárias, de temperatura, poluição, qualidade da água, fatores socioeconômicos (densidade populacional, aglomeração e pobreza) que cumprem um papel importante na cadeia de infecção.

O conhecimento e esclarecimento de cada um desses elementos permite o desenvolvimento e avaliação de medidas de controle. Por exemplo, melhoramentos no suprimento de água em Londres em 1850 preveniram novas epidemias de cólera décadas antes da identificação do agente responsável (BONITA *et al.*, 2006).

2.3 Modelos tradicionais de previsão de epidemias

Os modelos tradicionais de propagação de epidemias entram em uma de três categorias: modelagem de mecanismos de difusão, modelagem da curva de epidemia (séries temporais) e modelagem da rede de contatos (PASTOR-SATORRAS *et al.*, 2015).

Os modelos baseados em mecanismos de difusão ou modelos por **compartimentos** usam uma abordagem mecanística na qual são codificadas dinâmicas predefinidas de transmissão. Partem da suposição de que a população pode ser dividida em diferentes compartimentos ou classes, dependendo do estado da doença. Os estados são representados pelas siglas S para suscetíveis (indivíduos que podem ser infectados), I para infectados (indivíduos infectados) e R para recuperados (indivíduos recuperados da doença).

Para entender a evolução do contágio em função do tempo devem ser definidas transições para os indivíduos de um estado para outro. Desta forma, é possível definir variações na configuração do modelo. Por exemplo, o modelo suscetível-infectado-suscetível (SIS) (ANDERSON, 1991) com duas possíveis transições $S \to I$ e $I \to S$ assume que, após ser infectado, um indivíduo não desenvolve imunidade, podendo ser infectado de novo. Esta variação descreve bem o comportamento de uma doença de natureza endêmica.

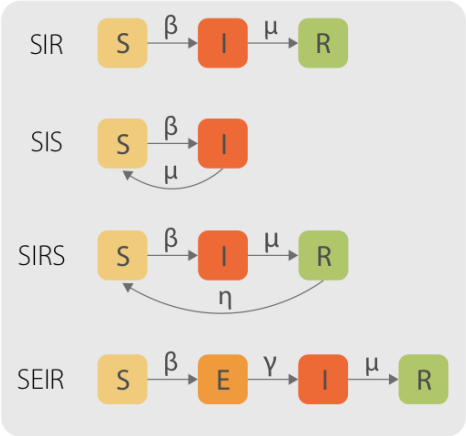
Outra configuração importante é a do modelo suscetível-infectado-recuperado (SIR) (ANDERSON, 1991) com duas possíveis transições $S \to I$ e $I \to R$, assumindo que o indivíduo se torna imune à doença ou morre, fazendo com que o processo sempre pare na ausência de indivíduos infectados. Outras configurações mais próximas a características biológicas das doenças são apresentadas na Figura 4. O modelo Suscetível-Infectado-Removido-Suscetível (SIRS) (ANDERSON, 1991) assume uma imunidade temporária, em que β é a taxa de infecção, μ a taxa de recuperação e η é a taxa na qual a imunidade do indivíduo é perdida. O modelo Suscetível-Exposto-Infectado-Removido (SEIR) (ARON; SCHWARTZ, 1984) introduz indivíduos que podem ser infectados, mas que não podem transmitir a doença num tempo determinado (γ).

Designando taxas de transição de infecção β e recuperação μ para cada instante é possível descrever modelos básicos de epidemias em termos de processos estocásticos de reação-difusão (KAMPEN, 1981).

Os modelos anteriores conseguem derivar equações determinísticas de processos epidêmicos básicos sob a suposição de que cada indivíduo é idêntico aos outros dentro de seu compartimento, obedecendo assim um tipo de comportamento homogêneo. Porém, o modelo ignora um aspecto social mediado pela rede de interações de cada indivíduo. A inclusão desta heterogeneidade é importante para a análise de propagação de epidemias, dado que na maioria das vezes epidemias apresentam padrões complexos de conectividade.

O modelo **Autorregressivo (AR)** é um dos modelos mais simples baseados em séries temporais. Como seu nome indica, faz uma autorregressão linear utilizando *pontos*

Figura 4 – Representações de modelos de epidemias em termos de processos reação-difusão.



Fonte: Extraído de Kampen (1981).

de dados passados (por exemplo, estados de pacientes) para modelar ou predizer estados futuros. Porém, devido à sua simplicidade pode não conseguir representar modificações inesperadas nas dinâmicas da epidemia (VIBOUD *et al.*, 2003) e apresentar limitações em tendências de longo prazo (DENG *et al.*, 2020).

Uma alternativa ainda mais simples dentro da família de abordagens estatísticas baseadas em séries temporais é a previsão por média histórica. Diferentemente dos modelos autorregressivos, que estimam coeficientes associados a observações passadas da própria série, a média histórica utiliza apenas a frequência ou o valor médio observado em períodos anteriores para definir uma predição constante. Esse tipo de linha de base tem sido utilizado em desafios e avaliações de previsão epidemiológica, como no FluSight/CDC, em que modelos de previsão de influenza são comparados com previsões baseadas na média histórica de temporadas anteriores (LUTZ *et al.*, 2019; REICH *et al.*, 2019).

Os modelos baseados em **Redes de Contato** moldam as relações entre indivíduos de uma população em uma rede, a partir da qual pode ser analisada a propagação de uma doença. A teoria de grafos fornece uma estrutura matemática para formalizar sua modelagem. Assim, nós podem representar indivíduos (ou populações), enquanto arestas as conexões entre eles (PASTOR-SATORRAS *et al.*, 2015).

A partir desta estrutura, modelos por compartimentos como SIS ou SIR podem ser incorporados para capturar dinâmicas de transmissão entre doenças (PASTOR-SATORRAS; VESPIGNANI, 2001; MORENO; PASTOR-SATORRAS; VESPIGNANI, 2002). Na modelagem, características dos grafos como conectividade, grau de distribuição e agrupamento, desempenham um papel importante para dar forma às complexas dinâmicas epidêmicas. Por outra parte, vários tipos de redes como redes aleatórias, redes de

mundo pequeno e redes livres de escala podem ser exploradas para entender a variação do espalhamento de doenças conforme a tipologia da rede (ERDOS; RENYI, 1959; WATTS; STROGATZ, 1998; BARABÁSI; ALBERT, 1999). Por exemplo, redes livres de escala com um pequeno número de nós de alto grau (*hubs*) podem permitir a rápida transmissão da doença.

Outros conceitos de interesse epidemiológico podem ser explorados. Por exemplo, o limiar epidêmico determina as condições sob as quais um pequeno surto pode transitar até virar uma epidemia. Esta é uma medida crítica para avaliar o potencial dos surtos (KERMACK; MCKENDRICK, 1927). O conceito pode ser representado a partir de processos de Markov para caracterizar as probabilidades de propagação da doença e estimar as velocidades de espalhamento (MIEGHEM; SAHNEH; SCOGLIO, 2014).

Por último, redes de contato também conseguem capturar a evolução temporal das conexões entre indivíduos, representando situações mais próximas da realidade. Esta é uma característica de análise desejável em termos epidemiológicos, dado que a estrutura de uma rede de contatos geralmente se vê modificada diante à transmissão de uma doença.

Neste trabalho, optou-se por usar a média histórica como linha de base por se tratar de um modelo estatístico simples, interpretável e sem dependência da estrutura espacial ou da janela recente de eventos. Assim, ela fornece uma referência mínima de desempenho baseada apenas na prevalência histórica de cada classe bacteriana em cada enfermaria.

3 Aspectos Teóricos

O detalhamento de conceitos teóricos é um processo básico para acompanhar o desenvolvimento do trabalho. Por isso, na Seção 3.1 serão abordados alguns conhecimentos elementares relacionados aos grafos tais como definições, representação e tipos. A Seção 3.2 apresenta a rede neural multicamada desde sua unidade básica o Perceptron. Tal conhecimento permitirá a exploração de outros tipos de redes necessárias para a compreensão das GNNs. Finalmente, a Seção 3.3 apresenta os principais métodos de representação de grafos para menores dimensões como vetores.

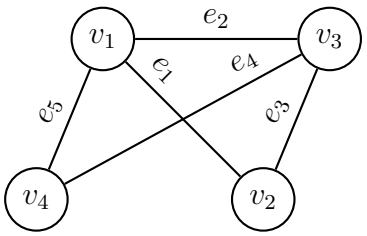
3.1 Teoria de grafos

A teoria de grafos é um ramo da matemática que lida com o estudo e a análise de estruturas chamadas grafos. Essas estruturas fornecem um marco sólido para a representação e estudo de problemas relacionados com vários cenários do mundo real, como redes sociais, redes de comunicação ou propagação de epidemias. Por isso, na continuação desta seção serão detalhados vários aspectos destas estruturas.

3.1.1 Definições, Propriedades e Medidas

Um **grafo** pode ser denotado por uma tupla $G = (V, E)$, onde $V = \{v_1, v_2, ..., v_N\}$ representa um conjunto de $N = |V|$ **nós** e $E = \{e_1, e_2, ..., e_M\}$ representa um conjunto de M arestas (relações) entre um par de nós (v_i, v_j) para qualquer v_i, v_j pertencente a V. O **tamanho** de um grafo G é definido pelo seu número de arestas. Um exemplo de grafo não dirigido, i.e., $e_i = (v_i, v_j) = (v_j, v_i)$, contendo 4 vértices e 5 arestas está ilustrado na Figura 5, onde $V = \{v_1, v_2, v_3, v_4\}$ e $E = \{e_1, e_2, e_3, e_4, e_5\}$.

Figura 5 – Exemplo de grafo não dirigido.

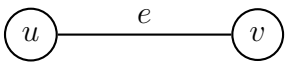


Fonte: Elaborado pelo autor.

Uma aresta $(u, v) \in E$ pode ser denotada pelos seus nós extremos $u \in V$ e $v \in V$, em que o nó u é **vizinho** de v, ou dito de outra forma u e v são **adjacentes**, conforme

ilustrado na Figura 6.

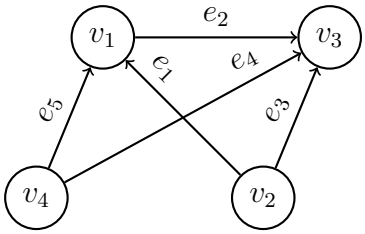
Figura 6 – Uma aresta e.



Fonte: Elaborado pelo autor.

Um grafo é **dirigido** quando suas arestas são direcionadas, por exemplo, uma aresta partindo de um vértice u para um vértice v, indicando um único sentido, conforme ilustrado na Figura 7.

Figura 7 – Grafo dirigido.



Fonte: Elaborado pelo autor.

O **grau de um nó** representa o número de vizinhos de um nó v_i ou o número de nós adjacentes ao nó v_i. Na Figura 5, $d(v_1) = 3$, já que o nó v_1 tem como vizinhos v_2, v_3 e v_4.

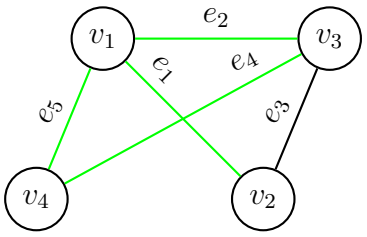
Um **passeio** é uma sequência finita alternada de nós e arestas, que inicia com um nó e termina com um nó, e na qual cada vértice é incidente com os nós imediatamente anteriores e posteriores. Se o passeio começa no nó u e termina no nó v, se trata de um passeio aberto. Se pelo contrário, o passeio começa no nó u e termina no nó u, este vai ser um passeio fechado. O **comprimento** de um passeio é dado pelo número de arestas do passeio. Para um grafo podem existir vários passeios com diferentes comprimentos. Um exemplo de passeio está ilustrado na Figura 8.

A **trilha** é um passeio cujas arestas são distintas, como pode ser visto na Figura 9.

Por sua vez, um **caminho** é um passeio cujos nós são distintos, como está ilustrado na Figura 10.

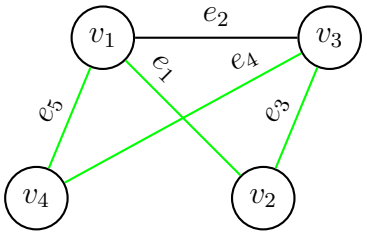
Um subgrafo $G' = (V', E')$ de $G = (V, E)$ é um grafo formado por um subconjunto de nós $V' \subseteq V$ e um subconjunto de arestas $E' \subseteq E$. O subconjunto V' deve conter todos os nós envolvidos nas arestas no subconjunto E'. O conceito de subgrafo está ilustrado na Figura 11.

Figura 8 – $W = v_1, e_5, v_4, e_4, v_3, e_2, v_1, e_1, v_2, e_1, v_1$ é um passeio fechado desde v_1 até v_1 de comprimento 5.



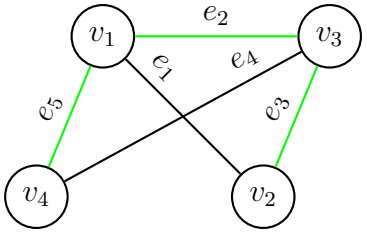
Fonte: Elaborado pelo autor.

Figura 9 – $W = v_1, e_1, v_2, e_3, v_3, e_4, v_4, e_5, v_1$ é uma trilha fechada desde v_1 até v_1 de comprimento 4.



Fonte: Elaborado pelo autor.

Figura 10 – $W = v_4, e_5, v_1, e_2, v_3, e_3, v_2$ é um caminho desde v_4 até v_2 de comprimento 3.



Fonte: Elaborado pelo autor.

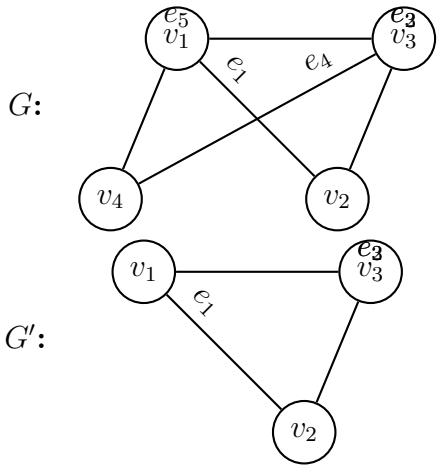
3.1.2 Representações

Uma das formas mais comuns de representar um grafo $G = (V, E)$ é por meio de sua **matriz de adjacência**, que pode ser denotada como $A \in \{0, 1\}^{N \times N}$, em que:

$$A_{ij} = \begin{cases} 1, & \text{se } (v_i, v_j) \in E \text{ e } i \neq j, \\ 0, & \text{caso contrário.} \end{cases} \tag{3.1}$$

Por exemplo, a matriz de adjacência do grafo da Figura 5, que se trata de um grafo não dirigido, é a matriz simétrica:

Figura 11 – Exemplo de Subgrafo. A figura acima contém um exemplo de grafo original G. Na figura abaixo há um exemplo de subgrafo G' de G, em que temos um subconjunto de nós $V' = \{v_1, v_2, v_3\}$ e um subconjunto de arestas $E' = \{e_1, e_2, e_3\}$.



Fonte: Elaborado pelo autor.

$$\mathbf{A} = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}.$$

Uma forma menos comum de representar uma matriz é por meio de sua **matriz de incidência**. Dado um grafo $G = (V, E)$ com n nós e m arestas, a sua matriz de incidência é $M \in \mathbb{R}^{n \times m}$, onde para grafos não dirigidos:

$$M_{ij} = \begin{cases} 1, & \text{se } v_i \text{ é incidente com } e_j, \\ 0, & \text{caso contrário.} \end{cases} \tag{3.2}$$

Já para grafos dirigidos temos:

$$M_{ij} = \begin{cases} -1, & \text{se } e_j \text{ sai de } v_i, \\ 1, & \text{se } e_j \text{ entra em } v_i, \\ 0, & \text{caso contrário.} \end{cases} \tag{3.3}$$

A seguinte matriz representa a matriz de incidência do grafo da Figura 5, onde linhas representam nós e colunas representam arestas:

$$\mathbf{M} = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}.$$

Dado um grafo $G = (V, E)$ com n nós, os graus dos vértices podem ser representados por uma matriz diagonal $D \in \mathbb{R}^{n \times n}$, em que:

$$D_{ii} = d(v_i). \tag{3.4}$$

Esta matriz é conhecida como a **matriz de grau**. Os graus dos vértices de um grafo poderiam ser representados de forma mais simples por meio de um vetor. Porém, é importante manter sua representação matricial devido a possibilitar sua adição e subtração com outras matrizes do mesmo tamanho. A seguinte matriz representa a matriz de grau do grafo da Figura 5:

$$\mathbf{D} = \begin{pmatrix} 3 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 2 \end{pmatrix}.$$

A **Matriz Laplaciana** ou matriz de diferença, é análoga ao operador Laplaciano em funções contínuas multivariadas. Dado um grafo $G = (V, E)$ com n nós, se todas as arestas são consideradas não dirigidas, a sua matriz Laplaciana $L \in \mathbb{R}^{n \times n}$, pode ser definida de duas formas:

$$\mathbf{L} = \mathbf{D} - \mathbf{A} \tag{3.5}$$

e,

$$\mathbf{L} = \mathbf{M}\mathbf{M}^{\top}. \tag{3.6}$$

Na primeira representação, D é a matriz de grau definida pela Equação 3.4, e A é a matriz de adjacências definida na Equação 3.1. A segunda representação é construída a partir da matriz de incidência M, definida na Equação 3.2. Assim, se tem que, com base em (NEWMAN, 2018; HAMILTON, 2020):

$$L_{ij} = \begin{cases} d(v_i), & \text{se } i = j, \\ -1, & \text{se } (v_i, v_j) \in E \text{ e } i \neq j, \\ 0, & \text{caso contrário.} \end{cases} \tag{3.7}$$

A seguinte matriz representa a matriz Laplaciana do grafo da Figura 5:

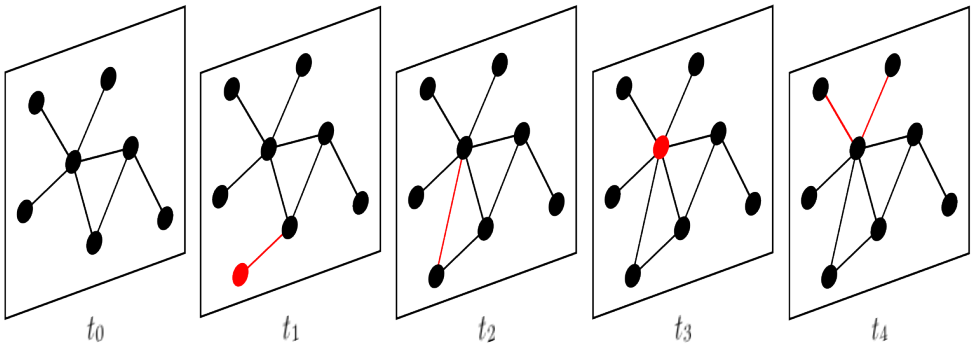
$$\mathbf{L} = \begin{pmatrix} 3 & -1 & -1 & -1 \\ -1 & 2 & -1 & 0 \\ -1 & -1 & 3 & -1 \\ -1 & 0 & -1 & 2 \end{pmatrix}.$$

A matriz Laplaciana apresenta três propriedades úteis. É uma matriz simétrica ($L^\top = L$), é positiva semi-definida ($x^\top Lx \geq 0, \forall x \in \mathbb{R}^{|V|}$) e, consequentemente, tem $|V|$ autovalores não negativos ($0 = \lambda_{|V|} \leq \lambda_{|V|-1} \ldots \leq \lambda_1$). Essas propriedades fornecem várias informações sobre a estrutura de um grafo, como o grau de conectividade, i.e, os autovalores pequenos obtidos a partir da matriz laplaciana guardam relação com o número de componentes conectados no grafo e, o maior autovalor, conhecido como o autovalor principal, guarda relação com a conectividade global e robustez do grafo (NEWMAN, 2018).

3.1.3 Grafos Dinâmicos Discretos

Evoluir ao longo do tempo é uma característica desejável quando se quer representar situações dinâmicas do mundo real. A evolução de um grafo pode ser representada como a série dos seus diferentes instantes de tempo, onde para cada observação no tempo t é registrada a observação de um grafo G_t.

Figura 12 – Exemplo de Grafo Dinâmico Discreto. Nos grafos observados nos tempos t_1-t_2 são adicionados um nó e duas arestas. Enquanto que nos tempos t_3-t_4, nós e arestas são atualizados.



Fonte: Elaborado pelo autor.

Formalmente, um *grafo dinâmico discreto* (MA; TANG, 2021) consiste nos T instantes do grafo observados ao longo da sua evolução. Os T instantes de um grafo podem ser denotados como $\{G_0, G_1, ..., G_T\}$ onde G_0 é o grafo observado no tempo 0. Um exemplo deste tipo de grafo pode ser encontrado na Figura 12.

3.2 Redes neurais artificiais

As redes neurais artificiais são técnicas de **aprendizado profundo** que assemelham o comportamento das redes neurais biológicas no sentido em que, a partir de uma informação, são capazes de gerar aprendizados para resolver tarefas.

3.2.1 Tipos de Aprendizado

As redes neurais artificiais seguem quatro tipos de aprendizado: **Aprendizado supervisionado**, **Aprendizado não-supervisionado**, **Aprendizado semi-supervisionado** e **Aprendizado por reforço**. Cada uma dessas abordagens apresenta diversas metodologias de treinamento para redes neurais artificiais, e seu uso dependerá, em grande medida, da informação disponível e do problema a ser resolvido.

- ***Aprendizado supervisionado*:**

O conhecimento do ambiente externo (até agora desconhecido pela rede neural) é representado em termos de um conjunto de exemplos de entrada-saída, a partir dos quais a rede neural é ensinada ou treinada, i.e., um conjunto de entrada de p características $X_1, X_2, \ldots X_p$ medidas em n observações e uma resposta ou saída Y para cada uma das n observações, o objetivo então é predizer Y usando $X_1, X_2, \ldots X_p$ num processo iterativo.

Durante este processo, a saída obtida pela rede difere da resposta desejada gerando um sinal de erro. É sob a influência deste erro e do vetor de características que os parâmetros da rede são ajustados. Depois de várias iterações os pesos são fixados e como resultado a informação do ambiente fica transferida e armazenada neles. Uma vez finalizado o processo de treinamento, espera-se que a rede seja capaz de interagir no ambiente externo de forma autônoma. Para medir o desempenho do sistema comumente são usadas as métricas **Erro Quadrático Médio (MSE)** e **Soma de Erros Quadráticos (SSE)** (HAYKIN, 2009).

- ***Aprendizado não-supervisionado*:**

Em aprendizado não supervisionado nenhuma fonte externa ou rotulo é fornecido, mas apenas o conjunto de entrada de p características $X_1, X_2, \ldots X_p$ medidas em n observações. Como consequência, a rede deve tentar aprender por si própria a distribuição de probabilidade do conjunto de características com a sorte de inferir conhecimento útil ou tendencias a partir dela.

O problema do aprendizado não supervisionado é fundamentalmente estimar uma densidade que provavelmente gerou as características. No entanto, também existem formas mais fracas de aprendizado não supervisionado, como estimativa de quantil, agrupamento, detecção de *outlier* e redução de dimensionalidade.

- ***Aprendizado semi-supervisionado*:**

Este tipo de aprendizado está entre o aprendizado supervisionado e não-supervisionado. O ambiente externo é representado por exemplos rotulados e não rotulados. Desta forma, o conjunto de informação pode ser dividido em duas partes: uma parte rotulada de l características $X_1, X_2, \ldots X_l$ medidas em n observações e uma resposta ou saída Y_l para cada uma das n observações e outra com apenas u características $X_{l+1}, X_2, \ldots X_{l+u}$ medidas em n observações.

Na pratica, o aprendizado semi-supervisionado funciona quando o conhecimento em $p(x)$ obtido a partir da informação não rotulada contém informação útil para a inferência de $p(y|x)$ (CHAPELLE; SCHOLKOPF; ZIEN, 2009).

- ***Aprendizado por reforço*:**

O aprendizado do mapeamento de entradas para saídas é realizado numa interação contínua com o ambiente externo e dirigida por um determinado objetivo, de forma tal que um índice ou recompensa seja maximizado. A rede ou agente não é fornecido com as ações que deve fazer, pelo contrário, ele deve descobrir quais ações produzem as melhores recompensas (SUTTON; BARTO, 2018).

O aprendizado por reforço é composto por quatro elementos. Uma **política de decisão** que define a forma em que o agente deve agir num determinado momento, i.e., um mapeamento desde estados percebidos do ambiente para ações a serem tomadas a partir desses estados. A política de decisão é estocástica, pois especifica probabilidades para cada ação.

Um **sinal de recompensa** a ser maximizado pelo agente em cada passo. Define seu objetivo e controla as políticas de decisão a partir de ações certas ou erradas, i.e., se uma determinada ação da política é seguida de uma baixa recompensa, a política trocará a ação por outra em eventos futuros.

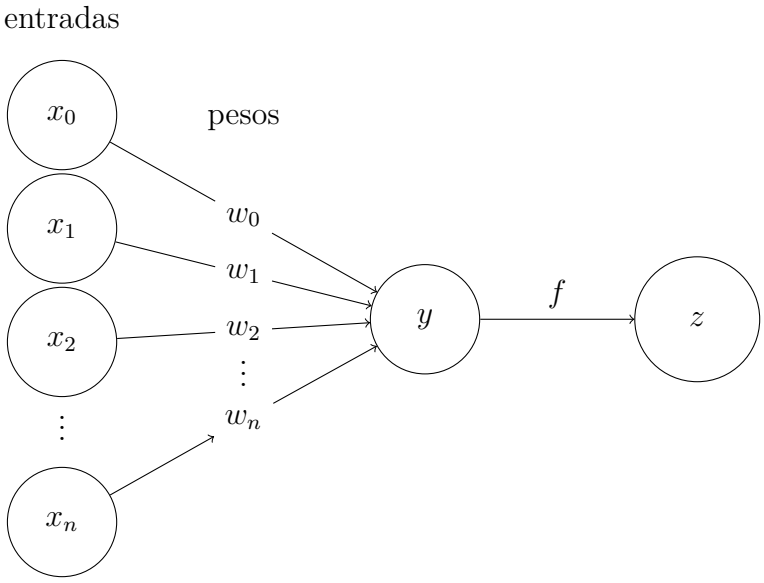
Uma **função de valor** que indica o valor total de recompensa que pode ser acumulado por um agente. A diferença do sinal de recompensa que indica as ações certas a serem realizadas no curto prazo, a função de valor indica as ações certas a longo prazo. Essas ações são de maior interesse já que obtêm a maior quantidade de recompensa ao logo do prazo.

E finalmente um **modelo do ambiente** que imite seu comportamento ou que permita fazer inferências sobre como o ambiente deve funcionar.

3.2.2 O Perceptron e a Rede Neural Multicamada (MLP)

A Figura 13 representa o **Perceptron**, a unidade básica que compõe uma rede neural clássica e cuja função principal é a de receber várias entradas e retornar uma saída. Especificamente, o Perceptron recebe n **entradas** $x_1, x_2, ..., x_n$, as quais são ponderadas por **pesos** $w_1, w_2, ..., w_n$, e um **viés** b. Com essas informações realiza uma soma ponderada $y = \sum_{i=1}^{n} w_i x_i + b$, cujo resultado passa depois por uma **função de ativação** f que mapeia o número real recebido para um número geralmente entre 0 e 1 (sendo 0 para desativada e 1 para totalmente ativada), retornando finalmente a *saída* $z = f(y)$. A conexão de múltiplos Perceptrons por meio de suas entradas e saídas define uma rede neural multicamada (MLP, do inglês *Multilayer Perceptron*), que estende esse princípio básico para arquiteturas com camadas sucessivas (HAYKIN, 1994).

Figura 13 – Perceptron.



Para gerar aprendizado procura-se otimizar os parâmetros (pesos e vieses) entre as unidades num processo iterativo, de tal forma que a rede consiga um desempenho desejado, minimize o erro do alvo até um valor aceitável ou execute um certo número de iterações (épocas). O **algoritmo de retropropagação** (WERBOS, 1974) baseado em **gradiente descendente** é comumente usado para otimizar os parâmetros no modelo.

O algoritmo se desenvolve em dois passos:

- ***Operações de avanço*:** dado um conjunto de parâmetros, a rede calcula os valores para cada unidade em sentido de inferência, i.e., das entradas para as saídas ;

- ***Retropropagação*:** computa o erro para cada parâmetro a ser otimizado e atualiza os parâmetros com suas derivadas parciais correspondentes (contribuição para o erro) numa ordem das saídas para as entradas. Por exemplo, se tomarmos um Perceptron e supormos que o alvo de otimização para a saída z é z_0, que pode ser aproximado otimizando os

parâmetros $w_1, w_2, ..., w_n, b$. Fazendo uso da regra da cadeia, a derivada de z com respeito a w_i e b pode ser deduzida (WERBOS, 1974):

$$\frac{\partial z}{\partial w_i} = \frac{\partial z}{\partial y}\frac{\partial y}{\partial w_i} = \frac{\partial f(y)}{\partial y}x_i, \tag{3.8}$$

$$\frac{\partial z}{\partial b} = \frac{\partial z}{\partial y}\frac{\partial y}{\partial b} = \frac{\partial f(y)}{\partial y}. \tag{3.9}$$

Definindo uma taxa de aprendizado η, a atualização para cada parâmetro é (WERBOS, 1974):

$$\Delta w_i = \eta(z_0 - z)\frac{\partial z}{\partial w_i} = \eta(z_0 - z)x_i\frac{\partial f(y)}{\partial y}, \tag{3.10}$$

$$\Delta b = \eta(z_0 - z)\frac{\partial z}{\partial b} = \eta(z_0 - z)\frac{\partial f(y)}{\partial y}. \tag{3.11}$$

No final, os pesos ajustados entre as conexões representam o conhecimento aprendido pela rede. Mudanças na configuração da rede alteram a forma de aprendizado. Desta forma, diferentes tipos de redes tem sido configuradas visando melhorar a capacidade de generalização em cenários diversos.

Uma MLP é composta por uma camada de entrada que recebe os estímulos externos da rede ou *características*, várias camadas **ocultas**, termo usado para denotar que não possuem conexão direta com o exterior da rede e cujas entradas e saídas são outras camadas da rede, e uma camada de saída que fornece os resultados computados pelas camadas ocultas. Cada camada é conectada com a camada adjacente. A agregação de mais camadas pode extrair representações estatísticas de ordem superior. A Figura 14 ilustra uma MLP.

3.2.3 Rede Neural Convolucional (CNN)

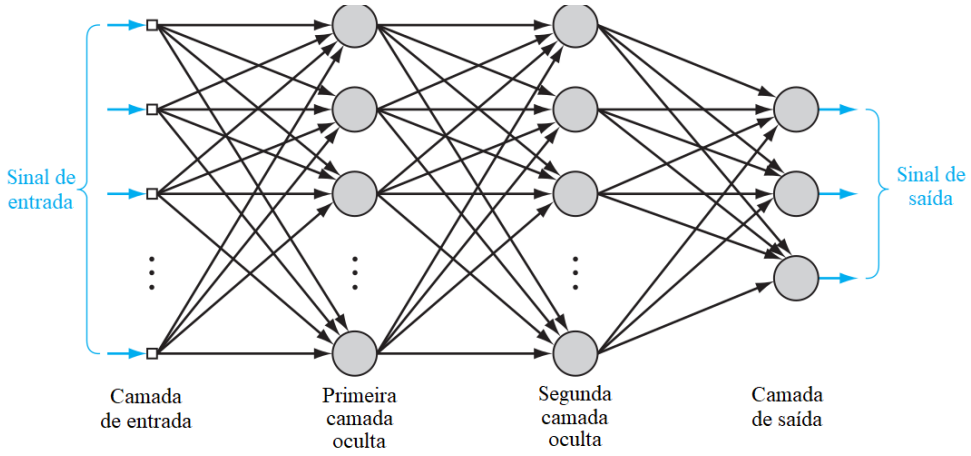
É um tipo especializado de rede para processar informação de topologias em forma de grade, tais como *séries temporais* 1D ou imagens (grade 2D de pixels) (GOODFELLOW; BENGIO; COURVILLE, 2016). Apresenta os mesmos elementos que uma MLP (neurônios, pesos treináveis e vieses), com a diferença de que introduz dois novos tipos de camadas:

- ***Camada de convolução*:** numa camada de convolução, os neurônios utilizam uma operação linear especializada chamada **convolução** (usualmente denotada com um asterisco), que é aplicada entre duas funções reais para produzir uma terceira função:

$$s(t) = (x * w)(t), \tag{3.12}$$

com base em (GOODFELLOW; BENGIO; COURVILLE, 2016).

Figura 14 – Exemplo de uma MLP. Arquitetura de uma rede neural multicamada com duas camadas ocultas, uma camada de entrada e uma camada de saída. Todas as camadas estão totalmente conectadas, o que quer dizer que cada unidade de uma camada está conectada com todas as unidades da camada anterior.



Fonte: Traduzido de Haykin (2009).

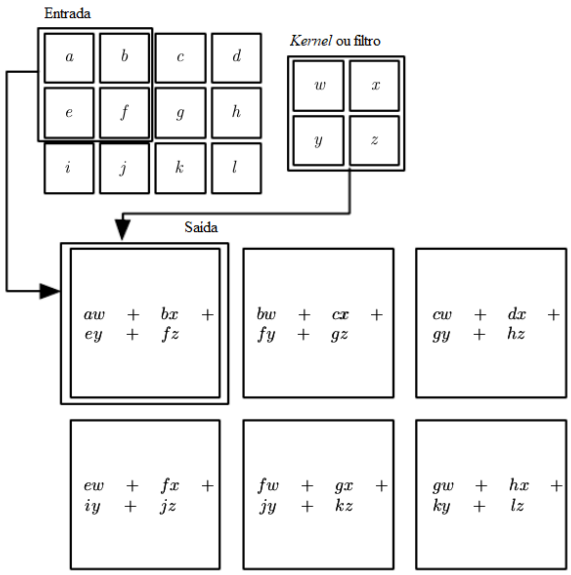
em que a função x faz referência à entrada, a função w atua como um ***kernel*** ou **filtro** e s corresponde à função resultante da operação. A variável t indica a posição em que a saída é calculada. A operação funciona como uma janela deslizante sobre a informação de entrada da camada, calculando a saída como uma média ponderada móvel entre os pesos definidos no *kernel* e os valores de entrada da camada. A operação de convolução está exemplificada na Figura 15.

A operação de convolução aplicada numa camada traz propriedades importantes que melhoram o desempenho de uma rede. A primeira delas tem a ver com as **conexões esparsas** da rede, isto quer dizer que, dado que o *kernel* tipicamente é de um tamanho menor do que a entrada, a quantidade de parâmetros extraídos resulta menor, representando uma melhora no desempenho computacional, já que reduz a quantidade de memória. Isto está ilustrado na Figura 16.

Numa rede neural tradicional, cada elemento na matriz de pesos é usado uma vez só quando é computada a saída de uma camada. Numa rede neural convolucional, cada membro do *kernel* é usado para cada posição da entrada, conforme pode ser visto na Figura 17. A propriedade de **compartilhamento de parâmetros** (GOODFELLOW; BENGIO; COURVILLE, 2016) implica o aprendizado de um único conjunto de parâmetros em vez de aprender um conjunto separado de parâmetros para cada localização.

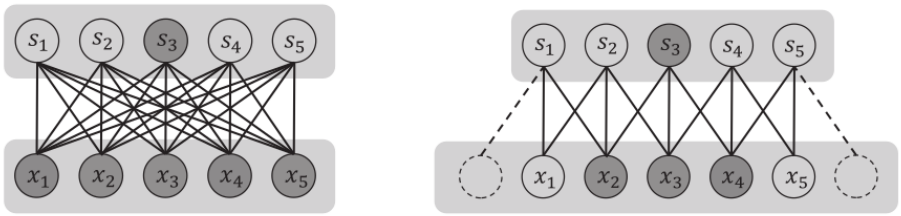
Outra propriedade importante introduzida pela operação de convolução faz com que as camadas convolucionais sejam **equivariantes** à translação. Isso significa que, se a entrada sofre uma translação, a saída sofre uma translação correspondente. Portanto,

Figura 15 – Exemplo de uma convolução 2D.



Fonte: Traduzido de Goodfellow, Bengio e Courville (2016).

Figura 16 – Exemplo da esparsidade de conexões em uma CNN. Na camada de rede neural tradicional, uma unidade $S3$ é afetada por todas as unidades da entrada, enquanto que na rede neural convolucional a mesma unidade é afetada apenas por três unidades. Essas unidades são chamadas de *campo receptivo* de $S3$.



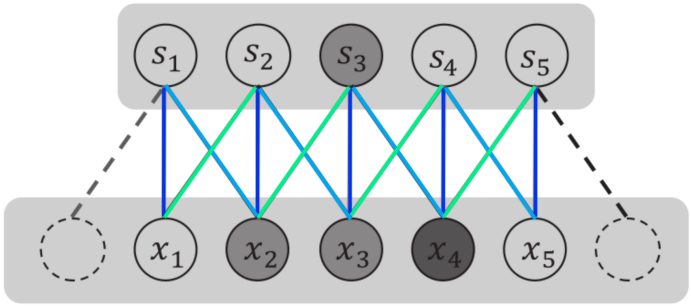
(a) Camadas em redes neurais tradicionais (b) Camadas em redes neurais convolucionais

Fonte: Traduzido de Goodfellow, Bengio e Courville (2016).

uma função $f(x)$ é equivariante a uma função g se $f(g(x)) = g(f(x))$ (GOODFELLOW; BENGIO; COURVILLE, 2016). Se realizarmos um deslocamento de uma unidade à direita das unidades de entrada na Figura 17, o padrão de saída também será deslocado uma unidade. Essa propriedade é útil em situações nas quais é de particular interesse identificar a existência de uma característica, mesmo quando sua localização varia, como na classificação de imagens. Porém, as camadas convolucionais convencionais não são, em geral, equivariantes a outros tipos de transformações, como mudanças de escala ou rotações.

- ***Camada de pooling*:** uma função de *pooling* substitui a saída de uma camada numa

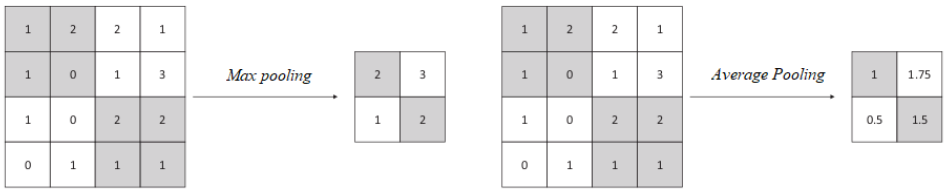
Figura 17 – Propriedade de compartilhamento de parâmetros da CNN. Dado que o *kernel* utilizado na imagem é de tamanho 3, cada um dos seus parâmetros é representado com uma linha de cor diferente. É observado que o conjunto de parâmetros se repete para cada unidade.



Fonte: Extraído de Goodfellow, Bengio e Courville (2016).

localização dada por uma sumarização estatística das saídas vizinhas. Como resultado, a altura e comprimento da informação são diminuídas, executando uma compressão de informação. Exemplos de operações de *pooling* estão na Figura 18.

Figura 18 – A operação de *max pooling* toma como saída o valor máximo na vizinhança local. A operação de *average pooling* toma como saída o valor meio da vizinhança local.

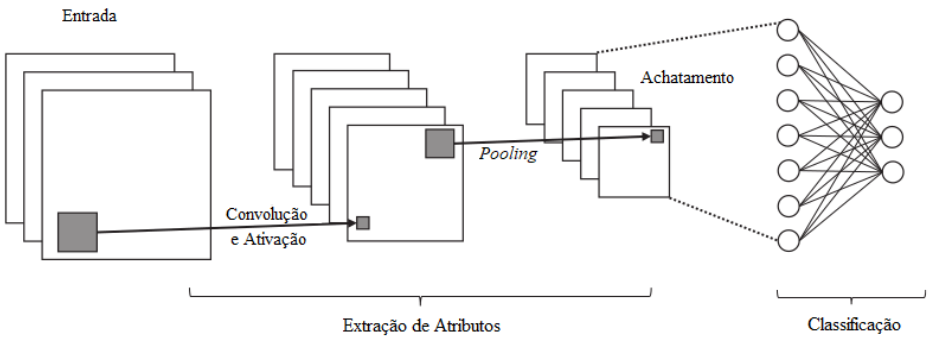


(a) *Max pooling* (b) *Average pooling*

Fonte: Extraído de Goodfellow, Bengio e Courville (2016).

A arquitetura geral de uma rede neural convolucional ilustrada na (Figura 19) apresenta dois componentes principais: o componente de extração e o componente de classificação. O componente de extração apresenta várias etapas. Na primeira etapa (**convolução**) são geradas várias convoluções em paralelo para produzir um conjunto de ativações lineares (MA; TANG, 2021). Na segunda etapa (**detecção**), cada ativação linear é levada para uma ativação não linear (por exemplo, usando funções ReLU (GOODFELLOW; BENGIO; COURVILLE, 2016) ou *tanh*). Finalmente, na terceira etapa (***pooling***) é usada uma função *pooling*. O componente de classificação é construído a partir de uma MLP. Geralmente, os dois componentes são conectados por uma operação de achatamento que reduz as matrizes de características extraídas pelo componente de extração para um simples vetor de características.

Figura 19 – Arquitetura geral de uma rede neural convolucional.



Fonte: Traduzido de Ma e Tang (2021).

3.2.4 Redes Neurais Recorrentes (RNNs)

São um tipo de rede capaz de processar informação de tipo sequencial $x^{(1)}, ..., x^{(t)}$ de modo que a entrada e a saída da rede são uma sequência de valores. Isto permite resolver várias tarefas como tradução de texto, classificação de um sentimento predominante a partir de um texto, reconhecimento de fala, entre outros.

O aprendizado de dados sequenciais apresenta dois problemas. O primeiro, devido a que os tamanhos das amostras sequenciais podem apresentar diferentes tamanhos, o tamanho de entrada e saída deve ser variável. O segundo, permitir o compartilhamento de parâmetros.

RNN soluciona os dois problemas aplicando uma mesma função de forma recorrente para cada elemento da sequência. Assim, a função pode ser aplicada recorrentemente tantas vezes como o tamanho da sequência o exija e o compartilhamento de parâmetros se torna uma característica inerente do processo.

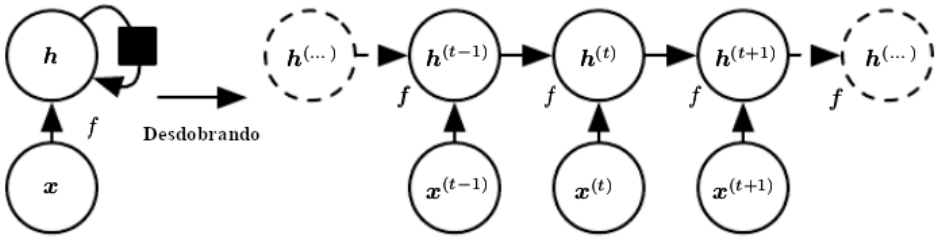
Qualquer função que envolva recorrência pode ser considerada como uma RNN. A seguinte equação representa esta ideia:

$$\mathbf{h}^t = f(\mathbf{h}^{t-1}, \mathbf{x}^t; \boldsymbol{\theta}), \tag{3.13}$$

em que t representa o instante ou a posição atual da sequência; $\mathbf{x}^t$ corresponde ao vetor de entrada recebido pela rede no instante t; $\mathbf{h}^{t-1}$ representa o estado oculto calculado no instante anterior; e $\mathbf{h}^t$ corresponde ao novo estado oculto produzido pela rede. A função f define a transformação recorrente aplicada às entradas e aos estados ocultos, enquanto $\boldsymbol{\theta}$ representa o conjunto de parâmetros aprendíveis da rede, como pesos e vieses, que são compartilhados ao longo dos diferentes instantes da sequência. A equação anterior é recorrente devido a que a definição de $\mathbf{h}$ no tempo t se refere à si mesma no tempo $t-1$. Quando a função é dirigida por um sinal externo $\mathbf{x}^t$, a unidade oculta $\mathbf{h}$ incorpora a

informação de toda a sequência anterior, como é ilustrado na Figura 20.

Figura 20 – Representação de uma rede recorrente em diagrama de circuito (esquerda), o quadrado preto indica um atraso para um instante de tempo, desde um estado no tempo $t+1$ para o estado no tempo t. A mesma rede reapresentada por meio de um grafo computacional, no qual cada nó é associado com uma instância de tempo (direita).



Fonte: Traduzido de Goodfellow, Bengio e Courville (2016).

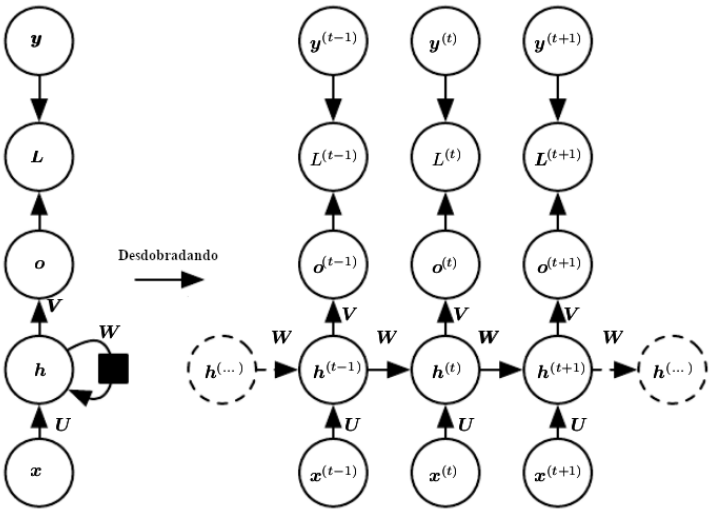
A arquitetura mais comum de RNN é apresentada na Figura 21. As operações de avanço começam com o estado inicial $\mathbf{h}^0$. Depois, para cada unidade de tempo desde $t = 1$ até $t = \tau$ são aplicadas as seguintes equações, com base em (GOODFELLOW; BENGIO; COURVILLE, 2016; GRAVES, 2012):

$$\mathbf{h}^{(t)} = \alpha(\mathbf{b} + \mathbf{W}\mathbf{h}^{(t-1)} + \mathbf{U}\mathbf{x}^{(t)}), \tag{3.14}$$

$$\hat{\boldsymbol{y}}^{(t)} = \text{softmax}(\mathbf{c} + \mathbf{V}\mathbf{h}^{(t)}). \tag{3.15}$$

em que $\mathbf{b}$ e $\mathbf{c}$ são os parâmetros treináveis, junto com as matrizes $\mathbf{W}$, $\mathbf{U}$ e $\mathbf{V}$. α é uma função de ativação não linear.

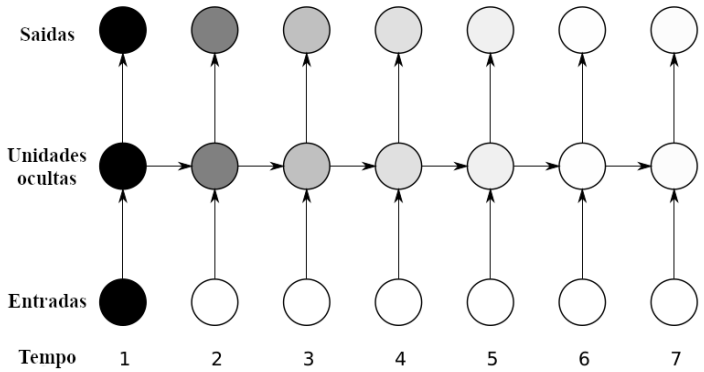
Figura 21 – A função de perda L mede o erro de **o** em relação ao alvo **y**. Todas as conexões da unidade oculta são parametrizadas. De **x** para **h** por meio da matriz de pesos **U**; de **h** para **h** por meio da matriz de pesos **W** e de **h** para a saída **o** por meio da matriz de pesos **V**.



Fonte: Traduzido de Goodfellow, Bengio e Courville (2016).

Existem aplicações nas quais é requerida a captura de **dependências de longo prazo** em informações sequenciais. Isto equivale a dizer que existe uma forte relação entre dois elementos de uma sequência que estão afastados um do outro. Trata-se de um problema para as RNNs, especificamente no aprendizado, já que quanto maior for o número de estados na sequência, mais tendem os gradientes a desaparecer ou explodir. Este problema está ilustrado na Figura 22, e é normalmente conhecido como **o problema do desaparecimento dos gradientes**.

Figura 22 – A sombra nas unidades da RNN desdobrada indica a sensibilidade às entradas. Quanto mais escura a sombra, maior o grau de sensibilidade. No tempo $t = 1$, a sensibilidade é total em todas as unidades. Porém, a sensibilidade apresenta um atraso ao longo da sequência à medida que novas entradas sobrescrevem as ativações das unidades ocultas.

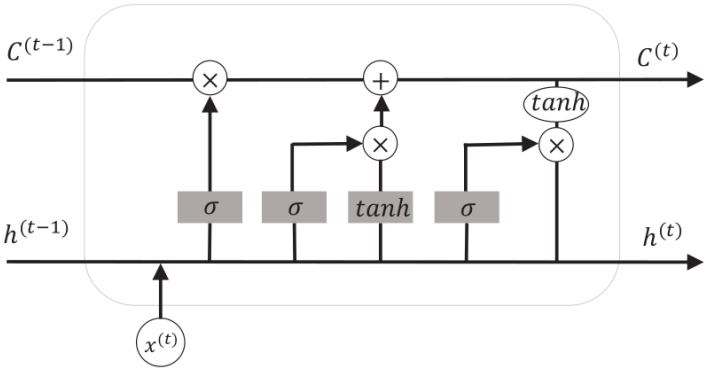


Fonte: Traduzido de Graves (2012).

Um dos modelos de RNNs mais efetivos e usados em aplicações práticas para ampliar a persistência dos dados ao longo da recorrência é o ***long short-term memory* (LSTM)** (HOCHREITER; SCHMIDHUBER, 1997).

A rede resolve o problema adicionando uma unidade de longo prazo ou **estado de célula** C paralelo à unidade de curto prazo. É como se fosse adicionada uma rota cujas derivações não se perdem nem explodem ao longo do tempo. Este estado de memória armazena informações chaves significativas para uma subsequência, que depois de serem usadas ou terminada a subsequência são esquecidas. A Figura 23 apresenta um bloco de LSTM.

Figura 23 – Ilustração de um bloco LSTM. O estado de célula **C** inclui a informação desde os estados prévios que são propagados até posição seguinte. A unidade oculta **h** decide como a informação deve ser propagada, podendo ser a saída, se for o caso.



Fonte: Extraído de Ma e Tang (2021).

LSTM controla o fluxo de informação que passa pela célula por meio de um sistema de três portas ou neurônios. A **porta *forget*** descrita na Equação 3.16 decide qual informação dos estados de células anteriores apagar:

$$\mathbf{f}^{(t)} = \sigma(\mathbf{W}^{\mathrm{f}}\mathbf{x}^{(t)} + \mathbf{U}^{\mathrm{f}}\mathbf{h}^{(t-1)} + \mathbf{b}^{\mathrm{f}}), \tag{3.16}$$

em que $\mathbf{f}^{\mathrm{t}}$ é o parâmetro junto com as matrizes de pesos $\mathbf{W}^{\mathrm{f}}$ e $\mathbf{U}^{\mathrm{f}}$ da porta *forget*, e σ é a função de ativação.

A **porta de entrada** descrita na Equação 3.17 é computada de forma similar à porta *forget*, mas com seus próprios parâmetros e matrizes de pesos, e decide que informação da nova entrada $\mathbf{x}^t$ deve ser armazenada no novo estado de célula:

$$\mathbf{i}^{(t)} = \sigma(\mathbf{W}^{\mathrm{i}}\mathbf{x}^{(t)} + \mathbf{U}^{\mathrm{i}}\mathbf{h}^{(t-1)} + \mathbf{b}^{\mathrm{i}}). \tag{3.17}$$

em que $\mathbf{i}^{(t)}$ representa o vetor de ativação da porta de entrada no instante t e $\mathbf{W}^{\mathrm{i}}$ e $\mathbf{U}^{\mathrm{i}}$ as respectivas matrizes de pesos da porta. Cada elemento desse vetor indica, por meio de um valor entre 0 e 1, a proporção de uma determinada informação candidata que poderá ser adicionada ao estado da célula. Valores próximos de 0 indicam que a informação deve ser pouco considerada, enquanto valores próximos de 1 indicam que ela deve ser amplamente incorporada.

A informação de entrada é processada para gerar valores candidatos $\tilde{\boldsymbol{C}}^{(t)}$ que logo são combinados com os valores do antigo estado de célula $\mathbf{C}^{(t-1)}$. Como resultado são obtidos valores atualizados para o novo estado de célula $\mathbf{C}^{(t)}$ combinando a memória anterior preservada e a nova informação candidata. O processo é descrito nas seguintes equações:

$$\tilde{\boldsymbol{C}}^{(t)} = \sigma(\mathbf{W}^{\mathrm{c}}\mathbf{x}^{(t)} + \mathbf{U}^{\mathrm{c}}\mathbf{h}^{(t-1)} + \mathbf{b}^{\mathrm{c}}), \tag{3.18}$$

$$\mathbf{C}^{(t)} = \mathbf{f}^{(t)} \odot \mathbf{C}^{(t-1)} + \mathbf{i}^{(t)} \odot \tilde{\boldsymbol{C}}^{(t)}, \tag{3.19}$$

em que $\mathbf{f}^{(t)}$ representa o vetor de ativação da porta de esquecimento no instante t e $\mathbf{W}^{\mathrm{c}}$ e $\mathbf{U}^{\mathrm{c}}$ as respectivas matrizes de pesos da porta. A notação $\odot$ representa o produto de Hadamard, isto é, a multiplicação realizada elemento a elemento entre vetores de mesma dimensão.

Finalmente, a **porta de saída**, formulada da mesma forma que as portas de entrada e *forget* e descrita na Equação 3.20 que determina quais partes do estado da célula preservar:

$$\mathbf{o}^{(t)} = \sigma(\mathbf{W}^{\mathrm{o}}\mathbf{x}^{(t)} + \mathbf{U}^{\mathrm{o}}\mathbf{h}^{(t-1)} + \mathbf{b}^{\mathrm{o}}), \tag{3.20}$$

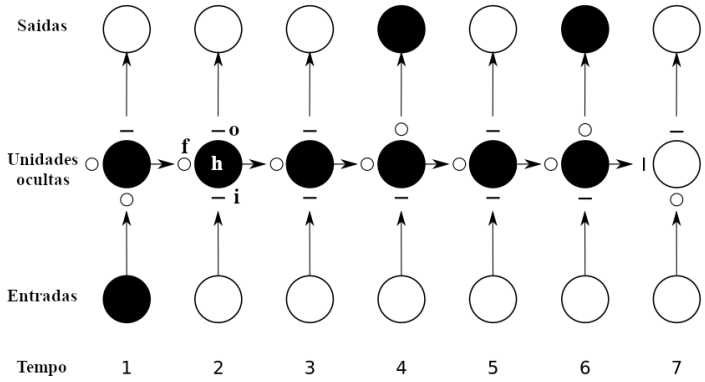
em que $\mathbf{W}^{\mathrm{o}}$ e $\mathbf{U}^{\mathrm{o}}$ são as matrizes de pesos da porta de saída. Isto com a finalidade de gerar o estado oculto $\mathbf{h}^{t}$, que poderá passar para a posição seguinte e funcionar como saída. A Equação 3.21, baseada em (HOCHREITER; SCHMIDHUBER, 1997; GRAVES, 2012), descreve esse processo:

$$\mathbf{h}^{(t)} = \mathbf{o}^{(t)} \odot tanh(\mathbf{C}^{(t)}). \tag{3.21}$$

Por meio deste modelo é possível preservar a informação do gradiente, como é ilustrado na Figura 24.

Uma alternativa mais compacta à LSTM é a *Gated Recurrent Unit* (GRU) (CHO *et al.*, 2014). Assim como a LSTM, a GRU utiliza mecanismos de portas para regular o fluxo de informação ao longo da sequência e reduzir o problema do desaparecimento do

Figura 24 – A sombra nas unidades da RNN desdobrada indica a sensibilidade às entradas. Quanto mais escura a sombra, maior o grau de sensibilidade. LSTM controla o fluxo de informação e sua persistência por meio do sistema de portas.



Fonte: Traduzido de Graves (2012).

gradiente. No entanto, enquanto a LSTM mantém dois estados distintos, o estado oculto $\mathbf{h}^{(t)}$ e o estado de célula $\mathbf{C}^{(t)}$, a GRU mantém apenas o estado oculto $\mathbf{h}^{(t)}$. Essa diferença torna a GRU mais simples e reduz o número de parâmetros treináveis.

Na LSTM, a porta de esquecimento $\mathbf{f}^{(t)}$ controla quanto do estado de célula anterior $\mathbf{C}^{(t-1)}$ deve ser preservado, enquanto a porta de entrada $\mathbf{i}^{(t)}$ controla quanto da nova informação candidata $\tilde{\mathbf{C}}^{(t)}$ será incorporada ao novo estado de célula. A GRU combina essas duas funções em uma única porta de atualização $\mathbf{z}^{(t)}$, responsável por controlar o equilíbrio entre o estado oculto anterior e o novo estado candidato.

A porta de atualização é definida por:

$$\mathbf{z}^{(t)} = \sigma(\mathbf{W}^z\mathbf{x}^{(t)} + \mathbf{U}^z\mathbf{h}^{(t-1)} + \mathbf{b}^z). \tag{3.22}$$

Valores altos em $\mathbf{z}^{(t)}$ favorecem a incorporação da nova informação candidata, enquanto valores baixos preservam maior parte do estado oculto anterior. A segunda porta da GRU é a porta de *reset*, definida por:

$$\mathbf{r}^{(t)} = \sigma(\mathbf{W}^r\mathbf{x}^{(t)} + \mathbf{U}^r\mathbf{h}^{(t-1)} + \mathbf{b}^r). \tag{3.23}$$

Essa porta controla quanto da informação passada será considerada no cálculo do novo estado candidato. Quando $\mathbf{r}^{(t)}$ assume valores baixos, parte do estado anterior é ignorada, permitindo que a unidade recorrente reduza a influência de informações passadas menos relevantes. O estado candidato é então calculado por:

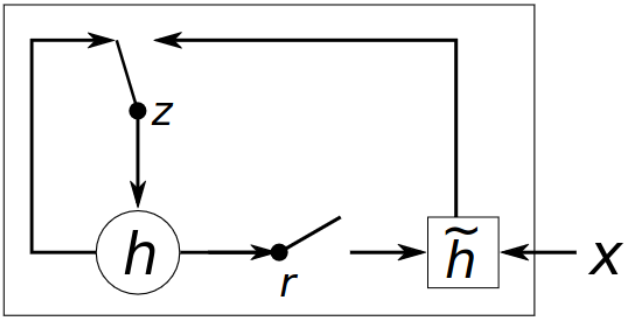
$$\tilde{\mathbf{h}}^{(t)} = \tanh(\mathbf{W}^h\mathbf{x}^{(t)} + \mathbf{U}^h(\mathbf{r}^{(t)} \odot \mathbf{h}^{(t-1)}) + \mathbf{b}^h). \tag{3.24}$$

Finalmente, o novo estado oculto é obtido por uma interpolação entre o estado anterior $\mathbf{r}^{(t)}$ e o estado candidato $\mathbf{h}^{(t-1)}$, com base em (CHO *et al.*, 2014; GRAVES, 2012):

$$\mathbf{h}^{(t)} = (1 - \mathbf{z}^{(t)}) \odot \mathbf{h}^{(t-1)} + \mathbf{z}^{(t)} \odot \tilde{\mathbf{h}}^{(t)}. \tag{3.25}$$

Assim, a GRU pode ser entendida como uma versão mais enxuta da LSTM: ela preserva o mecanismo de controle temporal por portas, mas remove o estado de célula explícito e reduz o número de portas minimizando a complexidade paramétrica. A Figura 25 apresenta o mecanismo de portas da unidade oculta GRU.

Figura 25 – A porta de atualização $\mathbf{z}$ decide se o estado oculto é atualizado por um novo estado $\tilde{\mathbf{h}}$ e a porta de *reset* $\mathbf{r}$ decide se o estado oculto anterior é ignorado.



Fonte: Traduzido de Graves (2012).

3.2.5 Modelo *Transformer* e mecanismos de atenção

Apesar do bom desempenho das redes baseadas em recorrência no processamento de informações sequenciais e dos esforços feitos para melhorar seu desempenho além dos seus próprios limites, as redes neurais baseadas em recorrência ainda apresentam limitações importantes. Primeiro, dado que a informação sequencial é codificada e atualizada num mesmo estado, dependências entre posições distantes em sequências longas podem não ser codificadas, levando para um gargalo em que informação contextual é perdida. Segundo, dado que as posições na sequência estão alinhadas com os tempos, i.e., o processamento é feito passo a passo de forma sequencial, isto impede a paralelização dos cômputos resultando numa perda de eficiência para sequências de maior tamanho (VASWANI *et al.*, 2017).

Em resposta a esses problemas, vários modelos têm implementado mecanismos de atenção, dotando as redes neurais com a capacidade para ponderar ou enfatizar em partes específicas da informação de entrada, permitindo assim o aprendizado de dependências de uma sequência sem importar sua distância (BAHDANAU; CHO; BENGIO, 2014; KIM *et al.*, 2017). Vários mecanismos de atenção baseiam-se em conceitos de bases de dados, i.e.,

uma consulta (q) pode ser realizada sobre um conjunto de tuplas chave-valor (k, v). Desta forma, se tomarmos uma sequência de entrada $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$, onde cada elemento representa um vetor de características e aplicarmos matrizes de pesos $\mathbf{W}_q, \mathbf{W}_k, \mathbf{W}_v$ na sequência de entrada, obteremos representações transformadas de consulta ($\mathbf{Q}$), chave ($\mathbf{K}$) e valor ($\mathbf{V}$):

$$\begin{aligned} \mathbf{Q} &= \mathbf{X} \cdot \mathbf{W}_q \\ \mathbf{K} &= \mathbf{X} \cdot \mathbf{W}_k \\ \mathbf{V} &= \mathbf{X} \cdot \mathbf{W}_v, \end{aligned} \tag{3.26}$$

em que $\mathbf{Q} \in \mathbb{R}^{N \times d_k}$, $\mathbf{K} \in \mathbb{R}^{M \times d_k}$ e $\mathbf{V} \in \mathbb{R}^{M \times d_v}$. N e M denotam o comprimento das consultas e chaves (ou valores), respectivamente, e d_k e d_v denotam as dimensões das chaves (ou consultas) e valores, respectivamente.

Dado que o objetivo da atenção é o de medir a similaridade entre a consulta e cada chave na sequência de chaves, as pontuações de similaridade podem ser computadas usando o produto escalar, que mede o quão similares são os vetores de chave com a consulta. A partir destas similaridades é aplicada uma função *softmax* para garantir que os pesos somem 1 e representem uma distribuição de probabilidade válida ao longo das chaves. Como resultado é obtida a matriz de atenção A, que aplicada a V retorna os valores mais relevantes para a consulta:

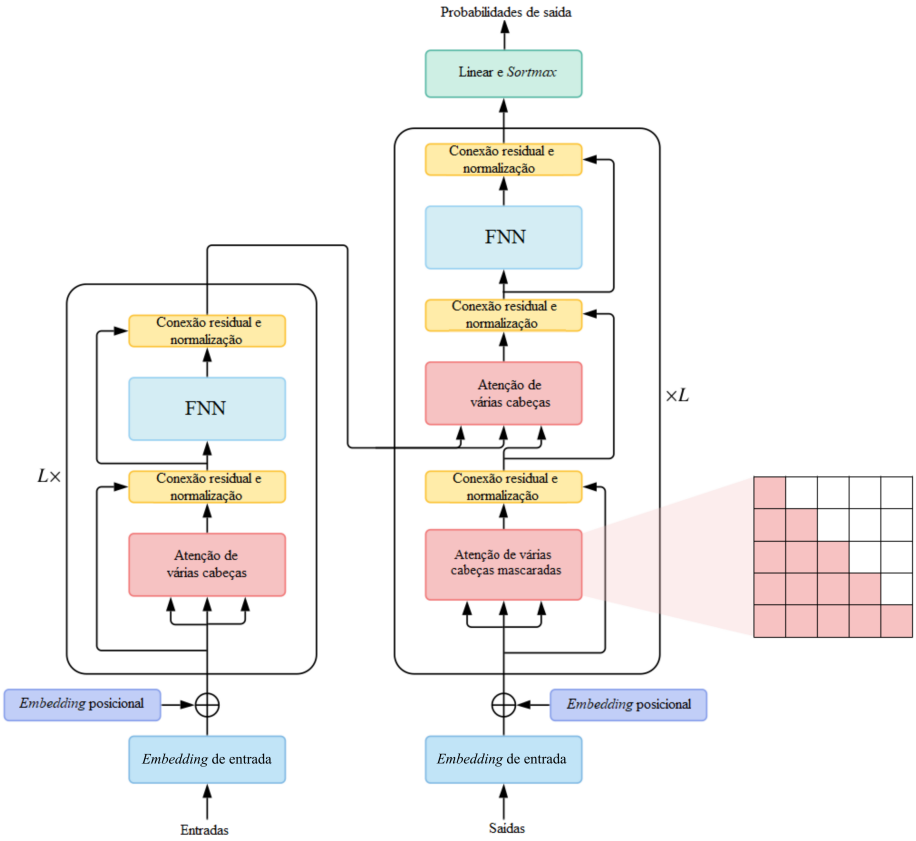
$$\mathbf{A} = \text{softmax}\left(\mathbf{Q} \cdot \mathbf{K}^\top\right). \tag{3.27}$$

Em resumo, o mecanismo de atenção pode ser descrito como uma função que mapeia uma consulta e um conjunto de chaves-valor para uma saída, onde a consulta, chaves, valores e saída são todos vetores. A saída é computada como uma soma ponderada dos valores, onde o peso atribuído a cada valor é computado por uma função de similaridade da consulta com a chave correspondente (VASWANI *et al.*, 2017).

Ainda que os mecanismos de atenção foram exitosamente incluídos em redes neurais recorrentes, apresentando melhor desempenho que arquiteturas *encoder-decoder sequence-to-sequence* para tarefas como tradução de máquina, ambos problemas continuavam presentes (ZHANG *et al.*, 2023). Em Vaswani *et al.* (2017) é proposto o ***Transformer***, um modelo para tradução de máquina que resolvia os problemas dos modelos baseados em recorrência por meio do uso de mecanismos de atenção do princípio ao fim. A Figura 26 apresenta o modelo *Transformer*.

O modelo apresenta uma estrutura *encoder-decoder* em que tanto o *encoder* como o *decoder* podem ser empilhados em L blocos idênticos. O ***encoder*** é composto de duas camadas. A primeira consiste num mecanismo de auto atenção de várias cabeças. A

Figura 26 – *Transformer* convencional



Fonte: Traduzido de Vaswani *et al.* (2017).

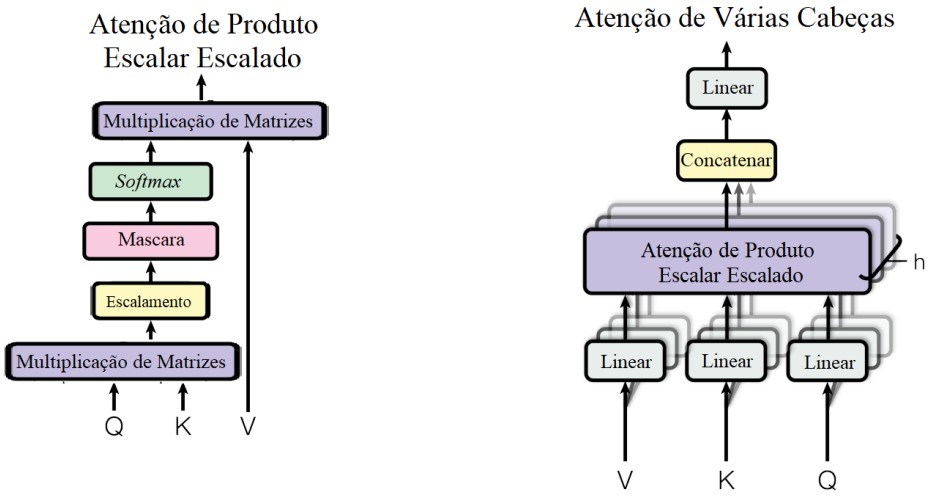
segunda é uma rede de avanço totalmente conectada a cada uma das posições da sequência. Conexões residuais são adicionadas ao redor de cada uma das camadas seguida de uma camada de normalização. Adicionalmente, para facilitar as conexões residuais, todas as camadas e os *embeddings* produzem saídas de dimensão d_{modelo}.

Em comparação com o *encoder*, no *decoder* é adicionada sobre a saída do *encoder* uma camada extra de auto atenção de várias cabeças. E de forma similar, conexões residuais seguidas de uma camada de normalização são agregadas. Adicionalmente, dado que o *decoder* apenas deve focar para posições prévias à saída imediatamente gerada, a primeira camada de auto atenção de várias cabeças é modificada para mascarar as posições subsequentes.

A seguir serão apresentados em maiores detalhes os elementos principais do modelo *Transformer* descrito anteriormente:

- ***Camadas de atenção*:** no modelo é proposto o mecanismo **atenção de produto escalar escalado** como mecanismo de atenção principal, que em comparação com o mecanismo de atenção da Equação 3.27 escala o produto entre Q e K por um fator de

Figura 27 – *Atenção de várias cabeças*



Fonte: Traduzido de Vaswani *et al.* (2017).

$\sqrt{d_k}$ para prevenir o desaparecimento de gradiente da função *softmax*:

$$\text{Atenção}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q} \cdot \mathbf{K}^\top}{\sqrt{d_k}}\right) \cdot \mathbf{V}. \tag{3.28}$$

As consultas, chaves e valores podem ser projetadas linearmente h vezes para dimensões d_k, d_k e d_v, respectivamente, a partir de diferentes projeções lineares aprendíveis. Sobre cada uma dessas versões projetadas é aplicada a função de atenção em paralelo. Uma vez projetadas são concatenadas, resultando em valores finais como ilustrado na Figura 27. Este processo é conhecido como **atenção de várias cabeças** e permite ao modelo focar de forma conjunta para informações desde diferentes subespaços de representação em posições diferentes:

$$\begin{aligned} \text{cabeça}_i &= \text{Atenção}\left(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V\right), \\ \text{Atenção de várias cabeças}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) &= \text{Concatenar}\left(\text{cabeça}_1, \ldots, \text{cabeça}_h\right)\mathbf{W}^O, \end{aligned} \tag{3.29}$$

em que as projeções são matrizes de parâmetros $\mathbf{W}_i^Q \in \mathbb{R}^{d_{modelo} \times d_k}$, $\mathbf{W}_i^K \in \mathbb{R}^{d_{modelo} \times d_k}$, $\mathbf{W}_i^V \in \mathbb{R}^{d_{modelo} \times d_v}$ e $\mathbf{W}^O \in \mathbb{R}^{hd_v \times d_{modelo}}$.

No *decoder*, o mecanismo de atenção deve ser restringido de forma que as consultas em cada posição possam focar unicamente em todas as chaves-valor anteriores a essa posição. Para isso, uma função de máscara é aplicada na matriz de atenção não normalizada $\hat{\mathbf{A}} = \exp\left(\frac{\mathbf{Q} \cdot \mathbf{K}^\top}{\sqrt{d_k}}\right)$, onde as posições restringidas são mascaradas de forma que $\hat{\mathbf{A}}_{ij} = -\infty \,\text{se}\, i < j$. Este processo é comumente conhecido como **atenção mascarada**, autorregressiva ou causal.

Para conectar a saída do *encoder* com o *decoder*, as consultas são projetadas desde as saídas da camada *decoder* anterior, enquanto que as chaves-valor são projetadas usando as saídas do *encoder* num mecanismo de **atenção cruzada**.

- ***Camada de avanço totalmente conectada*:** cada camada no *encoder* e *decoder* contém uma rede neural de avanço (FNN) que opera de forma separada e idêntica em cada posição:

$$\text{FNN}(\mathbf{X}) = \text{ReLU}\left(\mathbf{X}\mathbf{W}_1 + \mathbf{b}_1\right)\mathbf{W}_2 + \mathbf{b}_2, \tag{3.30}$$

em que $\mathbf{X}$ é a saída da camada anterior e $\mathbf{W}_1 \in \mathbb{R}^{d_{modelo} \times d_f}$, $\mathbf{W}_2 \in \mathbb{R}^{d_f \times d_{modelo}}$, $\mathbf{b}_1 \in \mathbb{R}^{d_f}$ e $\mathbf{b}_2 \in \mathbb{R}^{d_{modelo}}$ são parâmetros treináveis.

- ***Embeddings*:** para que a informação sequencial possa ser aproveitada pelo modelo, esta deve ser submetida a um processo de codificação em dois passos. No primeiro passo, é utilizado um ***embedding* de entrada** encarregado de converter cada *token* da sequência num vetor de dimensão d_{modelo}. No segundo passo, informação de ordem na sequência é adicionada nas representações vetoriais. Dado que o modelo *Transformer* carece de recorrência ou convolução, a informação posicional absoluta ou relativa é injetada por meio de um ***embedding* posicional** que atribui um código único para cada posição. O *embedding* posicional tem a mesma dimensão que o *embedding* de entrada, de tal forma que os dois possam ser somados. Uma forma de fazer isto é por meio de funções senóides e cossenóides de diferentes frequências:

$$\begin{aligned} \text{PE}_{pos,2i} &= \text{sen}\left(\frac{pos}{10000^{\frac{2i}{d_{modelo}}}}\right), \\ \text{PE}_{pos,2i+1} &= \cos\left(\frac{pos}{10000^{\frac{2i}{d_{modelo}}}}\right), \end{aligned} \tag{3.31}$$

em que *pos* é a posição e i é a dimensão, assim, cada dimensão do *embedding* posicional corresponde a uma senóide.

- ***Conexão residual e normalização*:** apresentam melhorias na aceleração e estabilidade do treinamento. As conexões residuais permitem que a propagação do sinal do gradiente possa fluir de forma efetiva durante o processo de retropropagação. Por sua vez, a camada de normalização faz com que os valores propagados entre camadas apresentem uma escala consistente de entrada, impedindo mudanças drásticas. Desta forma, o *encoder* pode ser escrito com base em (BAHDANAU; CHO; BENGIO, 2014; VASWANI *et al.*, 2017; ZHANG *et al.*, 2023), como:

$$\begin{aligned} \mathbf{X}' &= \text{Atencao}(\mathbf{X}), \\ \mathbf{H}' &= \text{Normalizacao}\left(\mathbf{X}' + \mathbf{X}\right), \\ \mathbf{H} &= \text{Normalizacao}\left(\text{FNN}(\mathbf{H}') + \mathbf{H}'\right), \end{aligned} \tag{3.32}$$

em que $\mathbf{X}$ representa a matriz de entrada do bloco do *encoder*. Cada linha de $\mathbf{X}$ corresponde à representação de um elemento da sequência. A matriz $\mathbf{X}'$ representa a saída produzida pela subcamada de atenção a partir de $\mathbf{X}$. A matriz $\mathbf{H}'$ representa a saída intermediária obtida após a soma residual entre $\mathbf{X}'$ e a entrada original $\mathbf{X}$, seguida da operação de normalização. Por fim, $\mathbf{H}$ representa a saída final do bloco do *encoder*. Ela é obtida após a aplicação da rede neural *feed-forward*, denotada por FNN, sobre a representação intermediária $\mathbf{H}'$, seguida de uma nova conexão residual e de uma operação de normalização.

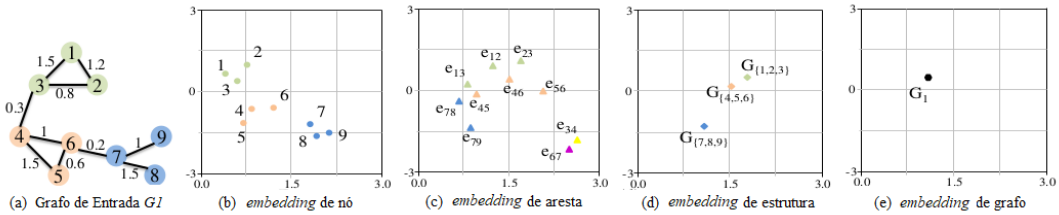
A modularidade do modelo permite seu uso em três formas diferentes. A primeira aplicação como *encoder-decoder*, i.e., a arquitetura completa do *Transformer*, especialmente usada na modelagem de sequência para sequência. A segunda aplicação usando unicamente o *encoder* para gerar representações da sequência de entrada, sendo comumente usado em tarefas de classificação ou rotulagem de sequências. Finalmente, uma terceira aplicação em que unicamente é usado o *decoder*, a camada de atenção cruzada que conecta encoder-decoder é removida permitindo a geração de sequências.

3.3 *Embeddings* de Grafos

Para fazer com que os grafos sejam compatíveis com métodos de aprendizado de máquina e aprendizado profundo, primeiro é necessário representar a sua complexidade topológica numa dimensão menor, i.e., dado um grafo $G = (V, E)$, o *embedding* de um grafo é um mapeamento $f : \mathbf{v}_i \rightarrow \mathbf{z}_i \in \mathbb{R}^d,\ \forall i \in |V|$ tal que $d \ll |V|$ e a função f preserva alguma medida de proximidade definida no grafo G. Assim, um grafo pode ser representado como um vetor de d-dimensões (para todo o grafo) como ilustrado na Figura 28(e), ou um conjunto de vetores de d-dimensões, cada vetor representando uma parte do grafo (por exemplo, nó, aresta, subgrafo) como ilustrado na Figura 28(b, c, d).

A configuração do *embedding* dependerá de uma parte, do tipo de grafo de entrada, o qual é dado e fixo (por exemplo, grafo homogêneo, heterogêneo, dinâmico, etc), e de outra, da tarefa que deseja ser realizada. Mais exatamente, a informação que precisa ser preservada para cumprir a tarefa. Por exemplo, se a tarefa consiste em agrupar nós, a informação dos nós deveria ser preservada no *embedding*, assim, nós que estão próximos no grafo deveriam também estar próximos no espaço do *embedding*, como ilustrado na Figura 28(b). Duas perguntas chave surgem neste ponto: (1) Qual informação preservar? e (2) Como a codificar no espaço do *embedding*.

Figura 28 – Grafo G_1 e *embeddings* de diferentes granularidades num espaço 2D.

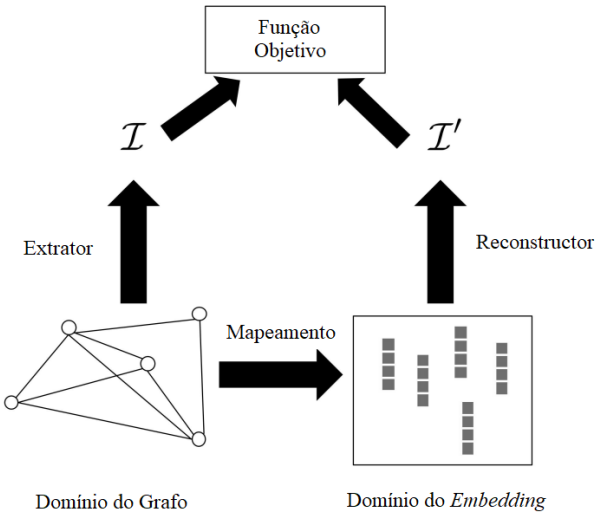


Fonte: Traduzido de Cai, Zheng e Chang (2018).

Para resolver a primeira questão é necessário identificar a granularidade adequada para a tarefa, i.e., se a tarefa é desenvolvida a nível de nós, pares de nós, subgrafos ou grafos inteiros. E de outra parte, o tipo de informação que pode ser extraída do grafo nessa granularidade (por exemplo, vizinhança, rol estrutural ou estado de um nó).

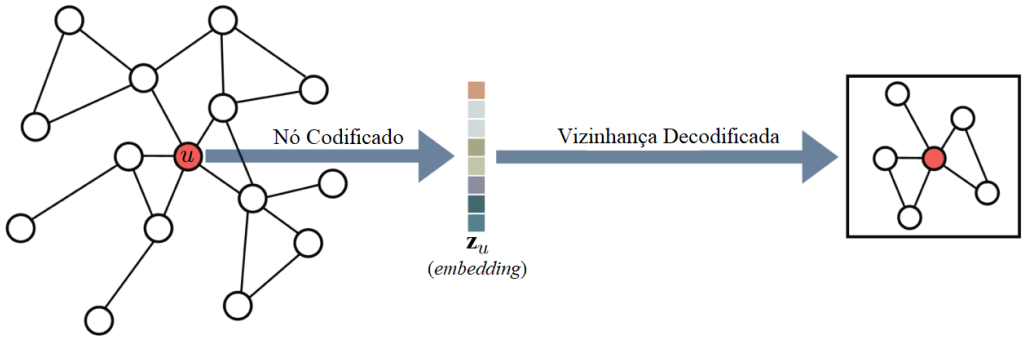
Em relação à segunda questão, Ma e Tang (2021) apresenta uma perspectiva (esquema ilustrado na Figura 29) que abrange uma grande parte dos métodos existentes, e que se baseia na intuição de que boas representações vetoriais no espaço do *embedding* devem ser capazes de reconstruir a informação do grafo que deseja ser preservada. A Figura 30 clarifica esta ideia.

Figura 29 – Arquitetura geral para *embeddings* de grafos. Possui 4 componentes básicos: (1) Uma função que mapeia informação do grafo para o domínio do *embedding*; (2) Uma função extratora que extrai a informação I desejada do grafo; (3) Uma função reconstrutora que constrói uma informação I' desde o *embedding*; (4) Uma função objetivo baseada nas informações I e I' a minimizar. O mapeamento é aprendido minimizando o erro de reconstrução.



Fonte: Traduzido de Ma e Tang (2021).

Figura 30 – A informação da vizinhança do nó deve poder ser reconstruída a partir da representação vetorial no domínio do *embedding*.



Fonte: Traduzido de Hamilton (2020).

A seguir, na Seção 3.3.1 serão exploradas algumas das técnicas de *embeddings* de grafos mais utilizadas. Depois, na Seção 3.3.2 serão revisadas algumas das tarefas mais comuns relacionadas com grafos.

3.3.1 Técnicas de *Embeddings* de Grafos

- **Fatoração de Matrizes:**

Embeddings de grafos baseados na fatoração de matrizes representam propriedades dos grafos (por exemplo, similaridade entre pares de nós) em forma de uma matriz e fatoram esta matriz para obter o *embedding*.

Uma matriz $A \in \mathbb{R}^{n \times n}$ pode ser escrita como o produto entre duas matrizes $X \in \mathbb{R}^{n \times d}$ e $Y \in \mathbb{R}^{d \times n}$ tal que:

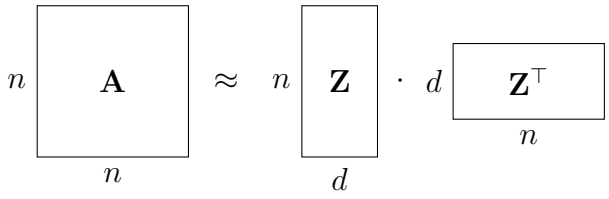
$$\mathbf{A} = \mathbf{XY}. \tag{3.33}$$

Partindo deste princípio, podemos intuir que uma matriz de adjacência A pode ser reconstruída como o produto interno de uma matriz de menor dimensão $Z \in \mathbb{R}^{|V| \times d}$. Porém, dado que o valor da dimensão d do *embedding* Z é muito menor que o número de nós, esta reconstrução é apenas aproximada, como é exemplificado na Figura 31.

Z pode ser aprendido minimizando uma função de perda, com base em (AHMED *et al.*, 2013; CAO; LU; XU, 2015; OU *et al.*, 2016; KOREN; BELL; VOLINSKY, 2009), tal que:

$$\mathcal{L} \approx ||\mathbf{ZZ}^\top - \mathbf{A}||_2^2. \tag{3.34}$$

Figura 31 – A matriz de adjacência A pode ser aproximada como o produto entre uma matriz de menor dimensão Z e sua transposição $Z^\top$.



Os métodos que utilizam este tipo de *embedding* especificamente são conhecidos como **métodos de produto interno**, já que definem uma medida de similaridade entre nós S como alguma função polinomial da matriz de adjacência, e os *embeddings* dos nós são otimizados de tal forma que $z_u^\top z_v \approx S[u, v]$. Algoritmos como *Graph Factorization* (AHMED *et al.*, 2013), *GraRep* (CAO; LU; XU, 2015) e *High-Order Proximity preserved Embedding (HOPE)* (OU *et al.*, 2016) baseiam-se nesta ideia, tendo diferenças em como definem a medida de similaridade S. Por exemplo, enquanto que *Graph Factorization* usa de forma direta a matriz de adjacência, *GraRep* define S a partir de potências da matriz de adjacência.

- **Passeios Aleatórios:**

Os métodos de *embeddings* de produto interno definidos anteriormente utilizam medidas de similaridade entre nós **determinísticas**. A partir desses métodos surgiram outros que adaptaram o enfoque do produto interno para usar medidas de similaridade **estocásticas**. Nesses métodos, *embeddings* de nós são otimizados de forma que dois nós são próximos no espaço do *embedding* se tendem a coocorrer em passeios aleatórios ao longo do grafo.

Partindo da definição de passeio da Seção 3.1.1, um passeio aleatório pode ser definido como uma sequência finita alternada de nós e arestas, que inicia com um nó e finaliza com um nó, na qual cada vértice é incidente com os nós imediatamente anteriores e posteriores e cujo passo se dá de forma aleatória.

Um passeio aleatório pode ser considerado como uma **cadeia de *Markov*** sobre o conjunto de nós V. Por isso, a equação que define um passeio aleatório é uma equação de probabilidades, onde a probabilidade de passar de um nó u para um nó v depende unicamente da posição atual, u. Esta probabilidade é calculada baseada na conectividade dos nós no grafo. Assim, para um grafo sem pesos, se tem que:

$$P(x(t+1) = v | x(t) = u) = \frac{1}{D_u}, \tag{3.35}$$

em que $x(t) = u$ é a posição do caminhante no tempo t e D_u é o grau do nó u. As probabilidades de transição entre todos os pares de nós são representadas pela matriz de

transição $P \in \mathbb{R}^{n \times n}$:

$$\mathbf{M} = \mathbf{D}^{-1}\mathbf{A}. \tag{3.36}$$

onde D^{-1} é a matriz inversa da matriz de grau.

De forma análoga a modelos de processamento de linguagem como ***SkipGram***, sequências de passeios aleatórios podem ser interpretadas como frases curtas (por exemplo, a sequência $[1, 2, 5, 3]$ gerada num passeio aleatório pode ser considerada como a frase "1 2 5 3"). Assim, a probabilidade de observar a palavra w_i (nó v_i) a partir de um contexto de palavras prévias (vértices prévios visitados no passeio) pode ser estimada como $Pr(v_i \,|\, (v_1, v_2, \cdots, v_{i-1}))$.

Porém, dado que o tamanho do contexto pode variar, isto inviabiliza o treinamento de *embeddings* de nós. ***DeepWalk*** (PEROZZI; AL-RFOU; SKIENA, 2014) propõe uma solução para este problema. Primeiro, em vez de usar um contexto para predizer o nó, usa o nó para predizer seu contexto. Segundo, o nó é alocado na metade de uma janela de tamanho w. Por último, remove a condição de ordem entre palavras (nós) e procura maximizar a probabilidade de ocorrência dos nós no contexto de v_i.

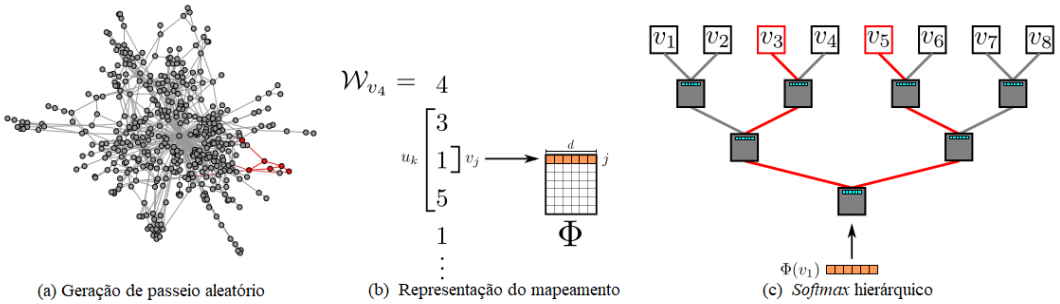
A função objetivo de *DeepWalk* consiste em maximizar a verossimilhança de predizer os nós contextuais dado um nó objetivo na sequência aleatória. A função objetivo pode ser definida como:

$$\mathcal{L} = \sum_{v_i \in \mathbf{V}} \sum_{u_k \in \mathbf{W}_{v_i}} -\log Pr(u_k \,|\, f(v_i)), \tag{3.37}$$

em que $\mathbf{V}$ representa o conjunto de nós do grafo e $v_i \in \mathbf{V}$ corresponde ao nó central, ou nó objetivo, cuja representação vetorial está sendo aprendida. A função $f(v_i)$ representa o vetor de características, ou *embedding*, associado ao nó v_i. O termo u_k representa um nó de contexto que ocorre próximo ao nó central v_i em uma sequência produzida por uma caminhada aleatória. O índice k identifica uma das posições consideradas dentro da janela de contexto. Por sua vez, $\mathbf{W}_{v_i}$ representa o conjunto, ou multiconjunto, de nós de contexto associados a v_i. Esses nós são obtidos a partir das ocorrências localizadas dentro de uma janela de tamanho predefinido ao redor de v_i nas caminhadas aleatórias. Um mesmo nó pode aparecer mais de uma vez em $\mathbf{W}_{v_i}$ caso seja observado repetidamente no contexto de v_i. Assim, o termo $\Pr(u_k \mid f(v_i))$ corresponde à probabilidade de observar o nó de contexto u_k condicionada à representação vetorial do nó central v_i. A minimização de $\mathcal{L}$ equivale à maximização da verossimilhança dos nós de contexto observados nas caminhadas aleatórias.

em que W_{v_i} é um passeio aleatório desde v_i. A probabilidade pode ser computada usando a função *Softmax* como:

Figura 32 – Visão geral do *DeepWalk*. Uma janela de tamanho $2w+1$ é deslizada ao longo do passeio aleatório W_{v4}, mapeando o nó central v_1 para sua representação $\Phi(v_1)$. A função *Hierarchical Softmax* fatora $P(v_3 \mid \Phi(v_1))$ e $P(v_5 \mid \Phi(v_1))$ sobre sequências de distribuição de probabilidade correspondentes aos caminhos começando no nó raiz da árvore binária e terminando nas folhas v_3 e v_5. A representação Φ é atualizada por descida do gradiente descendente para maximizar a probabilidade de v_1 coocorrer com o contexto $\{v_3, v_5\}$



Fonte: Traduzido de Perozzi, Al-Rfou e Skiena (2014).

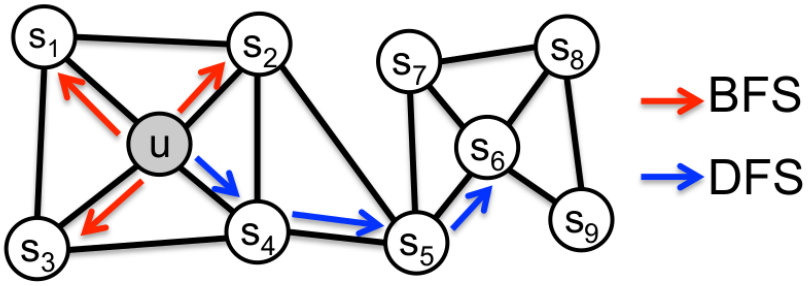
$$Pr(u_k \mid f(v_i)) = \frac{\exp(f(u_k) \cdot f(v_i))}{\sum_{v \in \mathbf{V}} \exp(f(v) \cdot f(v_i))}. \tag{3.38}$$

Dado que o fator normalizador na Equação 3.38 é caro em termos de cômputo, i.e., a somatória sobre o produto para cada nó no grafo tem complexidade $O(|V|)$, o cálculo da Equação 3.37 adquire complexidade $O(|V|^2)$, tornando-a impraticável.

DeepWalk e ***node2vec*** (GROVER; LESKOVEC, 2016) são dois algoritmos baseados em passeios aleatórios que solucionam este problema em formas distintas. Por sua parte, *DeepWalk* usa a função ***Hierarchical Softmax*** que aproxima a função *Softmax*. *Hierarchical Softmax* aloca os nós do conjunto para as folhas de uma árvore binária. Desta forma, o problema é reduzido a maximizar a probabilidade de um ramo específico da árvore, conseguindo assim reduzir a complexidade de $O(|V|)$ para $O(log|V|)$. O procedimento é ilustrado na Figura 32.

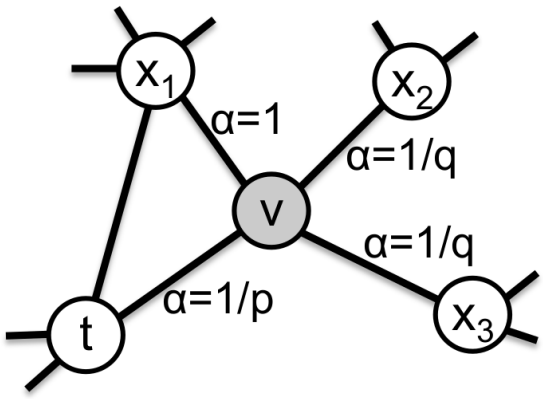
node2vec atinge o problema modificando a forma como é feita a exploração do grafo para a geração de sequências. Para isso são propostas duas formas de exploração. *Breadth-first Sampling* (BFS) restringe a busca para nós próximos do nó de origem, obtendo uma visão micro da sua vizinhança e gerando *embeddings* de nós estruturalmente próximos. Ao contrário da amostragem BFS, *Depth-first Sampling* (DFS) explora regiões profundas do grafo, realizando passeios aleatórios longos e afastados do nó de origem, obtendo assim amostras que expressam uma visão macro da sua vizinhança, essencial para inferir comunidades baseadas em homofilia. As Figuras 33 e 34 apresentam as estratégias de exploração de forma detalhada.

Figura 33 – Estratégias de amostragem BFS e DFS desde o nó $u(k = 3)$



Fonte: Traduzido de Grover e Leskovec (2016).

Figura 34 – As estratégias de amostragem são controladas pelo parâmetro de retorno p e o parâmetro de entrada-saída q. O parâmetro p, como seu nome indica, controla a probabilidade de que o nó atual retorne a um nó anterior, enquanto que o parâmetro q controla a probabilidade de que o nó atual explore de forma profunda ou próxima.



Fonte: Traduzido de Grover e Leskovec (2016).

Para fazer viável o processo de treinamento, *node2vec* aproxima a função de probabilidade *softmax* com um modelo de regressão logística. Em vez de considerar todos os nós no denominador da função *softmax*, apenas um pequeno subconjunto de **amostras negativas** é usado. As amostras negativas são geradas a partir de nós que não estão conectados de forma direta com o nó alvo. A ideia é considerar esses nós não-adjacentes como amostras negativas durante o treino, e assim incentivar o modelo para diferenciar entre nós positivos (adjacentes) e negativos (não-adjacentes). A probabilidade da regressão logística baseada em (PEROZZI; AL-RFOU; SKIENA, 2014; GROVER; LESKOVEC, 2016), é dada por:

$$Pr(u_k \mid f(v_i)) = \log(\sigma(f(u_k) \cdot f(v_i))) - \sum_{i=1}^{k} v_k \sim P_n \, log(\sigma(f(v_k) \cdot f(v_i)), \tag{3.39}$$

em que σ é uma função sigmoide que mapeia cada valor a uma probabilidade entre 0 e 1, k é o número de amostras negativas e $P_n(v_i)$ é uma distribuição aleatória (por exemplo, $P_n(v_i) \sim \frac{1}{|V|}$). Esta modificação faz com que a função objetivo diminua a complexidade de tempo para $O(k|V|)$.

- **Autoencoder:**

Desde uma perspectiva *encoder-decoder*, a arquitetura pode ser resumida em duas funções. A função de mapeamento ou *encoder* que mapeia cada nó do grafo num *embedding*, e a função reconstrutora ou *decoder* que usa a representação vetorial do *embedding* para reconstruir informação do nó no grafo original.

Formalmente, o *encoder* pode ser definido como uma função que mapeia nós $v \in V$ para um vetor *embedding* $\mathbf{z}_v \in \mathbb{R}^d$, onde $\mathbf{z}_v$ corresponde ao *embedding* do nó $v \in V$:

$$\text{ENC} : \mathbf{V} \rightarrow \mathbb{R}^d. \tag{3.40}$$

Para reconstruir informações do grafo original a partir do *embedding*, comumente são definidos *decoders* por pares:

$$\text{DEC} : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}^+. \tag{3.41}$$

Decoders por pares podem ser pensados como preditores de relação ou semelhança entre pares de nós (por exemplo, predizer se dois nós são vizinhos no grafo). Assim, a decodificação por pares dos *embeddings* $(\mathbf{z}_u, \mathbf{z}_v)$ é equiparável a reconstruir as relações entre os nós u e v. O objetivo é otimizar o *encoder* e *decoder* para minimizar a perda na reconstrução tal que:

$$\text{DEC}(\text{ENC}(u), \text{ENC}(v)) = \text{DEC}(\mathbf{z}_u, \mathbf{z}_v) \approx \mathbf{S}[u, v], \tag{3.42}$$

em que $\mathbf{S}[u, v]$ é assumido como uma medida de similaridade baseada em grafos entre dois nós. Para reconstruir a função objetivo definida na Equação 3.42, a prática padrão é a de minimizar uma perda de reconstrução $\mathcal{L}$ sobre um conjunto de treinamento de pares de nós $\mathcal{D}$ com base em (HAMILTON, 2020; CAI; ZHENG; CHANG, 2018):

$$\mathcal{L} = \sum_{(u,v) \in D} l(\text{DEC}(\mathbf{z}_u, \mathbf{z}_v), \mathbf{S}[u, v]), \tag{3.43}$$

em que l é uma função de perda (por exemplo, *mean-squared error ou entropia cruzada*) que mede a diferença entre valores de semelhança decodificados $\text{DEC}(\mathbf{z}_u, \mathbf{z}_v)$ e os valores verdadeiros $\mathbf{S}[u, v]$. Assim, o objetivo geral é o de treinar o *encoder* e *decoder* de tal forma que relações entre pares de nós podem ser efetivamente reconstruídas no conjunto de treinamento $\mathcal{D}$.

3.3.2 Aplicações de *Embeddings* de Grafos

Embeddings de grafos permitem uma ampla variedade de aplicações para a análise de grafos, desde que representações vetoriais podem ser processadas de forma eficiente em tempo e espaço. Nesta seção serão exploradas algumas das suas aplicações a partir de uma perspectiva de nível (nó, aresta e grafo).

- **Aplicações a Nível de Nó:**

Com frequência redes apresentam rótulos. Por exemplo, redes sociais podem estar rotuladas, indicando informações demográficas, interesses ou crenças. Por várias razões, uma grande porção de nós pode apresentar ausência de rótulos (por exemplo, usuários que não compartilham informações demográficas para proteger sua privacidade). Porém, rótulos ausentes podem ser inferidos.

A ***classificação de nós*** é uma das tarefas primárias em *embeddings* de grafos e consiste em atribuir um rótulo de classe para cada nó do grafo baseando-se no aprendizado de nós já rotulados, de tal forma que nós próximos no espaço vetorial deveriam ter os mesmos rótulos. O processo pode ser realizado em três passos. Primeiro, é aplicada alguma técnica de *embedding* para extrair as representações vetoriais dos nós. Depois, os nós com rótulos conhecidos são usados como conjunto de treinamento. Finalmente, um classificador (por exemplo, SVM, Regressão logística, K-Vizinhos mais próximos, etc.) é usado para aprender a partir do conjunto de nós rotulados, que uma vez treinado poderá inferir os rótulos para os nós restantes. Em contraste com o processo sequencial, trabalhos como Li, Zhu e Zhang (2016) optam por uma arquitetura unificada que otimiza de forma conjunta o *embedding* e a classificação do nó.

Ainda que nenhum nó apresente rótulo, estes podem ser agrupados baseando-se em suas propriedades estruturais. O ***agrupamento de nós*** divide os nós de uma rede em grupos de tal forma que nós pertencentes ao mesmo grupo apresentam uma maior similaridade estrutural entre eles quando comparados com nós de outros grupos. Esta similaridade estrutural pode ser baseada em comunidades de nós ou em sua equivalência estrutural. Por meio das representações vetoriais aprendidas no *embedding*, vários métodos não supervisionados de agrupamento como *Kmeans* (DARKEN; MOODY, 1990) podem ser adotados para seu agrupamento.

A tarefa de ***recomendação de nós*** consiste em recomendar k nós de interesse a

partir do nó atual e outros fatores como as propriedades do nó, relações ou topologia da rede. *Content-based filtering* e *Collaborative filtering* (KOREN; BELL; VOLINSKY, 2009) são dois métodos de recomendação populares baseados em multiplicação e fatoração de matrizes, respectivamente. A principal diferença consiste em que o primeiro recomenda nós baseados na similaridade entre atributos e propriedades de nós similares requerendo informação explícita ou descrições dos itens a serem recomendados, enquanto que o segundo faz a recomendação baseado nas preferências e comportamentos de nós (usuários) similares, não requer informação explicita ou descrições de itens sendo recomendados, sendo idôneo para situações em que as preferências de usuários não são descritas pelas características do item ou atributos.

- **Aplicações a Nível de Aresta:**

Redes são construídas a partir de entidades e suas interações que frequentemente podem ser incompletas ou imprecisas. A ***predição de arestas*** é uma tarefa que procura predizer interações faltantes que podem aparecer no futuro, estimando a probabilidade de existência de uma aresta entre dois nós. Dado que os *embeddings* são capazes de aprender representações vetoriais para cada nó, a similaridade entre nós pode ser facilmente estimada por meio de seu produto interno. Uma similaridade maior implica que dois nós podem ser aptos para serem conectados. Para avaliar o desempenho da tarefa de predição de arestas, geralmente são usadas *precision@k* e *Mean Average Precision* (MAP).

- **Aplicações a Nível de Grafo:**

A tarefa de ***classificação de grafos*** consiste em atribuir um rótulo para o grafo inteiro, aplicável quando o grafo inteiro é considerado como a unidade de informação. Geralmente, cada grafo é representado como um conjunto de *embeddings* de nós. Desta forma, dois grafos podem ser comparados a partir de seus conjuntos de *embeddings* de nós. Porém, existem outros enfoques. Yanardag e Vishwanathan (2015) decompõe o grafo num conjunto de subestruturas para representar cada uma como um *embedding* e comparar os grafos a partir da similaridade entre subestruturas.

3.3.3 Limitações

Os métodos de *embeddings* apresentados nesta secção são conhecidos como *shallow embeddings* ou *embeddings* rasos, e apesar de que tem oferecido soluções bem sucedidas a problemas da década passada, é importante mencionar que sofrem de algumas limitações (HAMILTON; YING; LESKOVEC, 2017), que os impossibilitam para certas aplicações desejáveis hoje em dia.

O primeiro problema que possuem os *shallow embeddings* consiste em que a função de mapeamento não compartilha nenhum parâmetro entre nós, i.e., apenas otimiza um único vetor para cada nó. Esta falta de compartilhamento entre parâmetros afeta a

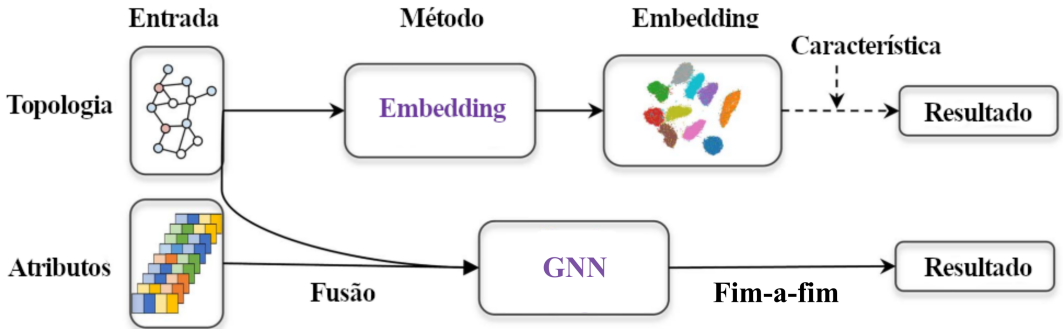
eficiência estatística e computacional, já que o número de parâmetros necessariamente aumenta para $O|V|$, o que pode ser intratável para grafos massivos.

O segundo problema está relacionado com a sua natureza *transdutiva*, i.e., o processo de aprendizado inclui o conjunto de treinamento e teste, o que implica que para cada novo exemplo no conjunto de teste, o processo inteiro deverá ser repetido. Em outras palavras, as técnicas carecem da capacidade para generalizar novos grafos. Esta restrição impede que *embeddings* rasos sejam usados em aplicações de natureza *indutiva* como o tratamento de grafos dinâmicos.

4 Redes Neurais Baseadas em Grafos (GNNs) e Previsão de Epidemias

As redes neurais baseadas em grafos (GNN's, do inglês *Graph Neural Networks*) são modelos que atuam sobre grafos utilizando redes neurais de fim-a-fim. Em comparação com os *embeddings* de grafos explorados no capítulo anterior, as GNN's apresentam várias características importantes, como o compartilhamento de parâmetros entre unidades, a capacidade de generalizar para novos grafos e a capacidade de extrair representações de alto nível, úteis para resolver diferentes tipos de tarefas em grafos. Outra diferença importante entre *embeddings* de grafos e GNN's consiste em que as GNN's aproveitam características topológicas do grafo ao mesmo tempo que características dos nós, como é ilustrado na Figura 35.

Figura 35 – Comparação entre métodos baseados em *embeddings* e métodos baseados em compartilhamento de mensagens



Fonte: Traduzido de Yi *et al.* (2021).

Os primeiros estudos de redes neurais em grafos extraíam representações de nós de alto nível aplicando um único conjunto de parâmetros sobre cada um dos nós de um grafo num processo iterativo e recorrente (SPERDUTI; STARITA, 1997; MICHELI; SONA; SPERDUTI, 2004). Porém, dadas as limitações computacionais, os estudos foram principalmente focados em grafos dirigidos e não-cíclicos. Posteriormente, os modelos propostos em Gori, Monfardini e Scarselli (2005), Scarselli *et al.* (2009) superaram esta limitação lidando com diferentes tipos de grafos (por exemplo, cíclicos, não-cíclicos, dirigidos e não-dirigidos). O modelo utiliza um mecanismo de propagação de informação que atualiza estados ocultos $\mathbf{h}$ dos nós, num intercâmbio de informações com sua vizinhança. Este processo é feito de forma recorrente até que um ponto de equilíbrio seja alcançado:

$$\begin{aligned} \mathbf{h}_v^t &= \sum_{u \in N(v)} f\left(\mathbf{x}_v, \mathbf{x}_{e_{(v,u)}}, \mathbf{x}_u, \mathbf{h}_u^{t-1}\right) \\ \mathbf{o}_v &= g\left(\mathbf{h}_v^t, \mathbf{x}_v\right), \end{aligned} \tag{4.1}$$

em que f e g são funções paramétricas (por exemplo, FNN), $\mathbf{x}_v \in \mathbb{R}^d$, $\mathbf{x}_{e_{(v,u)}} \in \mathbb{R}^c$, $\mathbf{x}_u \in \mathbb{R}^d$ e $\mathbf{h}_u^{t-1} \in \mathbb{R}^b$ são as características de v, as características das arestas em relação aos nós vizinhos, as características e estados ocultos dos seus nós vizinhos, respectivamente. Todos os estados, saídas, características de nós podem ser empilhados em matrizes de grafos $\mathbf{H}$, $\mathbf{O}$, $\mathbf{X}$ e $\mathbf{X_N}$, respectivamente. Assim, é obtida uma forma mais compacta da Equação 4.1:

$$\begin{aligned} \mathbf{H}^t &= f\left(\mathbf{H}^{t-1}, \mathbf{X}\right) \\ \mathbf{O}_v &= g\left(\mathbf{H}^t, \mathbf{X}_N\right). \end{aligned} \tag{4.2}$$

Para aprender os parâmetros das funções f e g, o modelo pode ser treinado de forma supervisionada de acordo a seguinte função de perda, com base em (SPERDUTI; STARITA, 1997; MICHELI; SONA; SPERDUTI, 2004; GORI; MONFARDINI; SCARSELLI, 2005; SCARSELLI *et al.*, 2009):

$$\mathcal{L} = \sum_{i=1}^{V} \left(s_i - o_i\right), \tag{4.3}$$

em que s_i é o valor-alvo real e o_i é o valor estimado pelo modelo para o nó V_i. O processo geral de treinamento segue os seguintes passos. Primeiro, os estados ocultos dos nós são atualizados de forma iterativa de acordo com a Equação 4.1 até a iteração T. Segundo, os gradientes do conjunto de pesos $\mathbf{W}$ das funções paramétricas f e g definidas anteriormente são computados com a função de perda da Equação 4.3. Terceiro, a partir do gradiente, os pesos são atualizados via gradiente descendente.

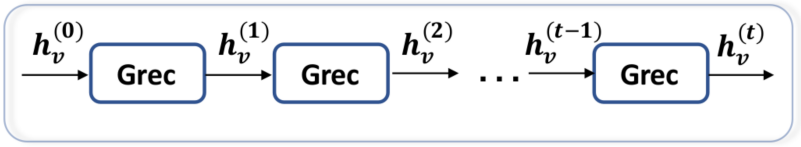
Apesar do modelo proporcionar uma forma efetiva de aplicar redes neurais no domínio dos grafos, apresenta duas limitações importantes. Primeiro, o processo para atualizar as unidades ocultas é iterativo e recorrente, tornando a computação ineficiente, já que para alcançar o ponto de equilíbrio são precisos T passos de cômputo. Segundo, o uso de um mesmo conjunto de parâmetros em todas as iterações reduz sua capacidade de representação.

4.1 Redes Neurais Convolucionais Baseadas em Grafos

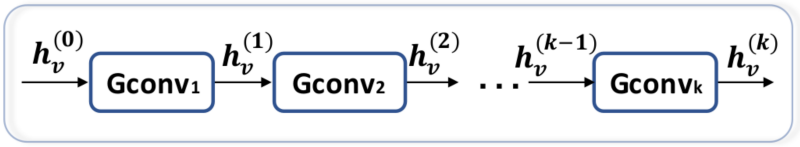
Com o sucesso das CNN's no aprendizado profundo, intuitivamente surge a ideia de generalizar a operação de convolução no domínio dos grafos. Ideia desafiante levando

em conta que a convolução é uma operação comumente aplicada no domínio do espaço euclidiano, i.e., tipos de dados em forma de grade 1D ou 2D, e os grafos como estruturas não regulares operam no domínio do espaço não-euclidiano. Em resposta a este problema surge o estudo das redes neurais convolucionais (ConvGNN's ou GCNNs) a partir de duas abordagens: As **ConvGNN's espaciais** e as **ConvGNN's espectrais**. A abordagem espacial retoma conceitos dos primeiros modelos de GNN's, com a diferença de que em vez de iterar os estados dos nós de forma recorrente, um número fixo de camadas com diferentes conjuntos de pesos **W** em cada camada são usadas. Esta distinção é ilustrada na Figura 36.

Figura 36 – Os modelos de GNN's recorrentes usam a mesma camada recorrente para atualizar as representações dos nós. Em comparação, as ConvGNN's usam diferentes camadas para atualizar as representações.



(a) Modelos de GNN's recorrentes



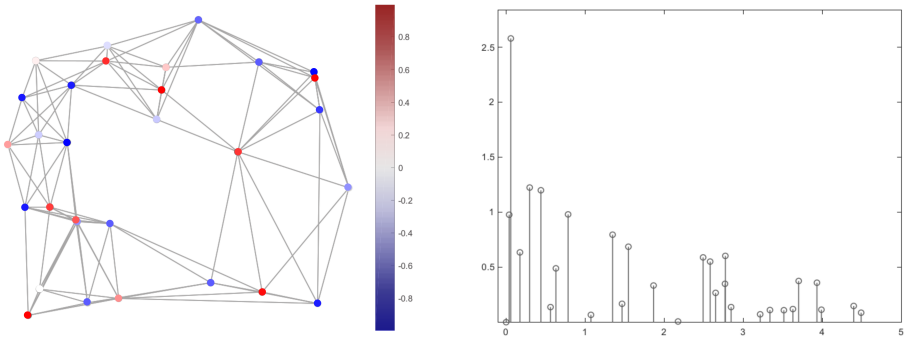
(b) ConvGNN's

Fonte: Traduzido de Liu e Zhou (2020).

Por outro lado, a abordagem espectral se apoia na teoria espectral de grafos, que estuda as propriedades de um grafo por meio da análise dos autovalores e autovetores de sua matriz Laplaciana. A partir desta teoria, Shuman *et al.* (2013) propõe o processamento de sinais em grafos, que define a transformada de Fourier em grafos, conceito chave na conversão do sinal de um grafo no domínio espacial para o domínio espectral. A Figura 37 ilustra as representações da conversão.

Baseado no processamento de sinais em grafos, Bruna *et al.* (2013) propõe a Rede Neural Convolucional Espectral (SCNN), que de forma similar com as GNN's recorrentes utiliza um processo de agregação de características para gerar representações no grafo. Porém, dado que a abordagem espectral depende dos autovalores e autovetores da matriz laplaciana (que a sua vez dependem da estrutura do grafo), modelos treinados a partir de uma estrutura específica não podem ser diretamente aplicados para um grafo de diferente estrutura. Apesar desta limitação, vários outros modelos de ConvGNN's espectrais foram

Figura 37 – ConvGNN Espectral. Na figura (b), o eixo x representa as bases de Fourier e o eixo y indica os correspondentes coeficientes de Fourier para o grafo



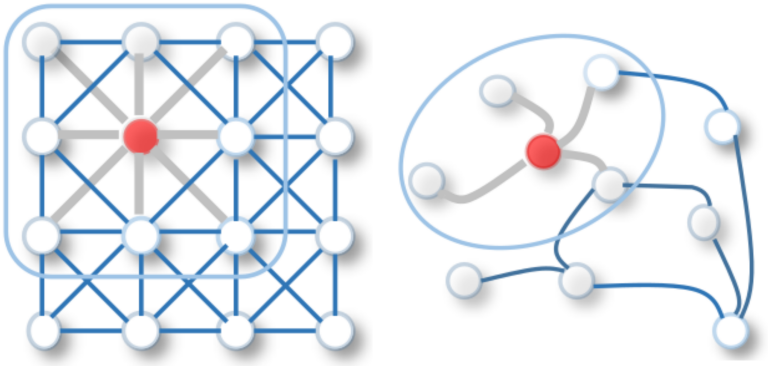
(a) O sinal de um grafo no domínio do espaço (b) O sinal de um grafo no domínio da frequência

Fonte: Traduzido de Liu e Zhou (2020).

propostos (DEFFERRARD; BRESSON; VANDERGHEYNST, 2016; KIPF; WELLING, 2016; LEVIE *et al.*, 2018).

Em comparação com as ConvGNN's espectrais que aplicam a operação de convolução no domínio espectral do grafo, as ConvGNN's espaciais aplicam a convolução de forma direta no grafo, baseando-se nas relações espaciais de cada nó (ATWOOD; TOWSLEY, 2016; NIEPERT; AHMED; KUTZKOV, 2016; HAMILTON; YING; LESKOVEC, 2017). Esta ideia é ilustrada na Figura 38.

Figura 38 – Convolução em Grafo. (a) um filtro de tamanho 3×3 é aplicado sobre a grade, tomando a média ponderada dos valores do "nó"central e seus vizinhos. De forma similar em (b), a convolução toma os valores do nó central e os de seus vizinhos para obter uma representação do nó central, i.e., a convolução propaga a informação dos nós por meio das arestas.



(a) Convolução numa grade 2D (b) Convolução no grafo

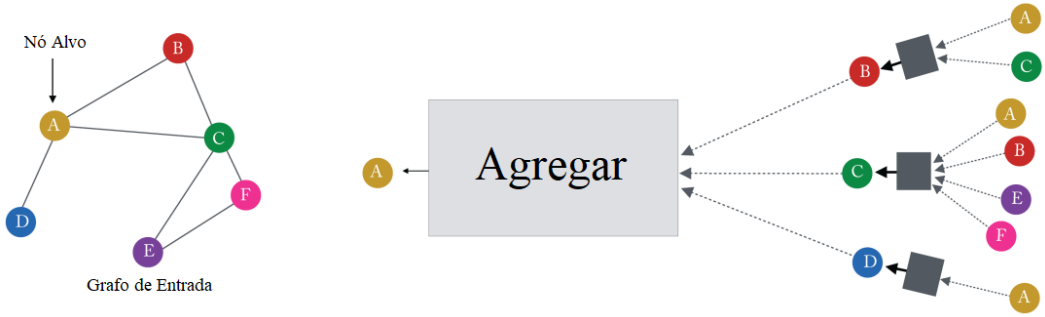
Fonte: Traduzido de Wu *et al.* (2020).

A maioria dos métodos de Redes Neurais Convolucionais baseadas em Grafos citados anteriormente podem ser considerados como instâncias do modelo de **Rede Neural de Passagem de Mensagens** (MPGNN's) proposto por Gilmer *et al.* (2017). Nesta arquitetura, vetores (mensagens) que descrevem os nós são compartilhados de forma iterativa. A cada iteração, um *embedding oculto* $\mathbf{h}_u^k$ correspondente a cada nó $u \in \mathcal{V}$ é atualizado de acordo com a informação compartilhada pela sua vizinhança $\mathcal{N}(u)$:

$$\begin{aligned} \mathbf{h}_u^{k+1} &= \text{ATUALIZAR}^{(k)}\left(\mathbf{h}_u^k, \text{AGREGAR}^{(K)}\left(\mathbf{h}_u^k, \forall v \in \mathcal{N}(u)\right)\right) \\ &= \text{ATUALIZAR}^{(k)}\left(\mathbf{h}_u^k, \mathbf{m}_{\mathcal{N}(u)}^k\right), \end{aligned} \tag{4.4}$$

em que *ATUALIZAR* e *AGREGAR* podem ser modeladas como funções arbitrárias diferenciáveis (FNN) e $\mathbf{m}_{\mathcal{N}(u)}$ é a mensagem compartilhada pelos vizinhos do nó u. O processo é realizado em dois passos. Primeiro, a função *AGREGAR* toma como entrada o conjunto de *embeddings* na vizinhança do nó $\mathcal{N}(u)$ e gera uma mensagem $\mathbf{m}_{\mathcal{N}(u)}^k$. Segundo, a função *ATUALIZAR* combina a mensagem gerada pela vizinhança com o *embedding* anterior $\mathbf{h}_u^{k-1}$ gerando um *embedding* atualizado $\mathbf{h}_u^k$. A Figura 39 descreve este processo de forma intuitiva.

Figura 39 – Exemplo da Operação de Agregação. A cada iteração k, cada nó recebe informação de sua vizinhança. À medida que as iterações progridem, os *embedding* de cada nó contém informações cada vez mais remotas do grafo



Fonte: Traduzido de Hamilton (2020).

Na iteração $k = 0$, as características de entrada do nó são usadas como *embeddings* $\mathbf{h}_u^0 = \mathbf{x}_u$. Para os casos em que os nós não têm características que compartilhar, estatísticas dos nós como seu grau, centralidade ou coeficiente de agrupamento podem ser usadas. Também podem ser incluídas características de identidade como *one-hot-encoding*. Depois de várias iterações de mensagens compartilhadas, a saída da camada final representa os *embeddings* para cada nó, os quais codificam de forma simultânea informações estruturais e de atributos dos nós da vizinhança alcançada:

$$\mathbf{z}_u = \mathbf{h}_u^{(K)}. \tag{4.5}$$

É importante mencionar que dado que a função *AGREGAR* toma como entrada um conjunto, faz com que o modelo seja equivariante à permutação (MARON *et al.*, 2019). Esta é uma propriedade desejável para garantir que a saída do modelo permaneça consistente independentemente da ordem dos nós num grafo. Desta forma, se a matriz de entrada é permutada, a saída da GNN não deveria ser afetada.

A Equação 4.6 apresenta de forma mais detalhada os conceitos da Equação 4.4:

$$\mathbf{h}_u^{k+1} = \sigma\left(\mathbf{W}_{\text{proprio}}^k \mathbf{h}_u^{k-1} + \mathbf{W}_{vizi}^k \sum_{v \in \mathcal{N}(u)} \mathbf{h}_v\right), \tag{4.6}$$

em que $\mathbf{W}_{\text{proprio}}^k$ e $\mathbf{W}_{vizi}^k \in \mathbb{R}^{d^{(k)} \times d^{(k-1)}}$ são matrizes de parâmetros treináveis para o nó atual e os nós vizinhos, respectivamente, e σ é uma função de ativação não linear (por exemplo, tanh ou ReLU). A MPGNN é análoga a uma MLP no sentido em que depende de operações lineares seguidas de uma única operação não linear. Primeiro, são adicionadas as mensagens compartilhadas pelos nós vizinhos. Em seguida, a informação dos vizinhos é combinada de forma linear com a informação prévia do *embedding*. Finalmente, é aplicada uma função não-linear. Definindo as operações a nível de grafo se tem que:

$$\mathbf{H}^{(t)} = \sigma\left(\mathbf{A}\mathbf{H}^{(k-1)}\mathbf{W}_{vizi}^k + \mathbf{H}^{(k-1)}\mathbf{W}_{\text{proprio}}^k\right), \tag{4.7}$$

em que $\mathbf{H}^{(k)} \in \mathbb{R}^{|V| \times d}$ denota a matriz de representação de nós na camada t na MPGNN (um nó por linha na matriz) e $\mathbf{A}$ é a matriz de adjacência do grafo.

A operação de agregação básica na Equação 4.6) $\left(\sum_{v \in \mathcal{N}(u)} \mathbf{h}_v\right)$, dado que apenas realiza a soma dos *embeddings* da vizinhança, pode ser numericamente instável e altamente sensitiva aos graus dos nós. Se, por exemplo, um nó u tem 100 vezes mais vizinhos do que um nó u', na hora de realizar a operação, existirá uma enorme diferença entre suas magnitudes. Uma solução para esse problema consiste em normalizar a operação de agregação baseada no número de graus do nó:

$$\mathbf{m}_{\mathcal{N}(u)} = \frac{\sum_{v \in \mathcal{N}(u)} \mathbf{h}_v}{|\mathcal{N}(u)|}. \tag{4.8}$$

Outra normalização que apresenta bons resultados para este problema em particular é a normalização simétrica usada por Kipf e Welling (2016):

$$\mathbf{m}_{\mathcal{N}(u)} = \sum_{v \in \mathcal{N}(u)} \frac{\mathbf{h}_v}{\sqrt{|\mathcal{N}(u)||\mathcal{N}(v)|}}. \tag{4.9}$$

4.2 Redes Neurais Baseadas em Grafos e mecanismos de atenção

Partindo da hipótese de que as contribuições dos nós vizinhos para o nó central no processo de agregação não são as mesmas, os mecanismos de atenção surgem como uma alternativa para melhorar este processo. Veličković *et al.* (2018) propõem uma camada baseada em mecanismos de atenção para computar os estados ocultos de cada nó num grafo atribuindo pesos de atenção para os nós vizinhos, em que a entrada da camada de atenção é um conjunto de características de nós $\mathbf{h} = \{\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_N\}$, onde $\mathbf{h}_i \in \mathbb{R}^F$, N é o número de nós e F é o número de características em cada nó. E a saída produz um novo conjunto de características de nós $\mathbf{h}' = \{\mathbf{h}'_1, \mathbf{h}'_2, \ldots, \mathbf{h}'_N\}, \mathbf{h}'_i \in \mathbb{R}^{F'}$.

Como primeiro passo, uma transformação linear parametrizada por meio de uma matriz de pesos $\mathbf{W} \in \mathbb{R}^{F' \times F}$ é aplicada para cada nó, depois um mecanismo de atenção compartilhado $a : \mathbb{R}^{F'} \times \mathbb{R}^{F'} \to \mathbb{R}$ computa os coeficientes de atenção

$$e_{u,v} = a\left(\mathbf{W}\mathbf{h}_u, \mathbf{W}\mathbf{h}_v\right). \tag{4.10}$$

Este mecanismo indica a importância das características do nó v para o nó u. Para fazer os coeficientes facilmente comparáveis para todos os nós vizinhos, uma normalização ao longo de todas as escolhas de v é feita usando a função *Softmax*:

$$\alpha_{u,v} = \text{softmax}_v\left(e_{u,v}\right) = \frac{\exp\left(e_{u,v}\right)}{\sum_{k \in \mathcal{N}(u)} \exp\left(e_{v,k}\right)}. \tag{4.11}$$

No artigo original, o mecanismo de atenção a é uma FNN de uma camada parametrizada por um vetor $\mathbf{a} \in \mathbb{R}^{2F'}$ e aplica uma função não linear σ:

$$\alpha_{u,v} = \frac{\exp\left(\sigma\left(\mathbf{a}^\top[\mathbf{W}\mathbf{h}_u || \mathbf{W}\mathbf{h}_v]\right)\right)}{\sum_{k \in \mathcal{N}(u)} \exp\left(\sigma\left(\mathbf{a}^\top[\mathbf{W}\mathbf{h}_u || \mathbf{W}\mathbf{h}_k]\right)\right)}, \tag{4.12}$$

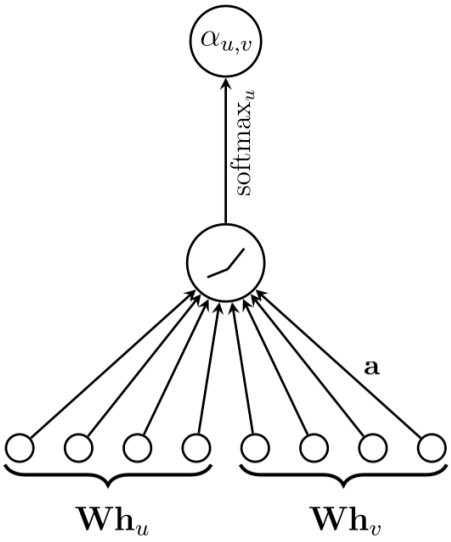
onde $||$ denota a operação de concatenação. A Figura 40 representa o funcionamento do mecanismo de atenção de forma esquemática.

Uma vez obtidos os coeficientes de atenção, são usados para computar uma combinação linear das características correspondentes a eles, obtendo assim as características de saída para cada nó depois de passar por uma função não linear:

$$\mathbf{h}'_u = \sigma\left(\sum_{v \in \mathcal{N}(u)} \alpha_{u,v} \mathbf{W}\mathbf{h}_v\right), \tag{4.13}$$

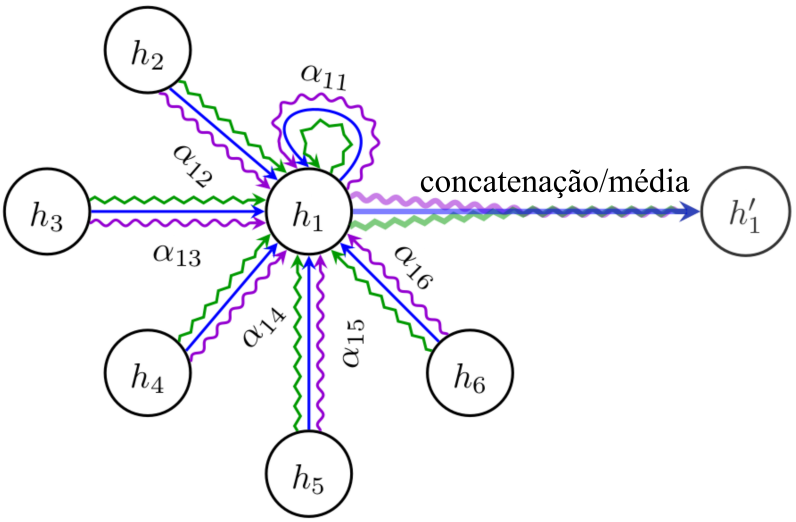
É possível também adicionar múltiplas cabeças de atenção como foi visto na Seção 3.2.5, de forma que K mecanismos de atenção independentes executam a transformação da Equação 4.13, com base em (VELIČKOVIĆ *et al.*, 2018):

Figura 40 – Esquema do mecanismo de atenção



Fonte: Adaptado de Veličković *et al.* (2018).

Figura 41 – Agregação de características no nó h_1 com 3 cabeças, representadas com flechas de cores diferentes. As características agregadas de cada cabeça podem ser concatenadas ou calculadas em média para obter h'_1



Fonte: Traduzido de Veličković *et al.* (2018).

$$\mathbf{h'}_u = \sigma\left(\frac{1}{K}\sum_{k=1}^{K}\sum_{v\in\mathcal{N}(u)}\alpha_{u,v}^{k}\mathbf{W}k\mathbf{h}_v\right). \tag{4.14}$$

O processo de agregação da camada de atenção de múltiplas cabeças está ilustrado na Figura 41.

Como resultado, a camada de atenção traz duas propriedades interessantes para as GNN's. Primeiro, a operação é eficiente dado que é paralelizável ao longo de vários pares de nós vizinhos. Segundo, pode ser aplicada a nós com diferentes graus especificando pesos arbitrários para cada um dos nós vizinhos.

4.3 Modelos de previsão de epidemias baseados em GNNs

Um dos primeiros trabalhos de amplo destaque na previsão de doenças foi o desenvolvido por Deng *et al.* (2020), no qual é proposto um modelo baseado em GNNs para a previsão da influenza de longo prazo (2-15 semanas) usando informação limitada (20 semanas). Para cada grafo são definidos seus nós como localizações (por exemplo, cidades ou estados) ao que estão associados séries temporais de tamanho T. O objetivo é o de predizer um perfil epidemiológico num ponto do futuro $T + h$, onde h faz referência ao horizonte ou tempo de previsão, lembrando que esta é uma tarefa comum nos modelos tradicionais de previsão de epidemias baseados em séries temporais que vimos na Seção 2.3.

A arquitetura proposta por Deng *et al.* (2020), chamada de Cross-location Attention based Graph Neural Networks (Cola-GNN), apresentada na Figura 42 é composta de três módulos. O primeiro, encarregado do aprendizado das dependências temporais de todas as localizações, utiliza o modelo de RNN (visto na Seção 3.2) e o junta com mecanismos de atenção para relacionar as informações espaciais. O segundo módulo chamado de convolução dilatada usa camadas convolucionais (visto também na Seção 3.2) adaptadas com filtros 1D com dilatações diferentes para extrair características locais das séries temporais originais. Como resultado destes dois procedimentos, a saída codifica padrões locais com tendências de curto e longo prazo. No último módulo é desenhado um modelo de propagação da gripe utilizando GNNs por meio do mecanismo de passagem de mensagem.

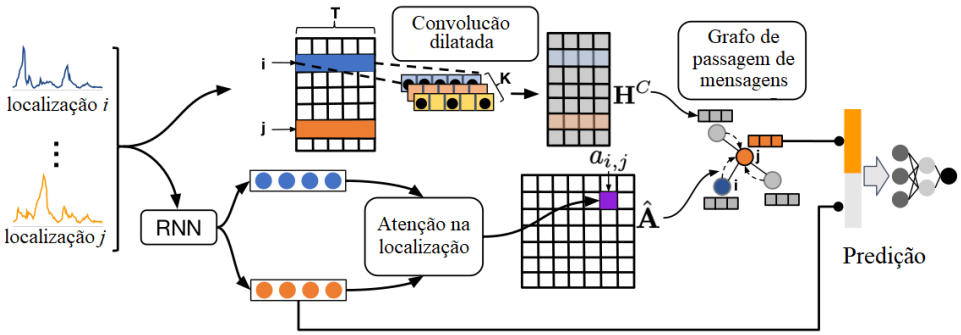
Finalmente, é realizada a predição combinando duas características: a) os estados ocultos da RNN de sua própria sequência de informação histórica para cada localização e b) as características extraídas do grafo.

A otimização é realizada comparando os valores das predições com os valores reais correspondentes, para logo otimizar a função de perda $l1$ normalizada por meio de gradiente descendente, com base em (DENG *et al.*, 2020):

$$\mathcal{L}(\Theta) = \sum_{i=1}^{N} \sum_{m=1}^{n_i} |\boldsymbol{y}_{i,m} - \hat{\boldsymbol{y}}_{i,m}| \lambda \mathcal{R}(\Theta), \tag{4.15}$$

em que n_i é o número de amostras na localização i obtida por uma janela deslocável, compartilhada por todas as localizações, $\boldsymbol{y}_{i,m}$ é o valor real da localização i na amostra m,

Figura 42 – Visão geral da arquitetura proposta. As séries temporais de cada localização são copiadas para (1) o módulo de RNN e (2) o módulo de convoluções dilatadas.



Fonte: Traduzido de Deng *et al.* (2020).

e $\hat{\boldsymbol{y}}_{i,m}$ é a predição do modelo. Θ são os parâmetros de aprendizagem e $\mathcal{R}(\Theta)$ é o termo de regularização. O pseudocódigo de todo o processo encontra-se no Algoritmo 1.

Algoritmo 1: Cola-GNN

Entrada: Séries temporais $\{\mathbf{X}, \mathbf{y}\}$ de múltiplas localizações, matriz geográfica de adjacência $\mathbf{A}^g$

Saída: Parâmetros do modelo $\boldsymbol{\Theta}$

for *cada época* **do**
  Amostra aleatória de *Mini batch*
  for *cada região i* **do**
    $\mathbf{h}_{i,T} \leftarrow$ Módulo RNN$(\mathbf{x}_i)$
    $\mathbf{h}_i^C \leftarrow$ Convolução dilatada$(\mathbf{x}_i)$
  end
  for *cada região par* (i, j) **do**
    $\hat{\boldsymbol{a}}_{i,j} \leftarrow$ Mecanismo de atenção$(\mathbf{h}_{i,T}, \mathbf{h}_{j,T}, \mathbf{A}^g)$
    ▷ Cálculos simultâneos para todas as localizações
  end
  for *cada região i* **do**
    $\mathbf{h}_i^l \leftarrow$ Passagem de mensagens$(\mathbf{h}_i^C, \hat{\boldsymbol{a}})$
    $\hat{\boldsymbol{y}}_{\boldsymbol{i}} \leftarrow$ Saída$([\mathbf{h}_{i,T}; \mathbf{h}_i^{(l)}])$
  end
  $\boldsymbol{\Theta} \leftarrow$ Retropropagação$(\mathcal{L}(\boldsymbol{\Theta}, \mathbf{y}, \hat{\boldsymbol{y}}, \boldsymbol{\Theta}))$
end

Durante a experimentação deste modelo, os valores reais são denotados como $\{\hat{y}_1, ..., \hat{y}_1\}$ e os valores previstos denotados como $\{y_1, ..., y_1\}$. Foram utilizadas métricas de avaliação durante o experimento como *Root Mean Squared Error (RMSE)*, *Mean Absolute Error (MAE)* e *Pearson's Correlation (PCC)*.

Os experimentos foram realizados em três conjuntos de dados de casos de influenza do mundo real (um do Japão e dois dos Estados Unidos) e foi comparado com vários modelos tradicionais estatísticos (*Autorregressão* AR, *Autorregressão Global* GAR, *Autorregressão Vetorizada* VAR e Média Móvel Autorregressiva ARMA) e modelos baseados em redes neurais como RNN, LSTM, CNN, GNN e combinações. A maioria dos modelos baseados em redes neurais apresentaram bons desempenhos, sendo o modelo proposto na Figura 42 superior na maioria das vezes. As métricas usadas de avaliação usadas nos modelos tradicionais, em geral, apresentaram resultados pobres quando em comparação com os modelos de redes neurais.

Observaremos no restante do capítulo como a arquitetura Cola-GNN foi importante para a criação de modelos seguintes de previsão baseados em GNNs devido à eficácia apresentada na previsão de epidemias.

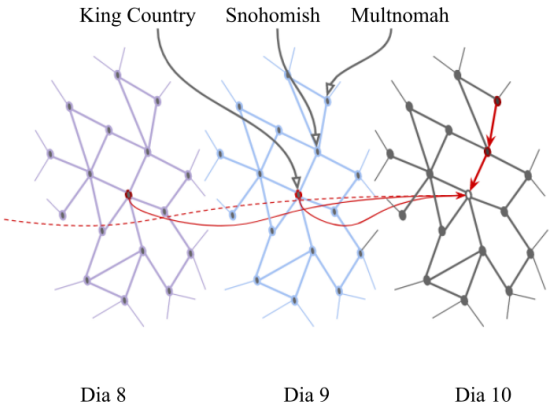
Paralelo ao surgimento da pandemia da COVID-19 em 2020, houve um aumento no desenvolvimento de modelos de GNN com o fim de preparar, entender e controlar o surto. Dois foram os tipos de informação amplamente fornecidos durante a pandemia. A primeira, de tipo espacial, associada aos fluxos de deslocamento humanos em determinadas partes do mundo (*Google Community Mobility Reports* (GOOGLE, 2022), *Facebook Movement Range Maps* (META, 2023)). E a segunda, de tipo temporal ou sequencial, informação associada às taxas de infecção disponibilizadas pelos diferentes países.

Imediatamente teve um surgimento de modelos de previsão da COVID-19 baseados em GNN espaço-temporais, em parte devido à natureza da informação e, de outra parte, pela eficácia e pertinência que o modelo Cola-GNN apresentou para resolver este tipo de problema.

Em Kapoor *et al.* (2020), informação de mobilidade é modelada num grafo ponderado em que cada nó representa uma locação (condado dos Estados Unidos da América EUA) e está conectado com um número arbitrário de nós locações. Os pesos das arestas correspondem com medidas de deslocamento de pessoas entre estados. Para modelar as dependências espaciais e temporais, o grafo apresenta dois tipos de arestas. No domínio do espaço as arestas são ponderadas e representam o movimento direto de uma locação para outra. No domínio do tempo, as arestas representam conexões binárias para dias passados. Assim, o grafo é composto de 100 camadas empilhadas, onde cada camada representa a conectividade do país nesse dia, e cada nó em cada camada é conectado por meio de arestas diretas aos 7 nós anteriores a ele no tempo. A Figura 43 apresenta uma representação visual.

No estudo é proposta uma GNN baseada no modelo espectral convolucional proposto por Kipf e Welling (2016) e modificada com conexões residuais entre camadas para evitar o desaparecimento das características do estado do nó. A predição do modelo $\mathbf{P}$ pode ser representada com base em (DENG *et al.*, 2020; KIPF; WELLING, 2016), como:

Figura 43 – Arestas espaciais representadas em azul e temporais representadas em vermelho. Para maior clareza, apenas as arestas temporais do nó central são apresentadas, porém cada nó apresenta conexões temporais ao longo da sequência.



Fonte: Extraído de Kapoor *et al.* (2020).

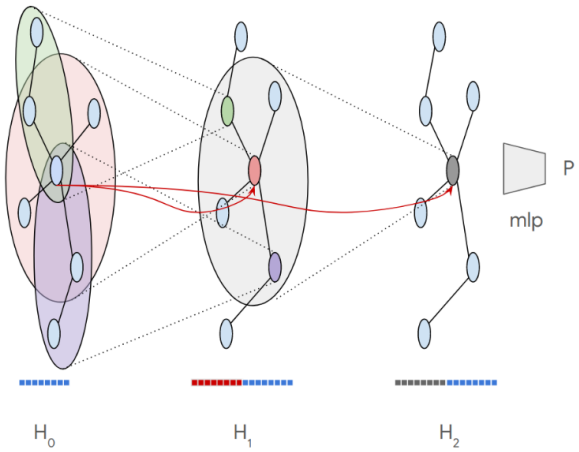
$$\begin{aligned} \mathbf{H}_0 &= \text{MLP}\left(\mathbf{x}_t|\mathbf{x}_{t-1}|\ldots|\mathbf{x}_{t-d}\right) \\ \mathbf{H}_{l+1} &= \sigma\left(\hat{\mathbf{A}}\mathbf{H}_l\mathbf{W}_l|\mathbf{H}_0\right) \\ \mathbf{P} &= \text{MLP}\left(\mathbf{H}_s\right), \end{aligned} \tag{4.16}$$

em que $\mathbf{H}$ representa o estado oculto na camada l, $\hat{\mathbf{A}}$ é a matriz de adjacência espectral normalizada, $\mathbf{W}$ é a matriz de pesos na camada l, $|$ é o operador de concatenação e σ é uma função não-linear. O primeiro *embedding* $\mathbf{H}_0$, codifica as características temporais $\mathbf{x}$ no tempo t dos passados d dias. A predição final $\mathbf{P}$ é a saída de uma MLP sobre s saltos espaciais. A Figura 44 apresenta uma representação visual.

O estudo foi conduzido nos EUA com informação de três conjuntos de dados diferentes. Do conjunto de dados *New York Times (NYT) COVID-19* (The New York Times, 2021) informações de relatórios de infecções e mortes para cada dia nos níveis de estado e país nos EUA são codificadas como características dos nós. Dos conjuntos de dados *Google COVID-19 Aggregated Mobility Research Dataset* e *Google Community Mobility Reports* são usadas informações de magnitude e dinâmica de deslocamento para configurar as relações entre nós.

Para a experimentação foi considerado um horizonte de eventos de sete dias (uma semana). A arquitetura usa um otimizador ADAM com uma taxa de aprendizado de $1e-5$, é composta de duas camadas MLP externas de entrada e saída e quatro camadas ocultas, uma de *embedding*, duas de agregação espacial com dois saltos e a última de predição, as matrizes de pesos para cada camada apresentam tamanhos $[64, 32, 32, 32]$.

Figura 44 – Em cada camada, o *embedding* do nó semente representado em azul, é concatenado e propagado para a camada seguinte de *embedding*. O *embedding* final é passado através de uma MLP usada para predizer P.



Fonte: Extraído de Kipf e Welling (2016).

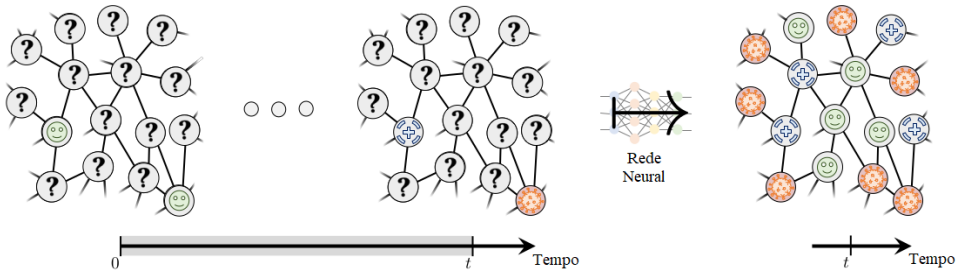
Cada camada implementa uma taxa de *dropout* de $0,5$ e um fator de regularização l2 de $5e-4$. Adicionalmente, o modelo é treinado com a função de perda *Mean Squared Logarithmic Error Loss* (MSLE). O conjunto de dados é dividido com os dias $59-120$ para treinamento e os dias $120-150$ para teste. Por último, o modelo foi comparado com os modelos *Previous Day*, *ARIMA* e *LSTM*, obtendo as melhores ou segundas melhores pontuações para cada métrica.

Os registros de mobilidade populacional podem ser representados de forma natural por meio de grafos, o que possibilita que a previsão de doenças como a COVID-19 seja abordada como um problema de rede de contatos. Nesse contexto, é possível aplicar técnicas de Redes Neurais de Grafos (GNNs) para modelar e analisar essas estruturas. A mobilidade entre regiões funciona como um importante indicador de interação espacial e, consequentemente, de risco de transmissão, uma vez que exerce grande influência sobre o avanço de uma pandemia — especialmente no caso de deslocamentos de longa distância. Dessa forma, o desenvolvimento da doença pode ser previsto a partir das informações sobre os deslocamentos de indivíduos entre diferentes posições geográficas.

Em Tomy *et al.* (2021) inserem um novo algoritmo baseado em GNNs capaz de inferir o estado de uma população depois de um processo de propagação de doença a partir de um número reduzido de exemplos. A reconstrução consegue uma acurácia acima do 70%. A modelagem é baseada no modelo tradicional de previsão por compartimentos, especificamente usando a configuração *Susceptible, Exposed, Infected, Recovered, Deceased (SEIRD)*. A Figura 45 apresenta uma representação visual do problema.

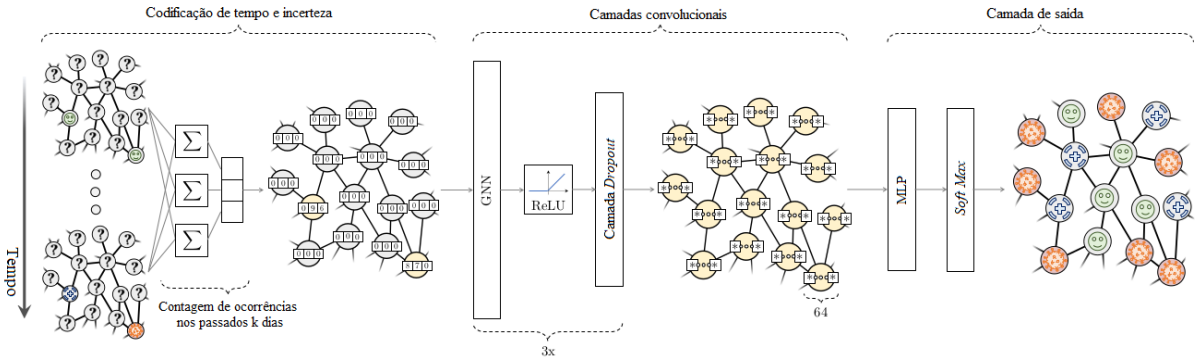
A arquitetura apresentada na Figura 46 é composta por três etapas. Na primeira se faz uma contagem das ocorrências dos três estados possíveis para os passados k dias

Figura 45 – Na prática, quando uma doença é propagada numa rede de indivíduos, os agentes de controle de saúde apenas conseguem medir o estado de um pequeno número de indivíduos, com a vantagem de testá-los continuamente. Na rede, cada nó é representado como um indivíduo, e o símbolo associado representa o estado do indivíduo. O símbolo de pergunta, por exemplo, simboliza os indivíduos cujo estado é desconhecido. No tempo inicial, os dois indivíduos conhecidos estão saudáveis, enquanto no tempo t estão infectados ou recuperados. O objetivo é reconstruir o estado completo da rede conhecendo a maior quantidade possível do estado dos indivíduos.



Fonte: Traduzido de Tomy *et al.* (2021).

Figura 46 – Arquitetura para a predição do estado de propagação da doença na população



Fonte: Traduzido de Tomy *et al.* (2021).

criando rótulos que codificam a informação temporal. A segunda etapa é composta de três camadas de grafos convolucionais. Finalmente, na última etapa, a informação é comprimida de novo pela camada de saída. Como resultado, a saída obtida é uma estimação do estado atual de propagação da epidemia.

No trabalho Cao *et al.* (2022), os autores também exploram as definições dadas pela modelagem tradicional de previsão por compartimentos, apresentando um bom desempenho na predição do número de casos confirmados de COVID-19.

Até o momento, muitos outros trabalhos relacionando GNNs e previsão da COVID-19 seguem sendo desenvolvidos. Em Hy *et al.* (2022), por exemplo, introduzem um mecanismo de atenção que seleciona a resolução adequada que ofereça a melhor predição

do estado de uma epidemia/pandemia, permitindo extrair informações a nível micro e macro. Em Xue *et al.* (2022), os autores predizem as ondas de surtos da COVID-19 em Tóquio, antecipando possíveis casos por meio das consultas de sintomas dos indivíduos.

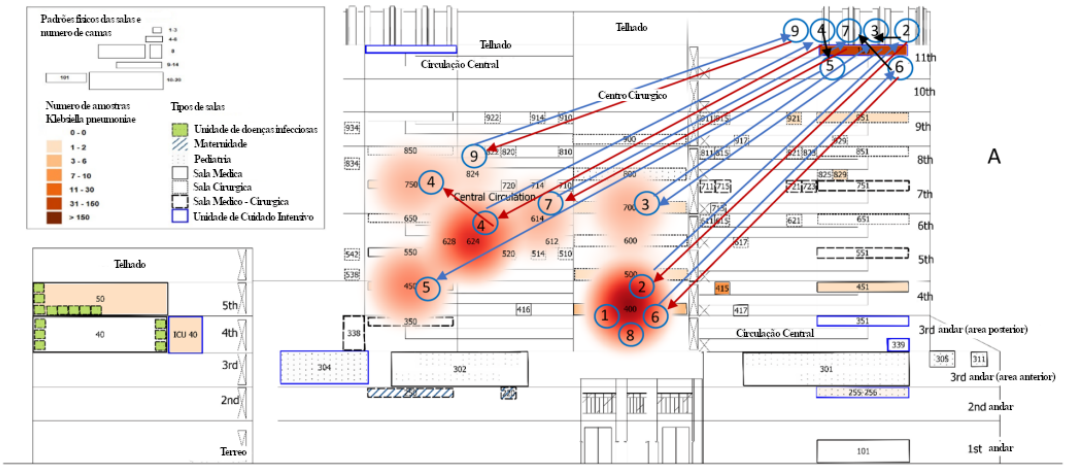
Outros estudos de interesse que não estão relacionados diretamente à COVID-19 foram os casos de Shah *et al.* (2020), no qual são estudadas as dinâmicas de contágio usando GNN para predizer o paciente zero, conseguindo inferir com êxito a fonte de um surto, e Mežnar, Lavrač e Škrlj (2021), em que é investigada em detalhes a capacidade das GNNs para predizer o efeito de uma epidemia (análises de espalhamento), devendo esta começar desde um indivíduo (paciente zero), comparando-a com simulações tradicionais que implicam um alto custo computacional. Os resultados demostraram uma grande diminuição do custo com estimações realistas.

5 Justificativa, Hipóteses e Metodologia

Nesse estudo, foi proposto o desenvolvimento de um modelo dinâmico de rede neural baseada em grafos para a previsão da circulação de espécies bacterianas em ambiente intra-hospitalar, cuja metodologia será apresentada subsequentemente, tendo como base o estudo realizado em um hospital público, federal e terciário do Rio de Janeiro, por Silva *et al.* (2021), sob a coordenação da Professora Marisa Zenaide Ribeiro Gomes.

No estudo foi investigada a circulação do complexo *Klebsiella pneumoniae*, que corresponde a um complexo de *Gram-Negative Bacteria* (BGN) que se apresenta em hospitais, comumente, com o perfil MDR, relacionado a altas taxas de morbidade e mortalidade (SILVA *et al.*, 2021). No entanto, durante o período de estudo (2014 - 2016), o agente apresentou uma evolução para cepas PDR/XDR, dando como resultado focos de infecção intratáveis e uma epidemia, a qual foi reportada no estudo subsequente (GOMES *et al.*, 2023). Para entender a dinâmica de circulação do agente foi realizada, no estudo inicial, uma análise estatística espacial usando dados de mobilidade ou transferência entre enfermarias intensivas ou não intensivas do hospital, de pacientes albergando o complexo, com os perfis de susceptibilidade microbiana MS, MDR, XDR e PDR. As ocorrências desses perfis de susceptibilidade foram então georreferenciadas e plotadas, por data, sobre mapas do hospital, construídos em AutoCAD, e depois levados para o software QGIS, dando origem a um Sistema de informação Geográfica (*Geographical Information System - GIS*) hospitalar, como é mostrado na Figura 47.

Figura 47 – O mapa apresenta as conexões e seu fluxo entre agrupamentos de pacientes infectados no hospital.



Fonte: Traduzido de Silva *et al.* (2021).

Duas são as hipóteses propostas nesta pesquisa: a) Técnicas de aprendizado profundo

baseadas em grafos aplicadas como modelos epidemiológicos podem explicar e capturar padrões de propagação de espécies bacterianas intra-hospitalares, no tempo e no espaço; b) modelos epidemiológicos baseados em técnicas de aprendizado profundo baseadas em grafos podem realizar melhores predições da presença de espécies bacterianas intra-hospitalares, em comparação com modelos tradicionais de previsão e modelos de aprendizado profundo não baseados em grafos.

A metodologia para verificar as hipóteses levantadas está detalhada nas próximas seções. A Seção 5.1 apresenta os diferentes dados gerais fornecidos pelo hospital terciário, universitário do Rio de Janeiro, que servirá de modelo, assim como suas relações. A Seção 5.2 apresenta a construção do grafo e a sequência de grafos para representar a circulação intra-hospitalar das espécies bacterianas detectadas. Na Seção 5.3 é descrita a tarefa de previsão. Na Seção 5.4 é apresentado o modelo espaço-temporal em detalhe. Finalmente, na Seção 5.5, são apresentadas as considerações éticas que respaldam o trabalho. Nessa primeira abordagem, em função do quantitativo de dados, optamos por investigar todos os germes detectados nas culturas coletadas durante os períodos especificados em todos os setores do hospital, sem se ater a um perfil de susceptibilidade aos antimicrobianos específico.

5.1 Conjunto de dados e configuração do hospital

A Tabela 1 apresenta informações do tamanho do hospital por andar e resultados de exames microbiológicos realizados em pacientes internados, dentro da rotina de investigação de processos infecciosos das equipes médicas e da Comissão de Controle de Infecção Hospitalar (CCIH) do hospital de estudo, entre 1 de outubro de 2022 e 29 de maio de 2024. Os dados utilizados foram coletados numa planilha de Excel, a partir de uma macro vinculada às bases de dados do sistema de informação do hospital. Esta macro foi desenvolvida pelos integrantes da pesquisa com o suporte do setor de Tecnologia ou da Informação (TI) do hospital. A partir desta planilha foram obtidos dois conjuntos de dados: (i) 8.706 registros hospitalares de pacientes que apresentaram detecção de bactérias em amostras biológicas (urina, sangue e *swab* retal, por exemplo) com resultados AST para os respectivos antibióticos testados para cada espécie bacteriana, e de acordo com as recomendações do *Clinical and Laboratory Standards Institute - CLSI*, *European Committee on Antimicrobial Susceptibility Testing - EUCAST* e *Brazilian Committee on Antimicrobial Susceptibility Testing - BRCAST*; (ii) 12.175 registros de admissões, transferências intra-hospitalares e altas dos pacientes envolvidos no primeiro conjunto de dados.

Informação adicional da infraestrutura do hospital usando mapas técnicos (*Computer-Aided Design - CAD*), estados atuais das enfermarias e relações de vizinhança foram

Tabela 1 – Informação geral do hospital - Distribuição das salas e camas do hospital nos diferentes andares considerados neste estudo.

Andar	N. Salas	N. Camas
2	15	96
3	17	122
4	14	117
5	10	122
6	6	31

Fonte: Setores de TI e CCIH do hospital.

fornecidos pelos membros da pesquisa principal, incluindo uma arquiteta do hospital, que atualizaram esses dados a partir dos mapas disponibilizados pelo Setor de Engenharia e Arquitetura do hospital CCIH. Após limpeza e validação cruzada dos dados, os conjuntos de dados foram consolidados em quatro arquivos .csv. O conjunto `nodes.csv` define os identificadores das enfermarias, coordenadas espaciais e capacidade de leitos; `edges.csv` contém as relações de vizinhança fixas entre nós (contiguidade espacial definida pela coluna `<spatial>` ou contiguidade por serviço, referentes a enfermarias pertencentes a um mesmo serviço e definida pela coluna `<service>`) que formam a topologia estática do hospital; `profiles.csv` inclui informação mascarada de registros de pacientes positivos para o perfil de bactéria, resultados de AST e datas de coleta; e, por último, `transferences.csv` apresenta os registros intra-hospitalares de transferências de pacientes entre leitos/enfermarias do hospital (os identificadores no campo `<patient_id>` são os mesmos id mascarados definidos em `profiles.csv`), por mortes, altas e/ou transferências para outras unidades hospitalares, com suas respectivas datas, usados para construir o histórico de mobilização do paciente ao longo do tempo. Os dados em conjunto permitem a construção do grafo base e sequência de grafos. A Tabela 2 detalha sua organização.

5.2 Construção do grafo

Para capturar padrões de deslocamento de espécies bacterianas, ao longo do hospital no tempo e espaço, primeiro é necessário definir o espaço em que tais deslocamentos vão acontecer, i.e., uma estrutura sobre a qual o modelo possa agregar informação topológica. Assume-se que a configuração do espaço físico não tenha mudado antes da coleta de informação.

A estrutura física do hospital é representada pelo grafo $G = (V, E_{base})$, onde $V = \{v_0, \ldots, v_N\}$ é o conjunto de salas e o nó v_0 representa o exterior do hospital, usado quando pacientes são admitidos ou dados em alta, óbito, ou transferido para outro hospital. Cada sala v é relacionada com um vetor de coordenadas 3D $\mathbf{c}_v = (x_{coor}, y_{coor}, z_{coor})$ obtido a partir de medidas dentro da escala dos mapas arquitetônicos do hospital (Fig. 48). Relações de vizinhança fixas definidas pelos pesquisadores, com o apoio de uma enfermeira

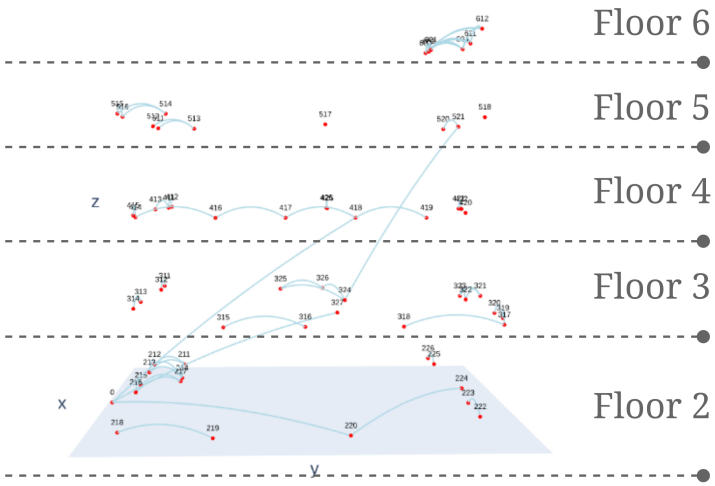
Tabela 2 – Organização do conjunto de dados

Arquivo	N. Linhas	N. Colunas	Headers
nodes.csv	64	8	<floor,ward,x,y,z,internal_id,name,bed_number>
edges.csv	499	8	<source,target,service,spatial,service_spatial>
profiles.csv	8,706	47	<id,collection_date,bacteria,acid_nalidixic, amikacin,amoxicillin,amoxicillin_clavulanate, ampicillin,ampicillin_sulbactam,azithromycin, azithromycin_oral,azithromycin_parenteral, aztreonam,carbapenemic,carbapenems,cephalexin, cephalothin,cefazolin,cefepime,cefotaxime,cefoxitin, ceftazidime,ceftazidime_gradient_strip, ceftazidime_avibactam,ceftolozane_tazobactam, ceftriaxone,ceftriaxone_gradient_strip,cefuroxime, cefuroxime_axetil,ciprofloxacin,ciprofloxacin_oral, ciprofloxacin_parenteral,clarithromycin, clarithromycin_oral,clarithromycin_parenteral, clindamycin,clindamycin_oral,clindamycin_ parenteral,chloramphenicol,colistin,daptomycin, doxycycline,eritromycin,ertapenem,fosfomycin, gentamicin,gentamicin_120>
transferences.csv	8,531	8	<patient_id,out_ward,in_ward,movement_date, movement_type,site,death,date_of_death>

Fonte: Setores de TI e CCIH do hospital.

da CCIH são representadas por meio de um conjunto de arestas estáticas $E_{base} \subseteq V \times V$.

Figura 48 – Grafo fixo do hospital. Cada ponto vermelho representa uma sala, e cada sala está acompanhada do seu número identificador. As linhas azuis representam as relações de vizinhança fixas. As linhas cinzas pontilhadas delimitam a separação aproximada por andar.



Cada sala v é associada com um vetor de características $\mathbf{x}_v \in \mathbb{R}^{d_v}$ em que d_v é a dimensão do vetor. Este vetor combina atributos estruturais e epidemiológicos como número de camas, ocupação atual e informação ponderada de pacientes hospedando

Tabela 3 – Atributos de sala (Nó). Visão geral dos atributos de sala (nó) usados na construção do grafo, incluindo tipo de dado e exemplos.

Atributo	Descrição	Tipo	Exemplo
coordinates	Coordenadas da sala no espaço do hospital (3D)	tuple	(-800.0, 90.0, 0.0)
<bed_number>	Número total de camas na sala	int	8
<occupied_bed_number>	Número de camas ocupadas na sala	int	1
<internal_id>	Identificador interno da sala	int	1
name	Nome da sala	string	neurologia feminina
<current_patients>	Lista de pacientes atuais ocupando a sala	lista de Patient	{32: Patient}
<patient_vectors>	Lista de vetores *one-hot* representando os perfis infecciosos dos pacientes que ocupam a sala	lista de arrays	{32: [1.0, 0.0, . . . , 0.0]}
<patient_panel_lite>	Lista de vetores *one-hot* com informação infecciosa resumida dos pacientes que ocupam a sala	lista de arrays	{32: [0.0, 0.0, 1.0, 0.0, 0.0]}

Tabela 4 – Atributos de cada paciente presente dentro das salas.

Atributo	Descrição	Tipo	Exemplo
id	Id do paciente (mascarado durante a ingesta da informação)	int	32
<resistance_profiles>	Lista de perfis microbiológicos registrados até a data do grafo, incluindo espécie bacteriana e resultados AST; cada item é um objeto do tipo PatientInfectionProfile	lista de PatientInfectionProfile	`[PatientInfection Profile]`

espécies bacterianas detectadas durante a investigação de quadros infecciosos ou de vigilância microbiológica preconizada na instituição. Os atributos de sala, paciente, e agente bacteriano usados para construir essas características estão detalhados nas Tabelas 3, 4 and 5.

A forma em que são realizados os deslocamentos ao longo do grafo depende do processo de ocorrências deste tipo de agentes bacterianos, e das movimentações dos pacientes ao longo do hospital. A confluência de pacientes hospedando a mesma espécie bacteriana, independentemente de apresentarem ou não perfil MDR, indicaria a necessidade de investigação de situação epidêmica, uma vez que pode indicar a transferência de bactérias entre pacientes, o que pode acontecer de forma direta, quando um paciente entra em contato com outros pacientes e profissionais da saúde, ou de forma indireta, quando

Tabela 5 – Atributos descrevendo cada perfil microbiológico usado para obter as características da representação microbiológica do paciente.

Atributo	Descrição	Tipo	Exemplo
antibiotics	Dicionário mapeando antibióticos com o resultado AST associado a esta amostra ou perfil microbiológico	dicionário	{amikacin: Sensível, ...}
bactéria	Bactéria associada com o perfil microbiológico	string	Pseudomonas aeruginosa
<collection_date>	Data de coleta de amostra para este perfil microbiológico	datetime.date	2022-10-05

equipamentos médicos, leitos comuns, roupas ou superfícies são contaminados (SILVA *et al.*, 2021; SNITKIN *et al.*, 2012).

É importante considerar que o raio de contaminação pode variar por duas razões. Primeiro, por que o paciente espirra, solta gazes e evacua, causando a aerossolização de partículas capazes de carregar o patógeno mais longe (MIRHOSEINI *et al.*, 2016). E segundo, por que alguns profissionais da saúde (por exemplo, enfermeiro, coletor de sangue, nutrólogo, etc) que circulam por todas as salas tem contato com vários pacientes (CASEWELL; PHILLIPS, 1977), além de aerossolização de partículas que podem ocorrer durante o cuidado com os pacientes, como na aspiração de vias aéreas, durante o recolhimento de roupas de camas, e até mesmo durante a higiene ambiental, se não são observadas as recomendações específicas para o cuidado do paciente e áreas clínicas, no ambiente hospitalar.

Porém, dado que o paciente é o principal reservatório da bactéria e que a informação fornecida na base de dados se refere a pacientes com amostras positivas para alguma espécie bacteriana registrada no estudo, apenas serão considerados como deslocamentos as transferências entre salas dos pacientes carregadores dos agentes bacterianos.

Dado que múltiplas transferências de pacientes podem ocorrer em um mesmo dia, os registros de movimentação são organizados como uma sequência de eventos discretos. A partir desses eventos, é construída uma sequência temporal de *snapshots* dinâmicos do grafo hospitalar $\mathcal{G} = \{G^{(1)}, \ldots, G^{(T)}\}$, todos definidos sobre o mesmo conjunto de nós V, correspondente às enfermarias ou unidades hospitalares. No experimento considerado, foram gerados $T = 1562$ *snapshots*.

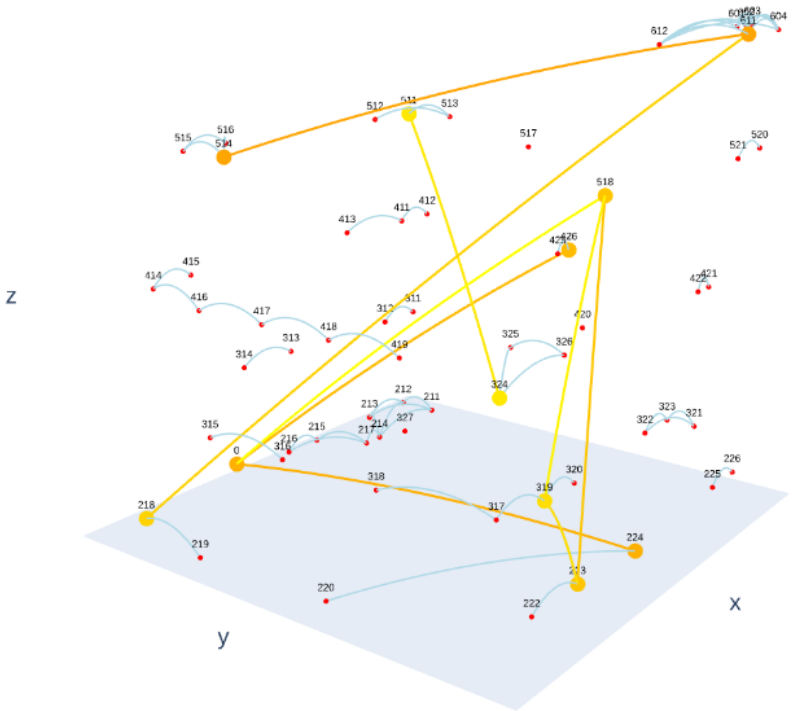
Cada *snapshot* $G^{(t)} = (V, E_{\text{base}} \cup E^{(t)}_{\text{mov}}, \mathbf{X}^{(t)})$ representa o estado do grafo após a aplicação de um grupo ordenado de transferências, sendo $E_{mov} \subseteq V \times V$ o conjunto de arestas dinâmicas ou transferências intra-hospitalares dos pacientes no evento t. Esse grupo é definido pela data da movimentação e pela ordem intra-diária do evento. Assim, um mesmo dia d pode estar associado a múltiplos *snapshots* $\{G^{(t)} : t \in \mathcal{T}_d\}$, quando há mais

de uma ordem de movimentação nesse dia, sendo $\mathcal{T}_d$ o conjunto de eventos associados a um dia d (Fig. 49).

O modelo utiliza diretamente a sequência de *snapshots* por evento como entrada. Para uma janela temporal iniciada no passo t, são fornecidos ao modelo os grafos $\{G^{(t)}, \ldots, G^{(t+T_{in}-1)}\}$. Cada *snapshot* contém a estrutura de vizinhança hospitalar selecionada, eventuais arestas temporárias associadas às transferências do passo corrente e atributos dinâmicos dos nós, como pacientes presentes e ocupação.

O alvo de previsão, por sua vez, é definido em escala de dias calendário. Dado o dia do último *snapshot* da janela de entrada, o modelo prediz a presença bacteriana nas enfermarias no horizonte futuro H, isto é, no dia correspondente a $d + H$. Para isso, utiliza-se o último *snapshot* disponível do dia alvo.

Figura 49 – *Snapshot* 1200. Os movimentos de pacientes acontecidos durante cada *snapshot* são apresentados usando uma escala de cor amarelo-laranja para distinguir cada movimentação de forma individual.



5.3 Definição do problema de previsão

Dada uma sequência temporal de grafos hospitalares, o objetivo é predizer, para cada sala e classe bacteriana, a probabilidade de presença futura no horizonte calendário H. Cada *snapshot* $G^{(t)}$ em $\mathcal{G}$ possui uma data calendário associada, denotada por $\tau(t)$.

Seja v_j o nó correspondente à sala j. Para cada instante t, o modelo recebe uma janela de T_{in} *snapshots* consecutivos $\left\{G^{(t-T_{in}+1)}, \ldots, G^{(t)}\right\}$, em que T_{in} representa o número de *snapshots*/eventos considerados na entrada. Um codificador espaço-temporal processa essa janela e produz uma representação latente $\mathbf{h}_j^{(t)} \in \mathbb{R}^{d_h}$ para cada sala j.

Seja B o número de classes bacterianas sob investigação. O decodificador mapeia $\mathbf{h}_j^{(t)}$ para um vetor de probabilidades com base em (SKARDING; GABRYS; MUSIAL, 2021; HAMILTON; YING; LESKOVEC, 2017),

$$\hat{\mathbf{y}}_j^{(\tau(t)+H)} = \varphi(W\mathbf{h}_j^{(t)} + \mathbf{b}), \qquad \hat{\mathbf{y}}_j^{(\tau(t)+H)} \in \{0,1\}^B, \tag{5.1}$$

em que $\mathbf{W} \in \mathbb{R}^{B\times d_h}$ e $\mathbf{b} \in \mathbb{R}^B$ são parâmetros treináveis, e φ é uma sigmoide aplicada elemento por elemento, permitindo estimar independentemente a probabilidade de presença de cada classe bacteriana. O componente $\hat{y}_{j,b}^{(\tau(t)+H)}$ representa a probabilidade prevista de que a classe bacteriana b estará presente na sala j no dia calendário $\tau(t)+H$.

$$y_{j,b}^{(\tau(t)+H)} = \begin{cases} 1, & \text{se a sala } j \text{ contém pelo menos um paciente infectado ou} \\ & \text{colonizado com a bactéria } b \text{ no dia calendário } \tau(t)+H, \\ 0, & \text{caso a sala } j \text{ não contenha paciente infectado ou colonizado com} \\ & \text{a bactéria } b \text{ no dia calendário } \tau(t)+H. \end{cases} \tag{5.2}$$

Elaborado pelo autor com base em (SKARDING; GABRYS; MUSIAL, 2021).

A tarefa é, portanto, formulada como uma classificação multirrótulo a nível de nó, em que indicadores futuros de presença bacteriana são previstos para cada sala. Diferentemente de uma formulação baseada em *snapshots* diários, a janela de entrada é definida em termos de eventos: T_{in} indica o número de *snapshots* consecutivos observados, e não necessariamente o número de dias anteriores. O horizonte H, por outro lado, é definido em dias calendário a partir da data associada ao último *snapshot* da janela.

5.4 Modelo espaço-temporal baseado em grafos

Cada *snapshot* $G^{(t)}$ é representado por uma matriz de adjacência $A^{(t)} \in \mathbb{R}^{N\times N}$ e uma matriz de características dos nós $X^{(t)} \in \mathbb{R}^{N\times d_v}$. As características dos nós codificam informações no nível da sala, obtidas a partir da agregação dos dados dos pacientes presentes e de atributos relacionados à ocupação da enfermaria (Fig. 51).

O vetor de características é construído de forma progressiva em dois níveis. No primeiro nível, os perfis microbiológicos dos pacientes são codificados em *embeddings* densos. Inicialmente, cada espécie bacteriana observada no conjunto de dados é mapeada para um identificador inteiro em um vocabulário bacteriano descrito no Apêndice B. Em seguida,

esse identificador indexa uma tabela de embeddings treinável, representada por uma matriz de parâmetros $\mathbf{E}_{bact} \in \mathbb{R}^{B \times d_b}$, em que $B = 96$ é o número de classes bacterianas e $d_b = 16$ é a dimensão do embedding bacteriano. Assim, cada classe bacteriana b é associada a um vetor denso $\mathbf{e}_b \in \mathbb{R}^{d_b}$.

Em paralelo, é construída uma representação fenotípica dos resultados AST a partir da lista fixa de antibióticos investigados no estudo. Cada antibiótico é representado por uma codificação *one-hot* de cinco estados: vazio, não testado, resistente, sensível e sensível à dose padrão. Assim, para $A = 44$ antibióticos, o vetor fenotípico AST possui dimensão $5A = 220$. Na configuração enriquecida, são adicionadas cinco estatísticas resumidas do painel AST, correspondentes às frações de antibióticos vazio, não testado, resistente, sensível e sensível à dose padrão, resultando em um vetor de dimensão $5A + 5 = 225$.

O vetor fenotípico é processado por uma MLP totalmente conectada, que projeta a representação AST original, de 220 ou 225 dimensões, para uma representação latente de 64 dimensões. Essa representação é então concatenada ao embedding treinável da bactéria, de 16 dimensões, formando um vetor combinado de 80 dimensões para cada perfil microbiológico. Como pacientes podem ter números diferentes de perfis, é aplicada uma média mascarada: apenas os perfis válidos são somados, e o resultado é dividido pelo número de perfis válidos do paciente. O vetor agregado resultante é então projetado por uma MLP final para uma representação de 128 dimensões, correspondente ao embedding final do paciente (Fig. 50).

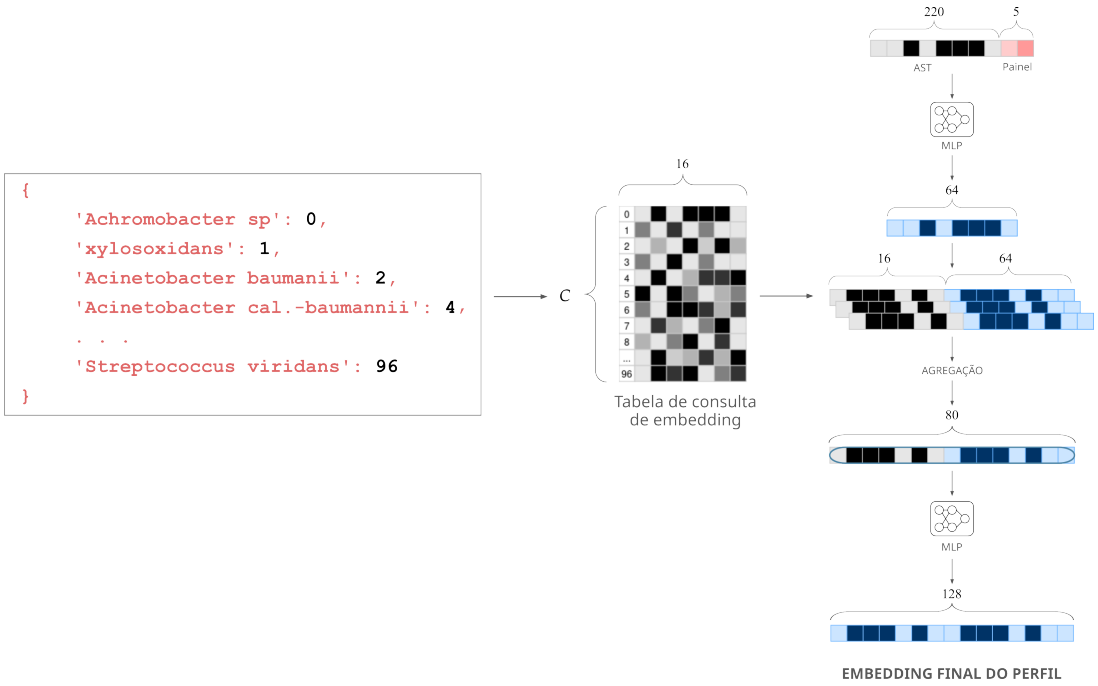
No segundo nível, os embeddings de 128 dimensões dos pacientes presentes em uma mesma enfermaria são agregados por *mean pooling*. Essa operação foi adotada porque trata os pacientes da enfermaria como um conjunto não ordenado, produzindo uma representação estável da composição microbiológica média da sala e evitando que o vetor agregado seja dominado por pacientes individuais ou pelo número absoluto de pacientes. As representações agregadas são então concatenadas com duas características escalares no nível do nó: número de pacientes presentes e razão de ocupação da enfermaria. Finalmente, o vetor resultante é projetado para uma dimensão latente fixa de 128, usada como dimensão de entrada dos modelos posteriores. Essa escolha busca equilibrar capacidade representacional e controle da complexidade paramétrica, seguindo a prática de representar atributos de nós em espaços latentes densos e compactos em modelos de aprendizado sobre grafos (HAMILTON; YING; LESKOVEC, 2017) (Fig. 51).

Uma instância de rede neural baseada em grafos como Graph Sample and Aggregate (GraphSAGE) (HAMILTON; YING; LESKOVEC, 2017) é utilizada para codificar a estrutura espacial do hospital:

$$Z^{(t)} = \mathrm{GNN}\Big(A^{(t)}, X^{(t)}\Big), \tag{5.3}$$

produzindo *embeddings* de nós $Z^{(t)} \in \mathbb{R}^{N \times \ell}$, em que $\mathbf{z}_j^{(t)}$ é a representação da sala j no

Figura 50 – As bactérias identificadas no estudo são codificadas como *embeddings* de tamanho 16 dentro de uma tabela de consulta; este vetor depois é concatenado ao *embedding* final do perfil do paciente.



Fonte: Elaborado pelo autor.

evento t e ℓ tem um tamanho de 128 dimensões.

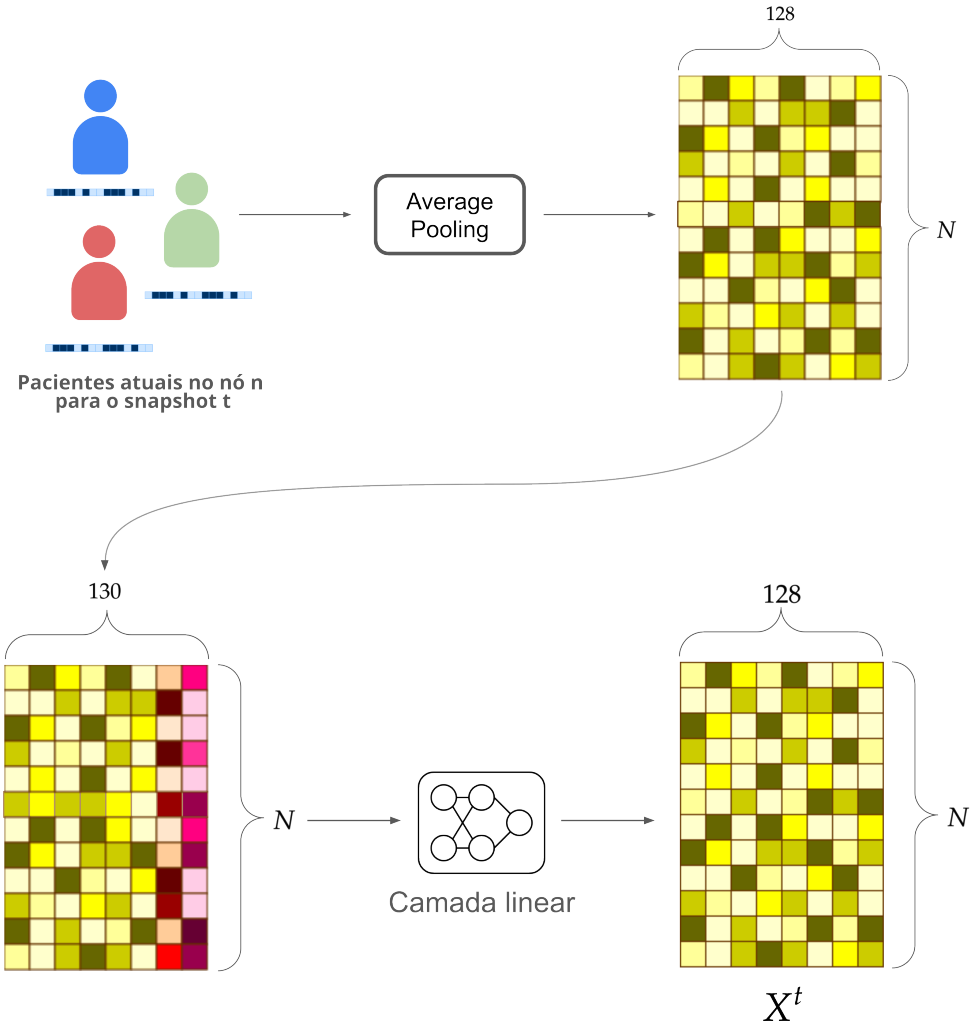
Este operador GNN é uma instância concreta do modelo (MPNN) descrito na Equação 4.4. Especificamente, para cada grafo, representações de nó são atualizadas por meio de um número fixo de camadas de passagem de mensagens, onde informação das salas vizinhas é agregada e combinada com a atual representação do nó via funções de atualização diferenciáveis. Assim, o mapeamento GNN$(\cdot)$ encapsula a aplicação dos operadores *ATUALIZAR* e *AGREGAR* definidos na formulação da MPNN, produzindo os *embeddings* espaciais finais para todas as salas no *snapshot* t (Fig. 52).

Representações temporais são modeladas de forma independente para cada sala usando alguma rede neural recorrente como *Gated Recurrent Unit* (GRU) (CHO *et al.*, 2014). Para cada sala j, o codificador temporal recebe a sequência de *embeddings* espaciais $\{\mathbf{z}_j^{(1)}, \ldots, \mathbf{z}_j^{(t)}\}$ e obtém como saída

$$\mathbf{h}_j^{(t)} = F\left(\mathbf{z}_j^{(1)}, \ldots, \mathbf{z}_j^{(t)}\right), \tag{5.4}$$

em que F denota a RNN $\mathbf{h}_j^{(t)} \in \mathbb{R}^{d_h}$ é o resultado da representação temporal (Fig. 53). Estas representações para cada nó passam por uma última camada linear que mapeia as representações para valores reais independentes por classe bacteriana que depois são transformados para probabilidades multirrótulo por meio de uma função sigmoide (Fig. 54).

Figura 51 – Sequência de agregação de *embeddings* de perfis dos pacientes pertencentes à mesma enfermaria. Como resultado, é obtido um vetor de representação de tamanho 128 para cada enfermaria no *snapshot* t.



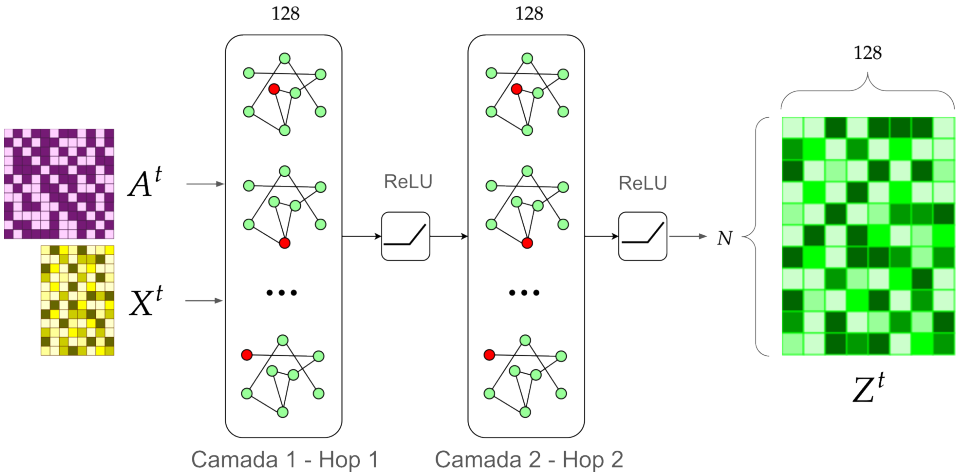
Fonte: Elaborado pelo autor.

Este modelo de duas etapas segue as formulações do modelo discreto espaço-temporal baseado em GNNs descrito por Skarding, Gabrys e Musial (2021). O Apêndice C apresenta o diagrama do modelo completo e os algoritmos do modelo estão descritos no Apêndice A.

5.5 Considerações éticas

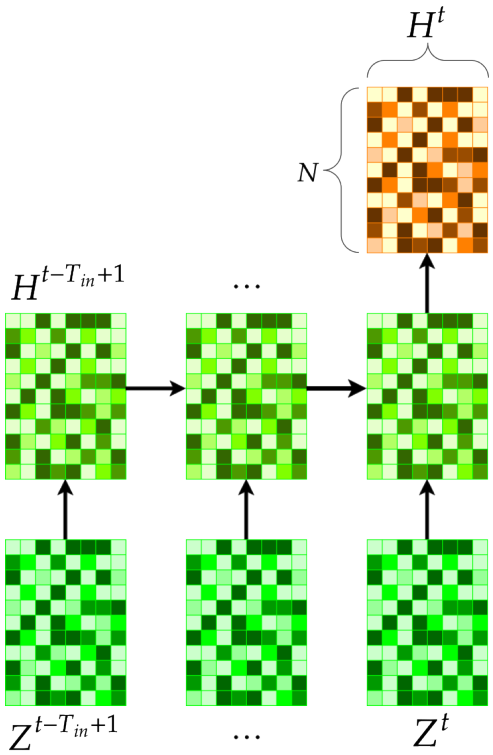
Trata-se de subprojeto do projeto principal "Contribuição ao Monitoramento e Controle de Microrganismos Multirresistentes em Hospital Universitário do Rio de Janeiro", aprovado pelo Comitê de Ética em Pesquisa com Seres Humanos do Hospital Universitário Pedro Ernesto, Universidade Estadual do Rio de Janeiro (UERJ), sob o número CAAE 71050523.5.0000.5259. O parecer consubstanciado correspondente é apresentado no Anexo A. Dessa forma a investigação espacial e temporal com aplicação

Figura 52 – O codificador espacial baseado em GNN é alimentado pela matriz do grafo $A^{(t)}$ e a matriz de características de nó $X^{(t)}$. Dado que o grafo não é muito complexo, apenas duas camadas de profundidade foram usadas para explorar a informação remota do grafo. Funções $ReLU$ foram consideradas para quebrar a linearidade entre camadas.



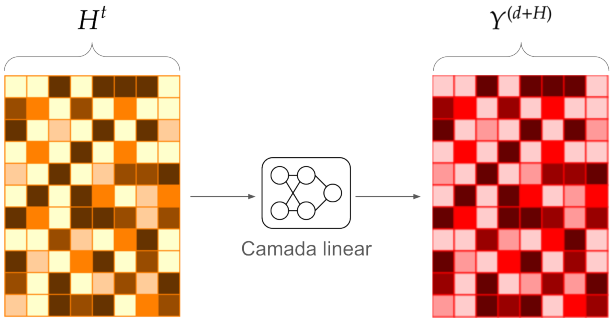
Fonte: Elaborado pelo autor.

Figura 53 – O codificador temporal gera representações temporais da sequência temporal estabelecida para a janela de entrada T_{in}.



Fonte: Elaborado pelo autor.

Figura 54 – O decodificador temporal compara as representações geradas a partir da janela temporal de entrada T_{in} contra o horizonte de previsão H .



Fonte: Elaborado pelo autor.

de inteligência artificial está aprovada.

6 Experimentos e resultados

Neste capítulo, são apresentadas a configuração dos experimentos e os principais resultados obtidos com os modelos propostos. Os experimentos foram organizados para avaliar o impacto de quatro fatores principais: a definição da vizinhança do grafo, o horizonte de previsão, a representação dos atributos no nível do paciente e da enfermaria, e a arquitetura preditiva utilizada. Foram avaliados horizontes de previsão de três, sete e quinze dias, considerando diferentes formas de construir as arestas entre enfermarias e diferentes combinações de atributos clínico-microbiológicos.

Além da linha de base por média histórica, foram comparadas arquiteturas espaciais, temporais e espaço-temporais, incluindo modelos baseados em GraphSAGE, GRU, atenção temporal e GAT. O objetivo é analisar em que medida a informação temporal, a estrutura espacial do hospital e a riqueza da representação dos atributos contribuem para a previsão da presença bacteriana futura em cada enfermaria.

6.1 Configuração do experimento

Para preservar a causalidade temporal, a sequência de grafos foi dividida cronologicamente: 70% dos grafos foram utilizados para treinamento, 15% para validação e 15% para teste. A construção das amostras seguiu uma estratégia de janela deslizante, na qual uma sequência de *snapshots* anteriores é utilizada como entrada T_{in} para prever a presença bacteriana em um horizonte futuro H. Foram avaliados três horizontes de previsão: $H = 3$, $H = 7$ e $H = 15$ dias.

A configuração experimental considerou três grupos principais de variáveis (Vizinhança do grafo, Horizonte de previsão e Atributos), resumidos na Tabela 6 e que foram avaliados em cinco arquiteturas principais, além de uma linha de base por média histórica. A Tabela 7 resume os modelos considerados e os grupos de variáveis. A combinação fatorial dessas variáveis resultou em $2 \times 3 \times 4 \times 5 = 120$ execuções experimentais correspondentes a vizinhança, horizonte, atributos e modelos respectivamente, desconsiderando a linha de base histórica, que foi calculada separadamente para cada configuração temporal relevante. Cada execução foi identificada por uma combinação única de arquitetura, vizinhança, horizonte de previsão e configuração de atributos. Os resultados completos das configurações experimentais, ordenados por horizonte de previsão, são apresentados no Anexo B.

A principal função de custo utilizada no treinamento foi a Entropia Cruzada Binária (*Binary Cross Entropy* – BCE) (GOODFELLOW; BENGIO; COURVILLE, 2016), adequada para problemas de classificação multirrótulo em que cada classe é tratada como

Tabela 6 – Variáveis consideradas na configuração experimental.

Grupo de variável	Configurações avaliadas
Vizinhança do grafo	**Espacial**: baseada apenas em contiguidade física entre enfermarias. **Serviço + espacial**: incorpora arestas espaciais, arestas por serviço e arestas classificadas simultaneamente como espaciais e de serviço.
Horizonte de previsão	Foram avaliados três horizontes futuros: 3, 7 e 15 dias.
Atributos	**Representação bacteriana + Enfermaria**: vetor da representação bacteriana no nível do paciente e vetor agregado no nível da enfermaria. **Representação bacteriana + Enfermaria + Escalares de enfermaria**: vetor da representação bacteriana no nível do paciente e vetor agregado da enfermaria concatenado com escalares da enfermaria. **Representação bacteriana + Painel de resumo AST + Enfermaria**: vetor da representação bacteriana concatenado com painel de resumo AST no nível do paciente e vetor agregado no nível da enfermaria. **Representação bacteriana + Painel de resumo AST + Enfermaria + Escalares de enfermaria**: vetor da representação bacteriana concatenado com painel de resumo AST no nível do paciente e vetor agregado da enfermaria concatenado com escalares da enfermaria.

Fonte: Elaborado pelo autor.

Tabela 7 – Arquiteturas avaliadas nos experimentos.

Modelo	Descrição
Média histórica	Linha de base que estima a probabilidade de presença bacteriana em cada enfermaria a partir da média empírica observada no período de treinamento.
GNN espacial	Modelo apenas espacial, baseado em GNN, que utiliza a estrutura do grafo em cada passo temporal.
GRU temporal	Modelo apenas temporal, baseado em GRU, que trata cada enfermaria como uma série temporal multivariada independente, sem passagem explícita de mensagens pelo grafo.
GNN+GRU	Modelo espaço-temporal que combina codificação espacial por GNN com modelagem temporal recorrente.
GNN+atenção temporal	Variação espaço-temporal que incorpora um mecanismo de atenção para o módulo temporal.
GAT+GRU	Variação espaço-temporal baseada em GAT para o módulo espacial, combinada com modelagem temporal recorrente.

Fonte: Elaborado pelo autor.

uma decisão binária independente. Para um conjunto de pares sala-classe bacteriana, a BCE é definida como, com base em (GOODFELLOW; BENGIO; COURVILLE, 2016):

$$\mathrm{BCE} = -\frac{1}{NB}\sum_{j=1}^{N}\sum_{b=1}^{B}\left[y_{j,b}\log(\hat{y}_{j,b}) + (1-y_{j,b})\log(1-\hat{y}_{j,b})\right], \tag{6.1}$$

em que N é o número de salas, B é o número de classes bacterianas, $y_{j,b} \in \{0,1\}$ indica a presença observada da classe bacteriana b na sala j, e $\hat{y}_{j,b} \in [0,1]$ é a probabilidade prevista pelo modelo. Valores menores de BCE indicam melhor ajuste entre as probabilidades previstas e os rótulos observados.

Essa função foi utilizada para guiar o treinamento dos modelos. Entretanto, a

avaliação final dos modelos não se restringiu à função de custo. Foram reportadas métricas complementares no conjunto de teste, incluindo AUPRC (*Area Under the Precision-Recall Curve*), *recall* e índice de Jaccard.

O *recall* mede a fração de ocorrências positivas corretamente identificadas pelo modelo, indicando sua capacidade de recuperar bactérias que de fato aparecem no horizonte de previsão e se encontra implicitamente definido na Equação 6.2 com base em (DAVIS; GOADRICH, 2006; SOKOLOVA; LAPALME, 2009).

$$P(\delta) = \frac{TP(\delta)}{TP(\delta) + FP(\delta)}, \qquad R(\delta) = \frac{TP(\delta)}{TP(\delta) + FN(\delta)}. \tag{6.2}$$

em que TP corresponde ao número de presenças bacterianas corretamente previstas, FP ao número de presenças previstas que não foram observadas, FN ao número de presenças observadas que não foram recuperadas pelo modelo e δ é o limiar que define o valor mínimo de probabilidade a partir do qual uma predição $\hat{y}_{j,b}$ é considerada positiva.

A AUPRC mede a qualidade do ranqueamento das predições ao longo de diferentes limiares de decisão, sendo especialmente informativa em cenários desbalanceados, nos quais as ocorrências positivas são relativamente raras. Para este caso, foi calculada tratando todos os pares (j, b) como uma única coleção de decisões binárias. Assim, as probabilidades previstas $\hat{y}_{j,b}$ são ranqueadas conjuntamente e a área sob a curva precisão-revocação é calculada sobre esse conjunto agregado.

Por fim, o índice de Jaccard avalia a similaridade entre o conjunto de bactérias previstas e o conjunto observado em cada enfermaria, penalizando tanto bactérias omitidas quanto predições excedentes. Para cada sala j, compara-se o conjunto de classes bacterianas observadas com o conjunto de classes previstas:

$$\mathcal{Y}_j = \{b : y_{j,b} = 1\}, \qquad \hat{\mathcal{Y}}_j = \{b : \tilde{y}_{j,b} = 1\}. \tag{6.3}$$

O Jaccard da sala j é dado por:

$$J_j = \frac{|\mathcal{Y}_j \cap \hat{\mathcal{Y}}_j|}{|\mathcal{Y}_j \cup \hat{\mathcal{Y}}_j|}. \tag{6.4}$$

A métrica final reportada corresponde à média do Jaccard sobre as enfermarias, com base em (JACCARD, 1901; SOKOLOVA; LAPALME, 2009):

$$\text{Jaccard}_{enfermaria} = \frac{1}{N} \sum_{j=1}^{N} J_j. \tag{6.5}$$

Os resultados de cada execução foram armazenados individualmente em diretórios de métricas específicos contendo um resumo dos tempos e métricas de execução de

Tabela 8 – Melhores configurações por horizonte de previsão no conjunto de teste

Horizonte	Melhor configuração	BCE	AUPRC	Recall	Jaccard
3 dias	**Vizinhança**: Serviço + espacial. **Atributos**: Todos. **Modelo**: Temporal.	0,0463	0,8433	0,7132	0,7375
7 dias	**Vizinhança**: Espacial. **Atributos**: Todos. **Modelo**: Temporal.	0,0568	0,7829	0,6551	0,6612
15 dias	**Vizinhança**: Serviço + espacial. **Atributos**: Todos. **Modelo**: Temporal.	0,0648	0,7485	0,6225	0,6124

cada uma das épocas para cada experimento e plots de desempenho para cada métrica. Adicionalmente, uma tabela consolidada em formato *Comma-Separated Values* (CSV) foi gerada para sumarizar as configurações experimentais, métricas finais de teste e tempo total de execução de cada modelo.

6.2 Resultados da previsão

Foi conduzido um experimento fatorial, considerando duas definições de vizinhança, três horizontes de previsão, quatro configurações de atributos e cinco arquiteturas de modelos. No total, foram avaliadas 120 configurações experimentais. Este conjunto de experimentos teve como objetivo comparar sistematicamente as escolhas de arquitetura, horizonte, vizinhança e representação dos atributos.

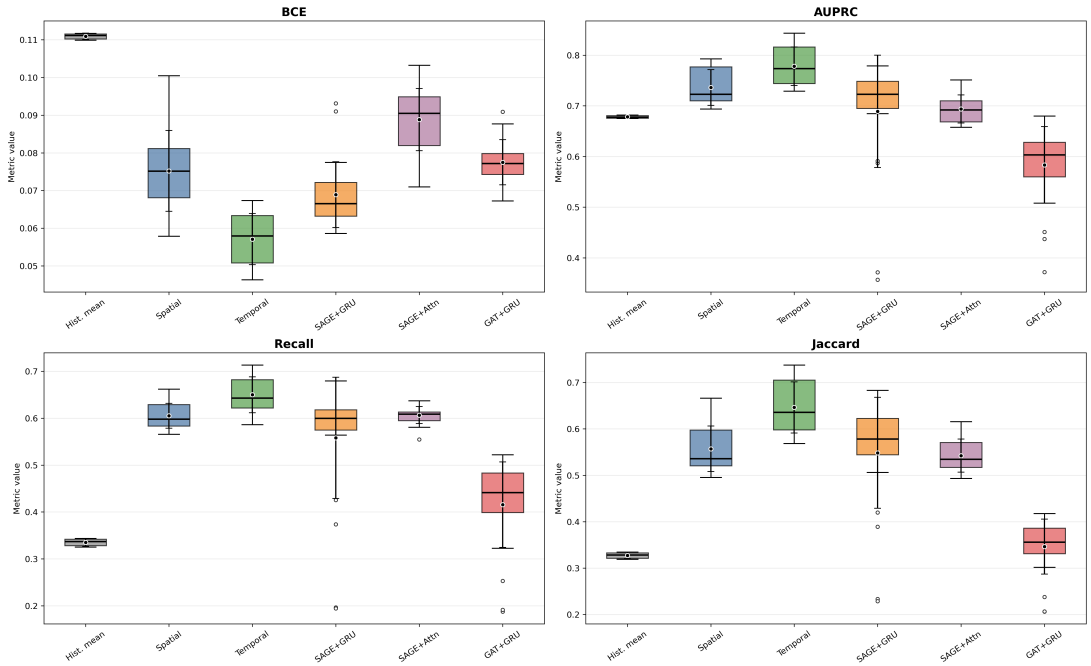
A Tabela 8 apresenta as melhores configurações observadas para cada horizonte de previsão no conjunto de teste. O modelo temporal baseado em GRU apresentou o melhor desempenho nos três horizontes, sugerindo que as regularidades temporais da presença de espécies bacterianas em nível de enfermaria possuem forte poder preditivo. O melhor resultado global foi obtido para o horizonte de três dias, usando vizinhança serviço + espacial e atributos da representação bacteriana + painel de resumo AST no nível do paciente combinados com atributos de enfermaria + escalares de enfermaria no nível da enfermaria.

Como esperado, o desempenho diminui à medida que o horizonte de previsão

aumenta. A AUPRC passa de 0,8433 no horizonte de três dias para 0,7829 em sete dias e 0,7485 em 15 dias. O mesmo padrão é observado no índice de Jaccard, que diminui de 0,7375 para 0,6612 e 0,6124, respectivamente. Esse comportamento reflete o aumento da incerteza em horizontes mais longos, nos quais admissões, altas, transferências e mudanças no estado microbiológico dos pacientes tornam-se mais difíceis de antecipar.

A Figura 55 resume o desempenho médio das arquiteturas considerando todas as combinações de vizinhança, horizonte e atributos. Em média, a GRU temporal apresentou o melhor desempenho entre os modelos avaliados, com menor BCE e maiores valores de AUPRC, revocação e Jaccard. O modelo apenas espacial obteve o segundo melhor AUPRC médio, indicando que a estrutura do grafo contém informação relevante. Entretanto, nas configurações avaliadas, essa informação espacial não foi suficiente para superar a modelagem temporal isolada.

Figura 55 – Desempenho médio por arquitetura considerando todas as configurações experimentais.



Fonte: Elaborado pelo autor.

Em relação às definições de vizinhança, a diferença média entre o grafo espacial e o grafo serviço + espacial foi pequena. A vizinhança serviço + espacial apresentou AUPRC médio de 0,6977, enquanto a vizinhança apenas espacial apresentou AUPRC médio de 0,6944. Portanto, neste conjunto de experimentos, a escolha da vizinhança teve impacto menor do que a escolha da arquitetura e do horizonte de previsão.

Quanto às configurações de atributos, a combinação de representação bacteriana + painel de resumo AST no nível do paciente com escalares no nível da enfermaria apresentou

o melhor desempenho médio, com AUPRC médio de 0,7102 e Jaccard médio de 0,5422. Esse resultado sugere que tanto o painel de resumo AST quanto os escalares agregados da enfermaria adicionam informação útil para a tarefa de previsão.

De forma geral, os resultados indicam que a tarefa é fortemente governada pela dinâmica temporal observada nas enfermarias. Embora a estrutura espacial do hospital contribua para o desempenho, os modelos que dependem exclusivamente ou principalmente da passagem de mensagens no grafo não superaram o modelo temporal. Assim, nas configurações avaliadas, a arquitetura temporal baseada em GRU com atributos AST + Painel de resumo AST e escalares da enfermaria constitui a principal configuração de referência.

6.3 Inconsistências nas movimentações de pacientes

Durante a construção da sequência de grafos, foi detectada uma inconsistência nos dados de movimentação de pacientes. À medida que a construção da sequência temporal progredia, a ocupação inferida em algumas enfermarias passava a exceder o número total de leitos disponíveis. Esse comportamento foi observado em várias enfermarias e aumentava de forma progressiva ao longo do tempo. Verificações automatizadas confirmaram essas divergências e demonstraram o surgimento de pacientes registrados simultaneamente em múltiplas enfermarias. Para mitigar esses problemas, foram removidas todas as transferências associadas a estados duplicados, eliminando aproximadamente 7,5% dos registros de movimentação.

Após a limpeza, foi detectada uma diminuição no número e na magnitude das violações de capacidade de leitos. Também foi conduzido um experimento com variáveis fixas, considerando vizinhança serviço+espacial, horizonte de previsão de sete dias ($H = 7$), janela de entrada de cinco eventos ($T_{in} = 5$) e atributos completos, para medir o impacto dessas inconsistências sobre o desempenho preditivo. A Tabela 9 compara os resultados obtidos com os dados originais e com os dados após a remoção de movimentações duplicadas.

Os resultados mostram que a limpeza melhora a consistência estrutural da sequência de grafos, mas seu impacto preditivo não é uniforme entre as arquiteturas. O modelo temporal baseado em GRU apresentou o ganho mais consistente após a limpeza, com redução do BCE de 0,0612 para 0,0554 e aumento simultâneo da AUPRC, recall e Jaccard. Por outro lado, alguns modelos apresentaram métricas superiores nos dados originais, especialmente o modelo espacial e a variação GAT+GRU. Assim, a remoção de duplicados é justificada principalmente pela correção da representação temporal da ocupação hospitalar, enquanto seu efeito nas métricas de previsão deve ser interpretado como dependente da arquitetura avaliada.

Tabela 9 – Resultados antes e depois da remoção de movimentações duplicadas de pacientes

Modelo	BCE		AUPRC		Recall		Jaccard	
	Original	**Limpo**	**Original**	**Limpo**	**Original**	**Limpo**	**Original**	**Limpo**
Espacial (GNN)	**0,0811**	0,0879	**0,7190**	0,7095	**0,6036**	0,5943	0,5277	**0,5428**
Temporal (GRU)	0,0612	**0,0554**	0,7536	**0,7871**	0,6490	**0,6657**	0,6388	**0,6610**
Espaço-temporal (GraphSAGE + GRU)	**0,0641**	0,0665	**0,7431**	0,7271	0,6047	**0,6169**	**0,5876**	0,5824
Espaço-temporal com atenção	**0,0888**	0,0898	**0,7037**	0,6870	**0,5866**	0,5864	0,4924	**0,4944**
Espaço-temporal (GAT + GRU)	**0,0776**	0,0785	**0,6203**	0,5904	**0,4732**	0,3918	**0,3697**	0,3311

Comparação no conjunto de teste entre a sequência original de grafos e a sequência após a remoção de movimentações duplicadas de pacientes. Em todas as execuções foram fixados vizinhança serviço+espacial, $H = 7$, $T_{in} = 5$ e atributos completos. Valores em negrito indicam o melhor resultado entre dados originais e limpos para cada modelo e métrica.

6.4 Discussão dos resultados

Os resultados indicam que a previsão da presença de espécies bacterianas em enfermarias é governada principalmente por regularidades temporais. A superioridade do modelo baseado em GRU sugere que persistência local, efeitos atrasados de presença bacteriana anterior e padrões operacionais recorrentes do hospital constituem o sinal mais informativo para antecipar a presença bacteriana futura. A queda de desempenho com o aumento do horizonte reforça essa interpretação: à medida que a previsão se afasta da janela observada, cresce a incerteza associada a admissões, altas, transferências entre enfermarias e mudanças no estado microbiológico dos pacientes.

A estrutura espacial do hospital contribuiu para a previsão, mas não foi o principal determinante do desempenho. O desempenho competitivo do modelo espacial indica que a conectividade entre enfermarias contém informação útil, provavelmente relacionada à proximidade física, vínculos operacionais e possibilidades de circulação de pacientes. Entretanto, a escolha específica entre vizinhança espacial e vizinhança serviço+espacial teve impacto médio reduzido, sugerindo que, neste conjunto de dados, arquitetura, horizonte de previsão e representação dos atributos foram fatores mais relevantes do que a definição das arestas fixas.

Em relação aos atributos, os resultados favorecem a combinação entre representação bacteriana, painel de resumo AST e escalares agregados da enfermaria. Essa configuração integra informação microbiológica no nível do paciente e contexto operacional no nível do nó, o que contribui para a construção de características mais informativas. Ainda assim, esse ganho depende da arquitetura: os atributos enriquecidos foram mais efetivos quando explorados por modelos capazes de capturar a dinâmica temporal.

As arquiteturas espaço-temporais não superaram a GRU temporal, apesar de incorporarem explicitamente a estrutura do grafo. Esse resultado sugere que o aumento da complexidade do módulo espacial não necessariamente se traduz em melhor capacidade

preditiva quando a topologia hospitalar é relativamente estática e os atributos explícitos das arestas são limitados. Em particular, mecanismos de atenção sobre vizinhos podem introduzir maior variância e risco de sobreajuste sem acrescentar informação proporcionalmente relevante ao sinal temporal dominante.

A análise das inconsistências nas movimentações de pacientes acrescenta uma dimensão importante à interpretação dos experimentos. A remoção de estados duplicados melhorou a coerência estrutural da sequência de grafos, reduzindo situações em que a ocupação inferida ultrapassava a capacidade de leitos ou em que pacientes apareciam simultaneamente em múltiplas enfermarias. Contudo, seu impacto preditivo não foi uniforme entre as arquiteturas. Assim, a limpeza deve ser entendida principalmente como uma correção estrutural necessária, cujo efeito nas métricas depende da forma como cada modelo utiliza a informação temporal e espacial.

No conjunto, os achados reforçam que a complexidade do modelo deve ser avaliada em relação ao sinal disponível nos dados. Neste cenário, modelos temporais mais simples foram mais efetivos do que arquiteturas espaço-temporais mais expressivas, enquanto a estrutura espacial e os atributos enriquecidos atuaram como fontes complementares de informação. Portanto, melhorias no pré-processamento, na consistência temporal e na representação dos atributos podem ser tão importantes quanto a introdução de componentes neurais mais complexos.

7 Conclusões, Limitações e Trabalhos Futuros

Este trabalho apresentou uma abordagem baseada em grafos dinâmicos para a previsão da presença futura de espécies bacterianas em nível de enfermaria em um hospital terciário. A partir de dados rotineiramente coletados de microbiologia, admissões, transferências, altas e óbitos, a estrutura física e operacional do hospital foi representada como uma sequência temporal de grafos, na qual os nós correspondem às enfermarias e as arestas representam relações de vizinhança espacial, de serviço e de movimentação de pacientes. A tarefa foi formulada como um problema de classificação multirrótulo em nível de nó, em que se busca estimar, para cada enfermaria, a probabilidade de presença de cada espécie bacteriana em horizontes futuros de três, sete e quinze dias. Nesta primeira abordagem, a análise concentrou-se na distribuição das espécies bacterianas detectadas, sem restringir a previsão a um perfil específico de susceptibilidade antimicrobiana.

Foram avaliadas 120 configurações experimentais, combinando duas definições de vizinhança do grafo, três horizontes de previsão, quatro representações de atributos e cinco arquiteturas preditivas, além de uma linha de base por média histórica. Os resultados indicam que a dinâmica temporal observada em cada enfermaria constitui o sinal preditivo dominante. O modelo temporal baseado em GRU obteve o melhor desempenho nos três horizontes avaliados, alcançando AUPRC de 0,8433 para três dias, 0,7829 para sete dias e 0,7485 para quinze dias. A redução progressiva do desempenho com o aumento do horizonte confirma que a incerteza associada a admissões, altas, transferências e mudanças microbiológicas dos pacientes cresce à medida que a previsão se distancia da janela observada.

A representação do hospital como grafo mostrou-se relevante, mas não foi o principal fator explicativo do desempenho. O modelo apenas espacial baseado em GraphSAGE apresentou o segundo melhor AUPRC médio entre as arquiteturas, sugerindo que a conectividade entre enfermarias contém informação útil sobre proximidade física e vínculos operacionais. Entretanto, os modelos espaço-temporais avaliados não superaram a GRU temporal, e a variação baseada em GAT+GRU apresentou desempenho inferior às demais arquiteturas em média. Esses resultados sugerem que, para a topologia hospitalar e o volume de dados considerados, o aumento da complexidade do componente espacial não se traduziu necessariamente em ganhos preditivos. A diferença média entre a vizinhança apenas espacial e a vizinhança serviço+espacial também foi pequena, indicando que arquitetura, horizonte de previsão e representação dos atributos tiveram maior impacto do que a escolha específica das arestas fixas.

Em relação aos atributos, a melhor configuração média combinou a representação

bacteriana no nível do paciente, o painel resumido de AST e os escalares agregados da enfermaria. Esse resultado reforça a importância de integrar informação microbiológica e contexto operacional da enfermaria na construção das características dos nós. A análise das inconsistências nas movimentações de pacientes também mostrou que a qualidade da sequência temporal é um aspecto crítico da modelagem. A remoção de aproximadamente 7,5% dos registros associados a estados duplicados reduziu violações de capacidade de leitos e tornou a representação da ocupação hospitalar mais coerente. Contudo, seu impacto nas métricas de previsão não foi uniforme entre as arquiteturas: a GRU temporal apresentou melhora consistente após a limpeza, enquanto alguns modelos espaciais obtiveram resultados superiores em parte das métricas com os dados originais. Assim, a limpeza deve ser entendida principalmente como uma correção estrutural necessária, cujo efeito preditivo depende do modelo utilizado.

À luz das hipóteses formuladas no Capítulo 5, os resultados sustentam parcialmente a primeira hipótese. A representação do hospital como grafo dinâmico permitiu organizar, codificar e analisar padrões espaço-temporais associados à presença de espécies bacterianas nas enfermarias, e o desempenho competitivo do modelo espacial indica que a conectividade hospitalar contém informação epidemiológica relevante. A segunda hipótese também foi sustentada parcialmente: embora os modelos baseados em grafos tenham superado a linha de base por média histórica em várias configurações, eles não apresentaram desempenho superior ao modelo neural temporal não baseado em grafos, representado pela GRU. Assim, os resultados indicam que, no conjunto de dados e nas configurações avaliadas, a informação temporal foi o componente preditivo dominante, enquanto a estrutura espacial atuou como fonte complementar de informação.

7.1 Limitações

Este trabalho apresenta algumas limitações que devem ser consideradas na interpretação dos resultados. Em primeiro lugar, os experimentos foram conduzidos com dados de um único hospital e de um período específico de coleta. Assim, embora os resultados indiquem potencial para a modelagem da presença bacteriana em ambiente intra-hospitalar, sua generalização para outros períodos, hospitais ou configurações assistenciais ainda precisa ser avaliada.

Em segundo lugar, a representação das interações hospitalares foi baseada principalmente em relações de vizinhança espacial, serviço e movimentação de pacientes. A topologia utilizada não incorporou pesos ou atributos explícitos associados à intensidade de contato, distância física detalhada, fluxo de profissionais, compartilhamento de equipamentos ou rotinas operacionais específicas. Portanto, parte das interações relevantes para a circulação de microrganismos pode não ter sido representada diretamente no grafo.

Outra limitação está relacionada à direção das arestas. O grafo foi tratado como não direcionado, o que é adequado para arestas estáticas de vizinhança espacial ou pertencimento ao mesmo serviço, pois essas relações tendem a ser simétricas. No entanto, essa suposição é menos apropriada para arestas dinâmicas de movimentação, nas quais a transferência de um paciente de uma enfermaria de origem para uma enfermaria de destino possui sentido definido. Ao ignorar essa direção, o modelo pode perder parte da assimetria associada ao fluxo hospitalar.

Também há limitações associadas à qualidade e à granularidade dos dados de movimentação. A necessidade de remover registros associados a estados duplicados indica que inconsistências operacionais podem afetar a construção da sequência dinâmica de grafos. Um dos principais desafios para a previsão está no ruído gerado pela reconstrução histórica das movimentações, especialmente quando processos internos do hospital, regras administrativas ou padrões de transferência não estão explicitamente registrados nos dados disponíveis. Além disso, nesta versão, a janela de entrada foi definida em termos de *snapshots* de eventos, enquanto o horizonte de previsão foi definido em dias calendário. Essa formulação preserva a ordem das movimentações observadas, mas não corresponde ainda a uma agregação diária completa dos estados hospitalares.

Adicionalmente, os escalares de enfermaria utilizados como atributos de nó não foram normalizados. Na prática, a ausência de normalização pode fazer com que variáveis com maior escala numérica dominem parcialmente a representação de entrada, afetando a estabilidade do treinamento e a contribuição relativa dos demais atributos. Do ponto de vista teórico, normalizar esses valores poderia tornar as magnitudes das características mais comparáveis, facilitar a otimização e reduzir a sensibilidade do modelo a diferenças absolutas entre enfermarias. Por outro lado, a normalização também pode alterar a interpretação direta de atributos operacionais, como número de pacientes ou razão de ocupação, exigindo cuidado na escolha da escala e na análise dos resultados.

7.2 Trabalhos futuros

Como trabalhos futuros, recomenda-se aprofundar a representação das interações hospitalares por meio de arestas ponderadas e atributos explícitos de contato, serviço, distância, fluxo de pacientes e circulação de profissionais. Essa extensão pode permitir que o grafo represente de forma mais fiel os mecanismos operacionais associados à circulação intra-hospitalar de espécies bacterianas. A inclusão de pesos nas arestas também abre espaço para o uso de técnicas de explicabilidade em inteligência artificial (XAI), voltadas a avaliar a influência de conexões específicas ou grupos de conexões sobre as predições do modelo.

Também é importante avançar na validação dos modelos com períodos adicionais

de coleta e diferentes hospitais. A avaliação em bases externas permitiria medir a robustez da abordagem, identificar limitações de generalização e distinguir padrões específicos da instituição estudada de padrões potencialmente transferíveis para outros contextos hospitalares.

Outra direção relevante é realizar análises por espécie bacteriana ou por perfil de resistência específico, incluindo MDR, XDR e PDR. Essa extensão permitiria aproximar a formulação preditiva de cenários de vigilância epidemiológica mais direcionados, nos quais determinados microrganismos ou perfis de resistência exigem monitoramento prioritário.

Do ponto de vista da modelagem do grafo, também seria relevante testar grafos completos, nos quais todas as enfermarias são conectadas entre si. Esse tipo de experimento permitiria avaliar se o modelo é capaz de revelar conexões indiretas não capturadas pelas definições atuais de proximidade física ou serviço, sugerindo relações latentes entre enfermarias que não aparecem explicitamente na topologia construída.

No componente neural, trabalhos futuros podem testar arquiteturas de GNN diferentes do GraphSAGE e avaliar diferentes configurações de profundidade no codificador espacial. A variação no número de camadas permitiria investigar até que ponto a incorporação de vizinhos mais distantes contribui para a previsão ou, ao contrário, introduz ruído e sobre suavização das representações dos nós.

Quanto à reconstrução da sequência temporal, é importante investigar políticas hospitalares internas relacionadas às movimentações de pacientes que não estejam sendo consideradas no modelo atual. Incorporar essas regras pode reduzir o ruído na reconstrução histórica dos eventos e melhorar a coerência dos *snapshots*. Além disso, obter a hora exata das transferências durante a coleta dos dados poderia permitir agrupamentos de eventos mais acurados, reduzindo ambiguidades na ordenação intra-diária das movimentações.

Do ponto de vista operacional, a abordagem pode ser estendida para detecção precoce de surtos e agrupamentos, visualização interativa da propagação de risco entre enfermarias e atualização online à medida que novas transferências e resultados microbiológicos forem registrados. Também é importante melhorar a visualização e diferenciação entre movimentações, grafos diários e eventos, incorporando informações úteis no nível do nó, da aresta e do grafo. Por fim, a integração com verificações automatizadas de qualidade dos dados e com ferramentas de apoio à interpretação das predições pode tornar esse tipo de modelo mais útil para equipes de prevenção e controle de infecção hospitalar.

Referências

ADISASMITO, W. B.; ALMUHAIRI, S.; BEHRAVESH, C. B.; BILIVOGUI, P.; BUKACHI, S. A.; CASAS, N.; BECERRA, N. C.; CHARRON, D. F.; CHAUDHARY, A.; ZANELLA, J. R. C. *et al.* One health: A new definition for a sustainable and healthy future. *PLoS Pathogens*, Public Library of Science, v. 18, n. 6, p. e1010537, 2022. Citado na página 26.

AGARWAL, A.; KAPILA, K.; KUMAR, S. Whonet software for the surveillance of antimicrobial susceptibility. *Medical Journal Armed Forces India*, Elsevier, v. 65, n. 3, p. 264–266, 2009. Citado na página 22.

AHMED, A.; SHERVASHIDZE, N.; NARAYANAMURTHY, S.; JOSIFOVSKI, V.; SMOLA, A. J. Distributed large-scale natural graph factorization. In: *Proceedings of the 22nd International Conference on World Wide Web*. [S.l.: s.n.], 2013. p. 37–48. Citado 2 vezes nas páginas 57 e 58.

ANDERSON, R. M. Discussion: The Kermack-McKendrick epidemic threshold theorem. *Bulletin o Mathematical Biology*, v. 53, n. 1, p. 3–32, 1991. ISSN 0092-8240. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0092824005800394>. Citado na página 28.

ARON, J. L.; SCHWARTZ, I. B. Seasonality and period-doubling bifurcations in an epidemic model. *Journal of Theoretical Biology*, v. 110, n. 4, p. 665–679, 1984. ISSN 0022-5193. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0022519384801502>. Citado na página 28.

ATWOOD, J.; TOWSLEY, D. Diffusion-convolutional neural networks. *Advances in Neural Information Processing Systems*, v. 29, 2016. Citado 2 vezes nas páginas 16 e 69.

Australian Government. *OUTBREAK: A One Health real-time AMR surveillance/decision-support system*. 2022. Activity and Research Directory. Projeto liderado pela University of Technology Sydney. Disponível em: <https://www.amr.gov.au/activity-and-research-directory/outbreak-one-health-real-time-amr-surveillancedecision-support-system>. Acesso em: 3 ago. 2026. Citado na página 25.

BAHDANAU, D.; CHO, K.; BENGIO, Y. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2014. Disponível em: <https://api.semanticscholar.org/CorpusID:11212020>. Citado 2 vezes nas páginas 50 e 54.

BARABÁSI, A.-L.; ALBERT, R. Emergence of scaling in random networks. *science*, American Association for the Advancement of Science, v. 286, n. 5439, p. 509–512, 1999. Citado na página 30.

BEAUDOIN, M.; KABANZA, F.; NAULT, V.; VALIQUETTE, L. Evaluation of a machine learning capability for a clinical decision support system to enhance antimicrobial stewardship programs. *Artificial Intelligence in Medicine*, v. 68, p. 29–36, 2016.

ISSN 0933-3657. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0933365716300574>. Citado na página 25.

BONITA, R.; BEAGLEHOLE, R.; KJELLSTRÖM, T.; ORGANIZATION, W. H. *Basic Epidemiology*. World Health Organization, 2006. (Free Online Access: World Health Organization). ISBN 9789241547079. Disponível em: <https://books.google.com.br/books?id=AAZGobMNTXgC>. Citado 2 vezes nas páginas 26 e 27.

BOX, G. E.; JENKINS, G. M.; REINSEL, G. C.; LJUNG, G. M. *Time series analysis: forecasting and control*. [S.l.]: John Wiley & Sons, 2015. Citado na página 24.

BRUNA, J.; ZAREMBA, W.; SZLAM, A.; LECUN, Y. Spectral networks and locally connected networks on graphs. *CoRR*, abs/1312.6203, 2013. Disponível em: <https://api.semanticscholar.org/CorpusID:17682909>. Citado 2 vezes nas páginas 16 e 68.

CAI, H.; ZHENG, V. W.; CHANG, K. C.-C. A comprehensive survey of graph embedding: Problems, techniques, and applications. *IEEE Transactions on Knowledge and Data Engineering*, IEEE, v. 30, n. 9, p. 1616–1637, 2018. Citado 2 vezes nas páginas 56 e 62.

CAO, Q.; JIANG, R.; YANG, C.; FAN, Z.; SONG, X.; SHIBASAKI, R. Mepognn: Metapopulation epidemic forecasting with graph neural networks. In: SPRINGER. *Joint European conference on machine learning and knowledge discovery in databases*. [S.l.], 2022. p. 453–468. Citado na página 79.

CAO, S.; LU, W.; XU, Q. Grarep: Learning graph representations with global structural information. In: *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. [S.l.: s.n.], 2015. p. 891–900. Citado 2 vezes nas páginas 57 e 58.

CASEWELL, M.; PHILLIPS, I. Hands as route of transmission for klebsiella species. *British Medical Journal*, British Medical Journal Publishing Group, v. 2, n. 6098, p. 1315–1317, 1977. Citado na página 86.

CHAGAS, A. d. L. d.; ARAúJO, J. C. d. S.; SERRA, J. C. P.; ARAúJO, K. M. d.; CUNHA, M. d. O.; CORREIA, A. d. R.; GONçALVES, L. M. B.; CARNEIRO, L. C. Co-infection of sars-cov-2 and klebsiella pneumoniae: A systematic review and meta-analysis. *Diagnostics*, v. 14, n. 11, 2024. ISSN 2075-4418. Disponível em: <https://www.mdpi.com/2075-4418/14/11/1149>. Citado na página 17.

CHAPELLE, O.; SCHOLKOPF, B.; ZIEN, A. Semi-supervised learning. *IEEE Transactions on Neural Networks*, IEEE, v. 20, n. 3, p. 542–542, 2009. Citado na página 38.

CHO, K.; MERRIËNBOER, B. van; BAHDANAU, D.; BENGIO, Y. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. [S.l.: s.n.], 2014. p. 1724–1734. Citado 3 vezes nas páginas 48, 50 e 90.

CUN, Y. L.; JACKEL, L. D.; BOSER, B.; DENKER, J. S.; GRAF, H. P.; GUYON, I.; HENDERSON, D.; HOWARD, R. E.; HUBBARD, W. Handwritten digit recognition:

Applications of neural network chips and automatic learning. *IEEE Communications Magazine*, IEEE, v. 27, n. 11, p. 41–46, 1989. Citado na página 16.

DARKEN, C.; MOODY, J. Fast adaptive k-means clustering: some empirical results. In: IEEE. *1990 IJCNN International Joint Conference on Neural Networks*. [S.l.], 1990. p. 233–238. Citado na página 63.

DAVIS, J.; GOADRICH, M. H. The relationship between Precision-Recall and ROC curves. In: *Proceedings of the 23rd International Conference on Machine Learning*. New York, NY, USA: Association for Computing Machinery, 2006. (ICML '06), p. 233–240. Citado na página 96.

DECLICH, S.; CARTER, A. O. Public health surveillance: historical origins, methods and evaluation. *Bulletin of the World Health Organization*, v. 72, n. 2, p. 285, 1994. Citado na página 20.

DEFFERRARD, M.; BRESSON, X.; VANDERGHEYNST, P. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in Neural Information Processing Systems*, v. 29, 2016. Citado 2 vezes nas páginas 16 e 69.

DENG, S.; WANG, S.; RANGWALA, H.; WANG, L.; NING, Y. Cola-gnn: Cross-location attention based graph neural networks for long-term ili prediction. In: *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*. New York, NY, USA: Association for Computing Machinery, 2020. (CIKM '20), p. 245–254. ISBN 9781450368599. Disponível em: <https://doi.org/10.1145/3340531.3411975>. Citado 4 vezes nas páginas 29, 74, 75 e 76.

ECDC. Publications. *Surveillance of antimicrobial resistance in Europe, 2021 data: executive summary*. [S.l.]: European Centre for Disease Prevention and Control, 2022. 6 p. p. Citado 2 vezes nas páginas 17 e 20.

ERDOS, P.; RENYI, A. On random graphs I. *Publicationes Mathematicae Debrecen*, v. 6, n. 290-297, p. 18, 1959. Citado na página 30.

FERETZAKIS, G.; SAKAGIANNI, A.; LOUPELIS, E.; KALLES, D.; MARTSOUKOU, M.; SKARMOUTSOU, N.; CHRISTOPOULOS, C.; LADA, M.; VELENTZA, A.; PETROPOULOU, S.; MICHELIDOU, S.; KALDIS, V.; CHATZIKYRIAKOU, R.; DALAINAS, I.; DIMITRELLOS, E. Using machine learning to predict antimicrobial resistance of acinetobacter baumannii, klebsiella pneumoniae and pseudomonas aeruginosa strains. *Studies in Health Technology and Informatics*, v. 281, p. 43–47, 2021. Citado na página 25.

FOOMANI, F. H.; MIRZA, S.; MUKHIDA, S.; SRIRAM, K.; YU, Z.; GUPTA, A.; GOPALAKRISHNAN, S. Machine learning techniques to identify antibiotic resistance in patients diagnosed with various skin and soft tissue infections. *CoRR*, abs/2202.13496, 2022. Disponível em: <https://arxiv.org/abs/2202.13496>. Citado na página 25.

GARCIA-VIDAL, C.; PUERTA-ALCALDE, P.; CARDOZO, C.; ORELLANA, M. A.; BESANSON, G.; LAGUNAS, J.; MARCO, F.; RIO, A. D.; MARTÍNEZ, J. A.; CHUMBITA, M. *et al.* Machine learning to assess the risk of multidrug-resistant gram-negative bacilli infections in febrile neutropenic hematological patients. *Infectious Diseases and Therapy*, Springer, v. 10, p. 971–983, 2021. Citado na página 25.

GASPAR, G. G.; FERREIRA, L. R.; FELICIANO, C. S.; JÚNIOR, C. P. C.; MOLINA, F. M. R.; VENDRUSCOLO, A. C. S.; BRADAN, G. M. A.; LOPES, N. A. P.; MARTINEZ, R.; BOLLELA, V. R. Pre-and post-covid-19 evaluation of antimicrobial susceptibility for healthcare-associated infections in the intensive care unit of a tertiary hospital. *Revista da Sociedade Brasileira de Medicina Tropical*, SciELO Brasil, v. 54, p. e0090–2021, 2021. Citado na página 17.

GHARBI, M.; MOORE, L.; GILCHRIST, M.; THOMAS, C.; BAMFORD, K.; BRANNIGAN, E.; HOLMES, A. Forecasting carbapenem resistance from antimicrobial consumption surveillance: lessons learnt from an oxa-48-producing klebsiella pneumoniae outbreak in a west london renal unit. *International Journal of Antimicrobial Agents*, Elsevier, v. 46, n. 2, p. 150–156, 2015. Citado 2 vezes nas páginas 17 e 24.

GIACOBBE, D. R.; MORA, S.; GIACOMINI, M.; BASSETTI, M. Machine learning and multidrug-resistant gram-negative bacteria: An interesting combination for current and future research. *Antibiotics*, v. 9, n. 2, 2020. ISSN 2079-6382. Disponível em: <https://www.mdpi.com/2079-6382/9/2/54>. Citado 2 vezes nas páginas 17 e 25.

GILMER, J.; SCHOENHOLZ, S. S.; RILEY, P. F.; VINYALS, O.; DAHL, G. E. Neural message passing for quantum chemistry. In: *Proceedings of the 34th International Conference on Machine Learning - Volume 70*. [S.l.]: JMLR.org, 2017. (ICML'17), p. 1263–1272. Citado na página 70.

GOMES, M. Z. R.; LIMA, E. M. de; AIRES, C. A. M.; PEREIRA, P. S.; YIM, J.; SILVA, F. H.; RODRIGUES, C. A. S.; OLIVEIRA, T. R. T. e.; SILVA, P. P. da; ELLER, C. M. *et al.* Outbreak report of polymyxin-carbapenem-resistant klebsiella pneumoniae causing untreatable infections evidenced by synergy tests and bacterial genomes. *Scientific Reports*, Nature Publishing Group UK London, v. 13, n. 1, p. 6238, 2023. Citado 2 vezes nas páginas 17 e 81.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. MIT Press, 2016. (Adaptive Computation and Machine Learning series). ISBN 9780262035613. Disponível em: <https://books.google.com.br/books?id=Np9SDQAAQBAJ>. Citado 9 vezes nas páginas 16, 40, 41, 42, 43, 45, 46, 94 e 95.

GOODMAN, K. E.; LESSLER, J.; HARRIS, A. D.; MILSTONE, A. M.; TAMMA, P. D. A methodological comparison of risk scores versus decision trees for predicting drug-resistant infections: A case study using extended-spectrum beta-lactamase (esbl) bacteremia. *Infection Control and Hospital Epidemiology*, v. 40, n. 4, p. 400—407, Abril 2019. ISSN 0899-823X. Disponível em: <https://doi.org/10.1017/ice.2019.17>. Citado na página 25.

GOOGLE. *Community Mobility Reports*. 2022. Data retrieved from Google Community Mobility Reports, <https://www.google.com/covid19/mobility/>. Citado na página 76.

GORI, M.; MONFARDINI, G.; SCARSELLI, F. A new model for learning in graph domains. In: *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.* [S.l.: s.n.], 2005. v. 2, p. 729–734 vol. 2. Citado 2 vezes nas páginas 66 e 67.

GRAVES, A. *Supervised Sequence Labelling with Recurrent Neural Networks*. Springer Berlin Heidelberg, 2012. (Studies in Computational Intelligence). ISBN 9783642247965. Disponível em: <https://books.google.com.br/books?id=4UauNDGQWN4C>. Citado 5 vezes nas páginas 45, 46, 48, 49 e 50.

GROVER, A.; LESKOVEC, J. Node2vec: Scalable feature learning for networks. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: Association for Computing Machinery, 2016. (KDD '16), p. 855–864. ISBN 9781450342322. Disponível em: <https://doi.org/10.1145/2939672.2939754>. Citado 2 vezes nas páginas 60 e 61.

HAMILTON, W. *Graph Representation Learning*. Morgan & Claypool Publishers, 2020. (Synthesis Lectures on Artificial Intelligence and Machine Learning). ISBN 9781681739649. Disponível em: <https://books.google.com.br/books?id=Csj-DwAAQBAJ>. Citado 4 vezes nas páginas 35, 57, 62 e 70.

HAMILTON, W. L.; YING, R.; LESKOVEC, J. Inductive representation learning on large graphs. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc., 2017. (NIPS'17), p. 1025–1035. ISBN 9781510860964. Citado 5 vezes nas páginas 16, 64, 69, 88 e 89.

HAYKIN, S. *Neural Networks: A Comprehensive Foundation*. Macmillan, 1994. ISBN 9780023527616. Disponível em: <https://books.google.com.br/books?id=PSAPAQAAMAAJ>. Citado na página 39.

HAYKIN, S. *Neural Networks and Learning Machines*. Prentice Hall, 2009. (Neural networks and learning machines, v. 10). ISBN 9780131471399. Disponível em: <https://books.google.com.br/books?id=K7P36lKzI_QC>. Citado 2 vezes nas páginas 37 e 41.

HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. *Neural Computation*, MIT Press, Cambridge, MA, USA, v. 9, n. 8, p. 1735–1780, nov 1997. ISSN 0899-7667. Disponível em: <https://doi.org/10.1162/neco.1997.9.8.1735>. Citado 2 vezes nas páginas 47 e 48.

HUANG, S. S.; YOKOE, D. S.; STELLING, J.; PLACZEK, H.; KULLDORFF, M.; KLEINMAN, K.; O'BRIEN, T. F.; CALDERWOOD, M. S.; VOSTOK, J.; DUNN, J. *et al.* Automated detection of infectious disease outbreaks in hospitals: a retrospective cohort study. *PLoS Medicine*, Public Library of Science San Francisco, USA, v. 7, n. 2, p. e1000238, 2010. Citado na página 23.

HY, T. S.; NGUYEN, V. B.; TRAN-THANH, L.; KONDOR, R. Temporal multiresolution graph neural networks for epidemic prediction. In: XU, P.; ZHU, T.; ZHU, P.; CLIFTON, D. A.; BELGRAVE, D.; ZHANG, Y. (Ed.). *Proceedings of the 1st Workshop on Healthcare AI and COVID-19, ICML 2022*. PMLR, 2022. (Proceedings of Machine Learning Research, v. 184), p. 21–32. Disponível em: <https://proceedings.mlr.press/v184/hy22a.html>. Citado na página 79.

JACCARD, P. Étude comparative de la distribution florale dans une portion des alpes et du jura. *Bulletin de la Société Vaudoise des Sciences Naturelles*, v. 37, n. 142, p. 547–579, 1901. Citado na página 96.

KAMPEN, N. van. *Stochastic Processes in Physics and Chemistry*. North-Holland, 1981. (North-Holland personal library). ISBN 9780444862006. Disponível em: <https://book.google.com.br/books?id=SZjvAAAAMAAJ>. Citado 2 vezes nas páginas 28 e 29.

KAPOOR, A.; BEN, X.; LIU, L.; PEROZZI, B.; BARNES, M.; BLAIS, M.; O'BANION, S. Examining covid-19 forecasting using spatio-temporal gnns. In: *Proceedings of the 16th International Workshop on Mining and Learning with Graphs*. [S.l.: s.n.], 2020. Citado 3 vezes nas páginas 16, 76 e 77.

KATTIS, A. A.; HOLIDAY, A.; STOICA, A.-A.; KEVREKIDIS, I. G. Modeling epidemics on adaptively evolving networks: a data-mining perspective. *Virulence*, Taylor & Francis, v. 7, n. 2, p. 153–162, 2016. Citado na página 17.

KERMACK, W. O.; MCKENDRICK, A. G. A contribution to the mathematical theory of epidemics. *Proceedings of the Royal Society of London. Series A, Containing papers of a mathematical and physical character*, The Royal Society London, v. 115, n. 772, p. 700–721, 1927. Citado na página 30.

KIM, Y.; DENTON, C.; HOANG, L.; RUSH, A. M. Structured attention networks. In: *International Conference on Learning Representations*. [s.n.], 2017. Disponível em: <https://openreview.net/forum?id=HkE0Nvqlg>. Citado na página 50.

KIPF, T. N.; WELLING, M. Semi-supervised classification with graph convolutional networks. *CoRR*, abs/1609.02907, 2016. Disponível em: <http://arxiv.org/abs/1609.02907>. Citado 5 vezes nas páginas 16, 69, 71, 76 e 78.

KOREN, Y.; BELL, R.; VOLINSKY, C. Matrix factorization techniques for recommender systems. *Computer*, IEEE, v. 42, n. 8, p. 30–37, 2009. Citado 2 vezes nas páginas 57 e 64.

KULLDORFF, M. Prospective time periodic geographical disease surveillance using a scan statistic. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, Wiley Online Library, v. 164, n. 1, p. 61–72, 2001. Citado na página 22.

KULLDORFF, M. *SaTScanTM user guide*. [S.l.]: Boston, 2006. Citado na página 22.

KULLDORFF, M.; NAGARWALLA, N. Spatial disease clusters: detection and inference. *Statistics in Medicine*, Wiley Online Library, v. 14, n. 8, p. 799–810, 1995. Citado na página 22.

LEVIE, R.; MONTI, F.; BRESSON, X.; BRONSTEIN, M. M. Cayleynets: Graph convolutional neural networks with complex rational spectral filters. *IEEE Transactions on Signal Processing*, IEEE, v. 67, n. 1, p. 97–109, 2018. Citado 2 vezes nas páginas 16 e 69.

LI, J.; ZHU, J.; ZHANG, B. Discriminative deep random walk for network classification. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. [S.l.: s.n.], 2016. p. 1004–1013. Citado na página 63.

LIANG, F.; QIAN, C.; YU, W.; GRIFFITH, D. W.; GOLMIE, N. T. Survey of graph neural networks and applications. *Wireless Communications and Mobile Computing*, Wireless Communications and Mobile Computing, n. 2022, 2022-07-28 04:07:00 2022. Disponível em: <https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=934576>. Citado na página 16.

LIU, Z.; ZHOU, J. *Introduction to Graph Neural Networks*. Morgan & Claypool Publishers, 2020. (Synthesis Lectures on Artificial Intelligence and Machine Learning). ISBN 9781681737669. Disponível em: <https://books.google.com.br/books?id=XOfYDwAAQBAJ>. Citado 2 vezes nas páginas 68 e 69.

LIVERMORE, D. M. Fourteen years in resistance. *International Journal of Antimicrobial Agents*, v. 39, n. 4, p. 283–291, 2012. Citado na página 17.

LÓPEZ-LOZANO, J.; LAWES, T.; NEBOT, C.; BEYAERT, A.; BERTRAND, X.; HOCQUET, D.; ALDEYAB, M.; SCOTT, M.; CONLON-BINGHAM, G.; FARREN, D. Thresholds study group a nonlinear time-series analysis approach to identify thresholds in associations between population antibiotic use and rates of resistance. *Nature Microbiology*, v. 4, n. 7, p. 1160–72, 2019. Citado 2 vezes nas páginas 17 e 24.

LU, J.; CHEN, J.; LIU, C.; ZENG, Y.; SUN, Q.; LI, J.; SHEN, Z.; CHEN, S.; ZHANG, R. Identification of antibiotic resistance and virulence-encoding factors in klebsiella pneumoniae by raman spectroscopy and deep learning. *Microbial Biotechnology*, v. 15, 11 2021. Citado na página 25.

LUTZ, C. S.; HUYNH, M. P.; SCHROEDER, M.; ANYATONWU, S.; DAHLGREN, F. S.; DANYLUK, G.; FERNANDEZ, D.; GREENE, S. K.; KIPSHIDZE, N.; LIU, L. *et al.* Applying infectious disease forecasting to public health: a path forward using influenza forecasting examples. *BMC Public Health*, Springer, v. 19, n. 1, p. 1659, 2019. Citado na página 29.

MA, Y.; TANG, J. *Deep Learning on Graphs*. Cambridge University Press, 2021. ISBN 9781108831741. Disponível em: <https://books.google.com.br/books?id=Xfw7EAAAQBAJ>. Citado 5 vezes nas páginas 37, 43, 44, 47 e 56.

MACESIC, N.; WALK, O. J. B. D.; PE'ER, I.; TATONETTI, N. P.; PELEG, A. Y.; UHLEMANN, A.-C. Predicting phenotypic polymyxin resistance in klebsiella pneumoniae through machine learning analysis of genomic data. *mSystems*, v. 5, n. 3, p. e00656–19, 2020. Disponível em: <https://journals.asm.org/doi/abs/10.1128/mSystems.00656-19>. Citado na página 25.

MAGIORAKOS, A.-P.; SRINIVASAN, A.; CAREY, R.; CARMELI, Y.; FALAGAS, M.; GISKE, C.; HARBARTH, S.; HINDLER, J.; KAHLMETER, G.; OLSSON-LILJEQUIST, B.; PATERSON, D.; RICE, L.; STELLING, J.; STRUELENS, M.; VATOPOULOS, A.; WEBER, J.; MONNET, D. Multidrug-resistant, extensively drug-resistant and pandrug-resistant bacteria: an international expert proposal for interim standard definitions for acquired resistance. *Clinical Microbiology and Infection*, v. 18, n. 3, p. 268–281, 2012. ISSN 1198-743X. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1198743X14616323>. Citado na página 21.

MARON, H.; BEN-HAMU, H.; SHAMIR, N.; LIPMAN, Y. Invariant and equivariant graph networks. In: *7th International Conference on Learning Representations*. OpenReview.net, 2019. Disponível em: <https://openreview.net/forum?id=Syx72jC9tm>. Citado na página 71.

MENEZES, M. *Detecção de bactérias resistentes a antibióticos triplicou na pandemia*. 2021. <https://portal.fiocruz.br/noticia/deteccao-de-bacterias-resistentes-antibioticos-triplicou-na-pandemia>. Citado na página 17.

META. *Facebook Movement Range Maps*. 2023. Data retrieved from Meta, <https://dataforgood.facebook.com/dfg/tools/movement-range-maps>. Citado na página 76.

MEŽNAR, S.; LAVRAČ, N.; ŠKRLJ, B. Prediction of the effects of epidemic spreading with graph neural networks. In: BENITO, R. M.; CHERIFI, C.; CHERIFI, H.; MORO, E.; ROCHA, L. M.; SALES-PARDO, M. (Ed.). *Complex Networks & Their Applications IX*. Cham: Springer International Publishing, 2021. p. 420–431. ISBN 978-3-030-65347-7. Citado na página 80.

MICHELI, A.; SONA, D.; SPERDUTI, A. Contextual processing of structured data by recursive cascade correlation. *IEEE Transactions on Neural Networks*, IEEE, v. 15, n. 6, p. 1396–1410, 2004. Citado 2 vezes nas páginas 66 e 67.

MIEGHEM, P. V.; SAHNEH, F. D.; SCOGLIO, C. M. An upper bound for the epidemic threshold in exact markovian SIR and SIS epidemics on networks. In: *Proceedings of the 53rd IEEE Conference on Decision and Control*. [S.l.]: IEEE, 2014. p. 6228–6233. Citado na página 30.

MIRHOSEINI, S. H.; NIKAEEN, M.; SHAMSIZADEH, Z.; KHANAHMAD, H. Hospital air: A potential route for transmission of infections caused by β-lactam–resistant bacteria. *American Journal of Infection Control*, Elsevier, v. 44, n. 8, p. 898–904, 2016. Citado na página 86.

MORENO, Y.; PASTOR-SATORRAS, R.; VESPIGNANI, A. Epidemic outbreaks in complex heterogeneous networks. *The European Physical Journal B-Condensed Matter and Complex Systems*, Springer, v. 26, p. 521–529, 2002. Citado na página 29.

MORSE, S. S. Factors in the emergence of infectious diseases. *Emerging Infectious Diseases*, v. 1, n. 1, p. 7, 1995. Citado na página 27.

NAGARWALLA, N. A scan statistic with a variable window. *Statistics in Medicine*, Wiley Online Library, v. 15, n. 7-9, p. 845–850, 1996. Citado na página 22.

NEWMAN, M. *Networks*. [S.l.]: Oxford University Press, 2018. Citado 2 vezes nas páginas 35 e 36.

NIEPERT, M.; AHMED, M.; KUTZKOV, K. Learning convolutional neural networks for graphs. In: PMLR. *International Conference on Machine Learning*. [S.l.], 2016. p. 2014–2023. Citado 2 vezes nas páginas 16 e 69.

OU, M.; CUI, P.; PEI, J.; ZHANG, Z.; ZHU, W. Asymmetric transitivity preserving graph embedding. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. [S.l.: s.n.], 2016. p. 1105–1114. Citado 2 vezes nas páginas 57 e 58.

PASTOR-SATORRAS, R.; CASTELLANO, C.; MIEGHEM, P. V.; VESPIGNANI, A. Epidemic processes in complex networks. *Reviews of Modern Physics*, American Physical Society (APS), v. 87, n. 3, p. 925–979, Agosto 2015. Disponível em: <https://doi.org/10.1103/2Frevmodphys.87.925>. Citado 2 vezes nas páginas 28 e 29.

PASTOR-SATORRAS, R.; VESPIGNANI, A. Epidemic spreading in scale-free networks. *Physical Review Letters*, American Physical Society, v. 86, n. 14, p. 3200, 2001. Citado na página 29.

PEI, S.; BLUMBERG, S.; VEGA, J. C.; ROBIN, T.; ZHANG, Y.; MEDFORD, R. J.; ADHIKARI, B.; SHAMAN, J. *et al.* Challenges in forecasting antimicrobial resistance. *Emerging Infectious Diseases*, Centers for Disease Control and Prevention, v. 29, n. 4, p. 679, 2023. Citado 2 vezes nas páginas 24 e 25.

PEROZZI, B.; AL-RFOU, R.; SKIENA, S. Deepwalk: Online learning of social representations. In: *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: Association for Computing Machinery, 2014. (KDD '14), p. 701–710. ISBN 9781450329569. Disponível em: <https://doi.org/10.1145/2623330.2623732>. Citado 3 vezes nas páginas 59, 60 e 61.

RATTANAUMPAWAN, P.; CHOORAT, C.; TAKONKITSAKUL, K.; TANGKOSKUL, T.; SEENAMA, C.; THAMLIKITKUL, V. A prospective surveillance study for multidrug-resistant bacteria colonization in hospitalized patients at a thai university hospital. *Antimicrobial Resistance & Infection Control*, BioMed Central, v. 7, n. 1, p. 1–11, 2018. Citado na página 24.

REICH, N. G.; BROOKS, L. C.; FOX, S. J.; KANDULA, S.; MCGOWAN, C. J.; MOORE, E.; OSTHUS, D.; RAY, E. L.; TUSHAR, A.; YAMANA, T. K. *et al.* A collaborative multiyear, multimodel assessment of seasonal influenza forecasting in the united states. *Proceedings of the National Academy of Sciences*, National Academy of Sciences, v. 116, n. 8, p. 3146–3154, 2019. Citado na página 29.

ROSENBLATT, F. Principles of neurodynamics. *Perceptrons and the theory of brain mechanisms*, Spartan Press, 1962. Citado na página 16.

RYU, S.; COWLING, B. J.; WU, P.; OLESEN, S.; FRASER, C.; SUN, D. S.; LIPSITCH, M.; GRAD, Y. H. Case-based surveillance of antimicrobial resistance with full susceptibility profiles. *JAC-Antimicrobial Resistance*, Oxford University Press, v. 1, n. 3, 2019. Citado na página 23.

SCARSELLI, F.; GORI, M.; TSOI, A. C.; HAGENBUCHNER, M.; MONFARDINI, G. The graph neural network model. *IEEE Transactions on Neural Networks*, v. 20, n. 1, p. 61–80, 2009. Citado 2 vezes nas páginas 66 e 67.

SHAH, C.; DEHMAMY, N.; PERRA, N.; CHINAZZI, M.; BARABÁSI, A.-L.; VESPIGNANI, A.; YU, R. Finding patient zero: Learning contagion source with graph neural networks. *arXiv preprint arXiv:2006.11913*, 2020. Disponível em: <https://arxiv.org/abs/2006.11913>. Citado na página 80.

SHUMAN, D. I.; NARANG, S. K.; FROSSARD, P.; ORTEGA, A.; VANDERGHEYNST, P. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Processing Magazine*, IEEE, v. 30, n. 3, p. 83–98, 2013. Citado na página 68.

SILVA, P. P. da; SILVA, F. A. da; RODRIGUES, C. A. S.; SOUZA, L. P.; LIMA, E. M. de; PEREIRA, M. H. B.; CANDELLA, C. N.; ALVES, M. Z. de O.; LOURENÇO, N. D.; TASSINARI, W. S. *et al.* Geographical information system and spatial–temporal statistics for monitoring infectious agents in hospital: a model using klebsiella pneumoniae complex. *Antimicrobial Resistance & Infection Control*, BioMed Central, v. 10, n. 1, p. 1–12, 2021. Citado 4 vezes nas páginas 17, 24, 81 e 86.

SKARDING, J.; GABRYS, B.; MUSIAL, K. Foundations and modeling of dynamic networks using dynamic graph neural networks: A survey. *IEEE Access*, IEEE, v. 9, p. 79143–79168, 2021. Citado 2 vezes nas páginas 88 e 91.

SNITKIN, E. S.; ZELAZNY, A. M.; THOMAS, P. J.; STOCK, F.; PROGRAM, N. C. S.; HENDERSON, D. K.; PALMORE, T. N.; SEGRE, J. A. Tracking a hospital outbreak of carbapenem-resistant klebsiella pneumoniae with whole-genome sequencing. *Science Translational Medicine*, American Association for the Advancement of Science, v. 4, n. 148, p. 116–148, 2012. Citado 2 vezes nas páginas 23 e 86.

SOKOLOVA, M.; LAPALME, G. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, v. 45, n. 4, p. 427–437, 2009. Citado na página 96.

SPERDUTI, A.; STARITA, A. Supervised neural networks for the classification of structures. *IEEE Transactions on Neural Networks*, IEEE, v. 8, n. 3, p. 714–735, 1997. Citado 2 vezes nas páginas 66 e 67.

STOKES, J. M.; YANG, K.; SWANSON, K.; JIN, W.; CUBILLOS-RUIZ, A.; DONGHIA, N. M.; MACNAIR, C. R.; FRENCH, S.; CARFRAE, L. A.; BLOOM-ACKERMANN, Z. *et al.* A deep learning approach to antibiotic discovery. *Cell*, Elsevier, v. 180, n. 4, p. 688–702, 2020. Citado na página 17.

SUTTON, R. S.; BARTO, A. G. *Reinforcement learning: An introduction*. [S.l.]: MIT press, 2018. Citado na página 38.

Söylemez, G.; YOUSEF, M.; KESMEN, Z.; BüYüKKIRAZ, M. E.; BAKIR-GUNGOR, B. Prediction of linear cationic antimicrobial peptides active against gram-negative and gram-positive bacteria based on machine learning models. *Applied Sciences*, v. 12, n. 7, 2022. ISSN 2076-3417. Disponível em: <https://www.mdpi.com/2076-3417/12/7/3631>. Citado na página 25.

TACCONELLI, E.; GóRSKA, A.; De Angelis, G.; LAMMENS, C.; RESTUCCIA, G.; SCHRENZEL, J.; HUSON, D.; CAREVIć, B.; PREOţESCU, L.; CARMELI, Y.; KAZMA, M.; SPANU, T.; CARRARA, E.; MALHOTRA-KUMAR, S.; GLADSTONE, B. Estimating the association between antibiotic exposure and colonization with extended-spectrum β-lactamase-producing gram-negative bacteria using machine learning methods: a multicentre, prospective cohort study. *Clinical Microbiology and Infection*, v. 26, n. 1, p. 87–94, 2020. ISSN 1198-743X. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1198743X19302472>. Citado na página 25.

The New York Times. *Coronavirus (COVID-19) Data in the United States*. The New York Times, 2021. Conjunto de dados baseado em relatórios de órgãos estaduais e locais de saúde dos Estados Unidos. Disponível em: <https://github.com/nytimes/covid-19-data>. Acesso em: 7 out. 2023. Citado na página 77.

TOMY, A.; RAZZANELLI, M.; LAURO, F. D.; RUS, D.; SANTINA, C. D. *Estimating the State of Epidemics Spreading with Graph Neural Networks*. arXiv, 2021. Disponível em: <https://arxiv.org/abs/2105.05060>. Citado 2 vezes nas páginas 78 e 79.

VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, Ł.; POLOSUKHIN, I. Attention is all you need. *Advances in Neural Information Processing Systems*, v. 30, 2017. Citado 5 vezes nas páginas 50, 51, 52, 53 e 54.

VELIČKOVIĆ, P.; CUCURULL, G.; CASANOVA, A.; ROMERO, A.; LIÒ, P.; BENGIO, Y. Graph attention networks. In: *International Conference on Learning Representations*. [s.n.], 2018. Disponível em: <https://openreview.net/forum?id=rJXMpikCZ>. Citado 2 vezes nas páginas 72 e 73.

VIBOUD, C.; BOëLLE, P.-Y.; CARRAT, F.; VALLERON, A.-J.; FLAHAULT, A. Prediction of the Spread of Influenza Epidemics by the Method of Analogues. *American Journal of Epidemiology*, v. 158, n. 10, p. 996–1006, 11 2003. ISSN 0002-9262. Disponível em: <https://doi.org/10.1093/aje/kwg239>. Citado na página 29.

WAIKHOM, L.; PATGIRI, R. Graph neural networks: Methods, applications, and opportunities. *arXiv preprint arXiv:2108.10733*, 2021. Citado na página 16.

WATTS, D. J.; STROGATZ, S. H. Collective dynamics of 'small-world'networks. *Nature*, Nature Publishing Group, v. 393, n. 6684, p. 440–442, 1998. Citado na página 30.

WERBOS, P. Beyond regression: New tools for prediction and analysis in the behavioral sciences. *PhD thesis, Committee on Applied Mathematics, Harvard University, Cambridge, MA*, 1974. Citado 2 vezes nas páginas 39 e 40.

WHO. *Antimicrobial resistance: global report on surveillance*. [S.l.]: World Health Organization, 2014. Citado 2 vezes nas páginas 17 e 22.

WHO. Global action plan on antimicrobial resistance. World Health Organization, 2015. Citado na página 21.

WHO. *Global antimicrobial resistance surveillance system: manual for early implementation*. [S.l.]: World Health Organization, 2015. Citado 2 vezes nas páginas 22 e 23.

WHO Collaborating Centre for Surveillance of Antimicrobial Resistance. *WHONET 2026: microbiology laboratory database software*. Boston: Brigham and Women's Hospital, 2026. Software para vigilância de infecções e resistência antimicrobiana. Disponível em: <https://whonet.org/software.html>. Acesso em: 3 ago. 2026. Citado na página 22.

WU, Z.; PAN, S.; CHEN, F.; LONG, G.; ZHANG, C.; PHILIP, S. Y. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, IEEE, v. 32, n. 1, p. 4–24, 2020. Citado na página 69.

XUE, J.; YABE, T.; TSUBOUCHI, K.; MA, J.; UKKUSURI, S. Multiwave covid-19 prediction from social awareness using web search and mobility data. In: *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: Association for Computing Machinery, 2022. (KDD '22), p. 4279–4289. ISBN 9781450393850. Disponível em: <https://doi.org/10.1145/3534678.3539172>. Citado na página 80.

XUE, L.; TANG, B.; CHEN, W.; LUO, J. Deept3: deep convolutional neural networks accurately identify gram-negative bacterial type iii secreted effectors using the n-terminal sequence. *Bioinformatics (Oxford, England)*, v. 35, 11 2018. Citado na página 25.

YANARDAG, P.; VISHWANATHAN, S. Deep graph kernels. In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. [S.l.: s.n.], 2015. p. 1365–1374. Citado na página 64.

YI, H.-C.; YOU, Z.-H.; HUANG, D.-S.; KWOH, C. K. Graph representation learning in bioinformatics: trends, methods and applications. *Briefings in Bioinformatics*, v. 23, n. 1, 09 2021. ISSN 1477-4054. Bbab340. Disponível em: <https://doi.org/10.1093/bib/bbab340>. Citado na página 66.

ZHANG, A.; LIPTON, Z. C.; LI, M.; SMOLA, A. J. *Dive into Deep Learning*. [S.l.]: Cambridge University Press, 2023. <https://D2L.ai>. Citado 2 vezes nas páginas 51 e 54.

ZHOU, J.; CUI, G.; HU, S.; ZHANG, Z.; YANG, C.; LIU, Z.; WANG, L.; LI, C.; SUN, M. Graph neural networks: A review of methods and applications. *AI Open*, Elsevier, v. 1, p. 57–81, 2020. Citado na página 16.

APÊNDICE A – Pseudocódigos dos algoritmos

Este apêndice apresenta os pseudocódigos dos principais procedimentos utilizados na construção da sequência dinâmica de grafos, na geração das representações, no treinamento dos modelos e no cálculo das métricas de avaliação.

Algoritmo 2: Esteira experimental do modelo proposto

Entrada: Dados de nós D_{nodes}, arestas D_{edges}, transferências $D_{\text{transfers}}$, perfis microbiológicos D_{profiles}, configurações de vizinhança $\mathcal{N}$, horizontes $\mathcal{H}$, variantes de modelo $\mathcal{M}$ e tamanho de janela T_{in}

Saída: Modelos treinados, resultados da linha de base, métricas de avaliação e resumos dos experimentos

$\mathcal{V}_{bact} \leftarrow$ ConstruirVocabulárioBacteriano(D_{profiles});

$\mathcal{A}_{ordem} \leftarrow$ DefinirOrdemDeAntibióticos(D_{profiles});

for *cada configuração de vizinhança* $n \in \mathcal{N}$ **do**

  $\mathcal{G} \leftarrow$ ConstruirSequênciaDinâmicaDeGrafos($D_{\text{nodes}}, D_{\text{edges}}, D_{\text{transfers}}, D_{\text{profiles}}, n$); `// usa o algoritmo 3`

  $(\mathcal{G}_{treino}, \mathcal{G}_{val}, \mathcal{G}_{teste}) \leftarrow$ DivisãoCronológica($\mathcal{G}$); `// divide a sequência em treino, validação e teste preservando a ordem temporal`

  for *cada horizonte* $H \in \mathcal{H}$ **do**

    $\mathcal{B} \leftarrow$ LinhaDeBaseMédiaHistórica($\mathcal{G}_{treino}, \mathcal{G}_{val}, \mathcal{G}_{teste}, T_{\text{in}}, H$); `// usa o algoritmo 6`

    for *cada variante de modelo* $m \in \mathcal{M}$ **do**

      $(M, \mathcal{R}) \leftarrow$ TreinarEAvaliarModelo($\mathcal{G}, m, T_{\text{in}}, H$); `// usa o algoritmo 7`

      ArmazenarResultados($n, H, m, \mathcal{B}, \mathcal{R}$);

    end

  end

end

Algoritmo 3: Construção da sequência dinâmica de grafos

Entrada: Dados de nós D_{nodes}, arestas D_{edges}, transferências $D_{\text{transfers}}$, perfis D_{profiles} e configuração de vizinhança n

Saída: Sequência dinâmica ordenada de grafos $\mathcal{G} = \{G_1, \ldots, G_T\}$

$G_0 \leftarrow$ ConstruirGrafoEstático$(D_{\text{nodes}}, D_{\text{edges}}, n)$;

$\mathcal{P} \leftarrow$ ConstruirPacientesEPerfis(D_{profiles}); `// associa cada paciente aos perfis microbiológicos disponíveis`

$D_{\text{transfers}} \leftarrow$ OrdenarTransferências$(D_{\text{transfers}})$; `// ordena os eventos de movimentação segundo a sequência temporal observada`

$D_{\text{transfers}} \leftarrow$ AtribuirOrdemIntraDia$(D_{\text{transfers}})$; `// define uma ordem interna para transferências registradas na mesma data`

$\mathcal{O} \leftarrow$ AgruparPorDataEOrdem$(D_{\text{transfers}})$;

$G \leftarrow G_0$;

$\mathcal{G} \leftarrow \emptyset$;

for *cada grupo ordenado de transferências* $o \in \mathcal{O}$ **do**

  AtualizarLocalizaçãoDosPacientes(G, o);

  for *cada paciente movimentado* p *em* o **do**

    $\mathbf{z}_p \leftarrow$ ConstruirRepresentaçãoDoPaciente$(p, o.data)$; `// usa o algoritmo 4`

    AtualizarEstadoDoNó$(G, p, \mathbf{z}_p)$; `// atualiza a enfermaria associada ao paciente movimentado com sua representação atual`

  end

  AtualizarOcupaçãoDosNós(G); `// recalcula a ocupação e os atributos dinâmicos das enfermarias`

  $G_t \leftarrow$ CopiarSnapshot(G);

  $\mathcal{G} \leftarrow \mathcal{G} \cup \{G_t\}$;

end

return $\mathcal{G}$;

Algoritmo 4: Construção da representação do paciente

Entrada: Perfis microbiológicos do paciente p disponíveis até o tempo t, vocabulário bacteriano $\mathcal{V}_{bact}$, ordem dos antibióticos $\mathcal{A}_{ordem}$ e características do paciente

Saída: Embedding do paciente $\mathbf{z}_p$

$\mathcal{Z}_p \leftarrow \emptyset$;

for *cada perfil microbiológico r do paciente p disponível até t* **do**

  $\mathbf{x}_r \leftarrow \textsc{CodificarFenótipoAST}(r, \mathcal{A}_{ordem})$;

  $b_r \leftarrow \textsc{CodificarBactéria}(r.\mathit{bactéria}, \mathcal{V}_{bact})$; `// mapeia a espécie bacteriana para seu identificador no vocabulário`

  $\mathbf{e}_r \leftarrow \textsc{CombinarFenótipoEBactéria}(\mathbf{x}_r, b_r)$; `// combina o vetor AST com o embedding treinável da bactéria`

  $\mathcal{Z}_p \leftarrow \mathcal{Z}_p \cup \{\mathbf{e}_r\}$;

end

$\mathbf{z}_p \leftarrow \textsc{AgregaçãoMascaradaPorMédia}(\mathcal{Z}_p)$; `// agrega apenas perfis válidos do paciente por média`

return $\mathbf{z}_p$;

Algoritmo 5: Construção da representação dos nós

Entrada: Snapshot do grafo G_t, características do paciente e características do nó

Saída: Matriz de características dos nós $\mathbf{X}t$

$\mathbf{X}t \leftarrow \emptyset$;

for *cada nó/enfermaria* $v \in V(G_t)$ **do**

  $\mathcal{Z}_v \leftarrow \emptyset$;

  for *cada paciente* p *localizado em* v **do**

    $\mathbf{z}_p \leftarrow$ ConstruirRepresentaçãoDoPaciente(p, t); `// usa o algoritmo 4`

    $\mathcal{Z}_v \leftarrow \mathcal{Z}_v \cup \{\mathbf{z}_p\}$;

  end

  $\mathbf{h}_v \leftarrow$ Média$(\mathcal{Z}_v)$;

  if *características escalares de nó estiverem habilitadas* **then**

    $\mathbf{s}_v \leftarrow [patient_count_v; occupancy_ratio_v]$; `// número de pacientes no nó e proporção de ocupação em relação à capacidade da enfermaria`

    $\mathbf{h}_v \leftarrow [\mathbf{h}_v; \mathbf{s}_v]$;

  end

  $\mathbf{x}_v \leftarrow$ ProjeçãoLinear$(\mathbf{h}_v)$; `// projeta a representação agregada do nó para a dimensão de entrada do modelo`

  $\mathbf{X}^t[v] \leftarrow \mathbf{x}_v$;

end

return $\mathbf{X}^t$;

Algoritmo 6: Linha de base por média histórica

Entrada: Sequências $\mathcal{G}_{treino}$, $\mathcal{G}_{val}$, $\mathcal{G}_{teste}$, tamanho de janela T_{in} e horizonte H

Saída: Métricas da linha de base na validação e no teste

$\mathcal{Y}_{treino} \leftarrow \emptyset$;

for *cada janela de treinamento* W *de tamanho* T_{in} **do**

  $G^{alvo} \leftarrow$ ObterGrafoAlvo(W, H);

  $\mathbf{Y}^{alvo} \leftarrow$ ExtrairPresençaBacteriana(G^{alvo});

  $\mathcal{Y}_{treino} \leftarrow \mathcal{Y}_{treino} \cup \{\mathbf{Y}^{alvo}\}$;

end

$\mathbf{P}_{hist} \leftarrow$ MédiaEmpírica$(\mathcal{Y}_{treino})$;

$\mathcal{R}_{val} \leftarrow$ CalcularMétricas$(\mathbf{P}_{hist}, \mathcal{G}_{val})$; `// usa o algoritmo 9`

$\mathcal{R}_{teste} \leftarrow$ CalcularMétricas$(\mathbf{P}_{hist}, \mathcal{G}_{teste})$; `// usa o algoritmo 9`

return $(\mathcal{R}_{val}, \mathcal{R}_{teste})$;

Algoritmo 7: Treinamento e avaliação do modelo

Entrada: Sequência de grafos $\mathcal{G}$, variante do modelo m, janela T_{in}, horizonte H, características do paciente e características do nó

Saída: Modelo treinado M e métricas finais de avaliação

$M \leftarrow \textsc{InicializarModelo}(m)$;

$(\mathcal{G}_{treino}, \mathcal{G}_{val}, \mathcal{G}_{teste}) \leftarrow \textsc{DivisãoCronológica}(\mathcal{G})$; `// divide a sequência em treino, validação e teste preservando a ordem temporal`

for *cada época* **do**

  for *cada janela de treinamento* $W = \{G_t, \ldots, G_{t+T_{\text{in}}-1}\}$ **do**

    for *cada snapshot* $G_\tau \in W$ **do**

      $\mathbf{X}_\tau \leftarrow \textsc{ConstruirRepresentaçãoDoNó}(G_\tau)$; `// usa o algoritmo 5`

    end

    $G^{alvo} \leftarrow \textsc{ObterGrafoAlvo}(W, H)$;

    $\mathbf{Y}^{alvo} \leftarrow \textsc{ExtrairPresençaBacteriana}(G^{alvo})$;

    $\mathbf{Z} \leftarrow \textsc{PassagemDireta}(M, \{\mathbf{X}_\tau\}, W)$; `// usa o algoritmo 8`

    $\mathcal{L} \leftarrow \text{BCEWithLogits}(\mathbf{Z}, \mathbf{Y}^{alvo})$;

    $M \leftarrow \textsc{AtualizarParâmetros}(M, \mathcal{L})$;

  end

  $\mathcal{R}_{val} \leftarrow \textsc{Avaliar}(M, \mathcal{G}_{val}, T_{\text{in}}, H)$; `// aplica a passagem direta do algoritmo 8 e calcula métricas pelo algoritmo 9`

end

$\mathcal{R}_{teste} \leftarrow \textsc{Avaliar}(M, \mathcal{G}_{teste}, T_{\text{in}}, H)$; `// aplica a passagem direta do algoritmo 8 e calcula métricas pelo algoritmo 9`

$\textsc{ArmazenarArtefatos}(M, \mathcal{R}_{val}, \mathcal{R}_{teste})$;

return $(M, \mathcal{R}_{teste})$;

Algoritmo 8: Passagem direta do modelo

Entrada: Modelo M, sequência de matrizes de características $\{\mathbf{X}_t, \ldots, \mathbf{X}_{t+T_{\text{in}}-1}\}$, estruturas de grafo correspondentes e variante do modelo

Saída: Logits $\mathbf{Z}$ para presença bacteriana por nó

if *M for espacial* **then**

  $\mathbf{H} \leftarrow$ CodificadorEspacial$(\mathbf{X}_{t+T_{\text{in}}-1}, G_{t+T_{\text{in}}-1})$; `// aplica passagem de mensagens para incorporar informação das enfermarias vizinhas`

end

else if *M for temporal* **then**

  $\mathbf{H} \leftarrow$ CodificadorTemporal$(\{\mathbf{X}_t, \ldots, \mathbf{X}_{t+T_{\text{in}}-1}\})$; `// usa GRU para modelar a evolução temporal das características dos nós`

end

else if *M for espaço-temporal* **then**

  for *cada passo* $\tau \in \{t, \ldots, t + T_{\text{in}} - 1\}$ **do**

    $\mathbf{S}_\tau \leftarrow$ CodificadorEspacial$(\mathbf{X}_\tau, G_\tau)$; `// aplica passagem de mensagens para incorporar informação das enfermarias vizinhas`

  end

  $\mathbf{H} \leftarrow$ CodificadorTemporal$(\{\mathbf{S}_t, \ldots, \mathbf{S}_{t+T_{\text{in}}-1}\})$; `// usa GRU para modelar a evolução temporal das representações espaciais`

end

$\mathbf{Z} \leftarrow$ CamadaLinearDePredição$(\mathbf{H})$;

return $\mathbf{Z}$;

Algoritmo 9: Cálculo das métricas

Entrada: Logits preditos $\mathbf{Z}$ e matriz alvo multilabel $\mathbf{Y}$

Saída: BCE, micro-AUPRC, micro-recall e Jaccard macro por nó

$\mathbf{Z} \leftarrow$ RemoverClasseDesconhecida$(\mathbf{Z})$; `// remove dos logits a classe reservada para bactérias ausentes ou não mapeadas`

$\mathbf{Y} \leftarrow$ RemoverClasseDesconhecida$(\mathbf{Y})$; `// remove dos alvos a classe reservada para bactérias ausentes ou não mapeadas`

BCE $\leftarrow$ BCEWithLogits$(\mathbf{Z}, \mathbf{Y})$;

$\mathbf{P} \leftarrow \sigma(\mathbf{Z})$;

$\text{AUPRC}_{micro} \leftarrow$ AUPRCMicro$(\mathbf{P}, \mathbf{Y})$;

$\text{Recall}_{micro} \leftarrow$ RecallMicro$(\mathbf{P}, \mathbf{Y}, 0.5)$;

$\text{Jaccard}_{ward} \leftarrow$ JaccardMacroPorNó$(\mathbf{P}, \mathbf{Y}, 0.5)$;

return $(\text{BCE}, \text{AUPRC}_{micro}, \text{Recall}_{micro}, \text{Jaccard}_{ward})$;

APÊNDICE B – Vocabulário bacteriano

Este apêndice apresenta o vocabulário bacteriano utilizado na codificação das classes de saída do modelo.

Tabela 10 – Vocabulário bacteriano utilizado no modelo

ID	**Classe bacteriana**	**ID**	**Classe bacteriana**	**ID**	**Classe bacteriana**
1	Achromobacter sp	33	Enterococcus sp	65	Pseudomonas putida
2	Achromobacter xylosoxidans	34	Escherichia coli	66	Pseudomonas sp
3	Acinetobacter baumanii	35	Fungo filamentoso	67	Pseudomonas stutzeri
4	Acinetobacter cal.-baumannii	36	Fusobacterium ulcerans	68	Raoultella spp.
5	Acinetobacter sp	37	Haemophilus sp	69	Salmonella sp
6	Aeromonas sp	38	Hafnia alvei	70	Serratia liquefaciens
7	Bacillus sp	39	Hafnia spp.	71	Serratia marcescens
8	Bacilo Gram-negativo não fermentador	40	Klebsiella aerogenes	72	Serratia sp
9	Bacilo Gram-negativo não fermentador (excluindo Acinetobacter, Complexo Burkholde)	41	Klebsiella oxytoca	73	Staphylococcus aureus
10	Bacilo Gram-positivo	42	Klebsiella pneumoniae	74	Staphylococcus aureus (MRSA)
11	Bacilos Gram negativos Anaeróbios	43	Klebsiella sp	75	Staphylococcus epidermidis
12	Bacteroides fragilis	44	Kocuria sp	76	Staphylococcus haemolyticus
13	Brevundimonas spp	45	Lactobacillus sp	77	Staphylococcus lugdunensis
14	Chryseobacterium sp	46	Leclercia adecarboxylata	78	Staphylococcus nao aureus
15	Citrobacter koseri	47	Leveduras	79	Staphylococcus saprophyticus

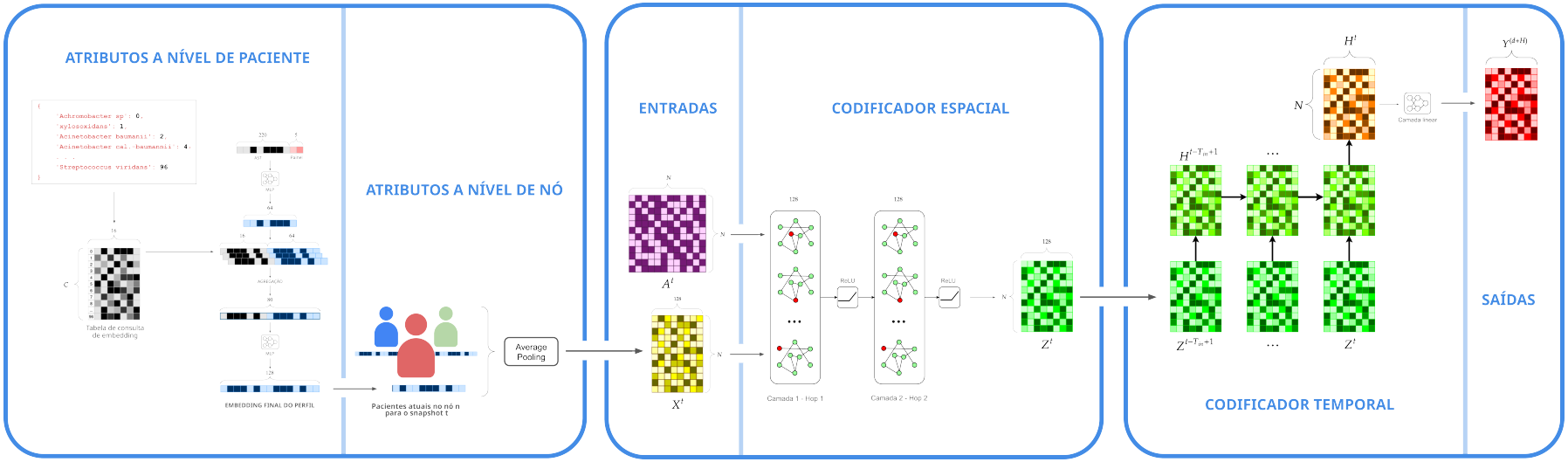
ID	Classe bacteriana	ID	Classe bacteriana	ID	Classe bacteriana
16	Citrobacter freundii	48	Listeria monocytogenes	80	Staphylococcus sp coagulase-negativa
17	Citrobacter sp	49	Micrococcus sp	81	Staphylococcus spp coagulase negativo (MRS)
18	Complexo Burkholderia cepacia	50	Morganella morganii	82	Staphylococcus spp. coagulase negativo, excluindo Staphylococcus saprophyticus
19	Complexo Enterobacter cloacae	51	Mycobacterium fortuitum	83	Stenotrophomonas maltophilia
20	Corynebacterium jeikeium	52	Neisseria meningitidis	84	Streptococcus agalactiae
21	Corynebacterium sp	53	Neisseria sp	85	Streptococcus anginosus
22	Cutibacterium acnes	54	Nocardia brasiliensis	86	Streptococcus dysgalactiae
23	Células leveduriformes	55	Pantoea sp	87	Streptococcus intermedius
24	Elizabethkingia spp.	56	Proteus mirabilis	88	Streptococcus pneumoniae
25	Enterobacter aerogenes	57	Proteus sp	89	Streptococcus pyogenes
26	Enterobacter sp	58	Proteus vulgaris	90	Streptococcus sanguinis
27	Enterobacterales	59	Providencia rettgeri	91	Streptococcus sp
28	Enterobacterales resistente a carbapenêmicos	60	Providencia sp	92	Streptococcus sp, alpha-hemolítico (grupo Viridans)
29	Enterococcus casseliflavus	61	Providencia stuartii	93	Streptococcus sp, beta-hemolítico Grupo C
30	Enterococcus faecalis	62	Pseudomonas aeruginosa	94	Streptococcus sp, beta-hemolítico não-Grupo A, B ou D
31	Enterococcus faecium	63	Pseudomonas aeruginosa mucóide	95	Streptococcus sp, beta-hemolítico não-Grupo A, B, C, D, F, G

ID	**Classe bacteriana**	**ID**	**Classe bacteriana**	**ID**	**Classe bacteriana**
32	Enterococcus faecium / faecalis resistente à vancomicina	64	Pseudomonas aeruginosa não mucóide	96	Streptococcus viridans

APÊNDICE C – Diagrama completo do modelo

Este apêndice apresenta o diagrama geral da arquitetura proposta, incluindo a construção das representações microbiológicas, a agregação no nível da enfermaria, os codificadores espacial e temporal e o mapeamento final para probabilidades multirrótulo.

Figura 56 – Diagrama completo do modelo proposto.



Fonte: Elaborado pelo autor.

ANEXO A – Parecer consubstanciado do Comitê de Ética

Este anexo apresenta o parecer consubstanciado do Comitê de Ética em Pesquisa referente ao projeto.

HOSPITAL UNIVERSITÁRIO PEDRO ERNESTO DA UNIVERSIDADE DO ESTADO DO RIO DE JANEIRO - HUUERJ

PARECER CONSUBSTANCIADO DO CEP

DADOS DO PROJETO DE PESQUISA

Título da Pesquisa: Contribuição ao Monitoramento e Controle de Microrganismos Multirresistentes em Hospital Universitário do Rio de Janeiro

Pesquisador: Marisa Zenaide Ribeiro Gomes

Área Temática:

Versão: 1

CAAE: 71050523.5.0000.5259

Instituição Proponente: Hospital Universitário Pedro Ernesto/UERJ

Patrocinador Principal: Financiamento Próprio

DADOS DO PARECER

Número do Parecer: 6.275.749

Apresentação do Projeto:

Transcrição editada do conteúdo registrado do protocolo "Nome do Arquivo: PB_INFORMAÇÕES_BÁSICAS_DO_PROJETO_2154957" e dos arquivos anexados à Plataforma Brasil.

Bactérias Gram-negativas (BGN) multirresistentes (MDR) aos antibióticos vêm se tornando um problema sério de saúde pública mundial. No Brasil,

esses microrganismos podem afetar diversos hospitais da região com significativa morbidade e mortalidade, como foi, por exemplo, o surto de

Klebsiella pneumoniae produtora de K. pneumoniae carbapenemase 2 (KPC-2) detectada no Distrito Federal em outubro de 2010 (Figura 1 e 2).

Cepas de BGN com suspeita de multirresistência detectadas nos serviços de saúde brasileiros são enviados para a Rede de Monitoramento de

Resistência Microbiana (RM) do Ministério da Sáude e ANVISA. Em um hospital da rede federal brasileira foi detectada uma ocorrência de K.

pneumoniae resistentes a todos os antibióticos testados em dezembro de 2014 e janeiro de 2015, dando origem a uma investigação que resultou em

uma nova metodologia de vigilância hospitalar espacial. Essa metodologia compreendeu o desenvolvimento de um Sistema de Informação

Geográfica (SIG) intrahospitalar em um hospital federal do Rio de Janeiro. Essa pesquisa permitiu

Endereço: Av. 28 de setembro, nº77 - CePeM - Centro de Pesquisa Clínica Multiusuário - 2º andar/sala nº 28 - prédio
Bairro: Vila Isabel **CEP:** 20.551-030
UF: RJ **Município:** RIO DE JANEIRO
Telefone: (21)2868-8253 **E-mail:** cep@hupe.uerj.br

ainda conhecer a história natural de infecções e
surtos hospitalares causado por bacteria, especificamente por K. pneumoniae MDR. A ocorrência crescente de BGN com perfil estensivamente
resistente (XDR) e panresistente (PDR) em hospitais ao redor do mundo, e a disponibilidade dessa nova ferramenta para o monitoramento espacial
por meio do SIG hospitalar desenvolvida pela nossa equipe, nos levou a propor nesse projeto a aplicação da metodologia espacial, em tempo real,
com a utilização de inteligência artificial, agora em uma instituição universitária de alta complexidade do Rio de Janeiro. Isso tudo visando contribuir,
no futuro, com as Comissões de Controle de Infecção Hospitalaro do Sistema Único de Saúde, para o monitoramento de diferentes espécies
bacterianas e perfis de susceptibilidade aos antimicrobianos, para o diagnóstico e controle precoces de casos e surtos hospitalares. Nesse estudo,
portanto, objetivamos dar continuidade a investigação espacial de infecções/colonizações por BGN-MDR que vierem a ocorrer em um hospital
brasileiro, no período de 2022 a 2028. Pretendemos correlacionar a caracterização molecular e fatores de virulência da(s) espécie(s), com fatores de
risco dos pacientes e do ambiente hospitalar, durante surtos ou períodos inter-epidêmicos. Pretendemos com isso determinar se há fatores
individuais clínicos e epidemiológicos dos pacientes, fatores ambientais do hospital ou do serviço de saúde, bem como fatores bacterianos que
pudessem ser preditivos para infecção versus colonização, assim como para o desenvolvimento de surtos por espécie(s) de BGN-MDR, ou fossem
de fatores de pior prognóstico. Para tanto, elaboramos este estudo parte retrospectivo parte prospectivo, para submissão ao Comitê de Ética em
Pesquisa para permitir a investigação de dados dos pacientes e dos hospitais. A investigação do trinômio pacientes, ambientes e agentes
bacterianos multirresistentes aos antibióticos, causando tanto infecção quanto colonização, deverá ser útil para ampliar o conhecimento sobre o
processo infeccioso nosocomial e determinar se há fatores associados à infecção e óbito por qualquer causa entre pacientes infectados, durante
internação hospitalar. Os relatos na literatura mundial referentes à elevada capacidade de disseminação e dificuldades para o monitoramento e

HOSPITAL UNIVERSITÁRIO PEDRO ERNESTO DA UNIVERSIDADE DO ESTADO DO RIO DE JANEIRO - HUUERJ

Continuação do Parecer: 6.275.749

controle de diferentes espécies, e perfis bacterianos multirresistente, ratificam a importância dessa pesquisa.

Objetivo da Pesquisa:

Objetivo Primário:

Investigar infecções/colonizações e surtos causados por BGN multirresistentes nos serviços de saúde de um Hospital Universitário do Rio de
Janeiro.

Objetivo Secundário:

i. Investigar a etiologia, os pacientes e ambientes durante ocorrências de infecção/colonização e surtos causados por BGN-MDR, de forma
retrospectiva e/ou prospectiva no Hospital Universitário do Rio de Janeiro. ii. Prosseguir na caracterização molecular das cepas de BGN detectadas
durante episódios epidêmicos e interepidêmicos. iii. Investigar os mecanismos de resistência de BGN-MDR identificados. iv. Investigar fatores de
virulência nas cepas isoladas. v. Investigar fatores individuais clínicos e epidemiológicos dos pacientes, bem como fatores bacterianos e do hospital,
ou serviço de saúde, que pudessem ser preditivos para infecções versus colonizações, assim como associados a surtos por BGN-MDR. vi. Investigar fatores bacterianos, dos pacientes e ambientes associados a pior
prognóstico entre pacientes infectados. vii. Investigar a sazonalidade e diferenças regionais na ocorrência de BGN-MDR e de seus mecanismos de
resistência. viii. Analisar a dinâmica espaço-temporal em tempo real de microrganismos multirresistentes e multissensíveis em hospital universitário
do Rio de Janeiro. ix. Investigar projetar e explorar modelos de aprendizado profundo baseados em grafos para a análise de padrões epidemiológicos
e microbiológicos bem como a predição de surtos hospitalares.

Avaliação dos Riscos e Benefícios:

Riscos:

Os riscos são mínimos, uma vez que não haverá nenhuma intervenção nos sujeitos da pesquisa e os autores se comprometem através do “Termos
de Confidencialidade e Anonimato” (Anexo 7), em garantir a privacidade dos pacientes e do hospital quanto às informações coletadas, considerando

Endereço: Av. 28 de setembro, nº77 - CePeM - Centro de Pesquisa Clínica Multiusuário - 2º andar/sala nº 28 - prédio
Bairro: Vila Isabel **CEP:** 20.551-030
UF: RJ **Município:** RIO DE JANEIRO
Telefone: (21)2868-8253 **E-mail:** cep@hupe.uerj.br

HOSPITAL UNIVERSITÁRIO PEDRO ERNESTO DA UNIVERSIDADE DO ESTADO DO RIO DE JANEIRO - HUUERJ

Continuação do Parecer: 6.275.749

que não serão utilizadas na pesquisa iniciais de nomes, números de registros, ou qualquer forma de identificação de pacientes envolvidos no estudo.
O hospital e a CCIH receberão os seus resultados, individualmente, durante e ao final da investigação, de forma sigilosa, objetivando contribuir para
a implementação de medidas de prevenção e controle de bactérias multirresistentes.
Benefícios:
O monitoramento de BGN-MDR deverá contribuir para avaliar, por meio de indicadores, o impacto das Políticas de Prevenção e Controle de
Bactérias Multirresistentes (PPCBM) que vem sendo instituídas pela CCIH do hospital. Este estudo deverá contribuir para o diagnóstico precoce de
surtos e para a prevenção e controle de infecções causados por BGN-MDR no próprio Hospital e servir de modelo para outras instituições de saúde
no Brasil e no mundo.

Comentários e Considerações sobre a Pesquisa:
O estudo compreende investigações prospectivas e retrospectivas de casos ou surtos de infeções/colonizações por microrganismos multirresistentes
em pacientes internados em um Hospital Universitário do Rio de Janeiro. As investigações poderão compreender períodos epidêmicos e interepidêmicos, causados por bactérias Gram-negativas (BGN) multirresistentes aos antibióticos. Os casos de infecções serão investigados a partir do
banco de dados do sistema automatizado do laboratório de Bacteriologia do Hospital Universitário do Rio de Janeiro, por meio de interfacemento de
sistemas ou acesso de banco de dados.

Considerações sobre os Termos de apresentação obrigatória:
Foram analisados os seguintes documentos de apresentação obrigatória:
1) Folha de Rosto para pesquisa envolvendo seres humanos;
2) Projeto de Pesquisa;
3) Termo de Consentimento Livre e Esclarecido;
4) Termo de Anuência Institucional;
5) Declaração de isenção de custos;
6) Orçamento;
7) Cronograma;
8) Currículo do pesquisador principal e demais colaboradores.

Endereço: Av. 28 de setembro, nº77 - CePeM - Centro de Pesquisa Clínica Multiusuário - 2º andar/sala nº 28 - prédio
Bairro: Vila Isabel **CEP:** 20.551-030
UF: RJ **Município:** RIO DE JANEIRO
Telefone: (21)2868-8253 **E-mail:** cep@hupe.uerj.br

HOSPITAL UNIVERSITÁRIO PEDRO ERNESTO DA UNIVERSIDADE DO ESTADO DO RIO DE JANEIRO - HUUERJ

Continuação do Parecer: 6.275.749

Conclusões ou Pendências e Lista de Inadequações:

O projeto pode ser realizado da forma como está apresentado. Após análise do protocolo foi verificado o atendimento à legislação vigente e o protocolo encontra-se apto para início. Caso o pesquisador precise fazer Emenda ao Projeto, é obrigatório o envio antecipado de Relatório Parcial via Notificação. A Emenda só poderá ser solicitada após aprovação da Notificação com relatório parcial. Diante do exposto e à luz da Resolução CNS nº466/2012, o projeto pode ser enquadrado na categoria – APROVADO.

Considerações Finais a critério do CEP:

De acordo com o item X.1.3.b, da Resolução CNS n. 466/12, o pesquisador deverá apresentar relatórios semestrais - a contar da data de aprovação do protocolo - que permitam ao CEP acompanhar o desenvolvimento do projeto. Esses relatórios devem conter as informações detalhadas - naqueles itens aplicáveis - nos moldes do relatório final contido no Ofício Circular n. 062/2011: conselho.saude.gov.br/web_comissoes/conep/index.htm, bem como deve haver menção ao período a que se referem. Para cada relatório, deve haver uma notificação separada. As informações contidas no relatório devem ater-se ao período correspondente e não a todo o período da pesquisa até aquele momento. Eventuais emendas (modificações) ao protocolo devem ser apresentadas de forma clara e sucinta, identificando-se, por cor, negrito ou sublinhado, a parte do documento a ser modificada, isto é, além de Apresentar o resumo das alterações, juntamente com a justificativa, é necessário destacá-las no decorrer do texto (item 2.2.H.1, da Norma Operacional CNS nº 001 de 2013). Pesquisador: Comunicar toda e qualquer alteração do projeto e no termo de consentimento livre e esclarecido, para análise das mudanças. Informar imediatamente qualquer evento adverso ocorrido durante o desenvolvimento da pesquisa; Os dados individuais de todas as etapas da pesquisa devem ser mantidos em local seguro por 5 anos para possível auditoria dos órgãos competentes.

Este parecer foi elaborado baseado nos documentos abaixo relacionados:

Tipo Documento	Arquivo	Postagem	Autor	Situação
Informações Básicas do Projeto	PB_INFORMAÇÕES_BÁSICAS_DO_PROJETO_2154957.pdf	03/07/2023 19:05:30		Aceito
Folha de Rosto	FolhaderostoIA.pdf	03/07/2023 18:57:06	Marisa Zenaide Ribeiro Gomes	Aceito
Declaração de Pesquisadores	AssiUeleres.pdf	03/07/2023 18:56:00	Marisa Zenaide Ribeiro Gomes	Aceito

Endereço: Av. 28 de setembro, nº77 - CePeM - Centro de Pesquisa Clínica Multiusuário - 2º andar/sala nº 28 - prédio
Bairro: Vila Isabel **CEP:** 20.551-030
UF: RJ **Município:** RIO DE JANEIRO
Telefone: (21)2868-8253 **E-mail:** cep@hupe.uerj.br

HOSPITAL UNIVERSITÁRIO PEDRO ERNESTO DA UNIVERSIDADE DO ESTADO DO RIO DE JANEIRO - HUUERJ

Continuação do Parecer: 6.275.749

Declaração de Pesquisadores	AssiDayse.pdf	03/07/2023 18:53:16	Marisa Zenaide Ribeiro Gomes	Aceito
Declaração de Pesquisadores	notificacolaboradordepesquisaIABarbara.pdf	03/07/2023 18:31:17	Marisa Zenaide Ribeiro Gomes	Aceito
Declaração de Pesquisadores	AssDavioCarvalho.pdf	03/07/2023 18:29:57	Marisa Zenaide Ribeiro Gomes	Aceito
Declaração de Pesquisadores	AssSheilaMariahdeSa.pdf	03/07/2023 18:29:40	Marisa Zenaide Ribeiro Gomes	Aceito
Declaração de Pesquisadores	AssColaboradores.pdf	03/07/2023 18:28:14	Marisa Zenaide Ribeiro Gomes	Aceito
Declaração de Instituição e Infraestrutura	AssMarcosLago.pdf	03/07/2023 18:27:10	Marisa Zenaide Ribeiro Gomes	Aceito
Declaração de Instituição e Infraestrutura	AssDrRobsonLeao.pdf	03/07/2023 18:26:55	Marisa Zenaide Ribeiro Gomes	Aceito
Declaração de Instituição e Infraestrutura	AssPenelopeZamboni.pdf	03/07/2023 18:26:23	Marisa Zenaide Ribeiro Gomes	Aceito
Declaração de Instituição e Infraestrutura	AssDrRuiTeofilo.pdf	03/07/2023 18:26:04	Marisa Zenaide Ribeiro Gomes	Aceito
Declaração de concordância	AssDrRonaldoDamiao.pdf	03/07/2023 18:21:50	Marisa Zenaide Ribeiro Gomes	Aceito
Projeto Detalhado / Brochura Investigador	ProjetoEspacialIAHUPE25052023docx.docx	03/07/2023 18:20:31	Marisa Zenaide Ribeiro Gomes	Aceito
Brochura Pesquisa	ProjetoEspacialIAHUPE25052023docx.pdf	03/07/2023 18:20:09	Marisa Zenaide Ribeiro Gomes	Aceito

Situação do Parecer:
Aprovado

Necessita Apreciação da CONEP:
Não

RIO DE JANEIRO, 01 de Setembro de 2023

Assinado por:
WILLE OIGMAN
(Coordenador(a))

Endereço: Av. 28 de setembro, nº77 - CePeM - Centro de Pesquisa Clínica Multiusuário - 2º andar/sala nº 28 - prédio
Bairro: Vila Isabel **CEP:** 20.551-030
UF: RJ **Município:** RIO DE JANEIRO
Telefone: (21)2868-8253 **E-mail:** cep@hupe.uerj.br

ANEXO B – Resultados completos das experimentações

Este anexo apresenta os resultados completos das configurações experimentais registradas. As linhas estão ordenadas por horizonte de previsão e incluem as métricas BCE, AUPRC, recall e Jaccard calculadas no conjunto de teste.

Tabela 11 – Resultados completos das configurações experimentais ordenados por horizonte

H	Vizinhança	Modelo	Atrib. paciente	Atrib. nó	BCE	AUPRC	Recall	Jaccard
3	Espacial	Média histórica	–	–	0,1117	0,6751	0,3254	0,3188
3	Espacial	Média histórica	–	–	0,1117	0,6751	0,3254	0,3188
3	Espacial	Espacial	AST	Nó	0,0599	0,7917	0,6622	0,6664
3	Espacial	Espacial	AST	Nó+escalares	0,0668	0,7791	0,6395	0,6251
3	Espacial	Espacial	AST+painel	Nó	0,0595	0,7879	0,6294	0,6215
3	Espacial	Espacial	AST+painel	Nó+escalares	0,0636	0,7887	0,6462	0,6307
3	Espacial	Temporal	AST	Nó	0,0512	0,8198	0,6588	0,6866
3	Espacial	Temporal	AST	Nó+escalares	0,0475	0,8250	0,6930	0,7041
3	Espacial	Temporal	AST+painel	Nó	0,0470	0,8362	0,7048	0,7273
3	Espacial	Temporal	AST+painel	Nó+escalares	0,0495	0,8259	0,7101	0,7149
3	Espacial	GraphSAGE+GRU	AST	Nó	0,0910	0,3712	0,1963	0,2333
3	Espacial	GraphSAGE+GRU	AST	Nó+escalares	0,0586	0,7789	0,6796	0,6832
3	Espacial	GraphSAGE+GRU	AST+painel	Nó	0,0594	0,7635	0,6126	0,6208
3	Espacial	GraphSAGE+GRU	AST+painel	Nó+escalares	0,0615	0,7618	0,6575	0,6578
3	Espacial	GraphSAGE+atenção	AST	Nó	0,0710	0,7512	0,6313	0,5997
3	Espacial	GraphSAGE+atenção	AST	Nó+escalares	0,0825	0,7401	0,6243	0,5741
3	Espacial	GraphSAGE+atenção	AST+painel	Nó	0,0798	0,7307	0,6343	0,6156
3	Espacial	GraphSAGE+atenção	AST+painel	Nó+escalares	0,0805	0,7233	0,6030	0,5698
3	Espacial	GAT+GRU	AST	Nó	0,0673	0,6798	0,5006	0,4150
3	Espacial	GAT+GRU	AST	Nó+escalares	0,0681	0,6746	0,4871	0,4178
3	Espacial	GAT+GRU	AST+painel	Nó	0,0870	0,4509	0,2529	0,2378
3	Espacial	GAT+GRU	AST+painel	Nó+escalares	0,0744	0,6345	0,4905	0,4040
3	Serviço+espacial	Média histórica	–	–	0,1117	0,6751	0,3254	0,3188
3	Serviço+espacial	Espacial	AST	Nó	0,0632	0,7764	0,6289	0,6248
3	Serviço+espacial	Espacial	AST	Nó+escalares	0,0685	0,7790	0,6413	0,5927
3	Serviço+espacial	Espacial	AST+painel	Nó	0,0579	0,7930	0,6342	0,6126
3	Serviço+espacial	Espacial	AST+painel	Nó+escalares	0,0738	0,7534	0,6258	0,5812
3	Serviço+espacial	Temporal	AST	Nó	0,0498	0,8131	0,7082	0,7204
3	Serviço+espacial	Temporal	AST	Nó+escalares	0,0515	0,8149	0,6786	0,7093
3	Serviço+espacial	Temporal	AST+painel	Nó	0,0479	0,8248	0,7024	0,7155

H	Vizinhança	Modelo	Atrib. paciente	Atrib. nó	BCE	AUPRC	Recall	Jaccard
3	Serviço+espacial	Temporal	AST+painel	Nó+escalares	0,0463	0,8433	0,7132	0,7375
3	Serviço+espacial	GraphSAGE+GRU	AST	Nó	0,0623	0,7549	0,6268	0,6358
3	Serviço+espacial	GraphSAGE+GRU	AST	Nó+escalares	0,0635	0,7418	0,6193	0,6275
3	Serviço+espacial	GraphSAGE+GRU	AST+painel	Nó	0,0611	0,7652	0,6344	0,6601
3	Serviço+espacial	GraphSAGE+GRU	AST+painel	Nó+escalares	0,0620	0,7699	0,6340	0,6479
3	Serviço+espacial	GraphSAGE+atenção	AST	Nó	0,0797	0,7006	0,6105	0,5807
3	Serviço+espacial	GraphSAGE+atenção	AST	Nó+escalares	0,0853	0,7313	0,6373	0,6087
3	Serviço+espacial	GraphSAGE+atenção	AST+painel	Nó	0,0794	0,7221	0,6247	0,5784
3	Serviço+espacial	GraphSAGE+atenção	AST+painel	Nó+escalares	0,0772	0,6965	0,6122	0,5605
3	Serviço+espacial	GAT+GRU	AST	Nó	0,0753	0,6040	0,4841	0,3961
3	Serviço+espacial	GAT+GRU	AST	Nó+escalares	0,0788	0,5601	0,3909	0,3189
3	Serviço+espacial	GAT+GRU	AST+painel	Nó	0,0742	0,6151	0,4895	0,3738
3	Serviço+espacial	GAT+GRU	AST+painel	Nó+escalares	0,0744	0,6220	0,5221	0,4071
7	Espacial	Média histórica	–	–	0,1111	0,6773	0,3367	0,3283
7	Espacial	Média histórica	–	–	0,1111	0,6773	0,3367	0,3283
7	Espacial	Espacial	AST	Nó	0,0719	0,7363	0,6058	0,5495
7	Espacial	Espacial	AST	Nó+escalares	0,0813	0,7101	0,5899	0,5270
7	Espacial	Espacial	AST+painel	Nó	0,0719	0,7252	0,5969	0,5214
7	Espacial	Espacial	AST+painel	Nó+escalares	0,0795	0,7220	0,5832	0,5248
7	Espacial	Temporal	AST	Nó	0,0554	0,7816	0,6320	0,6390
7	Espacial	Temporal	AST	Nó+escalares	0,0563	0,7768	0,6408	0,6319
7	Espacial	Temporal	AST+painel	Nó	0,0595	0,7698	0,6356	0,6286
7	Espacial	Temporal	AST+painel	Nó+escalares	0,0568	0,7829	0,6551	0,6612
7	Espacial	GraphSAGE+GRU	AST	Nó	0,0680	0,7057	0,5753	0,5667
7	Espacial	GraphSAGE+GRU	AST	Nó+escalares	0,0931	0,3567	0,1944	0,2284
7	Espacial	GraphSAGE+GRU	AST+painel	Nó	0,0668	0,7287	0,6138	0,5871
7	Espacial	GraphSAGE+GRU	AST+painel	Nó+escalares	0,0657	0,7316	0,6173	0,5962
7	Espacial	GraphSAGE+atenção	AST	Nó	0,0824	0,7061	0,5949	0,5140
7	Espacial	GraphSAGE+atenção	AST	Nó+escalares	0,0929	0,7033	0,6109	0,5200
7	Espacial	GraphSAGE+atenção	AST+painel	Nó	0,0978	0,6662	0,6097	0,5042
7	Espacial	GraphSAGE+atenção	AST+painel	Nó+escalares	0,0917	0,6784	0,5960	0,5178

H	Vizinhança	Modelo	Atrib. paciente	Atrib. nó	BCE	AUPRC	Recall	Jaccard
7	Espacial	GAT+GRU	AST	Nó	0,0737	0,6212	0,4350	0,3624
7	Espacial	GAT+GRU	AST	Nó+escalares	0,0702	0,6604	0,4826	0,3945
7	Espacial	GAT+GRU	AST+painel	Nó	0,0759	0,6386	0,4562	0,3834
7	Espacial	GAT+GRU	AST+painel	Nó+escalares	0,0709	0,6402	0,4668	0,3742
7	Serviço+espacial	Média histórica	–	–	0,1111	0,6773	0,3367	0,3283
7	Serviço+espacial	Espacial	AST	Nó	0,0702	0,7443	0,6056	0,5563
7	Serviço+espacial	Espacial	AST	Nó+escalares	0,0810	0,7134	0,5819	0,5187
7	Serviço+espacial	Espacial	AST+painel	Nó	0,0736	0,7236	0,5986	0,5538
7	Serviço+espacial	Espacial	AST+painel	Nó+escalares	0,0765	0,7205	0,5947	0,5337
7	Serviço+espacial	Temporal	AST	Nó	0,0578	0,7818	0,6574	0,6519
7	Serviço+espacial	Temporal	AST	Nó+escalares	0,0587	0,7584	0,6445	0,6248
7	Serviço+espacial	Temporal	AST+painel	Nó	0,0611	0,7561	0,5861	0,6004
7	Serviço+espacial	Temporal	AST+painel	Nó+escalares	0,0581	0,7621	0,6444	0,6467
7	Serviço+espacial	GraphSAGE+GRU	AST	Nó	0,0662	0,7335	0,5965	0,5879
7	Serviço+espacial	GraphSAGE+GRU	AST	Nó+escalares	0,0673	0,7386	0,6075	0,5883
7	Serviço+espacial	GraphSAGE+GRU	AST+painel	Nó	0,0775	0,5868	0,3735	0,3891
7	Serviço+espacial	GraphSAGE+GRU	AST+painel	Nó+escalares	0,0635	0,7461	0,6155	0,6119
7	Serviço+espacial	GraphSAGE+atenção	AST	Nó	0,0940	0,6685	0,5546	0,4987
7	Serviço+espacial	GraphSAGE+atenção	AST	Nó+escalares	0,0892	0,6976	0,6081	0,5435
7	Serviço+espacial	GraphSAGE+atenção	AST+painel	Nó	0,0871	0,6872	0,5906	0,5197
7	Serviço+espacial	GraphSAGE+atenção	AST+painel	Nó+escalares	0,0882	0,7062	0,6093	0,5352
7	Serviço+espacial	GAT+GRU	AST	Nó	0,0793	0,5924	0,4038	0,3460
7	Serviço+espacial	GAT+GRU	AST	Nó+escalares	0,0838	0,5080	0,3227	0,3019
7	Serviço+espacial	GAT+GRU	AST+painel	Nó	0,0784	0,6024	0,4026	0,3464
7	Serviço+espacial	GAT+GRU	AST+painel	Nó+escalares	0,0790	0,5756	0,3955	0,3132
15	Espacial	Média histórica	–	–	0,1099	0,6821	0,3436	0,3344
15	Espacial	Espacial	AST	Nó	0,0806	0,7100	0,5755	0,5265
15	Espacial	Espacial	AST	Nó+escalares	0,0921	0,6968	0,5863	0,5143
15	Espacial	Espacial	AST+painel	Nó	0,0831	0,6937	0,5827	0,5289
15	Espacial	Espacial	AST+painel	Nó+escalares	0,0837	0,7121	0,5871	0,5081
15	Espacial	Temporal	AST	Nó	0,0644	0,7291	0,5974	0,5686

H	Vizinhança	Modelo	Atrib. paciente	Atrib. nó	BCE	AUPRC	Recall	Jaccard
15	Espacial	Temporal	AST	Nó+escalares	0,0674	0,7350	0,6202	0,5941
15	Espacial	Temporal	AST+painel	Nó	0,0646	0,7358	0,6152	0,5802
15	Espacial	Temporal	AST+painel	Nó+escalares	0,0636	0,7393	0,6100	0,5994
15	Espacial	GraphSAGE+GRU	AST	Nó	0,0691	0,6947	0,5830	0,5479
15	Espacial	GraphSAGE+GRU	AST	Nó+escalares	0,0702	0,7165	0,5981	0,5497
15	Espacial	GraphSAGE+GRU	AST+painel	Nó	0,0662	0,7146	0,6010	0,5695
15	Espacial	GraphSAGE+GRU	AST+painel	Nó+escalares	0,0732	0,7083	0,5983	0,5563
15	Espacial	GraphSAGE+atenção	AST	Nó	0,0962	0,6627	0,6159	0,5422
15	Espacial	GraphSAGE+atenção	AST	Nó+escalares	0,0981	0,6707	0,5935	0,5204
15	Espacial	GraphSAGE+atenção	AST+painel	Nó	0,0933	0,6720	0,6076	0,5421
15	Espacial	GraphSAGE+atenção	AST+painel	Nó+escalares	0,0974	0,6674	0,5949	0,5154
15	Espacial	GAT+GRU	AST	Nó	0,0743	0,6257	0,4276	0,3355
15	Espacial	GAT+GRU	AST	Nó+escalares	0,0909	0,3715	0,1874	0,2062
15	Espacial	GAT+GRU	AST+painel	Nó	0,0877	0,4369	0,1911	0,2065
15	Espacial	GAT+GRU	AST+painel	Nó+escalares	0,0743	0,6173	0,4450	0,3515
15	Serviço+espacial	Média histórica	–	–	0,1099	0,6821	0,3436	0,3344
15	Serviço+espacial	Espacial	AST	Nó	0,0786	0,7068	0,5831	0,5095
15	Serviço+espacial	Espacial	AST	Nó+escalares	0,1005	0,6945	0,5658	0,4954
15	Serviço+espacial	Espacial	AST+painel	Nó	0,0770	0,6983	0,5771	0,5097
15	Serviço+espacial	Espacial	AST+painel	Nó+escalares	0,0903	0,7126	0,6033	0,5378
15	Serviço+espacial	Temporal	AST	Nó	0,0632	0,7384	0,6232	0,5871
15	Serviço+espacial	Temporal	AST	Nó+escalares	0,0633	0,7459	0,6296	0,5845
15	Serviço+espacial	Temporal	AST+painel	Nó	0,0649	0,7289	0,6161	0,5861
15	Serviço+espacial	Temporal	AST+painel	Nó+escalares	0,0648	0,7485	0,6225	0,6124
15	Serviço+espacial	GraphSAGE+GRU	AST	Nó	0,0662	0,6988	0,5729	0,5335
15	Serviço+espacial	GraphSAGE+GRU	AST	Nó+escalares	0,0721	0,6952	0,5945	0,5565
15	Serviço+espacial	GraphSAGE+GRU	AST+painel	Nó	0,0769	0,5913	0,4255	0,4199
15	Serviço+espacial	GraphSAGE+GRU	AST+painel	Nó+escalares	0,0723	0,6847	0,5639	0,5061
15	Serviço+espacial	GraphSAGE+atenção	AST	Nó	0,0947	0,6673	0,6075	0,5341
15	Serviço+espacial	GraphSAGE+atenção	AST	Nó+escalares	0,0955	0,6751	0,5922	0,5013
15	Serviço+espacial	GraphSAGE+atenção	AST+painel	Nó	0,0944	0,6686	0,6116	0,5258

H	Vizinhança	Modelo	Atrib. paciente	Atrib. nó	BCE	AUPRC	Recall	Jaccard
15	Serviço+espacial	GraphSAGE+atenção	AST+painel	Nó+escalares	0,1032	0,6578	0,5808	0,4933
15	Serviço+espacial	GAT+GRU	AST	Nó	0,0795	0,5591	0,3998	0,3424
15	Serviço+espacial	GAT+GRU	AST	Nó+escalares	0,0826	0,5486	0,4409	0,3388
15	Serviço+espacial	GAT+GRU	AST+painel	Nó	0,0789	0,5854	0,4572	0,3792
15	Serviço+espacial	GAT+GRU	AST+painel	Nó+escalares	0,0809	0,5738	0,4412	0,3597

DADOS CURRICULARES

IDENTIFICAÇÃO	
Nome	César Romano Delgado Marín
Data de nascimento	06/02/1996
Nacionalidade	Colômbia
Nome em citações bibliográficas	DELGADO MARIN, César Romano
Currículo Lattes	<http://lattes.cnpq.br/7733154993275911>
ORCID	<https://orcid.org/0000-0002-3096-2687>
FORMAÇÃO ACADÊMICA	
Graduação	2013–2019: Licenciatura em Design Tecnológico, Universidade Pedagógica Nacional, Bogotá (Colômbia).
PARTICIPAÇÃO EM EVENTOS CIENTÍFICOS	
XXXI SEMANA DA CIÊNCIA DA COMPUTAÇÃO, 2023, Rio Claro (SP). GNNs para a Previsão de Agentes Infecciosos. Palestra.	
64 CONGRESSO CIENTÍFICO DO HUPE: SAÚDE 5.0 - CONECTANDO TECNOLOGIA E HUMANIZAÇÃO, 2026, Rio de Janeiro (RJ). VIGILÂNCIA HOSPITALAR REATIVA À VIGILÂNCIA HOSPITALAR PREDITIVA UTILIZANDO DEEP LEARNING EM GRAFOS ESPAÇO-TEMPORAIS: MODELO EM HOSPITAL UNIVERSITÁRIO DO RIO DE JANEIRO. Poster.	