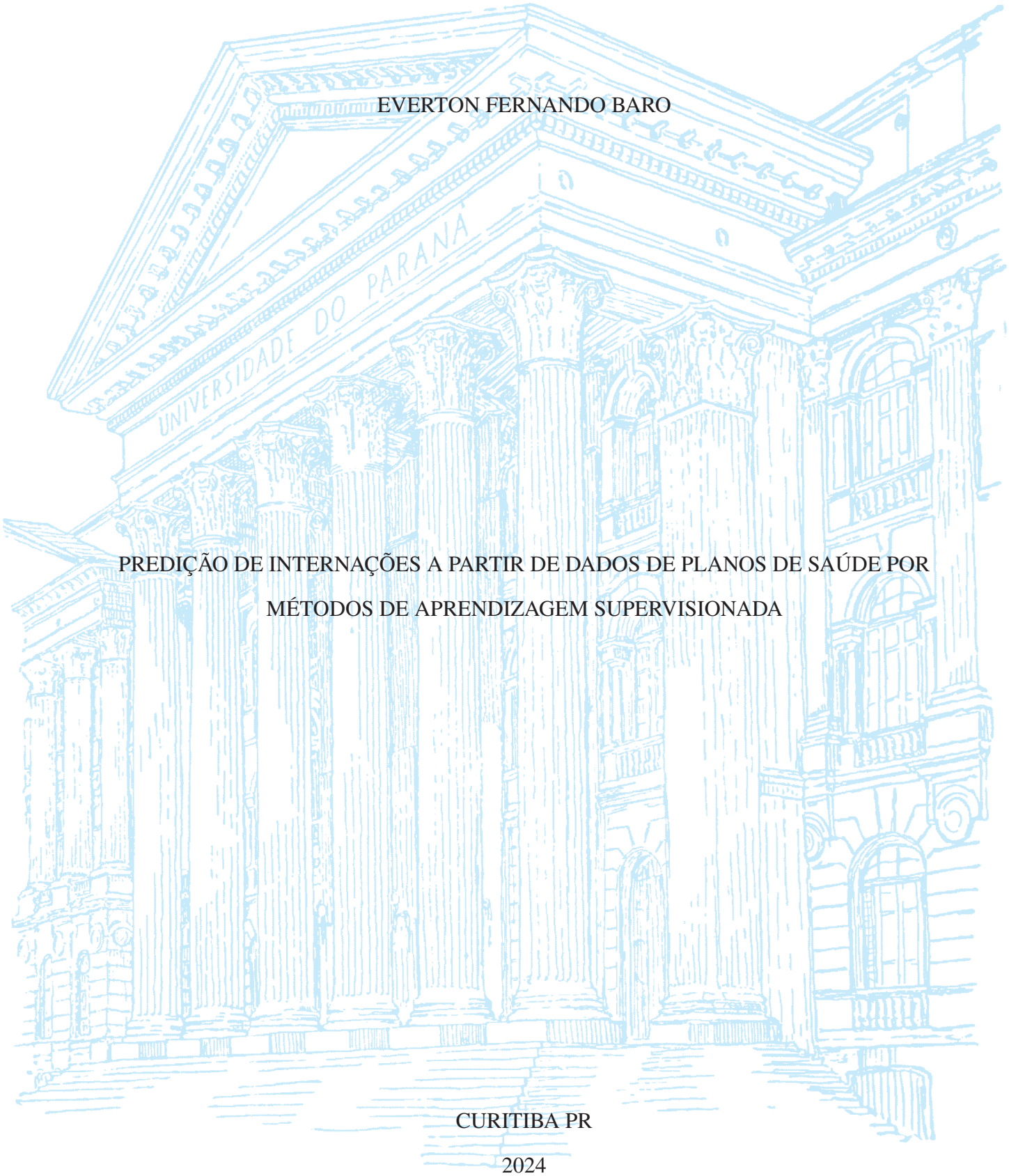
UNIVERSIDADE FEDERAL DO PARANÁ

EVERTON FERNANDO BARO



PREDIÇÃO DE INTERNAÇÕES A PARTIR DE DADOS DE PLANOS DE SAÚDE POR MÉTODOS DE APRENDIZAGEM SUPERVISIONADA

CURITIBA PR

2024

EVERTON FERNANDO BARO

PREDIÇÃO DE INTERNAÇÕES A PARTIR DE DADOS DE PLANOS DE SAÚDE POR MÉTODOS DE APRENDIZAGEM SUPERVISIONADA

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

Área de concentração: *Ciência da Computação*.

Orientador: Dr. Luiz Eduardo Soares de Oliveira.

Coorientador: Dr. Alceu de Souza Britto Jr.

CURITIBA PR

2024

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

Baro, Everton Fernando
Predição de internações a partir de dados de planos de saúde por métodos de aprendizagem supervisionada / Everton Fernando Baro. – Curitiba, 2024.
1 recurso on-line : PDF.

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

Orientador: Luiz Eduardo Soares de Oliveira
Coorientador: Alceu de Souza Britto Junior

1. Planos de Saúde. 2. Hospitais – Admissão e alta. 3. Acidente vascular cerebral. 4. Aprendizado do computador. I. Universidade Federal do Paraná. II. Programa de Pós-Graduação em Informática. III. Oliveira, Luiz Eduardo Soares de. IV. Britto Junior, Alceu de Souza. V. Título.

Bibliotecário: Elias Barbosa da Silva CRB-9/1894

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação INFORMÁTICA da Universidade Federal do Paraná foram convocados para realizar a arguição da tese de Doutorado de **EVERTON FERNANDO BARO** intitulada: **PREDIÇÃO DE INTERNAÇÕES A PARTIR DE DADOS DE PLANOS DE SAÚDE POR MÉTODOS DE APRENDIZAGEM SUPERVISIONADA**, sob orientação do Prof. Dr. LUIZ EDUARDO SOARES DE OLIVEIRA, que após terem inquirido o aluno e realizada a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.

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

CURITIBA, 03 de Dezembro de 2024.

Assinatura Eletrônica
04/12/2024 07:29:46.0
LUIZ EDUARDO SOARES DE OLIVEIRA
Presidente da Banca Examinadora

Assinatura Eletrônica
03/12/2024 19:08:38.0
GEORGE DARMITON DA CUNHA CAVALCANTI
Avaliador Externo (UNIVERSIDADE FEDERAL DE PERNAMBUCO)

Assinatura Eletrônica
04/12/2024 07:21:54.0
LUCAS FERRARI DE OLIVEIRA
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

Assinatura Eletrônica
04/12/2024 08:39:57.0
JULIO CÉSAR NIEVOLA
Avaliador Externo (PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ)

Assinatura Eletrônica
04/12/2024 11:18:06.0
ALCEU DE SOUZA BRITTO JR
Coorientador(a) (PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ)

Dedico este trabalho aos meus pais Mario Aparecido Baro e Clair Balan Baro.

AGRADECIMENTOS

Agradeço profundamente a todas as pessoas que foram fundamentais nesta jornada de doutorado.

Ao meu orientador, Luiz Eduardo Soares de Oliveira, sou profundamente grato pela orientação sábia e pela prontidão em responder minhas dúvidas. Sua orientação me guiou de forma decisiva em momentos críticos. Trabalhar sob sua orientação foi uma honra e uma grande oportunidade de aprendizado, que levarei para toda a minha vida profissional e acadêmica. Agradeço também ao co-orientador, Alceu de Souza Britto Jr., pelas revisões detalhadas e orientações enriquecedoras que aprimoraram significativamente meu trabalho. Foi um privilégio contar com sua expertise e dedicação, que contribuíram imensamente para o sucesso desta pesquisa.

À minha esposa, Míriam Juliana Pastori Bosco, agradeço por seu amor incondicional e apoio constante. Sua paciência e incentivo foram essenciais para que eu pudesse enfrentar os desafios dessa trajetória.

Aos meus pais, Mario Aparecido Baro e Clair Balan Baro, agradeço por serem sempre presentes e dedicados, torcendo e trabalhando incansavelmente pelo sucesso dos filhos. Sua constante motivação, apoio incondicional e ensinamentos foram fundamentais para que eu pudesse sonhar e alcançar meus objetivos. Vocês são um exemplo de amor e dedicação, sou grato por tudo o que fizeram por mim.

Aos meus irmãos, Gustavo Baro e Jean Baro, meu sincero agradecimento pela compreensão e apoio nos momentos em que estive ausente, focado nos estudos. Esta conquista é o resultado de um esforço conjunto, e sou grato a todos por terem deixado sua marca em minha vida.

Gostaria também de expressar minha gratidão aos membros da banca examinadora, George Darmiton da Cunha Cavalcanti, Lucas Ferrari de Oliveira e Julio César Nievola, pela disponibilidade, pelas valiosas contribuições e observações que enriqueceram ainda mais este trabalho.

Por fim, agradeço ao Instituto Federal do Paraná (IFPR), pelo apoio que possibilitou a realização deste trabalho, por meio do financiamento e do afastamento integral concedido para pós-graduação, e à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES).

RESUMO

As internações hospitalares constituem uma parte significativa dos gastos do sistema de saúde. Além de encarecer o sistema, em vários casos, essas internações podem significar um aumento na complexidade das condições de saúde dos pacientes, refletindo a gravidade de seus problemas médicos e suas implicações para a saúde. Portanto, a prevenção de internações não é apenas uma estratégia de redução de custos, mas também uma medida crucial para promover a qualidade de vida dos pacientes. Neste contexto, a previsão de internações surge como uma solução viável, possibilitando aos gestores otimizar a aplicação de recursos e que as iniciativas médicas promovam intervenções preventivas. Ademais, a análise preditiva pode enriquecer a literatura médica para diversos tipos de problemas de saúde, revelando, por exemplo, fatores que podem contribuir para complicações de doenças. A realização da previsão de internações exige a utilização de dados com características preditivas adequadas. Contudo, a obtenção desses dados não é uma tarefa trivial, tanto em relação ao acesso quanto à seleção das características a serem utilizadas nos treinamentos, demandando conhecimento especializado. Visando eliminar essa necessidade, esta pesquisa propõe uma estratégia de representação de dados de planos de saúde baseada em sentenças, facilitando sua utilização em treinamentos de algoritmos de aprendizado de máquina. Este trabalho também apresenta uma abordagem para a geração de um modelo de previsão de internações utilizando essas sentenças, com o objetivo de prever tanto internações gerais quanto para casos específicos, como casos de internações por AVC (Acidente Vascular Cerebral). As abordagens apresentadas abrangem desde técnicas tradicionais de aprendizado de máquina, como *Random Forest* e *Gradient Boosting*, até os mais recentes modelos de linguagem, como BERT e LLaMA. Os resultados experimentais obtidos demonstram a viabilidade dessa proposta, apresentando resultados que superam diversos trabalhos da literatura. Para internações em geral, foram alcançados F1-Score = 87,8 e AUC = 0,955, enquanto para internações por AVC, o melhor modelo atingiu F1-Score = 88,7 e AUC = 0,964. Além disso, os modelos gerados neste estudo possibilitam aplicações em várias outras áreas da saúde.

Palavras-chave: Planos de Saúde, Previsão de Internação, Previsão de AVC, Aprendizado de Máquina, Gradient Boosting, Random Forest, Modelos de Linguagem de Grande Escala, BERT, RoBERTa, LLAMA.

ABSTRACT

Hospitalizations constitute a significant part of health system expenditures. In addition to increasing costs, these hospitalizations can often indicate a rise in the complexity of patients' health conditions, reflecting the severity of their medical issues and their implications for health. Therefore, preventing hospitalizations is not just a cost-reduction strategy but also a crucial measure to enhance patients' quality of life. In this context, predicting hospitalizations emerges as a viable solution, allowing managers to optimize resource allocation and enabling medical initiatives to promote preventive interventions. Moreover, predictive analysis can enrich the medical literature for various health issues, revealing factors that may contribute to disease complications. Conducting hospitalization predictions requires the use of data with suitable predictive characteristics. However, obtaining this data is not a trivial task, both in terms of access and the selection of features for training, necessitating specialized knowledge. To eliminate this need, this research proposes a data representation strategy for health plans based on sentences, facilitating their use in training machine learning algorithms. This work also presents an approach for generating a hospitalization prediction model using these sentences, aimed at forecasting both general hospitalizations and specific cases, such as hospitalizations due to stroke. The approaches discussed range from traditional machine learning techniques, like Random Forest and Gradient Boosting, to the latest language models, such as BERT and LLaMA. The experimental results obtained demonstrate the viability of this proposal, showing results that surpass several studies in the literature. For hospitalizations in general, F1-Score = 87.8 and AUC = 0.955 were achieved, while for stroke-related hospitalizations, the best model achieved F1-Score = 88.7 and AUC = 0.964. Additionally, the models generated in this study enable applications in various other areas of health.

Keywords: Health Insurance, Hospitalization Prediction, Stroke Prediction, Machine Learning, Gradient Boosting, Random Forest, Large Language Model, BERT, RoBERTa, LLAMA.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ACRÔNIMOS

AB	*AdaBoost*
ANS	Agência Nacional de Saúde Suplementar
ANVISA	Agência Nacional de Vigilância Sanitária
API	*Application Programming Interface*
APS	Atenção Primária à Saúde
AUC	*Area Under the ROC Curve*
AVC	Acidente Vascular Cerebral
BERT	*Bidirectional Encoder Representations from Transformers*
BoW	*Bag-of-Words*
CA	*Cohort Analysis*
CART	*Classification and Regression Trees*
CID	Classificação Estatística Internacional de Doenças e Problemas Relacionados com a Saúde
CR	*Cox Regression*
DB1	*Data Base 1*
DB2	*Data Base 2*
DNA	Ácido Desoxirribonucleico
DS	*Descriptive statistics*
DT	*Decision Tree*
DW	*Data Warehouse*
ELMo	*Embeddings from Language Model*
FC	*Fully Connected Layer*
FN	Falso Negativos
FP	Falsos Positivos
GB	*Gradient Boosting*
GB	*Gigabyte*
GISC	*Gestione Integrata dello Scompenso Cardiaco*
GLMN	*Generalized Linear Model Net*
GPT	*Generative Pre-trained Transformer*
GPU	*Graphics Processing Unit*
HRS	*Hospitalization Risk Score*
IAM	Infarto Agudo do Miocárdio
IC	Insuficiência Cardíaca
ICFEP	Insuficiência Cardíaca com Fração de Ejeção Preservada
ICSAP	Internações por Condições Sensíveis à Atenção Primária

ID	Identificador
LB	*LogitBoost*
LlaMA	*Large Language Model Meta AI*
LLM	*Large Language Model*
LM	*Linear model*
LoRA	*Low-Rank Adaptation Of Large Language Models*
LR	*Logistic Regression*
LSTM	*Long Short Term Memory*
MLM	*Masked Language Model*
NB	*Naive Bayes*
NER	*Named Entity Recognition*
NLI	*Natural Language Inference*
NLP	*Natural Language Processing*
NN	*Neural network*
NSP	*Next Sentence Predictio*
PEFT	*Parameter Efficient Fine-Tuning*
PCA	*Principal Component Analysis*
PLN	Processamento de Linguagem Natural
QA	*Question Answering*
QloRA	*Quantized Low Rank Adaptation*
RDC	Resolução da Diretoria Colegiada
RES	Registro Eletrônico em Saúde
RF	*Random Forest*
RMS	*Random Subspace Method*
RNN	*Recurrent Neural Networks*
RoBERTa	*A Robustly Optimized BERT Pretraining Approach*
ROC	*Receiver Operating Characteristc*
SBERT	*Sentence Transformers*
SHAP	*SHapley Additive exPlanations*
SUS	Sistema Único de Saúde
SVM	*Support Vector Machine*
TF-IDF	*Term Frequency-Inverse Document Frequency*
VN	Verdadeiros Negativos
VP	Verdadeiros Positivos

SUMÁRIO

1 INTRODUÇÃO

Internações hospitalares representam uma parte expressiva dos custos do sistema de saúde brasileiro. Somente no SUS (Sistema Único de Saúde), no ano de 2013, foram gastos mais de 11 bilhões de reais com internações (Souza e Peixoto, 2017). Em 2019, se contabilizados os valores solicitados para pagamentos de procedimentos e insumos relacionados a internações aos planos de saúde, o valor ultrapassa 51 bilhões de reais[1]. Considerando esses valores, mesmo a redução de uma pequena porcentagem das internações torna-se significativa. Tais internações não são apenas caras, mas também potencialmente complexas para os pacientes, além de representar, em vários casos, a complicação do estado de saúde do paciente. Desta forma, quando possível, evitar que internações ocorram mostra-se benéfico, considerando aspectos monetários, qualidade de vida e saúde dos pacientes.

Uma possibilidade para evitar internações é tentar prevê-las. Com esse tipo de previsão, gestores hospitalares e de planos de saúde podem planejar e otimizar processos para reduzir custos. Para a medicina, conhecer antecipadamente possíveis casos de internação permite que ações de prevenção sejam tomadas com o objetivo de evitar que elas ocorram. Além disso, a análise destas previsões pode ser um importante objeto de estudo, permitindo, por exemplo, a descoberta de fatores que levam a complicações de doenças, alguns do quais talvez desconhecidos.

Na literatura científica, existem diversos trabalhos relacionados à previsão de internações. A maioria deles aborda problemas de saúde específicos, como aqueles relacionados a doenças cardíacas (Smith et al., 2011; Wang et al., 2012; Hebert et al., 2014a; Lorenzoni et al., 2019; Angraal et al., 2020), ou casos de reinternação (Singal et al., 2013; Choudhry et al., 2013; Baillie et al., 2013; Hebert et al., 2014a; Rana et al., 2014; Hao et al., 2014; Baig et al., 2020). Poucos artigos tratam da previsão de internações que abrangem diversos problemas de saúde simultaneamente (Hippisley-Cox e Coupland, 2013; Billings et al., 2013), ou seja, com um propósito mais generalista. Além disso, esses trabalhos utilizam abordagens que vão de métodos estatísticos a aprendizado de máquina.

Independentemente do método utilizado na realização de previsões, o fator preponderante são os dados, já que são eles que permitem a geração dos modelos preditivos. Considerando os estudos analisados nesta tese, a maioria utiliza dados relacionados a comorbidades, sintomas, aspectos socioeconômicos e sociais, sinais vitais, uso de medicação contínua e resultados de exames como insumo para geração de modelos. Eles realizam seleção e extração de características para treinamento considerando a literatura médica para o problema de saúde com o qual a internação se relaciona. Estes processos são muitas vezes exigentes, requerem conhecimentos especializados e podem não colaborar para a obtenção de bons modelos de predição, pois características discriminantes podem ser negligenciadas devido às especificações próprias das doenças abordadas na literatura médica, excluindo, desta forma, fatores nem sempre considerados e potencialmente determinantes. Assim, a utilização de métodos automatizados para seleção e extração de características dos dados para treinamento, mostra-se pertinente para melhorar a qualidade dos modelos gerados. Além disso, utilizar métodos automatizados podem reduzir o tempo normalmente gasto em pré-processamento e preparação de dados.

No Brasil, além do SUS, existem vários sistemas de gestão de operadoras de planos de saúde suplementar que mantêm dados de seus beneficiários. Nos planos de saúde, esses registros contêm dados demográficos, relativos a realização de exames, consultas, cirurgias, insumos utilizados, diagnóstico e demais procedimentos que os planos de saúde devem oferecer aos seus

[1] Cálculo realizado com base nos dados públicos da ANS (ANS, 2021)

beneficiários. Diferentemente da abordagem dos trabalhos encontrados na literatura, os dados dos planos de saúde incluem procedimentos e insumos utilizados pelos pacientes sobre todas as vezes que eles entraram em contato com o sistema de saúde. Esses dados, se devidamente organizados, podem representar um histórico das condições de saúde do paciente. Assim, mesmo que um problema ou doença não tenha sido diagnosticada ou registrado, os contatos com o sistema de saúde e insumos utilizados podem indicar o problema ou doença real do paciente. Entretanto, o número de procedimentos e insumos possíveis torna a seleção manual de características para os algoritmos de aprendizado de máquina algo que pode ser limitador para a geração dos modelos, sendo necessária a aplicação de métodos automatizados.

Assim, considerando a relevância da previsão de internações, a grande quantidade de dados de planos de saúde, e as vantagens na automatização na extração e seleção de características em dados de saúde, apresenta-se nesta tese uma abordagem para a geração de modelos de previsões de internações com capacidades generalistas, ou seja, que independa do problema de saúde com o qual a internação está relacionada e que também funcione para problemas específicos. Esta abordagem demonstra como transformar dados estruturados de planos de saúde em dados para formato textual representativo do histórico de saúde dos beneficiários, e como fazer a extração e seleção dessas características por meio de métodos avançados de PLN (Processamento de Linguagem Natural) para posterior utilização em algoritmos de aprendizado de máquina.

1.1 OBJETIVOS

O objetivo principal deste trabalho é propor um método baseado em aprendizagem de máquina, combinando dados de operadoras de planos de saúde, para criar um modelo preditivo de internações, de modo que o modelo gerado funcione para internações de diversos tipos de problemas e condições de saúde.

Os objetivos secundários deste trabalho são:

- Criar e disponibilizar uma base de dados com eventos históricos de pacientes internados e não-internados a partir de dados estruturados de planos de saúde.
- Analisar, testar e propor um método automatizado para seleção e extração de características dos dados históricos dos pacientes para treinamento de modelos preditores de internações.
- Propor e construir modelos de previsão de internações a partir de dados históricos dos pacientes.
- Validar os algoritmos e os protótipos desenvolvidos a partir de testes com a base de dados de internações, avaliando o desempenho.
- Avaliar a capacidade dos modelos quanto à capacidade de previsões de internações para problemas específicos.
- Avaliar os modelos para previsão de internação em diferentes períodos de antecedência.

1.2 DESAFIOS

A previsão de internações hospitalares é, muitas vezes, um objetivo primordial para médicos, gestores hospitalares e planos de saúde. A previsão de internações pode ajudar no

planejamento e organização de hospitais assim como de planos de saúde, fornecendo, além disso, suporte informacional a médicos para realização de ações preventivas e também na agilidade do atendimento em situações de emergência. A maior parte dos trabalhos encontrados na literatura desenvolveram modelos de previsão considerando internações para problemas de saúde específicos (Dai et al., 2015; Lorenzoni et al., 2019; Angraal et al., 2020), previsão de reinternação (Baig et al., 2020), ou não utilizam algoritmos de aprendizagem de máquina (Hippisley-Cox e Coupland, 2013). Assim, o principal desafio deste trabalho é o desenvolvimento de um modelo de previsão que seja capaz de prever internações para vários tipos de problemas de saúde utilizando métodos de aprendizagem de máquina. Com o objetivo de realizar essa tarefa, algumas questões relevantes devem ser devidamente abordadas:

- **Indisponibilidade de banco de dados com procedimentos realizados pelos pacientes:** Ao revisar a literatura, notou-se que a maioria dos trabalhos de previsão de internações não utilizam dados com os quais seja possível estabelecer a relação cronológica entre eles, fator este considerado importante neste trabalho, pois dependendo da técnica utilizada, permite estabelecer relações de causa e consequência e evoluções entre problemas de saúde, incorporando também contexto aos termos utilizados. Os dados utilizados por esses trabalhos normalmente referem-se a resultados de exames, uso de medicamentos contínuos, sinais vitais, presença de comorbidades, sintomas ou aspectos socio-econômicos e, na maioria das vezes, não são compartilhados com a comunidade científica. Assim, a obtenção de dados relacionados à saúde de pacientes que permitam estabelecer a relação cronológica entre eles foi um grande desafio.

- **Automatização da seleção e extração de características:** A seleção e extração de características é uma etapa que consome muito tempo, e na área de saúde é um fator ainda mais sensível, uma vez que características relevantes podem ser deixadas de lado por não fazerem parte, a princípio, de fatores de risco de um problema de saúde específico. Portanto, automatizar essa etapa pode ser importante para melhorar os modelos gerados.

- **Treinamento de LLMs (*Large Language Models*):** Treinar LLMs tornou-se um desafio crescente devido ao significativo tamanho que estes modelos alcançaram recentemente. Esse tamanho exige *hardware* cada vez mais poderoso e nem sempre acessível. Portanto, a utilização de técnicas que visam treinar modelos em *hardwares* mais modestos tornou-se um grande desafio.

1.3 HIPÓTESE

Considerando os aspectos anteriormente mencionados, o presente trabalho estabelece a seguinte hipótese básica: **"É possível criar um modelo de previsão de internações relacionado a diversos problemas de saúde utilizando dados históricos de procedimentos realizados por pacientes de planos de saúde"**. O propósito primário consiste em oferecer um sistema automático para auxiliar médicos, gestores de planos de saúde e hospitais na previsão de pacientes com risco de internação.

1.4 CONTRIBUIÇÕES

Este trabalho visa oferecer algumas contribuições científico-tecnológicas originais, dentre as quais, destacam-se:

- Planejamento, coleta, documentação, anonimização e publicação de um banco de dados com dados de eventos realizados em pacientes de planos de saúde organizados cronologicamente para casos de internação e não internação.
- Desenvolvimento de um processo automatizado de extração e seleção de características relacionada a dados de planos de saúde.
- Desenvolvimento de um modelo de previsão de internações eficiente e robusto baseado em dados de planos de saúde.
- Disponibilização de LLMs pré-treinadas para aplicações em diversos problemas da área da saúde.

1.5 ESTRUTURA DO DOCUMENTO

Este documento é composto por seis capítulos.

Neste capítulo, destacou-se a importância da previsão de internações e também foram delineados os objetivos, desafios e contribuições da pesquisa.

No Capítulo 2, apresenta-se ao leitor uma base sobre aspectos relacionados a terminologia e procedimentos relacionados a internações. Esse conhecimento é importante para o bom entendimento da pesquisa, da abordagem adotada e dos resultados alcançados. Além disso, com o objetivo de proporcionar ao leitor uma compreensão adequada das técnicas computacionais utilizadas neste estudo, também são apresentados conceitos normalmente utilizados em PLN, LLMs e algoritmos de aprendizagem de máquina selecionados. Apesar dessas intenções, assume-se que o leitor está familiarizado com a teoria geral relacionada a PLN e aprendizado de máquina, por isso, a revisão proposta não se pretende excessivamente detalhada.

Uma revisão do estado da arte para previsão de internações é apresentada no Capítulo 3. Abordagens de alguns trabalhos neste campo também são apresentados, além dos resultados alcançados por diferentes métodos.

O Capítulo 4 contém o modelo proposto neste trabalho juntamente com a metodologia que foi utilizada para validar os modelos construídos. Também são apresentados as bases de dados utilizadas, o processo de extração de características, as abordagens de treinamento das LLMs e os softwares utilizados.

Experimentos e resultados são apresentados no Capítulo 5. Inicialmente são detalhados os resultados por meio de métodos convencionais de aprendizado de máquina treinados a partir de três configurações diferentes dos dados. Em seguida os resultados dos modelos gerados a partir de LLMs, tanto para propósitos de previsões generalistas quanto para um problema de saúde específico com diferentes períodos de antecedência. Na sequência é apresentada uma análise do comportamento dos modelos gerados e uma discussão sobre os resultados de maneira geral.

No Capítulo 6 é apresentada a conclusão do trabalho e propostas de trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo apresentamos uma breve descrição das ferramentas teóricas utilizadas nesta tese, além de uma breve discussão sobre internações. Além disso, utilizou-se o problema de classificação de texto como guia de maneira a unificar conceitos, definições e modelos apresentados.

2.1 INTERNAÇÕES

De acordo com Ferrarini (Ferrarini, 1977), internação é a admissão de um paciente para ocupar um leito hospitalar. Já de acordo com a RDC nº 50, de 21 de fevereiro de 2002, proposta pela Agência Nacional de Vigilância Sanitária (ANVISA) (Ministério da Saúde, 2002), internação é a admissão de um paciente para ocupar o leito hospitalar, por um período igual ou maior que 24 horas. Também há a modalidade conhecida como hospital-dia que, segundo a RDC nº 50, é a modalidade de assistência à saúde, cuja finalidade é a prestação de cuidados durante a realização de procedimentos diagnósticos e/ou terapêuticos, que requeiram a permanência do paciente na unidade por um período de até 24 horas.

2.1.1 Internações evitáveis

Há doenças e condições relacionadas a internações que podem ser denominadas dentro de um grupo de internações evitáveis. Alguns trabalhos as denominam como hospitalizações evitáveis (Bindman, 1995; Culler et al., 1998), outros consideram como hospitalizações por condições sensíveis aos cuidados ambulatoriais (Yuen, 2004; Shi et al., 1999; Pappas et al., 1997). No Brasil, este grupo de doenças e condições foi estabelecido pela Portaria SAS/MS de nº 221, de 17 de abril de 2008 (Ministério da Saúde, 2008) e é o resultado do conceito de Internações por Condições Sensíveis à Atenção Primária (ICSAP), que são internações por doenças passíveis de controle e redução por meio da atenção básica acessível e efetiva, envolvendo prevenção e continuidade do cuidado. Ele é utilizado como indicador indireto para avaliar a qualidade da Atenção Primária à Saúde (APS). Esta lista pode ajudar na seleção de internações para serem investigadas a partir dos modelos gerados neste trabalho. Além disso, estudar as internações fora do escopo da ICSAP por meio dos modelos aqui propostos, pode resultar na identificação de novas condições que levam a essas internações, resultando talvez em novas políticas de saúde.

2.2 CLASSIFICAÇÃO DE TEXTO

Classificação de texto é um tópico de pesquisa desafiador devido à necessidade de organizar e categorizar o crescente número de documentos eletrônicos em todo o mundo. Ela pode ser aplicada em vários domínios, tais como: análise de sentimentos (Giménez et al., 2020), filtragem de spam (Kang e Yuan, 2014), identificação de autor (Vaithianathan et al., 2012) ou classificação de páginas da web (Buber e Diri, 2019) entre outras. Normalmente o processo de classificação de texto consiste de uma série de etapas, que resumidamente podem ser definidas em pré-processamento, representação do documento e classificação. A Figura 2.1 ilustra resumidamente essas etapas.

Figura 2.1: Esquema simplificado para classificação de texto.

2.3 PRÉ-PROCESSAMENTO DE TEXTO

Existem diversas técnicas de PLN para realizar o pré-processamento de texto, neste trabalho expomos os métodos apresentados em (Anandarajan et al., 2018), conforme a Figura 2.2.

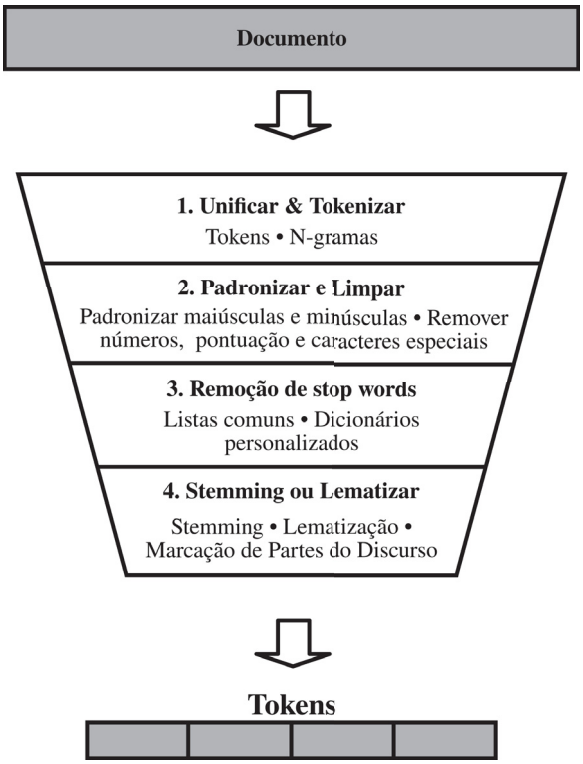


Figura 2.2: Etapas de pré-processamento de texto. Tradução da fonte (Anandarajan et al., 2018).

2.3.1 Tokenização

É o processo de decompor texto em termos que o compõe. Esses termos são chamados de *tokens*, que são sequências de caracteres entre dois espaços, ou entre um espaço e caracteres de pontuação. Para delimitar cada termo, normalmente são utilizados caracteres de quebras de linhas, espaços em branco entre outros. Normalmente, antes da tokenização o texto é todo convertido em letras minúsculas, para evitar que palavras iguais sejam tratadas como diferentes, como *"Antibiotics"* e *"antibiotics"*. Após a tokenização geralmente é realizada uma limpeza nos dados, em que são descartados os caracteres especiais. De modo geral, os caracteres especiais podem atrapalhar a geração de bons modelos e se encaixa na etapa de Padronização e Limpeza do processo de pré-processamento apresentado pela Figura 2.2. A Figura 2.3 exemplifica uma tokenização realizada sobre uma sentença de texto.

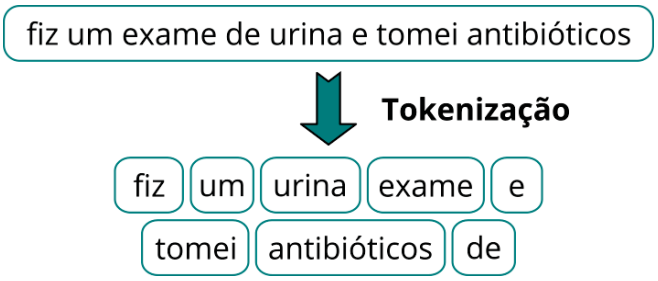


Figura 2.3: Exemplo de tokenização.

2.3.2 N-Gramas

N-Gramas é uma alternativa às palavras únicas do processo de tokenização. São tokens formados por sequências consecutivas de palavras com comprimento n. Por exemplo bigramas são tokens compostos por duas palavras que estão lado a lado no texto, uma única palavra é conhecida como unigrama. Uma das vantagens dos N-Gramas é que eles retêm informações sobre a co-ocorrência de palavras, pois agrupam palavas adjacentes ao mesmo token (Anandarajan et al., 2018). A Figura 2.4 apresenta exemplos de N-gramas para diferentes valores de n.

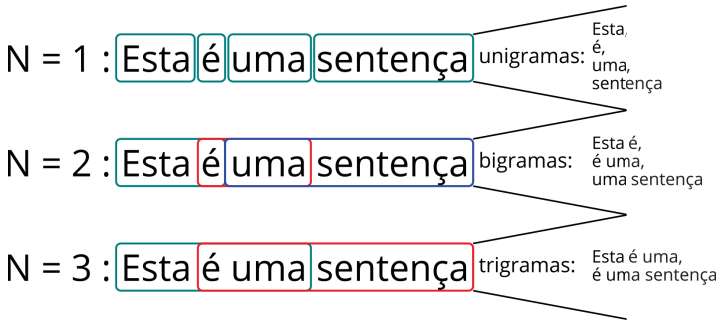


Figura 2.4: Exemplo de N-gramas.

2.3.3 Remoção de *stop words*

Stop Word ou palavra vazia é qualquer palavra em uma *stop list* que é filtrada, ou seja, removida antes ou depois do processamento de dados de linguagem natural (Rajaraman e Ullman, 2011). Não existe uma única e universal lista de palavras vazias usadas para todos as ferramentas de PLN. Normalmente a lista de palavras vazias inclui artigos, pronomes e preposições, como “e”, “dele”, “de” em português. Em alguns casos essas palavras, que geralmente são muito comuns em textos, parecem não fornecer muito valor quando o objetivo é o PLN e portanto poderiam ser removidos do texto.

Para remover as palavras vazias pode-se realizar a pesquisa em uma *stop list* e removê-las do texto, liberando assim espaço de memória e melhorando o tempo de processamento. Entretanto a remoção pode apagar dados relevantes e modificar o contexto de uma frase. Portanto é necessário tomar cuidado com a *stop list* escolhida. Uma opção é fazer testes empíricos para verificar se a remoção das palavras da lista de palavras vazias utilizada é benéfico ou não para o algoritmo.

2.3.4 *Stemming* ou Lematização

Lematização e *stemming* são técnicas de normalização cujo objetivo é relacionar palavras ou formas de palavras. Especificamente a lematização, tem a tarefa de encontrar a forma base, o lema[1], de uma determinada forma de palavra, já o *stemming* remove os afixos[2] de uma palavra e retorna o radical, a maior parte, comum e compartilhada por formas morfologicamente relacionadas. Essa normalização traz à tona relações gramaticais ou semânticas reais que de outra forma não seriam acessíveis por software, sendo assim, é importante em várias aplicações de PLN como classificação de texto e extração de informações (Korenius et al., 2004; Airio, 2006; Braschler e Ripplinger, 2004). A Tabela 2.1 exemplifica as diferenças dos resultados no processo de *Stemming* e Lematização.

Tabela 2.1: Exemplo de *Stemming* e Lematização.

Palavra original	**Stem**	**Lema**
amigos	amig	amigo
amigas	amig	amigo
amizade	amizad	amizade
carreira	carr	carreira
carreiras	carr	carreira

2.4 REPRESENTAÇÃO DO DOCUMENTO

A representação de documento é uma fase crucial em PLN pois preenche a lacuna entre o texto pré-processado e a capacidade da máquina interpretar, analisar e derivar significado desse texto. Como as máquinas precisam de ajuda para lidar com palavras, cada *token* (palavras) e/ou sequência de *tokens* (sentenças) precisam ser transformados para o formato numérico. Isso pode ser feito por meio de diferentes abordagens. Existem diversas abordagens que transformam os *tokens* em vetores numéricos como o *one-hot encoding* além de métodos que geram *word embeddings*. Existem também abordagens que buscam transformar toda a sentença em vetores numéricos como *Bag-of-Words* (BoW), *TF-IDF* e *Sentence embedding* (Barkan et al., 2019).

2.4.1 Codificação *One-Hot*

É a forma mais simples de representação de palavras/*tokens*, onde cada palavra é codificada como um vetor único. Cada palavra é representada como um vetor de zeros com um único número um no índice correspondente à palavra, entre todas as palavras possíveis, chamado no contexto de PLN de vocabulário. A Figura 2.5 exemplifica como se da essa representação para algumas palavras quando o método é empregado, e cujo vocabulário é $gato$, $tapete$, $esta$, no e o.

Para vocabulários grandes, esses vetores podem ficar muito longos pois contêm todas as posições do vetor com 0s, exceto um valor. Esta representação é considerada muito esparsa. Além disso, essa representação não captura nenhuma informação semântica dos *tokens*.

[1] Em linguística, lema ou lexia é a forma canônica de uma palavra.

[2] prefixos e sufixos

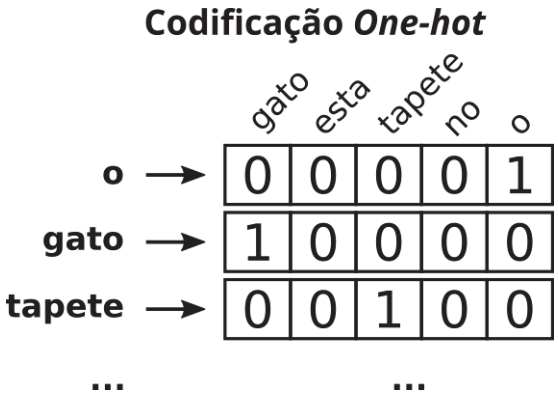


Figura 2.5: Exemplo de transformação de tokens para vetor utilizando codificação *One-hot*.

O vetor gerado tem a função unicamente de representar um *token* e portanto pode-se dizer que é um tipo rudimentar de *word embedding*. Para utilização em sentenças de texto torna-se necessário concatenar os vetores *one-hot* de cada palavra.

2.4.2 Codificação de cada palavra em um número único

Outra abordagem utilizada é a codificar cada palavra usando um único número. Considere a sentença "O gato esta no tapete", para ela pode-se atribuir 1 para "gato", 2 para "tapete"e assim por diante. Assim, pode-se codifica essa sentença como um vetor denso para representá-la como [3, 1, 4, 5, 2]. Portanto, ao invés de um vetor esparso, agora pode-se ter um vetor cheio, ou seja, onde todos os elementos estão preenchidos.

No entanto, existem duas desvantagens nessa abordagem:

- Ela é arbitrária, ou seja, não captura nenhuma relação entre as palavras.
- Uma codificação inteira pode ser um desafio para um modelo interpretar. Um classificador linear, por exemplo, aprende um único peso para cada recurso. Como não há relação entre a semelhança de duas palavras e a semelhança de suas codificações, essa combinação de peso de recurso não é significativa.

2.4.3 *Bag-of-Words* e TF-IDF

Uma outra forma de representar uma sentença é usando a frequência de palavras. Considere D como representando um conjunto de todas as entradas de texto (corpus), e seja $V = (w_1, ..., w_V)$ a sequência de todos os tipos de palavras únicas que aparecem no corpus ordenados por sua frequência (chama-se essa sequência de vocabulário). A representação *bag-of-words* (BoW) do texto T_i é definida como:

$$X_i = (X_{i,1}, ..., X_{i,V}), \tag{2.1}$$

onde $X_{i,j}$ captura algumas informações de frequência sobre a palavra w_j no texto T_i . Normalmente escolhemos um dos seguintes tipos de representação:

- $X_{i,j} \in 0, 1$ indica a ausência ou presença da palavra w_j no texto T_i.
- $X_{i,j} \in N$ indica o número de ocorrências da palavra w_j no texto T_i, (representa o BoW mais comum).

- $X_{i,j} \in R$ indica a frequência da palavra w_j no texto T_i redimensionada pela frequência com que esta palavra aparece em todos os documentos (esta métrica é conhecida como TF-IDF ou *Term Frequency-Inverse Document Frequency*).

  O TF-IDF pode ser calculado conforme as equações:

$$TF(t,d) = \frac{\text{número de vezes que } t \text{ aparece em } d}{\text{número total de termos em } d}, \tag{2.2}$$

$$IDF(t) = \log\left(\frac{N}{df}\right), \tag{2.3}$$

$$TF - IDF(t,d) = TF(t,d) * IDF(t), \tag{2.4}$$

onde d refere-se ao documento, N é o número total de documentos e df é o número de documentos com o termo t. Graficamente o BoW pode ser demonstrado conforme a Figura 2.6 e o TF-IDF conforme a Figura 2.7.

Sentenças	Bag-of-Words							
	deles	documento	é	este	primeiro	o	segundo	terceiro
['Este é o primeiro documento',	0	1	1	1	1	1	0	0
'Este documento é o segundo documento',	0	2	1	1	0	1	1	0
'E este é o terceiro deles']	1	0	1	1	0	1	0	1

Figura 2.6: Exemplo de conversão de sentenças usando BoW.

Sentenças	TF-IDF							
	deles	documento	é	este	primeiro	o	segundo	terceiro
['Este é o primeiro documento',	0	0.081	0	0	0.21	0	0	0
'Este documento é o segundo documento',	0	0.16	0	0	0	0	0.21	0
'E este é o terceiro deles']	0.21	0	0	0	0	0	0	0.21

Figura 2.7: Exemplo de conversão de sentenças usando TF-IDF.

Como o tamanho do vocabulário pode ser grande, esse processo pode gerar uma quantidade substancial de recursos. Portanto, é normal combinar o método BoW e TF-IDF com técnicas de redução de dimensionalidade, como análise de componentes principais (PCA) (Abdi e Williams, 2010), para reduzir a dimensão dos dados de entrada.

2.4.4 *Word embeddings*

Word embeddings são métodos que geram representações de palavras em um espaço de vetores de alta dimensão. Diferentemente do *one-hot encoding* estes métodos são capazes de capturar o contexto de uma palavra a partir do treinamento sobre um conjunto de sentenças. Ele se tornou popular em PLN a partir do artigo de Tomas Mikolov (*Google*) (Mikolov et al., 2013). Resumidamente, as palavras são convertidas em vetores que são usados para representar a palavra em um espaço multidimensional, no qual as palavras similares ficam próximas umas das outras. Isso permite que os modelos de aprendizado de máquina consigam compreender o significado das palavras e como elas estão relacionadas. A Figura 2.8 ilustra algumas relações semânticas a partir *embeddings* geradas pelo método Word2Vec (Lai et al., 2015) em um vetor tridimensional.

O contexto das palavras são normalmente incorporados ao vetor por meio de métodos de aprendizado auto-supervisionado, isso significa que o modelo é alimentado com grandes

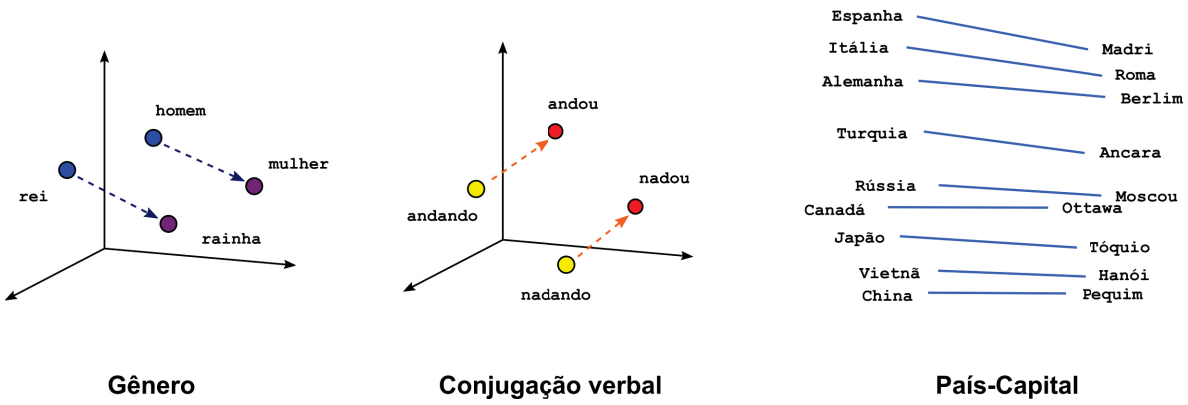


Figura 2.8: Ilustração de algumas relações semânticas e sintáticas capturadas por embeddings word2vec, vetores cujas palavras têm relações semelhantes (como gênero ou conjugação) tendem a também ter relações semelhantes no espaço vetorial.

conjuntos de dados de texto, como notícias, livros ou conversas, e aprende a associar cada palavra a um vetor baseado nas palavras que aparecem ao seu redor. Os modelos de *embedding* como o Word2Vec(Lai et al., 2015) e GloVe (Pennington et al., 2014) usam diferentes técnicas para aprender essas associações, entretanto, a ideia básica é que as palavras que aparecem frequentemente juntas em um contexto similar terão vetores similares. Por exemplo, as palavras "homem" e "mulher" terão vetores similares, pois são palavras relacionadas e tendem a aparecer em contextos semelhantes.

É importante destacar que para os métodos Word2Vec e GloVe, os *embeddings* para cada palavra são sempre os mesmos, ou seja, apesar de incorporar o contexto considerando os dados de treinamento, após o treinamento, para qualquer sentença os *embeddings* das palavras assumem sempre os mesmos valores, independente do contexto daquela sentença específica. Buscando resolver essa questão alguns modelos de *embedding* baseados nos *Transformers* (Vaswani et al., 2017) como BERT (Devlin et al., 2018), GPT-2 (Radford et al., 2019), e ELMo (Peters et al., 2018) , usam a técnica de aprendizado auto-supervisionado e um mecanismo de auto-atenção para compreender o contexto das palavras, esses modelos são chamados de *contextual embedding*, e geram *embeddings* para as palavras de acordo com o contexto em que ela aparece na sentença. Devido o uso desses modelos neste trabalho, a explicação sobre esses modelos é feita nas próximas seções.

2.4.5 Sequencia dos Dados

Dados sequenciais podem ser denominados quando os pontos no conjunto de dados são dependentes de outros pontos no conjunto. Alguns exemplos de dados sequenciais são sequencias de DNA (Gunasekaran et al., 2021), dados meteorológicos (Kang et al., 2020) e texto (Jang et al., 2020). Texto é uma das formas mais difundidas de dados em sequência. Ele pode ser entendido como uma sequência de caracteres ou de palavras, embora mais comumente utilizado como sequência de palavras.

Para classificação de sequências é necessário que toda a sentença seja transformado em valores numéricos, como apresentado anteriormente, os métodos BoW e TF-IDF tem essa capacidade de transformação, embora incorpore nenhum ou pouco contexto das sequências. Entretanto, os métodos de *word embeddings*, apesar de conseguirem agregar o contexto em que os tokens aparecem, geram apenas representações para os tokens, desconsiderando a representação de toda a sequência. Portanto é necessário concatená-los para que possam ser usados por algoritmos de classificação. Muitos métodos de *Deep-Learning* são utilizados para processamento de

dados em sequência, como é o caso das Redes Neurais Recorrentes ou RNNs (*Recurrent neural networks*) (Rumelhart et al., 1986), ou LSTM (*Long Short Term Memory*) (Sherstinsky, 2020). Assim como todas as redes neurais, os modelos de *Deep-Learning* não aceitam texto bruto de entrada e funcionam com vetores numéricos. Portanto, transformar a sequência de texto em uma representação numérica se faz necessário. A vetorização de texto é uma forma de transformar texto em sequência de números buscando preservar a sequência em que ocorrem.

Como já discutimos neste capítulo, processos de vetorização de texto consistem em aplicar algum esquema de tokenização e então associá-los a vetores numéricos. Esses vetores são empacotados em tensores de sequência que podem ser utilizados para alimentar redes neurais profundas. Existem múltiplas formas de associar um vetor a um *token*, tais como os métodos apresentados anteriormente *one-hot-encoding* (Dahouda e Joe, 2021) e *word2vec* (Goldberg e Levy, 2014). A API Tensorflow (Abadi et al., 2016) e Keras (Lee e Song, 2019) fornecem mecanismos automáticos e formas de realizar a transformação dos dados em tais vetores, oferecendo mecanismos para realizar todo o processo de pré-processamento. Nestas bibliotecas, o pré-processamento de texto incluem as etapas de tokenização, sequenciamento e preenchimento. A Figura 2.9 apresenta a sequência adotada por essas bibliotecas do pré-processamento de texto, em que não utiliza-se *word embedding*. Entretanto, este mesmo processo pode ser aplicado substituindo os índices das palavras pelos *word embeddings* gerados por algum método, formando assim matrizes de *embeddings* prontos para serem utilizados por métodos de *Deep-Learning* por exemplo.

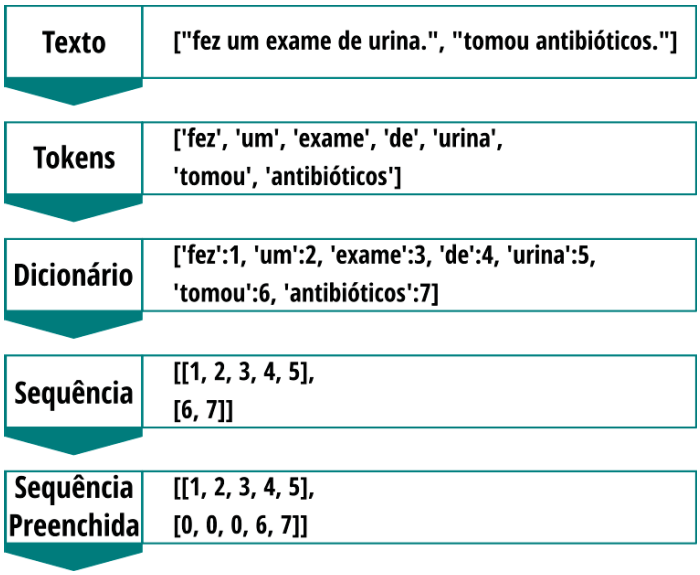


Figura 2.9: Passo a passo do pré-processamento de texto com *keras* e *tensorflow*.

Além do processo de tokenização apresentado na seção 2.3.1 a API também fornece mecanismos para representação das sentenças em uma sequências, preenchimento e truncamento, além de outros recursos.

2.4.5.1 Preenchimento (padding)

Em qualquer texto naturalmente haverá sentenças de tamanhos diferentes. Entretanto, as redes neurais requerem entradas com o mesmo tamanho. Por este motivo o preenchimento deve ser feito, podendo ser realizado com um valor numérico que não está relacionado a nenhum dos tokens pertencentes do dicionário gerado, sendo normalmente utilizado o número 0. Esse preenchimento pode ser feito antes ou após a sequência, dependendo na necessidade da aplicação.

O preenchimento no início da sequência é chamado de pré preenchimento (*pre padding*); ao fim da sequência é chamado de pós preenchimento (*post padding*). A Figura 2.10 exemplifica o pré e pós preenchimento.

Figura 2.10: Pré e pós preenchimento.

2.4.5.2 Truncamento (truncating)

Assim como é normal que sentenças tenham tamanhos diferentes, é comum que existam sentenças muito grandes. Dependendo da aplicação, um tamanho máximo para as sentenças pode ser definido, e nos casos em que a sentenças ultrapassem esse tamanho pode ser aplicado o truncamento, que permite eliminar uma parte da sequência mantendo o tamanho máximo determinado. Como no preenchimento, o truncamento pode ser feito no início ou no fim da sentença, sendo definido como pré truncamento ou pós truncamento, respectivamente. O truncamento pode ser aplicado juntamente com o preenchimento mantendo assim a uniformidade no comprimento das sequências. A Figura 2.11 apresenta o pré e pós truncamento aplicado conjuntamente com o preenchimento.

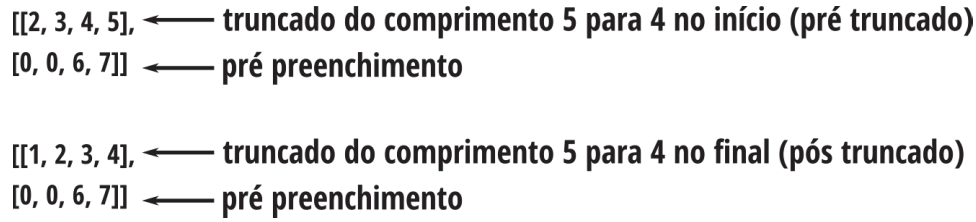


Figura 2.11: Pré e pós truncamento.

2.5 *LARGE LANGUAGE MODELS*

Nos últimos anos, a PLN tem experimentado avanços significativos. Uma das inovações mais notáveis foi o surgimento dos LLMs (*Large Language Model*), que vêm alcançando o estado da arte em várias tarefas desta área. Os LLMs apresentam uma arquitetura de rede neural chamada *transformer* que foi introduzida em 2017 por Vaswani (Vaswani et al., 2017), sendo altamente escalável e eficiente, permitindo que os LLMs sejam treinados em grandes volumes de dados. Neste trabalho, foram utilizados três LLMs diferentes, todos baseados na arquitetura do *transformers*, portanto, apresenta-se nas próximas subseções conceitos sobre *transformes* e as LLMs BERT (Devlin et al., 2018), RoBERTa (Liu et al., 2019) e LLaMA (Touvron et al., 2023b).

2.5.1 *Transformers*

Transformers (Vaswani et al., 2017) é uma arquitetura do tipo *encoder-decoder* (Badrinarayanan et al., 2017), que depende principalmente de mecanismos de atenção. Antes do surgimento do *Transformer*, a arquitetura mais utilizada para PLN era a *Rede Neural Recorrente* (RNN), que processa textos de maneira sequencial, dificultando a paralelização. Para resolver

esta questão, a arquitetura *Transformer* apresentou uma estrutura de auto-atenção, permitindo a paralelização computando as operações de todos os *tokens* de uma sentença ao mesmo tempo. A auto-atenção é capaz de captar informações semânticas fortes, por exemplo na sentença, "Maria comprou um biscoito e o comeu" o modelo de auto-atenção é capaz de determinar que o *token* "o" se refere ao *"biscoito"*.

A arquitetura do *Transformers* é dividida em duas grantes partes, o *Encoder* e o *Decoder*. O *Encoder* começa usando o mecanismo auto-atenção nos *embeddings* de palavras para agregar informações de cada *token*, criando uma nova representação rica em contexto para cada palavra de maneira simultânea. O *Decoder*, por outro lado, adota uma abordagem iterativa. Por exemplo, para tarefa de tradução, em vez de produzir a sentença traduzida inteira de uma vez, gera uma palavra por vez, então, o *decoder* utiliza a representação final produzida como entrada do próprio *decoder* para gerar a próxima palavra até prever a TAG <*END*>, indicando o final da sentença. Em essência, ele receberá a sentença a ser traduzida como entrada e produzirá a sentençça traduzida, uma palavra por vez. As Figuras 2.12 e 2.13 ilustram a arquitetura do modelo do transformador e do mecanismo de auto-atenção.

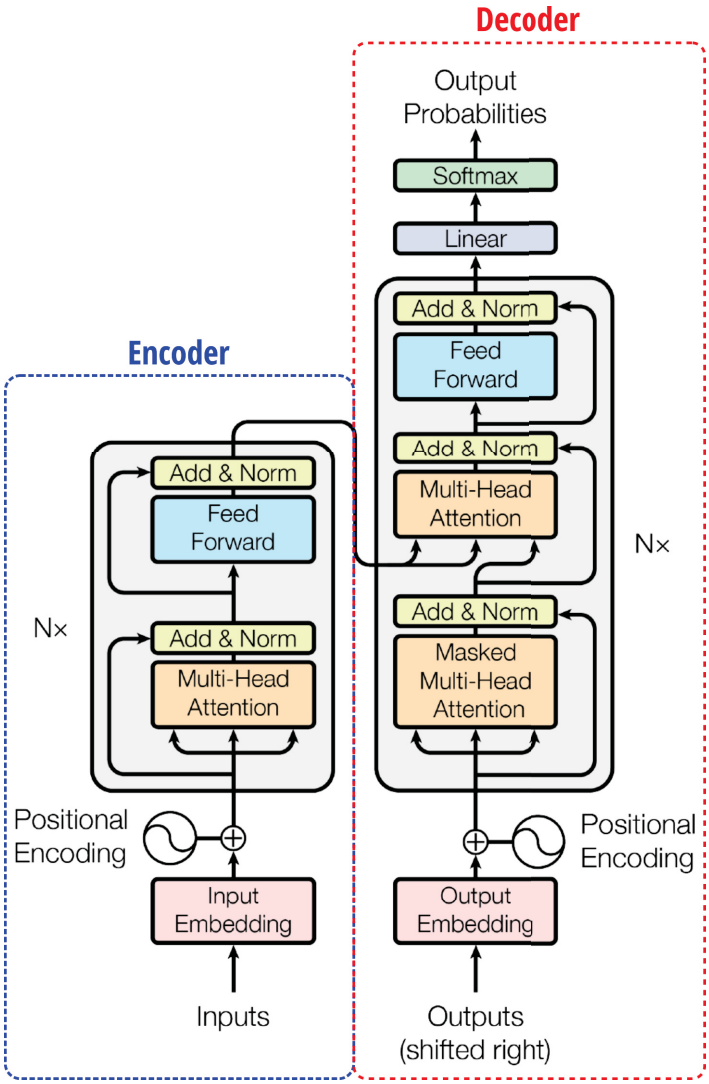


Figura 2.12: Arquitetura do modelo *Transformer*. Fonte (Vaswani et al., 2017).

A arquitetura do *Transformers* alcançou desempenho do estado da arte em tarefas de PLN e, devido ao seu mecanismo de atenção, foi capaz de ser treinado significativamente mais

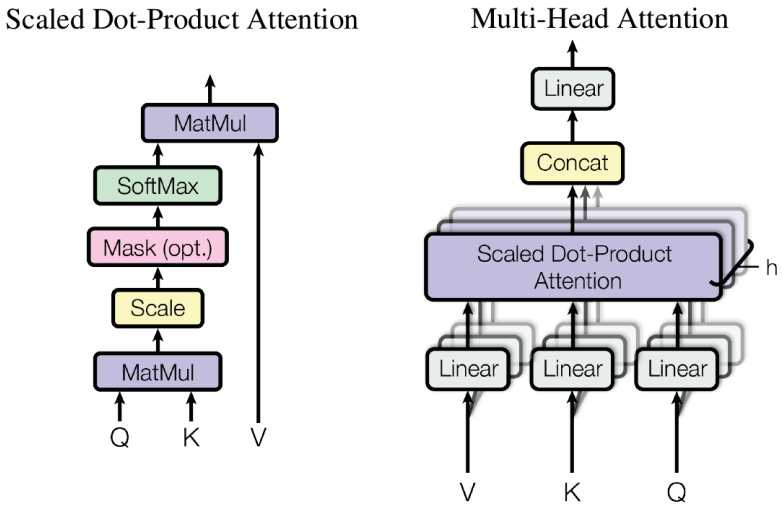


Figura 2.13: À esquerda, produto escalar do mecanismo de atenção. À direita, a *Multi-Head Attention* consiste em várias camadas de atenção funcionando em paralelo. Fonte (Vaswani et al., 2017).

rápido do que outros modelos centrados em camadas recorrentes ou convolucionais. Isso levou a uma série de novos modelos baseados na arquitetura do *Transformer* e na ideia de que atenção é tudo que você precisa. Uma variante dessa arquitetura é o BERT (Devlin et al., 2018), que rapidamente atingiu o estado da arte em diversas tarefas de PLN, como tarefas de classificação.

Os modelos que utilizam *Transformers* e consequentemente o mecanismo de auto-atenção, são capazes de considerar o contexto da ocorrência de uma palavra, diferentemente dos *embeddings* independentes de contexto mencionados anteriormente, word2vec e GloVe. Por exemplo, o vetor para "ando" teria diferentes *embeddings* no BERT para as sentenças "Eu ando cansada" e "Eu ando de tênis", enquanto word2vec produziria exatamente o mesma *embedding*. Essa forma de produzir *embeddings* do *Transformers* justifica mantermos a cronologia dos dados usados nesta pesquisa.

2.5.2 BERT

BERT (*Bidirectional Encoder Representations from Transformers*) (Devlin et al., 2018) é uma rede neural baseada no *encoder* da arquitetura do *transformers* (Vaswani et al., 2017). Ele foi projetado para pré-treinar representações bidirecionais profundas de textos não rotulados, condicionando conjuntamente o contexto esquerdo e direito em todas as camadas da rede, diferentemente por exemplo do GPT (Radford et al., 2019), que utilizam a arquitetura da esquerda para direita, em que todo token pode somente atender aos tokens anteriores nas camadas de auto-atenção do *Transformer*. A Figura 2.14 demonstra essa diferença. Nas próximas subseções apresenta-se alguns conceitos importantes da arquitetura do BERT além de tipos de treinamentos.

2.5.2.1 Representações de entrada e saída

Segundo os autores, para fazer o BERT lidar com uma variedade de tarefas, a representação de entrada é capaz de representar inequivocamente uma única sentença ou um par de sentenças do tipo pergunta e resposta em uma sequência de tokens. Para o BERT, "sequência" refere-se a uma entrada de tokens em sequência, que pode ser uma única sentença ou duas sentenças agrupadas.

O primeiro token de cada sequência é sempre um token especial de classificação ($[CLS]$), esse token é usado como representação da sequência agregada, ou seja, de toda a

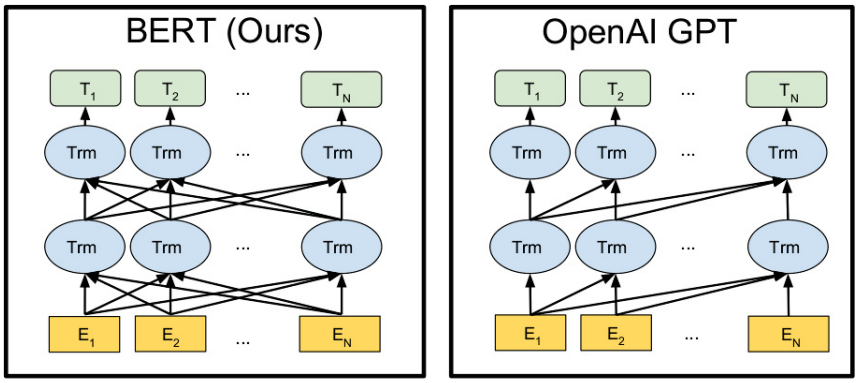


Figura 2.14: Diferenças nas arquiteturas de modelos para pré-treinamento. BERT que usa um *Transformer* bidirecional e o GPT da OpenAI que usa um *transformer* da esquerda para a direita. Fonte (Devlin et al., 2018).

sentença. Ele é muito utilizado para tarefas de classificação. Este token tem a capacidade de armazenar em seu estado, a representação de toda sentença. Os pares de sentenças são agrupados juntos em uma única sequência. Eles diferenciam as sentenças de duas maneiras. Primeiramente separam-se as duas sentenças com um token especial ($[SEP]$). Depois, adicionam um *embedding* para cada token indicando se ele pertence à sentença A ou à sentença B. Portanto, para um determinado token, sua representação de entrada é construída somando os correspondentes *embeddings* do token, segmento, e posição. A Figura 2.15, apresenta a composição desses vetores que representam a entrada do modelo.

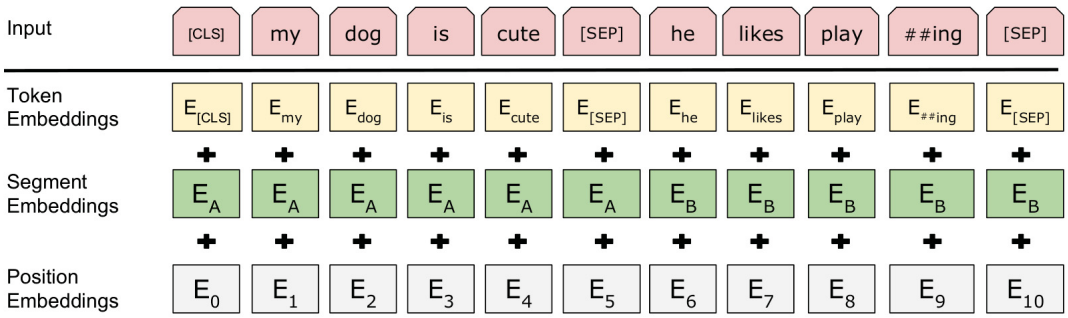


Figura 2.15: Representação de entrada do BERT. Os *embeddings* de entrada são a soma dos *embeddings* de token, dos *embeddings* de segmentação e dos *embeddings* de posição. Fonte (Devlin et al., 2018).

2.5.2.2 Pré treinamento

O BERT pemite seu pré-treinamento a partir de duas tarefas auto-supervisionados, o *Masked Language Model* (MLM) e o *Next Sentence Prediction* (NSP).

No treinamento pela tarefa MLM simplesmente mascara-se aleatoriamente alguma porcentagem dos tokens de entrada e, em seguida é feita a previsão desses tokens mascarados. Assim, para fazer a previsão, os vetores de saída correspondentes aos tokens mascarados são passados para uma saída softmax sobre o vocabulário. A porcentagem de tokens escolhidas para treinar o BERT foi de 15%. Embora esse treinamento permita obter um modelo bidirecional pré-treinado, uma desvantagem é que se cria uma incompatibilidade entre o pré-treinamento e o *fine-tuning*, pois o token $[MASK]$ não aparece no *fine-tuning*. Assim, para mitigar esse problema, não é sempre que um token selecionado para ser "mascarado"de fato recebe o token $[MASK]$. Neste caso, o gerador de dados para treinamento quando seleciona o i-ésimo token

substitui ele 80% das vezes pelo token $[MASK]$, 10% das vezes por um token aleatório, e 10% das vezes não altera o token. Então, o token T_i será usado para predizer o token original com perda de entropia cruzada.

Muitas tarefas importantes, como a de Perguntas e Respostas (QA - *Question Answering*) e Inferência em Linguagem Natural (NLI - *Natural Language Inference*) baseiam-se no entendimento da relação entre duas sentenças. Para treinar um modelo que entenda essas relações, um treinamento binário para NSP por ser trivialmente gerado a partir de qualquer corpus monolíngue. No treinamento, ao escolher as sentenças A e B para cada exemplo determina-se que 50% das vezes B é a próxima sentença real de A (rotulada como *IsNext*) e 50% das vezes é uma sentença aleatória (rotulada como *NotNext*). No trabalho do (Devlin et al., 2018) é demonstrado que esse treinamento melhora muito os resultados tanto para tarefa QA quanto para NLI.

2.5.2.3 Fine-tuning

O *fine-tuning* no BERT depende de qual tarefa pretende-se ajustar os parâmetros do modelo. Para cada tarefa, simplesmente conecta-se as entradas e saídas ao modelo BERT e ajusta-se todos os parâmetros de ponta a ponta. Se comparado ao pré-treinamento, o *fine-tuning* é relativamente menos caro em processamento, conforme discutido na seção 2.6. Em tarefas do tipo QA, o modelo recebe uma pergunta e uma sequência de texto e mascara-se a resposta na sequência. Usando o BERT, um modelo de QA pode ser treinado aprendendo dois vetores extras que marcam o início e o fim da resposta. Em tarefas do tipo *Named Entity Recognition* (NER), o modelo recebe uma sequência de texto e deve a marcar os vários tipos de entidades que aparecem no texto (Pessoa, Organização, Data, etc.). Usando BERT, um modelo NER pode ser treinado alimentando o vetor de saída de cada token em uma camada de classificação que prevê o rótulo NER. Para tarefa de classificação, como análise de sentimentos, uma camada completamente conectada (FC - Fully Connected Layer) pode ser conecta ao token especial de classificação $[CLS]$ que prevê o rótulo da classificação. A Figura 2.16 exemplifica o *fine-tuning* para tarefa de classificação.

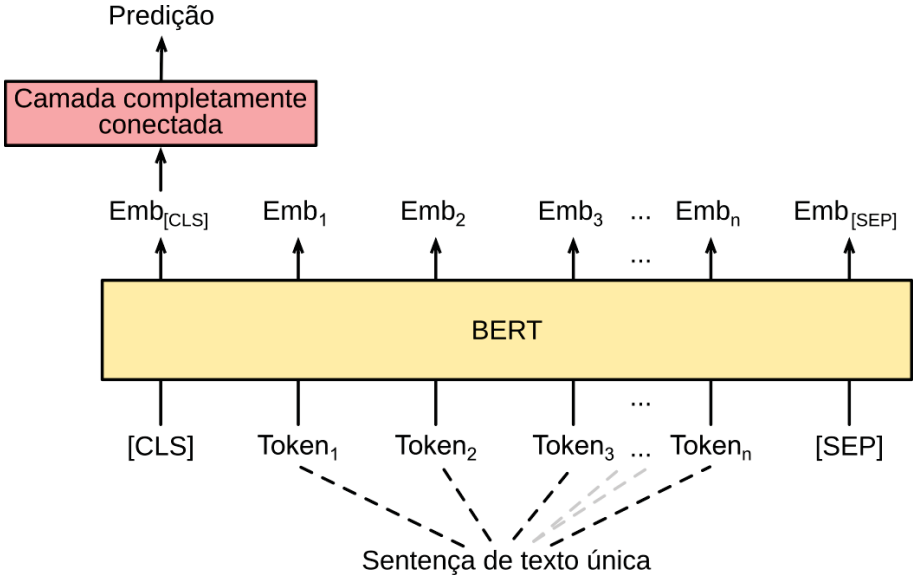


Figura 2.16: Exemplo de *fine-tuning* para tarefa de classificação com a camada completamente conectada ao token $[CLS]$.

2.5.3 RoBERTa

RoBERTa (*A Robustly Optimized BERT Pretraining Approach*) foi proposto por Liu et al. (Liu et al., 2019). Ele modifica os principais hiperparâmetros do BERT, remove o objetivo de pré-treinamento da próxima frase (NSP) e utiliza taxas de aprendizagem muito maiores. Essas e outras otimizações foram realizadas permitindo ao modelo alcançar o estado da arte em várias tarefas de PLN, além de reduzir o tempo de treinamento.

Pelo fato do RoBERTa manter a mesma arquitetura do BERT o método de pré-treinamento dele é o mesmo do BERT exceto pela tarefa NSP. Pelo mesmo motivo o *fine-tuning* também é feito como no BERT.

2.5.4 LLaMA

O LLaMA (*Large Language Model Meta AI*) é um LLM lançado pela *Meta AI* (Touvron et al., 2023b) que têm consistentemente alcançado o estado da arte em várias tarefas de PLN. No projeto original são disponibilizados modelos que variam de 7 bilhões a 65 bilhões de parâmetros. Diferentemente do BERT e do RoBERTa, que permitem entradas de até 512 *tokens*, ele aceita entradas de até 2048 *tokens*, algo importante para a análise proposta nesta pesquisa. Ele também é baseado na arquitetura do *Transformers* e incluí várias melhorias que foram propostas posteriormente e utilizadas em diferentes modelos, como o PaLM (Chowdhery et al., 2022). Como principais diferenças relacionadas do *Transformers* e as fontes de inspiração para das mudança da arquitetura, tem-se:

- Pré-normalização [GPT3] - Para melhorar a estabilidade do treinamento, normalizou-se a entrada de cada subcamada do *Transformers*, em vez de normalizar a saída. Usou-se a função de normalização RMSNorm, proposta por (Zhang e Sennrich, 2019).

- Função de ativação SwiGLU [PaLM] - Substituiu-se a não linearidade ReLU pela função de ativação SwiGLU, introduzida por (Shazeer, 2020) para melhorar o desempenho. A dimensão de $\frac{2}{3}4d$ em vez de $4d$ como no PaLM.

- *Embeddings* Rotativas [GPTNeo] - Removeu-se os *embeddings* posicionais absolutos e, em vez disso, adicionou-se *embeddings* posicionais rotativos (RoPE), proposto por (Su et al., 2021), em cada camada da rede.

Além disso, devido ao grande tamanho das estruturas do LLaMA, são necessários extensíveis recursos computacionais para treinamento e *fine-tuning*. Desta forma, considerando os recursos de *hardware* disponíveis para esta pesquisa, a utilização de PEFT (*Parameter Efficient Fine-Tuning*) (Han et al., 2024) foi necessária, pois reduz a quantidade de recursos computacionais necessários nos treinamentos. Estas técnicas apresentam uma solução prática para o *fine-tuning* e é melhor detalhado na seção 2.7.

2.6 *TRANSFER LEARNING*

Transfer Learning é uma técnica de aprendizado de máquina na qual o conhecimento adquirido por meio de uma tarefa ou conjunto de dados é usado para melhorar o desempenho do modelo em outra tarefa relacionada a um conjunto de dados diferente. Resumidamente, o *Transfer Learning* usa o que foi aprendido em um ambiente para melhorar a generalização em outro ambiente. O *Transfer Learning* tem muitas aplicações, desde a resolução de problemas de regressão até o treinamento de modelos de *deep learning* e é muito utilizado quando se trabalha

com LLMs. Usando a definição de (Pan e Yang, 2010), temos que, dado uma fonte de domínio D_S e uma tarefa de aprendizagem T_S, um domínio alvo D_T e uma tarefa de aprendizagem T_T, o *Transfer Learning* visa ajudar a melhorar a função preditiva alvo $f_T(.)$ em D_T utilizando o conhecimento em D_S e T_S, onde $D_S \neq D_T$ ou $T_S \neq T_T$.

Em PLN essa técnica é vantajosa principalmente quando se utiliza LLMs porque:

- **Melhora a eficiência no uso de dados**: os modelos de LLMs geralmente exigem grandes quantidades de dados para funcionar bem. O *Transfer Learning* permite que os modelos sejam pré-treinados em um grande corpus de texto, como a Wikipedia, e depois ajustados em um conjunto de dados menor e específico para outra tarefa. Isso reduz a necessidade de uma enorme quantidade de dados rotulados para cada tarefa específica.

- **Gera economia de recursos**: treinar modelos de linguagem em larga escala a partir do zero pode ser computacionalmente caro e demorado. Ao começar com um modelo pré-treinado, o processo de *fine-tunning* requer menos recursos, tornando-o mais acessível.

- **Proporciona melhoria de desempenho**: modelos pré-treinados já aprenderam recursos e padrões linguísticos úteis de grandes quantidades de texto. O *fine-tunning* desses modelos em uma tarefa específica geralmente leva a um desempenho melhorado em comparação ao treinamento de um modelo do zero, especialmente quando a tarefa tem uma quantidade limitada de dados rotulados.

- **Permite aprendizado contínuo:** depois que um modelo é treinado, ele pode ser facilmente atualizado ou adaptado com novos dados, permitindo que ele aprenda continuamente e melhore seu desempenho ao longo do tempo.

2.7 *PARAMETER-EFFICIENT FINE-TUNING*

Muitas aplicações em PLN dependem da adaptação de um modelo de LLM pré-treinado. Tal adaptação geralmente é feita via *fine-tuning*, que atualiza todos os parâmetros do modelo pré-treinado. A principal desvantagem do *fine-tuning* é que o modelo ajustado contém tantos parâmetros quanto o modelo original. À medida que modelos cada vez maiores são treinados a cada poucos meses, isso deixa de ser um mero inconveniente para um desafio crítico de treinamento.

O *Parameter-Efficient Fine-Tuning* (PEFT) (Han et al., 2024) é uma abordagem que fornece uma solução prática para ajustar com eficiência grandes modelos para várias tarefas. Em particular, PEFT refere-se ao processo de ajuste dos parâmetros de grandes modelos pré-treinados para adaptá-los a uma tarefa ou domínio específico, minimizando ao mesmo tempo o número de parâmetros adicionais introduzidos e também recursos computacionais necessários. Esta abordagem é particularmente importante quando se lida com LLMs com número elevado de parâmetros, como o LLaMA (Touvron et al., 2023b) com modelos de até 65 bilhões de parâmetros, ou o GPT-3 (Brown et al., 2020) com 175 bilhões de parâmetros.

Quando se lida com LLMs há diversas abordagens PEFT, que atuam de maneira diferente nos modelos, tais como IA3 (Liu et al., 2022), *Prefix tuning* (Li e Liang, 2021) e LoRA (Hu et al., 2021). Neste trabalho utilizou-se uma extensão do LoRA chamado QLoRA (Dettmers et al., 2023) para fazer o *fine-tuning* do OpenLLaMA3B (Larcher et al., 2023).

2.7.1 LoRA

O LoRA (*Low-Rank Adaptation Of Large Language Models*) é inspirado nos trabalhos de (Li et al., 2018) e (Aghajanyan et al., 2020) que mostram que os modelos sobre-parametrizados aprendidos de fato estão em uma dimensão intrínseca baixa. Assim, o LoRA foi desenvolvido com base na hipótese de que a mudança nos pesos durante *fine-tuning* do modelo tem uma baixa "classificação intrínseca" (*intrinsic rank*). Desta forma, o LoRA introduz pequenos módulos de adaptação ao modelo pré-treinado e então treina somente os parâmetros destes módulos, mantendo os parâmetros do modelo pré-treinado congelados, como mostrado pela Figura 2.17.

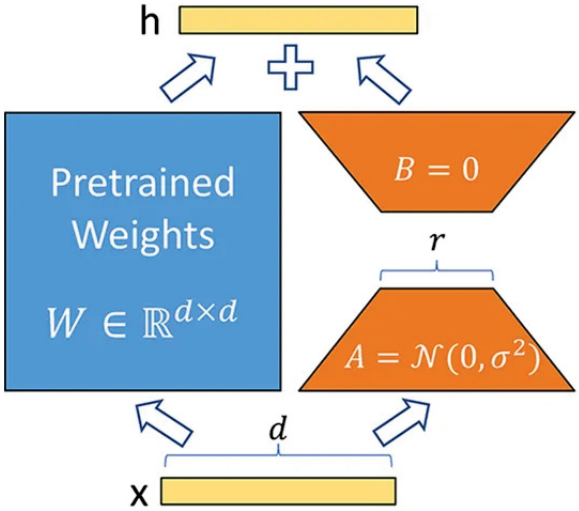


Figura 2.17: Reparametrização do LoRA em que apenas A e B são treinados. Fonte (Hu et al., 2021).

Segundo Liu et al. (Hu et al., 2021) o LoRA possui várias vantagens importantes, tais como:

- Um modelo pré-treinado pode ser compartilhado e usado para construir muitos pequenos módulos LoRA para diferentes tarefas. Pode-se congelar o modelo compartilhado e alternar tarefas com eficiência, substituindo as matrizes A e B na Figura 2.17, reduzindo significativamente o requisito de armazenamento e a alternância de tarefas.
- LoRA torna o treinamento mais eficiente e reduz a barreira de entrada de *hardware* em até 3 vezes ao usar otimizadores adaptativos, uma vez que não é preciso calcular os gradientes ou manter os estados do otimizador para a maioria dos parâmetros. Em vez disso, otimiza-se apenas as matrizes injetadas, muito menores e de classificação baixa (*low-rank*).
- O *design* linear simples permite mesclar as matrizes treináveis com pesos congelados quando implantados, não introduzindo nenhuma latência de inferência em comparação com um modelo totalmente ajustado, por construção.
- LoRA é ortogonal a muitos métodos anteriores e pode ser combinado com muitos deles, como o *prefix-tuning*.

2.7.2 Quantização em LLMs

QLoRA (*Quantized Low Rank Adaptation*) é a versão estendida de LoRA que funciona quantificando a precisão dos parâmetros de peso do LLM pré-treinado para precisão de 4 bits.

Normalmente, os parâmetros de modelos treinados são armazenados em um formato de 32 bits, mas o QLoRA comprime para um formato de 4 bits. Isso reduz o espaço de memória do LLM, tornando possível ajustá-lo em uma única GPU. Esse método reduz significativamente o espaço de memória, tornando possível executar modelos de LLM em *hardware* menos potente.

O QLoRA introduz várias inovações projetadas para reduzir o uso de memória sem sacrificar o desempenho:

- Quantização *NormalFloat* de 4 bits - um tipo de dados de quantização que produz melhores resultados empíricos do que *Integers* de 4 bits e *Floats* de 4 bits.
- Quantização *Double* - um método que quantifica as constantes de quantização, economizando uma média de cerca de 0,37 bits por parâmetro (aproximadamente 3 GB para um modelo 65B)
- Otimizadores de página - utiliza a memória unificada NVIDIA para evitar picos de memória de ponto de verificação do gradiente, que ocorrem quando processa-se um mini-lote com sequência de comprimento longo.

Em um experimento no trabalho (Dettmers et al., 2023), o QLoRa mostrou ser capaz em reduzir de 780GB para menos 48GB a memória de GPU necessária para fazer o *fine-tuning* de uma LLM de 65 bilhões de parâmetros. Neste caso, sem degradar o tempo de execução ou o desempenho preditivo em comparação com um *fine-tuning* convencional de 16 bits. A Figura 2.18 apresenta um diagrama no qual se compara diferentes métodos de *fine-tuning*.

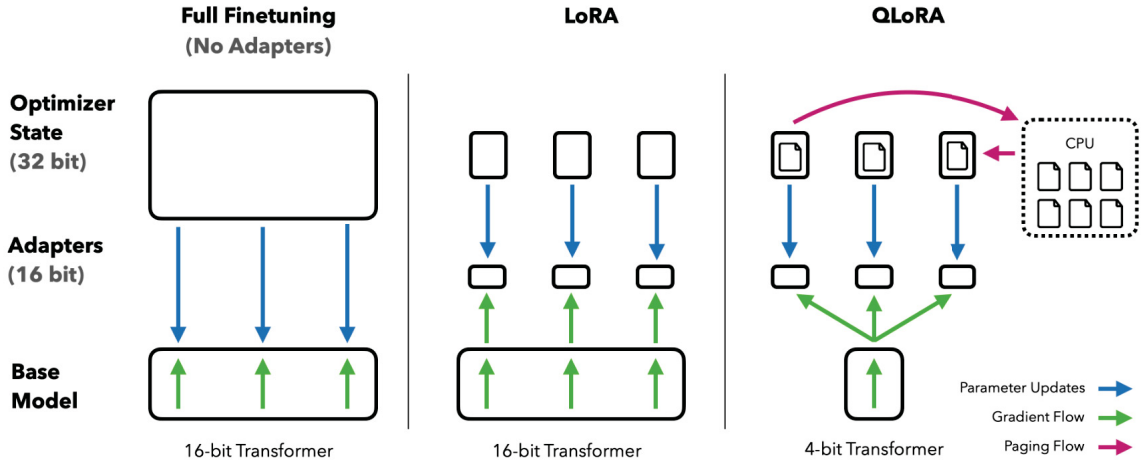


Figura 2.18: Diferentes métodos de *fine-tuning* e seus requisitos de memória. QLORA melhora em relação ao LoRA quantizando o modelo *transformers* com precisão de 4 bits e usando otimizadores paginados para lidar com picos de memória. Fonte (Dettmers et al., 2023).

2.8 MODELOS PRÉ-TREINADOS UTILIZADOS

Considerando que os dados deste pesquisa estão originalmente em português, foram utilizados os seguintes modelos pré-treinados:

- BERTimbau (Souza et al., 2020): É um modelo BERT em português treinado a partir do BrWaC (*Brazilian Web as Corpus*) (Wagner Filho et al., 2018) com 2,68 bilhões de tokens em 3,53 milhões de documentos.

- Open-Cabrita3b (Larcher et al., 2023): É um modelo baseado no OpenLLaMA que foi pré-treinando utilizando o subconjunto em português do conjunto de dados mC4 (Xue et al., 2020), denominado mC4-pt.

2.9 SENTENCE TRANSFORMERS

Sentence Transformers (SBERT) é uma *framework* Python para acessar, usar e treinar *embeddings* de texto e imagem em modelos do estado da arte. Foi proposto por Nils Reimers and Iryna Gurevych (Reimers e Gurevych, 2019) e foi utilizado nesta tese para extração dos *embeddings* das sentenças históricas dos pacientes de planos de saúde e posterior utilização para treinamento usando o algoritmo de classificação *Random Forest*. Para essa tarefa, ele utiliza uma LLM pré-treinada como entrada além das sentenças para retornar os *embeddings*. A Figura 2.19 apresenta um diagrama resumido do funcionamento do *framework*.

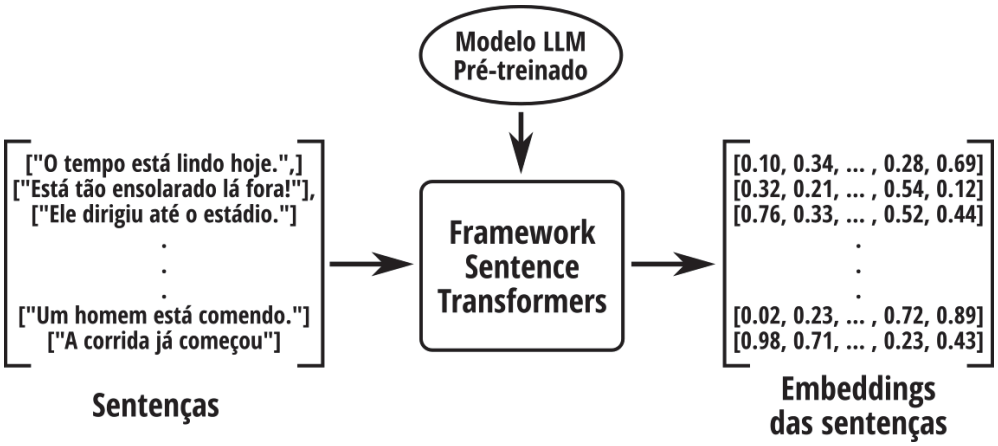


Figura 2.19: Diagrama de funcionamento do *Sentence Transformers* para extração de *embeddings* de sentenças.

Além da forma como foi utilizado nesta tese o *Sentence Transformers* pode ser usado para:

- Calcular uma representação vetorial (*embeddings*) de tamanho fixo de textos ou imagens;
- Fazer o cálculo de similaridade de *embeddings* muito rapidamente;
- Utilização em um ampla gama de tarefas, como similaridade textual semântica, pesquisa semântica, agrupamento, classificação e mineração de paráfrase.

2.10 *RANDOM FOREST*

Random Forest (RF) é uma abordagem *ensemble* (Sagi e Rokach, 2018) que combina Árvores de Decisão (Quinlan, 1986). Foi introduzido por Leo Breiman (Breiman, 2001) inspirado no trabalho de Amit e Geman (Amit e Geman, 1997). É um híbrido de *bagging* (Bühlmann e Yu, 2002) e *Random Subspace Method (RMS)* (Ho, 1998) e usa árvores de decisão como classificador base. *Random Forest* pode ser usado tanto para classificação como para regressão, portanto as variáveis preditoras podem ser ou categóricas ou contínuas.

Segundo Cutler et al. (Cutler et al., 2012), do ponto de vista computacional, *Random Forests* são interessantes porque:

- Funcionam tanto com regressão quanto com classificação.

- São relativamente rápidos para treinar e prever.
- Dependem apenas de um ou dois parâmetros de ajuste.
- Têm uma estimativa embutida do erro de generalização.
- Podem ser usados diretamente para problemas de alta dimensionalidade.
- Podem facilmente ser implementados em paralelo.

Além desses pontos interessantes o *Random Forest* é capaz de lidar com dados desbalanceados. Ainda segundo Cutler et al. (Cutler et al., 2012), estatisticamente as *Random Forests* são atraentes por causa dos recursos adicionais que fornecem, tais como:

- avaliação de importância de variáveis;
- imputação de valor ausente;
- visualização;
- detecção de *outliers*.

Seu funcionamento é baseado em um *ensemble* de árvores de decisão em que cada árvore depende de uma coleção de variáveis aleatórias. Formalmente, para um vetor aleatório p-dimensional $X = (X_1, ..., X_p)^T$ representando as variáveis de entrada ou preditoras e uma variável aleatória Y representando a variável de saída, assume-se uma distribuição conjunta desconhecida $P_{XY}(X, Y)$. O objetivo é então encontrar uma função de predição $f(X)$ para prever Y. A função de predição é determinada por uma função de perda $L(Y, f(X))$ e definida para minimizar o valor esperado da perda

$$E_{xy}(L(Y, f(X))) \tag{2.5}$$

em que os subscritos denotam a expectativa em relação à distribuição conjunta de X e Y.

Intuitivamente, $L(Y, f(X))$ é uma medida de quão perto $f(X)$ está de Y; ele penaliza valores de $f(X)$ que estão distantes de Y. Típicas escolhas de L são erro quadrático $L(Y, f(X)) = (Y - f(X))^2$ para regressão e perda 0-1 para classificação:

$$L(Y, f(X)) = I(Y \neq f(X)) = \begin{cases} 0, & if\ Y = f(X) \\ 1, & caso\ contrario \end{cases} \tag{2.6}$$

Minimizar $E_{xy}(L(Y, f(X)))$ para erro quadrático dá a esperança condicional

$$f(x) = E(Y|X = x) \tag{2.7}$$

também conhecida como função de regressão. Na classificação, se o conjunto de valores de Y é denotado por Γ, minimizar $E_{xy}(L(Y, f(X)))$ com perda 0-1 dá

$$f(x) = \arg\max_{y \in \Gamma} P(Y = y|X = x), \tag{2.8}$$

também conhecido com regra de Bayes.

Ensembles constroem f em termos de uma coleção chamados "aprendizes base" $h_1(x), ..., h_J$ que são combinados para chegar em um "preditor *ensemble*" $f(X)$. Na regressão, o preditor *ensemble* é a média dos aprendizes base

$$f(X) = \frac{1}{J}\sum_{j=1}^{J} h_j(x), \tag{2.9}$$

enquanto que na classificação, $f(X)$ é a classe predita mais frequente (votação)

$$f(x) = \arg\max_{y\in\Gamma} \sum_{j=1}^{J} I(y = h_j(x)), \tag{2.10}$$

No *Random Forest* o j-ésimo aprendiz base é uma árvore denotada por $h_j(X, \theta_j)$, em que θ_j é uma coleção de variáveis aleatórias e os θ_j's são independentes para $j = 1, ..., J$.

2.11 XGBOOST

O XGBoost (*Extreme Gradient Boosting*) é também uma abordagem *ensemble* (Sagi e Rokach, 2018) e utiliza a técnica *boosting* (Schapire, 2003) e assim como no RF também tem sua construção baseada em árvores de decisão (Quinlan, 1986), com o número de estimadores ajustável. Ele realiza o treinamento sequencial de vários previsores fracos para gerar um previsor forte, em que cada previsor busca corrigir o antecessor associando pesos para cada instância de treinamento, seu foco é reforçar as instâncias incorretamente previstas de modo a aumentar o peso das mesmas (Bartlett et al., 1998; Chen e Guestrin, 2016). Diferentemente do RF, no XGBoost as árvores são aprendidas sequencialmente e com base no desempenho de todas as árvores anteriores. No aumento do gradiente, em cada estágio, uma nova árvore de decisão é aprendida com o objetivo de corrigir os erros obtidos pelas árvores existentes (Chen e Guestrin, 2016).

O foco do método *boosting* é reduzir o viés e a variância a cada novo modelo criado, com base nas dificuldades enfrentadas pelo modelo anterior (Suen et al., 2005). O viés poderia ser definido como o quanto as predições são diferentes dos valores reais enquanto que a variância define a sensibilidade do modelo para realizar predições em outros conjuntos de dados. O XGBoost utiliza *gradient boosting*, uma extensão do método anterior, no qual um gradiente descendente é aplicado para melhorar as árvores, de acordo com o erro dos modelos anteriores.

O XGBoost pode ser descrito brevemente da seguinte forma: para um determinado conjunto de dados $D = (x_i, y_i)(|D| = n, x_i \in \mathbb{R}^m, y_i \in \mathbb{R})$, com x_i, y_i (entradas e saídas, respectivamente), m características e n observações, o modelo usa K funções aditivas para prever saídas (Chen e Guestrin, 2016):

$$\hat{y_i} = \emptyset(X_i) = \sum_{k=1}^{k} f_k(X_i), f_k \in F. \tag{2.11}$$

com $\hat{y_i}$ sendo a saída do modelo e F o espaço da árvore de regressão, definida como:

$$F = f(x) = \omega_{q(x)}(q : \mathbb{R}^m \rightarrow T, \omega \in \mathbb{R}^T) \tag{2.12}$$

A estrutura de cada árvore é representada por q, enquanto o número de folhas e seus pesos são representados por T e ω, respectivamente. Além disso, o termo f_k representa uma estrutura de árvore independente com pesos de folha ω. No processo de otimização da árvore de regressão, a seguinte função objetivo deve ser minimizada (Chen e Guestrin, 2016):

$$L(\emptyset) = \sum_{i} l(\hat{y_i}, y_i) + \sum_{k} \Omega(f_k) \tag{2.13}$$

Existe também uma função de perda convexa l que mede a diferença entre $\hat{y}_i$ e y_i que são, respectivamente, a previsão dada pelo modelo e o valor real. O termo Ω penaliza a complexidade das árvores de regressão e é dado por (Chen e Guestrin, 2016):

$$\Omega(f_k) = \gamma T + \frac{1}{2}\lambda\|\omega\|^2 \tag{2.14}$$

em que γ e λ são parâmetros de regularização para ajudar a reduzir a complexidade do modelo e proteger contra *overfitting*.

No entanto, os modelos que usam aumento de gradiente são treinados de forma aditiva. Nesses casos, a seguinte função objetivo é minimizada:

$$L(\emptyset) = \sum_i l(y_i, \hat{y}_i{}^{(t-1)} + f_t(x_i)) + \Omega(f_t) \tag{2.15}$$

o número de iterações f_t é adicionado na função objetivo (Chen e Guestrin, 2016).

2.12 SHAP

SHAP (*SHapley Additive exPlanations*) proposto por (Lundberg e Lee, 2017) é um método de explicação de modelo baseado no Teoria do Valor de Shapley (Shapley, 1953), proposto em 1952 para distribuir o valor de um jogo entre os jogados. Ele busca explicar a importância de cada atributo na inferência do modelo, para isso mede a alteração média da previsão do modelo quando o valor do atributo varia nas previsões. Além disso, fornece tanto explicações globais quanto locais.

Para interpretação de uma previsão individual, os valores *Shapley* são calculados considerando que cada atributo é um jogador que coopera com os demais para receber a recompensa. Assim, os valores *Shapley* correspondem à contribuição de cada atributo para a previsão do modelo. A recompensa é a previsão real feita pelo modelo menos o valor médio de todas as previsões. No final, os jogadores dividem essa recompensa de acordo com sua contribuição, essa divisão é calculada pelos valores de *Shapley* e reflete a importância de cada variável.

Para interpretação global do modelo, calcula-se a média das contribuições de cada atributo considerando todas as previsões realizadas pelo modelo.

Formalmente, o valor SHAP para um recurso i é calculado como a média ponderada das contribuições marginais de i em todos os subconjuntos possíveis S de recursos que não incluem i:

$$\phi_i = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!}[f(S \cup \{i\}) - f(S)] \tag{2.16}$$

em que:

- N é o conjunto de todas as características;
- S é um subconjunto de recursos que não contém i;
- $f(S)$ é a previsão do modelo quando apenas as características do subconjunto S estão presentes.

Esta equação representa o valor SHAP ϕ como a contribuição do recurso para a diferença na predição, calculada em média sobre todas as combinações possíveis de recursos. Embora esta fórmula forneça uma atribuição rigorosa e justa da saída do modelo para suas entradas, os

valores SHAP resultantes podem ser difíceis de interpretar para não especialistas, especialmente ao lidar com modelos grandes.

Por fim, é importante acrescentar que o SHAP é agnóstico para o cálculo dos valores de Shapley, podendo ser aplicado a qualquer modelo de Aprendizado de Máquina. A Figura 2.20 ilustra um dos gráficos chamado *waterfall*, no qual os atributos idade, gênero, pressão arterial e índice de massa corporal (IMC) contribuem para a saída apresentada. Nele, a *taxa base = 0.1* é o valor médio de saída do modelo para todos os exemplos usados no teste. A *saída = 0.4* representa o valor da saída para o exemplo cujas entradas são *idade=65*, *sexo=F*, *Pressão Arterial=180* e *IMC=40*. A diferença entre *taxa base* e a *saída* que é 0.3 é apresentado pelas contribuições positivas ou negativas para a inferência considerando cada atributo de entrada, representado pelo gráfico do lado direito da Figura 2.20.

Figura 2.20: Exemplo de explicação para uma instância com SHAP. Tradução de (Lundberg, 2018)

2.13 CONSIDERAÇÕES FINAIS

Este capítulo apresentou definições sobre internações e internações evitáveis. Também foi apresentado conceitos sobre classificação de texto e alguns métodos relacionados a essa temática como pré-processamento de texto e representação de documentos. Além disso, foram apresentados conceitos sobre Transfer Learning, LLMs e a aplicação delas em PLN, os algoritmos de classificação *Random Forest* e XGBoost e também o método de interpretabilidade SHAP.

No próximo capítulo a análise do estado da arte de trabalhos da literatura relacionados a previsão de internações são apresentados e discutidos.

3 ESTADO DA ARTE

A abordagem dos assuntos anteriores é pertinente para compor o embasamento teórico para o entendimento do escopo do problema tratado neste trabalho: previsão de internações. Diversos métodos são encontrados na literatura referentes a previsão de internações, sendo assim, neste capítulo são apresentados trabalhos encontrados na literatura sobre abordagens para a previsão de internações que utilizam métodos de aprendizagem de máquina ou estatísticos.

3.1 TRABALHOS RELACIONADOS

Nesta seção são apresentados os trabalhos encontrados na literatura cujo objetivo é prever algum tipo de internação com perspectivas de dirigir ações preventivas, para reduzir custos ou melhorar o atendimento. Eles apresentam abordagens que vão desde métodos estatísticos tradicionais a diferentes métodos de aprendizado de máquina.

Dentre essas pesquisas, vale destacar as de Smith *et al.* (Smith et al., 2011), Monsen *et al.* (Monsen et al., 2012), Wang *et al.* (Wang et al., 2012), Baillie *et al.* (Baillie et al., 2013), Billings *et al.* (Billings et al., 2013), Choudhry *et al.* (Choudhry et al., 2013), Hippisley-Cox *et al.* (Hippisley-Cox e Coupland, 2013), Singal *et al.* (Singal et al., 2013), Hao *et al.* (Hao et al., 2014), Hebert *et al.* (Hebert et al., 2014b), Rana *et al.* (Rana et al., 2014), Dai *et al.* (Dai et al., 2015), Barak-Corren *et al.* (Barak-Corren et al., 2017), Patel *et al.* (Patel et al., 2018), Lorenzoni *et al.* (Lorenzoni et al., 2019), Angraal *et al.* (Angraal et al., 2020) e Baig *et al.* (Baig et al., 2020).

Smith *et al.* (Smith et al., 2011) desenvolveram um modelo para prever o risco dentro do período de 5 anos para mortalidade ou hospitalização por Insuficiência Cardíaca (IC). Para construção do modelo os autores utilizaram análise de coorte e regressão de Cox (Lunn e McNeil, 1995). Em relação aos dados utilizaram 4.969 pacientes que foram diagnosticados com IC, e a seleção das características foi realizada sem nenhum método automatizado e baseado em fatores de riscos associados a IC. Como resultado conseguiram gerar um modelo com AUC (*Area Under the ROC Curve*) de 0,71.

Monsen *et al.* (Monsen et al., 2012) propuseram uma medida para predizer o risco de internações entre pacientes em cuidados domiciliares, o HRS (*Hospitalization Risk Score*). Os dados utilizados foram de registros eletrônicos representando 1643 atendimentos domiciliares. O desenvolvimento do HRS foi baseado em estatística descritiva e modelos lineares generalizados. A medida proposta alcançou AUC de 0,63.

O trabalho de Wang *et al.* (Wang et al., 2012) utilizou regressão multinomial para geração de modelos de predição que identificassem pacientes com IC com alto risco de internação ou morte. Os pesquisadores utilizaram dados clínicos e administrativos de mais de 198 mil pacientes que receberam atendimento da *Veterans Health Administration* e que tiveram pelo menos um diagnóstico primário ou secundário de IC no ano anterior. Os dados foram selecionados a partir de 6 categorias clinicamente relevantes de dados sociodemográficos, condições médicas, sinais vitais, uso de serviços de saúde, testes laboratoriais e medicamentos. Como resultado, Wang mostrou que para os pacientes pertencentes aos 5% com maior risco de morte ou internação, 27% deles ocorria no decorrer dos 30 dias subsequentes e 80% no decorrer do ano seguinte.

Baillie *et al.* (Baillie et al., 2013) utilizam análise de coorte para desenvolver um modelo de previsão de reinternação dentro de 30 dias após a alta do paciente. Utilizando dados históricos dos pacientes Baillie *et al.* conseguiram alcaçar uma AUC de 0,62.

Billings *et al.* (Billings et al., 2013) propôs um modelo de previsão de internações utilizando regressão logística com base em dados históricos ambulatoriais, de atendimentos em emergências, internações, comorbidades e demográficos de 1.836.099 pessoas. Como melhor resultado, conseguiu-se uma AUC de 0,78 para um horizonte temporal de 12 meses.

Choudhry *et al.* (Choudhry et al., 2013) apresentou uma proposta para geração de modelos de previsão de reinternações por todas as causas considerando os próximos 30 dias. Foram utilizados dados demográficos, sociais, contatos com serviços de saúde, histórico de sinais vitais, exames físicos, medicamentos, procedimentos, testes laboratoriais e interações ambientais. Para geração dos modelos Choudhry *et al.* utilizaram análise de coorte e regressão logística e alcançaram como melhor resultado a AUC de 0,76.

Já Hippisley-Cox *et al.* (Hippisley-Cox e Coupland, 2013) propuseram um algoritmo para estimar o risco de internação hospitalar de emergência para pacientes entre 18 e 100 anos, considerando como horizonte temporal os dois anos seguintes. Para construção do algoritmo utilizaram análise de coorte (Glenn, 2005) e dados da *QResearch database* (`http://www.qresearch.org`). O algoritmo proposto conseguiu atingir a AUC de 0,78 para homens e 0,77 para mulheres.

O trabalho de Singal *et al.* (Singal et al., 2013) propõe um modelo automatizado para prever o risco de reinternação para pacientes com cirrose em um horizonte temporal de 30 dias. O trabalho utiliza dados de prontuários eletrônicos disponíveis desde o início da primeira hospitalização, além de dados socieconômicos. Para geração de modelo Singal *et al.* utilizam análise de coorte e regressão logística. Como resultado o modelo alcançou uma AUC de 0,68.

Hao *et al.* (Hao et al., 2014) desenvolveram um modelo baseado em árvore de decisão com características discriminantes de registros médicos para estimar o risco de reinternação com horizonte temporal de 30 dias após a alta do paciente. Hao analisou 14.680 características dos dados e utilizou análise de variância e o algoritmo RF para selecionar as características relevantes dos dados, resultando em 127 características selecionadas. O método proposto conseguiu atingir uma AUC de 0,7.

Hebert *et al.* (Hebert et al., 2014b) utilizaram registros eletrônicos abrangentes para criar modelos de risco de reinternação, considerando um horizonte temporal de 30 dias, para problemas de insuficiência cardíaca congestiva, infarto agudo do miocárdio ou pneumonia. Foram 3.968 exemplos utilizados que abrangeram dados demográficos, de comorbidades, laboratoriais e de medicação, em um período de dois anos. A seleção das características levou em consideração aspectos relacionados a estes problemas de saúde. Para construção dos modelos, utilizaram análise de coorte com regressão logística e alcançaram o melhor desempenho para modelos de reinternação por infarto agudo do miocárdio e pneumonia em uma coorte de validação de amostra aleatória (intervalo de AUC 0,73 a 0,76), mas chegaram a um desempenho ruim em uma coorte de validação histórica (AUC de 0,66 para ambos). O modelo de insuficiência cardíaca congestiva teve baixo desempenho em ambas as coortes de validação (AUC 0,63 e 0,64).

Rana *et al.* (Rana et al., 2014) aplicam dados hospitalares para criar um modelo para identificar o risco de reinternação após casos de internação por infarto agudo do miocárdio (IAM). Para construir o modelo, Rana *et al.* utilizaram regressão logística e análise de coorte com dados de 1.660 internações, contendo dados demográficos, comorbidades e diagnósticos anteriores. Os modelos gerados foram avaliados quanto à capacidade de identificar pacientes com alto risco de reinternação por doença isquêmica do coração para os próximos 30 dias e aqueles com risco de reinternação por todas as causas dentro de 12 meses após a hospitalização inicial por IAM. Como resultado o melhor modelo alcançou AUC de 0,78 para reinternação em 30 dias e 0,72 para reinternação em 12 meses.

O trabalho de Dai *et al.* (Dai et al., 2015) utiliza métodos de aprendizado supervisionado para realizar a predição de internações relacionadas a doenças do coração. No trabalho, Dai utilizou dados do maior hospital com rede de segurança da Nova Inglaterra, a *Boston Medical Center*, e focou em pacientes com pelo menos um diagnóstico ou procedimento relacionado a problemas do coração no período de 2005 a 2010. Os registros utilizados continham dados demográficos, histórico de visitas, problemas dos pacientes, medicamentos, resultados de exames laboratoriais, procedimentos e observações clínicas. Com o objetivo de prever se pacientes sofreriam internação no ano seguinte, Dai *et al.* conseguiram mostrar que com menos de 30% de falsos positivos é possível alcançar uma taxa de verdadeiros positivos em torno de 82%.

Já Barak-Corren *et al.* (Barak-Corren et al., 2017) utilizaram dados que normalmente são coletados rotineiramente na chegada de um paciente ao pronto-socorro para desenvolver um modelo de previsão de internação com o objetivo de reduzir o tempo de espera do paciente até o momento da internação, caso fosse necessário. Barak-Corren *et al.* construíram o modelo preditivo utilizando uma abordagem hibrida, executando o treinamento com um algoritmo de regressão logística sobre os resultados gerados por um classificador Naive Bayes. Com os dados coletados dos pacientes nos primeiros 30 minutos após a sua chegada ao pronto-socorro, o modelo gerado conseguiu identificar 73,4% das internações com especificidade de 90% e 35,4% das internações com especificidade de 99,5%. Uma estimativa feita no trabalho calculou que o modelo poderia economizar 5.917 horas por ano ou 30 minutos por internação.

O trabalho de Patel *et al.* (Patel et al., 2018) testa quatro algoritmos de aprendizado de máquina, *Decision tree*, *Random Forest*, *Logistic LASSO Regression* e *Gradient Boosting* (GB) para realizar a previsão da necessidade ou não de internações pediátricas relacionadas a asma. Utilizaram como insumo dados clínicos disponíveis no momento da triagem dos pacientes, integrados com dados não clínicos contendo informações sobre clima, características do bairro e carga viral da comunidade. A extração e seleção de características dos dados foram realizados por meios tradicionais e considerando características que os autores jugaram importantes para predição de internação por asma. Alcançaram o melhor resultado com o GB como uma AUC de 0,84.

Lorenzoni *et al.* (Lorenzoni et al., 2019) também apresentaram um estudo comparativo entre a performance de modelos gerados por oito técnicas de Aprendizado de Máquina diferentes, cujo objetivo foi predição de internação de pacientes com IC. Os pesquisadores utilizaram dados do *Gestione Integrata dello Scompenso Cardiaco* (GISC), um projeto contínuo aplicado na região de Puglia, no sul da Itália. Os dados continham registros relacionados a características demográficas, clínicas e histórico médico de 380 pacientes com IC avaliados no período entre 2001 e 2015. O trabalho conseguiu identificar 77,8% das internações com especificidade de 85,7%.

O trabalho de Angraal *et al.* (Angraal et al., 2020) buscou desenvolver modelos para predizer mortalidade e internação de pacientes com um tipo específico de IC a insuficiência cardíaca com fração de ejeção preservada (ICFEP). Foram utilizados dados demográficos, clínicos, laboratoriais, eletrocardiográficos e de questionários com informações tais como: frequência de sintomas, limitações físicas, sociais, etc. O trabalho conseguiu alcançar uma média de AUC de 0,72 para mortalidade e 0,76 para internação para o melhor modelo gerado.

Baig *et al.* (Baig et al., 2020) utilizaram aprendizado de máquina para desenvolver um modelo para predizer casos de reinternação em até 30 dias após a alta do paciente. Baig utilizou registros eletrônicos referentes a 213.440 internações de hospitais públicos da região de Auckland, Nova Zelândia. Os dados cobrem 2 anos de internações entre 2015 e 2016 e contêm informações tais como: gênero, idade, etnia, tipo, tempo e custo de internação e comorbidades.

Como resultado conseguiu-se gerar um modelo para todas as causas de reinternação em 30 dias com AUC de 0,75 sobre todo o conjunto de dados.

Como se constata nos trabalhos elencados, observa-se que em parte significativa deles estão relacionados a previsão de internações para problemas cardíacos. Este interesse se justifica pelo fato de que problemas cardíacos são responsáveis por cerca de 30% dos custos totais com internações nos Estados Unidos (Jiang et al., 2011), além de tais predições terem potencial de prevenção. No Brasil, em um estudo realizado na região de São José do Rio Preto, no interior paulista (Ferreira et al., 2014), esta mesma proporção se mantém. Além disso, os resultados alcançados pelos trabalhos apresentados anteriormente mostram o potencial da utilização de registros eletrônicos em saúde para previsão de internações.

Na literatura, é comum encontrar em trabalhos relacionados à aprendizagem de máquina e saúde, a expressão Registro Eletrônico em Saúde (RES) (Shickel et al., 2018), que define-se como um conjunto de dados de saúde e assistência de um paciente durante o decorrer de sua vida. Os dados abrangem todo tipo de informação referente ao paciente, tais como: procedimentos, consultas, administração de medicamentos, resultados de exames e informações demográficas (Araujo et al., 2014). No Brasil, além do SUS, existem vários sistemas de gestão de operadoras de planos de saúde suplementar que mantêm RES. Nos planos de saúde, esses registros contêm dados demográficos e relativos a realização de exames, consultas, cirurgias, insumos utilizados, diagnóstico por meio da CID (Classificação Estatística Internacional de Doenças e Problemas Relacionados com a Saúde) e demais procedimentos que os planos de saúde devem oferecer aos seus beneficiários. Sendo assim, explorar o potencial desses dados em abordagens de aprendizagem de máquina parece promissor, uma vez que os registros desses dados estão relacionados efetivamente às verdadeiras condições de saúde dos pacientes.

Para estabelecer algum nível de comparação entre os trabalhos elencados neste capítulo e os resultados deste trabalho, apresenta-se por meio das Figuras 3.1, 3.2 e 3.3 os gráficos comparativos das AUCs alcançadas pelos trabalhos para os horizontes temporais de um dia ou menos, trinta dias e um ano ou mais. Cada barra do gráfico está separada por cores que indicam se são de internações ou reinternações para todos os tipos de problemas de saúde possíveis, ou internação e reinternação para problemas específicos de saúde.

As Tabelas 3.1 e 3.2 apresentam as informações sumarizadas para os trabalhos elencados neste capítulo. A coluna “Dados sequenciais?” indica se os dados utilizados na geração do modelo mantêm a sequência de ocorrência durante o processo de aprendizagem. A coluna “Inter. p/ problema específico” indica se o modelo foi gerado para previsão de internação para um problema de saúde específico.

3.2 CONSIDERAÇÕES FINAIS

Apesar da literatura especializada abranger uma variedade de métodos para previsões de internações, nenhum dos trabalhos abrangem as características dos dados utilizados neste trabalho. A natureza sequencial dos dados e o nível de detalhes em relação aos contatos dos pacientes com os serviços de saúde são diferentes dos encontrados nas investigações acessadas. Essa característica nos dados permite dependendo da técnica utilizada, a extração de contexto das ocorrências. Técnicas que utilizam a arquitetura *transformers*, mesmo em implementações como do BERT, que utiliza representações bidirecionais, gera *embeddings* diferentes simplesmente pela ordem em que os termos aparecem em uma sentença. Portanto, a sequência de ocorrência dos eventos de saúde torna-se uma condição relevante não considerado nos trabalhos analisados nesta seção. Os resultados dos trabalhos estudados mostram que ainda são necessários esforços para melhora dos modelos de previsões de internações, sendo que o melhor resultado dos trabalhos foi

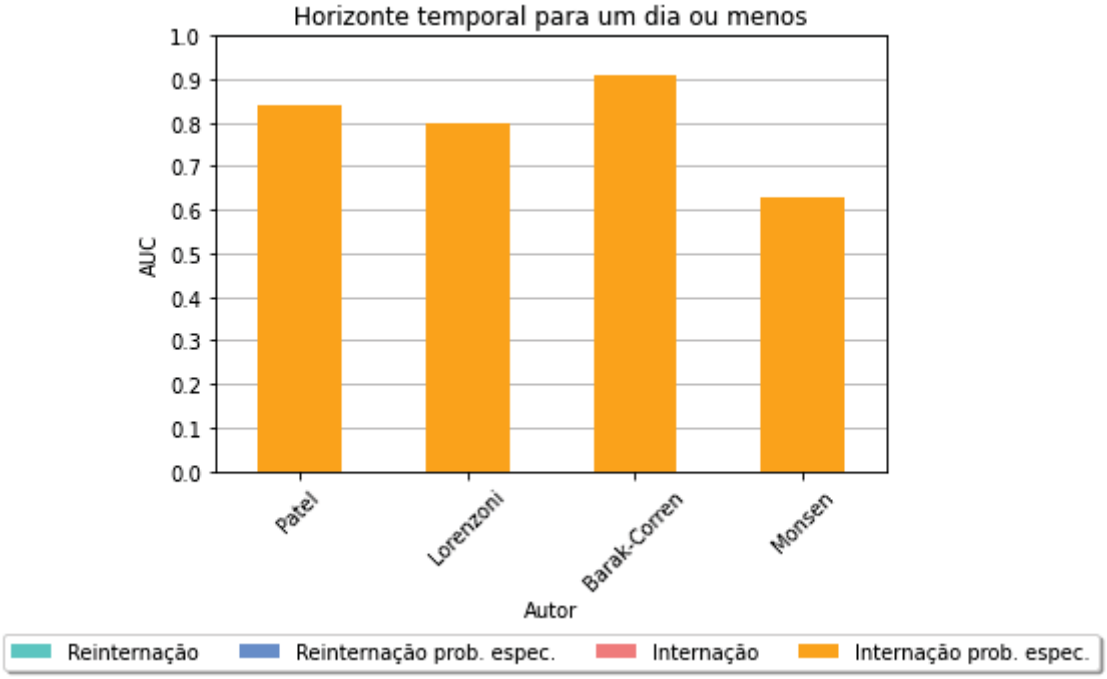


Figura 3.1: AUC para modelos dos trabalhos com horizonte temporal de um dia ou menos.

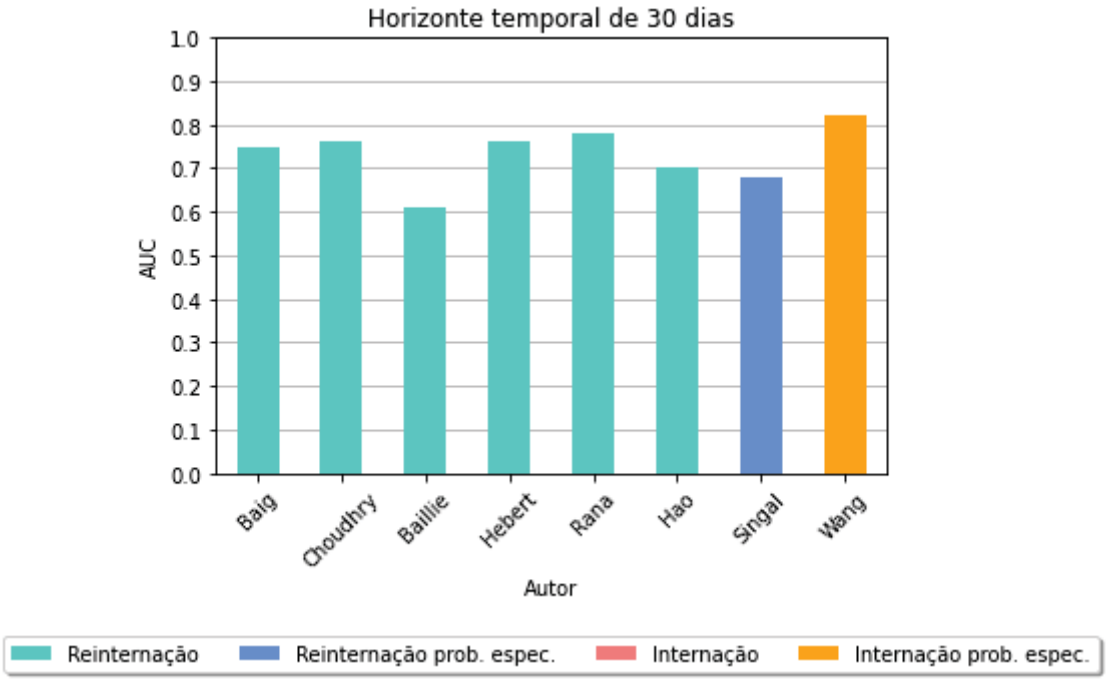


Figura 3.2: AUC para modelos dos trabalhos com horizonte temporal de 30 dias.

Tabela 3.1: Dados sumarizados dos trabalhos selecionados para o estado da arte.

Referência	Seleção de características automatizada?	Dados sequenciais?	Internação ou reinternação?	Inter. p/ problema específico	Método
(Smith et al., 2011)	Não	Não	Internação	Sim	CA, CR
(Monsen et al., 2012)	Não	Não	Internação	Sim	DS, LM
(Wang et al., 2012)	Não	Não	Internação	Sim	LR
(Baillie et al., 2013)	Não	Não	Reinternação	Não	CA
(Billings et al., 2013)	Não	Não	Internação	Não	LR
(Choudhry et al., 2013)	Não	Não	Reinternação	Não	CA, LR
(Hippisley-Cox e Coupland, 2013)	Não	Não	Internação	Não	CA
(Singal et al., 2013)	Não	Não	Reinternação	Sim	LR, CA
(Hao et al., 2014)	Sim	Não	Reinternação	Não	DT, CA
(Hebert et al., 2014b)	Não	Não	Reinternação	Não	CA, LR
(Rana et al., 2014)	Não	Não	Reinternação	Não	CA, LR
(Dai et al., 2015)	Não	Não	Internação	Sim	SVM, AB, LR, NB
(Barak-Corren et al., 2017)	Não	Não	Internação	Sim	LR, NB
(Patel et al., 2018)	Não	Não	Internação	Sim	DT, RF, LR, GB
(Lorenzoni et al., 2019)	Não	Não	Internação	Sim	GLMN,LR, CART, LB, AB, SVM, NN
(Angraal et al., 2020)	Não	Não	Internação	Sim	LR, RF, GB, SVM
(Baig et al., 2020)	Não	Não	Reinternação	Não	GB, RF, AB

CA (Cohort Analysis), LR (Logistic Regression), GB (Gradient boosting), DT (Decision Tree), RF (Random Forest), SVM (Support Vector Machine), CART (Classification and Regression Trees), NB (Naive Bayes), CR (Cox Regression), NN (Neural network), DS (Descriptive statistics), AB (AdaBoost), GLMN (Generalized Linear Model Net), LB (LogitBoost), LM (Linear model)

Tabela 3.2: Continuação dos dados sumarizados dos trabalhos selecionados para o estado da arte.

Referência	Dados abertos p/ comun. científica	Resultados Interpretados	Nº exemplos	Horizonte temporal	AUC
(Smith et al., 2011)	Não	Não	4696	5 Anos	0,71
(Monsen et al., 2012)	Não	Não	1643	?	0,63
(Wang et al., 2012)	Não	Não	198640	30 Dias e 1 Ano	0,82 e 0,81
(Baillie et al., 2013)	Não	Não	120396	30 Dias	0,61
(Billings et al., 2013)	Sim	Não	1836099	1 Ano	0,78
(Choudhry et al., 2013)	Não	Não	126479	30 Dias	0,76
(Hippisley-Cox e Coupland, 2013)	Sim	Não	6673458	2 Anos	0,78
(Singal et al., 2013)	Não	Não	1291	30 Dias	0,68
(Hao et al., 2014)	Não	Não	487572	30 Dias	0,70
(Hebert et al., 2014b)	Não	Não	3968	30 Dias	0,76
(Rana et al., 2014)	Não	Não	1660	30 Dias e 1 Ano	0,78 e 0,72
(Dai et al., 2015)	Não	Sim	45579	1, 3, 6 e 12 meses	?
(Barak-Corren et al., 2017)	Não	Não	59033	Mesmo dia	0,91
(Patel et al., 2018)	Não	Sim	29392	Mesmo dia	0,84
(Lorenzoni et al., 2019)	Não	Sim	380	?	0,80
(Angraal et al., 2020)	Sim	Sim	1767	3 Anos	0,72
(Baig et al., 2020)	Não	Não	180118	30 Dias	0,75

AUC - Area Under the ROC Curve

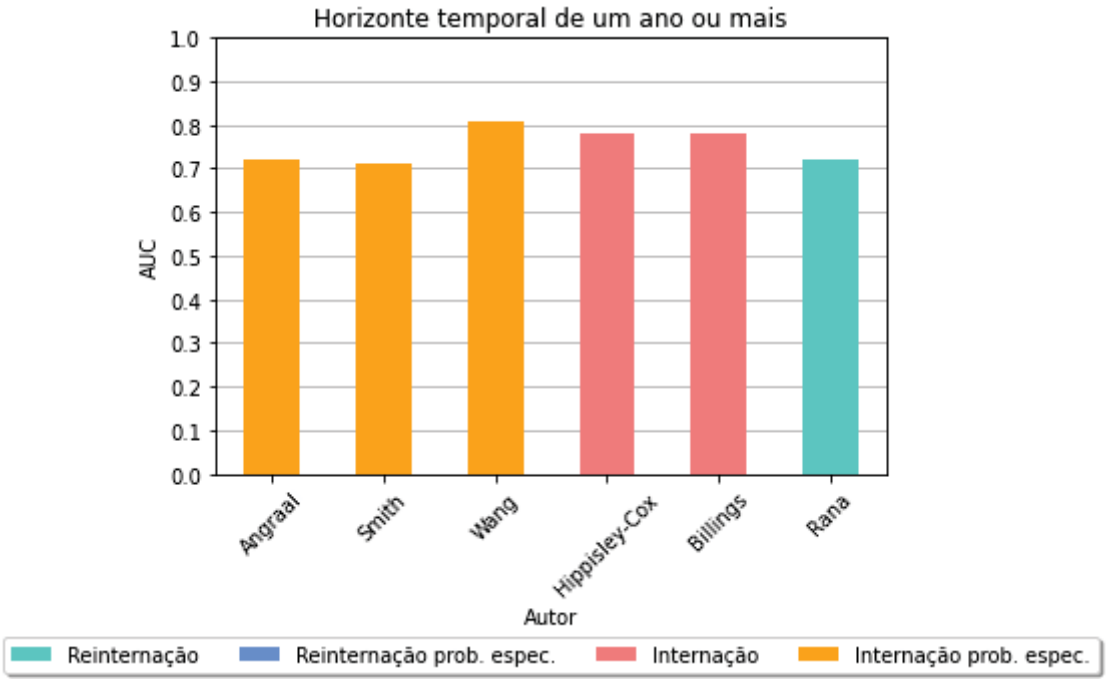


Figura 3.3: AUC para modelos dos trabalhos com horizonte temporal de um ano ou mais.

uma AUC de 0,91, em que o modelo gerado foi para um problema específico de saúde. Para os trabalhos que buscam modelos para qualquer tipo de internação, mesmo caso da proposta deste trabalho, os resultados são inferiores aos modelos para problemas específicos, sendo o melhor entre eles com AUC de 0,78.

A natureza dos dados utilizados e os métodos de extração de suas características podem ter papel importante na melhoria desses resultados, uma vez que a maior parte dos trabalhos utiliza métodos não automáticos para a extração e seleção de características, normalmente relacionados a alguns problemas de saúde específicos.

O método proposto, gerador do modelo de previsão de internações para qualquer tipo de internação, treinado com base em dados de planos de saúde, é apresentado no próximo capítulo.

4 MÉTODO PROPOSTO

Este capítulo apresenta o método proposto para as previsões de internações por meio de dados estruturados de planos de saúde e a abordagem metodológica utilizada para realizar os experimentos desta pesquisa. Para construir e avaliar o modelo proposto foram realizadas as seguintes tarefas:

- Organizar e construir uma base de dados histórica de beneficiários rotulados por ocorrência de internação a partir de uma base de dados de planos de saúde estruturada. Etapa 1 da Figura 4.1.
- Extrair características dos dados históricos dos beneficiários por meio de técnicas de PLN. Etapa 2 da Figura 4.1.
- Treinar modelos de previsão de internação baseados em métodos tradicionais de aprendizado de máquina como *Random Forest*, *Gradient Boosting* e também com LLMs. Etapa 3 e 4 da Figura 4.1.
- Combinar os modelos de previsão a fim de gerar novos modelos buscando maximizar os resultados. Etapa 4 da Figura 4.1.
- Avaliar os modelos de previsão construídos a fim de verificar como se comportam ao testar dados não vistos. Etapa 4 da Figura 4.1.
- Investigar os modelos construídos para previsão de internações para um problema específico. Etapa 4 da Figura 4.1.
- Investigar o desempenho dos modelos de previsão para diferentes períodos de antecedência. Etapa 4 da Figura 4.1.

Assim, neste capítulo, inicialmente propõe-se uma breve discussão da abordagem proposta e uma análise sobre a automatização da extração de características. Em seguida, apresentam-se os dados utilizados e o processo de sua transformação em dados históricos. Depois, o método de construção e preparação dos dados para utilização pelos algoritmos de aprendizagem de máquina. Em seguida, as estratégias metodológicas utilizadas para a realização dos experimentos relacionados a previsões de internações, destacando-se: métodos de treinamentos dos modelos e métricas de avaliação.

4.1 ABORDAGEM PROPOSTA E O PROCESSO PARA AUTOMATIZAR A EXTRAÇÃO DE CARACTERÍSTICAS

O principal objetivo desse trabalho é construir um método baseado em aprendizado de máquina para fazer a previsão de internações independentemente do problema de saúde ao qual a internação está relacionada. No entanto, conforme discutido no Capítulo 3, na maioria dos trabalhos, os modelos criados para previsões de internação são para problemas específicos de saúde, e a seleção e extração de características para treinamento dos modelos é altamente dependente do problema de saúde ao qual a internação se relaciona. Desta forma, a seleção e extração de características depende de conhecimento prévio específico sobre quais fatores

estão relacionados ao problema de saúde, aumentando consideravelmente o tempo gasto na etapa de preparação de dados, além de possível negligenciamento de características relevantes para predição. Além disso, fatores históricos relacionados a outros problemas ocorridos com beneficiários são desconsiderados na maioria do trabalhos analisados. Assim, uma abordagem que permita geração de modelos de previsão que sejam independentes dessa necessidade de conhecimento prévio e permita analisar também fatores históricos do beneficiários seria de grande importância para esta área de pesquisa.

Desta forma, considerando que os planos de saúde armazenam os dados de seus beneficiários para fazer a administração de suas carteiras, é notório que dentre os dados administrativos há dados relacionados a procedimentos realizados pelos beneficiários. Entretanto, estão organizados de maneira estruturada para utilização pelos sistemas dos planos de saúde, tornando difícil sua utilização em métodos de aprendizado de máquina. Ao estudar e analisar tais dados, percebeu-se que se organizados cronologicamente poderiam formar um tipo de narrativa histórica da saúde de cada beneficiário. Assim, alterando-se a estrutura dos dados desta forma, pode-se aplicar métodos de PLN para extrair de maneira automática características que sejam relevantes para previsões de internações. Desta forma, propõe-se uma abordagem para organizar os dados dos planos de saúde para facilitar a extração de características para posterior aplicação em métodos de aprendizado de máquina com o objetivo de prever internações. A Figura 4.1 retrata de maneira resumida o processo de transformação, extração de características e treinamento dos modelos de classificação.

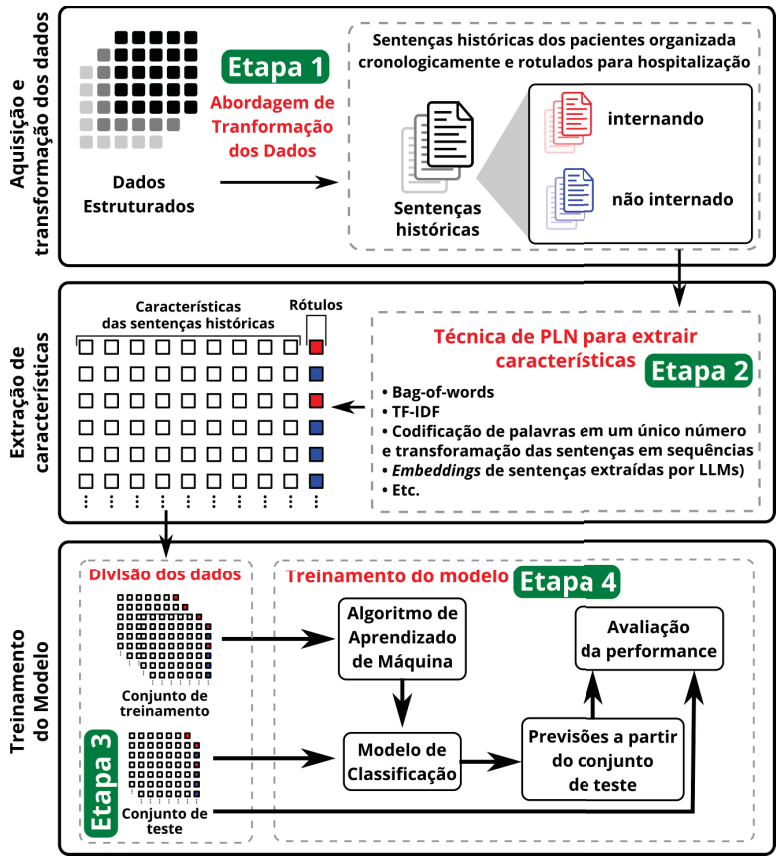


Figura 4.1: Diagrama de blocos simplificado representando a abordagem proposta.

Para alcançar os objetivos do trabalho, em primeiro lugar, criou-se um método de transformação de dados estruturados de sistemas de planos de saúde para dados históricos dos beneficiários (processo correspondente à Etapa 1 da Figura 4.1). Parte desses dados foram

disponibilizado para a comunidade científica (Baro et al., 2022). As bases de dados e o método de transformação são apresentados nas Seções 4.2 e 4.3.

A partir dos dados transformados em sentenças históricas, utilizou-se duas técnicas de extração de características, ambos utilizados em métodos de PLN. O primeiro utiliza a codificação de palavras em um único número e posterior transformação das sentenças em sequências. O segundo por meio de *embeddings* extraídas das sentenças históricas, utilizando LLMs pré-treinadas. Neste último, utilizou-se modelos pré-treinados selecionados da literatura e também modelos pré-treinados exclusivamente com dados de planos de saúde. Portanto, nesta etapa, também há treinamento de modelos dependendo da técnica adotada. Esses processos são apresentados na Seção 4.4 e correspondem à Etapa 2 da Figura 4.1.

A partir das características extraídas, os dados são divididos para treinamento e teste, conforme a Etapa 3 da Figura 4.1. Então, na Etapa 4, são realizados o treinamento e o teste dos modelos. Os treinamentos foram feitos adotando diferentes abordagens e são explicados neste capítulo.

4.2 BASE DE DADOS

Foram utilizadas duas bases de dados diferentes provenientes do mesmo modelo físico. A primeira é proveniente de um *Data warehouse* (DW) de um plano de saúde do estado brasileiro de Santa Catarina. Nele há dados administrativos, de procedimentos realizados em hospitais, laboratórios e consultórios, dados demográficos, insumos utilizados e de diagnósticos referentes aos contatos dos beneficiários[1] com serviços de saúde.

Nesta base, há dados referentes a 34.932 beneficiários com 1.703.960 registros que abrangem o período de 02/02/2000 até 10/04/2013. Os dados estão distribuídos em várias tabelas do DW, sendo a de maior relevância para este trabalho a tabela denominada *fato_evento*. Ela contém registros de todo e qualquer procedimento e contato com serviços de saúde realizados por qualquer beneficiário associado ao plano de saúde e está ligada às tabelas de dimensões que permitem realizar junções para alcançar maior detalhamento dos dados, tais como: data de realização de um procedimento ou contato com serviço de saúde, dados demográficos, especialidade, regime de atendimento, tipo de tratamento, descrição do procedimento e diagnóstico quando aplicável. Os diagnósticos da base seguem a codificação da CID-10 (Organization et al., 2004). A Figura 4.2 apresenta o diagrama simplificado do DW dessa base de dados.

A segunda, disponibilizada na segunda metade do período dessa pesquisa, possui 89.339.526 registros relacionados a 445.199 beneficiários situados no período de janeiro de 1990 até dezembro de 2021. Essa base é proveniente da mesma estrutura da primeira, entretanto, foi fornecida já em uma única tabela, conforma Tabela 4.1. Para simplificar a referência sobre essas duas bases, se utilizará o nome DB1 (*Data Base 1*), para a base menor, e DB2 (*Data Base 2*), para a base maior. É importante destacar que todos os dados destas bases foram anonimizados e só foram disponibilizados para pesquisa após análise e aprovação do departamento jurídico da empresa fornecedora dos dados, que considerou os procedimentos adequados, dispensando, portanto, a exigência de aprovação ética adicional. Na seção 4.2.1, apresenta-se uma análise dos dados de cada uma das bases, buscando demonstrar o perfil delas.

A descrição do evento presente nas bases de dados fornecidas e apresentada na Tabela 4.1 é de grande importância para a gestão dos planos de saúde e para essa pesquisa. Portanto, para

[1]Neste trabalho, usa-se o termo beneficiário e paciente como sinônimos. Por definição, beneficiário é um consumidor de plano de saúde e paciente é uma pessoa que está sendo cuidada por algum profissional da área da saúde. Como utiliza-se dados de planos de saúde relacionados a pessoas que recebem cuidados médicos, utiliza-se então esses dois termos como sinônimos.

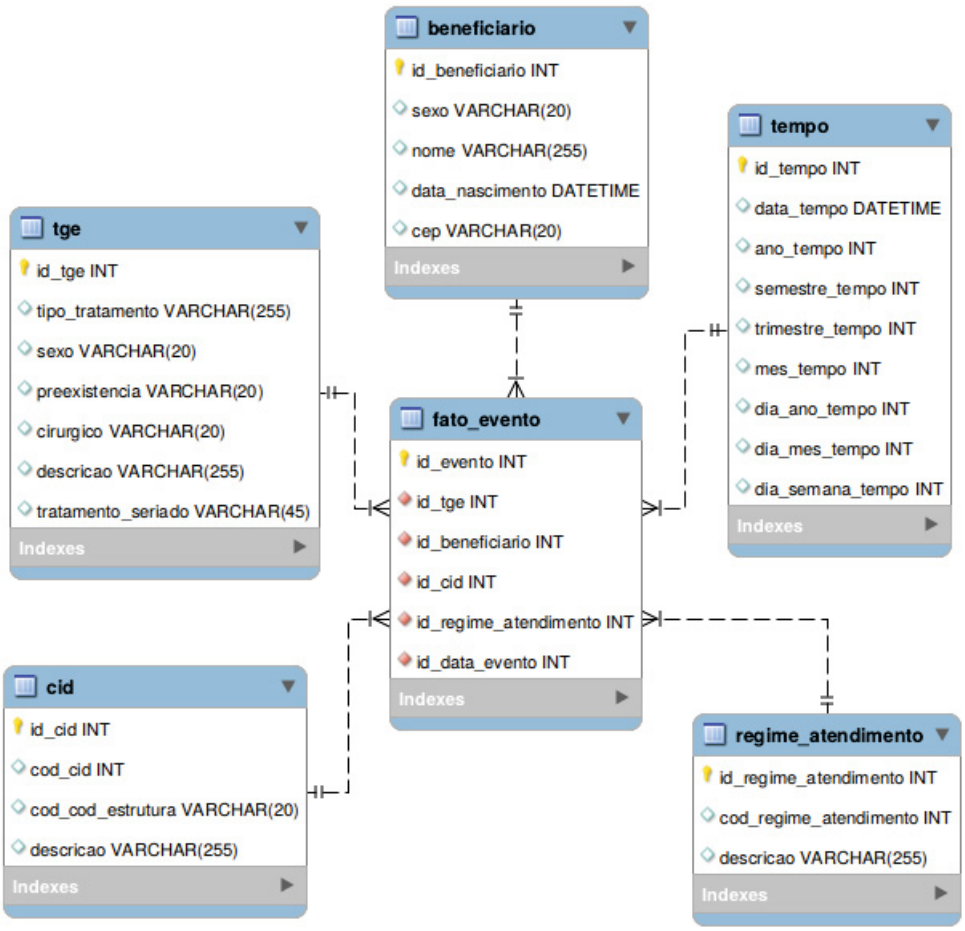


Figura 4.2: Diagrama simplificado do DW utilizado.

Tabela 4.1: Estrutura da tabela da base de dados fornecida.

Coluna	Descrição
ID	Identificador único da ocorrência do evento
BENEFICIARIO	Código identificador do beneficiário
SEXO	Identificação do sexo do beneficiário
DATA_NASCIMENTO	Data de nascimento do beneficiário
DATA	Data da ocorrência do evento
ESPECIALIDADE	Especialidade do atendimento em que o evento ocorreu
REGIME_DE_ATENDIMENTO	Especifica o tipo de serviço relacionado ao evento.
CODIGO_EVENTO	Código identificador da descrição do evento
DESCRICAO_EVENTO	Descrição do evento realizado
CID	Código da CID-10 do beneficiário no momento da geração do procedimento

esclarecer seu significado apresenta-se sua definição: Descrição do Evento refere-se à informação detalhada sobre um procedimento ou atendimento em saúde, conforme a lista estabelecida pela Agência Nacional de Saúde Suplementar (ANS) no Rol de Procedimentos e Eventos em Saúde. Essa lista define consultas, exames, cirurgias, tratamentos e outros serviços que os planos de saúde são obrigados a oferecer, variando de acordo com o tipo de plano contratado (ambulatorial, hospitalar com ou sem obstetrícia, referência ou odontológico). A descrição do evento, portanto, caracteriza a natureza do serviço prestado, sendo um elemento essencial para a organização e análise dos dados em saúde.

4.2.1 Análise das bases DB1 e DB2

Buscando verificar o perfil das bases DB1 e DB2, apresenta-se os histogramas de distribuição de serviços prestados por idade e por ano, conforme Figuras 4.3 e 4.4.

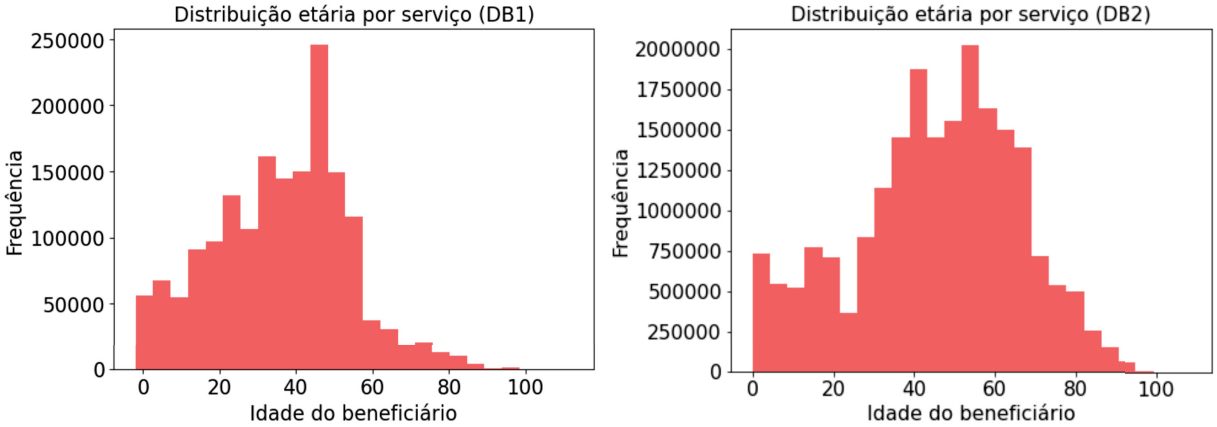


Figura 4.3: Histograma da distribuição de serviços prestados por idade dos beneficiários (Bases DB1 e DB2).

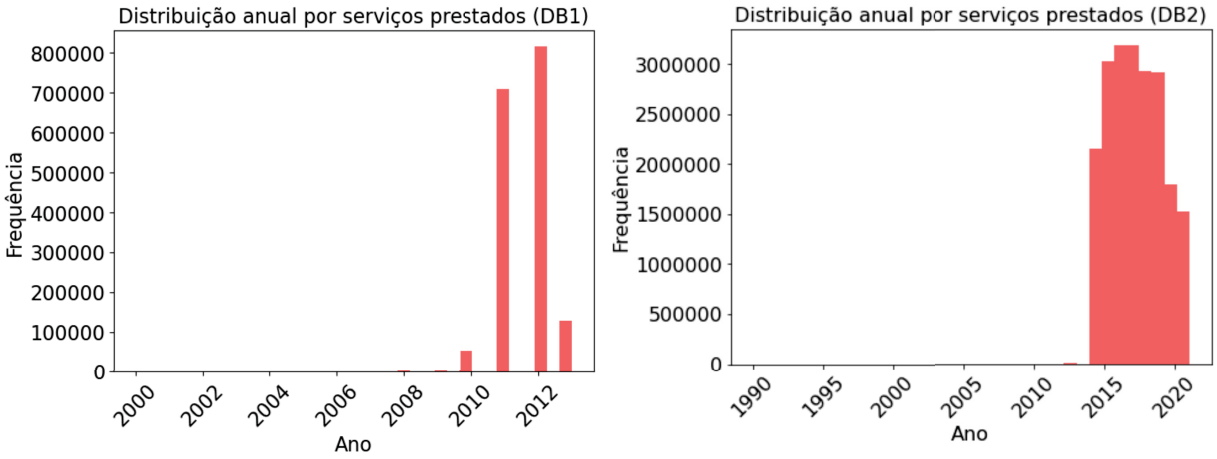


Figura 4.4: Histograma da distribuição de serviços prestados por ano (Bases DB1 e DB2).

Na Figura 4.3, verifica-se que na base DB1 existe uma predominância de beneficiários com menos de 60 anos, enquanto que na base DB2 nota-se que há uma predominância de beneficiários entre 35 e 65 anos, possuindo uma proporção relativamente maior de beneficiários com idade superior a 60 anos. Já na Figura 4.4, pode-se observar que na base DB1 a quantidade de exemplos anteriores a 2010 é praticamente insignificante, concentrando-se os dados no período de 4 anos, entre 2010 e 2013, com predominância de exemplos nos anos de 2011 e 2012. Na

base DB2, a quantidade de exemplos anteriores a 2014 também é insignificante, concentrando os dados no período de 8 anos, entre 2014 e 2021, abrangendo o dobro do intervalo temporal se comparado com a base DB1.

Em relação ao sexo dos beneficiários, tem-se que 19.036 são do sexo masculino e 15.896 do sexo feminino, na base DB1, enquanto na DB2 há 215.432 beneficiários do sexo masculino e 230.079 do sexo feminino, havendo, portanto, uma concentração maior de mulheres, diferentemente da base DB1. Além disso, a base DB1 possui 2.585 CIDs distintas das quais 835 estão relacionadas a eventos de internação. Já a base DB2 possui 8.399 CIDs distintas, das quais 6.913 estão relacionadas a eventos de internação.

4.3 ORGANIZAÇÃO DE DADOS EM HISTÓRICOS DOS BENEFICIÁRIOS

Considerando-se os objetivos deste trabalho, realizou-se o processo de construção do histórico com base em três características extraídas dos dados: Especialidade, Descrição do evento e CID. Essas características foram selecionadas por estarem presentes em ambas as bases de dados disponibilizadas para pesquisa e pelo seu grau de importância no contexto dos dados analisados. Com base nessas características, foram geradas três bases de dados distintas, rotuladas pela ocorrência ou não de internação.

Para chegar até esses dados históricos foram realizados os seguintes procedimentos:

- Junção e seleção dos dados em um única tabela;
- Remoção de ruídos;
- Agregação de dados;
- Encadeamento dos dados.

4.3.1 Junção e seleção dos dados

Buscando simplificar a criação dos históricos, foi realizada a junção dos dados do DW da base DB1 em uma única tabela, e além das três características selecionadas para criação do histórico que são Especialidade, Descrição do evento e CID, utilizou-se também de dados dos beneficiários (como código, sexo e idade, data de ocorrência do evento), fundamentais para a criação do histórico, e o regime de atendimento que permite identificar quais eventos são de internação. O mesmo processo foi realizado para a base DB2, exceto pela junção, pois a base obtida já estava em uma única tabela. A síntese resultante das características e suas descrições pode ser vista na Tabela 4.2.

4.3.2 Remoção de ruídos

Em ambas as bases, foram removidos dos dados exemplos relacionados a beneficiários com idade negativa, evidenciando claramente erro nos exemplos. Observando os dados da base DB1, notou-se que um conjunto relativamente pequeno dos dados pertencia a datas anteriores a 2010 e que estava cronologicamente distante do conjunto. Sendo assim, realizou-se a remoção dos registros anteriores a 2010. O mesmo foi feito com a base DB2 para registros anteriores a 2014. As Figuras 4.5 e 4.6 demonstram a distribuição dos dados antes e após a remoção de ruídos para a base DB1 e DB2, respectivamente.

Tabela 4.2: Características selecionadas.

Característica	Descrição
ID do Beneficiário	Código identificador dos exemplos de um beneficiário
Idade	Idade do beneficiário no dia da ocorrência do procedimento
Sexo	Identificação do sexo do beneficiário
Data	Data da ocorrência do procedimento
Especialidade	Especialidade do atendimento em que o procedimento foi realizado
Descrição do evento	Descrição do procedimento realizado
CID	Código da CID-10 do beneficiário no momento da geração do procedimento
Regime de Atendimento	Se o atendimento foi em regime ambulatorial, internação hospitalar, internação - hospital dia ou assistência domiciliar.

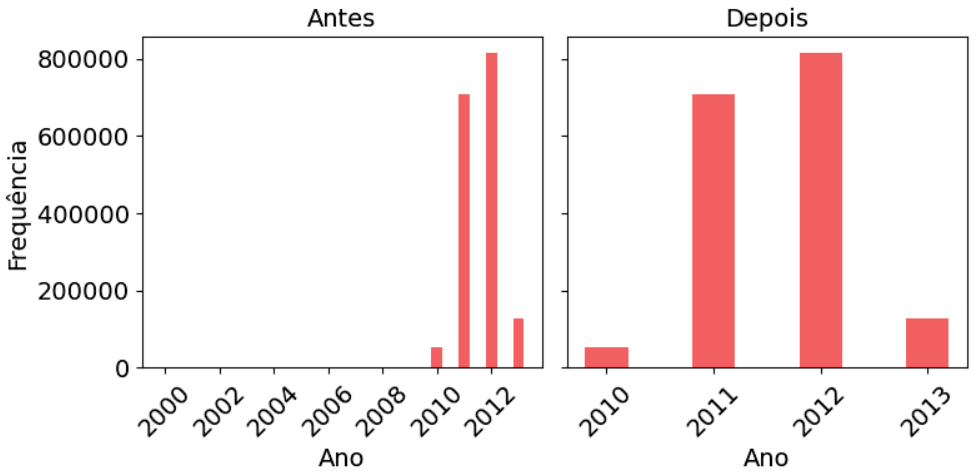


Figura 4.5: Distribuição dos dados da base DB1 antes e após a remoção de ruídos.

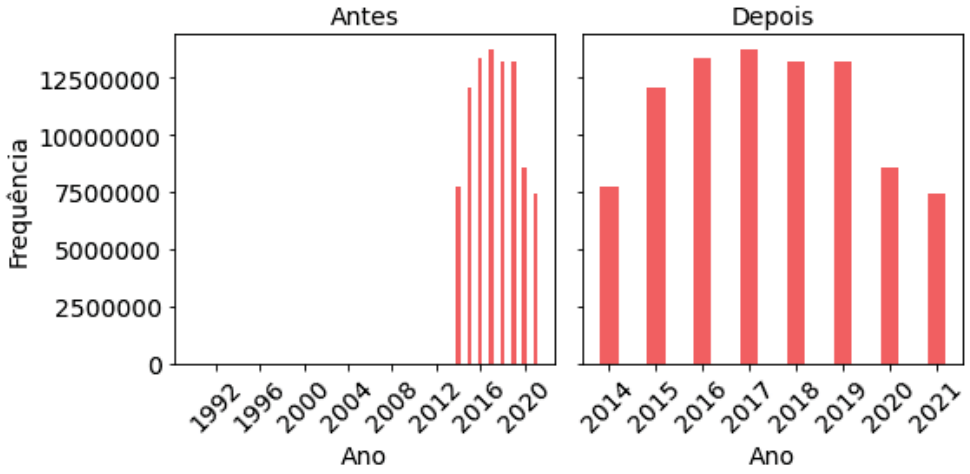


Figura 4.6: Distribuição dos dados da base DB2 antes e após a remoção de ruídos.

4.3.3 Agregação de dados

Recorrentemente, quando um beneficiário entra em contato com serviços de saúde, são gerados vários registros de eventos e procedimentos numa mesma data. Verificou-se isto em muitos registros das bases de dados em questão. Observando-se, então, os exemplos de um beneficiário em uma mesma data, constatou-se que todas as características dos dados (ex.: especialidade, regime de atendimento, CID etc.) são as mesmas. A exceção verificada foi quanto à descrição do evento, que registra os procedimentos e insumos relacionados com o beneficiário em um atendimento.

Desta forma, com o objetivo de se criar um registro histórico, para um beneficiário, com base em alguma característica dos dados, buscou-se, então, permitir apenas um registro por data para cada beneficiário. Para chegar a esta configuração, foi necessário resolver o problema da descrição do evento, uma vez que, nos dados, é normal haver vários eventos em um mesmo dia para um mesmo beneficiário. Assim, o processo de agregação dos dados foi realizado para todas as características, e as descrições dos eventos foram concatenadas em um único registro. A Figura 4.7 exemplifica o processo para os registros de um beneficiário em uma mesma data.

Beneficiario	Idade	Data	Sexo	Especialidade	Regime de Atendimento	Descrição do Evento	CID
28739	50	2011-08-22	M	CLÍNICA MÉDICA	INTERNAÇÃO HOSPITALAR	AGULHA DESCARTAVEL 25 X 07	J32.9
28739	50	2011-08-22	M	CLÍNICA MÉDICA	INTERNAÇÃO HOSPITALAR	AGULHA DESCARTAVEL 40 X 12	J32.9
28739	50	2011-08-22	M	CLÍNICA MÉDICA	INTERNAÇÃO HOSPITALAR	CATETER VENOSO PERIFERICO NO 18	J32.9
28739	50	2011-08-22	M	CLÍNICA MÉDICA	INTERNAÇÃO HOSPITALAR	CATETER NASAL TIPO OCULOS	J32.9
28739	50	2011-08-22	M	CLÍNICA MÉDICA	INTERNAÇÃO HOSPITALAR	DISPOSITIVO DE INFUSAO MULTIPLA DE 3 VIAS	J32.9

Beneficiario	Idade	Data	Sexo	Especialidade	Regime de Atendimento	Descrição do Evento	CID
28739	50	2011-08-22	M	CLÍNICA MÉDICA	INTERNAÇÃO HOSPITALAR	AGULHA DESCARTAVEL 25 X 07 AGULHA DESCARTAVEL 40 X 12 CATETER VENOSO PERIFERICO NO 18 CATETER NASAL TIPO OCULOS DISPOSITIVO DE INFUSAO MULTIPLA DE 3 VIAS	J32.9

Figura 4.7: Exemplo de agregação dos dados.

Após a remoção de ruídos e a agregação, reduziu-se o número de registros de 1.703.960 para 578.094, para a base DB1, e de 89.339.526 para 20.721.377, para a base DB2. Após estas etapas, cada exemplo passou a representar o atendimento do beneficiário em um determinado dia em que todas as descrições de todos os eventos neste dia estão agregados em um único exemplo.

4.3.4 Encadeamento dos dados

Após a agregação, cada beneficiário passou a ter no máximo um exemplo por dia. Esta condição dos dados permite organizar as descrições dos eventos, ou outro atributo de cada beneficiário cronologicamente, formando assim o que chamamos, neste trabalho, de narrativa histórica do paciente. Esta construção permite aos algoritmos de aprendizado de máquina, após a extração de características e treinamento, relacionar diversos eventos de saúde pelos quais o

beneficiário passou. Considerando que o objetivo principal desse trabalho é prever internações, esses históricos foram arranjados considerando exemplos até o dia anterior a um exemplo de internação. Como um beneficiário pode ter passado por mais de uma internação, várias narrativas históricas podem ser geradas para um mesmo beneficiário. Para os casos em que há internação, o exemplo é rotulado com o valor 1 e 0 para o caso contrário.

Para gerar as narrativas históricas, realizou-se o encadeamento dos dados de um dos atributos, Especialidade, Descrição do evento ou CID, até o momento em que um registro de internação é encontrado. Para identificar se um registro é de internação, o atributo "Regime de Atendimento" precisa conter o valor "Internação Hospitalar" ou "Internação - Hospital dia". Como há registros nos dados de beneficiários com nenhuma, uma ou várias internações, é necessário executar um processo lógico para gerar a narrativa histórica. Para exemplificar o processo de encadeamento dos dados, apresenta-se a Figura 4.8 para exemplos históricos com duas ou nenhuma internação.

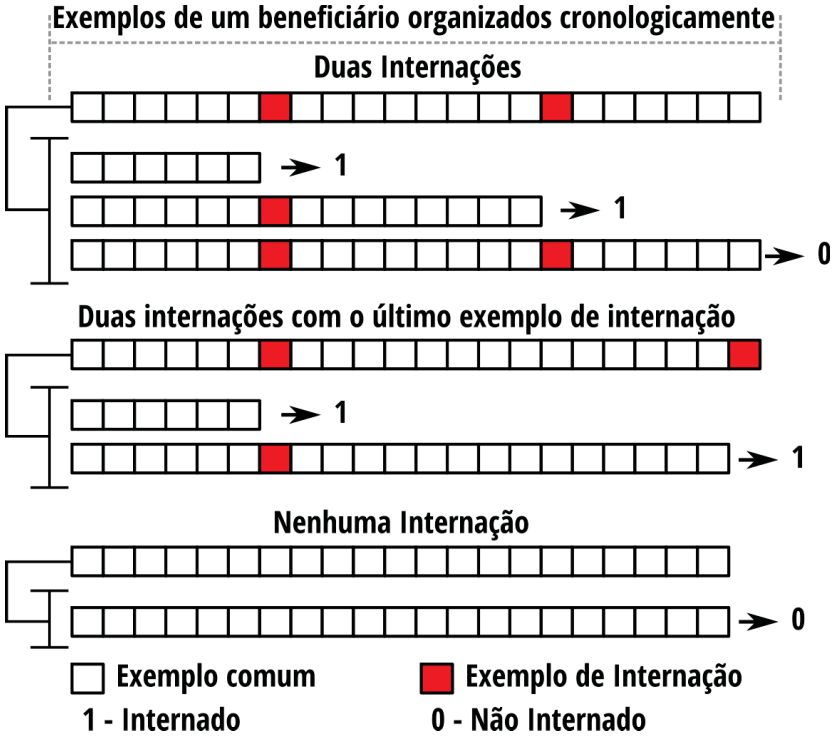


Figura 4.8: Processo de encadeamento dos dados e geração de narrativa histórica.

Após o processo de transformação dos dados, gerou-se uma estrutura de dados conforme apresentado na Tabela 4.3, em que o *id* contém o código identificador dos registros históricos de um mesmo beneficiário, *sentença* contém a sequência cronológica dos dados separados por espaço em branco em uma única *string* e *classe* contém valores 0 ou 1 que representam "não internado" ou "internado", respectivamente.

Além disso, no fim de cada sentença foi concatenado o valor do atributo "sexo" do beneficiário, dado considerado importante para os treinamentos. Para a base DB1, foram criados três conjuntos de dados, um para cada uma das características: Especialidade, Descrição do evento e CID. Portanto, há um conjunto de dados referente ao histórico de especialidades, um conjunto de dados referente ao histórico de eventos e um conjunto de dados referente ao histórico de CIDs do beneficiário. Para a base DB2, foi criado apenas o conjunto referente ao histórico de eventos, pois a proporção de CIDs informadas é menor, e, além disso, percebeu-se, durante os experimentos preliminares com a base DB1, que a descrição dos eventos possuía maior poder

Tabela 4.3: Estrutura resultante da criação dos históricos.

Característica	Tipo
id	Inteiro
sentença	String
classe	Inteiro

discriminante. Para exemplificar o resultado da organização dos dados em sentenças históricas, o apêndice A apresenta um exemplo concreto da sentença histórica com sexo do beneficiário adicionado ao final da sentença gerada a partir da base DB2.

4.3.5 Bases de dados resultantes

Após a geração das sentenças históricas, as bases passaram a ter a seguinte configuração:

- Base DB1

  Número de exemplos: 38.524;

  Exemplos para internação: 3.772;

  Exemplos para não internação: 34.752.

- Base DB2

  Número de exemplos: 880.193;

  Exemplos para internação: 451.649;

  Exemplos para não internação: 428.544.

4.4 EXTRAÇÃO E SELEÇÃO DE CARACTERÍSTICAS DOS DADOS

Cada exemplo pertencente aos conjuntos de dados gerados pelas etapas anteriores contém pontos que são dependentes de outros pontos. Portanto, por definição, esses dados podem ser denominados sequenciais. A forma como foram organizados se assemelha muito a uma sentença de texto, e neste trabalho foram rotulados para internação ou não. Desta forma, considerando cada exemplo como uma sentença de texto, pode-se utilizar métodos de PLN para fazer as extrações de características.

Na revisão da literatura, observou-se que a maioria dos trabalhos relacionados a previsões de internações realiza a extração e seleção de características considerando a literatura médica para o problema de saúde ao qual as internações estão relacionadas, ou concentra-se em poucas características de fácil acesso, selecionadas por algum critério relacionado a problemas de saúde. Na abordagem adotada aqui, a extração e a seleção de características são feitas de forma automática, deixando por conta das técnicas de PLN adotadas essa responsabilidade. Desta maneira, características que inicialmente poderiam ser ignoradas pelos profissionais da saúde são selecionadas para colaborar para a predição. Além disso, elas podem representar descobertas em relação a fatores não correlacionados anteriormente a determinados problemas de saúde. Para esses casos, a aplicação de métodos de interpretabilidade como SHAP pode ajudar na descoberta desses fatores.

Foram utilizadas duas abordagens, a primeira faz uso da codificação de *tokens* em um único número para posterior criação da sequência. A segunda utilizou métodos de extração de características das sentenças por meio do *framework sentence transformers*, ou diretamente da LLM quando se utiliza a camada completamente conectada para classificação. A seguir apresenta-se o processo de pré-processamento dos dados e as técnicas de extração utilizadas.

4.4.1 Pré-processamento das sentenças

Independentemente do método utilizado para extração de características, as sentenças precisam passar por um processo de preparação. Quando se utiliza LLMs, parte significativa do pré-processamento das sentenças é feita pelo próprio modelo, uma vez que o tokenizador, vocabulário e gerador de sequências é parte integrante do modelo. Quando não se utiliza LLMs, um processo de preparação precisa ser executado. Para esses casos, adotou-se o processo conforme o fluxograma da Figura 4.9, que apresenta as etapas executadas para preparação dos dados. As sentenças do conjunto de dados passam todas por um processo de tokenização para extração dos *tokens*, cada um deles é então indexado a um número inteiro único formando o vocabulário da base de dados. Essa indexação é utilizada para o processo de vetorização das sentenças, transformando-as em sequências numéricas. Após o processo de vetorização, as sequências passam por um processo de pré-preenchimento e pré-truncamento para que todas as sequências tenham o mesmo tamanho, conforme descrito nas seções 2.4.5.1 e 2.4.5.2.

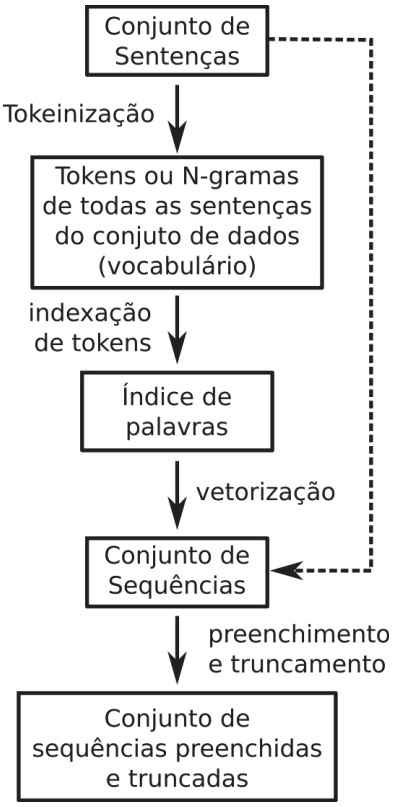


Figura 4.9: Fluxograma do processo adotado neste trabalho para extração de características e transformação dos dados.

Além da função de manter o mesmo tamanho das sentenças, o pré-preenchimento e o pré-truncamento foram escolhidos com o objetivo de manter nas sequências os dados mais recentes antes das internações. Portanto, quando for necessário eliminar parte da sequência (truncamento), o que se descartou foi sempre a parte mais antiga do histórico. O tamanho das

sequências foi determinado de acordo com o método utilizado para extração de características. A Figura 4.10 apresenta um exemplo da execução dessas etapas para um conjunto de dados de duas sentenças.

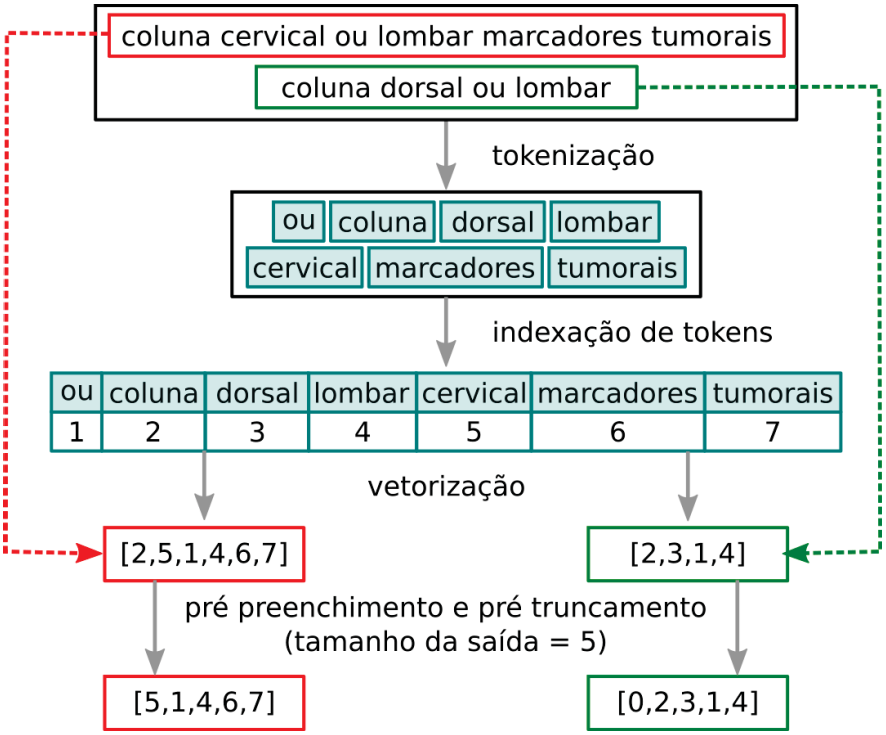


Figura 4.10: Processo de tokenização, preenchimento e truncamento realizados sobre o campo sentença dos dados.

4.4.2 Extração de sequências das sentenças da base DB1

A base DB1 foi preparada ainda na primeira etapa deste trabalho, e foi utilizada para o treinamento do *Random Forest* e *Gradient Boosting*. Para o treinamento, foi utilizada a sequência final gerada conforme o processo apresentado na Figura 4.10. Este processo é equivalente ao apresentado nas seções 2.4.2 e 2.4.5, gerando uma sequência de números de mesmo tamanho que representam as sentenças. A extração de sequências foi realizada para Especialidade, Descrição do Evento e CID e o tamanho das sequências foi determinado pelo número médio de *tokens* entre todas as sequências do conjunto de dados. A Tabela 4.4 apresenta o tamanho das sequências utilizadas para cada conjunto de dados gerado. Nessa extração foi utilizado o pré-preenchimento e o pré-truncamento, descartando-se, sempre quando aplicável, as partes da sequência que representam os dados mais antigos.

Tabela 4.4: Tamanho da sequência por Conjunto de Dados.

Conjunto de dados	Qtd. de características
Especialidade	49
Descrição do Evento	364
CID	20

Após esse processo, obteve-se como resultado sequências para a base DB1 que possui um conjunto de dados de 38.524 registros rotulados como “internado” (1) ou “não internado”

(0), sendo 3.772 para internados e 34.752 para não internados. A razão entre as duas classes é de 9:1, sendo consideravelmente desbalanceadas. Deste modo, aplicou-se o *undersampling* (Fernández et al., 2018), buscando balancear as duas classes. Assim, chegou-se a um conjunto de dados com 7.544 exemplos, sendo 3.772 para internados e 3.772 para não internados. Estes dados foram então utilizados para treinamento e teste para o *Random Forest* e *Gradient Boosting*. A estrutura final dos dados foram incorporados em uma tabela (Tabela 4.5) para aplicação nos algoritmos de classificação, em que *n* é o número de características do conjunto de dados.

Tabela 4.5: Exemplo da estrutura final dos dados para cada conjunto (DB1).

Característica	**Tipo**
id	Inteiro
C0	Inteiro
C1	Inteiro
...	...
Cn	Inteiro
classe	Inteiro

4.4.3 Extração de *embeddings* das sentenças da base DB2

Na base DB2, foi utilizado o framework *Sentence Transformers* (Reimers e Gurevych, 2019) para extrair *embeddings* das sentenças históricas dos pacientes. Para fazer a extração, é necessário utilizar um modelo de LLM pré-treinado. Nesse trabalho, as LLMs são de extrema importância e além de serem utilizadas para extração dos *embeddings*, também foram utilizadas para geração de modelos preditores. Desta forma, buscando maximizar os resultados, diversos treinamentos foram realizados sobre os modelos, gerando novos modelos para análise comparativa. O BERTimbau (Souza et al., 2020), o Open-Cabrita3b (Larcher et al., 2023) e o RoBERTa (Liu et al., 2019) foram os modelos escolhidos, sendo deste último utilizada apenas sua arquitetura para treinamento de um modelo com os dados exclusivos da base DB2. As LLMs dependem de um tokenizador próprio e o utilizam para gerar a sequência de entrada para treinamento e inferência do modelo. Portanto, para extrair os *embeddings* das sentenças utilizando o framework *Sentence Transformers*, basta passar as sentenças e o modelo para o framework que ele retorna os *embeddings* das sentenças. A Figura 4.11 exemplifica o processo de extração de embeddings por meio do *framework*.

Esse processo de extração dos *embeddings* por meio do *Sentence Transformers* foi necessário apenas quando se pretendeu utilizar outros métodos de treinamento como o *Random Forest* e *Gradient Boosting*. Quando utiliza-se a própria LLM com uma camada completamente conectada para classificação, essa extração não é necessária.

4.5 DIVISÃO DOS DADOS PARA TREINAMENTO E TESTE

Das duas bases preparadas contendo as sentenças históricas, apresenta-se aqui as divisões realizadas para treinamento e teste. Além disso, elas foram nomeadas para facilitar a referência a elas ao longo do texto.

A primeira divisão que apresentamos é a divisão da base DB1. Ela foi utilizada para o treinamento com o *Random Forest* e *Gradient Boosting*, que chamamos aqui de métodos

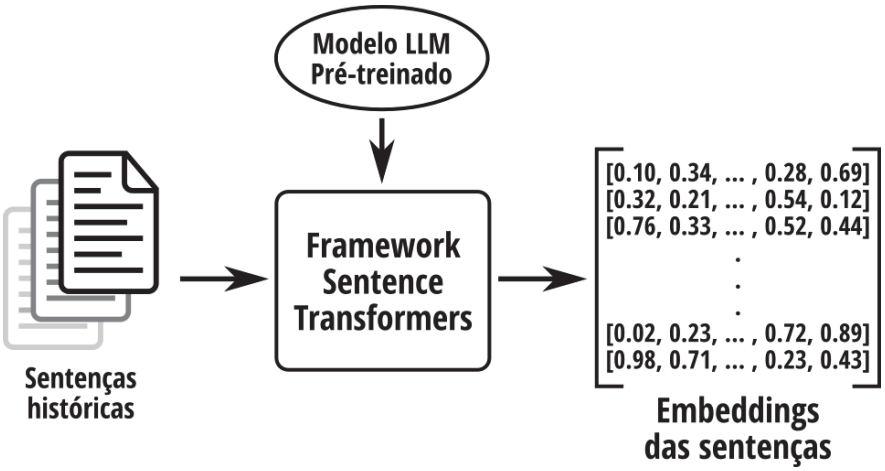


Figura 4.11: Processo de extração de *embeddings* por meio do *framework Sentence Transformers*.

convencionais. Devido ao desbalanceamento dos dados, como citado, foi aplicado *undersampling*, resultando em uma base com 7.544 exemplos perfeitamente balanceados. Nesta base, a divisão se deu em 70% para treinamento e 30% para teste. Esse particionamento foi realizado cinco vezes selecionado os dados de aleatoriamente para utilização nos treinamentos dos modelos. A Figura 4.12 exemplifica a divisão e a nomenclatura utilizada. Para garantir a generalização dos modelos, não foi permitido que exemplos de pacientes do conjunto de treinamento também estivessem nos dados de teste.

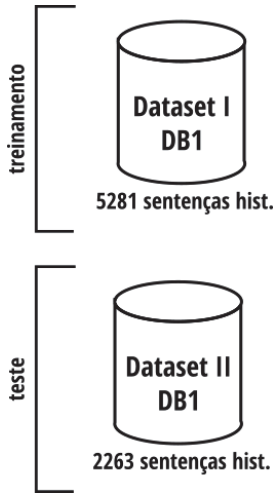


Figura 4.12: Divisão da base DB1 para treinamento e teste.

A segunda divisão é da base DB2. Contendo 880.193 sentenças históricas, ela foi utilizada no treinamento das LLMs. Foi dividida em três partes, sendo a maior para treinamento e *fine-tuning* das LLMs, e duas partes para teste e treinamento dos modelos de previsão de internações que utilizam de alguma forma LLMs. A Figura 4.13 exemplifica essa divisão.

4.6 TREINAMENTO DAS LLMS

A fim de avaliar a eficiência de diferentes métodos de previsão de internação por meio de LLMs, realizou-se treinamentos considerando 2 modelos pré-treinados, o BERTimbau e o Open-Cabrita3B, além do treinamento do RoBERTa a partir do zero, somente com dados de

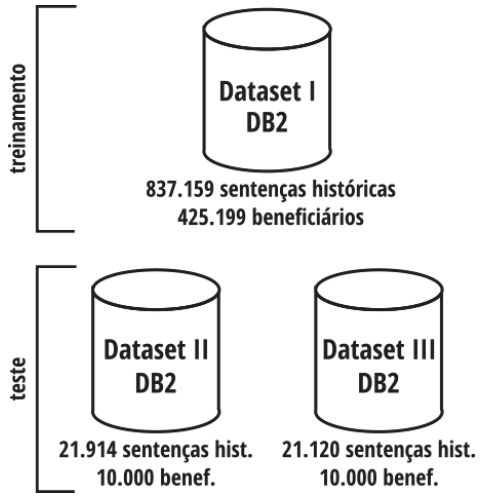


Figura 4.13: Divisão da base DB2 para treinamento e teste.

planos de saúde. Ao final desses treinamentos, foram gerados 3 modelos pré-treinados que foram utilizados nos experimentos deste trabalho. A Figura 4.14 resume a sequência de treinamentos.

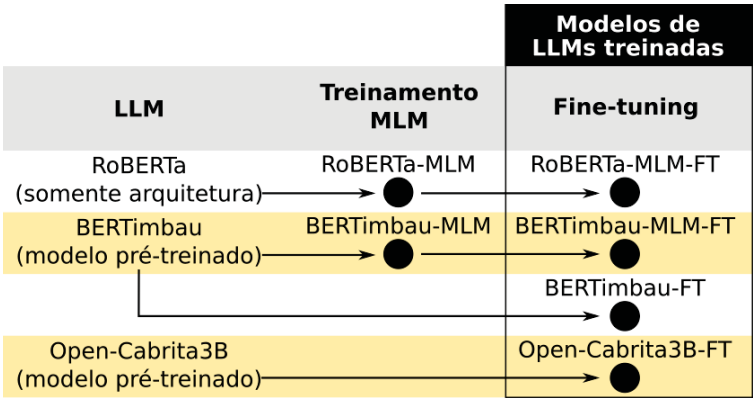


Figura 4.14: Sequência de treinamento das LLMs.

4.6.1 Treinamento do RoBERTa

Considerando que o RoBERTa possui uma estrutura bem otimizada, exigindo *hardware* mais modesto, ele foi escolhido para fazer o treinamento a partir do zero, ou seja, sem a utilização de um modelo pré-treinado. Portanto, todos os pesos desse modelo foram ajustados a partir dos dados históricos dos planos de saúde que constituem o *corpus* deste trabalho.

Para esse treinamento, foi gerado um tokenizador próprio, treinado a partir dos dados agregados das descrições dos eventos da base DB2, portanto, o tokenizador dele é fortemente baseado neste domínio. Para o pré-treinamento do modelo, foi utilizado o método auto-supervisionado MLM sobre os dados históricos dos beneficiários. Foram utilizados os 837.159 exemplos do *DB2-dataset I*, executados em duas épocas, resultando em um modelo que nomeamos de RoBERTa-MLM. O *fine-tuning* foi realizado em duas épocas para tarefa PLN *sequence for classification*, como apresentado na Figura 2.16, utilizando 10% do *DB2-dataset I* com rotulação para internação com um dia de antecedência, este modelo é referenciado neste trabalho de RoBERTa-MLM-FT. Ambos os treinamentos foram realizados com a sequência máxima permitida pelo RoBERTa, que é de 512 tokens. A Figura 4.15 apresenta de forma

resumida a sequência de treinamento realizada para obtenção dos modelos, e a Tabela 4.6, os hiperparâmetros usados nos treinamentos.

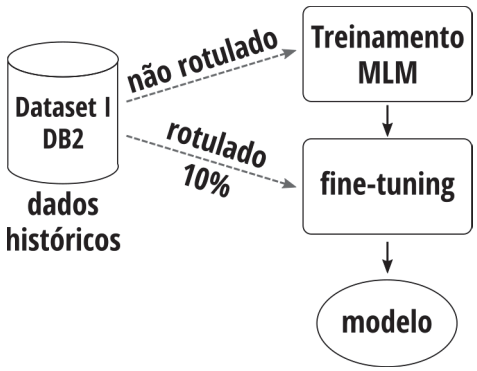


Figura 4.15: Sequência de treinamento RoBERTa.

Tabela 4.6: Hiperparâmetros usados no treinamento MLM e *fine-tuning* do RoBERTa.

Hiperparâmetro	**Valor**
Learning rate	1e-4
Maximum sequence length	512
Gradient accumulation step	4
Batch size	16
Mask %	15%

4.6.2 Treinamento do BERTimbau

O BERTimbau é um modelo BERT treinado em português. De acordo com o trabalho de Souza et al. (Souza et al., 2023), vem atingindo o estado da arte em várias tarefas de PLN, superando os modelos multilinguísticos para português. Assim, considerando que se utiliza neste trabalho dados em português, ele foi escolhido para treinamento e comparação da efetividade da utilização desses dados em seu treinamento, considerando que se trabalhou com dados sobre um domínio específico.

Três seções de treinamento foram conduzidas. A primeira foi o treinamento autossupervisionado MLM a partir do modelo pré-treinando do BERTimbau, resultando em um modelo que chamamos de BERTimbau-MLM. O segundo e o terceiro envolvem o *fine-tuning* para a tarefa PLN *sequence for classification*: um sobre o modelo original do BERTimbau, resultando no BERTimbau-FT, e o outro sobre o BERTimbau-MLM, resultando no BERTimbau-MLM-FT.

O treinamento MLM foi realizado em uma época com os mesmos 837.159 exemplos do *DB2-dataset I*, usados no treinamento do RoBERTa. As seções de *fine-tuning* foram conduzidas em uma época utilizando 10% do *DB2-dataset I*, que consistem de dados históricos com rotulação para internação com um dia de antecedência. Todos os treinamentos foram executados com a sequência máxima permitida pelo BERTimbau, que é de 512 *tokens*. A Figura 4.16 apresenta de forma resumida as sequências de treinamento realizadas para obtenção dos modelos e a Tabela 4.7 os hiperparâmetros usados nos treinamentos.

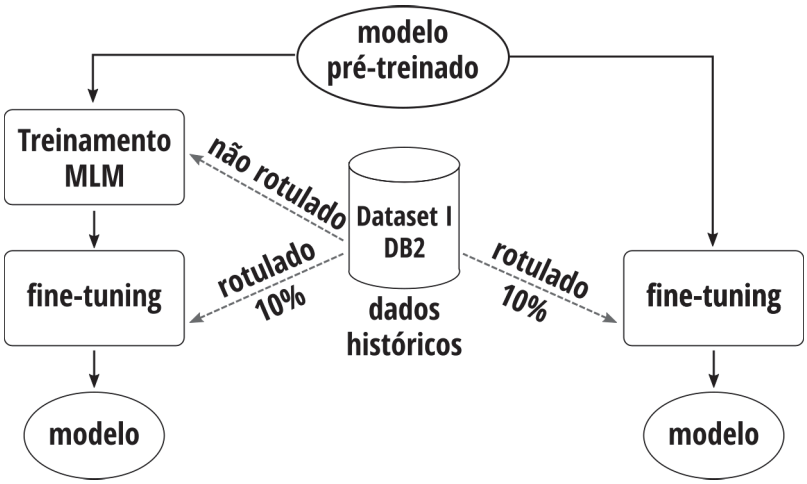


Figura 4.16: Sequências de treinamentos BERTimbau.

Tabela 4.7: Hiperparâmetros usados no treinamento (MLM) e fine-tuning do BERTimbau.

Hiperparâmetro	**Valor**
Learning rate	1e-4
Maximum sequence length	512
Gradient accumulation step	4
Batch size	16
Mask %	15%

4.6.3 Treinamento do Open-Cabrita3B

Considerando que o limite máximo das sentenças para as arquiteturas do BERTimbau e RoBERTa são de 512 tokens, vale verificar o tamanho das sentenças do conjunto de treinamento para saber se uma proporção significativa delas está sendo descartadas. Desta forma, gerou-se um *boxplot* para verificar o tamanho das sentenças da base DB2, considerando o número de tokens. A Figura 4.17 apresenta o gráfico em que se observa que parte significativa das sentenças históricas constituem-se de mais de 512 *tokens*. Portanto, constata-se que parte significativa das sentenças é descartada, tanto no treinamento dos modelos, quanto na inferência das internações. Desta forma, utilizar modelos que possibilitem o treinamento de sentenças maiores permite que se faça um contraponto sobre os outros dois modelos testados e que se verifique se dados mais antigos são importantes na inferência de internações.

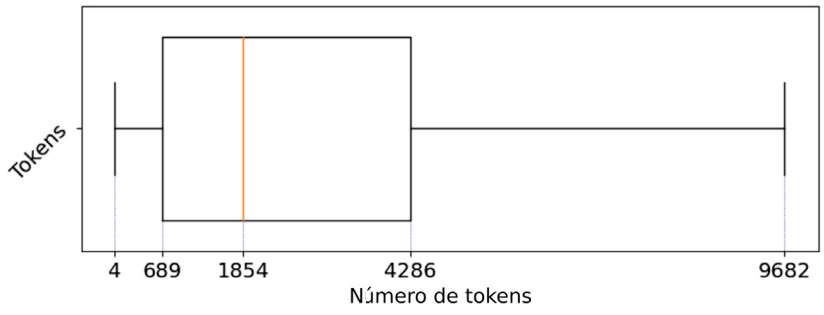


Figura 4.17: Boxplot para número de tokens das sentenças históricas (Tokenizer BERTimbau).

Uma alternativa para sentenças maiores é utilizar modelos que as aceitem. Uma alternativa neste caso é o LLaMA, que em sua primeira versão permite sequências de até 2048 *tokens*, treinado massivamente com dados em inglês. Assim, buscando uma alternativa em português, adotou-se neste trabalho o Open-Cabrita3B (Larcher et al., 2023), que possui 3 bilhões de parâmetros e funciona com sequências de até 2048 tokens. Ele é derivado do OpenLLaMA (Touvron et al., 2023a), uma reprodução aberta do LLaMA.

Devido ao alto custo computacional para o pré-treinamento do modelo, como já foi mencionado na subseção 2.5.4, realizou-se apenas o *fine-tuning* para tarefa PLN *sequence for classification*, utilizando para isto a técnica QLoRA PEFT. Como sequência máxima, utilizou-se o máximo permitido pelo modelo, ou seja, 2048 tokens, e 10% do *DB2-dataset I* dos dados históricos rotulados. O treinamento foi realizado em uma época. A Figura 4.18 apresenta uma síntese da sequência do treinamento realizado para a obtenção do modelo resultante, e a Tabela 4.8 e 4.9 apresentam os hiperparâmetros utilizados do *fine-tuning* e do QLoRA PEFT.

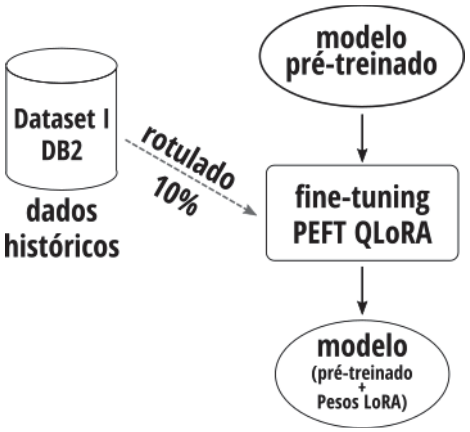


Figura 4.18: Sequências de treinamento Open-Cabrita3B.

Tabela 4.8: Hiperparâmetros usados no fine-tuning.

Hiperparâmetro	**Valor**
Learning rate	5e-4
Maximum sequence length	2048
Gradient accumulation step	16
Batch size	4

Tabela 4.9: Hiperparâmetros do LoRA PEFT usados fine-tuning.

Hiperparâmetro	**Valor**
r	8
alpha	32
dropout	0.1

4.7 TREINAMENTO DO *RANDOM FOREST* E *GRADIENT BOOSTING*

Outra abordagem utilizada para realizar a previsão das internações foi por meio de modelos gerados utilizando-se os dados sequenciais da base DB1, obtidos conforme a Tabela 4.4. Os modelos foram gerados a partir do treinamento do *Random Forest* e *Gradient Boosting*. Esses algoritmos foram escolhidos devido à velocidade de treinamento e aos excelentes resultados reportados na literatura. Além disso, foram utilizados como um fator de comparação para avaliar os modelos gerados a partir das LLMs. A Figura 4.19 apresenta de maneira simplificada o processo de treinamento. Além disso, o *Random Forest* foi também treinado a partir de características da base DB2, extraídas por meio das LLMs. Esses modelos foram utilizados para uma comparação mais direta entre as inferências das LLMs com a camada completamente conectada e inferências de modelos treinados a partir de características extraídas das LLMs. A Figura 4.20 mostra resumidamente esse processo de treinamento. Os hiperparâmetros utilizados nos treinamentos do *Random Forest* e *Gradient Boosting* foram os padrões da implementação do *Scikit-learning* (Pedregosa et al., 2011). Mais detalhes sobre os softwares utilizados estão no apêndice B

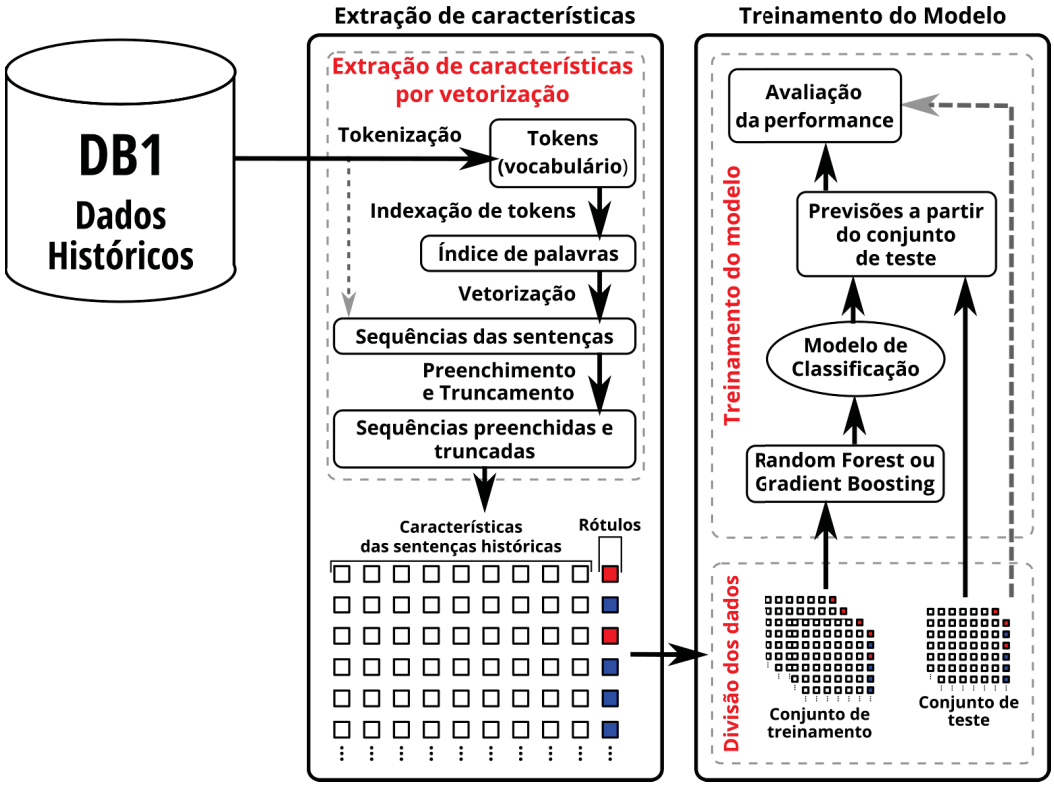


Figura 4.19: Diagrama simplificado do treinamento do *Random Forest* e *Gradient Boosting* por meio de características extraídas por métodos convencionais.

4.8 COMBINAÇÃO DE CLASSIFICADORES

Buscando maximizar os resultados e também avaliar a possível complementariedade entre os modelos, aplicou-se a combinação de classificadores. Foram realizados três métodos de combinação diferentes. A primeira é aplicada para os modelos treinados a partir da base DB1. Esta combinação difere dos métodos tradicionais, pois os modelos combinados são treinados e testados a partir de diferentes fontes de informação, mas que derivam de uma fonte comum, ou

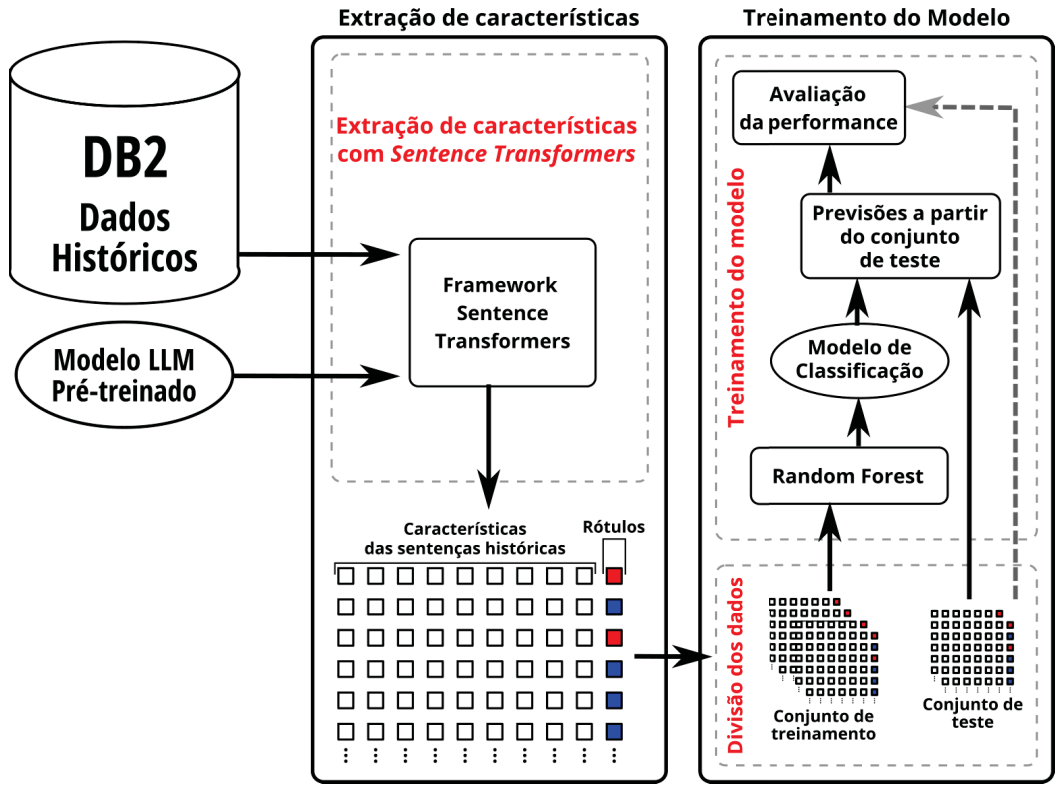


Figura 4.20: Diagrama simplificado do treinamento do *Random Forest* por meio de características extraídas a partir de LLMs.

seja, o que varia é a fonte de informação e não o método de aprendizado. Nele o método de fusão utilizado foi a soma das probabilidades das inferências. A Figura 4.21 exemplifica esse processo.

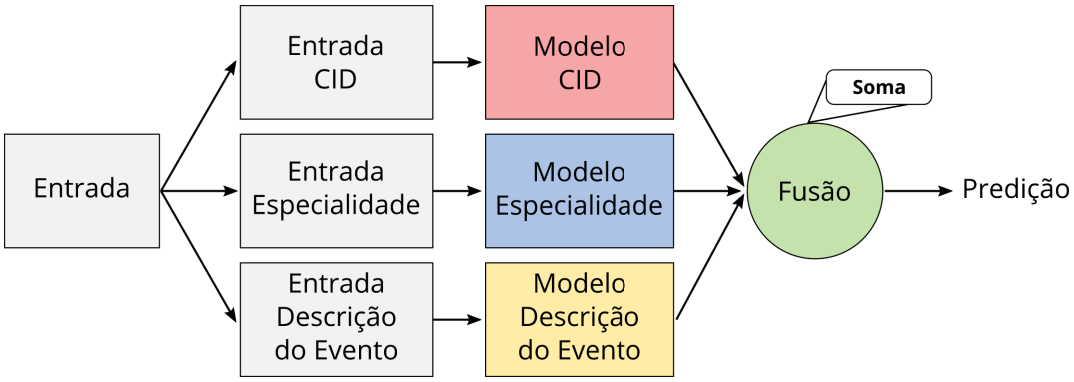


Figura 4.21: Combinação de classificadores treinados a partir de diferentes fontes derivadas.

O segundo e o terceiro métodos de combinação são aplicados para os modelos treinados a partir da base DB2. O segundo faz a combinação entre o *Random Forest*, treinado a partir dos *embeddings* extraídos por uma LLM e a mesma LLM com uma camada completamente conectada para classificação. Nesta combinação, o método de fusão utilizado foi a média das probabilidades das inferências. A Figura 4.22 exemplifica este método.

O terceiro método faz a combinação entre os três classificadores gerados a partir das LLMs com camada completamente conectada. Como método de fusão, também se utilizou a média das probabilidades. A Figura 4.23 exemplifica este método.

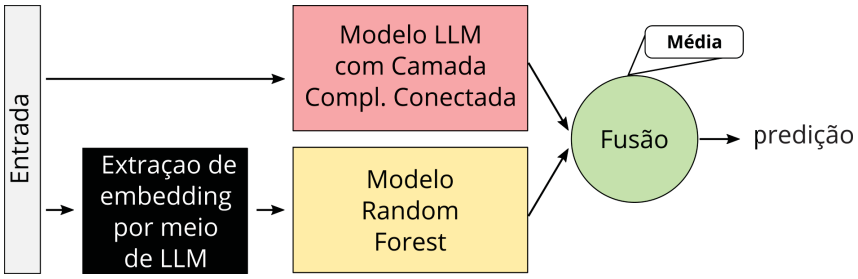


Figura 4.22: Combinação entre modelo com camada totalmente conectada e *Random Forest*.

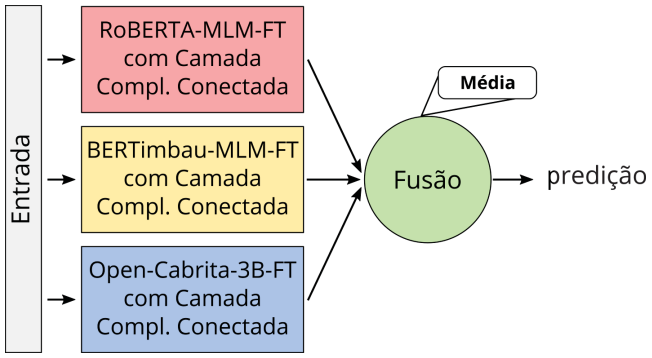


Figura 4.23: Combinação entre modelo RoBERTa-MLM-FT, BERTimbau-MLM-FT e Open-Cabrita3B-FT com camada totalmente conectada.

4.9 MÉTRICAS DE AVALIAÇÃO

Independentemente do tipo de algoritmo utilizado para a tarefa de classificação, os modelos gerados por esses algoritmos precisam ser avaliados. Vários métodos de avaliação podem ser calculados com auxílio de uma matriz de confusão, um método útil para analisar o quão bem um modelo pode reconhecer exemplos de diferentes classes. Em problemas de classificação binária, duas classes são consideradas na predição, a positiva e a negativa. Assim, os resultados possíveis para a classificação de uma instância de teste são quatro, divididos em dois grupos, o grupo das predições corretas: Verdadeiros Positivos (VP) ou Verdadeiros Negativos (VN), o grupo das predições incorretas: Falsos Positivos (FP) ou Falso Negativos (FN). A Tabela 4.10 exibe uma matriz de confusão para duas classes.

Tabela 4.10: Matriz de confusão para predição de duas classes.

		Classe Verdadeira	
		Positivo	Negativo
Classe prevista	Positivo	Verdadeio Positivo (VP)	Falso Positivo (FP)
	Negativo	Falso Negativo (FN)	Verdadeiro Negativo (VN)

Para avaliar os modelos deste trabalho, foram utilizadas as métricas Sensibilidade, Especificidade, *F1-Score* e AUC (*Area under the ROC Curve*). Sensibilidade e Especificidade

foram escolhidas por serem métricas muito utilizadas na área da medicina, pois o custo de classificações incorretas é normalmente muito alto. Por exemplo, o custo de se classificar incorretamente um paciente doente como sadio para uma dada doença grave é muito maior do que classificar um paciente sadio como doente, pois, no primeiro caso, a falta do diagnóstico pode levar à morte do paciente (Prati et al., 2008). Analogamente, considerando o problema deste trabalho, não prever uma internação para um caso grave também pode levar à morte do paciente.

Segundo Guimarães (Guimarães, 1985), sensibilidade é a capacidade que um teste tem de discriminar, dentre os suspeitos de uma patologia, aqueles efetivamente doentes, e especificidade é a capacidade que o mesmo teste tem de ser negativo, em face de uma amostra de indivíduos que sabidamente não tem a doença em questão. Essas informações são de extrema importância neste trabalho, dependendo da aplicação que se fará com o modelo, se na medicina ou na gestão. A definição da sensibilidade é dada pela Equação 4.1 e a especificidade pela Equação 4.2. A equação da sensibilidade é a mesma do *Recall* que é mais comumente utilizada em avaliação de modelos de aprendizagem de máquina.

$$Sensibilidade = \frac{VP}{VP + FN} \tag{4.1}$$

$$Especificidade = \frac{VN}{VN + FP} \tag{4.2}$$

Além da Sensibilidade e da Especificidade, o *F1-Score* também foi utilizado. Ele leva em consideração a precisão e o *recall*, e é considerada uma medida melhor do que a precisão quando conjuntos de dados desbalanceados são utilizado para classificação binária. A equação da precisão é dada pela Equação 4.3, o *recall* é dado pela mesma fórmula da Equação 4.1, e o *F1-Score* é definido pela Equação 4.4.

$$Precisao = \frac{VP}{VP + FP} \tag{4.3}$$

$$F1 - Score = \frac{VP}{VP + \frac{1}{2}(FP + FN)} = 2 \cdot \frac{precisao \cdot recall}{precisao + recall} \tag{4.4}$$

Utilizou-se também a curva ROC (*Receiver Operating Characteristc*) (Fawcett, 2006) para avaliar e compreender os modelos gerados. Na revisão da literatura realizada, ela foi utilizada por parte significativa dos trabalhos para avaliar os modelos. A curva ROC apresenta a taxa de verdadeiros positivos sobre a taxa de falsos positivos em várias configurações de limiares. Na literatura médica (Hajian-Tilaki, 2013; Park et al., 2004; Ma et al., 2013) é comum a utilização da sensibilidade e da especificidade para afastar ou confirmar uma hipótese diagnóstica, como mencionado anteriormente, e a curva ROC relaciona exatamente essas duas métricas. Além disso, por meio da curva ROC é possível definir um limiar que permite escolher qual taxa de falsos positivos é aceitável. Neste trabalho, a determinação desse limiar permite escolher o máximo de previsões de internações falsas que pode ser tolerado pelo modelo. Para comparar os modelos por meio de um único número considerando a curva ROC, utilizou-se a área sob a curva, comumente chamada de AUC (*Area under the ROC Curve*).

4.10 CONSIDERAÇÕES FINAIS

Neste capítulo foi apresentada a proposta de geração de modelos de previsão de internações por meio de dados estruturados de planos de saúde. O método proposto busca criar modelos para predizer internações generalistas e também para problemas específicos que são

avaliados no próximo capítulo. O processo de extração de características apresentado se diferencia dos encontrados na literatura sendo um processo com maior nível de automatização, deixando a cargo dos algoritmos a escolha das melhores características. Além disso, apresentou-se o processo de treinamento de diferentes LLMs que são utilizados em diversos experimentos, além dos processo de treinamento do *Random Forest* e *Gradient Boosting*.

No próximo capítulo são apresentados os resultados e a discussão dos experimentos realizados para a previsão de internações.

5 EXPERIMENTOS E RESULTADOS

Buscando provar a hipótese desta pesquisa, este capítulo apresenta uma série de experimentos realizados usando o conjunto de dados DB1 e DB2. Inicialmente é definido o protocolo experimental adotado. Em seguida, são apresentados e discutidos os resultados obtidos por um sistema de classificação supervisionado convencional utilizando *Random Forest* e *Gradient Boosting*. Esta abordagem constitui a base de comparação com outras abordagens. Depois apresenta-se um resumo dos resultados alcançados pela aplicação da abordagem com as LLMs, bem como uma comparação com os melhores resultados do método convencional. Além disso, são aplicadas as estratégias de fusão dos classificadores, utilizados a fim de avaliar a possível complementaridade entre as representações, visando melhorar o desempenho das previsões. Na sequência, apresenta-se a aplicação dos melhores modelos para previsão de internações por AVC (Acidente Vascular Cerebral), buscando verificar se o modelo generalista construído tem vantagens em relação aos modelos treinados exclusivamente com dados para esse problema específico. Finalmente, apresenta-se uma comparação dos melhores resultados com os encontrados na literatura, a aplicação do método SHAP sobre a inferência de uma internação por AVC, além de uma discussão sobre os resultados.

5.1 PROTOCOLO EXPERIMENTAL

Os experimentos conduzidos nesse trabalho são baseados nos dois conjuntos de dados resultantes das etapas apresentadas nas seções 4.3 e 4.4. A base de dados DB1 foi utilizada para o treinamento e teste para o que chamamos de sistema de classificação convencional, em que foram utilizados os algoritmos *Random Forest* e *Gradient Boosting* para geração dos modelos preditivos. Foram utilizadas as bases derivadas com Especialidade, Descrição do evento e CID, construídos conforme metodologia apresentada na seção 4.4.2. Cada um desses conjuntos foi aleatoriamente dividido em duas partes, 70% para o conjunto de treinamento e 30% para teste conforme Figura 4.12. Para garantir que o classificador estivesse generalizando para novos beneficiários, garantiu-se que os pacientes pertencentes ao conjunto de treinamento não estivessem no conjunto de teste. O particionamento em conjuntos de treinamento e teste foi realizado cinco vezes, resultando em diferentes conjuntos a cada execução dos algoritmos. Os resultados apresentados representam a média das cinco execuções. Além disso, foi realizada a combinação dos classificadores considerando as diferentes bases derivadas, conforme apresentado na seção 4.8 e exemplificado na Figura 4.21.

Em relação as LLMs, a fim de determinar a qualidade e o melhor método para inferência de internações generalistas, foram realizados experimentos com os três modelos resultantes, descritos na seção 4.6. Os modelos RoBERTa e BERTimbau foram testados com os *embeddings* extraídos por meio do *framework Sentence Transformers*, e aplicados ao *Randon Forest*. Além disso, foi realizada a inferência direta a partir da LLM, adicionando uma camada completamente conectada para a classificação de sequências. O modelo Open-Cabrita3B foi testado somente com a camada completamente conectada devido à utilização do treinamento com PEFT, dificultando a extração dos *embeddings* por meio do *framework Sentence Transformers*. A Figura 5.1 apresenta de forma resumida a sequência de passos realizados.

Além disso, foram realizados dois tipos de combinação entre os classificadores. A primeira abordagem faz a combinação entre o *Random Forest*, treinado a partir dos *embeddings* extraídos pela LLM, conforme apresentado na seção 4.8 e exemplificado pela Figura 4.22. A

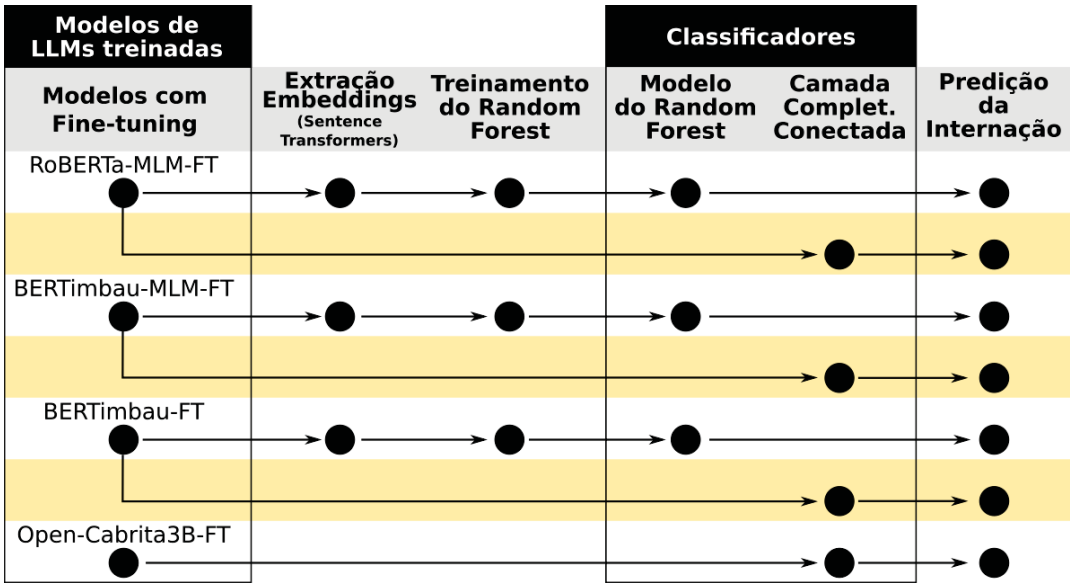


Figura 5.1: Sequência de passos dos experimentos.

segunda abordagem faz a combinação entre os três classificadores gerados a partir das LLMs com a camada completamente conectada, conforme apresentado na seção 4.8 e exemplificado pela Figura 4.23.

Os dados utilizados nos testes dos métodos que utilizam LLMs de alguma forma correspondem ao *DB2-dataset II e III*, apresentado pela Figura 4.13, dados esses não utilizados no treinamento das LLMs. O *DB2-dataset II* foi utilizado para experimentos das LLMs generalistas e extração de *embeddings* para o *Random Forest*. Nestes casos o *DB2-dataset II* foi dividido em 15% para teste, ou seja, 3287 exemplos, e para os casos em que as LLMs foram utilizadas apenas como extratores de características, utilizou-se os 85% restantes, ou seja, 18626 exemplos para essa extração e treinamento do *Randon Forest*, conforme a Figura 4.20.

Para avaliar a capacidade dos modelos preverem internações para casos específicos, foram selecionados exemplos dos *DB2-datasets II e III* de casos de internações para AVC. Além disso, para esses mesmos dados foram geradas sentenças históricas para diferentes períodos de antecedência, 5, 15, 30 e 60 dias com objetivo de avaliar a capacidade dos modelos em prever internações de diferentes períodos de antecedência.

Todos os experimentos foram avaliados pelas métricas *F1-Score*, Sensibilidade, Especificidade e AUC, além da apresentação da curva média ROC dos modelos gerados.

5.2 RESULTADOS POR MEIO DE MÉTODOS CONVENCIONAIS

A Tabela 5.1 apresenta o desempenho do *Random Forest* e *Gradient Boosting* para os três conjuntos derivados dos dados e a combinação de classificadores aplicados conforme a Figura 4.21. Observa-se que, a combinação do GB obteve o melhor resultado para F1-Score e AUC, com valores muito próximos dos modelos gerados com “Descrição de Evento”. Esses resultados demonstram que a representação de dados de planos de saúde, mantendo sua linearidade temporal em forma textual, apresenta uma certa eficácia para utilização na geração de modelos de previsão de internação. Além disso, demonstra que as sentenças históricas formadas pelas descrições de eventos possuem poder discriminante para internações significamente superior as demais.

Tabela 5.1: Média de AUC, Sensibilidade, Especificidade, F1-Score e desvio padrão dos classificadores treinados com diferentes conjuntos de dados e a combinação. Negrito mostra os melhores resultados.

	Métrica	GB	RF
Especialidade	F1-Score	66,8±0,9	**67,6±0,4**
	Sensibilidade	65,5±1,3	**68,6±1,1**
	Especificidade	**69,5±1,1**	65,5±1,4
	AUC	**75,4±0,6**	74,6±0,5
CID	F1-Score	62,1±0,7	**64,4±1,0**
	Sensibilidade	54,6±1,2	**60,5±1,8**
	Especificidade	**78,7±2,1**	72,5±1,5
	AUC	**72,8±0,8**	70,2±0,4
Descrição de eventos	F1-Score	73,2±0,5	**73,8±0,9**
	Sensibilidade	72,3±0,8	**76,3±1,2**
	Especificidade	**74,6±1,3**	69,5±1,1
	AUC	**81,3±0,7**	80,7±0,7
Combinação	F1-Score	**73,9±0,4**	72,1±0,8
	Sensibilidade	**72,5±1,2**	72,4±1,7
	Especificidade	**76,4±1,7**	71,7±1,6
	AUC	**82,0±0,6**	80,5±0,5

Embora alguns modelos obtenham resultados semelhantes em algumas métricas para determinados conjuntos de dados, algumas diferenças podem ser notadas por meio da curva ROC. As figuras 5.2 e 5.3 apresentam o comportamento dos modelos por meio da curva ROC para cada conjunto de dados. Observa-se que, apesar da AUC dos modelos gerados a partir de "Especialidade" e "CID" atingirem valores próximos, o comportamento dos modelos é diferente.

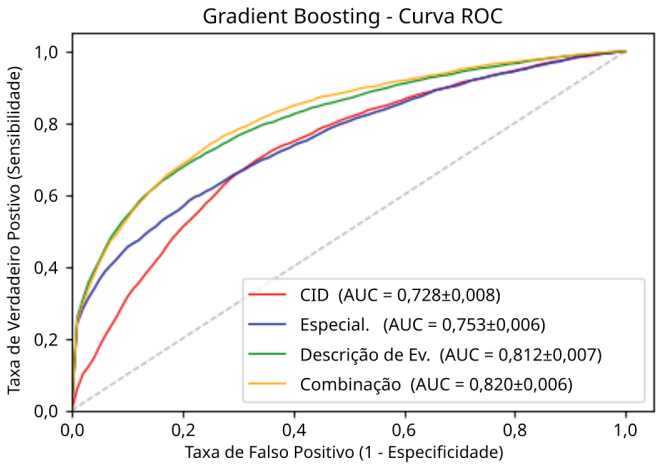


Figura 5.2: Comportamento dos modelos gerados a partir do GB.

Uma análise importante a ser realizada com esses modelos é a de sensibilidade e especificidade. Sensibilidade é a capacidade do modelo de discriminar entre os suspeitos de internação e os que serão internados. Por outro lado, a especificidade representa a capacidade de o modelo ser negativo entre aqueles que não serão hospitalizados.

No entanto, a equação de especificidade (Eq. 4.2) mostra que quanto menor a taxa de falsos positivos, maior o valor de especificidade. Portanto, a especificidade também é uma

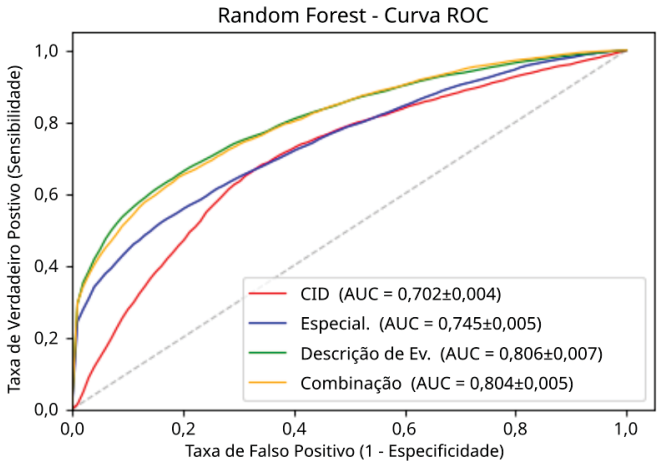


Figura 5.3: Comportamento de modelos gerados a partir de RF.

forma de analisar a taxa de falsos positivos do modelo. Assim, quanto maior a especificidade, menor a incerteza quanto à eficácia da hospitalização que o modelo infere. Portanto, para análise, assume-se que:

- Alta sensibilidade: baixa incerteza quanto às inferências de não internação;
- Alta especificidade: baixa incerteza quanto às inferências das internações.

Assim, ao observar a sensibilidade e a especificidade dos três conjuntos de dados, chegou-se a uma conclusão interessante em relação aos CIDs históricos do conjunto de dados: em geral, os modelos gerados a partir dele atingem valores mais modestos de sensibilidade, mas valores elevados de especificidade, significando que as internações inferidas pelo modelo são tão confiáveis quanto os modelos gerados com o conjunto de dados "Descrição do Evento" ou com a combinação de modelos. Essa constatação pode ser útil para quando apenas CIDs estão disponíveis, podendo-se utilizá-las para geração de modelos com esse objetivo. Isso facilita a escolha de modelos para diferentes tarefas. Por exemplo, explicar uma previsão de hospitalização a partir de CIDs pode ser mais fácil do que explicar a partir das descrições de eventos. Para demonstrar as diferenças entre sensibilidade e especificidade, a Figura 5.4 apresenta um gráfico com essas duas métricas para os diferentes modelos gerados por meio do GB.

5.3 RESULTADOS POR MEIO DE LLMS

Apresenta-se nesta seção os resultados alcançados por meio das várias abordagens testadas com LLMs. Os resultados derivam da aplicação sobre a base DB2. Nestes experimentos, utilizou-se somente as descrições dos eventos, pois como se utiliza modelos pré-treinados, os tokens de uma sequência não seriam reconhecidos, dado que são códigos. Além disso, o número de CIDs informados nos dados são mais raros do que na base DB1.

5.3.1 RoBERTa

A Tabela 5.2 apresenta os resultados dos testes com o modelo RoBERTa-MLM-FT, usando o *Random Forest* (RF) e também com a camada completamente conectada (FC - *Fully Connected Layer*). Além disso, apresenta-se a combinação (CB) dos dois métodos, utilizando a média das probabilidades como método de fusão, conforme a Figura 4.22.

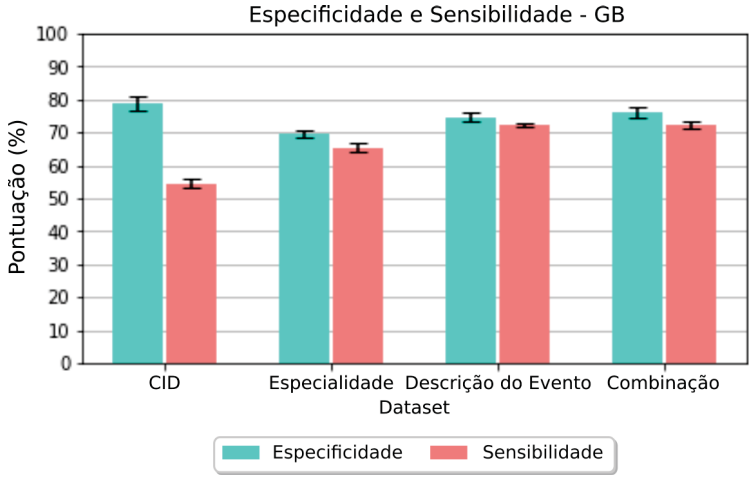


Figura 5.4: Média e desvio padrão de Sensibilidade e Especificidade com GB para os diferentes conjuntos de dados testados e combinação de classificadores.

Tabela 5.2: Média da AUC, Sensibilidade, Especificidade, F1-Score e desvio padrão dos classificadores treinados do zero a partir da estrutura do RoBERTa. Em negrito mostra-se os melhores resultados.

Modelo	Métrica	FC	RF	CB
RoBERTa-MLM-FT	F1-Score	86,4±0,5	86,5±0,6	**87,1±0,4**
	Sensibilidade	84,5±0,7	**89,8±0,6**	87,7±0,4
	Especificidade	**89,0±0,8**	82,9±1,3	86,7±0,9
	AUC	94,8±0,4	94,2±0,3	**95,0±0,3**

Embora tanto o *F1-Score* quanto o AUC alcancem resultados próximos, é possível observar que ocorre praticamente a inversão de valores entre sensibilidade e especificidade para o RF e FC. O RF atinge valores mais altos para sensibilidade enquanto o FC para especificidade. Além disso, a combinação entre os dois métodos equilibra os valores dessas duas métricas e melhora o F1-Score e o AUC.

5.3.2 BERTimbau

A Tabela 5.3 apresenta os resultados dos testes com o RF e com o FC para dois modelos pré-treinados, BERTimbau-FT e BERTimbau-MLM-FT. Este experimento buscou verificar também os ganhos do treinamento MLM sobre um modelo já pré-treinado. Também apresenta-se a combinação dos dois métodos, utilizando a média das probabilidades como método de fusão, conforme a Figura 4.22.

Estes experimentos também revelaram a quase inversão dos valores de sensibilidade e especificidade para o RF e a FC. Entre os dois modelos testados, o BERTimbau-MLM-FT alcança resultados ligeiramente melhores, indicando que o treinamento MLM contribuiu para o melhoramento dos resultados finais.

Comparando com os resultados do RoBERTa-MLM-FT apresentados na seção 5.3.1, observa-se que o RF alcançou resultados melhores para a métrica de sensibilidade, que está fortemente relacionada aos acertos das previsões de internações. Isto demonstra que essa abordagem é eficiente dependendo do objetivo de aplicação do modelo. Assim, considerando que o custo computacional e o tempo de inferência através do RF é menor do que diretamente de uma LLM com a camada completamente conectada, pode-se, então, gerar uma base completa

Tabela 5.3: Média da AUC, Sensibilidade, Especificidade, F1-Score e desvio padrão dos classificadores treinados com BERTimbau. Em negrito mostra-se os melhores resultados.

Modelo	Métrica	FC	RF	CB
BERTimbau-FT	F1-Score	85,2±0,6	85,6±0,6	86,0±0,6
	Sensibilidade	81,8±0,9	87,9±0,7	85,0±0,8
	Especificidade	89,6±1,0	83,2±0,9	87,6±0,9
	AUC	94,1±0,4	93,9±0,5	94,3±0,4
BERTimbau-MLM-FT	F1-Score	85,7±0,5	86,2±0,9	**86,4±0,6**
	Sensibilidade	82,4±0,7	**89,1±0,8**	86,2±0,8
	Especificidade	**90,0±0,8**	83,1±1,2	86,9±0,7
	AUC	94,4±0,4	94,2±0,4	**94,7±0,4**

com *embeddings* das sentenças em servidores mais potentes, para posterior uso em modelos RF em dispositivos mais modestos.

5.3.3 Open-Cabrita3B

A Tabela 5.4 apresenta os resultados dos experimentos com o Open-Cabrita3B-FT usando a FC como método de predição.

Tabela 5.4: Média da AUC, Sensibilidade, Especificidade, F1-Score e desvio padrão do classificador treinados com OpenCabrita3B.

Modelo	Métrica	FC
OpenCabrita3B-FT	F1-Score	87,8±0,7
	Sensibilidade	92,4±0,4
	Especificidade	82,6±1,3
	AUC	95,4±0,3

O Open-Cabrita3B-FT alcançou resultados similares para *F1-Score* e AUC se comparado aos demais modelos testados neste trabalho, alcançando maior resultado para sensibilidade. Esse resultado sugere que a inclusão de sentenças mais longas podem causar um impacto positivo na previsão de internações, o que se torna mais evidente a partir dos resultados dos experimentos descritos na seção 5.6, ou seja, quando os experimentos são realizados para um problema específico com diferentes períodos de antecedência.

5.3.4 Combinação

A Tabela 5.5 apresenta os resultados da combinação dos três modelos: RoBERTa-MLM-FT, BERTimbau-MLM-FT e OpenCabrita3B-FT com a camada completamente conectada. O método de fusão utilizado nesta combinação também foi a média das probabilidades, e segue o esquema apresentado na Figura 4.23

Os resultados da combinação demonstram um modelo mais balanceado, evidenciado pela proximidade entre os valores de sensibilidade e especificidade. Além disso, esta combinação permitiu alcançar os melhores resultados entre todos os modelos generalistas.

Tabela 5.5: Média da AUC, Sensibilidade, Especificidade, Pontuação F1 e desvio padrão da Combinação.

Modelo	Métrica	FC
Combinação	F1-Score	87,5±0,8
	Sensibilidade	87,8±0,9
	Especificidade	87,5±0,9
	AUC	95,5±0,3

5.4 COMPARAÇÃO ENTRE LLMS E O MÉTODO CONVENCIONAL

Nesta seção apresenta-se um comparativo entre a abordagem convencional e a abordagem por meio de LLMs (Tabela 5.6). Para a comparação, selecionou-se a técnica com melhores resultados de cada abordagem. No caso do método convencional, selecionou-se a combinação do *Gradient Boosting*. Para o método com LLM, selecionou-se a combinação do RoBERTa-MLM-FT, BERTimbau-MLM-FT e Open-Cabrita-FT.

Tabela 5.6: Resultados dos melhores modelos de ambas abordagens. Em negrito apresenta-se os melhores resultados.

Métrica	CB (Método convencional)	CB (LLM com FC)	Ganho
F1-Score	73,9±0,4	**87,5±0,8**	+18,4%
Sensibilidade	72,5±1,2	**87,8±0,9**	+21,1%
Especificidade	76,4±1,7	**87,5±0,9**	+14,5%
AUC	82,0±0,6	**95,5±0,3**	+16,4%

Considerando o resultado apresentado na Tabela 5.6, é significativo o ganho de performance em relação ao método convencional, demonstrando a eficácia da utilização de LLMs para previsões de internações por meio de dados de planos de saúde.

5.5 EXPERIMENTOS PARA INTERNAÇÕES POR PROBLEMA ESPECÍFICO

Para avaliar a capacidade dos modelos para previsão de internações para um problema específico, examinou-se os conjuntos de dados *DB2-dataset II e III*, que consistem em dados não utilizados no treinamento dos modelos, para identificar casos de internações por AVC (correspondente a CID I64). Esse tipo de internação foi escolhido pois desempenha um papel crítico na prestação de cuidados oportunos e eficazes aos indivíduos que sofrem AVC, com o objetivo de minimizar danos cerebrais, maximizar a recuperação e prevenir futuros AVCs. Desses dados, foram selecionados 134 beneficiários que sofreram internação por AVC e que continham registros com datas de, no mínimo, 360 dias anteriores à internação. A partir desses beneficiários, foram construídos os históricos por descrição de eventos, gerando 200 exemplos de internações por AVC para um dia de antecedência. A diferença entre o número de exemplos e beneficiários mostra que há casos em que o beneficiário sofreu reinternação pelo mesmo problema. Além disso, sobre os mesmos dados de teste foram selecionados aleatoriamente 200 exemplos de históricos de casos de não internação para compor o conjunto de teste, resultando assim em 400 exemplos de teste. Os testes foram realizados sobre os LLMs treinados anteriormente, sem a realização de *fine-tune* específico para AVC, ou seja, os modelos generalistas foram utilizados. Destaca-se aqui, por meio da Figura 5.5, a distribuição dos exemplos históricos, considerando o número de *tokens* gerados pelo tokenizador do BERTimbau, em que é possível observar que

mais de 50% dos exemplos possuem mais de 3.214 *tokens*. Portanto, um número significativo de exemplos supera o número máximo de *tokens* do maior modelo testado, o *Open-Cabrita3B-FT*.

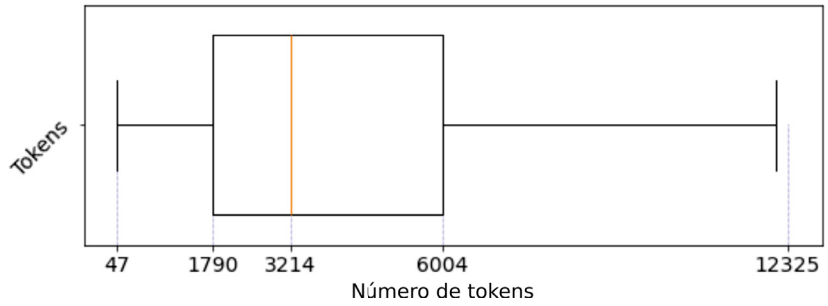


Figura 5.5: Boxplot para número de tokens das sentenças históricas de internação por AVC.

Esses dados foram então aplicados sobre os três modelos treinados com a camada completamente conectada. Foi realizada também a combinação desses três classificadores, utilizando a média das probabilidades como método de fusão, como apresentado na Figura 4.23. A Tabela 5.7 apresenta os resultados alcançados.

Tabela 5.7: AUC, Sensibilidade, Especificidade, F1-Score do teste com internações por AVC. Em negrito apresenta-se os melhores resultados.

Modelo	Métrica	FC
OpenCabrita3B-FT	F1-Score	84,8
	Sensibilidade	**93,5**
	Especificidade	76,5
	AUC	96,4
RoBERTa-MLM-FT	F1-Score	88,4
	Sensibilidade	89,0
	Especificidade	88,0
	AUC	96,4
BERTimbau-MLM-FT	F1-Score	**88,7**
	Sensibilidade	88,5
	Especificidade	**89,0**
	AUC	95,6
Combinação	F1-Score	**88,7**
	Sensibilidade	90,5
	Especificidade	87,0
	AUC	**96,5**

A Figura 5.6 apresenta as matrizes de confusão dos experimentos para cada modelo. A Figura 5.7 apresenta a curva ROC dos resultados para os classificadores testados.

A mesma tendência entre especificidade e sensibilidade, como verificada na seção 5.3, se repete para as internações por AVC, ou seja, o Open-Cabrita3B-FT, se comparado aos demais modelos, atinge valores maiores para sensibilidade e menores para especificidade. A combinação dos classificadores alcançou F1-Score de 88,7%. Dados os resultados alcançados nesta seção, pode-se concluir que os modelos treinados com propósito generalista também funcionam para a previsão de internações para problemas específicos.

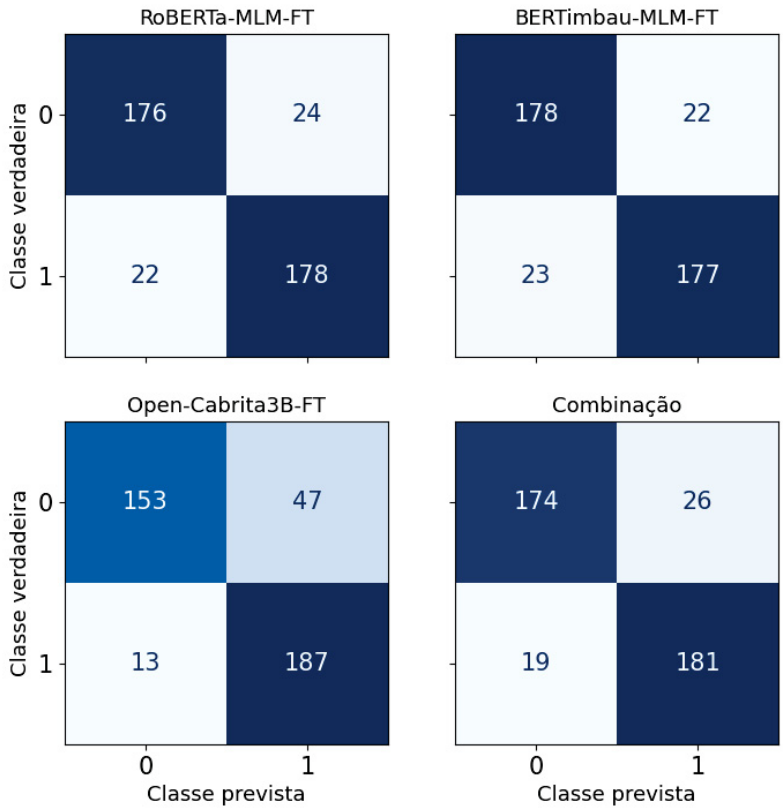


Figura 5.6: Matrizes de confusão para beneficiários com AVC por modelo testado.

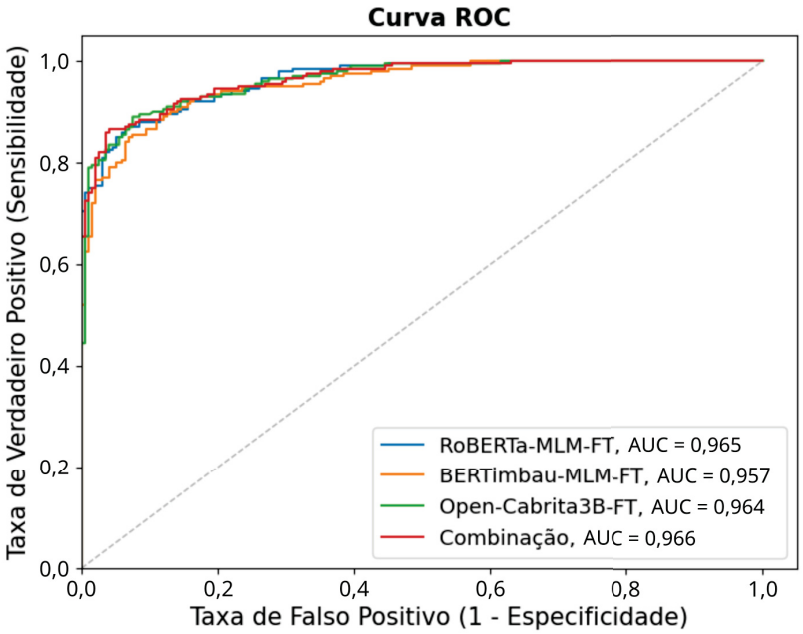


Figura 5.7: Curva ROC da combinação de classificadores para AVC.

5.6 EXPERIMENTOS PARA DIFERENTES PERÍODOS DE ANTECEDÊNCIA

A fim de avaliar a capacidade dos modelos de prever internações com antecedência e, considerando os resultados dos experimentos de internações por AVC, utilizou-se os mesmos dados testados, apresentados na seção 5.5, mas com os históricos gerados para diferentes períodos de antecedência: 5, 15, 30 e 60 dias. Além disso, prever internações por AVC com maior antecedência, mesmo que poucos casos, pode ajudar em várias aplicações, como ações preventivas por meio de alteração de tratamentos ou direcionando maior atenção ao paciente. Desta forma, buscando analisar o comportamento do modelo para estes períodos de antecedência, a Figura 5.8 apresenta os resultados para os três modelos testados neste trabalho e também a combinação dos classificadores, proposta na Figura 4.23.

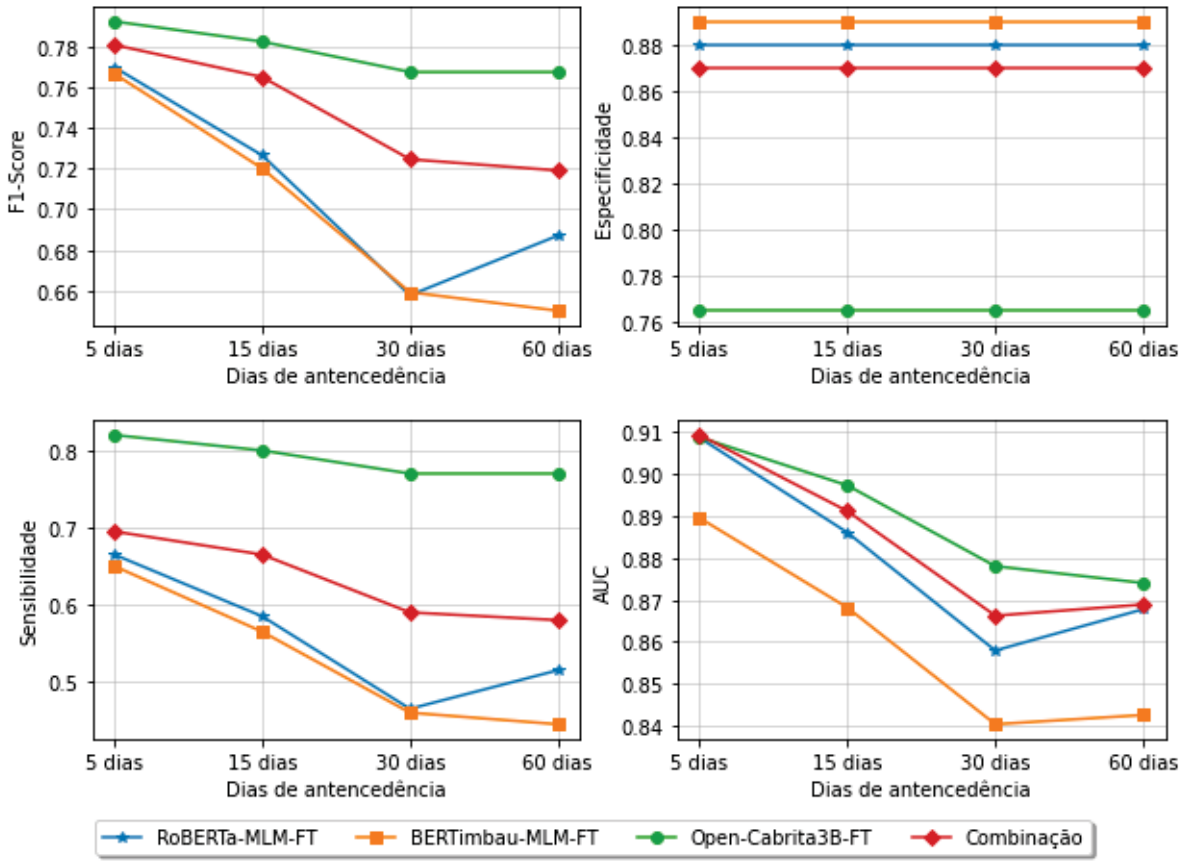


Figura 5.8: Previsões de internações por AVC para diferentes períodos de antecedência.

Considerando os resultados apresentados na Figura 5.8, fica clara a vantagem do Open-Cabrita3B-FT em relação aos outros modelos. Ele alcança AUC inicial muito similar aos demais para poucos dias de antecedência, entretanto, se destaca dos demais conforme os dias de antecedência aumentam. Ele é bom justamente nos acertos dos casos de internações, tendo em vista os valores alcançados pela sensibilidade deste modelo. Considerando que o Open-Cabrita3B-FT é um modelo maior que os demais, e trabalha com 2048 *tokens*, os resultados alcançados podem indicar a importância do tamanho das sentenças na previsão das internações.

5.7 COMPARAÇÃO COM LLM TREINADA SOMENTE COM DADOS DE INTERNAÇÕES POR AVC

Os resultados apresentados pela Figura 5.8 demonstram a capacidade dos modelos generalistas para previsão de internação para um problema específico. Entretanto, surge uma questão: Se a LLM for treinada somente com sentenças históricas de internações por AVC, os resultados não seriam melhores ou ao menos iguais? Buscando responder a essa pergunta, realizou-se o *fine-tuning* do Open-Cabrita3B apenas com dados de internações por AVC. O Open-Cabrita3B foi selecionado pois alcançou os melhores resultados dos experimentos.

Os dados de treinamento foram selecionados do *dataset I* da base DB2, e resultaram em 7808 exemplos, sendo 3904 casos de internação por AVC e 3904 casos de não internação selecionados aleatoriamente do restante dos dados. As configurações de treinamento foram as mesmas apresentadas na subseção 4.6.3, exceto pelos dados de treinamento. Para o teste utilizou-se o dados de internações por AVC com 60 dias de antecedência. A Tabela 5.8 apresenta os resultados para este teste.

Tabela 5.8: AUC, Sensibilidade, Especificidade, F1-Score do teste com internações por AVC com 60 dias de antecedência, do modelo Open-Cabrita3B treinado apenas com internações e com todos os dados. Em negrito apresenta-se os melhores resultados.

Modelo	Métrica	FC
OpenCabrita3B apenas com dados de AVC	F1-Score	70,7
	Sensibilidade	56,0
	Especificidade	**87,0**
	AUC	81,2
OpenCabrita3B com todos os dados	F1-Score	**76,7**
	Sensibilidade	**77,0**
	Especificidade	76,5
	AUC	**87,4**

Os resultados da Tabela 5.8 indicam que os modelos generalistas possuem uma maior capacidade de inferência de internações se treinados com todos os dados, independentemente do tipo de internação. É provável que a maior quantidade de dados de treinamento permita ao modelo encontrar relações nas sentenças que os poucos dados de uma internação específica não permitem encontrar.

5.8 COMPARAÇÃO COM OUTROS TRABALHOS DA LITERATURA

Os trabalhos utilizados para comparação foram (Monsen et al., 2012) que propuseram uma medida para predizer o risco de internações entre pacientes em cuidados domiciliares, (Lorenzoni et al., 2019) que apresentaram um estudo comparativo entre a performance de modelos gerados por oito técnicas de Aprendizado de Máquina diferentes, cujo objetivo foi predição de internação de pacientes com Insuficiência Cardíaca, (Barak-Corren et al., 2017) que desenvolveram um modelo de previsão de internação com o objetivo de reduzir o tempo de espera do paciente no pronto-socorro até o momento da internação, e (Patel et al., 2018) que testam quatro algoritmos de aprendizado de máquina, *Decision tree*, *Random Forest*, *Logistic LASSO Regression* e *Gradient Boosting* (GB) para realizar a previsão da necessidade ou não de internações pediátricas relacionadas a asma. Esses trabalhos abordam internações com horizonte

temporal de um dia ou menos de antecedência, o mesmo utilizado em parte significativa dos experimentos deste trabalho. Salienta-se que esses trabalhos se referem a problemas de saúde específicos, não testados nesse trabalho. A Tabela 5.9 apresenta os resultados dos trabalhos e do melhor resultado alcançado nesta pesquisa.

Tabela 5.9: Comparação com outros trabalhos para um dia ou menos.

Referência	**AUC**
(Monsen et al., 2012)	0,63
(Lorenzoni et al., 2019)	0,80
(Barak-Corren et al., 2017)	0,91
(Patel et al., 2018)	0,84
Combinação LLMs	0,955

Pode-se observar por meio da Tabela 5.9 que o método proposto supera os demais, mesmo se tratando de um modelo generalista.

5.9 EXPLICABILIDADE DOS MODELOS

Dado o potencial de pesquisa em outras áreas do conhecimento a partir da análise e das previsões realizados dos modelos gerados neste trabalho, como a análise de fatores que levam a determinados tipos de internações ou complicações, torna-se pertinente a utilização de métodos que permitam a interpretabilidade dos resultados. Além disso, dependendo da aplicação dos modelos, as predições podem afetar decisões sensíveis, o que reforça a necessidade de confiança e entendimento do resultado. Assim, considerando que este não é o foco principal desta tese, mas dada a importância da interpretabilidade dos resultados das inferências, apresenta-se a aplicação do método de explicação SHAP (*SHapley Additive exPlanations*) que tem obtido bastante destaque na área de *Explainable AI* (Linardatos et al., 2020) sobre o modelo RoBERTa-MLM-FT.

O modelo RoBERTa-MLM-FT foi escolhido devido à velocidade de inferência, o que afeta o tempo de treinamento do *explainer* do SHAP. O treinamento levou 2 horas e 18 minutos e foi realizado sobre 2379 exemplos balanceados, dos quais 200 são os de internação por AVC com 60 dias de antecedência, os mesmos utilizados nos experimentos apresentados na seção 5.5.

Para demonstrar a explicabilidade sobre uma previsão, foi selecionado um dos exemplos de internação por AVC pertencentes ao conjunto de treinamento do *explainer*. Este exemplo pode ser visto no apêndice A.2. A Figura 5.9 apresenta um gráfico de texto gerado pelo SHAP que permite visualizar a contribuição das características de uma única instância sobre a inferência positiva, neste caso da internação. Os valores SHAP usados por esse gráfico explicam de forma aditiva como o impacto do desmascaramento de cada palavra altera a saída do modelo do valor base (*base value*), calculado quando toda a entrada é mascarada, para o valor de predição final. Além disso, foi gerado também o gráfico chamado *waterfall* (cascata), Figura 5.10, que fornece a contribuição de cada característica para a saída do modelo para uma previsão específica. A parte inferior do gráfico começa com o valor esperado do modelo (*E[f(X)]*) que neste gráfico é o valor médio de saída do modelo para todos os exemplos usados no teste. Em seguida, cada linha mostra como cada característica contribuí positivamente (vermelho), ou negativamente (azul) para se chegar ao valor de saída do modelo (*f(x)*) para o exemplo específico testado. A Figura 5.10 apresenta as 14 primeiras características em um gráfico *waterfall* aplicado sobre a mesma sentença, observada na Figura 5.9. Assim, o gráfico *waterfall* é usado aqui para demonstrar a importância dos tokens na previsão de uma única internação.

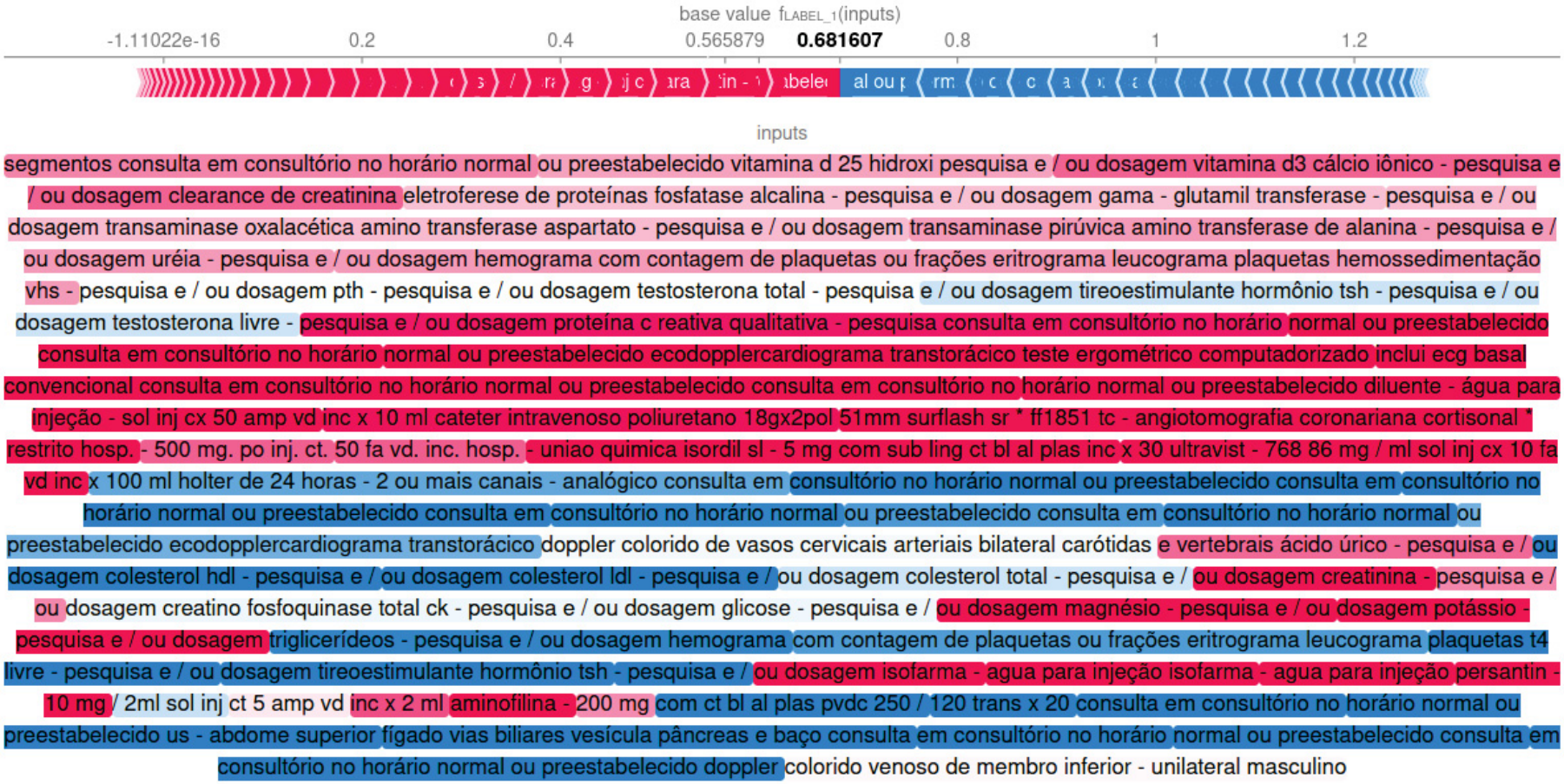


Figura 5.9: Contribuição das características sobre a previsão positiva para internação.

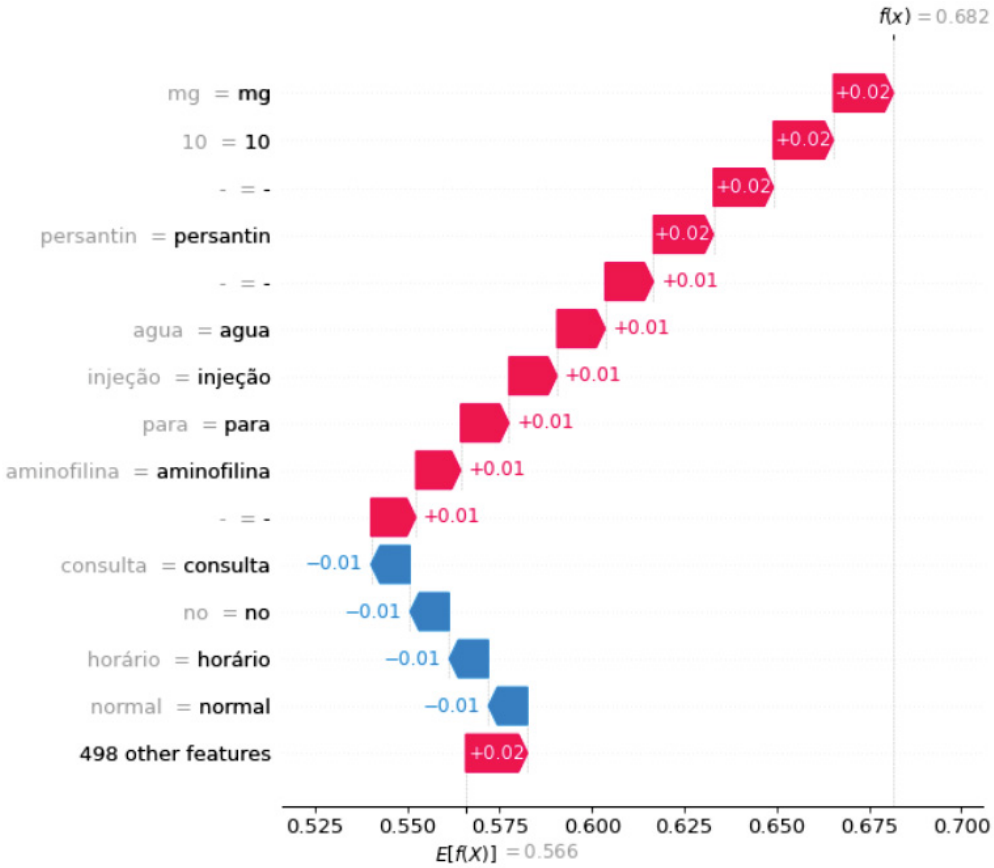


Figura 5.10: Contribuição dos 14 primeiras características para inferência da sentença.

Por meio do gráfico da Figura 5.9 pode-se observar os termos e trechos da sentença que mais contribuíram para a previsão da internação segundo o SHAP. O tamanho significativo da sentença dificulta a interpretação, mas para um especialista pode ser muito útil. Uma forma de ajudar a direcionar o olhar é observando o gráfico da Figura 5.10. Nele é possível observar que os termos "persantin"e "aminofilina"contribuem positivamente para inferência da internação. O "persantin"é indicado como auxiliar em testes diagnósticos ergométricos e na ecocardiografia, o que pode indicar, juntamente com outros termos da sentença, que o modelo identificou que o paciente está com algum problema cardíaco. Além disso, o momento em que ocorre na sentença pode talvez indicar que complicações na saúde do paciente estão ocorrendo. Já o "aminofilina"é indicada para doenças caracterizadas por broncoespasmo, como a asma brônquica ou o broncoespasmo associado com bronquite crônica e enfisema. Vale destacar que o caso analisado é de um paciente que sofreu internação por AVC, caberia então a um especialista analisar se os fatores destacados pelo *explainer* são conhecidos pela medicina como complicadores para ocorrência de AVC ou se há algumas relações não conhecidas que mereçam investigações mais profundas.

Além disso, a Figura 5.11 apresenta um gráfico de barras gerado pelo SHAP que busca apresentar o impacto médio de todas as palavras para previsão da classe de internação. Neste caso a média é obtida a partir dos exemplos utilizados no treinamento do *explainer*. Este gráfico tem um propósito de apresentar uma visão geral do modelo. Neste caso o gráfico apresenta as 14 primeiras características que o *explainer* considerou mais importante. Observa-se que ao todo o *explainer* considerou 5026 características, entretanto, o modelo possui entrada para 30.000 tokens, o que demonstra que o *explainer* não conseguiu analisar a totalidade das características, provavelmente, ele esta abrangendo apenas os tokens presentes nos dados de treinamento. Além

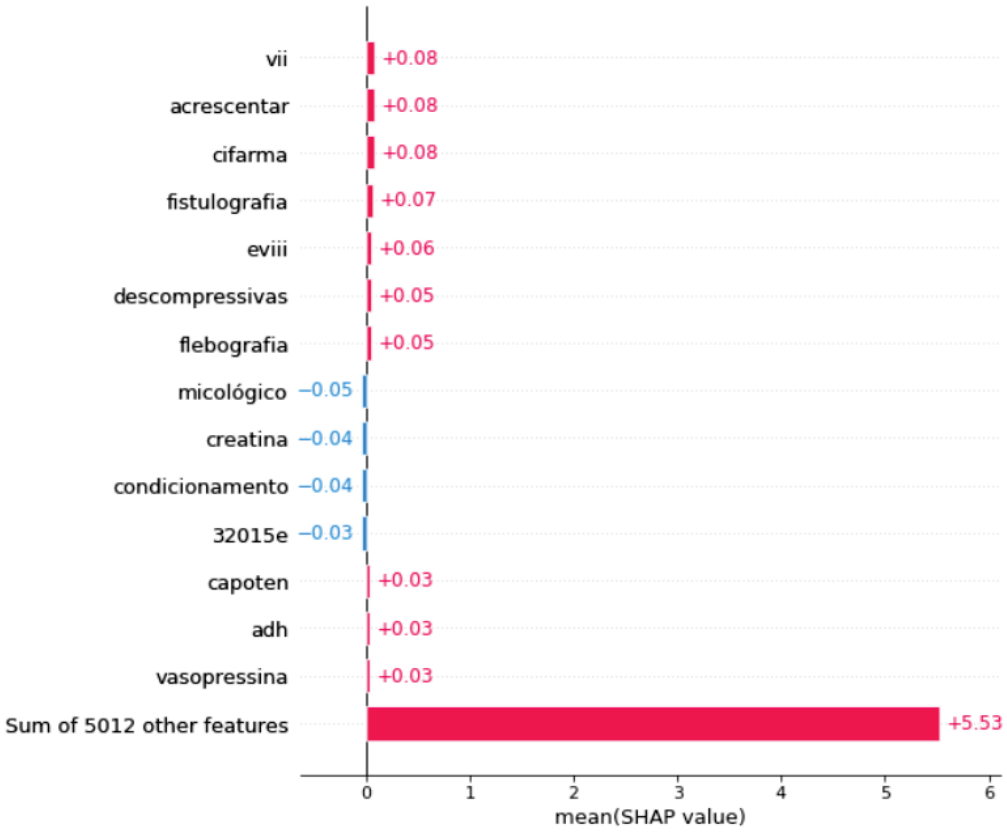


Figura 5.11: Impacto médio das características do modelo para previsão da classe positiva (internação).

disso, esse gráfico torna difícil a interpretação geral para modelos grandes, com muitos parâmetros de entrada, como os utilizados nesta investigação. Além disso, para se ter uma interpretação mais fiel do modelo seria necessário treinar o *explainer* com parte significativa dos dados, o que pode se tornar inviável dado o tempo gasto no treinamento de apenas 2379 exemplos.

Portanto, considerando os resultados alcançados, pode-se concluir que o SHAP pode ser usado como ferramenta de investigação sobre os resultados das inferências dos modelos propostos nesta tese, principalmente para previsões individuais. Entretanto, para interpretação global do modelo, para os modelos adotados, as possibilidades do *framework* são limitadas, dada a grande quantidade de entradas possíveis desses modelos e o tempo necessário para treinamento do *explainer* para uma quantidade significativa e representativa dos dados.

5.10 DISPONIBILIZAÇÃO DOS MODELOS

Ainda que os resultados obtidos pelo Open-Cabrita3B-FT sejam melhores do que os dos demais modelos, estes não são desprezíveis. Como o Open-Cabrita3B-FT exige maior poder de hardware para treinamento e inferência, os demais modelos apresentam resultados significativos, exigindo menor poder de processamento. Além disso, RoBERTa-MLM-FT é fortemente relacionado ao domínio dos dados de planos de saúde, possuindo um tokenizador próprio para esse vocabulário. Esse modelo e os demais podem ser potencialmente eficazes em outras aplicações, cujos dados apresentem a mesma estrutura, tais como: previsões de custos, previsão de doenças e complicações de saúde, análise de fatores relacionados a complicações de saúde, análise de tratamentos, estimativa de dias de internação, etc.

Assim, dado o potencial de uso desses modelos, eles foram disponibilizados para a comunidade científica. Além deles, foram treinados vários outros modelos em inglês, a partir da

tradução dos dados de planos de saúde utilizados neste trabalho. Para a tradução dos dados foi utilizado o modelo *translation-pt-en-t5* descrito no trabalho de Lopes et al. (Lopes et al., 2020). Os modelos em inglês disponibilizados são:

- **RoBERTa-MLM-EN** - Modelo treinado da mesma forma que o RoBERTa-MLM, mas com dados em inglês.
- **BERT + MLM** - Modelo treinado a partir do BERT-base (Devlin et al., 2018) com dados em inglês.
- **BioBERT + MLM** - Modelo treinado patir do BioBERT (Lee et al., 2020) com dados em inglês.
- **Bio_ClinicalBERT + MLM** - Modelo treinado a patir do Bio_ClinicalBERT (Alsentzer et al., 2019) com dados em inglês.
- **OpenLLaMA3Bv2 + Fine-tuning** - Modelo ajustado a partir do OpenLLaMA3Bv2 (Touvron et al., 2023a) com dados em inglês.
- **OpenLLaMA7Bv2 + Fine-tuning** - Modelo ajustado a partir do OpenLLaMA7Bv2 (Touvron et al., 2023a) com dados em inglês.
- **OpenLLaMA13B + Fine-tuning** - Modelo ajustado a partir do OpenLLaMA13B (Touvron et al., 2023a) com dados em inglês.

Os links para download dos modelos, detalhes dos treinamentos e resultados alcançados pelos modelos em inglês e português estão disponíveis no apêndice C.

5.11 LIMITAÇÕES

Os modelos treinados a partir do zero, RoBERTa-MLM e RoBERTa-MLM-EN, com dados exclusivamente de históricos de beneficiários de planos de saúde, não foram testados com sentenças de texto tradicionais. Sendo assim, não se espera que esses modelos tenham bons resultados para problemas que utilizem dados desse tipo.

5.12 DISCUSSÃO

Os resultados apresentados neste capítulo demonstram a capacidade do método proposto para prever internações. A abordagem adotada de transformação dos dados estruturados dos planos de saúde em sentenças históricas dos beneficiários permitiu a utilização de métodos de PLN para fazer a extração de características de maneira automatizada, dispensando conhecimento prévio sobre problemas de saúde e internações na etapa de pré-processamento dos dados.

Os resultados alcançados por meio dos métodos convencionais, mesmo aplicados sobre uma base de dados consideravelmente menor, demonstraram-se interessantes, pois conseguiram superar alguns trabalhos da literatura que produziram modelos para o mesmo horizonte temporal, apesar de serem aplicados para problemas específicos. Assim, a partir desses resultados, abriu-se caminhos para exploração dos dados por meio das LLMs. Além disso, ficou claro nesses resultados que a descrição de eventos foi a característica entre as testadas que apresentou maior capacidade discriminante. Desta forma, a partir da obtenção de novos dados, e em maior volume, e utilizando a descrição de eventos, foi possível treinar LLMs, gerando modelos deste tipo fortemente relacionados ao domínio no qual os dados de planos de saúde se aplicam.

Considerando os experimentos com as LLMs, o ganho de performance em relação ao método convencional foi bem significativo, chegando a 18,4% para métrica *F1-Score* para o melhor modelo generalista. Se comparado com os demais da literatura apresentados nesta tese, supera todos com o mesmo horizonte temporal, mesmo se tratando de um modelo generalista.

Os experimentos para internações por AVC evidenciaram a capacidade dos modelos generalistas de prever internações para problemas específicos, superando LLMs treinadas exclusivamente com dados de internações por AVC. Esses resultados e dados utilizados, abre um grande escopo para investigação de internações para outros casos específicos. Além disso, os resultados para diferentes períodos de antecedência demonstram mais uma capacidade dos modelos gerados, ampliando o escopo de aplicações desses modelos.

Outro aspecto investigado nos experimentos foi quanto à influência do tamanho das sentenças e das LLMs sobre os resultados das previsões. Observou-se que os modelos maiores alcançam resultados melhores. No caso do Open-Cabrita3B, este aspecto tornou-se mais evidente quando se ampliou o horizonte temporal das previsões. Considerando os resultados destes experimentos, parece promissor treinar modelos desse tipo inteiramente com dados de planos de saúde para uma investigação. Embora, o treinamento de modelos tão grandes exijam recursos de *hardware* extremamente caros, poderia gerar modelos potencialmente relevantes para comunidade científica, e talvez alcançar melhores resultados.

Os dados da base DB1 foram disponibilizados para comunidade científica. A base DB2 ainda precisa de autorizações para a disponibilização. Além das bases, os modelos de LLMs gerados têm grande potencial de aplicação em outros problemas além da previsão de internações, e foram também disponibilizados. Entre as possíveis aplicações destes modelos, destacam-se:

- Análise de tratamentos e complicação de saúde;
- Identificação de condições sensíveis à atenção primária (ICSAP);
- Prevenção de desperdício de recursos;
- Utilização desses modelos em complementariedade com outros métodos de previsão.

5.13 CONSIDERAÇÕES FINAIS

Neste capítulo, foram apresentados e discutidos os experimentos e os resultados obtidos pelo método proposto. Os resultados alcançados pelos diversos modelos gerados evidenciam a viabilidade e a eficácia da abordagem adotada, demonstrando que a utilização de dados históricos de pacientes, extraídos de planos de saúde, pode ser empregada no treinamento de modelos para previsão de internações, independentemente do problema de saúde associado. Dessa forma, a hipótese desta tese é confirmada. Ademais, os resultados revelam a capacidade dos modelos de prever internações em casos específicos, apresentando desempenhos superiores em comparação aos modelos treinados apenas com exemplos desses casos. Foram também desenvolvidos diversos modelos de LLMs, que foram disponibilizados para a comunidade científica, possuindo grande potencial para aplicação em outros tipos de problemas da área da saúde e para a investigação de diferentes categorias de internações.

No próximo capítulo é apresentada a conclusão e também possibilidades de trabalhos futuros.

6 CONCLUSÃO

A previsão de internações hospitalares pode ser considerada um objetivo importante para médicos, gestores hospitalares e planos de saúde. A previsão de internações pode ajudar no planejamento e organização de hospitais assim como de planos de saúde. Além disso, fornece suporte informacional a médicos para realização de ações preventivas e também na agilidade do atendimento em situações de emergência. Portanto, desenvolver métodos que permitam realizar tais previsões torna-se pertinente. Diversos trabalhos encontrados na literatura demonstram esforços nesse sentido, entretanto, a dificuldade de se obter dados, e os processos de seleção e extração de características encontrados na literatura, são muitas vezes custosos e orientados a determinados problemas de saúde, o que requer conhecimento especializado, e, muitas vezes, não colaboram para a obtenção de bons modelos de previsão.

Assim, obter acesso a dados que tenham poder discriminante em relação a internações é um desafio. Os planos de saúde no Brasil acumulam dados referentes aos procedimentos, diagnósticos e contatos de seus beneficiários com o sistema de saúde, e esses tipos de dados podem representar, em diversos casos, as verdadeiras condições de saúde de um beneficiário. Esta pesquisa conseguiu acesso a dados desse tipo em grande quantidade e demonstrou por meio dos resultados alcançados a viabilidade na geração de modelos para previsão de internações tanto generalistas como para problemas específicos. Além disso, a forma como os dados foram organizados, tornou possível utilizá-los com diferentes métodos de extração e seleção de características, permitindo a escolha dos melhores para o treinamento dos modelos de previsão. Parte dos dados utilizados foi disponibilizada para a comunidade científica, assim como as LLMs geradas, dado que possuem potencial de aplicação em outros problemas da área da saúde.

Considerando os objetivos desta pesquisa, desenvolveu-se um modelo de aprendizado de máquina para prever internações usando apenas dados de planos de saúde, confirmando a hipótese desta tese. Os modelos gerados obtiveram resultados superiores aos encontrados na literatura, considerando o horizonte temporal de um dia, além de bons resultados para horizontes temporais mais longos. Além disso, constatou-se a eficácia dos modelos quanto a sua capacidade de prever internações para problemas específicos, ampliando o potencial de uso dos modelos gerados

6.1 TRABALHOS FUTUROS

Nesta seção, são apresentados algumas direções potencialmente promissoras para trabalhos futuros, vislumbrados no decorrer do desenvolvimento desta tese:

- **Previsão do tempo de internação:** Ainda na etapa de preparação desta tese, percebeu-se que é possível determinar o número de dias em que o paciente permaneceu internado. Desta forma, é possível gerar modelos de regressão, cujo objetivo seja realizar esta estimativa. Além disso, é possível que os modelos de LLMs treinados nesta tese sejam utilizados nesse processo.

- **Previsão dos custo da internação:** Considerando as bases de planos de saúde utilizadas neste trabalho, sabe-se que cada evento da base está associado ao seu custo. Assim, pode-se calcular da mesma forma como se calcula os dias de internação de um paciente, o valor total gasto na internação. Também é possível gerar modelos de regressão para prever esse custo, sendo possível, neste caso utilizar LLMs pré-treinadas neste trabalho.

- **Previsão de complexidade da internação:** Assim como é possível prever o número de dias e custo de uma internação, pode-se prever seu nível de complexidade. Para isso, é necessário que um especialista da área de medicina avalie a complexidade dos eventos de internação para que uma estimativa seja calculada a partir deles. Essa estimativa pode então ser combinada com o número de dias e custo da internação a fim de gerar um índice que representaria sua complexidade. Desta forma, podem ser construídos modelos de regressão para prever esse índice. As LLMs treinadas nesta tese, também podem ser usadas nesse processo.

- **Analisar modelos por faixa etária:** Analisar os modelos por faixa etária pode ajudar a entender para quais tipos de beneficiários esses modelos podem ser melhor aplicados, o que minimizaria possíveis erros de aplicação dos modelos.

- **Avaliar modelos para diversos tipos de internações:** Os resultados dos modelos na previsão de internações por AVC demonstram o grande potencial na previsão de internações para outros problemas de saúde. Como foi apresentado nesta tese, a maior base de dados possui internações relacionadas a 6919 CIDs diferentes. Portanto, investigações direcionadas podem ser realizadas considerando diferentes objetivos e propósitos, como o uso desses modelos para identificação de ICSAP para internações relacionadas a um determinado tipo de doença.

- **Desenvolver e aplicar métodos de interpretabilidade das previsões:** Dependendo da aplicação, a interpretabilidade das previsões de internações pode ser muito importante. Embora tenha-se aplicado nesta tese o método de interpretabilidade SHAP é necessário uma investigação mais aprofundada sobre o tema. Encontrar o melhor método para os modelos propostos nesta tese é um grande desafio. Fatores como precisão, simplicidade das explicações, além da adequação à área da saúde, devem ser considerados. Um exemplo de aplicabilidade da interpretação dos resultados é a análise de fatores de riscos para internações.

- **Treinamento de LLMs maiores somente com dados de planos de saúde:** Os melhores resultados alcançados nesta tese foram atingidos com uma LLM que aceita sentenças de até 2048 tokens. Devido a limitações de *hardware*, não foi possível fazer o treinamento auto-supervisionado nessas LLMs. Assim, considerando os resultados alcançados, torna-se interessante fazer esse treinamento, buscando, assim, avaliar possíveis melhorias nos resultados para o escopo da saúde. Uma forma de fazer isso de maneira mais otimizada e com *hardware* menos potente (se comparado os utilizados hoje no estado da arte para área) seria tentar alterar a arquitetura de modelos, como o do RoBERTa, que já são otimizados, para trabalhar com sentenças maiores, para então compará-los com os modelos treinados por essa pesquisa.

- **Treinar e avaliar LLMs com dados da ANS:** Com base na lei de acesso à informação, a ANS disponibiliza parte significativa dos dados de todos os planos de saúde do Brasil. Analisando esses dados, que são anonimizados, chegou-se à conclusão de que não seria possível gerar histórico dos beneficiários por meio deles. Entretanto, notou-se que é possível chegar até o nível de agregação dos dados, que representa as descrições de eventos de um atendimento do beneficiário. Como esses dados abrangem todo o Brasil, é interessante fazer o pré-treinamento de LLMs a partir desses dados para avaliá-los na previsão de internações.

- **Plataforma para disponibilização de dados modelos e resultados:** Considerando-se as potencialidades de uso dos dados e dos modelos gerados nesta tese, pretende-se criar uma plataforma WEB para disponibilizar os dados, modelos e resultados, bem como apresentar para a comunidade científica os potenciais de pesquisa na área. Embora parte dos dados tenha sido disponibilizada, a maior parte ainda precisa de autorização para publicação.

REFERÊNCIAS

(2021). Portal brasileiro de dados abertos - procedimentos hospitalares por uf. `https://dados.gov.br/dataset/procedimentos-hospitalares-por-uf`. Accessed: 2021-11-26.

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M. et al. (2016). Tensorflow: A system for large-scale machine learning. Em *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, páginas 265–283.

Abdi, H. e Williams, L. J. (2010). Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459.

Aghajanyan, A., Zettlemoyer, L. e Gupta, S. (2020). Intrinsic dimensionality explains the effectiveness of language model fine-tuning.

Airio, E. (2006). Word normalization and decompounding in mono- and bilingual IR. *Information Retrieval*, 9(3):249–271.

Alsentzer, E., Murphy, J. R., Boag, W., Weng, W.-H., Jin, D., Naumann, T. e McDermott, M. B. A. (2019). Publicly available clinical bert embeddings.

Amit, Y. e Geman, D. (1997). Shape quantization and recognition with randomized trees. *Neural computation*, 9(7):1545–1588.

Anandarajan, M., Hill, C. e Nolan, T. (2018). *Practical Text Analytics*. Springer-Verlag GmbH.

Angraal, S., Mortazavi, B. J., Gupta, A., Khera, R., Ahmad, T., Desai, N. R., Jacoby, D. L., Masoudi, F. A., Spertus, J. A. e Krumholz, H. M. (2020). Machine learning prediction of mortality and hospitalization in heart failure with preserved ejection fraction. *JACC: Heart Failure*, 8(1):12–21.

Araujo, T. V., Pires, S. R. e Bandiera-Paiva, P. (2014). Adoção de padrões para registro eletrônico em saúde no brasil. *Revista Eletrônica de Comunicação, Informação e Inovação em Saúde*, 8(4).

Badrinarayanan, V., Kendall, A. e Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495.

Baig, M. M., Hua, N., Zhang, E., Robinson, R., Spyker, A., Armstrong, D., Whittaker, R., Robinson, T. e Ullah, E. (2020). A machine learning model for predicting risk of hospital readmission within 30 days of discharge: validated with LACE index and patient at risk of hospital readmission (PARR) model. *Medical & Biological Engineering & Computing*, 58(7):1459–1466.

Baillie, C. A., VanZandbergen, C., Tait, G., Hanish, A., Leas, B., French, B., Hanson, C. W., Behta, M. e Umscheid, C. A. (2013). The readmission risk flag: Using the electronic health record to automatically identify patients at risk for 30-day readmission. *Journal of Hospital Medicine*, 8(12):689–695.

Barak-Corren, Y., Fine, A. M. e Reis, B. Y. (2017). Early prediction model of patient hospitalization from the pediatric emergency department. *Pediatrics*, 139(5):e20162785.

Barkan, O., Razin, N., Malkiel, I., Katz, O., Caciularu, A. e Koenigstein, N. (2019). Scalable attentive sentence-pair modeling via distilled sentence embedding.

Baro, E. F., Oliveira, L. S. e de Souza Britto Junior, A. (2022). Predicting hospitalization from health insurance data. Em *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, páginas 2790–2795.

Bartlett, P., Freund, Y., Lee, W. S. e Schapire, R. E. (1998). Boosting the margin: A new explanation for the effectiveness of voting methods. *The annals of statistics*, 26(5):1651–1686.

Billings, J., Georghiou, T., Blunt, I. e Bardsley, M. (2013). Choosing a model to predict hospital admission: an observational study of new variants of predictive models for case finding. *BMJ Open*, 3(8):e003352.

Bindman, A. B. (1995). Preventable hospitalizations and access to health care. *JAMA: The Journal of the American Medical Association*, 274(4):305.

Braschler, M. e Ripplinger, B. (2004). How effective is stemming and decompounding for german text retrieval? *Information Retrieval*, 7(3):291–316.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A. et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Buber, E. e Diri, B. (2019). Web page classification using RNN. *Procedia Computer Science*, 154:62–72.

Bühlmann, P. e Yu, B. (2002). Analyzing bagging. *The Annals of Statistics*, 30(4).

Chen, T. e Guestrin, C. (2016). XGBoost. ACM.

Chollet, F. et al. (2015). keras.

Choudhry, S. A., Li, J., Davis, D., Erdmann, C., Sikka, R. e Sutariya, B. (2013). A public-private partnership develops and externally validates a 30-day hospital readmission risk prediction model. *Online Journal of Public Health Informatics*, 5(2).

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., Schuh, P., Shi, K., Tsvyashchenko, S., Maynez, J., Rao, A., Barnes, P., Tay, Y., Shazeer, N., Prabhakaran, V., Reif, E., Du, N., Hutchinson, B., Pope, R., Bradbury, J., Austin, J., Isard, M., Gur-Ari, G., Yin, P., Duke, T., Levskaya, A., Ghemawat, S., Dev, S., Michalewski, H., Garcia, X., Misra, V., Robinson, K., Fedus, L., Zhou, D., Ippolito, D., Luan, D., Lim, H., Zoph, B., Spiridonov, A., Sepassi, R., Dohan, D., Agrawal, S., Omernick, M., Dai, A. M., Pillai, T. S., Pellat, M., Lewkowycz, A., Moreira, E., Child, R., Polozov, O., Lee, K., Zhou, Z., Wang, X., Saeta, B., Diaz, M., Firat, O., Catasta, M., Wei, J., Meier-Hellstern, K., Eck, D., Dean, J., Petrov, S. e Fiedel, N. (2022). Palm: Scaling language modeling with pathways.

Culler, S. D., Parchman, M. L. e Przybylski, M. (1998). Factors related to potentially preventable hospitalizations among the elderly. *Medical care*, páginas 804–817.

Cutler, A., Cutler, D. R. e Stevens, J. R. (2012). Random forests. Em *Ensemble Machine Learning*, páginas 157–175. Springer US.

Dahouda, M. K. e Joe, I. (2021). A deep-learned embedding technique for categorical features encoding. *IEEE Access*, 9:114381–114391.

Dai, W., Brisimi, T. S., Adams, W. G., Mela, T., Saligrama, V. e Paschalidis, I. C. (2015). Prediction of hospitalization due to heart diseases by supervised learning methods. *International Journal of Medical Informatics*, 84(3):189–197.

Dettmers, T., Pagnoni, A., Holtzman, A. e Zettlemoyer, L. (2023). Qlora: Efficient finetuning of quantized llms.

Devlin, J., Chang, M.-W., Lee, K. e Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding.

Fawcett, T. (2006). An introduction to ROC analysis. 27(8):861–874.

Fernández, A., García, S., Galar, M., Prati, R. C. e Krawczyk, B. (2018). *Learning from Imbalanced Data Sets*. Springer-Verlag GmbH.

Ferrarini, C. D. T. (1977). Conceitos e definições em saúde. *Revista Brasileira de Enfermagem*, 30(3):314–338.

Ferreira, J. B. B., Borges, M. J. G., dos Santos, L. L. e Forster, A. C. (2014). Internações por condições sensíveis à atenção primária à saúde em uma região de saúde paulista, 2008 a 2010. 23(1):45–56.

Giménez, M., Palanca, J. e Botti, V. (2020). Semantic-based padding in convolutional neural networks for improving the performance in natural language processing. a case of study in sentiment analysis. *Neurocomputing*, 378:315–323.

Glenn, N. D. (2005). *Cohort analysis*, volume 5. Sage.

Goldberg, Y. e Levy, O. (2014). word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method.

Guimarães, M. (1985). Exames de laboratório: sensibilidade, especificidade, valor preditivo positivo. *Revista da Sociedade Brasileira de Medicina Tropical*, 18:117–120.

Gunasekaran, H., Ramalakshmi, K., Arokiaraj, A. R. M., Kanmani, S. D., Venkatesan, C. e Dhas, C. S. G. (2021). Analysis of DNA sequence classification using CNN and hybrid models. *Computational and Mathematical Methods in Medicine*, 2021:1–12.

Hajian-Tilaki, K. (2013). Receiver operating characteristic (roc) curve analysis for medical diagnostic test evaluation. *Caspian journal of internal medicine*, 4(2):627.

Han, Z., Gao, C., Liu, J., Zhang, J. e Zhang, S. Q. (2024). Parameter-efficient fine-tuning for large models: A comprehensive survey.

Hao, S., Jin, B., Shin, A. Y., Zhao, Y., Zhu, C., Li, Z., Hu, Z., Fu, C., Ji, J., Wang, Y., Zhao, Y., Dai, D., Culver, D. S., Alfreds, S. T., Rogow, T., Stearns, F., Sylvester, K. G., Widen, E. e Ling, X. B. (2014). Risk prediction of emergency department revisit 30 days post discharge: A prospective study. *PLoS ONE*, 9(11):e112944.

Hebert, C., Shivade, C., Foraker, R., Wasserman, J., Roth, C., Mekhjian, H., Lemeshow, S. e Embi, P. (2014a). Diagnosis-specific readmission risk prediction using electronic health data: a retrospective cohort study. *BMC Medical Informatics and Decision Making*, 14(1).

Hebert, C., Shivade, C., Foraker, R., Wasserman, J., Roth, C., Mekhjian, H., Lemeshow, S. e Embi, P. (2014b). Diagnosis-specific readmission risk prediction using electronic health data: a retrospective cohort study. *BMC medical informatics and decision making*, 14(1):1–9.

Hippisley-Cox, J. e Coupland, C. (2013). Predicting risk of emergency admission to hospital using primary care data: derivation and validation of QAdmissions score. *BMJ Open*, 3(8):e003482.

Ho, T. K. (1998). The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):832–844.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L. e Chen, W. (2021). Lora: Low-rank adaptation of large language models.

Jang, B., Kim, M., Harerimana, G., ug Kang, S. e Kim, J. W. (2020). Bi-LSTM model to increase accuracy in text classification: Combining word2vec CNN and attention mechanism. *Applied Sciences*, 10(17):5841.

Jiang, H. J., Russo, C. A. e Barrett, M. L. (2011). Nationwide frequency and costs of potentially preventable hospitalizations, 2006: statistical brief# 72.

Kang, H. e Yuan, X. (2014). Natural language processing technologies for multi-level intelligent spam mail-filter. *International Journal of Machine Learning and Computing*, 4(3):271–274.

Kang, J., Wang, H., Yuan, F., Wang, Z., Huang, J. e Qiu, T. (2020). Prediction of precipitation based on recurrent neural networks in jingdezhen, jiangxi province, china. *Atmosphere*, 11(3):246.

Korenius, T., Laurikkala, J., Järvelin, K. e Juhola, M. (2004). Stemming and lemmatization in the clustering of finnish text documents. Em *Proceedings of the Thirteenth ACM conference on Information and knowledge management - CIKM '04*. ACM Press.

Lai, S., Liu, K., Xu, L. e Zhao, J. (2015). How to generate a good word embedding?

Larcher, C., Piau, M., Finardi, P., Gengo, P., Esposito, P. e Caridá, V. (2023). Cabrita: closing the gap for foreign languages.

Lee, H. e Song, J. (2019). Introduction to convolutional neural network using keras; an understanding from a statistician. *Communications for Statistical Applications and Methods*, 26(6):591–610.

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H. e Kang, J. (2020). Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Li, C., Farkhoor, H., Liu, R. e Yosinski, J. (2018). Measuring the intrinsic dimension of objective landscapes.

Li, X. L. e Liang, P. (2021). Prefix-tuning: Optimizing continuous prompts for generation.

Linardatos, P., Papastefanopoulos, V. e Kotsiantis, S. (2020). Explainable ai: A review of machine learning interpretability methods. *Entropy*, 23(1):18.

Liu, H., Tam, D., Muqeeth, M., Mohta, J., Huang, T., Bansal, M. e Raffel, C. (2022). Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. e Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach.

Lopes, A., Nogueira, R., Lotufo, R. e Pedrini, H. (2020). Lite training strategies for portuguese-english and english-portuguese translation.

Lorenzoni, G., Sabato, S. S., Lanera, C., Bottigliengo, D., Minto, C., Ocagli, H., Paolis, P. D., Gregori, D., Iliceto, S. e Pisanò, F. (2019). Comparison of machine learning techniques for prediction of hospitalization in heart failure patients. *Journal of Clinical Medicine*, 8(9):1298.

Lundberg, S. (2018). Shap. Accessed on August 20, 2024.

Lundberg, S. e Lee, S.-I. (2017). A unified approach to interpreting model predictions.

Lunn, M. e McNeil, D. (1995). Applying cox regression to competing risks. *Biometrics*, 51(2):524.

Ma, H., Bandos, A. I., Rockette, H. E. e Gur, D. (2013). On use of partial area under the ROC curve for evaluation of diagnostic performance. *Statistics in Medicine*, 32(20):3449–3458.

Mikolov, T., Chen, K., Corrado, G. e Dean, J. (2013). Efficient estimation of word representations in vector space.

Ministério da Saúde (2002). Regulamento técnico para planejamento, programação, elaboração e avaliação de projetos físicos de estabelecimentos assistenciais de saúde.

Ministério da Saúde (2008). Portaria n. 221, de 17 de abril de 2008. *Diário Oficial [da] República Federativa do Brasil*.

Monsen, K., Swanberg, H., Oancea, S. e Westra, B. (2012). Exploring the value of clinical data standards to predict hospitalization of home care patients. *Applied Clinical Informatics*, 03(04):419–436.

Organization, W. H. et al. (2004). *ICD-10: international statistical classification of diseases and related health problems: tenth revision*. World Health Organization.

Pan, S. J. e Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359.

Pappas, G., Hadden, W. C., Kozak, L. J. e Fisher, G. F. (1997). Potentially avoidable hospitalizations: inequalities in rates between us socioeconomic groups. *American Journal of public health*, 87(5):811–816.

Park, S. H., Goo, J. M. e Jo, C.-H. (2004). Receiver operating characteristic (roc) curve: practical review for radiologists. *Korean journal of radiology*, 5(1):11–18.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L. e Lerer, A. (2017). Automatic differentiation in pytorch. Em *NIPS-W*.

Patel, S. J., Chamberlain, D. B. e Chamberlain, J. M. (2018). A machine learning approach to predicting need for hospitalization for pediatric asthma exacerbation at the time of emergency department triage. 25(12):1463–1470.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. e Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Pennington, J., Socher, R. e Manning, C. D. (2014). Glove: Global vectors for word representation. Em *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, páginas 1532–1543.

Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K. e Zettlemoyer, L. (2018). Deep contextualized word representations. Em *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Association for Computational Linguistics.

Prati, R., Batista, G., Monard, M. et al. (2008). Curvas roc para avaliação de classificadores. *Revista IEEE América Latina*, 6(2):215–222.

Quinlan, J. R. (1986). Induction of decision trees. *Machine learning*, 1:81–106.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I. et al. (2019). Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Rajaraman, A. e Ullman, J. D. (2011). Data mining. Em *Mining of Massive Datasets*, páginas 1–17. Cambridge University Press.

Rana, S., Tran, T., Luo, W., Phung, D., Kennedy, R. L. e Venkatesh, S. (2014). Predicting unplanned readmission after myocardial infarction from routinely collected administrative hospital data. *Australian Health Review*, 38(4):377.

Reimers, N. e Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks.

Rumelhart, D. E., Hinton, G. E. e Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, 323(6088):533–536.

Sagi, O. e Rokach, L. (2018). Ensemble learning: A survey. *WIREs Data Mining and Knowledge Discovery*, 8(4).

Schapire, R. E. (2003). The boosting approach to machine learning: An overview. *Nonlinear estimation and classification*, páginas 149–171.

Shapley, L. S. (1953). Stochastic games. *Proceedings of the national academy of sciences*, 39(10):1095–1100.

Shazeer, N. (2020). Glu variants improve transformer.

Sherstinsky, A. (2020). Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *Physica D: Nonlinear Phenomena*, 404:132306.

Shi, L., Samuels, M. E., Pease, M., Bailey, W. P. e Corley, E. H. (1999). Patient characteristics associated with hospitalizations for ambulatory care sensitive conditions in south carolina. *Southern medical journal*, 92(10):989–998.

Shickel, B., Tighe, P. J., Bihorac, A. e Rashidi, P. (2018). Deep EHR: A survey of recent advances in deep learning techniques for electronic health record (EHR) analysis. *IEEE Journal of Biomedical and Health Informatics*, 22(5):1589–1604.

Singal, A. G., Rahimi, R. S., Clark, C., Ma, Y., Cuthbert, J. A., Rockey, D. C. e Amarasingham, R. (2013). An automated model using electronic medical record data identifies patients with cirrhosis at high risk for readmission. *Clinical Gastroenterology and Hepatology*, 11(10):1335–1341.e1.

Smith, D. H., Johnson, E. S., Thorp, M. L., Yang, X., Petrik, A., Platt, R. W. e Crispell, K. (2011). Predicting poor outcomes in heart failure. *The Permanente Journal*, 15(4):4.

Souza, D. K. d. e Peixoto, S. V. (2017). Estudo descritivo da evolução dos gastos com internações hospitalares por condições sensíveis à atenção primária no brasil, 2000-2013. *Epidemiologia e Serviços de Saúde*, 26:285–294.

Souza, F., Nogueira, R. e Lotufo, R. (2020). Bertimbau: Pretrained bert models for brazilian portuguese. Em Cerri, R. e Prati, R. C., editores, *Intelligent Systems*, páginas 403–417, Cham. Springer International Publishing.

Souza, F., Nogueira, R. e Lotufo, R. (2023). Bert models for brazilian portuguese: Pretraining, evaluation and tokenization analysis. *Applied Soft Computing*, 149:110901.

Su, J., Lu, Y., Pan, S., Murtadha, A., Wen, B. e Liu, Y. (2021). Roformer: Enhanced transformer with rotary position embedding.

Suen, Y. L., Melville, P. e Mooney, R. J. (2005). *Combining Bias and Variance Reduction Techniques for Regression Trees*, páginas 741–749. Springer Berlin Heidelberg.

TensorFlow (2022). Tensorflow core v2.8.0 | api documentation.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F. et al. (2023a). Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E. e Lample, G. (2023b). Llama: Open and efficient foundation language models.

Vaithianathan, R., Jiang, N., Ashton, T. et al. (2012). A model for predicting readmission risk in new zealand. *Faculty of Business and Law. AUT University*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. e Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Wagner Filho, J. A., Wilkens, R., Idiart, M. e Villavicencio, A. (2018). The brwac corpus: a new open resource for brazilian portuguese. Em *Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018)*.

Wang, L., Porter, B., Maynard, C., Bryson, C., Sun, H., Lowy, E., McDonell, M., Frisbee, K., Nielson, C. e Fihn, S. D. (2012). Predicting risk of hospitalization or death among patients with heart failure in the veterans health administration. *The American Journal of Cardiology*, 110(9):1342–1349.

Wolf, T., Debut, L., Sanh, V., Chaumond, J. et al. (2020). Transformers: State-of-the-art natural language processing. Em Liu, Q. e Schlangen, D., editores, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, páginas 38–45, Online. Association for Computational Linguistics.

Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A. e Raffel, C. (2020). mt5: A massively multilingual pre-trained text-to-text transformer.

Yuen, E. J. (2004). Severity of illness and ambulatory care–sensitive conditions. *Medical Care Research and Review*, 61(3):376–391.

Zhang, B. e Sennrich, R. (2019). Root mean square layer normalization. *Advances in Neural Information Processing Systems*, 32.

APÊNDICE A – EXEMPLOS DE SENTENÇAS HISTÓRICAS GERADAS

A.1 EXEMPLO DE UMA SENTENÇA HISTÓRICA DE PACIENTE ANTES DA OCORRÊNCIA DE AVC

“ceratoscopia computadorizada - monocular paquimetria ultrassônica - monocular microscopia especular de córnea - monocular paquimetria ultrassônica - monocular microscopia especular de córnea - monocular consulta em consultório no horário normal ou preestabelecido ceratoscopia computadorizada - monocular proteína c teste imunológico medicamentos em geral hemossedimentação vhs - pesquisa e/ou dosagem gases hemograma com contagem de plaquetas ou frações eritrograma leucograma plaquetas ecg convencional de até 12 derivações corpos estranhos pólipos ou biópsia - em consultório cerumen - remoção bilateral consulta em consultório no horário normal ou preestabelecido vídeo-endoscopia naso-sinusal com ótica flexível cultura bacteriana em diversos materiais biológicos creatinina - pesquisa e/ou dosagem ecg convencional de até 12 derivações colesterol hdl - pesquisa e/ou dosagem hemograma com contagem de plaquetas ou frações eritrograma leucograma plaquetas ácido úrico - pesquisa e/ou dosagem glicemia após sobrecarga com dextrosol ou glicose - pesquisa e/ou dosagem colesterol ldl - pesquisa e/ou dosagem colesterol total - pesquisa e/ou dosagem captopril - 50 mg com ct bl al plas trans x 30 água p/ injeção rivotril - 0 5 mg com cx bl al plas inc x 20 solução de cloreto de sódio b.braun - 9 mg/ml sol inj iv cx 50 fa plas inc sist fech x 100 ml dramin b6 dl - 3 mg/ml + 5 mg/ml + 100 mg/ml + 100 mg/ml sol inj cx 100 amp vd amb x 10 ml emb frac consulta em consultório no horário normal ou preestabelecido atensina - 0 10 mg com ct bl al pl inc x 30 vertix - 10 mg com ct bl al plas inc x 50 taxa de sala de observação até 6 horas consulta em consultório no horário normal ou preestabelecido anticorpos antitireóide tireoglobulina - pesquisa e/ou dosagem antimicrossomal - pesquisa e/ou dosagem proteína c reativa quantitativa - pesquisa e/ou dosagem t4 livre - pesquisa e/ou dosagem sífilis - vdrl colesterol total - pesquisa e/ou dosagem hemograma com contagem de plaquetas ou frações eritrograma leucograma plaquetas curva insulínica 6 dosagens - pesquisa e/ou dosagem curva glicêmica 6 dosagens - pesquisa e/ou dosagem colesterol ldl - pesquisa e/ou dosagem colesterol vldl - pesquisa e/ou dosagem colesterol hdl - pesquisa e/ou dosagem fator antinúcleo fan - pesquisa e/ou dosagem triglicerídeos - pesquisa e/ou dosagem tireoestimulante hormônio tsh - pesquisa e/ou dosagem masculino”.

A.2 EXEMPLO DE UMA SENTENÇA HISTÓRICA AVALIADA POR MEIO DO SHAP

“segmentos consulta em consultório no horário normal ou preestabelecido vitamina d 25 hidroxi pesquisa e / ou dosagem vitamina d3 cálcio iônico - pesquisa e / ou dosagem clearance de creatinina eletroferese de proteínas fosfatase alcalina - pesquisa e / ou dosagem gama - glutamil transferase - pesquisa e / ou dosagem transaminase oxalacética amino transferase aspartato - pesquisa e / ou dosagem transaminase pirúvica amino transferase de alanina - pesquisa e / ou dosagem uréia - pesquisa e / ou dosagem hemograma com contagem de plaquetas ou frações eritrograma leucograma plaquetas hemossedimentação vhs - pesquisa e / ou dosagem pth - pesquisa e / ou dosagem testosterona total - pesquisa e / ou dosagem tireoestimulante hormônio tsh - pesquisa e / ou dosagem testosterona livre - pesquisa e / ou dosagem proteína c reativa qualitativa - pesquisa consulta em consultório no horário normal ou preestabelecido consulta em consultório no horário normal ou preestabelecido ecodopplercardiograma transtorácico teste ergométrico computadorizado inclui ecg basal convencional consulta em consultório no

horário normal ou preestabelecido consulta em consultório no horário normal ou preestabelecido diluente - água para injeção - sol inj cx 50 amp vd inc x 10 ml cateter intravenoso poliuretano 18gx2pol 51mm surflash sr * ff1851 tc - angiotomografia coronariana cortisonal * restrito hosp. - 500 mg. po inj. ct. 50 fa vd. inc. hosp. - uniao quimica isordil sl - 5 mg com sub ling ct bl al plas inc x 30 ultravist - 768 86 mg / ml sol inj cx 10 fa vd inc x 100 ml holter de 24 horas - 2 ou mais canais - analógico consulta em consultório no horário normal ou preestabelecido consulta em consultório no horário normal ou preestabelecido consulta em consultório no horário normal ou preestabelecido consulta em consultório no horário normal ou preestabelecido ecodopplercardiograma transtorácico doppler colorido de vasos cervicais arteriais bilateral carótidas e vertebrais ácido úrico - pesquisa e / ou dosagem colesterol hdl - pesquisa e / ou dosagem colesterol ldl - pesquisa e / ou dosagem colesterol total - pesquisa e / ou dosagem creatinina - pesquisa e / ou dosagem creatino fosfoquinase total ck - pesquisa e / ou dosagem glicose - pesquisa e / ou dosagem magnésio - pesquisa e / ou dosagem potássio - pesquisa e / ou dosagem triglicerídeos - pesquisa e / ou dosagem hemograma com contagem de plaquetas ou frações eritrograma leucograma plaquetas t4 livre - pesquisa e / ou dosagem tireoestimulante hormônio tsh - pesquisa e / ou dosagem isofarma - agua para injeção isofarma - agua para injeção persantin - 10 mg / 2ml sol inj ct 5 amp vd inc x 2 ml aminofilina - 200 mg com ct bl al plas pvdc 250 / 120 trans x 20 consulta em consultório no horário normal ou preestabelecido us - abdome superior fígado vias biliares vesícula pâncreas e baço consulta em consultório no horário normal ou preestabelecido consulta em consultório no horário normal ou preestabelecido doppler colorido venoso de membro inferior - unilateral masculino”

APÊNDICE B – SOFTWARES E HARDWARES UTILIZADOS

B.1 SOFTWARES UTILIZADOS

Para organizar os dados, extrair características, construir o sistema de previsão e realizar os experimentos, utilizou-se a linguagem *python* associada a bibliotecas para processamento de linguagem natural (PLN) e aprendizado de máquina. Assim foram utilizadas as seguintes ferramentas de software:

- *Scikit-learning* (Pedregosa et al., 2011), uma biblioteca de código aberto para aprendizado de máquina em Python.
- *TensorFlow* (Abadi et al., 2016), uma biblioteca de código aberto para aprendizado de máquina aplicável em uma ampla variedade de tarefas. Fornece API em Python, C++ e JavaScript (TensorFlow, 2022).
- *Keras* (Chollet et al., 2015), uma biblioteca de código aberto escrita em Python. É capaz de rodar em cima do TensorFlow, Microsoft Cognitive Toolkit, R, Theano, ou PlaidML.
- *Pytorch* (Paszke et al., 2017), é uma biblioteca de aprendizado de máquina baseada na biblioteca *Torch*, usada para aplicações como visão computacional e processamento de linguagem natural, originalmente desenvolvida pela Meta AI e agora parte da Linux Foundation.
- *Transformers* (Wolf et al., 2020), é uma biblioteca de código aberto para *deep learning* desenvolvida pela *Hugging Face* (ref. site), projetada para diversas modalidades, como processamento de linguagem natural, visão computacional, áudio e aplicações multimodais.

B.2 HARDWARE UTILIZADO

Os treinamentos foram realizados em uma GPU GeForce NVIDIA RTX A5000 24GB dos laboratórios do Departamento de Informática da Universidade Federal do Paraná (UFPR).

APÊNDICE C – OUTRAS LLMS TREINADAS

C.1 LLMS TREINADAS EM INGLÊS

Apresenta-se aqui os hiperparâmetros de treinamento, resultados e links para download dos modelos treinados em inglês. De maneira geral, os modelos descritos nesta tese e dados usados nos testes podem ser encontrados no link `https://huggingface.co/efbaro`.

C.1.1 RoBERTa-MLM-EN

Tabela C.1: Hiperparâmetros usados no treinamento MLM e *fine-tuning* com dados em inglês.

Hiperparâmetro	**Valor (MLM)**	**Valor (*fine-tuning*)**
Learning rate	1e-4	5e-5
Maximum sequence length	512	512
Gradient accumulation step	4	16
Batch size	32	64
Mask %	15%	-
Epochs	2	2

Tabela C.2: Média da AUC, Sensibilidade, Especificidade, F1-Score e desvio padrão do modelo ajustado previsão de internação a partir de uma camada completamente conectada.

F1-Score	**Especificidade**	**Sensibilidade**	**AUC**
89,0±0,4	90,7±0,4	87,2±0,8	96,2±0,1

Tabela C.3: Links para download.

Modelo	**Link**
RoBERTa-MLM-EN (pré-treinado)	`https://huggingface.co/efbaro/HealthHistoryRoBERTa-en`
RoBERTa-MLM-EN-FT (*fine-tuning*)	`https://huggingface.co/efbaro/HealthHistoryRoBERTa-en-ft`

C.1.2 BERT + MLM

Tabela C.4: Hiperparâmetros usados no treinamento MLM e *fine-tuning* com dados em inglês.

Hiperparâmetro	**Valor (MLM)**	**Valor (*fine-tuning*)**
Learning rate	1e-4	5e-5
Maximum sequence length	512	512
Gradient accumulation step	4	16
Batch size	16	32
Mask %	15%	-
Epochs	5	2

Tabela C.5: Média da AUC, Sensibilidade, Especificidade, F1-Score e desvio padrão do modelo ajustado previsão de internação a partir de uma camada completamente conectada.

F1-Score	**Especificidade**	**Sensibilidade**	**AUC**
88,9±0,3	90,5±0,7	87,1±0,9	96,1±0,2

Tabela C.6: Links para download.

Modelo	**Link**
BERT+MLM (pré-treinado)	`https://huggingface.co/efbaro/HealthHistoryBERT-en`
BERT+MLM-FT (*fine-tuning*)	`https://huggingface.co/efbaro/HealthHistoryBERT-en-ft`

C.1.3 BioBERT + MLM

Tabela C.7: Hiperparâmetros usados no treinamento MLM e *fine-tuning* com dados em inglês.

Hiperparâmetro	**Valor (MLM)**	**Valor (*fine-tuning*)**
Learning rate	1e-4	1e-4
Maximum sequence length	512	512
Gradient accumulation step	4	4
Batch size	16	16
Mask %	15%	-
Epochs	1	1

Tabela C.8: Média da AUC, Sensibilidade, Especificidade, F1-Score e desvio padrão do modelo ajustado previsão de internação a partir de uma camada completamente conectada.

F1-Score	Especificidade	Sensibilidade	AUC
86,7±0,4	89,6±0,3	83,7±0,8	94,9±0,2

Tabela C.9: Links para download.

Modelo	Link
BioBERT+MLM (pré-treinado)	`https://huggingface.co/efbaro/HealthHistoryBioBERT-en`
BioBERT+MLM-FT (*fine-tuning*)	`https://huggingface.co/efbaro/HealthHistoryBioBERT-en-ft`

C.1.4 Bio-ClinicalBERT + MLM

Tabela C.10: Hiperparâmetros usados no treinamento MLM e *fine-tuning* com dados em inglês.

Hiperparâmetro	Valor (MLM)	Valor (*fine-tuning*)
Learning rate	1e-4	1e-4
Maximum sequence length	512	512
Gradient accumulation step	4	4
Batch size	16	16
Mask %	15%	-
Epochs	1	1

Tabela C.11: Média da AUC, Sensibilidade, Especificidade, F1-Score e desvio padrão do modelo ajustado previsão de internação a partir de uma camada completamente conectada.

F1-Score	Especificidade	Sensibilidade	AUC
87,1±0,1	89,0±0,4	85,1±1,1	94,9±0,2

Tabela C.12: Links para download.

Modelo	Link
Bio-ClinicalBERT+MLM (pré-treinado)	`https://huggingface.co/efbaro/HealthHistoryBio_ClinicalBERT-en`
Bio-ClinicalBERT+MLM-FT (*fine-tuning*)	`https://huggingface.co/efbaro/HealthHistoryBio_ClinicalBERT-en-ft`

C.1.5 OpenLLaMA3Bv2, OpenLLaMA7Bv2 e OpenLLaMA13B + Fine-tuning (FT)

Esses três modelos utilizaram os mesmos hiperparâmetros para treinamento, variando apenas a quantidade de dados de treinamento. Os modelos OpenLLaMA3Bv2 e OpenLLaMA7Bv2 utilizaram 167.431 sentenças históricas enquanto OpenLLaMA13B utilizou 418.579.

Tabela C.13: Hiperparâmetros usados no *fine-tuning* com dados em inglês.

Hiperparâmetro	**Valor (*fine-tuning*)**
Learning rate	5e-5
Maximum sequence length	1024
Gradient accumulation step	8
Batch size	8
Epochs	1

Tabela C.14: Hiperparâmetros do LoRA PEFT usados *fine-tuning*.

Hiperparâmetro	**Valor (*fine-tuning*)**
r	8
alpha	32
dropout	0.1

Tabela C.15: Média da AUC, Sensibilidade, Especificidade, F1-Score e desvio padrão do modelo ajustado previsão de internação a partir de uma camada completamente conectada.

Modelo	**F1-Score**	**Especificidade**	**Sensibilidade**	**AUC**
OpenLLaMA3Bv2+FT	86,5±0,6	94,6±0,4	77,9±1,3	95,6±0,3
OpenLLaMA7Bv2+FT	87,9±0,7	92,8±0,7	83,8±1,3	95,7±0,1
OpenLLaMA13B+FT	89,0±0,5	93,4±0,6	84,2±1,1	96,5±0,3

Tabela C.16: Links para download.

Modelo	**Link**
OpenLLaMA3Bv2+Fine-tuning (*fine-tuning*)	`https://huggingface.co/efbaro/HealthHistoryOpenLLaMA3Bv2-en-ft`
OpenLLaMA7Bv2+Fine-tuning (*fine-tuning*)	`https://huggingface.co/efbaro/HealthHistoryOpenLLaMA7Bv2-en-ft`
OpenLLaMA13B+Fine-tuning (*fine-tuning*)	`https://huggingface.co/efbaro/HealthHistoryOpenLLaMA13B-en-ft`