UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE ENGENHARIA DE COMPUTAÇÃO

FILIPE FARIA DIAS

Recuperação de evidências em relatórios de ensaios clínicos utilizando o modelo biomédico RoBERTa

Monografia apresentada como requisito parcial para a obtenção do grau de Bacharel em Engenharia da Computação

Orientador: Prof. Dra. Viviane P. Moreira
Co-orientador: Abel Corrêa Dias

Porto Alegre
2023

“If I have seen farther than others,
it is because I stood on the shoulders of giants.”
— Sir Isaac Newton

AGRADECIMENTOS

Primeiramente agradeço aos meus pais, Maria Leonilda Faria Lins e Rogério Conceição Prestes Dias, por todo o apoio que me proporcionaram. Também ao meu irmão, Fábio Faria Dias, pela ajuda e por todo o apoio durante toda a minha graduação.

À Professora Viviane Pereira Moreira por toda a ajuda e orientação neste trabalho. Ao Abel Corrêa Dias, que me ajudou a realizar esta conquista com o seu conhecimento e disponibilidade.

Aos meus amigos Alencar da Costa, Gabriel Augusto Engel, Mathues Woeffel Camargo, Raphael Scherpinski Brandão por todo o apoio, ajuda e companhia durante toda a graduação.

Por fim, agradeço a todos os professores e colaboradores da UFRGS, que contribuíram em minha jornada acadêmica.

RESUMO

Nos últimos anos, houve um aumento significativo de publicações de relatórios de ensaios clínicos com mais de 10.000 relatórios somente para câncer de mama. Consequentemente, tornou-se inviável para os profissionais de saúde ficarem atualizados sobre toda a literatura, com o fim de fornecer o melhor tratamento possível de acordo com os sintomas dos pacientes, dada a elevada quantidade de informações disponíveis a todo momento. Seguindo nesse contexto, o *workshop* SemEval de 2023 propôs um desafio que envolve desenvolver um sistema que faz a recuperação de um conjunto de evidências que suportam uma consulta em relatórios de ensaios clínicos. Muitos times participaram desse desafio utilizando diversas técnicas diferentes. Observou-se que as técnicas que utilizaram modelos generativos obtiveram os melhores resultados com relação à métrica F1, contudo, os modelos discriminativos que implementaram um modelo com base no DeBERTa-large também obtiveram resultados competitivos. O objetivo do trabalho foi desenvolver um modelo que faz a recuperação de evidências nesses relatórios clínicos utilizando o modelo Biomed RoBERTa. Nossa abordagem envolveu realizar uma serie de treinamentos variando a métrica a de otimização (acurácia, reovcação e F1) e os hiperparâmetros (taxa de aprendizado e tamanho máximo da sequência de entrada). Nossos melhores resultados foram obtidos com o treinamento baseado na métrica de revocação, que foram superiores ao resultado que obtivemos no *workshop*, com o valor de F1 de 0,733.

Palavras-chave: Aprendizado de máquina. Configuração de hiperparâmetros. *Deep learning*. Processamento de linguagem natural. Recuperação de evidências.

Evidence retrieval in clinical trial reports using the biomedical RoBERTa model.

ABSTRACT

In recent years, there has been a significant increase in the publication of clinical trial reports, with over 10,000 reports for breast cancer alone. Consequently, it has become unfeasible for healthcare professionals to stay updated on the entire literature in order to provide the best possible treatment based on patients' symptoms, given the vast amount of constantly available information. In this context, the SemEval 2023 workshop proposed a challenge involving the development of a system that retrieves a set of evidence supporting a query in clinical trial reports. Many teams participated in this challenge using various techniques. It was observed that techniques using generative models achieved the best results in terms of the F1 metric; however, discriminative models implementing a DeBERTa-large-based model also achieved competitive results. The objective of this work was to develop a model for evidence retrieval in these clinical reports using the Biomed RoBERTa model. Our approach involved a series of training iterations, varying the optimization metric (accuracy, recall, and F1) and hyperparameters (learning rate and maximum input sequence length). Our best results were obtained with training based on the recall metric, which outperformed our workshop result with an F1 score of 0.733.

Keywords: Evidence retrieval, Deep learning, Hiperparameter settings, Machine learning, Natural language processing.

LISTA DE ABREVIATURAS E SIGLAS

BERT	Bidirectional Encoder Representations from Transformers
CTR	Clinical Trial Data
ILN	Inferência em Linguagem Natural
ML	Modelos de Linguagem
MLM	Masked Language Modeling
NLI4CT	Natural Language Inference for Clinical Trials
NSP	Next Sentence Prediction
PLN	Processamento de Linguagem Natural
RI	Recuperação de Informação
RoBERTa	Robustly Optimized BERT Pretraining Approach
UFRGS	Universidade do Rio Grande do Sul

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

Relatórios de ensaios clínicos são documentos que descrevem os detalhes de um estudo clínico, que é um estudo que avalia a segurança e eficácia de uma intervenção médica, como um medicamento, procedimentos cirúrgicos, entre outros. O relatório fornece uma descrição completa da metodologia, dos resultados e das conclusões do estudo. Desta maneira, os relatórios de ensaios clínicos são uma evidência científica sobre tratamentos médicos e são utilizados para tomar a melhor decisão sobre uma determinada condição médica.

Nos últimos anos, houve um aumento significativo de publicações de relatórios de ensaios clínicos. Atualmente, existem mais de 10.000 relatórios apenas para câncer de mama (JULLIEN, 2023). Consequentemente, com o passar dos anos, tornou-se inviável para os profissionais de saúde ficarem atualizados sobre toda a literatura (DEYOUNG et al., 2020). Seguindo nesse contexto, a inferência em linguagem natural (ILN), que envolve determinar se uma hipótese pode ser inferida (dizer se é verdadeira ou falsa) a partir de uma premissa, oferece uma oportunidade de apoiar a interpretação e recuperação em larga escala de evidências médicas. Logo, o desenvolvimento de sistemas que dão suporte a decisões clínicas podem melhorar significativamente a maneira que buscamos evidências atualizadas para providenciar o melhor tratamento possível para as pessoas (SUTTON et al., 2020).

O SemEval *Semantic Evaluation* é uma serie de *workshops* (também pode ser visto como competições) de pesquisa internacional em processamento de linguagem natural (PLN) realizados anualmente com objetivo de avançar o estado da arte da análise semântica computacional e de PLN. O PLN é uma área que permite que computadores entendam, interpretem e gerem textos em linguagem humana. Em cada *workshop* anual, são apresentadas um conjunto de tarefas nas quais os participantes competem contribuindo avanços na área de PLN. O SemEval disponibiliza um conjunto de dados, métricas de avaliação e uma plataforma para que os participantes submetam os seus sistemas para cada tarefa.

O *workshop* do SemEval de 2023 propôs a Tarefa 7 (JULLIEN et al., 2023a), intitulada *Multi-evidence Natural Language Inference for Clinical Trial Data* (NLI4CT). Essa tarefa é dividida em duas sub-tarefas: inferência em linguagem natural (ILN) e recuperação de evidências. A tarefa de ILN consistia em realizar uma relação de inferência entre uma hipótese e um (ou dois) relatório(s) de ensaio(s) clínico(s) e extrair do(s) relató-

rio(s) as informações que corroboram para o resultado da inferência. Em outras palavras, a partir de uma hipótese e um ensaio clínico, os sistemas tinham que dizer se o texto do ensaio confirmava ou contradizia a hipótese. Para responder a essa pergunta, é necessário localizar os trechos do relatório do ensaio clínico que abordam o tópico contido na hipótese.

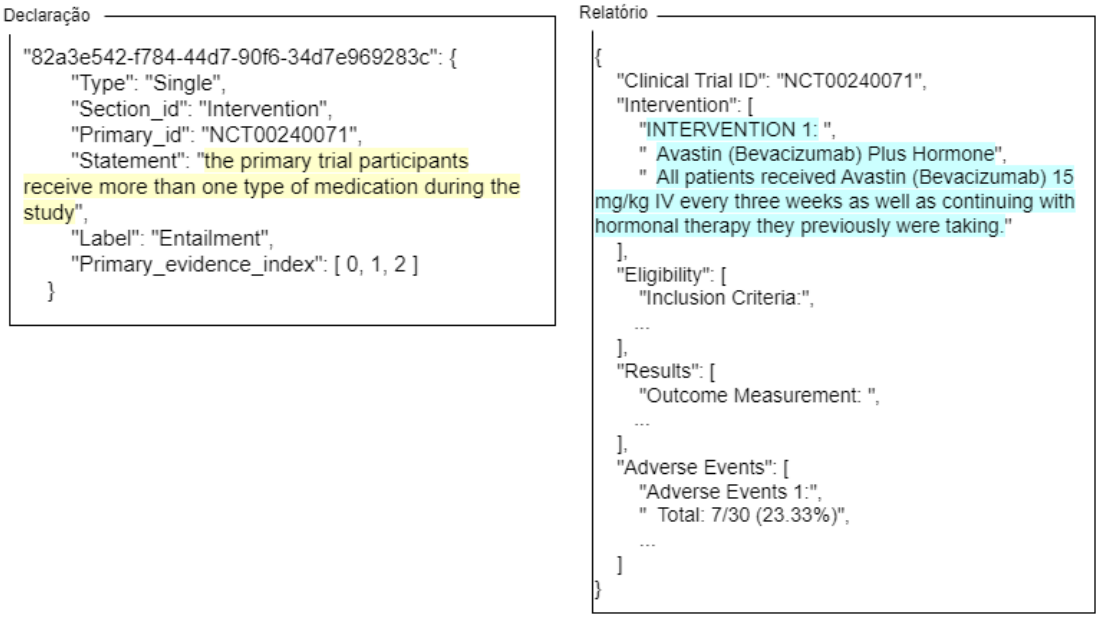
A segunda sub-tarefa da Tarefa 7 do SemEval de 2023 é a tarefa de recuperação de evidências. A recuperação de evidências é uma área de recuperação de informação (RI) e de PLN que tem como objetivo identificar e recuperar partes relevantes de um texto a partir de uma hipótese (pergunta, consulta) fornecida. Portanto, a tarefa de recuperação de evidências do SemEval de 2023 consiste em identificar e recuperar as premissas de um (ou dois) relatório(s) clínico(s) com base em uma determinada hipótese. A Figura 1.1 mostra um exemplo da tarefa de recuperação de evidências, onde as premissas que foram identificadas e recuperadas estão destacadas em azul, enquanto a hipótese está destacada em amarelo.

Visto isso, o objetivo deste trabalho é fornecer um modelo para realizar a sub-tarefa de recuperação de evidências da Tarefa 7 do SemEval de 2023, utilizando o modelo de linguagem Biomed RoBERTa e o conjunto de dados (NLI4CT) fornecido na tarefa. Visto que a competição avalia as submissões com base na métrica F1, concentramos em buscar o melhor resultado para essa métrica. Para isso, treinamos o nosso modelo várias vezes com diferentes configurações de hiperparâmetros, tais como a taxa de aprendizado e o tamanho máximo da sequência de entrada. Além disso, utilizamos diferentes métricas de otimização de treinamento, como a acurácia, revocação e F1. Posteriormente, os resultados foram avaliados em termos das métricas F1, precisão e revocação(*recall*).

Com essa abordagem, conseguimos o valor de F1 de 0.733, utilizando uma taxa de aprendizado de $5e-5$ e um tamanho máximo da sequência de entrada de 512. Realizamos o treinamento com a revocação como métrica de otimização. O resultado de F1 encontrado mostra que ainda podem ser feitas melhorias no modelo.

O restante deste trabalho está dividido em mais cinco capítulos. O Capítulo 2, apresenta a fundamentação teórica, onde descrevemos os conceitos necessários para o entendimento do trabalho. O Capítulo 3, apresenta alguns trabalhos relacionados com propostas semelhantes e que serviram de inspiração para este trabalho. O Capítulo 4 explica como foi desenvolvido o trabalho. O Capítulo 5 apresenta os resultados desse trabalho. E no Capítulo 6, apresentamos um resumo do trabalho e apontamos sugestões para trabalhos futuros.

Figura 1.1: Exemplo de funcionamento de uma tarefa de recuperação de evidências. Dada a hipótese (em amarelo) o sistema recupera as premissas do relatório de ensaio clínico (em azul).



Fonte: O Autor

2 FUNDAMENTAÇÃO TEÓRICA

O objetivo deste capítulo é apresentar os principais conceitos necessários para o entendimento do trabalho proposto. Primeiramente, na Seção 2.1 apresentamos os conceitos relacionados a redes neurais e de aprendizado profundo. Nas Seções 2.2 e 2.3, detalhamos, respectivamente, as definições de *word embeddings* e *embeddings* contextuais. Seguindo, na Seção 2.4 descrevemos o significado de modelos de linguagem, mais especificamente de modelos de linguagem generativos e discriminativos. Na Seção 2.5 introduzimos a definição de *Transformers*, visto que grande parte dos modelos citados nessa monografia apresenta este conceito. A Seção 2.6 apresenta os modelos de linguagem que utilizam como base a arquitetura *Transformer*. As Seções 2.7, 2.8 e 2.9 descrevem, respectivamente, os conceitos de recuperação evidências, ILN e o *fine-tuning*. A Seção 2.10 descreve alguns conceitos que utilizamos no treinamento do modelo.

2.1 Redes Neurais

As redes neurais pertencem a um subconjunto de algoritmos de *machine learning* (aprendizado de máquina). Aprendizado de máquina consiste na implementação de sistemas que aprendem a realizar determinadas tarefas utilizando apenas algoritmos e dados. Em aprendizado de máquina, existem diversos tipos de aprendizado diferentes, os principais estão definidos a seguir:

- **Aprendizado supervisionado**: no aprendizado supervisionado, treinamos um algoritmo a partir de dados rotulados para realizar predições ou classificações. Nesse tipo de abordagem, o algoritmo recebe dados de entrada e de saída, o algoritmo aprende a construir uma função que faz a relação entre os dados de entrada e os dados de saída. O objetivo é que o algoritmo aprenda uma função que realiza o mapeamento correto de dados novos que não têm o resultado da saída.
- **Aprendizado não supervisionado**: no aprendizado não supervisionado, o conjunto de dados fornecido para treinar um algoritmo não possui um rótulo de saída desejado, diferentemente do aprendizado supervisionado. Nesse tipo de abordagem, os algoritmos tentam encontrar padrões, agrupamentos e relações do conjunto de dados.
- **Aprendizado por reforço**: no aprendizado por reforço, um agente aprende a tomar

decisões em um ambiente de acordo com os *feedbacks* imediatos que informam a recompensa recebida pela ação. O aprendizado por reforço não recebe dados de entrada nem de saída, ele gera os próprios valores ao ir tomando decisões no ambiente. O objetivo é definir um comportamento que maximiza as recompensas recebidas pelo agente. Esse comportamento é estabelecido a partir das experiências que ele obteve ao longo do treinamento, que é onde ele tomou diversas decisões e encontrou um comportamento com maior recompensa.

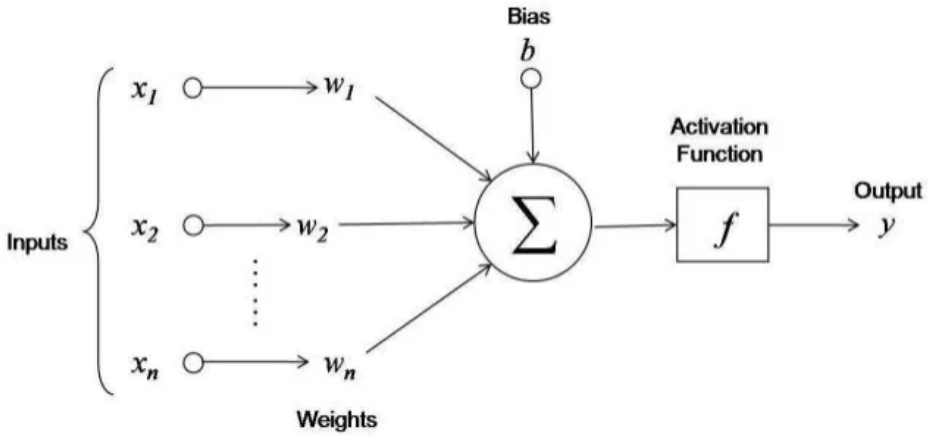
Dentre os tipos de aprendizados descritos anteriormente, as redes neurais se encaixam com os algoritmos de aprendizado supervisionado. A estrutura e o comportamento das redes neurais são inspiradas em como um cérebro humano funciona. Logo, as redes neurais são constituídos por um sistema de neurônios, que recebem um conjunto de dados de entrada e emitem um conjunto de sinais de dados de saída,

A Figura 2.1 mostra o processo de funcionamento de um neurônio. Cada neurônio da rede neural recebe um conjunto de dados como entrada e multiplica cada entrada por um conjunto de pesos correspondente. Em seguida, esses valores que foram multiplicados são somados entre si e são adicionados por uma constante chamada de *bias* (também conhecido como *threshold*) que ajuda a ativar (produzir uma saída) o neurônio. Em sequência, uma função de ativação é aplicada na soma para gerar uma saída. A função de ativação tem como objetivo produzir uma saída ou determinar o nível de ativação de um neurônio. Um exemplo de função de ativação é a função Sigmoide, que transforma a saída do neurônio em um valor entre 0 (não transmite impulso) e 1 (transmite um impulso).

A rede neural é um conjunto de neurônios interligados que propagam informações entre si. A Figura 2.2 mostra a estrutura de uma rede neural de múltiplas camadas, onde a camada N_i representa as entradas e N_o representa a camada de saída. As camadas de neurônios ficam entre as camadas N_i e N_o, elas são chamadas de camadas profundas (*hidden layers*).

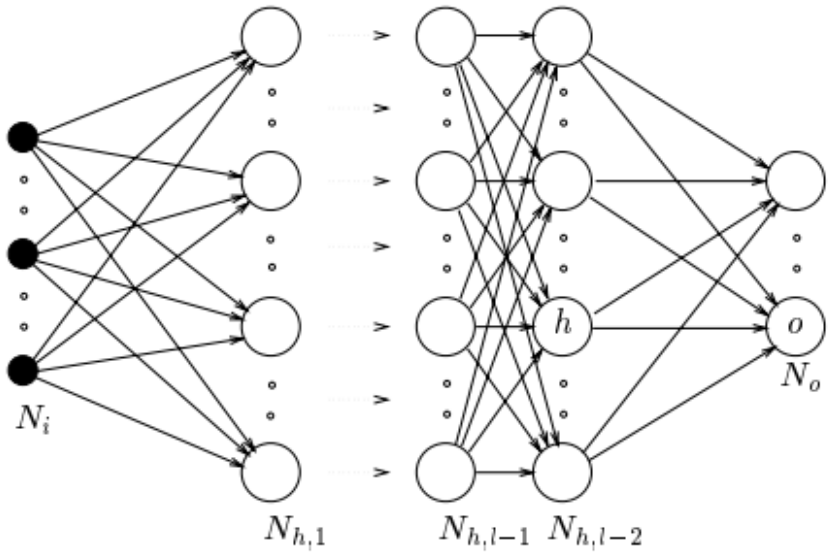
O trabalho proposto utiliza de técnicas de aprendizado profundo (*deep learning*), que é um subconjunto de aprendizado de máquina. Ele realiza o treinamento de redes neurais com várias camadas de neurônios conhecidas como camadas profundas (*hidden layers*). Em outras palavras, aprendizado profundo se refere a uma rede neural com diversas camadas profundas, aumentando assim a profundidade da rede. O uso de múltiplas camadas profundas permite que o modelo aprenda a representar as entradas de forma que facilite realizar predições dos resultados das saídas (LECUN; BENGIO; HINTON, 2015).

Figura 2.1: Exemplo de funcionamento de um neurônio. O neurônio realiza a soma ponderada com o conjunto de entrada e os pesos relacionados. Em seguida é somado um *bias*, e, em sequência, é aplicado uma função de ativação que transforma a saída do neurônio em um valor entre 0 ou 1



Fonte: Arnx (2019)

Figura 2.2: Estrutura de uma rede neural.



Fonte: Krose (1996)

2.2 *Word Embeddings*

Uma vez que computadores não compreendem textos, somente números, surgiu a necessidade de representar textos de uma forma que seja compreensível por máquinas. As *word embeddings* são um tipo de representação de palavras ou frases fundamental para (PLN). PLN é um campo da ciência da computação, mais especificamente um campo da inteligência artificial, que explora como computadores podem ser usados para aprender, compreender e produzir conteúdos em linguagem humana, tanto escrita quanto falada. Portanto, há muitas aplicações de PLN, como sistemas de pergunta e resposta, análise de sentimentos, *chatbots*, motores de busca, tradução automática e muitos outros.

A linguagem natural é a maneira com que pessoas naturalmente se comunicam com outras para expressar pensamentos, emoções e ideias. Contudo, a linguagem natural apresenta desafios que dificultam a definição de regras e de vocabulários para a linguagem humana. Alguns desses desafios é que a linguagem humana contém muita variabilidade, visto que cada idioma apresenta regras diferentes. Além disso, a linguagem natural é ambígua e dependente de contexto (HIRSCHBERG; MANNING, 2015). Isso torna a compreensão do significado de textos e das falas uma tarefa bastante complexa.

Visto isto, a busca por uma representação que capture as similaridades semânticas e sintáticas das palavras é algo muito requisitado em PLN. Para resolver esse problema de representação, uma técnica proposta é a *word embedding*. *Word embedding* normalmente realiza o mapeamento das palavras em vetores em um espaço vetorial capaz de codificar o significado semântico e sintático das palavras.

Segundo Harris (1954), palavras que ocorrem em contextos similares contêm significados similares. Logo, como os textos são representados em vetores no espaço vetorial, as palavras que contém significados similares vão estar próximas no espaço vetorial.

Contudo, um problema que as *word embeddings* apresentam, é que elas não capturam muito bem o contexto das palavras. Como por exemplo, a frase: "Ele trabalha no banco e sentou-se no banco do jardim", a palavra "banco"contém diferentes contextos, e ao realizar a *word embeddings* da frase de exemplo, o vetor correspondente à palavra "banco"irá capturar somente um dos contextos. Sendo assim, foi proposto os *embeddings* contextuais.

2.3 *Embeddings* Contextuais

Embeddings contextuais são representações de palavras que capturam o significado de uma palavra em um determinado contexto. Diferentemente dos *word embeddings*, que atribuem uma representação vetorial para cada palavra, independentemente do contexto, as *embeddings* contextuais geram vetores diferentes para cada palavra em diferentes contextos. Iremos apresentar o *BERT*, que é uma das técnicas de *embeddings* contextuais na Seção 2.6.1.

2.4 Modelos de Linguagem

Modelos de linguagem (ML) são modelos que determinam a probabilidade de uma sequência de palavras (GOODMAN, 2001). Em outras palavras, envolve realizar o treinamento de um modelo que aprende estimar a probabilidade da próxima palavra ou frase com base em um contexto especifico. Muitas aplicações de PLN utilizam esses modelos, pois eles são a base para a predições de palavras, normalmente em aplicações que geram textos, como aplicações de pergunta e resposta e de tradução automática.

Os modelos de linguagem podem ser divididos em dois tipos de métodos, os métodos probabilísticos e os métodos que são baseados em redes neurais:

- **Métodos probabilísticos**: utilizam de princípios da probabilidade para criar um modelo que identifica, a partir de predições, a próxima palavra ou sequencia de palavras de acordo com palavras anteriores do documento.
- **Métodos baseados em redes neurais**: abordagens mais modernas de modelos de linguagem utilizam redes neurais para realizar a predição da probabilidade da próxima palavra pois, modelos treinados com redes neurais conseguem resolver problemas mais complexos. Esse tipo de método permite com que os modelos de linguagem aprendam as complexidades e padrões da linguagem e capturem o significado semântico das palavras a partir dos *embeddings*, possibilitando numa melhor predição das probabilidades das palavras.

2.4.1 Modelos de Linguagem Generativos

Modelos de linguagem generativos são derivados do conceito de inteligência artificial generativa. A inteligência artificial generativa consiste em modelos e algoritmos cujo objetivo é gerar novos conteúdos, como texto, imagens, música, áudio e vídeos. Dessa forma, modelos de linguagem generativos são modelos capazes de gerar textos coerentes e contextualizados, frases por frases ou palavra por palavra.

Esses modelos são normalmente treinados com uma grande quantidade de dados específicos para uma determinada tarefa. Os modelos dependem desses dados para aprender os padrões, a estrutura e contextos para gerar textos coerentes e contextualizados na área de especialização.

2.4.2 Modelos de Linguagem Discriminativos

Modelos de linguagem discriminativos são modelos que derivam de abordagens da inteligência artificial discriminativa. Inteligência artificial discriminativa consiste em modelos que aprendem as relações e os padrões presentes no conjunto de dados para definir a classificação de uma classe. Portanto, modelos de linguagem discriminativos são modelos que realizam a tarefa de classificação. Em outras palavras, eles têm como objetivo atribuir o rótulo (classe ou categoria) mais apropriado com base em uma entrada de texto.

2.5 *Transformer*

Transformer é um tipo de arquitetura de redes neurais que foi proposta por Vaswani et al. (2017), e revolucionou o campo de PLN ganhando muita popularidade em diversas aplicações. *Transformers* podem aprender de forma eficiente a representar o significado de informações analisando grandes quantidades de dados. Além disso, os *Transformers* foram introduzidos para resolver o problema de tradução automática. Sua arquitetura segue o modelo *encoder-decoder*, onde o *encoder* gera uma sequência representativa do texto de entrada e o *decoder* gera uma sequência de texto representativa relacionada com o texto de entrada.

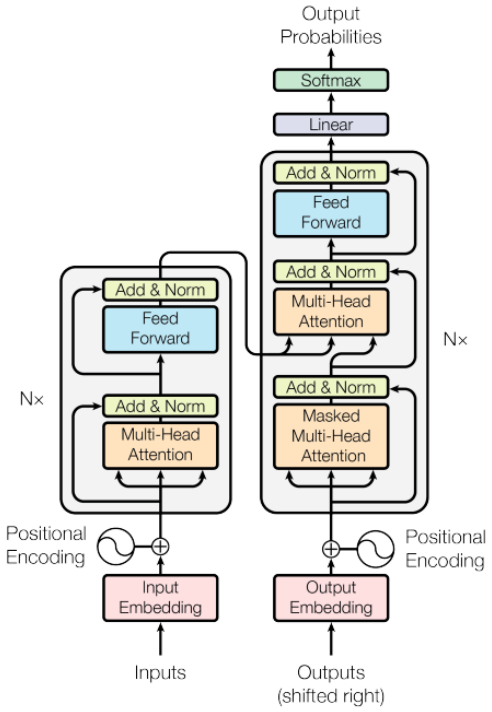
Um modelo *Transformer* consiste na arquitetura de *encoder* e *decoder* como mos-

tra a Figura 2.3. A entrada do *encoder* consiste na aplicação de *positional encoding* em uma sequência de *word embeddings*. *Positional encoding* são utilizados para informar a posição dos elementos da sequência de entrada do modelo. É necessário informar a posição de cada palavra pois os *Transformers* não possuem mecanismos que lidam com a ordem das palavras e informar a posição é importante para realizar o processamento dos dados. Os *encoders* são formados por seis camadas de pilhas idênticas para melhor representação dos dados. Cada uma dessas pilhas recebe uma sequência de *word embeddings* com a informação da posição de cada palavra da sequência de entrada.

O *encoder* é composto por dois componentes principais: o mecanismo de atenção, e o componente de propagação (*feed forward*). O mecanismo de atenção permite que o modelo calcule a relevância de diferentes elementos de uma sequência (como por exemplo, palavras em uma frase) durante o processamento de cada elemento dessa sequência. Os elementos com as maiores pontuações de relevância da sequência são considerados os mais importantes. Isso faz com que os *Transformers* possam processar sequências simultaneamente e com um custo computacional eficaz. Em outras palavras, os mecanismos de atenção servem para encontrar informações contextuais importantes de uma sequência de elementos. Depois de passar pelo mecanismo de atenção, o componente de propagação recebe como entrada os vetores de atenção e realiza o processamento de cada elemento do vetor, adicionando informações relacionadas ao posicionamento dos elementos, contribuindo para uma melhor representação contextual da sequência de entrada.

O *decoder* também é formado por seis pilhas. Cada uma delas também realiza o *positional encoding* em sequências de *word embeddings* de saída (*target*). Sendo assim, a sequência de saída com as informações da posição de cada palavra da sequência é utilizada como entrada no *decoder* que contém três componentes: um mecanismo de atenção, um mecanismo de atenção *encoder-decoder* e o componente de propagação. O primeiro mecanismo de atenção tem o mesmo propósito da camada de *encoder*, ou seja, encontrar informações contextuais importantes em toda a sequência de saída. Em seguida, no mecanismo de atenção *encoder-decoder*, é realizado o mapeamento do vetor de atenção de saída com a saída do *encoder*, que é um vetor representativo da sequência de entrada. Esse mapeamento permite que o modelo capture a relação entre a sequência de entrada e a sequência de saída. Após o mapeamento da relação da sequência de entrada e de saída, também é feita a propagação melhorando a representação contextual desse mapeamento. A saída do *decoder* é uma sequência de vetores que representam as predições ou generalizações do modelo.

Figura 2.3: Arquitetura de um *Transformer*. Na esquerda nós temos o bloco *encoder* e na direita o *decoder*



Fonte: Vaswani et al. (2017)

2.6 Modelos de linguagem baseado em *Transformers*

A área de PLN teve um grande impacto com a introdução dos *Transformers* (VASWANI et al., 2017). Esse modelo revolucionou a maneira que as máquinas entendem e geram a linguagem humana. Como descrito na Seção 2.5, os *transformers* utilizam de mecanismos de atenção para capturar o contexto e relação entre as palavras. Visto isso, foram criados modelos de linguagem que utilizam da arquitetura de *transformers* para obter um melhor entendimento e representação dos dados.

Os modelos de linguagem baseado em *Transformers* geralmente são treinados sobre grandes quantidades de dados sobre um determinado tema para obter ótimos conhecimentos contextuais e realizar as tarefas de PLN com excelência. A seguir, apresentamos alguns modelos de linguagem que foram mencionados no trabalho proposto.

2.6.1 BERT

BERT é um modelo de linguagem introduzido em 2018 pelo Google, cuja sigla significa *Bidirectional Encoder Representations from Transformers* (DEVLIN et al., 2018). O BERT é um modelo de linguagem moderno que aprende a entender e representar a linguagem humana a partir de um grande conjunto de dados. Ele é muito bom em aprender as informações contextuais e as relações das palavras, fazendo com que seja muito útil em diversas aplicações de PLN. Além disso, os *embeddings* contextuais que o BERT providencia são ricos em representações de palavras e sensitivos com o contexto das palavras.

Historicamente, os modelos de linguagem somente faziam a leitura do texto de entrada de forma unidirecional (da esquerda para direita ou da direta para a esquerda) fazendo com que os modelos de linguagem somente conseguissem o contexto de palavras futuras ou anteriores do texto. Contudo, o BERT utiliza da arquitetura *Transformer* que permite uma leitura bidirecional do texto por causa dos mecanismos de atenção. Com esses mecanismos, o BERT é capaz de realizar uma leitura completa do texto de uma vez só, o que permite uma captura melhor do contexto das palavras.

A arquitetura do BERT é baseada na arquitetura *Transformer* (Figura 2.3). No entanto, o BERT utiliza apenas a parte do *encoder* pois o objetivo do BERT é aprender o contexto dos dados de entrada e gerar representações contextuais desses dados. O *encoder* é muito bom para realizar essa tarefa pois ele é responsável por codificar as informações de contexto nas representações de palavras de forma bidirecional, permitindo que o BERT possa capturar as nuances e relações semânticas de todo o texto.

A camada do *decoder* não é utilizada pois ela é usada normalmente em tarefas que envolvem decodificação e geração de sequências de texto, como tradução automática e geração de texto. Ou seja, essa camada é utilizada para gerar sequências de saída de acordo com as sequências de entrada, o que não é o foco do BERT.

2.6.2 RoBERTa

RoBERTa é um modelo de linguagem que foi apresentado em 2019 pelo Facebook AI, o seu acrônimo significa *Robustly Optimized BERT Pretraining Approach* (LIU et al., 1907). O modelo RoBERTa é uma extensão e otimização do modelo BERT, apresentado na Seção 2.6.1. O modelo RoBERTa utiliza a mesma arquitetura que o BERT, a arquite-

tura de *encoder* do *Transformer*. No entanto, a diferença acontece durante o procedimento de treinamento.

O modelo RoBERTa é pré-treinado com um conjunto de dados muito maior que o do modelo BERT. Isso permite um melhor entendimento dos dados e do contexto deles. Além disso, o modelo BERT é treinado utilizando o *Masked Language Modeling* (MLM) e *Next Sentence Prediction* (NSP). O NSP aprende a entender as relações entra as sentenças de um texto. O objetivo principal é determinar se uma sentença de um par de sentenças é a próxima da que está sendo analisada. MLM envolve treinar um modelo para predizer as palavras que foram mascaradas aleatoriamente no texto. Isso faz com que o modelo aprenda a relação da palavra e do seu contexto.

Diferentemente do BERT, o modelo RoBERTA realiza o treinamento utilizando somente MLM, o que resulta em uma melhor captura das informações contextuais das sentenças. Além disso, o NSP adiciona uma complexidade no processo de treinamento. O treinamento do RoBERTa é realizado com sequências de texto mais longas, o que melhora a paralelização e a performance do treinamento. Sendo assim, o modelo RoBERTa melhora a qualidade de entendimento contextual do conjunto de dados e a performance do modelo em comparação com o BERT.

2.6.3 DeBERTa

DeBERTa (HE et al., 2020) é um modelo de linguagem que também é baseado no BERT. DeBERTa significa *Decoding-enhanced BERT with Disetangled Attention* e foi introduzido em 2020 pelo time de pesquisa da Microsoft. O DeBERTa apresenta uma melhor performance quando comparado aos resultados dos modelos BERT e RoBERTa. Isso se deve graças a duas novas abordagens: o mecanismo de atenção desvinculado e o segundo é o *enhanced mask decoder*.

O mecanismo de atenção desvinculado utiliza dois vetores de *embeddings* ao invés de um como é na arquitetura *transformer*. Um dos vetores de *embeddings* contém a representação do conteúdo da entrada e o segundo contém a posição relativa ao primeiro vetor. Isso permite uma melhor representação das relações sintáticas e semânticas do texto, fazendo com que o modelo DeBERTa compreenda melhor a estrutura e as nuances da linguagem.

A abordagem *enhanced mask decoder* diz que para realizar uma representação contextual precisa dos dados de entrada, é preciso utilizar as posições absolutas das pala-

vras para realizar corretamente as predições os valores mascarados do MLM. O modelo DeBERTa utiliza os *positional encoders* (descrito na Seção 2.5) somente pouco antes da função de ativação do modelo, que é onde é realizado a predição das máscaras do MLM.

Com essas novas abordagens, o modelo DeBERTa mostra que é um modelo que consegue compreender contextos complexos e capturar uma melhor representação semântica e sintática do conjunto de dados. O modelo DeBERTa apresenta uma performance superior aos outros modelos citados em diversas tarefas de PLN.

2.7 Recuperação de Evidências

Recuperação de evidências é um conceito utilizado em recuperação de informação, mais especificamente em sistemas de PLN que precisam encontrar e recuperar informações relevantes que apoiam uma determinada hipótese, consulta ou questão. Em outras palavras, a recuperação de evidências envolve o processo de identificar e recuperar partes de informação (evidências), de um conjunto de documentos, que suportam uma determinada hipótese. Recuperação de informação (RI) é uma área da computação que estuda como encontrar materiais (normalmente documentos) de dados não estruturados (geralmente textos) que satisfaçam uma necessidade de informação a partir de uma grande coleção de dados (MANNING, 2009).

Existem diversos métodos e estratégias para identificar e recuperar informações relevantes de um conjunto de dados. Esses métodos geralmente combinam técnicas de PLN e recuperação de informação para recuperar evidências relevantes. Um exemplo de abordagem moderna de recuperação de evidências é a de Soleimani, Monz and Worring (2020), que utilizou o modelo BERT pré-treinado para a tarefa de recuperação de evidências.

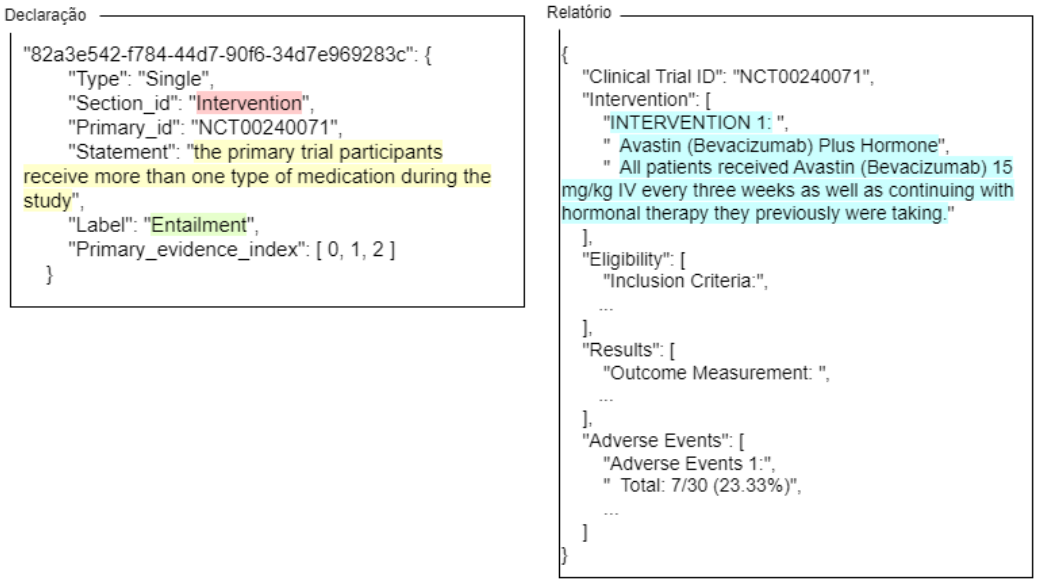
2.8 Inferência em Linguagem Natural

Inferência em linguagem natural (ILN) é a tarefa em PLN que envolve determinar se uma hipótese pode ser inferida a partir de uma premissa (MACCARTNEY, 2009). A ILN pode classificar a hipótese em três classes de acordo com a premissa: verdadeiro (*entailment*), falso (*contradiction*), e indeterminado (*neutral*). O objetivo é determinar se hipótese suporta, contradiz, ou não tem relação com a premissa de acordo com o conteúdo

do texto.

A tarefa de ILN do *workshop* SemEval de 2023 consiste na implementação de um sistema aprende a realizar a inferência entre uma hipótese e uma premissa em relatórios de ensaios clínicos. Como por exemplo, como a classe (em verde) de inferência é *entailment* o sistema deve encontrar as premissas (em azul) que suportam a hipótese (em amarelo). Caso a classe fosse *contradiction*, o sistema deveria encontrar as premissas que contradizem a hipótese.

Figura 2.4: Exemplo de funcionamento da tarefa de ILN do SemEval de 2023. As premissas (em azul) suportam (em verde) a hipótese (em amarelo). Em vermelho está a seção onde devem ser encontradas as premissas para fazer a inferência.



Fonte: O Autor

2.9 *Fine-tuning*

Fine-tuning, ou ajuste fino, é uma técnica de aprendizado de máquina muito utilizada em aprendizagem por transferência, que envolve pegar um modelo que foi pré-treinado em um grande conjunto de dados para uma tarefa específica, e treinar ele mais um pouco com um conjunto de dados menor, mas para outra tarefa diferente, porém relacionada. Aprendizagem por transferência é uma técnica de aprendizado de máquina que implica em aplicar o conhecimento aprendido em uma tarefa para melhorar a performance de uma tarefa diferente.

2.10 Treinamento

O treinamento do nosso modelo foi realizado utilizando os algoritmos de *forward propagation* e *backpropagation*. *Forward propagation* é o processo de propagar os dados de entrada do modelo através das camadas da rede neural para calcular as predições de saída. Em outras palavras, *forward propagation* calcula as predições do modelo com base no conjunto de entrada. Essas predições então são comparadas com os valores de saída verdadeiros para calcular o erro do modelo. Em seguida, após obtermos o erro do modelo, utilizamos o algoritmo de *backpropagation* para ajustar os pesos da rede neural, calculando os gradientes descendentes com base no erro de saída do modelo.

Realizamos os treinamentos do nosso modelo utilizando diferentes configurações de hiperparâmetros, como a taxa de aprendizado e o tamanho máximo da sequência de entrada. A taxa de aprendizado é um hiperparâmetro que também é utilizado para atualizar os pesos da rede neural. Ele determina a velocidade em que o algoritmo converge em direção aos pesos ideais para minimizar o erro do modelo. O tamanho máximo da sequência de entrada refere-se ao tamanho máximo que o *tokenizador* do modelo RoBERTa irá dividir o conjunto de dados de entrada em *tokens*.

O tamanho máximo da sequência de entrada consiste na sequência de entrada do *encoder* da arquitetura *Transformer*. Além disso, também utilizamos do conceito de tamanho de lote (*batch size*), que é a quantidade de exemplos de entrada e saída que são processadas em paralelo durante cada iteração de treinamento.

Para prevenir o *overfitting* do treinamento do modelo, utilizamos a regularização L1 e o otimizador AdamW (LOSHCHILOV; HUTTER, 2018). A regularização L1 é utilizada para diminuir o *overfit* encorajando o aumento da esparsidade dos pesos do modelo fazendo com que alguns deles sejam zero. Isso reduz a complexidade do modelo ao excluir características menos relevantes dos dados.

O AdamW é uma técnica usada para evitar o *overfitting*. Essa técnica envolve desacoplar o *weight decay* da atualização dos parâmetros, e adiciona um outro termo de penalização na função de erro que penaliza os valores dos pesos. Isso incentiva o modelo a ter pesos com valores menores e mais balanceados. O *weight decay* é uma técnica que desencoraja o modelo a utilizar pesos muito grandes para as características dos dados. Isso melhora a capacidade de generalização do modelo.

3 TRABALHOS RELACIONADOS

Neste capítulo, relataremos alguns dos principais trabalhos que participaram da tarefa de recuperação de evidências em relatórios de ensaios clínicos no *workshop* do SemEval de 2023. Embora tenhamos encontrado uma quantidade considerável de artigos relacionados à tarefa de recuperação de evidências no contexto do SemEval, o número de pesquisas relacionadas à recuperação de evidências na área de biomedicina ainda é relativamente baixo. No entanto, à medida que a recuperação de evidências se torna cada vez mais presente em diversas aplicações tecnológicas e com a realização de *workshops*, conferências e desafios, espera-se que a quantidade de pesquisas na área aumente.

Como podemos observar na Tabela 3.1, os trabalhos submetidos utilizaram cinco técnicas diferentes. Neste capítulo, abordaremos os trabalhos que utilizaram das três técnicas mais utilizadas: modelos generativos, modelos discriminativos e os modelos pré-treinados com dados biomédicos.

Tabela 3.1: Resumo das técnicas e modelos implementados nos trabalhos da tarefa 7 do SemEval 2023

Técnica/tipo de modelo	**Submissões #**
ML Generativo	8
ML Discriminativo	16
Baseado em ontologia	1
Baseado em regras semanticas	1
Pré-treinamento biomédico	12

Fonte: Jullien et al. (2023b)

Modelos de linguagem generativas normalmente são projetadas para aprender a distribuição de probabilidade conjunta de P(X,Y) onde X é o texto de entrada, como as premissas dos relatórios de ensaios clínicos ou consultas, e Y é a probabilidade de saída gerada após uma camada de classificação, ou um rótulo gerado por um modelo que utiliza somente o *decoder* da arquitetura *Transformer* (JULLIEN et al., 2023b). Ao analisar a Tabela 3.1, notamos que 8 participantes usam modelos generativos.

A abordagem feita por Zhou et al. (2023) foi um dos modelos generativos propostos. Eles propuseram um sistema de inferência de multi-granularidade que utiliza uma codificação conjunta da premissa e da hipótese. A granularidade se refere ao nível que é feita a *tokenização*. Ela pode ser feita em nível de palavras, sentenças, documentos, entre outros. O termo multi-granular indica que a representação foi feita usando mais de um

tipo de representação de *tokens*. Eles usaram um codificador em nível de sentença que aprende o contexto semântico da hipótese e da premissa, a seguir, é feita a codificação em nível de *token* para extrair informações de cada *tokens* da representação conjunta das premissas e das hipóteses. Por fim, o módulo de classificação indica se a sentença suporta a hipótese. Essa foi a melhor abordagem, conseguindo um resultado de F1 de 0,853.

A abordagem que obteve o segundo melhor resultado em termos de F1 foi a de Zhao et al. (2023). Eles treinaram um modelo de classificação de sentença utilizando como base o modelo de linguagem pré treinado *DeBERTaV3-large* (HE; GAO; CHEN, 2021). Especificamente, eles pré-processaram os dados disponibilizados pela tarefa 7 do SemEval e as utilizaram como entrada para o modelo *DeBERTaV3-large*. Em seguida, aplicaram uma camada linear para fazer a predição dos rótulos. Com essa abordagem eles conseguiram um valor de F1 de 0,842.

Outro trabalho pertencente ao grupo de modelos generativos, foi conduzido por Huang et al. (2023), que utilizaram um modelo baseado no GPT-2 (RADFORD et al., 2019), cuja sigla significa *Transformer* pré-treinado generativo (*generative pré-training transformer*). O GPT-2 foi usado em conjunto com uma camada linear para realizar a classificação binária. O modelo recebe como entrada cada premissa dos relatórios de ensaios clínicos junto com a sua hipótese correspondente.

Modelos de linguagem discriminativas codificam a probabilidade condicional P(Y|X) projetadas para determinar a decisão entre as classes diferentes. Esses modelos são configurados para estimar a probabilidade de uma classe Y (por exemplo, se a evidência é relevante ou não), dado uma entrada X (que pode ser hipótese e/ou os relatórios de ensaio clínicos).

A abordagem desenvolvida pelo nosso time da UFRGS (DIAS et al., 2023) é um dos trabalhos propostos com modelo discriminativo. Este modelo foi construído com base no modelo PairSCL (LI et al., 2022). Além disso, também utilizamos o modelo pré-treinado Biomed RoBERTa (GURURANGAN et al., 2020) para melhorar a performance do sistema em ambas as tarefas inferência e recuperação de evidências. O modelo Biomed RoBERTa foi utilizado para fazer com que o modelo submetido aprendesse melhor o contexto e as nuances linguísticas presentes em textos biomédicos. Sendo assim, fizemos o *fine-tuning* do PairSCL usando os dados de treinamento providenciados pelo SemEval (NLI4CT) (JULLIEN et al., 2023b), combinados com o MedNLI (ROMANOV; SHIVADE, 2018) e o MultiNLI (WILLIAMS; NANGIA; BOWMAN, 2017), atingindo os resultados de F1 de 0,681. O MedNLI e o MultiNLI são conjuntos de dados que foram

construídos para tarefas de inferência. O MedNLI contém dados biomédicos anotados por especialistas da área biomédica. E o MultiNLI é um conjunto de dados que possui de textos falados e escritos, de diferentes domínios.

Outra abordagem de modelo discriminativo foi apresentada por Vladika and Matthes (2023). Eles adotaram um sistema de *pipeline*, realizando primeiro a recuperação de evidências e, em seguida, usando os resultados como entrada para a tarefa de inferência. Além disso, desenvolveram um sistema conjunto que aprende simultaneamente a tarefa de inferência e de recuperação de evidências. Posteriormente, os dois sistemas foram combinados para melhorar a performance do sistema como em geral, essa técnica é chamada de *ensemble system* (GANAIE et al., 2022). Ambos os sistemas, (o de *pipeline* e o sistema conjunto) utilizaram de um modelo pré-treinado, e o que resultou em uma melhor performance foi o DeBERTa-V3 (HE; GAO; CHEN, 2021). Como resultado, esse trabalho teve um valor de F1 de 0.818.

Uma técnica que foi bastante utilizada nos trabalhos foi a de pré-treinamento biomédico, que envolve treinar um modelo utilizando um grande conjunto de dados biomédicos não rotulados. Alguns dos trabalhos citados anteriormente utilizam essa técnica. Esses dados são utilizados para capturar características e padrões gerais dentro do domínio biomédico antes de fazer o *fine-tuning*. Um outro trabalho que utiliza essa técnica é o (VASSILEVA et al., 2023), que faz o *fine-tuning* do modelo BioM-Bert-Large (ALROWILI; VIJAY-SHANKER, 2021) para realizar a classificação das evidências. Esse modelo utiliza dados biomédicos do *PubMed Abstracts*, *PubMed Central*, mais um vocabulário de domínio geral (livros e a *Wikipédia em inglês*).

Outra abordagem que adota dessa técnica foi a de Alameldin and Williamson (2023), que utilizou o GatorTron (YANG et al., 2022) como modelo base e também para realizar o *fine-tuning* do modelo. O GatorTron, similarmente, também utilizou de dados biomédicos, como dados os artigos *PubMed*, artigos da *Wikipédia*, e notas clínicas do Sistema de Saúde da Universidade da Florida.

De maneira geral, conseguimos ver que os participantes que utilizaram modelos generativos, adicionaram uma camada de saída e fizeram o *fine-tuning* para gerar uma probabilidade ou produziram diretamente os rótulos que dizem se é evidência ou não. Já os trabalhos com modelos discriminativos adicionam uma camada a mais às camadas de um modelo pré-treinado e realizam o *fine-tuning* para calcular a probabilidade de uma evidência ser ou não relevante.

A Tabela 3.2 mostra um resumo das técnicas utilizadas pelos participantes da com-

petição. Nela conseguimos ver que os trabalhos que implementaram modelos generativos apresentaram melhores resultados. Os participantes que implementaram modelos discriminativos usando modelos biomédicos pré-treinados, como o DeBERTa-v3-large (HE; GAO; CHEN, 2021) e o Biom-BERT-large (ALROWILI; VIJAY-SHANKER, 2021), também conseguiram resultados competitivos.

Tabela 3.2: Resumo das técnicas e modelos submetidos da tarefa 7 do SemEval 2023. (G) equivale a técnicas com modelos generativos, (D) modelos discriminativos.

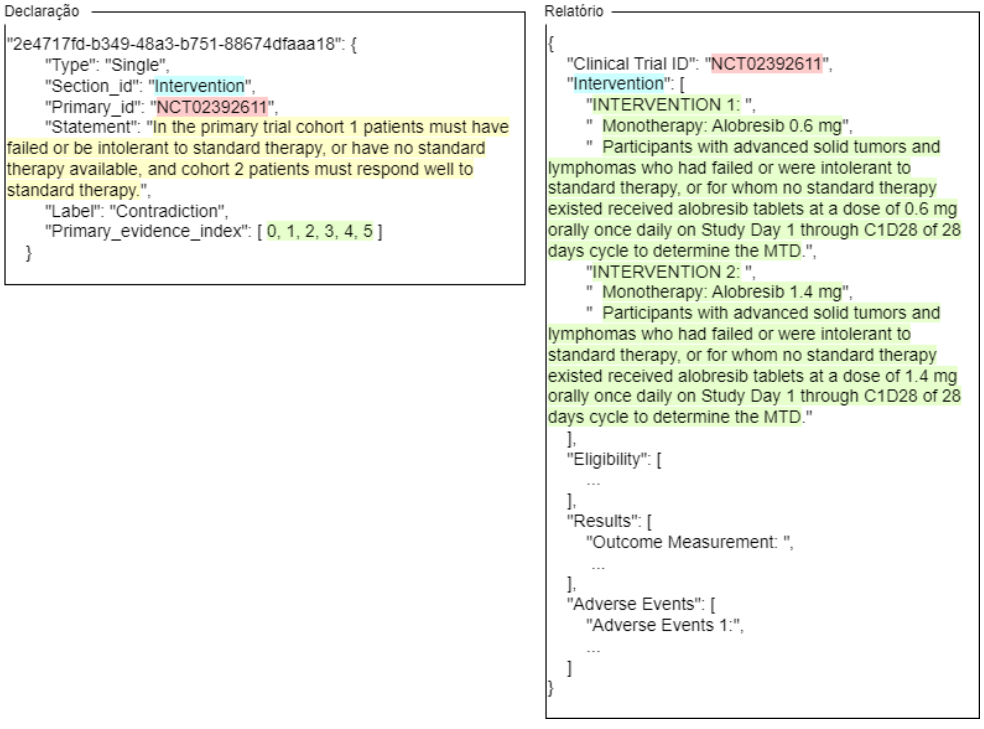
Trabalho @Nome do Time	Abordagem	Generativo/ Discriminativo	Tarefa 2		
			F1	Precisão	Revocação
Zhou et al. (2023) @THiFLY	MGNet, BiLSTM and SciFive model ensembling	G + D	0.853	0.811	0.898
Zhao et al. (2023) @HW-TSC	Zero-shot ChatGPT for entailment and DeBERTaV3 for retrieval.	G + D	0.842	0.816	0.871
Vassileva et al. (2023) @FMI-SU	Contextual Data Augmentation to fine-tuneBioM-BERT-Large	D	0.827	0.779	0.881
Vladika and Matthes (2023) @Sebis	Ensemble of a pipeline and joint system based-on DeBERTa-v3	D	0.818	0.772	0.868
Huang et al. (2023) @CPIC	Ensembled GPT-2 models with differentparameter sizes and random seeds.	G + D	0.810	0.789	0.833
Alameldin and Williamson (2023)) @Clemson NLP	GatorTron-BERT	D	0.806	0.802	0.811
Bevan, Turbitt and Aboshokor (2023) @MDC	PubMedBERT for evidence retrieval, andBio-LinkBERT classifies entailment	D	0.804	0.814	0.795
Rajamanickam and Rajaraman (2023) @I2R	Evidence level inferences with T5	G + D	0.802	0.797	0.807
Chen et al. (2023) @NCUEE-NLP	Soft voting ensemble mechanism based onBio-Link/BioBERT	D	0.794	0.803	0.786
Mahendra, Spina and Verspoor (2023) @ITTC	BM25 and Word Mover Distance	-	0.719	0.579	0.948
Dias et al. (2023) @INF-UFRGS	EvidenceSCL using a modified PairSCL model and pre-trained Biomed RoBERTa checkpoints.	D	0.681	0.615	0.764
Neves (2023) @Bf3R	Sentence-based BERT similarity model pre-trained on ClinicalBERT embeddings.	D	0.671	0.583	0.789
Mohamed and Srinivasan (2023) @SSNSheerinKavitha	Semantic Rule based Clinical Data Analysis, TF-IDF, and BM25	-	0.572	0.542	0.606

Fonte: Jullien et al. (2023b)

4 MATERIAIS E MÉTODOS

Neste trabalho, realizamos a implementação de um modelo para solucionar a tarefa de recuperação de evidências em relatórios de ensaios clínicos proposto na tarefa 7 do *workshop* do SemEval de 2023 (NLI4CT). Neste capítulo, descrevemos a implementação deste modelo que faz a recuperação das evidências nos relatórios de ensaios clínicos. Além disso, também descrevemos como foram conduzidos os experimentos e como realizamos a avaliação do trabalho proposto. Na Seção 4.1 estão detalhados o conjunto de dados disponibilizado para estudo, na Seção 4.2 mostramos como o pré-processamento desses dados é realizado, a Seção 4.3 explica a arquitetura do modelo, a Seção 4.4 descreve as métricas e métodos utilizados para avaliar o modelo, e a Seção 4.5 detalha a configuração dos experimentos realizados.

Figura 4.1: Exemplo funcionamento do processo de recuperação de evidências. Para buscar as evidências que apoiam a hipótese destacada em amarelo, o relatório de ensaio clínico correspondente ao *Primary_id* (em vermelho) da declaração é buscado. Após identificar o relatório clínico, é preciso identificar a seção no relatório que corresponde ao *Section_id* (em azul) na declaração, onde nele contém as premissas que devem ser recuperadas. O campo *Primary_evidence_index* corresponde ao índice das evidências no qual estão as premissas relevantes no relatório do ensaio clínico (em verde)



Fonte: O Autor

4.1 Conjunto de Dados

O conjunto de dados disponibilizado para a tarefa NLI4CT, é baseado em uma coleção de relatórios de ensaios clínicos de câncer de mama, que contêm rótulos anotados por especialistas da área. A Figura 4.2 mostra a estrutura de um relatório. Nele conseguimos ver que cada relatório é dividido em quatro seções, que são apresentadas a seguir:

- ***Eligibility criteria***: um conjunto de condições que permitem que pacientes participem do ensaios clínicos, como idade, gênero, histórico médico.
- ***Intervention***: informação sobre o tipo, dosagem, frequência e duração do tratamento que está sendo estudado no ensaio clínico.
- ***Results***: relata o resultado do ensaio, com dados como o número de participantes, resultados medidos, unidades e resultados do ensaio clínico.
- ***Adverse Events***: sinais e sintomas observados nos pacientes durante o ensaio clínico.

Além disso, o conjunto de dados NLI4CT (disponibilizado na tarefa 7 do *workshop* SemEval) também dispõe de um total de 2400 declarações que são correspondentes aos relatórios de ensaio clínicos, essas declarações foram escritas e anotadas por especialistas da área médica, a Figura 4.3 mostra um exemplo de duas declarações anotadas por especialistas. As declarações disponibilizadas foram divididas pelo *workshop* da seguinte maneira: 1700 declarações para o conjunto de dados de treinamento, 200 para o conjunto de validação, e 500 para o conjunto de teste.

A seguir apresentamos o significado de cada campo de uma declaração:

- **"ID"**: identificador da declaração.
- ***"Type"***: informa se as premissas que apoiam com a hipótese estão em um, ou em dois relatórios de ensaios clínicos, se o campo *"Type"* for *"Single"* significa que somente um relatório corrobora para a declaração, se for *"Comparison"*, significa que a declaração é suportada por dois relatórios.
- ***"Section_id"***: informa em qual seção do relatório está a premissa que suporta a declaração.
- ***"Primary_id"***: identificador do relatório do ensaio, ele informa em qual relatório está a premissa que suporta a declaração. Se o *"Type"* for *"Comparison"*, nós também iremos ter o ***"Secondary_id"*** que é o identificador do segundo relatório no

qual estão as premissas relevantes para a declaração.

- ***"Statement"***: declaração que precisa ser justificada. Ela também pode ser chamada por hipótese, consulta, ou pergunta.
- ***"Label"***: campo que pode conter os valores *"Entailment"* ou *"Contradiction"*, esse campo informa se as premissas contradizem a hipótese (declaração), ou suportam ela.
- ***"Primary_evidence_index"***: lista de índices que indicam onde está a premissa relevante para a declaração na seção determinada. Caso o *"Type"* for *"Comparison"*, nós iremos ter o *"Secondary_evidence_index"*, que é a lista de índices que informa as premissas relevantes do segundo relatório clínico relevante.

Figura 4.2: Exemplo de Relatório de Ensaio Clínico.


```
{
   "Clinical Trial ID": "NCT00503906",
   "Intervention": [
      "INTERVENTION 1: ",
      "  Abraxane, Avastin and Gemcitabine",
      "  Each treatment cycle is 28 days. Participants will be treated until disease progression:",
      .....
   ],
   "Eligibility": [
      "Inclusion Criteria",
      "  Patients must either be:",
      "  No previous chemotherapy regimen for metastatic breast cancer.",
      "  18 years of age or older.",
      ......
   ],
   "Results": [
      "Outcome Measurement: ",
      "  Median Progression-Free Survival",
      "  Progression-free survival will be measured from the first dose date to the earliest date of
documented evidence of progressive disease or the date of death due to any causes, whichever
occurs first.",
      ......
   ],
   "Adverse Events": [
      "Adverse Events 1:",
      "  Total: 8/29 (27.59%)",
      "  Leukopenia  [1]1/29 (3.45%)",
      ......
   ]
}
```


Fonte: O Autor

Figura 4.3: Exemplo de Declarações.


```
{
"59cd7909-00c3-4c23-9a08-a42dbc8eabdd": {
      "Type": "Single",
      "Section_id": "Intervention",
      "Primary_id": "NCT00503906",
      "Statement": "Participants of the primary trial will not receive the intervention for a set number of cycles,
the cycles will continue until disease progression",
      "Label": "Entailment",
      "Primary_evidence_index": [ 0, 1, 2, 3, 4, 5 ]
   },

"e6a4e9a6-56b8-4a30-9743-eb02688c090f": {
      "Type": "Single",
      "Section_id": "Eligibility",
      "Primary_id": "NCT00550771",
      "Statement": "A patient with a node positive T2 N2 M0 adenocarcinoma of the breast would be eligible for
the primary trial, as would a patient with a node negative adenocarcinoma of the breast with a tumor diameter
of 1.5cm",
      "Label": "Entailment",
      "Primary_evidence_index": [ 5, 6, 7, 8 ]
   },
}
```


Fonte: O Autor

4.2 Pré-Processamento dos Dados

Durante a etapa de pré-processamento dos dados, utilizamos o conjunto de dados NLI4CT (os relatórios de ensaios clínicos e as declarações) disponibilizado para criar outros três conjuntos de dados diferentes, um de treinamento, outro de validação, e um de teste. Para os três conjuntos de dados, fizemos a extração dos dados dos relatórios clínicos e das declarações.

Para realizar o pré-processamento dos conjuntos de dados de treinamento e de validação, pegamos as informações de cada declaração e de cada relatório correspondente. A Figura 4.4 mostra um exemplo dos campos de uma declaração e do seu relatório correspondente, além da relação entre os seus campos.

A correspondência entre a declaração e o relatório é estabelecida através do campo *"Primary_id"* da declaração e do identificador do relatório clínico (em vermelho na Figura 4.4). Uma vez que identificado o relatório, mapeamos cada seção e suas premissas correspondentes, classificando-as de acordo com a sua relevância (se são ou não evidências). Isso é feito verificando os campos *"Section_id"* e o *"Primary_evidence_index"* da declaração com as seções e as premissas do relatório correspondente.

Na Figura 4.4, a seção onde estão localizadas as premissas relevantes está destacada em azul, enquanto as premissas que são relevantes (as evidências) estão em verde. A determinação da relevância das premissas do relatório envolve verificar se a seção coincide com o campo da declaração *"Section_id"*. Caso a seção for a mesma, então examinamos se o índice de cada premissa (como por exemplo, a premissa *"Adverse Events*

1", em verde na Figura 4.4, corresponde à premissa de índice 0) está presente no campo *"Primary_evidence_index"* da declaração. Caso estiver, essa premissa é considerada uma evidência, caso não estiver ela não é uma evidência. Logo, todas as premissas que não estão na seção *"Section_id"* não serão classificadas como evidência.

O pré-processamento do conjunto de dados de teste é praticamente igual ao dos outros dois conjuntos. Contudo, os dados de teste não possuem rótulos, as declarações não contêm os campos *"Primary_evidence_index"*, *"Secondary_evidence_index"*, e *"Label"*, logo, nós não podemos rotular as premissas como evidências ou não evidências nesse caso.

Figura 4.4: Exemplo da relação entre uma declaração e um relatório.

Declaração


```
"c7dc582a-4a58-4cb5-b980-b2d2b6a128c3": {
      "Type": "Single",
      "Section_id": "Adverse Events",
      "Primary_id": "NCT00331630",
      "Statement": "no cardiac or neural adverse events
were recorded in the primary trial",
      "Label": "Entailment",
      "Primary_evidence_index": [ 0, 1, 2, 3 ]
  }
```


Relatório


```
{
   "Clinical Trial ID": "NCT00331630",
   "Intervention": [
      "INTERVENTION 1: ",
      "  Treatment With Lapatinib and Abraxane",
      ......
   ],
   "Eligibility": [
      "DISEASE CHARACTERISTICS:",
      "  Histologically confirmed breast cancer",
      "  Clinical stage I-III disease",
      ......
   ],
   "Results": [
      "Outcome Measurement: ",
      "  Clinical Response Rate (cRR)",
      ......
   ],
   "Adverse Events": [
      "Adverse Events 1:",
      "  Total: 2/30 (6.67%)",
      "  Dry Eyes  [1]1/30 (3.33%)",
      "  Diarrhea  [2]1/30 (3.33%)"
   ]
}
```


Fonte: O Autor

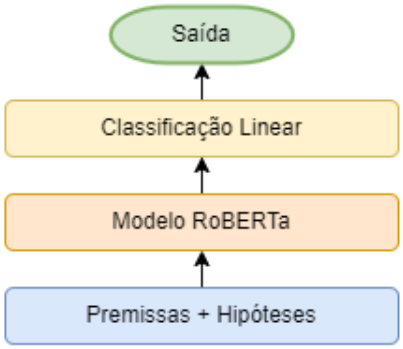
4.3 Arquitetura do Modelo

O modelo proposto possui quatro componentes principais: a entrada do sistema (que consiste nas premissas e nas hipóteses), uma camada que consiste no modelo de linguagem Biomed RoBERTa, uma camada de classificação linear, e a saída do modelo. A Figura 4.5 representa graficamente essa arquitetura.

A primeira camada consiste na *tokenização* dos pares de entrada (premissa e hipótese) utilizando o *tokenizador* RoBERTa. Realizamos a *tokenização* do par (premissa e hipótese) em nível de sentença para obter uma melhor representação dos dados, per-

mitindo que o modelo compreenda e processe os dados textuais de uma maneira eficaz. A *tokenização* transforma o texto em representações numéricas e em um formato consistente, permitindo que o modelo compreenda e processe os dados de maneira mais eficiente. A *tokenização* também permite com que o modelo capture nuances linguísticas, compreenda o contexto e processe os dados de maneira mais eficaz. Basicamente, ele prepara os dados de texto para serem processados pelo modelo.

Figura 4.5: Arquitetura do sistema.



Fonte: O Autor

A segunda camada é a do modelo *Biomed RoBERTa* (GURURANGAN et al., 2020), que é uma variante do modelo RoBERTa (LIU et al., 1907). O modelo *Biomed RoBERTa* foi treinado especificamente utilizando textos biomédicos, fazendo com que o modelo tenha um melhor entendimento de nuances da linguagem utilizada em documentos biomédicos. Posto isto, essa camada recebe como entrada os dados *tokenizados*, e codifica as sentenças *tokenizadas* em *embeddings* contextuais que capturam a relação semântica entre as palavras das sentenças. *Embeddings* é uma forma de representar os dados textuais (como frases, sentenças, documentos) em vetores que contêm as relações semânticas e contextuais dos dados.

A terceira camada consiste na componente de classificação linear, que recebe como entrada os *embeddings* da camada Biomed RoBERTa e realiza a classificação linear desses dados para predizer a relevância da evidência. A camada de classificação linear utiliza os *embeddings* contextualizados (Seção 2.3) da camada Biomed RoBERTa como entrada e executa uma transformação linear sobre eles. Essa transformação envolve multiplicar os *embeddings* contextualizados (variável "x"na equação) por uma matriz de pesos (variável "A") adicionando uma matriz que corresponde ao viés (variável "b"). O resultado é uma representação linear conhecida como *logits* que são os vetores não nor-

malizados com a representação das predições de relevância de cada premissa e hipótese correspondentes. A seguinte equação representa a transformação linear utilizada:

$$y = x * A^T + b$$

Em seguida, os *logits* (os resultados da classificação linear) são utilizados na função de ativação que gera um vetor (normalizado) com as probabilidades das predições. Após, utilizamos a entropia cruzada para realizar o cálculo do erro entre as predições e os rótulos verdadeiros de cada evidência. A seguir, a Equação 4.1 mostra como é feita a entropia cruzada. Na sequência, apresentamos a Equação 4.2 que descreve a função de ativação *softmax* utilizada.

$$L_n = -\sum_{n=1}^{N} y_n \log(f(n)) \tag{4.1}$$

$$f(n) = \frac{exp(x_{n,y_n})}{\sum_{i=1}^{N} exp(x_{n,i})} \tag{4.2}$$

Onde N é o número total de classes, f(n) é a Equação 4.2 que representa a função de ativação, y são os rótulos de saída verdadeiros, e x são os *logits* com a representação das entradas preditas.

A Equação 4.1 realiza o cálculo do erro pela entropia cruzada. A entropia cruzada é utilizada para medir o quão diferente são as probabilidades preditas pelo modelo e as probabilidades verdadeiras das classes. Em outras palavras, ela avalia o quão bem o modelo está prevendo as evidências. Caso a probabilidade predita fique muito próxima das probabilidades, a entropia cruzada será baixa, indicando que o modelo está com um bom desempenho. Além disso, ela serve como uma maneira de otimizar o modelo ao realizar o ajuste dos pesos durante o treinamento para minimizar a entropia cruzada. Isso significa que o modelo é otimizado para realizar predições mais próximas das probabilidades verdadeiras.

Para obter as predições das evidências, utilizamos a função de ativação *softmax* (Equação 4.2), que são as probabilidades preditas de cada classe, 1 ou 0. Quanto mais perto de 1, significa que a relação entre a premissa e a hipótese é relevante, quanto mais perto de 0 significa que não é relevante.

4.4 Avaliação

Durante a etapa de treinamento e de validação do modelo, realizamos a avaliação de acordo com a métrica que está sendo otimizada. Como por exemplo, caso o treinamento esteja utilizando como otimização a acurácia, a avaliação do modelo durante o treinamento será feito pela acurácia. Logo, a avaliação do modelo durante treinamento e a validação é feito pelas métricas acurácia, revocação e F1.

A tarefa de recuperação de evidências do *workshop* do SemEval avalia os sistemas dos participantes da competição a partir das métricas de precisão, revocação, e F1. As abordagens dos participantes são classificados (ranqueados) a partir do valor da métrica F1. Sendo assim, o nosso modelo também é avaliado utilizando as métricas precisão, revocação e F1. As métricas de precisão e revocação são calculadas a partir de termos que descrevem os resultados da classificação, ou seja, a partir dos classes recuperadas e das classes corretas.

Figura 4.6: Matriz de Confusão.

		Classe Predita	
		+	-
Classe Verdadeira	+	**VP**	**FN**
	-	**FP**	**VN**

Fonte: O Autor

A Figura 4.6 mostra uma matriz de confusão onde pode ser observado os termos que descrevem os resultados da classificação. O termo verdadeiro positivo (VP) corresponde às evidências que foram recuperadas como relevantes e que a classe verdadeira é relevante. O verdadeiro negativo (VN) ocorre quando as evidências recuperadas não são relevantes, e a classe verdadeira da evidência também não é relevante. O falso positivo (FP) é quando a evidência recuperada é relevante, mas a classe verdadeira da evidência não é relevante. E o falso negativo (FN) é quando a evidência recuperada não é relevante, mas a classe verdadeira é relevante. A seguir apresentamos como são calculadas as métricas as que foram utilizadas para avaliar o modelo:

- **Acurácia**: indica a proporção das evidências que foram classificadas (ou preditas) corretamente do número total de evidências. Uma acurácia alta indica que o modelo é bom ao realizar predições sobre o que é uma evidência ou não. Calculamos a acurácia da seguinte maneira:

$$ACC = \frac{VP + VN}{VP + VN + FP + FN}$$

- **Precisão**: essa métrica indica a proporção das evidências relevantes recuperadas do total de evidências recuperadas. Uma precisão alta significa que quando o modelo recupera uma evidência relevante, normalmente ele está correto. Calculamos a precisão da seguinte maneira:

$$P = \frac{\mid VP \mid}{\mid VP \mid + \mid FP \mid}$$

- **Revocação**: indica a proporção de evidências relevantes recuperadas do total de evidências relevantes. Uma revocação alta significa que o modelo é muito bom em recuperar todas as evidências relevantes no relatório clínico, sem perder nenhuma evidência relevante na recuperação. Calculamos a revocação da seguinte maneira:

$$R = \frac{\mid VP \mid}{\mid VP \mid + \mid FN \mid}$$

- ***F1-Score***: é a média harmônica da precisão e da revocação. Ou seja, um modelo com alto valor de F1 é capaz de dizer se uma evidência é ou não relevante, ao mesmo tempo que consegue recuperar um grande número de evidências relevantes. Essa métrica é calculada da seguinte maneira:

$$F1 = \frac{2 \cdot P \cdot R}{P + R}$$

4.5 Configuração dos Experimentos

Com o objetivo de estabelecer o melhor modelo possível para a tarefa de recuperação de evidências em relatórios de ensaios clínicos, utilizamos diferentes configurações de experimentos. Em outras palavras, conduzimos os experimentos utilizando diferentes configurações de hiperparâmetros (taxa de aprendizado e o tamanho máximo da sequência de entrada) e diferentes métricas de otimização (acurácia, revocação e a F1).

Para realizar esses experimentos, fizemos o *fine-tuning* do modelo pré-treinado Biomed RoBERTa (GURURANGAN et al., 2020) com todos os conjuntos de dados NLI4CT pré-processados para realizar a tarefa de classificação proposta (recuperação de evidências). O *fine-tuning* foi realizado por 10 épocas utilizando o otimizador AdamW (LOSHCHILOV; HUTTER, 2018). Escolhemos o modelo Biomed RoBERTa devido a ele ser treinado em um extenso conjunto de dados biomédicos, permitindo que, no treinamento, o modelo aprenda a representar o contexto dos dados biomédicos e aprimore a classificação das evidências.

Nossos experimentos foram projetados para buscar a configuração ideal de hiperparâmetros, explorando variações nas taxas de aprendizado e no *maximum sequence length* (que é o tamanho máximo que a sequência de entrada irá ter após a *tokenização*). As taxas de aprendizado utilizadas foram $1e-5$, $3e-5$ e $5e-5$, enquanto para o tamanho máximo da sequência de entrada, utilizamos os valores 128 e 512. Utilizamos um tamanho do lote (*batch size*) de 16 e adicionamos uma regularização L1 de 0.1 para diminuir o *overfitting*. Além disso, também utilizamos um *weight decay* de 0.01 para prevenir o *overfitting*. Utilizamos o *cosine annealing scheduler* que é um agendador que atualiza a taxa de aprendizado de acordo com os valores da curva do cosseno. Isso significa que a taxa de aprendizado diminui gradualmente seguindo a curva, e após um ciclo do cosseno, a taxa de aprendizado aumenta novamente. Essa técnica é utilizada para prevenir que a aprendizagem fique presa em um valor mínimo e permite uma melhor capacidade de aprendizado do modelo.

Para definir as configurações ideais da taxa de aprendizado e do tamanho máximo da sequência de entrada, realizamos uma série de treinamentos utilizando a acurácia como métrica de otimização. Isso implica, que durante o treinamento, o modelo aprende a resolver o problema de recuperação de evidência, sempre buscando melhorar a sua acurácia. Após obtermos a melhor configuração de hiperparâmetros do modelo, realizamos um novo treinamento utilizando como otimização as métricas F1 e revocação, utilizando o melhor conjunto de hiperparâmetros obtidos.

5 RESULTADOS

Neste capítulo apresentamos os resultados que foram obtidos a partir da configuração dos experimentos propostos na Seção 4.5. Em conjunto com os resultados, apresentamos como os experimentos foram conduzidos, realizamos a análise dos resultados obtidos de cada experimento e também realizaremos uma comparação com os resultados da abordagem que o nosso time da UFRGS obteve ao participar do tarefa de recuperação de evidências do SemEval de 2023 (DIAS et al., 2023).

Conduzimos os experimentos utilizando diferentes tipos de configurações de hiperparâmetros e de métricas de otimização para avaliar a performance do modelo proposto. As métricas utilizadas para avaliar o modelo foram a precisão, revocação, e a F1 *score*. A métrica F1 é utilizada para ranquear os sistemas submetidos na tarefa de recuperação de evidências do SemEval de 2023. A Tabela 5.1 mostra os resultados do modelo no qual utilizamos a acurácia como métrica de otimização. A tabela mostra que o melhor resultado foi obtido com a taxa de aprendizado de $5e-5$ e com o tamanho máximo de entrada de 128.

Tabela 5.1: Resultados do modelo utilizando como otimização a acurácia.

taxa de aprendizado	tam_max_entrada	F1(validação/teste)	precisão(validação/teste)	revocação(validação/teste)
1,00E-05	128	0,732/0,696	0,884/0,845	0,782/0,591
3,00E-05	128	0,719 /0,665	0,877/0,828	0,609/0,555
5,00E-05	128	0,745/**0,706**	0,816/0,803	0,685/**0,630**
1,00E-05	512	0,740/0,692	0,821/0,808	0,674/0,606
3,00E-05	512	0,712/0,653	0,882/0,805	0,598/0,549
5,00E-05	512	0,738/0,652	0,918/**0,885**	0,617/0,517

Fonte: O Autor

Ao analisar os resultados da Tabela 5.1, observamos que os escores para a revocação estavam baixos, enquanto a precisão era alta. Como nosso objetivo era melhorar os resultados de F1, identificamos a necessidade de melhorar a revocação. Visto que a métrica F1 é uma média harmônica da precisão e da revocação, decidimos realizar um novo treinamento do modelo utilizando como métrica de otimização a revocação e a F1.

Após analisar os resultados dos experimentos que utilizaram como métrica de otimização a acurácia para descobrir a melhor configuração de hiperparâmetros, utilizamos a taxa de aprendizado descoberta ($5e-5$) para realizar o treinamento utilizando as métricas revocação e F1 como métricas de otimização. Além disso, esses treinamentos foram realizados utilizando os dois tamanhos máximos da sequência de entrada para verificar se a otimização com as outras métricas também é melhor com o tamanho da sequência de 128.

Tabela 5.2: Resultados do modelo utilizando como otimização a métrica F1.

taxa de aprendizado	tam_max_entrada	F1(validação/teste)	precisão(validação/teste)	revocação(validação/teste)
5,00E-05	128	0,757/**0,684**	0,850/**0,848**	0,683/0,573
5,00E-05	512	0,718/0,680	00,833/0,820	0,632/**0,581**

Fonte: O Autor

Tabela 5.3: Resultados do modelo utilizando como otimização a métrica revocação.

taxa de aprendizado	tam_max_entrada	F1(validação/teste)	precisão(validação/teste)	revocação(validação/teste)
5,00E-05	128	0,727/0,685	0,822/**0,829**	0,652/0,583
5,00E-05	512	0,763/**0,733**	0,734/0,740	0,796/**0,726**

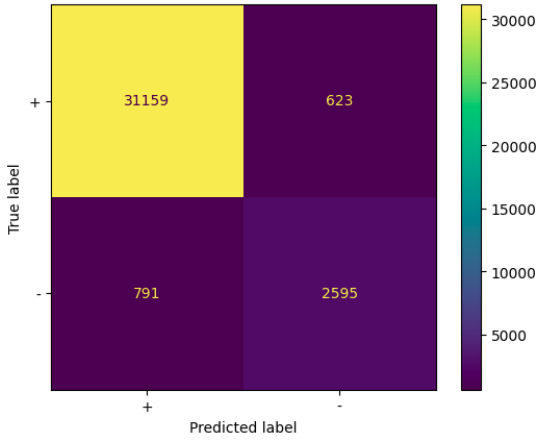
Fonte: O Autor

As Tabelas 5.2 e 5.3 mostram os resultados obtidos dos treinamentos utilizando como otimização as métricas F1 e revocação, respectivamente. Ao analisar os resultados obtidos dos modelos treinados com todas as métricas de otimização, observamos que o melhor resultado de F1 foi obtido ao realizar a otimização com a revocação. A melhora do resultado de F1 ocorreu devido ao balanceamento dos resultados da precisão e da revocação, a discrepância entre essa duas métricas ficou muito menor em relação às outras duas métricas de otimização utilizadas. Além disso, o tamanho máximo da sequência de entrada que obteve o melhor resultado foi de 512. Ao analisar os resultados da Tabela 5.3 que utilizou da revocação como métrica de otimização, conseguimos ver que os resultados das avaliações que levam em conta a revocação são melhores com o tamanho da sequência de 512.

Além disso, podemos fazer uma análise dos termos das matrizes de confusão das Figuras 5.1 e 5.2. Esse tipo de análise nos ajuda a entender melhor como os valores de precisão, revocação e F1 se alteram de acordo com a natureza da otimização do treino do modelo. Em outras palavras, essa análise nos ajuda a ter uma noção mais clara do que esperar dos resultados obtidos nesta pesquisa. Um ponto importante a ser ressaltado é que os valores das matrizes de confusão 5.1 e 5.2 são referentes aos dados brutos de classificação dos modelos. Ou seja, eles não representam os valores contidos nas tabelas dos resultados das avaliações pois eles não são os dados que foram formatados para a avaliação na plataforma da competição do SemEval. Sendo assim, podemos usá-las para realizar uma análise comparativa do comportamento dos modelos durante a otimização.

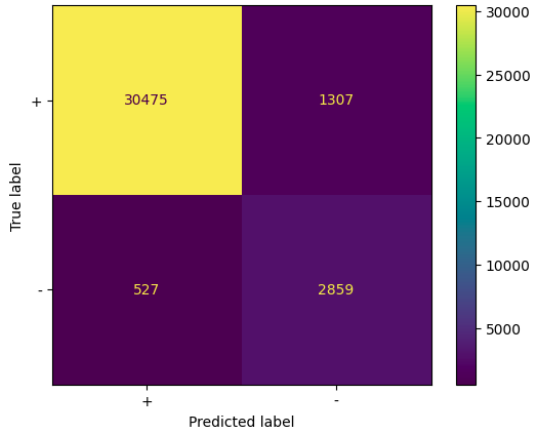
Observamos que o número de FP do modelo otimizado com a métrica revocação aumentou quando comparado com o número de FP do modelo otimizado com a acurácia. Consequentemente, a precisão do modelo otimizado pela revocação diminuiu. Ainda assim, houve uma melhora no valor de F1 pois a combinação das duas métricas (revocação e precisão) ficou mais harmonizada.

Figura 5.1: Matriz de confusão do modelo otimizado com a revocação, com tamanho da sequência de entrada 512, taxa de aprendizado $5e^{-}5$ utilizando o conjunto de dados de validação. A matriz de confusão utiliza os dados brutos da classificação.



Fonte: O Autor

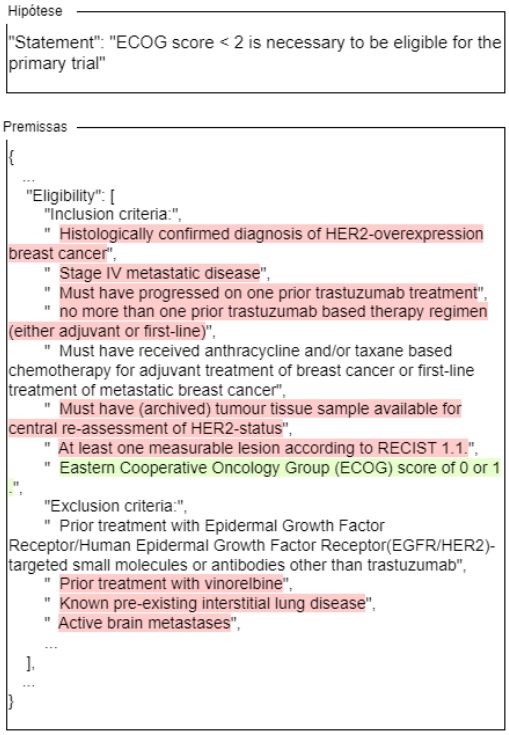
Figura 5.2: Matriz de confusão do modelo otimizado com a acurácia, tamanho da sequência de entrada 128, taxa de aprendizado $5e^{-}5$ utilizando o conjunto de dados de validação. A matriz de confusão utiliza os dados brutos da classificação.



Fonte: O Autor

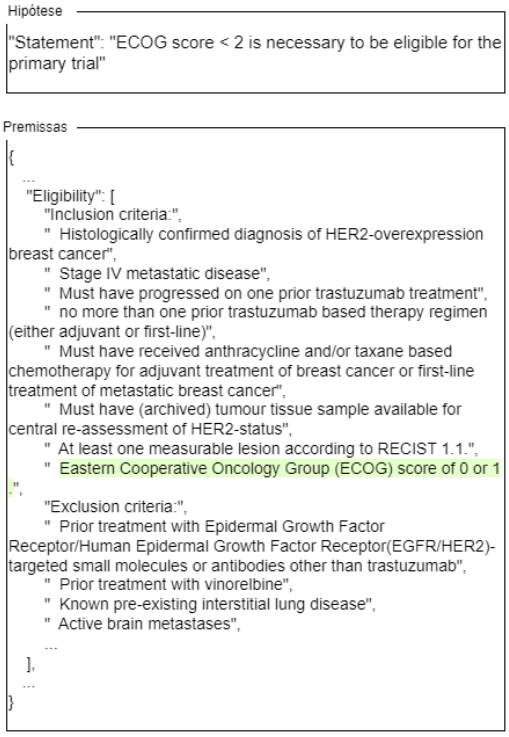
A seguir são mostrados dois exemplos de recuperações de evidências obtidos pelo modelo quando o treinamento foi realizado com as métricas de otimização revocação (Figura 5.3) e acurácia (Figura 5.4). Observamos que o exemplo da revocação recupera um alto número de evidências FP (veja as premissas destacadas em vermelho na Figura 5.3) pois o seu treinamento tem como objetivo recuperar todas as evidências relevantes no relatório clínico, sem perder nenhuma evidência relevante na recuperação. No entanto, ao realizar o treinamento utilizando a acurácia como métrica de otimização, o modelo é muito bom em dizer se uma evidência é relevante ou não. Isso acontece pois o seu treinamento tem como objetivo melhorar a capacidade do modelo dizer se uma premissa é ou não relevante. Observe que a Figura 5.4 não apresenta FP.

Figura 5.3: Exemplo da recuperação de evidências que o modelo realizou ao ser otimizado com a revocação. Destacado em vermelho temos as premissas FP, e destacado em verde a premissa VP, ou seja, a evidência recuperada corretamente.



Fonte: O Autor

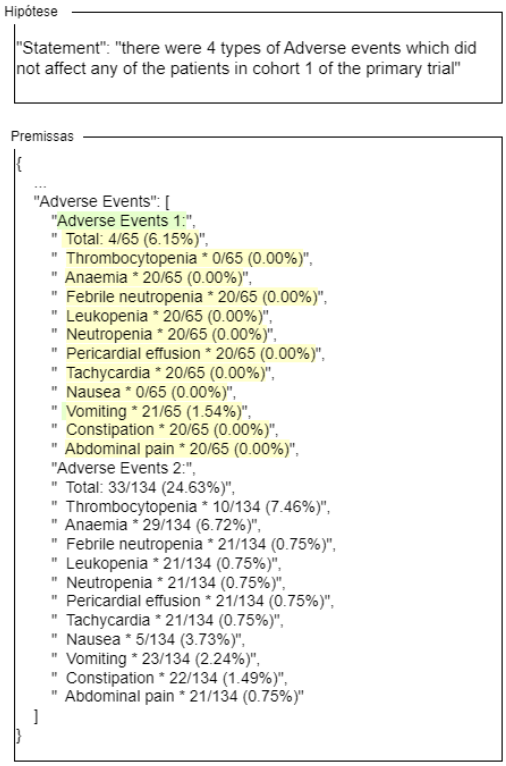
Figura 5.4: Exemplo da recuperação de evidências que o modelo realizou ao ser otimizado com a acurácia. Destacado em verde está a premissa recuperada corretamente (VP).



Fonte: O Autor

Seguindo essa análise, podemos observar que modelos otimizados pela acurácia tendem a aumentar os FN. Observe como o número de FN aumenta na matriz de confusão do modelo otimizado pela acurácia (Figura 5.2) em relação à matriz de confusão do modelo otimizado pela revocação (Figura 5.1). A Figura 5.5 mostra um exemplo de FN (destacados em amarelo) classificado em um modelo otimizado pela acurácia. Podemos ver um comportamento diferente em modelos otimizados pela revocação, onde são classificados menos FN (Figura 5.6).

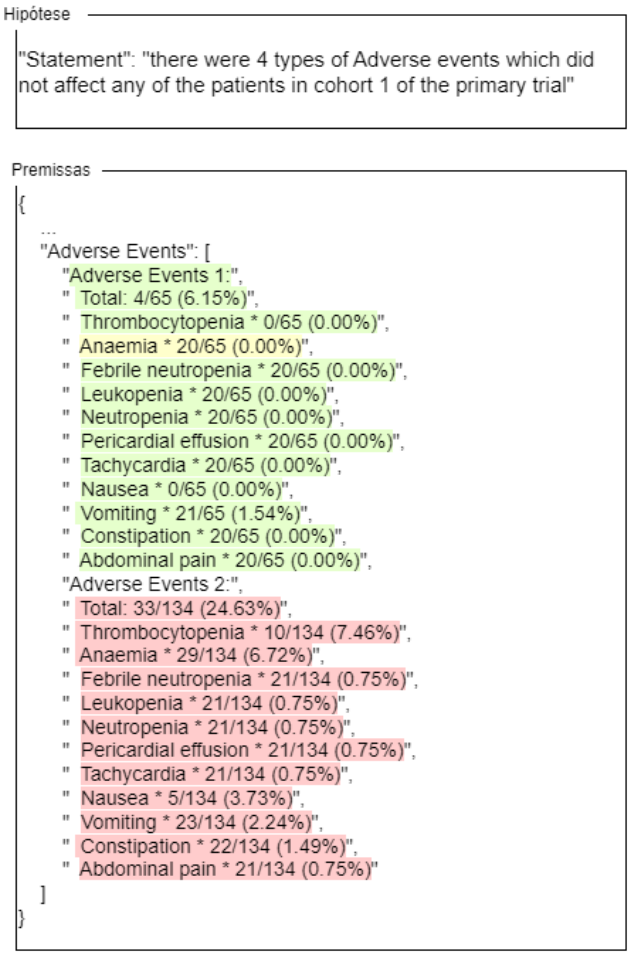
Figura 5.5: Exemplo da recuperação de evidências que o modelo realizou ao ser otimizado com a acurácia. Destacado em amarelo temos as premissas FN, e destacado em verde a premissa VP, ou seja, a evidência recuperada corretamente.



Fonte: O Autor

A Tabela 5.4 mostra o melhor resultado obtido pelo nosso modelo e os resultados dos experimentos realizados pelo time da UFRGS durante a competição. O EvidenceSCL-2L é o modelo que foi utilizado para realizar a tarefa de ILN do SemEval de 2023. Empregamos a técnica de transferência de aprendizagem para transformar o modelo da tarefa de ILN em um modelo capaz de realizar a tarefa de recuperação de evidências. As abordagens EvidenceSCL-3L e PairSCL-3L podem ser usadas para as duas tarefas. Os detalhes contextuais de cada uma das abordagens empregadas pelo time da UFRGS está detalhada em (DIAS et al., 2023). Também realizamos a comparação com o resultado do Okapi

Figura 5.6: Exemplo da recuperação de evidências que o modelo realizou ao ser otimizado com a revocação. Destacado em amarelo temos a premissa FN, e destacado em verde a premissas VP, ou seja, as evidências recuperadas corretamente.



Fonte: O Autor

BM25, que foi disponibilizado na Tarefa 7 do SemEval no *starter script*[1].

Ao comparar os resultados da Tabela 5.4 conseguimos ver que a nossa abordagem superou os resultados obtidos dos experimentos realizados pelo nosso time da UFRGS e pelos resultados obtidos pelo *starter script*. Contudo, ao comparar com os resultados obtidos na Tabela 3.2, que são as submissões feitas pelos outros times da competição, conseguimos ver que o nosso resultado ainda está abaixo do ideal.

Por fim, após analisar todos os resultados, observamos que o modelo proposto superou os resultados obtidos pelo time da UFRGS durante a competição. No entanto, é importante notar que os resultados do modelo proposto ainda não alcançaram um nível competitivo. Isso indica que há espaço para a investigação de novos métodos que pos-

[1]Starter script: <https://sites.google.com/view/nli4ct/semeval-2023/get-data-and-starting-kit>

Tabela 5.4: Em negrito, evidenciamos o nosso melhor resultado e os resultados da implementação do time da UFRGS na tarefa de recuperação de evidências do SemEval de 2023 com os conjuntos de dados de validação/teste. Os números entre parenteses indicam a posição que ficamos na competição da tarefa de recuperação de evidências do SemEval de 2023. Os resultados obtidos pelo modelo proposto ocorreu depois da competição está identificado pelo *.

	F1	**Precisão**	**Revocação**
Okapi BM25	0,322 / 0,350	0,422 / 0,469	0,261 / 0,279
EvidenceSCL-2L	0,211 / **0,681 (21)**	0,641 / **0,615 (19)**	0,126 / **0.764 (21)**
EvidenceSCL-3L	0,839 / 0,610	0,907 / 0,517	0,782 / 0,743
PairSCL-3L	0.660	0,520	0,903
Biomed RoBERTa*	0,763 / **0,733 (19)**	0,734 / **0,740 (17)**	0,796 / **0,726 (22)**

Fonte: O Autor

sam melhorar o desempenho. Melhorias futuras podem incluir o treinamento de modelos generativos, ou até o *fine-tuning* de modelos biomédicos pré-treinados usando um conjunto de dados ainda mais extenso do que o modelo Biomed RoBERTa (utilizado nesse trabalho).

6 CONCLUSÃO

Este trabalho propõe uma solução para a tarefa de recuperação de evidências do *workshop* SemEval de 2023. Desenvolvemos um modelo que realiza a recuperação de evidências em relatórios de ensaios clínicos, utilizando o modelo Biomed RoBERTa pré-treinado com dados da área biomédica. Nossa abordagem envolveu realizar uma serie de treinamentos variando as métricas de otimização (acurácia, revocação e F1) e também variando a configuração de dois hiperparâmetros (taxa de aprendizado e tamanho). Essa serie de treinamentos foi conduzida com o objetivo de encontrar a configuração que resultasse no melhor desempenho em relação à métrica F1-*score*.

A taxa de aprendizado que resultou no melhor valor da métrica F1 foi de $5e-5$. Além disso, conseguimos observar que realizar o treinamento utilizando a acurácia como otimização é mais eficaz com o tamanho máximo da sequência de entrada de 128. Contudo, quando realizamos o treinamento utilizando a revocação como métrica de otimização, notamos que é melhor utilizar o tamanho máximo da sequência de 512,

O melhor valor de F1 do modelo foi obtido ao realizar o treinamento que emprega a revocação como métrica de otimização. Além disso, a configuração de hiperparâmetros que resultou no melhor desempenho foi a que possuía a taxa de aprendizado de $5e-5$ e um tamanho máximo da sequência de entrada de 512. Com isso, conseguimos superar os resultados alcançados pela implementação *EvidenceSCL-2L* da equipe da UFRGS, obtendo um valor de F1 de 0,733, o que seria equivalente ao 19º lugar na competição.

Os experimentos que realizamos mostraram que ainda temos oportunidades para melhorias. Futuros experimentos podem envolver o uso de modelos generativos para realizar a tarefa de recuperação de evidências, ou realizar o *fine-tuning* de modelos pré-treinados com dados biomédicos em um conjunto de dados muito maior do que utilizamos no trabalho proposto. A tarefa de inferência de linguagem natural permaneceu presente no *workshop* SemEval de 2024, sugerindo que é um tópico relevante e que ainda está em desenvolvimento, com oportunidades para melhorias onde se buscam resultados mais significativos.

REFERÊNCIAS

ALAMELDIN, A.; WILLIAMSON, A. Clemson nlp at semeval-2023 task 7: Applying gatortron to multi-evidence clinical nli. In: **Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023)**. [S.l.: s.n.], 2023. p. 1598–1602.

ALROWILI, S.; VIJAY-SHANKER, K. Biom-transformers: building large biomedical language models with bert, albert and electra. In: **Proceedings of the 20th workshop on biomedical language processing**. [S.l.: s.n.], 2021. p. 221–227.

ARNX, A. **First Neural Network for beginners explained (with code)**. Towards Data Science, 2019. Available from Internet: <https://towardsdatascience.com/first-neural-network-for-beginners-explained-with-code-4cfd37e06eaf>.

BEVAN, R.; TURBITT, O.; ABOSHOKOR, M. Mdc at semeval-2023 task 7: Fine-tuning transformers for textual entailment prediction and evidence retrieval in clinical trials. In: **Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023)**. [S.l.: s.n.], 2023. p. 1287–1292.

CHEN, C.-Y. et al. Ncuee-nlp at semeval-2023 task 7: Ensemble biomedical linkbert transformers in multi-evidence natural language inference for clinical trial data. In: **Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023)**. [S.l.: s.n.], 2023. p. 776–781.

DEVLIN, J. et al. Bert: Pre-training of deep bidirectional transformers for language understanding. **arXiv preprint arXiv:1810.04805**, 2018.

DEYOUNG, J. et al. Evidence inference 2.0: More data, better models. **arXiv preprint arXiv:2005.04177**, 2020.

DIAS, A. C. et al. Team inf-ufrgs at semeval-2023 task 7: Supervised contrastive learning for pair-level sentence classification and evidence retrieval. In: **Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023)**. [S.l.: s.n.], 2023. p. 700–706.

GANAIE, M. A. et al. Ensemble deep learning: A review. **Engineering Applications of Artificial Intelligence**, Elsevier, v. 115, p. 105151, 2022.

GOODMAN, J. T. A bit of progress in language modeling. **Computer Speech & Language**, Elsevier, v. 15, n. 4, p. 403–434, 2001.

GURURANGAN, S. et al. Don't stop pretraining: Adapt language models to domains and tasks. **arXiv preprint arXiv:2004.10964**, 2020.

HARRIS, Z. S. Distributional structure. **Word**, Taylor & Francis, v. 10, n. 2-3, p. 146–162, 1954.

HE, P.; GAO, J.; CHEN, W. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. **arXiv preprint arXiv:2111.09543**, 2021.

HE, P. et al. Deberta: Decoding-enhanced bert with disentangled attention. **arXiv preprint arXiv:2006.03654**, 2020.

HIRSCHBERG, J.; MANNING, C. D. Advances in natural language processing. **Science**, American Association for the Advancement of Science, v. 349, n. 6245, p. 261–266, 2015.

HUANG, M. et al. Cpic at semeval-2023 task 7: Gpt2-based model for multi-evidence natural language inference for clinical trial data. In: **Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023)**. [S.l.: s.n.], 2023. p. 397–401.

JULLIEN, M. **NLI4CT - Semeval 2023**. 2023. Available from Internet: <https://sites.google.com/view/nli4ct/semeval-2023>.

JULLIEN, M. et al. Semeval-2023 task 7: Multi-evidence natural language inference for clinical trial data. In: **Proceedings of the 17th International Workshop on Semantic Evaluation**. [S.l.: s.n.], 2023.

JULLIEN, M. et al. Semeval-2023 task 7: Multi-evidence natural language inference for clinical trial data. **arXiv preprint arXiv:2305.02993**, 2023.

KROSE, B. **An introduction to neural networks**. [S.l.: s.n.], 1996.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **nature**, Nature Publishing Group UK London, v. 521, n. 7553, p. 436–444, 2015.

LI, S. et al. Pair-level supervised contrastive learning for natural language inference. In: IEEE. **ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.], 2022. p. 8237–8241.

LIU, Y. et al. Roberta: a robustly optimized bert pretraining approach (2019). **arXiv preprint arXiv:1907.11692**, CoRR, v. 364, 1907.

LOSHCHILOV, I.; HUTTER, F. Fixing weight decay regularization in adam. 2018.

MACCARTNEY, B. **Natural language inference**. [S.l.]: Stanford University, 2009.

MAHENDRA, R.; SPINA, D.; VERSPOOR, K. Ittc at semeval 2023-task 7: Document retrieval and sentence similarity for evidence retrieval in clinical trial data. In: **Proceedings of the 17th International Workshop on Semantic Evaluation, Toronto, Canada. Association for Computational Linguistics**. [S.l.: s.n.], 2023.

MANNING, C. D. **An introduction to information retrieval**. [S.l.]: Cambridge university press, 2009.

MOHAMED, S. S. N.; SRINIVASAN, K. Ssnsheerinkavitha at semeval-2023 task 7: Semantic rule based label prediction using tf-idf and bm25 techniques. In: **Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023)**. [S.l.: s.n.], 2023. p. 950–957.

NEVES, M. Bf3r at semeval-2023 task 7: a text similarity model for textual entailment and evidence retrieval in clinical trials and animal studies. In: **Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023)**. [S.l.: s.n.], 2023. p. 125–129.

RADFORD, A. et al. Language models are unsupervised multitask learners. **OpenAI blog**, v. 1, n. 8, p. 9, 2019.

RAJAMANICKAM, S.; RAJARAMAN, K. I2r at semeval-2023 task 7: Explanations-driven ensemble approach for natural language inference over clinical trial data. In: **Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023)**. [S.l.: s.n.], 2023. p. 1630–1635.

ROMANOV, A.; SHIVADE, C. Lessons from natural language inference in the clinical domain. **arXiv preprint arXiv:1808.06752**, 2018.

SOLEIMANI, A.; MONZ, C.; WORRING, M. Bert for evidence retrieval and claim verification. In: SPRINGER. **Advances in Information Retrieval: 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14–17, 2020, Proceedings, Part II 42**. [S.l.], 2020. p. 359–366.

SUTTON, R. T. et al. An overview of clinical decision support systems: benefits, risks, and strategies for success. **NPJ digital medicine**, Nature Publishing Group UK London, v. 3, n. 1, p. 17, 2020.

VASSILEVA, S. et al. Fmi-su at semeval-2023 task 7: Two-level entailment classification of clinical trials enhanced by contextual data augmentation. In: **Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023)**. [S.l.: s.n.], 2023. p. 1454–1462.

VASWANI, A. et al. Attention is all you need. **Advances in neural information processing systems**, v. 30, 2017.

VLADIKA, J.; MATTHES, F. Sebis at semeval-2023 task 7: A joint system for natural language inference and evidence retrieval from clinical trial reports. **arXiv preprint arXiv:2304.13180**, 2023.

WILLIAMS, A.; NANGIA, N.; BOWMAN, S. R. A broad-coverage challenge corpus for sentence understanding through inference. **arXiv preprint arXiv:1704.05426**, 2017.

YANG, X. et al. Gatortron: A large clinical language model to unlock patient information from unstructured electronic health records. **arXiv preprint arXiv:2203.03540**, 2022.

ZHAO, X. et al. Hw-tsc at semeval-2023 task 7: Exploring the natural language inference capabilities of chatgpt and pre-trained language model for clinical trial. In: **Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023)**. [S.l.: s.n.], 2023. p. 1603–1608.

ZHOU, Y. et al. Thifly research at semeval-2023 task 7: A multi-granularity system for ctr-based textual entailment and evidence retrieval. **arXiv preprint arXiv:2306.01245**, 2023.