**Atividade 2: Aplicação de Modelos de QA em Documentos com Hugging Face**

Disciplina: Processamento de Linguagem Natural  
Universidade Federal de Sergipe

**Equipe:  
José Batista  
Carlos Melo  
Roberdan Tamyr  
Arthur Matheus**

Sumário

[1. Introdução 3](#_Toc204682837)

[2. Descrição da Atividade 3](#_Toc204682838)

[3. Metodologia 3](#_Toc204682839)

[3.1. Configuração do Ambiente 3](#_Toc204682840)

[Instalação de arquivos necessários 4](#_Toc204682841)

[Extração de texto do .docx 4](#_Toc204682842)

[Extração de texto do .pdf 4](#_Toc204682843)

[3.3. Modelos e Perguntas Selecionados 5](#_Toc204682844)

[3.4. Processo de Question Answering (QA) 5](#_Toc204682845)

[3.5. Critérios de Avaliação 5](#_Toc204682846)

[4. Resultados e Análise 6](#_Toc204682847)

[4.1. Tabela Comparativa de Resultados 6](#_Toc204682848)

[4.2. Gráfico de Desempenho 7](#_Toc204682849)

[4.3. Análise dos Modelos 7](#_Toc204682850)

[5. Conclusão 8](#_Toc204682851)

[6. Referências 8](#_Toc204682852)

[7. Repositório 8](#_Toc204682853)

[8. Notebook 8](#_Toc204682854)

[9. Vídeo 9](#_Toc204682855)

[10. Participação 9](#_Toc204682856)

# 1. Introdução

Este documento detalha a implementação e avaliação de um sistema de Perguntas e Respostas (Question Answering - QA) que utiliza modelos de Linguagem Natural (LLMs) disponíveis na plataforma Hugging Face. A atividade tem como objetivo extrair informações de documentos nos formatos DOCX e PDF para responder a perguntas específicas, comparando a eficácia de diferentes modelos de PLN.  
 O desafio central é desenvolver um script capaz de processar textos não estruturados de diversas fontes, aplicar múltiplos modelos de QA e avaliar seu desempenho de forma objetiva. A avaliação foca no alinhamento das respostas geradas com o conteúdo original dos documentos, utilizando uma combinação de análise manual e métricas de similaridade semântica. Este trabalho explora a aplicação prática de LLMs em tarefas de extração de informação, um campo fundamental no Processo de Linguagem Natural.

# 2. Descrição da Atividade

A atividade consiste na criação de um script em Python, no formato de um Notebook do Google Colab, para implementar um sistema de Perguntas e Respostas. A base de conhecimento para o sistema são dois documentos fornecidos: “dicionariodedados.docx” e “doencasrespiratoriascronicas.pdf”.  
 O processo envolve a seleção de, no mínimo, três modelos de QA gratuitos da plataforma Hugging Face. Para cada um dos dois documentos, foram elaboradas três perguntas distintas, totalizando seis perguntas para o teste de cada modelo.  
 O principal objetivo é comparar o desempenho dos modelos, avaliando a precisão e o alinhamento das respostas em relação ao conteúdo dos textos de referência. A avaliação de desempenho é apresentada através de uma tabela comparativa detalhada e um gráfico visual, destacando a efetividade de cada modelo e os critérios utilizados para a comparação.

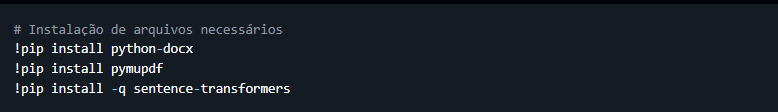
# 3. Metodologia

A metodologia foi estruturada em cinco etapas principais: configuração do ambiente, extração de conteúdo dos documentos, seleção de modelos e perguntas, execução do processo de QA e, por fim, a avaliação dos resultados.

## 3.1. Configuração do Ambiente

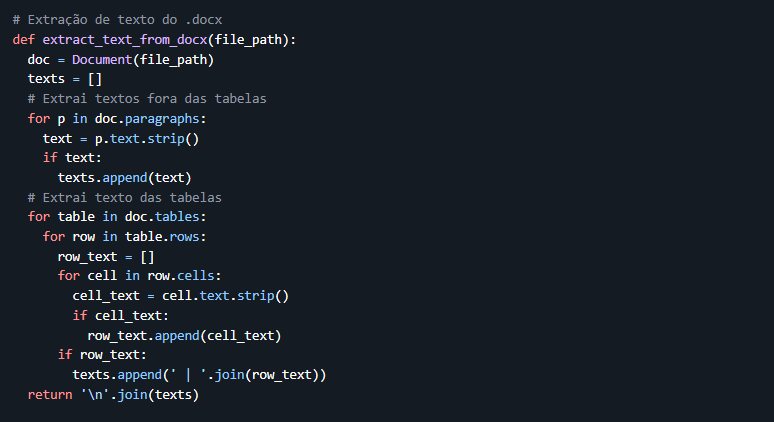
O ambiente de desenvolvimento foi configurado no Google Colab, garantindo a reprodutibilidade do projeto. As bibliotecas essenciais para a manipulação de arquivos, processamento de linguagem natural e visualização de dados foram instaladas.

### Instalação de arquivos necessários

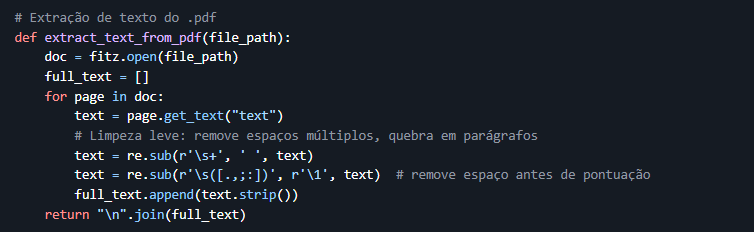
3.2. Extração de Conteúdo dos Documentos

Para que os modelos pudessem processar as informações, o texto foi extraído dos arquivos DOCX e PDF. Foram criadas funções específicas para cada tipo de arquivo, utilizando as bibliotecas python-docx e PyMuPDF.

### Extração de texto do .docx



### Extração de texto do .pdf



Como os modelos de QA possuem uma limitação no tamanho do contexto que podem processar de uma só vez, os textos extraídos foram divididos em blocos menores (chunks) para garantir que nenhuma informação fosse perdida durante a análise.

## 3.3. Modelos e Perguntas Selecionados

Foram escolhidos três modelos de Question Answering do Hugging Face, com arquiteturas distintas, para avaliar a diversidade de desempenho:

1. ***deepset/roberta-base-squad2***: Um modelo baseado na arquitetura RoBERTa, otimizado para tarefas de QA.
2. ***distilbert-base-cased-distilled-squad***: Uma versão mais leve e rápida do BERT, ideal para cenários com restrições de recursos.
3. ***timpal0l/mdeberta-v3-base-squad2***: Um modelo multilingual baseado em DeBERTa, que pode ter um bom desempenho com textos em português.

Para cada documento, foram formuladas três perguntas, com suas respectivas respostas esperadas extraídas diretamente dos textos:

Documento: Dicionário de Dados.docx

* **Pergunta 1**: Qual o nome da tabela LFCES004 no banco de produção federal?
* **Pergunta 2**: Na tabela RLESTABCOMPLEMENTAR, o que representa o campo QTDE\_SUS?
* **Pergunta 3**: Quais são os domínios do campo INDGESTOR, da tabela FCESGEST?

Documento: doencas\_respiratorias\_cronicas.pdf

* **Pergunta 1**: Quais são os principais fatores de risco preveníveis para doenças respiratórias crônicas (DRC)?
* **Pergunta 2**: Qual é a definição clássica de sintomático respiratório?
* **Pergunta 3**: Como a tosse pode ser classificada?

## 3.4. Processo de Question Answering (QA)

Para cada modelo, o processo de QA foi executado iterando sobre cada pergunta e os blocos de texto do documento correspondente. O pipeline question-answering da biblioteca transformers foi utilizado para obter uma resposta de cada bloco. A resposta final para cada pergunta foi aquela com a maior pontuação de confiança (score) entre todos os blocos analisados.

## 3.5. Critérios de Avaliação

A avaliação da efetividade dos modelos foi baseada em três critérios:

1. Avaliação Manual: As respostas geradas foram classificadas manualmente em três categorias para uma análise qualitativa rápida:  
   ✔️ (Correta): A resposta do modelo é idêntica ou semanticamente equivalente à resposta esperada.  
   ➖ (Parcialmente Correta): A resposta contém parte da informação correta, mas está incompleta ou imprecisa.  
   ❌ (Incorreta): A resposta está errada ou não tem relação com a pergunta.
2. Similaridade Semântica (Cosseno): Para uma avaliação quantitativa, a similaridade de cosseno entre os embeddings da resposta do modelo e da resposta esperada foi calculada. Utilizou-se o modelo ***sentence-transformers/all-MiniLM-L6-v2*** para gerar os embeddings. Essa métrica varia de -1 a 1, onde valores mais próximos de 1 indicam maior similaridade semântica.
3. Confiança do Modelo: A pontuação de confiança (score) retornada pelo próprio modelo de QA foi registrada. Este valor indica o quão confiante o modelo está em sua própria resposta.

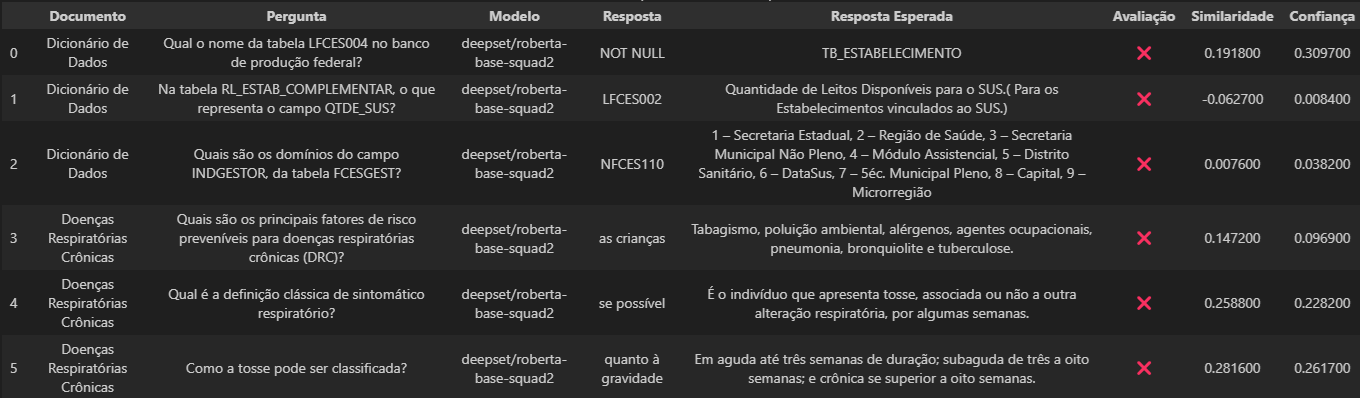
# 4. Resultados e Análise

Os resultados da avaliação foram compilados em tabelas detalhadas para cada modelo e em um gráfico comparativo para facilitar a análise.

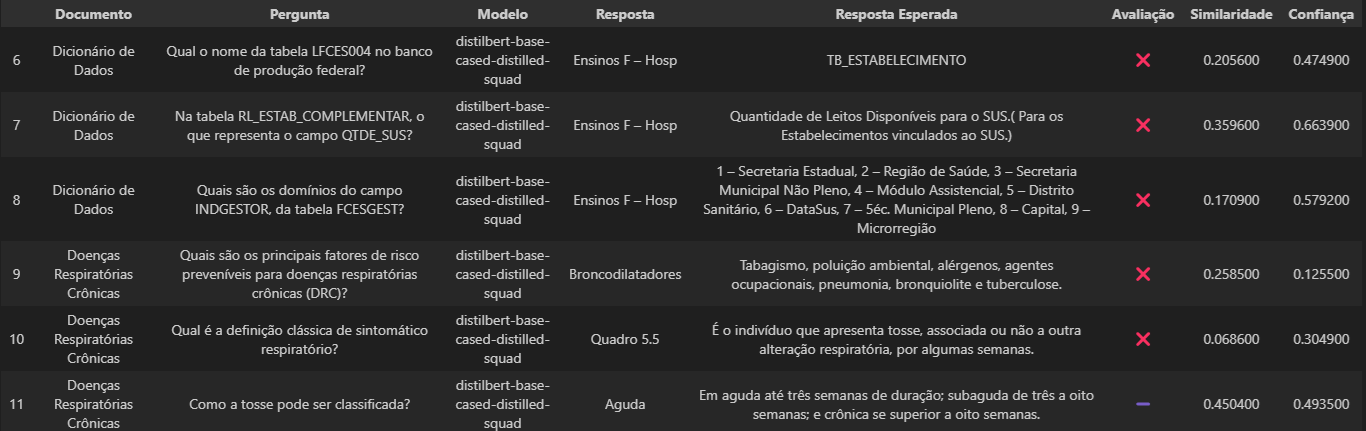
## 4.1. Tabela Comparativa de Resultados

As tabelas a seguir exibem os resultados detalhados para cada modelo, incluindo a pergunta, a resposta gerada, a resposta esperada e as métricas de avaliação.

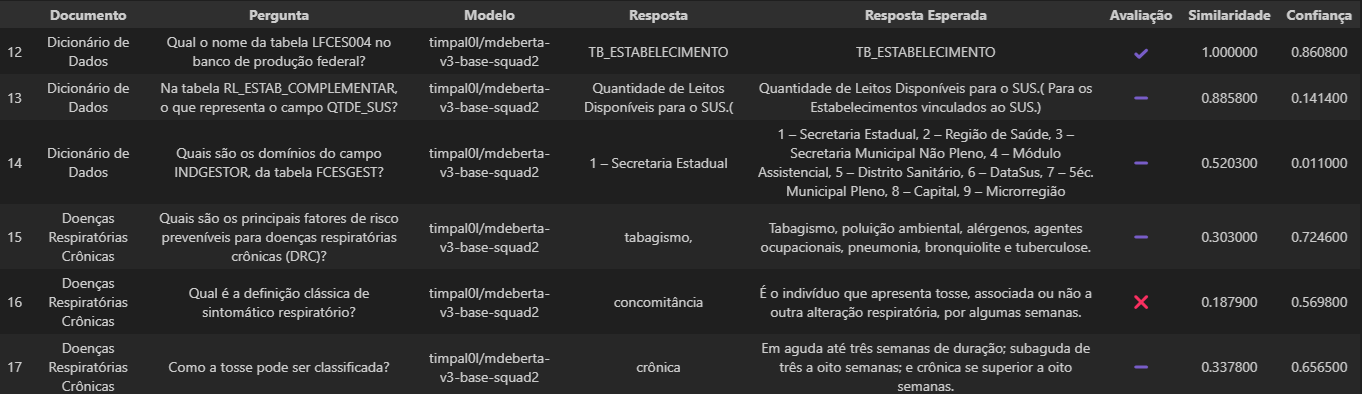
Resultados para: ***deepset/roberta-base-squad2***



Resultados para: ***distilbert-base-cased-distilled-squad***

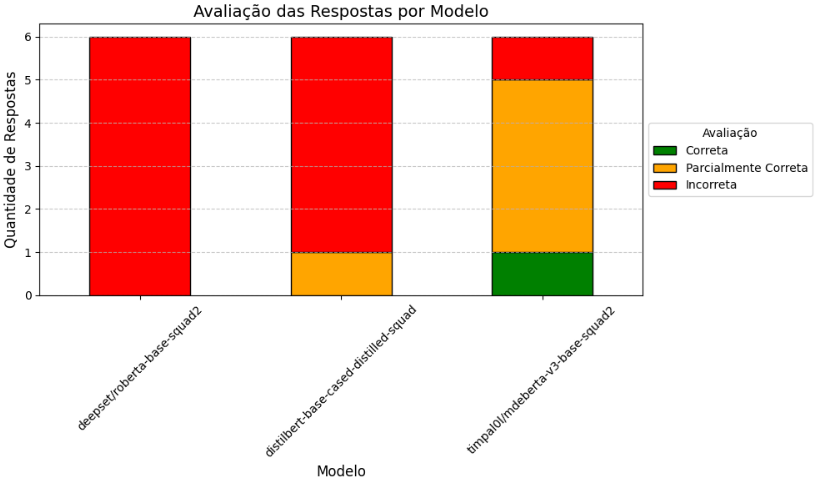


Resultados para: ***timpal0l/mdeberta-v3-base-squad2***



## 4.2. Gráfico de Desempenho

O gráfico de barras empilhadas abaixo resume a avaliação manual, mostrando a quantidade de respostas corretas, parcialmente corretas e incorretas para cada modelo.



## 4.3. Análise dos Modelos

* deepset/roberta-base-squad2: Este modelo teve o pior desempenho, não conseguindo acertar nenhuma pergunta de forma completa ou parcial. As respostas foram curtas e fora de contexto, e as pontuações de confiança foram consistentemente baixas, indicando que o próprio modelo não estava seguro de suas respostas.
* distilbert-base-cased-distilled-squad: O desempenho foi ligeiramente melhor que o do RoBERTa, com uma resposta classificada como parcialmente correta ("Aguda" para a classificação da tosse). No entanto, a maioria das respostas foi incorreta, apesar de apresentar scores de confiança mais altos, o que sugere um excesso de confiança do modelo em respostas erradas.
* timpal0l/mdeberta-v3-base-squad2: Este foi o modelo com o melhor desempenho entre os três. Ele forneceu uma resposta totalmente correta e quatro respostas parcialmente corretas. Suas respostas, embora muitas vezes incompletas, eram contextualmente relevantes. Curiosamente, suas pontuações de confiança foram relativamente baixas, mesmo para a resposta correta, o que pode indicar uma melhor calibração ou uma maior "cautela" do modelo. Sua capacidade de lidar com a estrutura tabular do documento de dicionário de dados foi notavelmente superior à dos outros modelos.

# 5. Conclusão

A atividade demonstrou que a escolha do modelo de Question Answering é crucial para o sucesso da tarefa. Modelos mais leves como o DistilBERT ou mesmo modelos robustos como o RoBERTa podem não ter o desempenho esperado em domínios específicos ou com textos em português, especialmente quando o conteúdo é estruturado (como em tabelas).  
 O modelo timpal0l/mdeberta-v3-base-squad2, de base multilíngue, mostrou-se mais eficaz para os documentos analisados, conseguindo extrair informações relevantes tanto do texto corrido do PDF quanto dos dados tabulares do DOCX. Isso reforça a importância de testar e validar diferentes arquiteturas de modelos para encontrar a mais adequada a um determinado caso. de usos nfdjfnjs s njndjfnsdjfn jnfj sdjn jsdnfj njdn js ndj jdnfjs jdsnfjsd  
 A avaliação também destacou que a pontuação de confiança de um modelo nem sempre se correlaciona diretamente com a precisão da resposta. Portanto, uma combinação de métricas quantitativas, como a similaridade de cosseno, e uma avaliação qualitativa manual é fundamental para uma análise de desempenho completa e confiável.

# 6. Referências

* Hugging Face. *Transformers Documentation*. Disponível em: <https://huggingface.co/docs/transformers/index.>
* Hugging Face. *Sentence-Transformers Documentation*. Disponível em: <https://huggingface.co/sentence-transformers.>
* Model Card. *deepset/roberta-base-squad2*. Disponível em: <https://huggingface.co/deepset/roberta-base-squad2.>
* Model Card. *distilbert-base-cased-distilled-squad*. Disponível em: <https://huggingface.co/distilbert-base-cased-distilled-squad.>
* Model Card. *timpal0l/mdeberta-v3-base-squad2*. Disponível em: <https://huggingface.co/timpal0l/mdeberta-v3-base-squad2.>

# 7. Repositório

<https://github.com/shykiu77/PLN_QA_LLM_2025_Melo_Carlos/tree/main>

# 8. Notebook

https://colab.research.google.com/drive/1v7O56jkR\_l8kB-sMQdtzuRnsvMM4CG5B?usp=sharing

# 9. Vídeo

* [URL para o vídeo da apresentação]

# 10. Participação

* Arthur Matheus: Implementação do código
* Carlos Melo: Escrita do documento