Themis

Gabriel Soares Baptista
Departamento De Informática (DI)
Universidade Federal do Espírito Santo (UFES)
Vitória, Brasil
gsoaresbaptista@gmail.com

Humberto Giuri Calente
Departamento De Informática (DI)
Universidade Federal do Espírito Santo (UFES)
Vitória, Brasil
humbertogiuri@gmail.com

***Abstract*—Este trabalho aborda o aprimoramento de um Modelo de Linguagem de Grande Escala (LLMs) para um banco de questões jurídicas. Utiliza-se o processo de *Fine-Tuning* do modelo Canarim-7B com dados da OAB e documentos da legislação brasileira, com o objetivo de desenvolver um modelo capaz de fornecer respostas justificadas para questões de natureza semelhante. A metodologia envolve a coleta de dados jurídicos de diversas fontes e a aplicação de técnicas de treinamento de LLMs do estado da arte, incluindo o algoritmo QLoRA, que demonstra eficácia mesmo com recursos computacionais reduzidos. Além disso, o trabalho abrange a criação de uma aplicação com interface gráfica interativa e um servidor para o modelo, exemplificando como esse tipo de modelo pode ser integrado em produtos nacionais.**

***Index Terms*—LLM, Canarim-7B, OAB, Fine-Tuning, QLoRA**

I. Introdução

Nos últimos anos, os Modelos de Linguagem de Grande Escala (LLMs) [5] emergiram como uma força poderosa no campo da inteligência artificial, revolucionando a maneira como máquinas entendem e geram linguagem humana. Esses modelos, treinados em vastos conjuntos de dados, demonstraram habilidades notáveis em uma variedade de tarefas linguísticas. No entanto, quando se trata de domínios especializados, como o direito, esses modelos frequentemente enfrentam desafios devido à natureza específica e técnica da linguagem jurídica.

Este projeto visa abordar essa lacuna, concentrando-se no retreinamento da LLM Canarim-7B [3] com um conjunto de dados jurídicos, incluindo provas da Ordem dos Advogados do Brasil (OAB). A rede Canarim-7B é um modelo LLM com foco na língua portuguesa que foi pré-treinada utilizando os pesos da LLaMA2-7B [4]. O objetivo é aprimorar a capacidade do modelo de responder com precisão a questões de concursos como a prova da OAB. Para isso, foi desenvolvida uma interface interativa, assemelhando-se a uma versão especializada do ChatGPT, onde os usuários podem fazer login e submeter questões jurídicas, tendo suas interações anteriores salvas para referência futura.

A relevância deste projeto estende-se além do aprimoramento técnico de um modelo de linguagem. Ao integrar conhecimentos jurídicos específicos na Canarim-7B, o projeto busca criar um recurso que possa auxiliar na compreensão e no estudo do direito, democratizando o acesso a informações jurídicas especializadas. Além disso, ele abre caminho para pesquisas futuras sobre a aplicação de LLMs em outras áreas especializadas, demonstrando o potencial desses modelos para se adaptarem a diferentes campos de conhecimento.

A metodologia adotada inclui a coleta de dados jurídicos e a implementação de técnicas de treinamento específicas para integrar este conhecimento ao modelo Canarim-7B. A interface desenvolvida é um componente crítico do projeto, permitindo uma interação eficiente e intuitiva com o modelo re-treinado.

O código pode ser acessado através em GitHub - Themis. Enquanto a versão quantizada do modelo em GGUF com a quantização Q5_K_M está disponibilizada no site Hugging Face.

II. Conjunto de Dados

Nesta seção, apresentamos os dados coletados e tratados para o *Fine-Tuning* do modelo de linguagem natural *Canarim-7B*. A seleção e curadoria desses dados visaram abranger uma ampla gama de informações jurídicas, garantindo ao modelo os recursos necessários para responder com precisão a diversas questões, especialmente aquelas relacionadas à Ordem dos Advogados do Brasil (OAB) e ao cenário jurídico brasileiro.

A integração estratégica de dados específicos da OAB revelou-se essencial para garantir uma preparação adequada do modelo, orientada para a abordagem precisa do tipo de questionamento desejado. Este enriquecimento deliberado do conjunto de dados promove uma compreensão mais profunda do sistema, sintonizando o modelo com os desafios que se almeja superar. Essa abordagem visa estabelecer uma base sólida, capacitando o modelo a lidar com as questões particulares que podem surgir no contexto da Ordem dos Advogados do Brasil.

Simultaneamente, a inclusão de dados provenientes do Vade Mecum desempenha um papel crucial ao assegurar que o modelo seja treinado com todas as informações essenciais, fornecendo-lhe o conhecimento necessário para abordar as complexidades intrínsecas ao campo jurídico.

Portanto, as próximas subseções buscam detalhar o processo de extração de informações de cada fonte de dados mencionada.

A. Provas da OAB

Nesta subseção, delineamos o processo de extração dos dados das provas da OAB.

As que questões das provas da OAB são elaboradas para avaliar o conhecimento e as habilidades dos futuros advogados em várias áreas do direito, incluindo ética profissional, legislação e procedimentos judiciais.

Composto por informações essenciais, como número da prova, número da questão, enunciado, alternativas de resposta (A, B, C, D), alternativa correta e justificativa, estas questões foram respondidas por advogadas reais. Dessa forma, tem uma contribuição valiosa, uma vez que eleva significativamente a qualidade geral do conjunto de dados final.

Os dados das provas podem ser obtidos por sites do próprio governos ou das bancas que venham a aplicar os concursos. Contudo, essas respostas apresentam apenas os gabaritos, dessa forma, é mais interessante buscarmos uma fonte de dados externa com que forneça os comentários necessários a cerca da opção correta. Por este motivo, os dados foram retirados do site Vade Mecum Brasil que fornece esses comentários oriundo de profissionais da área.

Fig. 1. Site - Vade Mecum Brasil.

Na figura 1 podemos ver que o site apresenta as provas aplicadas de abril de 2014 até julho de 2022. Onde cada um dos *link* acima nos leva para uma página que podemos ver as questões e as opções de resposta correspondentes, como mostrado na figura 2.

Note que inicialmente as respostas não são mostradas pelas questões, uma vez que o site tem como foco o estudo, portanto, faz sentido não mostrar as resoluções até com que o botão em ciano seja clicado. Portanto, o *web scraping* feito deverá clicar em cada um dos botões das páginas para que seja possível extrair o conteúdo.

Esse trabalho dividiu as questões mostradas nas figuras 1 e 2 em 3 partes principais, mostradas na figura 3, sendo elas a pergunta, as respostas, sendo que cada letra representa uma coluna no banco de dados e por fim a resposta comentada. Logo, cada uma das 80 questões de cada prova presentes no site foram extraídas e colocadas em um banco de dados SQLite.

Portanto, a tabela *oab_provas* do banco de dados contém as colunas *id*, *prova*, *numero_questao*, *enunciado*, *alternativa_a*, *alternativa_b*, *alternativa_c*, *alternativa_correta*, e *justificativa*. As questões extraídas totalizaram 1838 questões divididas entre as provas disponíveis no site.

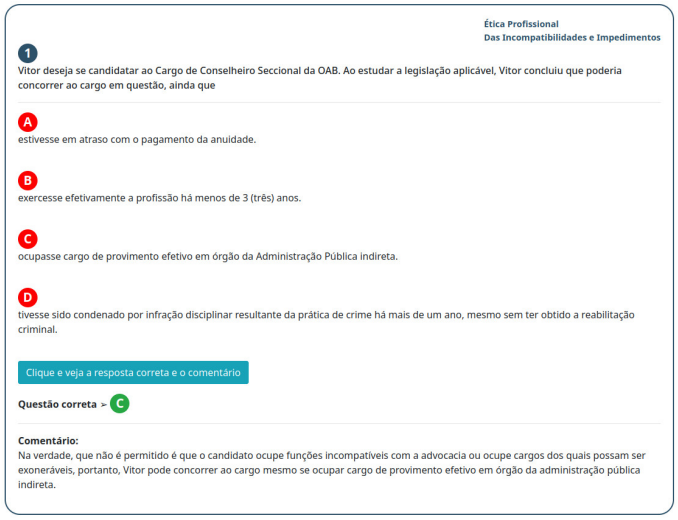


Fig. 2. Exemplo de uma questão do site Vade Mecum Brasil.

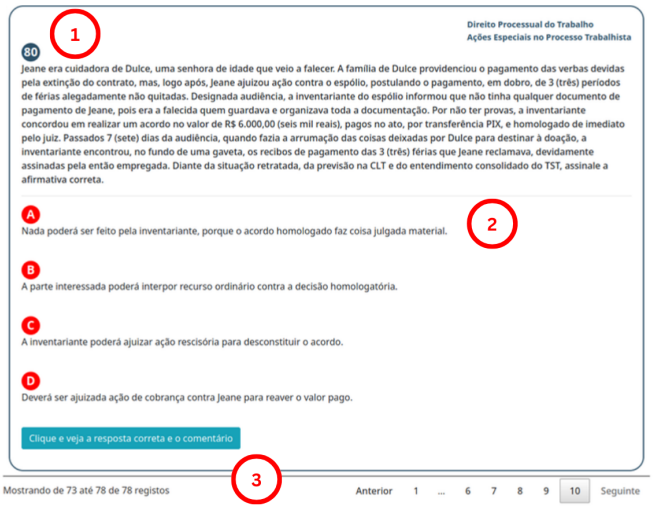


Fig. 3. Exemplo da divisão das questões. Onde em um é representado a pergunta, em dois cada uma das opções e em três a justificativa que será mostrada ao clicar-se no botão em ciano. As respostas ainda serão subdivididas.

Destaca-se que o modelo é alimentado com uma pergunta e sua resposta correspondente. Nesse contexto, as alternativas são incorporadas à própria pergunta, enquanto a resposta engloba as colunas da alternativa correta, acompanhada de sua respectiva justificativa.

B. *Vade Mecum*

Nesta subseção, delineamos o processo de extração dos dados do Vade Mecum[1], em específico um compêndio de leis fornecido gratuitamente pela Biblioteca do Senado sob a licença Creative Commons.

A incorporação do Vade Mecum visa expandir o conhecimento jurídico da LLM, proporcionando informações detal-

[1]Para mais informações sobre o Vade Mecum, acesse a página da Biblioteca do Senado: Biblioteca do Senado - Vade Mecum

hadas sobre legislação para capacitá-la a responder às questões das provas com precisão, reforçando a capacidade do modelo para lidar com nuances específicas do cenário jurídico nacional.

Observe que o Vade Mecum contém uma série de documentos que regem a legislação brasileira. Portanto, além do processo de extração de dados do PDF, é necessário realizar um tratamento mais sofisticado e uma transformação para convertê-los em perguntas. Em outras palavras, requer um processo de ETL (extração, transformação e carga) mais elaborado em comparação com os dados das provas da OAB.

Constituição da República Federativa do Brasil

Preâmbulo

Nós, representantes do povo brasileiro, reunidos em Assembleia Nacional Constituinte para instituir um Estado democrático, destinado a assegurar o exercício dos direitos sociais e individuais, a liberdade, a segurança, o bem-estar, o desenvolvimento, a igualdade e a justiça como valores supremos de uma sociedade fraterna, pluralista e sem preconceitos, fundada na harmonia social e comprometida, na ordem interna e internacional, com a solução pacífica das controvérsias, promulgamos, sob a proteção de Deus, a seguinte Constituição da República Federativa do Brasil.

TÍTULO I – DOS PRINCÍPIOS FUNDAMENTAIS

Art. 1º A República Federativa do Brasil, formada pela união indissolúvel dos Estados e Municípios e do Distrito Federal, constitui-se em Estado Democrático de Direito e tem como fundamentos:
I – a soberania;
II – a cidadania;
III – a dignidade da pessoa humana;
IV – os valores sociais do trabalho e da livre iniciativa;
V – o pluralismo político.
Parágrafo único. Todo o poder emana do povo, que o exerce por meio de representantes eleitos ou diretamente, nos termos desta Constituição.

Art. 2º São Poderes da União, independentes e harmônicos entre si, o Legislativo, o Executivo e o Judiciário.

Art. 3º Constituem objetivos fundamentais da República Federativa do Brasil:
I – construir uma sociedade livre, justa e solidária;
II – garantir o desenvolvimento nacional;
III – erradicar a pobreza e a marginalização e reduzir as desigualdades sociais e regionais;
IV – promover o bem de todos, sem preconceitos de origem, raça, sexo, cor, idade e quaisquer outras formas de discriminação.

Art. 4º A República Federativa do Brasil rege-se nas suas relações internacionais pelos seguintes princípios:
I – independência nacional;
II – prevalência dos direitos humanos;
III – autodeterminação dos povos;
IV – não intervenção;
V – igualdade entre os Estados;
VI – defesa da paz;
VII – solução pacífica dos conflitos;
VIII – repúdio ao terrorismo e ao racismo;
IX – cooperação entre os povos para o progresso da humanidade;
X – concessão de asilo político.
Parágrafo único. A República Federativa do Brasil buscará a integração econômica, política, social e cultural dos povos da América Latina, visando à formação de uma comunidade latino-americana de nações.

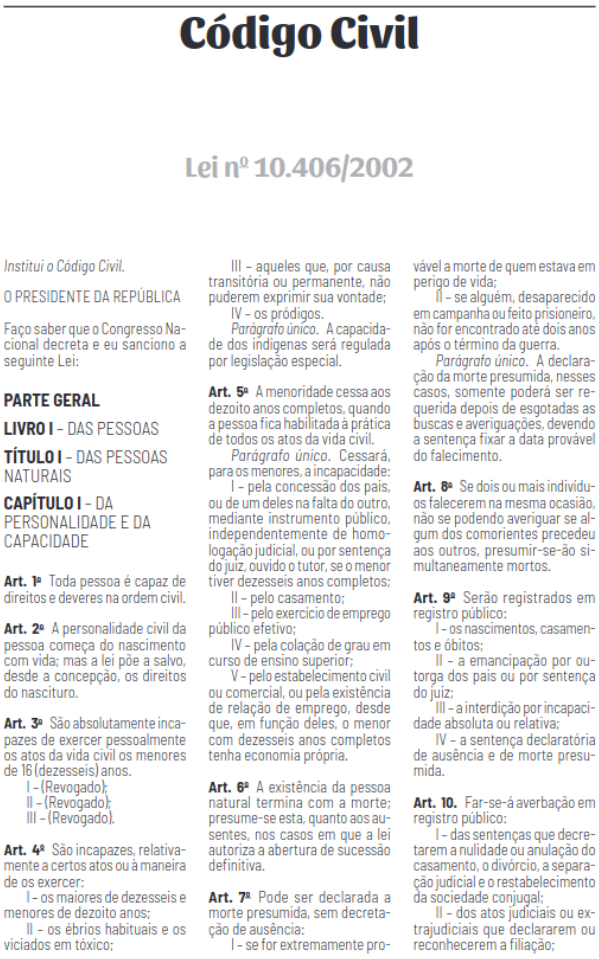
TÍTULO II – DOS DIREITOS E GARANTIAS FUNDAMENTAIS

CAPÍTULO I – DOS DIREITOS E DEVERES INDIVIDUAIS E COLETIVOS

Art. 5º Todos são iguais perante a lei, sem distinção de qualquer natureza, garantindo-se aos brasileiros e aos estrangeiros residentes no País a inviolabilidade do direito à vida, à liberdade, à igualdade, à segurança e à propriedade, nos termos seguintes:
I – homens e mulheres são iguais em direitos e obrigações, nos termos desta Constituição;
II – ninguém será obrigado a fazer ou deixar de fazer alguma coisa senão em virtude de lei;
III – ninguém será submetido a tortura nem a tratamento desumano ou degradante;
IV – é livre a manifestação do pensamento, sendo vedado o anonimato;
V – é assegurado o direito de resposta, proporcional ao agravo, além da indenização por dano material, moral ou à imagem;
VI – é inviolável a liberdade de consciência e de crença, sendo assegurado o livre exercício dos cultos religiosos e garantida, na forma da lei, a proteção aos locais de culto e a suas liturgias;
VII – é assegurada, nos termos da lei, a prestação de assistência religiosa nas entidades civis e militares de internação coletiva;
VIII – ninguém será privado de direitos por motivo de crença religiosa ou de convicção filosófica ou política, salvo se as invocar para eximir-se de obrigação legal a todos imposta e recusar-se a

Fig. 4. Example of a figure caption.

A figura 4 mostra um exemplo de página contida na versão do documento fornecido pela biblioteca do senado. Dessa forma, Nota-se que o documento possui uma organização onde os artigos são contidos semanticamente dentro de outras estruturas que podem ou não englobar mais de um artigo. Entretanto, é importante observar que o documento omite essas estruturas quando não há a ocorrência de mais de uma delas. Por exemplo, o artigo 1 do título 1 está contido apenas dentro desse título, uma vez que não existem mais capítulos associados. Por outro lado, o título 2 contém mais de um capítulo, resultando no artigo 5 contido dentro dessa estrutura.



Código Civil

Lei nº 10.406/2002

Institui o Código Civil.

O PRESIDENTE DA REPÚBLICA

Faço saber que o Congresso Nacional decreta e eu sanciono a seguinte Lei:

PARTE GERAL

LIVRO I – DAS PESSOAS

TÍTULO I – DAS PESSOAS NATURAIS

CAPÍTULO I – DA PERSONALIDADE E DA CAPACIDADE

Art. 1º Toda pessoa é capaz de direitos e deveres na ordem civil.

Art. 2º A personalidade civil da pessoa começa do nascimento com vida; mas a lei põe a salvo, desde a concepção, os direitos do nascituro.

Art. 3º São absolutamente incapazes de exercer pessoalmente os atos da vida civil os menores de 16 (dezesseis) anos.
I – (Revogado);
II – (Revogado);
III – (Revogado).

Art. 4º São incapazes, relativamente a certos atos ou à maneira de os exercer:
I – os maiores de dezesseis e menores de dezoito anos;
II – os ébrios habituais e os viciados em tóxico;
III – aqueles que, por causa transitória ou permanente, não puderem exprimir sua vontade;
IV – os pródigos.
Parágrafo único. A capacidade dos indígenas será regulada por legislação especial.

Art. 5º A menoridade cessa aos dezoito anos completos, quando a pessoa fica habilitada à prática de todos os atos da vida civil.
Parágrafo único. Cessará, para os menores, a incapacidade:
I – pela concessão dos pais, ou de um deles na falta do outro, mediante instrumento público, independentemente de homologação judicial, ou por sentença do juiz, ouvido o tutor, se o menor tiver dezesseis anos completos;
II – pelo casamento;
III – pelo exercício de emprego público efetivo;
IV – pela colação de grau em curso de ensino superior;
V – pelo estabelecimento civil ou comercial, ou pela existência de relação de emprego, desde que, em função deles, o menor com dezesseis anos completos tenha economia própria.

Art. 6º A existência da pessoa natural termina com a morte; presume-se esta, quanto aos ausentes, nos casos em que a lei autoriza a abertura de sucessão definitiva.

Art. 7º Pode ser declarada a morte presumida, sem decretação de ausência:
I – se for extremamente provável a morte de quem estava em perigo de vida;
II – se alguém, desaparecido em campanha ou feito prisioneiro, não for encontrado até dois anos após o término da guerra.
Parágrafo único. A declaração da morte presumida, nesses casos, somente poderá ser requerida depois de esgotadas as buscas e averiguações, devendo a sentença fixar a data provável do falecimento.

Art. 8º Se dois ou mais indivíduos falecerem na mesma ocasião, não se podendo averiguar se algum dos comorientes precedeu aos outros, presumir-se-ão simultaneamente mortos.

Art. 9º Serão registrados em registro público:
I – os nascimentos, casamentos e óbitos;
II – a emancipação por outorga dos pais ou por sentença do juiz;
III – a interdição por incapacidade absoluta ou relativa;
IV – a sentença declaratória de ausência e de morte presumida.

Art. 10. Far-se-á averbação em registro público:
I – das sentenças que decretarem a nulidade ou anulação do casamento, o divórcio, a separação judicial e o restabelecimento da sociedade conjugal;
II – dos atos judiciais ou extrajudiciais que declararem ou reconhecerem a filiação;

Fig. 5. Example of a figure caption.

A estruturação dos arquivos é notavelmente complexa, dada a importância desses documentos na regulação da legislação de um país. Por exemplo, na Figura 5, observamos a presença das estruturas de título, acrescentando camadas adicionais de complexidade. Portanto, caso haja interesse, uma análise mais detalhada pode ser conduzida, anualizando o banco de dados gerado para compreender a estrutura final do arquivo. Nesse banco de dados, todos os elementos possíveis na estrutura são utilizados, sendo preenchidos como elementos vazios (NULL) nas linhas da tabela quando omitidos.

Na Figura 6, apresentamos a divisão das estruturas para a geração do conjunto de dados. O retângulo preto representa o nome do código, enquanto cada retângulo colorido representa um dado distinto. Observa-se as divisões que as linhas terão devido aos elementos presentes em cada estrutura. Importante notar que todos os elementos terão tantas colunas quanto o elemento mais interno em uma dessas estruturas.

É relevante mencionar que os artigos são separados para possibilitar a disponibilização individual a modelos. Portanto, teremos uma linha associada ao Título 1 da Constituição para

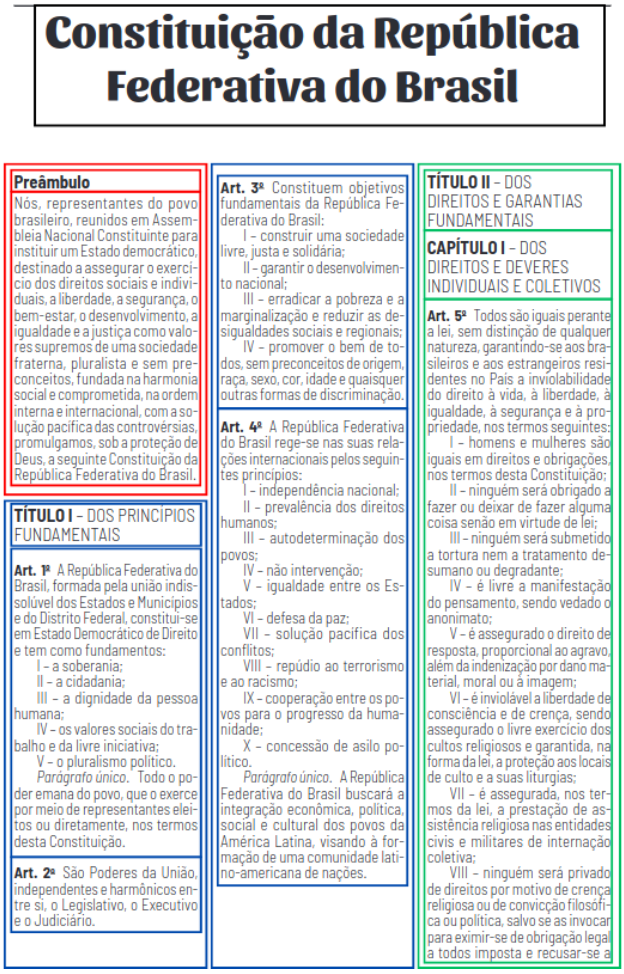


Fig. 6. Example of a figure caption.

o Artigo 1, outra para o Artigo 2, e assim por diante. Essa abordagem permite uma manipulação mais granular dos dados, facilitando a análise e treinamento de modelos.

O banco de dados SQLite resultante desses dados possui 7107 linhas. No entanto, como mencionado anteriormente, essas linhas representam os artigos, não as perguntas desejadas para alimentar os modelos. Para atender a essa necessidade, geramos algumas perguntas utilizando uma estrutura como ”Gostaria de entender o que está previsto no Artigo ### do(a) ###”, em que os códigos (###) são substituídos pelos correspondentes através do código em Python. Esse processo visa adequar os dados ao formato desejado para fornecer perguntas contextualizadas aos modelos.

C. Processamento dos Conjuntos

Para utilizar os dados coletados no processo de Fine-Tuning, se fez necessário padronizar prompts para apresentar para a rede. Como o intuito do projeto é fazer com que a rede responda questões, principalmente de múltipla escolha, transformou todos os dados em perguntas e respostas. Para as questões obtidas nas provas da OAB, padronizou a pergunta como o enunciado da questão seguido por suas alternativas, e como resposta, uniu a questão correta com o comentário da respectiva questão.

Passando para os dados obtidos do Vade Mecum, foi feito três tipos de processamento. Primeiramente, utilizou uma biblioteca python chamada G4F, que faz requisições para a LLM do Bing, para criar perguntas e respostas baseadas nos artigos da constituição. Dessa forma, a ferramenta gerava uma pergunta e uma resposta baseada em determinada lei. Para obter esse resultado, foi utilizado o seguinte padrão de prompt:

```
prompt = (
    'Com base no seguinte texto, formule '
    'uma pergunta cuja resposta esteja '
    'contida no texto e retorne '
    'APENAS uma linha contendo a pergunta '
    'e resposta separadas por ;. Responda '
    'de maneira completa e cite a fonte do '
    'artigo fornecido. Evite perguntas sobre '
    'titulos. Artigo: "'
    f'{data}\n"\n'
    '""'
)
```

O segundo processamento também utilizou da biblioteca B4F para gerar perguntas. Mas dessa vez, as perguntas geradas foram de múltipla escolha, para simular questões da OAB e de concursos. Um dos prompts utilizados é apresentado abaixo.

```
prompt = (
    'Com base no texto fornecido, elabore '
    'uma pergunta de multipla escolha '
    'com 5 opcoes (letras A a E). '
    'Alem disso, forneca uma resposta '
    'comentada explicando por que a opcao '
    'escolhida esta correta. '
    'Evite perguntas sobre titulos. Artigo: '
    f'{data}\n"\n'
    '""'
)
```

O terceiro processamento focou em fazer com que a rede conseguisse obter o conhecimento das leis presentes no livro. Dessa forma, formatou a pergunta com com frases do tipo ”O que diz no artigo x da fonte y?” e como resposta passou o texto referente ao artigo retirado do Vade Mecum, possibilitando que a rede aprenda as leis necessárias.

Com todos os dados transformados em perguntas e respostas, foi possível criar um banco de dados SQLite contendo os campos: ID, question e answer. Esse conjunto de dados foi utilizado na técnica de Fine-Tuning. A tabela a seguir, apresenta a divisão da quantidade de dados obtido de cada uma das 3 transformações.

Vale ressaltar, que ao se utilizar a biblioteca BF4, não é possível garantir um padrão nas respostas obtidas nas requisições. Dessa forma, foi necessário validar manualmente todos as repostas obtidas através da biblioteca, gerando um

Dados	Quantidade
OAB	1343
Vade Mecum - Múltipla Escolha	1039
Vade Mecum - Pergunta Simples	7519
Vade Mecum - Leis	6965
Total	**16866**

TABLE I
QUANTIDADE DE DADOS COLETADOS PARA TREINAMENTO DA CANARIM-7B

esforço de validação manual demasiado para que todas as respotas ficassem em um padrão aceitável para uso.

III. *Back-end* E *Front-end*

A. *Back-end*

O Back-end da Themis foi desenvolvido utilizando Python, aproveitando a biblioteca `ctransformers` para executar o modelo em formato GPT (Generative Pre-trained Transformer) quantizado. O framework escolhido para o desenvolvimento foi o Starlette, proporcionando uma base sólida para construção de APIs assíncronas.

1) Arquitetura e Estrutura: A arquitetura do Back-end segue os princípios da Clean Architecture, promovendo uma divisão clara entre as responsabilidades e uma estrutura modular. O sistema é dividido em módulos distintos:

1) **Application:** Contém os casos de uso da aplicação.
2) **Domain:** Define as entidades e regras de negócios.
3) **External:** Lida com interfaces externas, como integrações com o modelo GPT.
4) **Infra:** Implementa a infraestrutura, incluindo interação com o PostgreSQL.
5) **Presentation:** Envolve a apresentação de dados ao usuário, cuidando das rotas da API.

2) Banco de Dados: O Back-end utiliza o PostgreSQL como banco de dados relacional para armazenar informações essenciais da aplicação. Essa escolha foi feita considerando a robustez e a confiabilidade que o PostgreSQL oferece para aplicações complexas.

3) Funcionalidades e Rotas: O Back-end oferece diversas rotas para suportar as funcionalidades principais da Themis:

- Registro de Novos Usuários
- Login
- Geração de Access Tokens e Token Refresh
- Recebimento de Mensagens Específicas
- Consulta de Todas as Mensagens do Usuário
- Resposta em Streaming do Modelo

4) Integração com o Modelo GPT: O modelo GPT é integrado ao Back-end por meio do módulo External, utilizando a biblioteca `ctransformers`. O sistema permite uma resposta em streaming do modelo, proporcionando uma experiência similar ao ChatGPT.

B. *Front-end*

A interface do usuário da Themis foi desenvolvida utilizando tecnologias web padrão, com foco na usabilidade e experiência do usuário. O framework web escolhido para a implementação foi o React, utilizando TypeScript para fornecer uma base sólida para o desenvolvimento.

1) Tecnologias Utilizadas:

- HTML, CSS, JavaScript
- Framework: React
- Linguagem: TypeScript

2) Interface Gráfica: O design da interface gráfica prioriza a simplicidade e a intuição, seguindo as práticas modernas de design de interfaces React. Elementos visuais, como botões e formulários, foram cuidadosamente projetados para proporcionar uma experiência amigável ao usuário.

3) Integração com o Back-end: A comunicação entre o Front-end e o Back-end é realizada por meio de APIs RESTful, seguindo boas práticas de desenvolvimento web. A integração eficiente e dinâmica entre as duas partes da aplicação é facilitada pela estrutura modular e organizada do Back-end.

A utilização de TypeScript no Front-end proporciona benefícios adicionais, como verificação de tipos em tempo de compilação, tornando o código mais robusto e menos propenso a erros.

Algumas capturas da tela podem ser vistas nas figuras 7, 8, 9 e a versão para aparelhos móveis (telas pequenas) pode ser vista em 7.

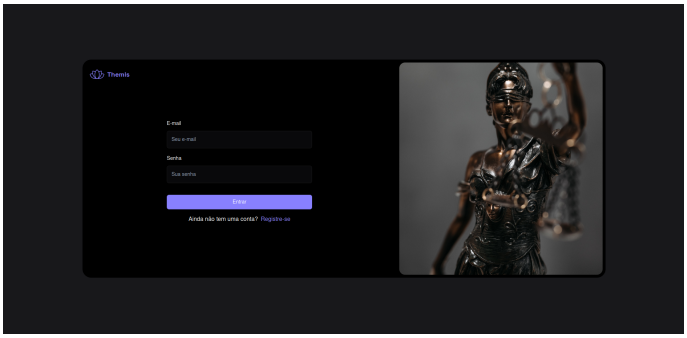


Fig. 7. Tela de Login da Themis.

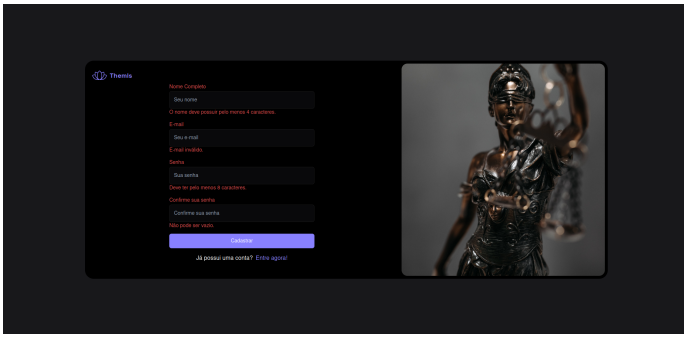


Fig. 8. Tela de Cadastro da Themis.

IV. FINE-TUNING

Fine-Tuning é um processo no aprendizado de máquina onde um modelo previamente treinado em um grande conjunto de dados é novamente treinado, mas desta vez em um conjunto

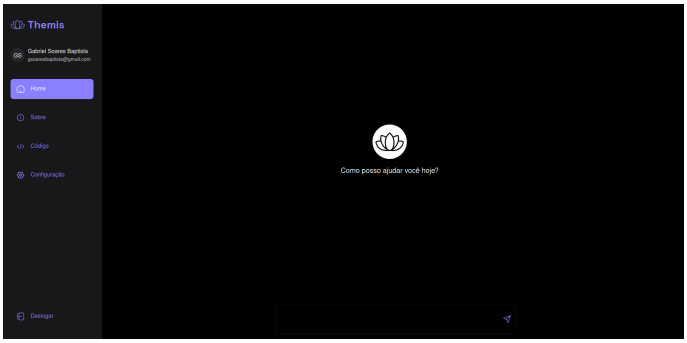


Fig. 9. Tela de Conversa da Themis.

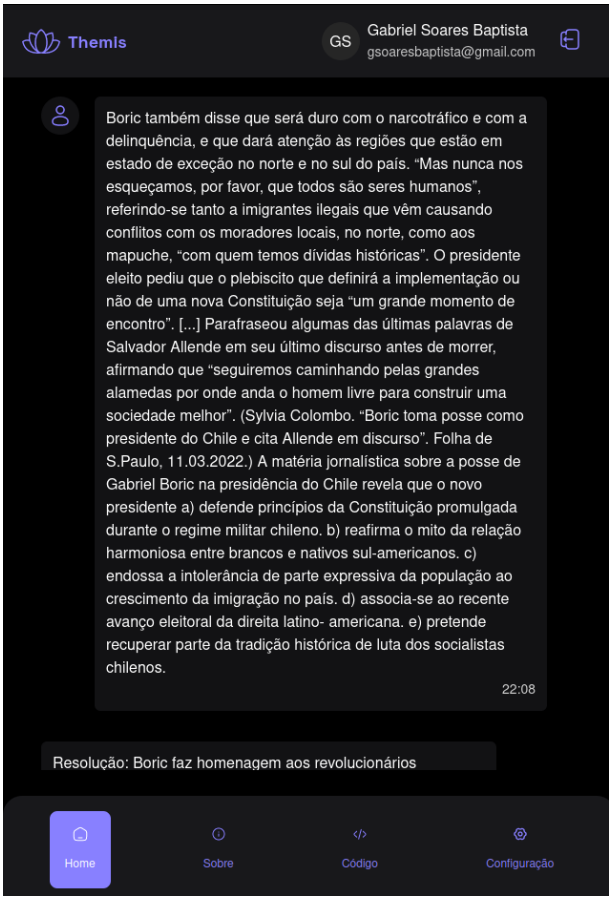


Fig. 10. Exemplo da Versão Mobile Responsiva da Themis.

menor e mais específico. Este processo é uma etapa de suma importância nos Modelos de Linguagem de Grande Escala (LLMs), pois permite ajustar um modelo com vasto conhecimento previamente adquirido em tarefas ou áreas específicas.

As vantagens do Fine-Tuning são muitas. Primeiro, ele aproveita o conhecimento previamente adquirido durante um longo treinamento, economizando de começar um modelo do zero. Além disso, precisa de menos dados do que começar do zero, o que é ótimo quando não há muitos dados disponíveis. Por fim, é menos custoso computacionalmente, democratizando o uso das LLMs para tarefas específicas.

No entanto, aprimorar Modelos de Linguagem de Grande Escala (LLMs) através do Fine-Tuning apresenta seus próprios desafios, sendo o principal a restrição de hardware. Naturalmente, os LLMs são modelos grandes, contendo bilhões de parâmetros que demandam uma quantidade considerável de recursos de computação para serem treinados. Isso implica que o Fine-Tuning de LLMs geralmente necessita de equipamentos de alta performance, que nem sempre estão ao alcance de todos os pesquisadores e desenvolvedores. Frequentemente, essa limitação de hardware atua como um obstáculo, gerando uma certa hesitação em se empenhar no Fine-Tuning de LLMs.

Neste contexto, a utilização da técnica LoRA (Low-Rank Adaptation) surge como uma solução potencial. LoRA, introduzida por [2], é uma abordagem que visa reduzir a complexidade computacional do Fine-Tuning de LLMs ao aplicar adaptações de baixo posto aos parâmetros do modelo. Essa técnica pode ajudar a contornar as restrições de hardware, tornando o processo de Fine-Tuning mais acessível e viável para uma gama mais ampla de pesquisadores e desenvolvedores, mesmo aqueles com recursos computacionais limitados.

A. *Low-Rank Adaptation (LoRA)*

LoRA (Low-Rank Adaptation) é uma técnica para ajustar modelos LLM como o GPT-3 de forma eficiente. Ela envolve congelar os pesos originais do modelo e introduzir um conjunto separado de pesos que representam as diferenças necessárias para uma tarefa específica. Esses pesos ajustados são armazenados separadamente, mas mesclados para inferência, evitando aumento nos requisitos de memória da GPU.

O diferencial da LoRA é a redução de parâmetros treináveis. Ao invés de ajustar matrizes de peso de alto posto e alta dimensão, a LoRA emprega uma decomposição de baixo posto, representando esses pesos como a multiplicação de duas matrizes menores. Esse método resulta em menos números para ajustar, melhorando a eficiência computacional. Após encontrar os pesos ajustados, eles são adicionados ao modelo original para melhorar o desempenho na tarefa especificada.

LoRA torna o processo de Fine-Tuning eficiente e acessível, especialmente no contexto deste projeto realizado no Google Colab com uma única GPU modelo A100 de 40GB de memória. Embora esta GPU seja potente, ela não se compara às necessidades de grandes modelos de linguagem. Utilizando LoRA, foi possível adaptar esses modelos avançados para funcionar em hardware menos potente, democratizando o acesso a essas ferramentas para pesquisas e projetos menores, destacando o valor prático da técnica no ambiente do Colab.

Nesse projeto, foi utilizado uma adaptação da técnica LoRA, o Quantized Low-Rank Adaptation (QLoRA) [1]. Essa adaptação é uma versão quantizada do algoritmo, utilizando-se de uma abordagem inovadora de alta precisão para quantizar o modelo pré-treinado para 4 bits.

V. Conclusão

Este projeto visou aprimorar a capacidade da LLM Canarim-7B, focada na língua portuguesa, com a

implementação do algoritmo Quantized Low-Rank Adaptation (QLoRA). O desafio central enfrentado neste processo foi a restrição de hardware, uma barreira comum no treinamento e no aprimoramento de LLMs avançadas, impossibilitando o uso de LLMs maiores.

O uso do QLoRA demonstrou ser uma solução eficaz para mitigar essas limitações, permitindo o Fine-Tuning do modelo Canarim-7B em um hardware menos poderoso, como o disponível no Google Colab. No entanto, é importante destacar que, embora o QLoRA ofereça uma abordagem mais acessível para o aprimoramento de LLMs, os modelos resultantes ainda não alcançam o mesmo nível de desempenho e precisão que seria possível com hardware mais robusto e modelos com maiores números de parâmetros.

A experiência deste projeto ressalta a necessidade contínua de inovações no campo da inteligência artificial, não apenas no desenvolvimento de técnicas de treinamento mais eficientes, mas também na busca de soluções de hardware que possam suportar as demandas crescentes de modelos de linguagem cada vez maiores e mais complexos. Embora tenhamos feito progressos significativos, o desafio de hardware permanece uma questão central, limitando o potencial total dessas tecnologias avançadas.

Em conclusão, o Projeto Themis contribuiu para a compreensão do direito brasileiro através da LLM Canarim-7B, demonstrando a viabilidade de aplicar LLMs em áreas especializadas como o direito. No entanto, a evolução contínua do hardware e das técnicas de treinamento será crucial para democratizar o potencial uso LLMs em aplicações práticas.

REFERENCES

[1] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms, 2023.

[2] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021.

[3] Maicon Domingues. canarim-7b (revision 08fdd2b), 2023.

[4] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023.

[5] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc.