UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Matheus Lisboa Oliveira dos Santos

AVALIAÇÃO DE GRANDES MODELOS DE LINGUAGEM QUANTIZADOS NA RESOLUÇÃO DE QUESTÕES DO ENEM

CAMPINA GRANDE - PB

2023

Matheus Lisboa Oliveira dos Santos

AVALIAÇÃO DE GRANDES MODELOS DE LINGUAGEM QUANTIZADOS NA RESOLUÇÃO DE QUESTÕES DO ENEM

Trabalho de Conclusão Curso apresentado ao Curso Bacharelado em Ciência da Computação do Centro de Engenharia Elétrica e Informática da Universidade Federal de Campina Grande, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

Orientador : Cláudio Elízio Calazans Campelo

CAMPINA GRANDE - PB

2023

Matheus Lisboa Oliveira dos Santos

AVALIAÇÃO DE GRANDES MODELOS DE LINGUAGEM QUANTIZADOS NA RESOLUÇÃO DE QUESTÕES DO ENEM

Trabalho de Conclusão Curso apresentado ao Curso Bacharelado em Ciência da Computação do Centro de Engenharia Elétrica e Informática da Universidade Federal de Campina Grande, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

BANCA EXAMINADORA:

Cláudio Elízio Calazans Campelo
Orientador – UASC/CEEI/UFCG

Cláudio de Souza Baptista
Examinador – UASC/CEEI/UFCG

Francisco Vilar Brasileiro
Professor da Disciplina TCC – UASC/CEEI/UFCG

Trabalho aprovado em: 28 de Junho de 2023.

CAMPINA GRANDE - PB

RESUMO

Embora os grandes modelos de linguagem (LLMs) representem uma revolução na forma como interagimos com computadores, permitindo a construção de perguntas complexas e a capacidade de raciocinar sobre uma sequência de declarações, seu uso é restrito devido à necessidade de hardware dedicado para a execução. Neste estudo, avaliamos o desempenho de LLMs baseados nos modelos LLaMA de 7 e 13 bilhões, submetidos a um processo de quantização e executados em hardware doméstico. Os modelos considerados foram alpaca, koala e vicuna. Para avaliar a eficácia desses modelos, desenvolvemos um banco de dados contendo 1006 perguntas do ENEM (Exame Nacional do Ensino Médio). Nossa análise revelou que o modelo de melhor desempenho alcançou uma acurácia de aproximadamente 40% tanto para os textos originais das perguntas em português quanto para suas traduções em inglês. Além disso, avaliamos a eficiência computacional dos modelos medindo o tempo necessário para a execução. Em média, os LLMs de 7 e 13 bilhões levaram aproximadamente 20 e 50 segundos, respectivamente, para processar as consultas em uma máquina equipada com um processador AMD Ryzen 5 3600x.

Benchmarking quantized LLaMa-based models on the Brazilian Secondary School Exam

ABSTRACT

Although large language models (LLMs) represent a revolution in the way we interact with computers allowing the construction of complex questions and the ability to reason over a sequence of statements, their use is restricted due to the need for dedicated hardware for execution. In this study we evaluate the performance of LLMs based on the 7 and 13 billion LLaMA models, subjected to a quantization process and run on home hardware. The models considered were alpaca, koala, and vicuna. To evaluate the effectiveness of these models, we developed a database containing 1006 questions from the ENEM (National High School Exam). Our analysis revealed that the best performing model achieved an accuracy of approximately 40% for both the original texts of the Portuguese questions and their English translations. In addition, we evaluated the computational efficiency of the models by measuring the time required for execution. On average, the 7 and 13 billion LLMs took approximately 20 and 50 seconds, respectively, to process the queries on a machine equipped with an AMD Ryzen 5 3600x processor.

Avaliação de grandes modelos de linguagem quantizados na resolução de questões do ENEM

Trabalho de Conclusão de Curso

Matheus Lisboa Oliveira dos Santos (Aluno), Cláudio Campelo (Orientador)
Departamento de Sistemas e Computação
Universidade Federal de Campina Grande
Campina Grande, Paraíba - Brasil

RESUMO

Although large language models (LLMs) represent a revolution in the way we interact with computers, allowing the construction of complex questions and the ability to reason over a sequence of statements, their use is restricted due to the need for dedicated hardware for execution. In this study, we evaluate the performance of LLMs based on the 7 and 13 billion LLaMA models, subjected to a quantization process and run on home hardware. The models considered were alpaca, koala, and vicuna. To evaluate the effectiveness of these models, we developed a database containing 1006 questions from the ENEM (National High School Exam). Our analysis revealed that the best performing model achieved an accuracy of approximately 40% for both the original texts of the Portuguese questions and their English translations. In addition, we evaluated the computational efficiency of the models by measuring the time required for execution. On average, the 7 and 13 billion LLMs took approximately 20 and 50 seconds, respectively, to process the queries on a machine equipped with an AMD Ryzen 5 3600x processor.

PALAVRAS-CHAVE

Grande modelos de linguagem, LLMs, ENEM, GGML, LLaMA, Quantização.

1 INTRODUÇÃO

Com a introdução do artigo *Attention is all you need* [12], o campo de processamento de linguagem natural (NLP) passou por uma revolução significativa. Tarefas que anteriormente eram dominadas por heurísticas e algoritmos de aprendizado de máquina começaram a obter resultados de ponta com o uso dos Transformers [14]. Essa arquitetura de redes neurais tem como principal objetivo dar atenção às partes mais relevantes das entradas, como palavras-chave ou áreas com pessoas em uma imagem, por exemplo.

Com o surgimento dos *Transformers*, uma classe de modelos de redes neurais que é treinada com o intuito de predizer a próxima palavra dada uma sequência de palavras anteriores, teve suas métricas elevadas no estado da arte. Essa categoria de modelos ficou conhecida como modelos de linguagem, e suas primeiras aplicações foram voltadas para a geração de *word embeddings* [1]. Essa técnica permite atribuir dinamicamente palavras a um espaço vetorial semântico, onde palavras semelhantes estão próximas umas das outras. Posteriormente, foram utilizadas arquiteturas de codificador-decodificador conhecidas como *seq2seq*, que faziam uso de *transformers* para alcançar o estado da arte em tarefas de codificação e decodificação de texto. Um exemplo notável é a tradução de textos entre diferentes idiomas, mesmo quando esses textos possuem comprimentos diferentes.

Com a introdução da família de modelos GPT (*Generative pre-trained transformer*), modelos treinados por meio de aprendizado não supervisionado ganharam popularidade. Esses modelos eram pré-treinados em grandes quantidades de dados não rotulados, e retinham um conhecimento geral em seu treinamento. Em seguida, eram treinados novamente com uma quantidade muito menor de dados e por períodos de tempo reduzidos para tarefas específicas. No entanto, o lançamento do Chat-GPT, um modelo treinado para interações humanas por meio de conversas, trouxe uma visibilidade ainda maior para esses modelos.

Esses modelos apresentaram uma inovação significativa na forma como ocorre a interação entre humanos e computadores, permitindo uma comunicação intuitiva por meio de diálogos, em que as respostas são precisamente adequadas ao que é solicitado. Isso resulta em uma economia significativa de tempo em comparação com a pesquisa tradicional em motores de busca. No entanto, é importante destacar que esses modelos não são *open-source*. Para o modelo Chat-GPT não é disponibilizado seu código-fonte ou dados que foram utilizados para treinar, o que impossibilita que pesquisadores realizem estudos sobre seu funcionamento interno. Além disso, essa falta de transparência pode ameaçar a validade dos experimentos desses modelos, visto que pode haver contaminação por dados disponíveis na internet.

No entanto, empresas como a Meta[1] têm adotado uma abordagem *open-source* ao disponibilizar grandes modelos de linguagem (*Large Language Models - LLMs*) como base para pesquisadores e entusiastas conduzirem suas pesquisas. Os modelos divulgados pela Meta possuem tamanhos de 7, 13, 30 e 65 bilhões de parâmetros. Embora esses modelos sejam considerados menores em comparação com a família GPT (por exemplo, o GPT-3.5 Turbo possui 154 bilhões de parâmetros), ainda é necessário dispor de hardware dedicado para executá-los, o que restringe a pesquisa a pessoas que têm acesso a esses recursos.

Contudo, como foi demonstrado por [15], é possível diminuir a quantidade de memória necessária para a utilização desses modelos com um processo de quantização. Esse processo visa diminuir a precisão dos pesos das camadas escondidas dos modelos, ao custo de perda de desempenho. Utilizando técnicas de quantização, um projeto visa utilizar uma API escrita do zero em C/C++ para execução de modelos, sem a necessidade de GPUs dedicadas[2]. Os

[1] https://about.meta.com/br/
[2] https://github.com/ggerganov/llama.cpp

modelos são baseados no LLaMA, publicado pela Meta [11], são eles: Vicuna[3], Koala[4] e Alpaca[5], todos possuem duas variantes, uma de 7 e outra de 13 bilhões de parâmetros. Isso permitiu que qualquer pessoa pudesse experimentar o potencial desses modelos, visto que seria possível executar inferências em hardware doméstico.

O Exame Nacional do Ensino Médio (ENEM) é uma prova que é prestada anualmente por alunos do ensino médio em todo o país, e serve como porta de entrada para faculdades em todo o Brasil, representando assim um desafio, que muitos estudantes se preparam o ano todo. Como demonstrado por [13] esses LLMs conseguem generalizar conhecimento, aumentando a quantidade de atividades que eles executar a medida que aumentam a quantidade de parâmetros. Dito isto, avaliar o desempenho desses grandes modelos de linguagem em questões do ENEM se torna um bom referencial do quão robustos esses grandes modelos são, visto esses são modelos de propósito geral, e não foram treinados para responder perguntas.

Consequentemente, o objetivo deste estudo consiste em avaliar modelos de linguagem quantizados, baseados no LLaMA [11], capazes de operar em hardware doméstico, utilizando questões do ENEM como cenário de análise. Para tal finalidade, produzimos um banco de dados de questões criteriosamente estruturado, contendo os textos das questões juntamente com as respostas corretas. A base de dados engloba um total de 1.006 questões, abrangendo o período de 2010 a 2022. A base produzida tem grande potencial para análises de grandes modelos de linguagem, e também para outros estudos no campo do processamento de linguagem natural.

Os experimentos conduzidos em nosso estudo visam responder as seguintes questões de pesquisa:

- **Q1** - Qual a eficácia de modelos quantizados, baseados no LLaMA, treinados em inglês, na resolução de questões do ENEM, descritas em português do Brasil?
- **Q2** - Qual a eficácia de modelos quantizados, baseados no LLaMA, treinados em inglês, na resolução de questões do ENEM, traduzidas de português do Brasil para o inglês?
- **Q3** - Qual a eficiência (em termos de tempo de execução em um computador de hardware modesto) de modelos quantizados, baseados no LLaMA, quando utilizados para resolução de questões do ENEM?

2 TRABALHOS RELACIONADOS

O uso de grandes modelos de linguagem está avançando muito rapidamente em muitos campos de pesquisa. Um deles foi a medicina, onde pesquisadores utilizaram o modelo PALM [3], treinado pela Google, para performar perguntas e respostas no domínio médico. Esse modelo foi avaliado no *United States Medical Licensing Examination* (USMLE) [9]. A análise dos resultados demostrou que o modelo retornou respostas que entraram em consenso com especialistas em 92.6% das questões, demonstrando que esses modelos podem ser bastante benéficos para ajudar médicos em seus atendimentos.

Como demonstrado por [10], já existem esforços no treinamento de grandes modelos de linguagem para resolução de questões. De acordo com o estudo comparativo disponibilizado pelos autores, o modelo deles apresentou um desempenho melhor que todos os outros modelos disponíveis no mercado, com exceção do GPT-4, para exames em inglês e chinês. O modelo foi avaliado nos seguintes conjuntos de dados: **MMLU, AGIEval e C-Eval**, e teve as seguintes métricas: **67.2, 49.2 e 62.7**, respectivamente; Contra **86.2, 56.4, 68.7** do GPT-4.

Adicionalmente, há relatos de pesquisas no treinamento de modelos de linguagem com o foco em criar uma cadeia de pensamentos, onde o modelo seja capaz de explicar o por quê das suas respostas [6]. Isso pode ajudar a criar modelos de linguagem que sejam cada vez mais capazes de fornecer respostas que sejam úteis para humanos. Pensando em um contexto de resposta de questões, um modelo que fosse capaz de explicar o raciocínio por trás da resposta a uma alternativa, seria muito útil a estudante, por exemplo.

Em um contexto brasileiro, uma equipe de pesquisadores propôs utilizar o GPT-4 [8] para avaliar questões do ENEM [7]. O modelo apresentou **87.29%** de acurácia nas questões do ano de 2022, contra **73.73%** de acurácia do gpt-3.5-turbo. Essa melhora se deu pelo aumento do tamanho do modelo, para comportar também imagens. Isso mostra que esses modelos conseguiram desempenhar melhor que grande parte dos humanos que prestam esse exame anualmente.

Modelos de linguagem quantizados estão em foco, visto a quantidade de recursos computacionais necessária para executá-los [4, 5]. Contudo, esses estudos abordam a avaliação utilizando métricas abstratas[6]. Esse trabalho visa avaliar esses modelos quantizados de maneira tangível, verificando o quão bem eles conseguem responder uma prova desafiadora como a do ENEM.

3 FUNDAMENTAÇÃO TEÓRICA

Esta seção introduz alguns conceitos relevantes para melhor compreensão do restante do artigo.

3.1 Grandes modelos de linguagem

Um dos fatores determinantes para a alta eficácia apresentada por alguns modelos de linguagem é o tamanho[13]. Por exemplo, o modelo GPT-3[2], publicado pela empresa OpenAI[7], tem 175 bilhões de parâmetros, resultantes de 34 dias de treinamento em 1.024 GPUs Nvidia A100. O custo estimado para esse treinamento foi de 4.6 milhões de dólares.

Para efeito de comparação, o modelo LLaMA [11] de 7 bilhões de parâmetros, publicado pela Meta, necessita de uma GPU com, pelo menos, 28 GB de memória para executar uma inferência[8]. Esses requisitos são proibitivos, visto que esses equipamentos custam caro.

3.2 LLaMA.cpp

LLaMA.cpp[9] é um projeto que visa criar uma API para inferência de grandes modelos de linguagem em CPU, usando C/C++ e técnicas que permitem que os modelos não sejam carregados completamente na memória. Os modelos são baseados no LLaMA[11], e podem executar em computadores domésticos.

[3]https://lmsys.org/blog/2023-03-30-vicuna/
[4]https://bair.berkeley.edu/blog/2023/04/03/koala/
[5]https://crfm.stanford.edu/2023/03/13/alpaca.html
[6]https://huggingface.co/docs/transformers/perplexity
[7]https://openai.com/
[8]https://discuss.huggingface.co/t/llama-7b-gpu-memory-requirement/34323
[9]https://github.com/ggerganov/llama.cpp

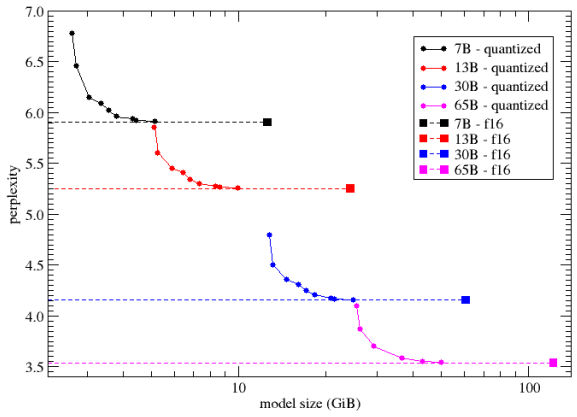

Figure 1: Perda de desempenho de modelos quantizados.

No entanto, é importante ressaltar que esses benefícios não são obtidos sem custos. Para permitir a execução do LLaMA.cpp, é necessário reduzir o tamanho dos modelos, o que é alcançado por meio da aplicação de uma técnica de quantização. Essa técnica envolve a compressão dos pesos nas camadas ocultas dos modelos, resultando em uma redução do espaço necessário para sua armazenagem. A Figura 1 ilustra que, à medida que o nível de quantização aumenta, ou seja, ocorre perda de precisão nas camadas, a métrica de perplexidade aumenta.

Para a condução dos experimentos descritos neste artigo, todos os modelos foram quantizados em Q4. De acordo com os autores do repositório, esse nível de quantização ocasiona uma piora de aproximadamente 2% na métrica de perplexidade. Maiores detalhes sobre o processo de quantização podem ser encontrados na Seção 3.3.

3.3 Processo de quantização dos modelos

O processo de quantização dos modelos utilizados no LLaMA.cpp é descrito no projeto Ggml[10]. Esse projeto visa comprimir diferentes modelos de linguagem, não só os baseados em LLaMA, quantizando também modelos da familia GPT, como GPT-2 e GPT-J.

Os pesos das camadas escondidas de um modelo sem quantização são representados como *floats* de 16 bits. No processo de quantização descrito em[11], um conjunto de *QK* pesos são representados como uma parte inteira mais uma parte em ponto flutuante. Por exemplo, para uma quantização de *Q4*, um bloco de 4 pesos de uma camada, cada um sendo eles representados em float16, são representados como um fator de escala float32 mais 2 inteiros de 2 bytes cada. Segundo o autor, essa abordagem reduziu em 75% o tamanho dos modelos.

[10]https://github.com/ggerganov/ggml
[11]https://github.com/ggerganov/ggml/pull/27

4 METODOLOGIA

Esta seção apresenta a metodologia adotada para avaliar os modelos. Discute-se como foi construída a base de dados para avaliação, os modelos utilizados, e os experimentos conduzidos.

4.1 Conjunto de dados

Uma das principais contribuições deste artigo é a disponibilização de uma base de dados estruturada e validada, composta por um grande número de questões do Exame Nacional do Ensino Médio (ENEM).

As questões consistem basicamente de três partes: a primeira sendo uma porção de texto, tabelas ou imagens, ou uma combinação delas. A segunda parte sendo uma pergunta sobre a primeira parte. E por fim cinco alternativas, sendo só uma delas correta.

Essa base de dados foi desenvolvida com foco nas questões que podem ser respondidas apenas com base em texto, uma vez que os modelos que serão avaliados possuem capacidade de compreensão textual. No total, a base de dados contém 1.006 questões, nas quais foram identificados os textos de descrição, as alternativas e as respostas corretas. O processo de coleta dessas questões seguiu o seguinte procedimento:

- Coleta das provas do ENEM, de 2010 a 2022, em formato PDF, obtidas do Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP)[12].
- Utilização de ferramenta[13] para extração de texto de cada arquivo PDF.
- Definição de heurística para concatenação do texto de cada questão, agrupando descrição, pergunta e alternativas.
- Filtragem de questões que não se enquadravam no escopo dos experimentos.

Os seguintes critérios foram estabelecidos para remoção de questões não adequadas aos experimentos:

- Questões contendo alguma imagem, tabela ou equação; visto que os modelos que utilizaremos só conseguem compreender texto.
- Qualquer questão que não fosse possível distinguir quais partes do texto eram as alternativas, visto que essa parte era de suma importância para os modelos.
- Questões que não foram processadas adequadamente pela ferramenta de extração de conteúdo de arquivos PDF. Essas questões apresentavam, por exemplo, caracteres estranhos em seu conteúdo.

Para remover questões que contém imagens, tabelas ou equações, foram utilizadas heurísticas para verificar se dentro da questão existe alguma das palavras-chave, como: **tabela, quadro, figura, imagem**. Com isso, conseguimos remover uma grande quantidade de questões que seriam impossíveis para os modelos responderem.

A distribuição dessas questões por ano é exibida na Figura 2. Não foram extraídas questões para os anos de 2010 e 2021, por problemas na leitura do PDF. A distribuição das questões por área de conhecimento pode ser visualizada na Figura 3. Nesta figura, é

[12]https://www.gov.br/inep/pt-br/areas-de-atuacao/avaliacao-e-exames-educacionais/enem/provas-e-gabaritos
[13]https://pymupdf.readthedocs.io/en/latest/document.html

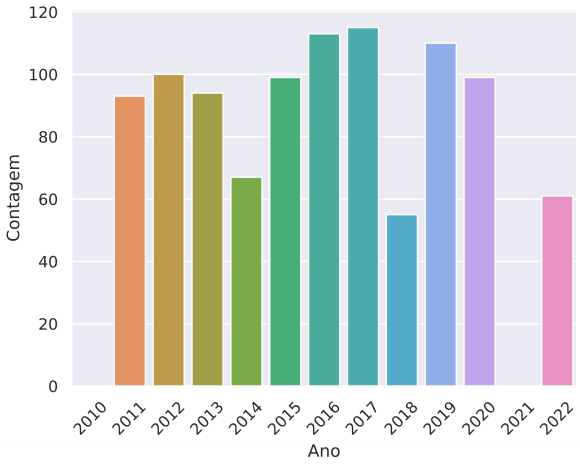

Figure 2: Contagem de questões extraídas por ano

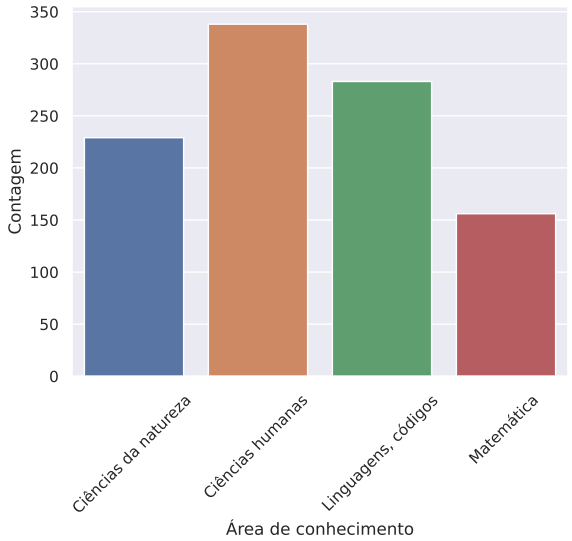

Figure 3: Distribuição de questões por área de conhecimento

possível notar que matemática e ciências da natureza e suas tecnologias foram as áreas com menos questões, decorrente da filtragem de questões que contém gráficos, equações e tabelas.

A anotação das respostas foi realizada manualmente, com base nos gabaritos disponibilizados em formato PDF, sendo registradas em um arquivo em formato *Json*. Preferimos uma abordagem manual, uma vez que a implementação de um *script* para automação seria muito custosa, visto que os arquivos PDF possuem uma estrutura bastante variada.

O *dataset* produzido está disponível gratuitamente, assim como os artefatos utilizados para sua produção (arquivos em formato PDF e código fonte dos *scripts* de processamento e transformação dos dados) em[14].

4.2 Modelos avaliados

Foram selecionados modelos de linguagem que estivessem alinhados com o objetivo do estudo, ou seja, modelos de grande porte capazes de serem executados em máquinas domésticas. Os modelos foram obtidos no repositório de modelos do HuggingFace[15], e foram disponibilizados por usuários que realizaram a quantização. Os modelos foram testados para verificar se são compatíveis com LLaMA.cpp[16]. Essa ferramenta proporciona a execução de modelos baseados no LLaMA [11] em máquinas domésticas, por meio do emprego de técnicas de quantização e leitura seletiva das partes necessárias para a execução do modelo.

Para os experimentos, foram utilizados modelos baseados no LLaMA de 7 e 13 bilhões de parâmetros, resultantes do ajuste fino (fine-tuning) dos modelos originais. São eles:

- **LLaMA 7b, 13b**: Modelos treinados do zero em um conjunto de dados diverso, que vem de várias fontes. São elas: **English CommonCrawl, C4, Github, Wikipedia, Gutenberg and Books3, ArXiv e Stack Exchange**. Esse conjunto de dados contém aproximadamente 1,4 trilhão de tokens, mas, para os modelos de 7 e 13 bilhões de parâmetros foi utilizado um subconjunto de 1 trilhão.
- **Alpaca 7b, 13b**: Modelos resultantes do fine-tuning dos modelos LLaMA com um conjunto de 52 mil exemplos de perguntas e respostas, esse modelo foi treinado para performar melhor em cenários de perguntas e respostas.
- **Koala 7b, 13b**: Fine-tuning dos modelos LLaMA, mas treinados 117 mil iterações de usuários com o ChatGPT[17]. Esse modelo foi treinado para desempenhar melhor em diálogos.
- **Vicuna 7b, 13b**: Fine-tuning dos modelos LLaMA, mas treinados com um conjunto de 70 mil iterações dos usuários com o ChatGPT, através do conjunto ShareGPT, que são conversações cedidas pela comunidade com o modelo.

Um ponto a se destacar é que os dados utilizados para treinar esses modelos são em sua grande maioria em inglês, e também não foram encontrados indícios de que esses modelos teriam sido expostos a dados de questões do ENEM durante suas etapas de treino ou validação, o que poderia invalidar os resultados apresentados na Seção 5.

4.3 Definição dos experimentos

Como já foi discutido, esses modelos de linguagem só conseguem receber uma porção de texto na entrada e retornar outra porção de texto na sua saída, então uma parte integral da atividade é definir o formato do texto que vai ser utilizado para alimentá-los. Para a elaboração dos *prompts*, foi adotada a metodologia proposta no curso *Prompt Engineering*[18], disponibilizado pela OpenAI, empresa que publicou os modelos da familia GPT. Embora o curso seja focado nos modelos GPT, grande parte dos modelos que utilizamos nesse

[14]https://github.com/wineone/tcc-matheus-lisboa
[15]https://huggingface.co/models
[16]https://github.com/ggerganov/llama.cpp
[17]https://openai.com/blog/chatgpt
[18]https://www.deeplearning.ai/short-courses/chatgpt-prompt-engineering-for-developers/

```
Qual é a alternativa correta para a questão que está entre <>? responda só com a letra que
representa a alternativa:

<Questão 57) No caso do Departamento de Defesa dos Estados Unidos, a ênfase está posta no
traçado de uma estratégia geral de desarticulação, não só dos inimigos reais como dos pote
nciais, inserida na concepção preventiva que supõe que a mínima dissidência é um sinal de
perigo e de guerra futura. Deve-se ter capacidade para responder a uma guerra convencional
tanto quanto para enfrentar um inimigo difuso, atentando simultaneamente para todas as áre
as geográficas do planeta. Trata-se, sem dúvida, da estratégia com pretensões mais abrange
ntes que se desenvolveu até agora.
CECEÑA, A. E. Hegemonias e emancipações no século XXI.
Tomando o texto como parâmetro, qual tendência contemporânea impulsiona a formulação de es
tratégias mais abrangentes por parte do Estado americano?
A) Erradicação dos conflitos em territórios.
B) Propagação de organizações em redes.
C) Eliminação das diferenças regionais.
D) Ampliação de modelo democrático.
E) Projeção da diplomacia mundial.>
```

Figure 4: Exemplo de questão que foi utilizada para avaliação.

experimento tem como base dados que foram extraídos de conversação com o Chat-GPT, então é esperado que o funcionamento dos modelos que utilizamos sejam em certo grau parecidos com as metodologias disponibilizadas no curso. A abordagem seguida foi a de pedir para o modelo responder a alternativa correta, e sinalizar somente a letra da alternativa, para facilitar a verificação da efetividade dos modelos e o cômputo das métricas de avaliação. A Figura 4 exibe um exemplo de *prompt*.

Para realizar a comparação dos modelos, foram executados dois experimentos. O primeiro, visando responder a **Q1**, comparou a acurácia dos modelos executando todos os modelos em todas as questões, substituindo o texto das questões no *prompt* e coletando o resultado dos modelos em texto. O segundo experimento foi pensado para responder a **Q2**, para isso foram traduzidas todas as questões, e também o *prompt*, e foram computadas todas as respostas. Para tradução foi utilizada a biblioteca TextBlob[19].

Também foram avaliados os tempos de execução para esses modelos, visando responder **Q3**. A avaliação foi conduzida utilizando duas máquinas, sendo uma equipada com processador AMD Ryzen 5 3600x e outra com processador Intel i9 9900k. Foi coletado o tempo em segundos para a inferência das questões em português e inglês, sendo as questões em português executadas na máquina com o 3600x e as questões em inglês na máquina equipada com o 9900k. Os resultados são apresentados na Seção 5.

4.4 Avaliação dos modelos

Com o intuito de avaliar a assertividade dos modelos ao responder as questões das provas, nós adotamos a métrica de acurácia, que é definida como a quantidade de questões corretas, dividida pela quantidade total de questões.

$$acc = \frac{\#correct}{\#total} \tag{1}$$

1: Acurácia dos modelos

Um dos problemas encontrados foi como identificar qual alternativa foi a predita pelo modelo, diante da natureza geracional de texto. Para a grande maioria dos *prompts*, o modelo apresentou uma saída bem objetiva, contendo apenas uma letra representando algumas das alternativa possíveis (A, B, C, D, E). Porém, em outras situações, a saída do modelo consistiu em um texto sobre a questão, seguido da letra representando a reposta. Além dessas, foram observadas ainda saídas contendo textos longos, sem muito sentido e sem conter uma resposta objetiva. Diante disso, foi definido um conjunto de heurísticas para capturar a alternativa selecionada pelo modelo. A Tabela 1 apresenta a porcentagem de questões que tiveram uma alternativa corretamente atribuída. Uma inspeção manual foi executada para garantir que as heurísticas identificaram todas as alternativas que estivessem disponíveis.

[19]https://pypi.org/project/textblob/

Table 1: Cobertura de questões com alternativa identificada

Experimento	% de cobertura
Execução em português	0.995
Execução em inglês	0.990

5 RESULTADOS E DISCUSSÕES

Esta seção apresenta e discute os resultados observados a partir dos experimentos conduzidos. Serão respondidas as questões de pesquisa:

- **Q1** - Qual a eficácia dos modelos em questões em português?
- **Q2** - Qual a eficácia dos modelos em questões traduzidas para inglês?
- **Q3** - Qual o tempo necessário para executar esses modelos em máquinas domésticas?

5.1 Q1 e Q2 - Qual a eficácia dos modelos em questões descritas em português e inglês.

Abordando as **Q1** e **Q2**, foi avaliada a acurácia dos modelos no conjunto de questões. Na Tabela 2, é apresentado o desempenho dos modelos. Observa-se que alguns modelos, como LLaMA 7 e 13b, Alpaca 7b e Koala 7b e 13b, obtiveram um desempenho semelhante ao de um classificador aleatório. Isso sugere que esses modelos podem não ser capazes de compreender adequadamente as perguntas e fornecer as respostas corretas, tanto em inglês quanto em português. No entanto, eles demonstraram uma capacidade de reconhecer que o texto fornecido se trata de uma questão e foram capazes de indicar uma alternativa, mesmo que incorreta.

Durante a fase de inferência, foi observado um fenômeno de viés nos modelos analisados. A maioria desses modelos demonstrou uma tendência consistente em gerar uma única opção como resultado. A distribuição percentual das questões identificadas em português durante essa fase para cada modelo é ilustrada na Figura 5, enquanto a distribuição para o idioma inglês é representada na Figura 6. Com exceção dos modelos Llama 7b, Vicuna 7b e Vicuna 13b, todos os demais apresentaram um viés significativo para a alternativa A, contrariando a expectativa de uma distribuição equilibrada entre todas as opções. Notavelmente, o modelo Vicuna 13b exibiu um viés em direção à alternativa B para ambos os idiomas, enquanto o modelo Llama 7b mostrou um viés para a alternativa D no idioma português e para a alternativa B no inglês. O modelo Vicuna de 7 bilhões de parâmetros foi identificado como aquele com o menor viés, pois não apresentou uma inclinação significativa para nenhuma das opções ou idiomas. Ainda assim, os modelos pareceram apresentar um viés

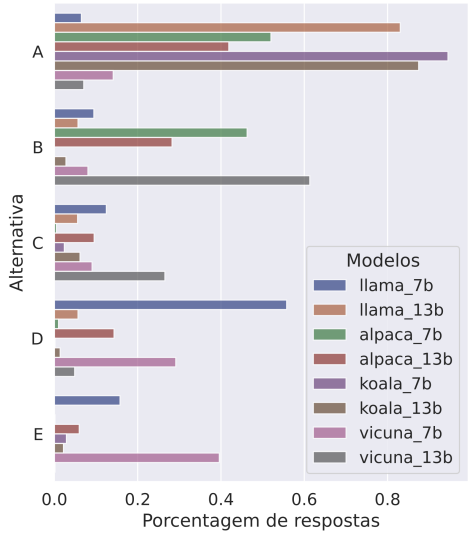

Figure 5: Distribuição das alternativas, questões em português

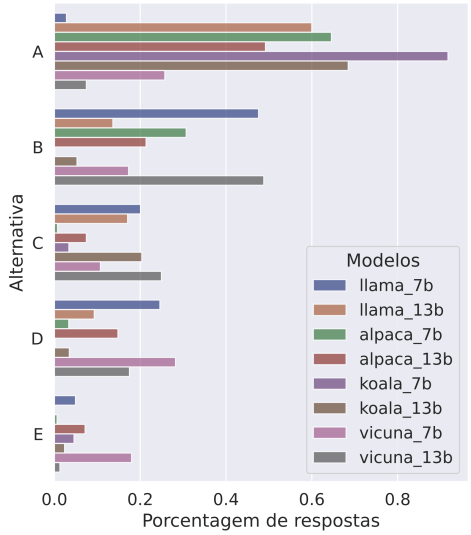

Figure 6: Distribuição das alternativas, questões em inglês

mais pronunciado em relação ao português, enquanto mostraram um viés menos acentuado em relação ao inglês.

No entanto, os modelos Alpaca 13b, Vicuna 7b e 13b apresentaram um desempenho significativamente superior, com o Vicuna de 7b alcançando uma taxa de acurácia de aproximadamente 40% para o idioma inglês e o Alpaca de 13b atingindo 40% de acurácia para o idioma português. Embora esses resultados estejam distantes dos relatados por [7] para o Chat-GPT, eles são bastante promissores, considerando que esses modelos são alternativas de código

Table 2: Acurácia geral dos modelos

Modelo	Linguagem	
	Português	Inglês
LLaMA 7b	0.225	0.251
LLaMA 13b	0.207	0.230
Alpaca 7b	0.203	0.205
Alpaca 13b	**0.400**	0.339
Koala 7B	0.183	0.193
Koala 13b	0.243	0.289
Vicuna 7b	0.327	**0.399**
Vicuna 13b	0.336	0.397

aberto, foram submetidos a quantização e podem ser executados em máquinas domésticas, sem a necessidade de hardware especializado. No entanto, uma hipótese que não foi confirmada é a de que esses modelos teriam um desempenho melhor nas questões em inglês. Surgiu a suspeita de que a qualidade das traduções pode não ter sido tão boa, portanto, como próximos passos, será necessário avaliar outras ferramentas de tradução de texto.

Para observar melhor a capacidade dos modelos, foram comparadas também as métricas nas quatro áreas de conhecimento da prova do ENEM, são elas: Ciências humanas e suas tecnologias, Ciências da natureza e suas tecnologias, Matemática e suas tecnologias, Linguagens, Linguagens, códigos e suas tecnologias. As métricas podem ser encontradas na Tabela 3. Tanto para português quanto para inglês, os modelos conseguiram performar muito bem nas áreas de ciências humanas e suas tecnologias e códigos e suas tecnologias, com o Vicuna 7b tendo 50% e 43% de acurácia, respectivamente; na área de ciências da natureza o resultado piorou um pouco, com o Vicuna 7b conseguindo 33%; já na área de matemática e suas tecnologias foi onde todos os modelos não conseguiram performar nada bem, com os modelos não passando de aproximadamente 24% para português e aproximadamente 26% de acurácia para o inglês, resultados que, em alguns casos, são piores que modelos aleatórios.

5.2 Q3 - Qual a eficiência dos modelos, em termos de tempo de execução?

Outro fator de grande importância para avaliação desses modelos é tempo de execução das inferência realizadas, visto que esses modelos são grandes modelos de linguagem. Para responder à Q3, foram conduzidos 2 experimentos. Em cada um deles, todos os modelos realizaram uma inferência para cada uma das questões do conjunto de dados. Durante a execução, foram computados os tempos para realização das inferências (em segundos). Foram utilizadas duas máquinas, uma equipada com um AMD Ryzen 5 3600x e outra equipada com um Intel i9 9900k. A Tabela 4 tem os tempos médios para execução das questões.

Os modelos com 13 bilhões demoram consistentemente mais que modelos de 7 bilhões de parâmetros. Porém, considerando que esses modelos não necessitam de GPUs dedicadas, esses tempos de execução não são proibitivos, e permitem o uso desse grandes modelos de linguagem por qualquer interessado.

Table 3: Acurácia dos modelos por área de conhecimento

Modelo	Ciências humanas e suas tecnologias		Ciências da natureza e suas tecnologias		Matemática e suas tecnologias		Linguagens, códigos e suas tecnologias	
	Português	Inglês	Português	Inglês	Português	Inglês	Português	Inglês
LLaMA 7b	0.221	0.292	0.227	0.209	0.217	**0.262**	0.233	0.262
LLaMA 13b	0.204	0.248	0.205	0.222	0.141	0.179	0.250	0.243
Alpaca 7b	0.233	0.201	0.205	0.213	0.128	0.141	0.208	0.240
Alpaca 13b	**0.473**	0.426	**0.375**	**0.366**	0.205	0.121	**0.441**	0.335
Koala 7B	0.180	0.201	0.196	0.196	0.121	0.134	0.212	0.215
Koala 13b	0.266	0.360	0.213	0.253	0.128	0.141	0.303	0.314
Vicuna 7b	0.384	**0.508**	0.262	0.331	**0.243**	0.198	0.356	**0.434**
Vicuna 13b	0.408	0.500	0.275	0.349	**0.243**	0.256	0.353	0.392

Table 4: Tempo médio de inferência para os modelos (segundos)

Modelo	Amd ryzen 5 3600x	Intel i9 i9900k
LLaMA 7b	18.3	16.8
LLaMA 13b	35.5	27.9
Alpaca 7b	20.5	17.2
Alpaca 13b	45.3	34.2
Koala 7B	21.4	15.9
Koala 13b	41.2	33.7
Vicuna 7b	21.7	17.0
Vicuna 13b	64.2	46.0

6 CONCLUSÕES E TRABALHOS FUTUROS

Este estudo apresentou uma base de dados para avaliação de modelos de linguagem em português, oferecendo uma contribuição para pesquisas futuras. Além disso, realizamos uma avaliação de modelos de linguagem quantizados, que podem ser executados em hardware doméstico, ampliando a disseminação e acessibilidade desses modelos, que representam uma revolução no campo de processamento de linguagem natural.

Embora os resultados possam parecer pouco expressivos, é importante salientar que esses modelos de linguagem são significativamente menores e foram treinados com uma quantidade inferior de dados em comparação com as opções disponíveis no mercado, que são de código fechado. Apesar dessas limitações, os resultados indicam que os modelos *open-source* estão progredindo rapidamente, e espera-se que melhorem seu desempenho em tarefas dessa natureza.

Este artigo tem o intuito de fornecer uma base para pesquisas futuras, e, portanto, apresentamos algumas ideias que surgiram durante o desenvolvimento do estudo. São elas:

- **Expansão da base de dados**: A fim de restringir o escopo deste estudo, foram consideradas apenas as provas do ENEM dos anos de 2010 a 2022. No entanto, acreditamos que os *scripts* gerados possam ser generalizados para outros anos do ENEM, ampliando ainda mais essa base de dados.
- **Avaliação desses modelos em outras bases de dados**: Uma tarefa similar seria avaliar esses modelos em questões de concursos públicos. Porém, como ocorrem inúmeros concursos anualmente, as provas dessas seleções podem ser utilizadas para construir uma base de dados ainda mais abrangente e robusta.
- **Treinamento de modelos**: A base de dados disponibilizada contém uma quantidade considerável de questões. Seria interessante explorar a possibilidade de treinar esses modelos de linguagem para executar a tarefa de responder a perguntas.
- **Considerar outros modelos**: Como demonstrado em [10], já existem modelos treinados com o propósito de explicar qual o seu raciocínio para responder perguntas. Diante disso, experimentos futuros podem analisar em mais profundidade o racional que levou o modelo a uma determinada resposta.
- **Avaliar outras ferramentas de tradução**: Uma hipótese que não foi verificada foi que os modelos performariam melhor em um conjunto de dados em inglês, visto que esses modelos foram treinados com a grande maioria dos dados nessa linguagem. Contudo, os resultados foram muito parecidos, então foi levantada a hipótese de que a tradução não sido muito boa. Com isto, uma atividade que pode ser frutífera é avaliar outras ferramentas de tradução, para verificar se, de fato, há uma discrepância entre as linguagens.
- **Considerar modelos multimodais**: Conforme demonstrado em [7], o modelo GPT-4 obteve um desempenho impressionante nas questões do ENEM, em parte devido à sua capacidade de processar informações visuais em conjunto com o texto. Acredita-se que modelos multimodais desse tipo estejam disponíveis em código aberto em um futuro próximo.
- **Investigar os vieses dos modelos**: Através dos experimentos realizados neste estudo, não foi possível compreender a razão para os vieses observados no comportamento dos modelos. Portanto, em futuras investigações, esse fenômeno pode ser investigado de forma aprofundada.

7 AGRADECIMENTOS

Um primeiro agradecimento vai aos meus pais, Aureni e Fabiano, por acreditarem que na educação existe um caminho transformador.

Um segundo agradecimento vai a todo o corpo que compõe a UFCG, em especial ao meu orientador, Cláudio. Sem cada um deles meu caminho até aqui não seria possível.

O agradecimento final vai a *Rafaela*, com quem dividi todos os momentos da graduação, que para mim foi muito divertida. *Te amo do tamanho de um compitador.*

REFERÊNCIAS BIBLIOGRÁFICAS

[1] Felipe Almeida and Geraldo Xexéo. 2023. Word Embeddings: A Survey. arXiv:cs.CL/1901.09069

[2] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. arXiv:cs.CL/2005.14165

[3] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, and Et. Al. Paul Barham. 2022. PaLM: Scaling Language Modeling with Pathways. arXiv:cs.CL/2204.02311

[4] Jeonghoon Kim, Jung Hyun Lee, Sungdong Kim, Joonsuk Park, Kang Min Yoo, Se Jung Kwon, and Dongsoo Lee. 2023. Memory-Efficient Fine-Tuning of Compressed Large Language Models via sub-4-bit Integer Quantization. arXiv:cs.LG/2305.14152

[5] Sehoon Kim, Coleman Hooper, Amir Gholami, Zhen Dong, Xiuyu Li, Sheng Shen, Michael W. Mahoney, and Kurt Keutzer. 2023. SqueezeLLM: Dense-and-Sparse Quantization. arXiv:cs.CL/2306.07629

[6] Subhabrata Mukherjee, Arindam Mitra, Ganesh Jawahar, Sahaj Agarwal, Hamid Palangi, and Ahmed Awadallah. 2023. Orca: Progressive Learning from Complex Explanation Traces of GPT-4. arXiv:cs.CL/2306.02707

[7] Desnes Nunes, Ricardo Primi, Ramon Pires, Roberto Lotufo, and Rodrigo Nogueira. 2023. Evaluating GPT-3.5 and GPT-4 Models on Brazilian University Admission Exams. arXiv:cs.CL/2303.17003

[8] OpenAI. 2023. GPT-4 Technical Report. arXiv:cs.CL/2303.08774

[9] Karan Singhal, Shekoofeh Azizi, Tao Tu, S. Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, Perry Payne, Martin Seneviratne, Paul Gamble, Chris Kelly, Nathaneal Scharli, Aakanksha Chowdhery, Philip Mansfield, Blaise Aguera y Arcas, Dale Webster, Greg S. Corrado, Yossi Matias, Katherine Chou, Juraj Gottweis, Nenad Tomasev, Yun Liu, Alvin Rajkomar, Joelle Barral, Christopher Semturs, Alan Karthikesalingam, and Vivek Natarajan. 2022. Large Language Models Encode Clinical Knowledge. arXiv:cs.CL/2212.13138

[10] InternLM Team. 2023. InternLM: A Multilingual Language Model with Progressively Enhanced Capabilities. https://github.com/InternLM/InternLM-techreport.

[11] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. LLaMA: Open and Efficient Foundation Language Models. arXiv:cs.CL/2302.13971

[12] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. arXiv:cs.CL/1706.03762

[13] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent Abilities of Large Language Models. *Transactions on Machine Learning Research* (2022). https://openreview.net/forum?id=yzkSU5zdwD Survey Certification.

[14] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Association for Computational Linguistics, Online, 38–45. https://doi.org/10.18653/v1/2020.emnlp-demos.6

[15] Zhewei Yao, Xiaoxia Wu, Cheng Li, Stephen Youn, and Yuxiong He. 2023. ZeroQuant-V2: Exploring Post-training Quantization in LLMs from Comprehensive Study to Low Rank Compensation. arXiv:cs.LG/2303.08302