UNIVERSIDADE ESTADUAL PAULISTA "JÚLIO DE MESQUITA FILHO"
FACULDADE DE CIÊNCIAS - CAMPUS BAURU
DEPARTAMENTO DE COMPUTAÇÃO
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

NICOLE BARBOSA GOMES

DESENVOLVIMENTO DE UMA APLICAÇÃO PARA ANÁLISE DE DESEMPENHO DE GRANDES MODELOS DE LINGUAGEM

BAURU
Novembro/2023

NICOLE BARBOSA GOMES

DESENVOLVIMENTO DE UMA APLICAÇÃO PARA ANÁLISE DE DESEMPENHO DE GRANDES MODELOS DE LINGUAGEM

Trabalho de Conclusão de Curso do Curso de Bacharelado em Ciência da Computação da Universidade Estadual Paulista "Júlio de Mesquita Filho", Faculdade de Ciências, Campus Bauru.
Orientador: Prof. Dr. João Paulo Papa

BAURU
Novembro/2023

G633d

Gomes, Nicole Barbosa
Desenvolvimento de uma aplicação para análise de desempenho de grandes modelos de linguagem / Nicole Barbosa Gomes. -- Bauru, 2023
50 f. : il.

Trabalho de conclusão de curso (Bacharelado - Ciência da Computação) - Universidade Estadual Paulista (Unesp), Faculdade de Ciências, Bauru
Orientador: João Paulo Papa

1. Processamento de linguagem natural (Computação). 2. Aprendizado do computador. 3. Inteligência Artificial. I. Título.

Sistema de geração automática de fichas catalográficas da Unesp. Biblioteca da Faculdade de Ciências, Bauru. Dados fornecidos pelo autor(a).

Essa ficha não pode ser modificada.

Nicole Barbosa Gomes

Desenvolvimento de uma Aplicação para Análise de Desempenho de Grandes Modelos de Linguagem

Trabalho de Conclusão de Curso do Curso de Bacharelado em Ciência da Computação da Universidade Estadual Paulista “Júlio de Mesquita Filho”, Faculdade de Ciências, Campus Bauru.

Banca Examinadora

Prof. Dr. João Paulo Papa
Orientador
Universidade Estadual Paulista "Júlio de Mesquita Filho"
Faculdade de Ciências
Departamento de Ciência da Computação

Profa. Dra. Simone das Graças Domingues Prado
Universidade Estadual Paulista "Júlio de Mesquita Filho"
Faculdade de Ciências
Departamento de Ciência da Computação

Prof. Dr. Kelton Augusto Pontara da Costa
Universidade Estadual Paulista "Júlio de Mesquita Filho"
Faculdade de Ciências
Departamento de Ciência da Computação

Bauru, 16 de Novembro de 2023.

Dedico esse trabalho à minha família e aos meus amigos, por estarem comigo em todos os passos do caminho.

Agradecimentos

Agradeço, principalmente, aos meus pais, por tudo que me proporcionaram e por serem meus maiores apoiadores sempre. Eu não seria nada sem eles. Também ao meu irmão e aos membros da minha família, presentes em vários momentos da minha jornada. Em especial, a minha avó que, desde o começo, torceu muito por mim.

Sou grata as minhas amigas que me ouviram quando precisei e, mesmo com a distância, foram muito importantes nesse caminho. Aos meus amigos de faculdade, agradeço por estarem sempre comigo e por deixarem os dias difíceis um pouco mais fáceis.

Agradeço também ao meu orientador por todas as oportunidades e pelos ensinamentos. Aos demais professores do curso, sou grata por terem contribuído para a minha formação como profissional e, também, como pessoa.

Sou grata por todas essas pessoas que, de alguma forma, fizeram parte da minha vida. Sem elas, eu não poderia ter chegado até aqui.

"Assim como os nomes têm poder, as palavras têm poder. Palavras podem acender fogo nas mentes dos homens. Palavras podem arrancar lágrimas dos corações mais duros. Há sete palavras que farão uma pessoa te amar. Há dez palavras que quebrarão a vontade de um homem forte."

O Nome do Vento, Patrick Rothfuss

Resumo

A linguagem é uma necessidade humana desde sua origem. Com métodos mais rudimentares ou mais rebuscados, o ser humano encontra maneiras de se comunicar. A partir do surgimento dos computadores e das mudanças proporcionadas por eles, um novo desafio também chegou: descobrir como gerar conversações com as máquinas. Os estudos a esse respeito levaram a invenções importantes que provocaram ideias antes consideradas impossíveis. A área de aprendizado de máquina foi responsável pela maior parte desses avanços e, graças ao ramo, modelos de redes neurais artificiais foram cada vez mais explorados. Como um dos resultados, os *large language models*, caracterizados por serem compostos por essas redes, apresentaram resultados admiráveis ao produzir textos muito parecidos com os escritos por humanos. O sucesso desses modelos despertou o interesse de empresas que passaram a desenvolver suas próprias versões. A partir disso, cresce a demanda por entender quais deles são mais apropriados para certas aplicações, e outras questões de desempenho. Com esse propósito, o projeto buscou desenvolver um site que pudesse reunir *large language models* a fim de que gerassem respostas baseadas em instruções do usuário. Além disso, a aplicação deveria ser capaz de receber *feedbacks* sobre o melhor texto para uma instrução específica e, dessa maneira, possibilitar análises dos dados coletados para descobrir os pontos fracos e fortes de cada modelo. No final do desenvolvimento, esses objetivos foram atingidos e o site se tornou uma ferramenta intuitiva, para atender a um grande público, e proveitosa para captura de informações relevantes e estudo dos retornos recebidos.

Palavras-chave: *Large Language Models*; Modelos de linguagem; Inteligência artificial; Aprendizado de Máquina; *Transformers*.

Abstract

Language has been a human necessity since its origin. With methods ranging from rudimentary to more elaborate, humans have found ways to communicate. With the advent of computers and the changes they brought, a new challenge emerged: discovering how to engage in conversations with machines. Studies in this regard led to important inventions that sparked ideas once considered impossible. The field of machine learning has been responsible for most of these advances, and, thanks to this, artificial neural network models have been increasingly explored. As a result, large language models, characterized by being composed of these networks, have shown remarkable results in producing texts very similar to those written by humans. The success of these models has piqued the interest of companies that began to develop their own versions. From there, the demand has grown to understand which of these models are more suitable for specific applications and other performance-related questions. To address this, the project aimed to develop a website that could bring together large language models in order for them to generate responses based on user instructions. Additionally, the application should be capable of receiving feedback on the best text for a specific instruction, thereby enabling analysis of the collected data to discover the strengths and weaknesses of each model. By the end of development, these objectives were achieved, and the website became an intuitive tool to serve a wide audience, proving to be valuable for capturing relevant information and studying the received feedback.
Keywords: Large Language Models; Language Models; Artificial Intelligence; Machine Learning; Transformers.

Lista de figuras

Figura 1 – Exemplo do funcionamento do mecanismo de *self-attention* 19
Figura 2 – Funcionamento de um modelo de linguagem 21
Figura 3 – Representação do conceito de *transfer learning* 22
Figura 4 – Exemplo de uso de *prompt design* 23
Figura 5 – Interface do Figma 25
Figura 6 – Interface do Visual Studio Code 26
Figura 7 – Interface do XAMPP 26
Figura 8 – Interface do MySQL WorkBench 27
Figura 9 – Desenvolvimento do protótipo no Figma 29
Figura 10 – Variáveis usadas no arquivo PHP 30
Figura 11 – Envio das informações para o banco de dados 30
Figura 12 – Página inicial do site 31
Figura 13 – Seção das empresas dos modelos 32
Figura 14 – Seção dos modelos implementados 32
Figura 15 – Seção de dicas para as instruções 33
Figura 16 – Ação de um dos botões de dicas para instruções 33
Figura 17 – Seção de propriedades dos modelos 34
Figura 18 – Alterar propriedades 35
Figura 19 – Seção de comparação dos modelos 35
Figura 20 – Código da requisição para a API da empresa 36
Figura 21 – Código do cálculo do tempo de resposta de um modelo 37
Figura 22 – Espaço dedicado para a resposta de uma empresa 37
Figura 23 – Código da função para mostrar a resposta 38
Figura 24 – Seção de *feedback* do usuário 38
Figura 25 – Espaço para justificativa da escolha da resposta 39
Figura 26 – Rodapé do site 39
Figura 27 – Perguntas do formulário utilizado 42
Figura 28 – Parte da planilha correspondente as respostas para a primeira pergunta 42
Figura 29 – Parte da planilha correspondente as respostas para a segunda pergunta 43
Figura 30 – Gráfico dos modelos escolhidos 43
Figura 31 – Gráfico das empresas escolhidas 44
Figura 32 – Gráfico dos tempos de resposta (baseados na ocorrência) 44
Figura 33 – Gráfico dos modelos escolhidos 45
Figura 34 – Gráfico das empresas escolhidas 45
Figura 35 – Gráfico dos tempos de resposta (baseados na ocorrência) 46
Figura 36 – Gráfico dos tamanhos das respostas escolhidas 46

Lista de quadros

Quadro 1 – Especificações do notebook usado 24

Lista de abreviaturas e siglas

API	*Application Programming Interface*
CSS	*Cascading Style Sheets*
CSV	*Comma-separated values*
HTML	*HyperText Markup Language*
LLM	*Large Language Model*
PHP	*Hypertext Preprocessor*
PLN	Processamento de Linguagem Naural
RNA	Redes Neurais Artificiais
RNN	*Recurrent Neural Networks*
SQL	*Structured Query Language*
VSC	Visual Studio Code
XML	*Extensible Markup Language*

Sumário

1 INTRODUÇÃO 13
1.1 Problemática 14
1.2 Justificativa 15
1.3 Objetivos 16
1.3.1 Objetivo Geral 16
1.3.2 Objetivos Específicos 16

2 FUNDAMENTAÇÃO TEÓRICA 17
2.1 Redes Neurais 17
2.2 Transformers 18
2.3 Processamento de Linguagem Natural 19
2.4 *Large Language Models* 20
2.5 *Prompt Design* 22

3 MATERIAIS E MÉTODOS 24
3.1 Materiais Gerais 24
3.1.1 Figma 24
3.1.2 Visual Studio Code 25
3.1.3 XAMPP 26
3.1.4 MySQL Workbench 27
3.2 Modelos 27
3.2.1 OpenAI 27
3.2.2 AI21 Labs 28
3.2.3 Cohere 28
3.3 Métodos 29
3.3.1 Desenvolvimento do Protótipo 29
3.3.2 Desenvolvimento da Aplicação 29

4 ANÁLISE DOS RESULTADOS 41
4.1 Experimento 41
4.2 Resultados e Discussões 43

5 CONSIDERAÇÕES FINAIS 48
5.1 Trabalhos Futuros 48

REFERÊNCIAS 49

1 Introdução

Desde sua origem, a humanidade busca diferentes meios de expressar uma necessidade inerente à sua existência: a comunicação. Seja através de pinturas rupestres ou da própria fala, o ser humano usa as maneiras possíveis, contemporâneas a ele, de demonstrar seus sentimentos e manifestar seus desejos.

Com a evolução dos computadores e sua presença cada vez mais marcante na vida de tantas pessoas, a demanda por diálogo, característica do homem, também começou a ter importância na área das máquinas. Apesar das linguagens de programação, que proporcionam uma comunicação mais engessada entre homem-máquina, garantida por palavras reservadas que têm a função de exprimir as vontades do programador, a carência por conversas mais intuitivas e orgânicas ainda existia.

Na década de 1960, Eliza foi uma invenção capaz de demonstrar o início de uma solução para esse problema. De acordo com Hüllermeier, Kruse e Hoffmann (2010), o sistema analisava as entradas do usuário (escritas em linguagem natural) e, quando elas se encaixavam em uma estrutura pré-determinada, Eliza, que representava o papel de uma psicoterapeuta, respondia com alguma frase condizente, muitas das vezes. Apesar de nem sempre apresentar respostas completamente adequadas, o programa causava grande impacto e proporcionava aos usuários a sensação de se comunicar com outro ser humano através de uma máquina.

Tudo isso evidenciou ainda mais a necessidade de criação de ferramentas que pudessem, de fato, entender a linguagem do ser humano, sem que as entradas, obrigatoriamente, preenchessem moldes específicos que não fossem as normas da língua utilizada. Essa é, segundo os mesmos autores, a maior dificuldade abordada pela área de Processamento de Linguagem Natural (PLN), responsável pelo estudo das particularidades da compreensão de sentenças, fornecidas pelo usuário, por um computador.

Para que algo desse tipo fosse possível, as Redes Neurais Artificiais (RNA), criadas a fim de simular o processo de pensamento humano, são fundamentais. Fleck et al. (2016) afirmam que, através da interligação das menores unidades que compõem uma RNA, os neurônios artificiais, problemas mais complicados podem ser resolvidos. Além disso, elas são capazes de aprender através de dados de entrada e generalizar o conhecimento adquirido para que seja aplicado em outros casos, que não são os de treinamento utilizados inicialmente.

Mais especificamente, graças a uma inovação vinda das RNA, grandes avanços no ramo de entendimento de linguagem natural foram alcançados. O *transformer* é,

segundo Lin et al. (2022), um modelo de aprendizagem profunda que, também devido ao fato de utilizar um mecanismo chamado *self-attention*, se destacou em diversas áreas de aprendizado de máquina. De acordo com Khan et al. (2022), esse diferencial na arquitetura permite a identificação de relacionamentos entre as palavras da frase analisada, mesmo que elas tenham certa distância entre si.

Um dos modelos que aplica as vantagens do *transformer* e, por esse e outros fatores, apresenta resultados impressionantes é o *Large Language Model* (LLM). Ao reunir a compreensão apurada das conexões entre palavras e o treinamento que, nesse caso, é feito a partir de uma imensa quantidade de dados, essa estrutura permite a realização de uma diversidade de tarefas relacionadas a linguagem, de maneira muito satisfatória. Kasneci et al. (2023) afirma que um outro ponto importante é o processo de treinar, primeiramente, o modelo de linguagem com um grande volume de dados e, depois, treiná-lo para tarefas mais específicas. Essa decisão tem, comprovadamente, melhorado o desempenho em variadas atividades de linguagem.

Uma das aplicações mais recentes para o último modelo é o ChatGPT. Produzido pela OpenAI, o *chatbot* usa os princípios do LLM para disponibilizar diversas funções e executá-las com muita precisão, apesar de ser suscetível a erros e capaz de responder às demandas dos usuários com inconsistências.

Ao enxergar o tremendo potencial dos LLMs, diversas empresas optaram por investir em pesquisas e criar, por conta própria, suas versões de modelos desse tipo. Todas têm particularidades e vantagens que as diferenciam umas das outras, sejam elas a respeito das tarefas para as quais são treinadas ou a quantidade de parâmetros que utilizam.

Mesmo que ainda sejam consideradas novidades, é perceptível que as mudanças causadas pelos LLMs não são passageiras e tendem a ser cada vez mais exploradas.

1.1 Problemática

A complexidade presente em uma língua, geralmente, é muito alta. Além de ser uma construção viva, ou seja, sempre sujeita a alterações ou adições, também proporciona diferentes interpretações que dependem do contexto em que certas palavras são utilizadas e de como elas se comportam dentro de uma frase. Os coloquialismos também representam uma característica da língua que, apesar de não ser prevista pelas normas, é aceita, em muitas situações, socialmente. Esses aspectos devem ser levados em consideração quando o objetivo é compreender a intenção por trás de alguma sentença.

O sentimento imposto em uma frase é extremamente importante para determinar o propósito embutido nela. E esta é uma dedução que o contexto pode proporcionar quando a relação entre as palavras explicita o desejo do usuário. Se a ideia é analisar *feedbacks* recebidos de clientes, por exemplo, observar essas particularidades é imprescindível.

Um LLM pode, ocasionalmente, oferecer respostas tendenciosas ou incorretas. Existe um nível de impacto no usuário que, ao receber essas informações, caso não esteja ciente das falhas de um modelo desse tipo, pode tomá-las como verdade.

Além disso, escolher uma das diversas opções disponíveis de modelos desse tipo pode ser uma tarefa difícil. Ainda mais quando as características que os tornam interessantes para certas necessidades, ou não, são desconhecidas para quem deve tomar a decisão. Ou, quando seus desempenhos não são mensurados.

1.2 Justificativa

Tornar a compreensão de linguagem cada vez mais precisa é um propósito atrativo, já que o número de aplicações possíveis graças a uma funcionalidade como essa é muito alto. Seja na educação, por exemplo, em que alunos e professores podem utilizar um LLM para obter respostas que contribuam com o aprendizado e ensino dos mais diversos assuntos. Também para empresas que buscam melhorar a qualidade de seus produtos e, como medida intermediária, usam um modelo de linguagem para entender, em maior escala, as opiniões dos seus clientes.

A geração de respostas mais naturais e condizentes com o tema de uma conversação (em um *chatbot*, por exemplo) pode ser um fator determinante para o engajamento do usuário que, caso isso aconteça, se sente mais disposto a usar o *software* outras vezes. Não só isso, como também contribui com o nível de satisfação de quem utiliza o serviço.

A compreensão dos LLMs, que já estão prontos, pode ser um divisor de águas para uma empresa que, por exemplo, busca implementar algum deles e precisa fazer a escolha que traga os melhores resultados para as necessidades existentes naquele momento. Uma outra perspectiva importante é entender como os usuários recebem as respostas oferecidas por um modelo desse tipo, ou seja, descobrir o quanto elas são convincentes.

1.3 Objetivos

1.3.1 Objetivo Geral

O objetivo geral desse trabalho é construir uma aplicação que envolva os LLMs e que possa, além de demonstrar a importância e capacidade dessas estruturas, gerar comparações entre os diferentes modelos. Ao final do projeto, a ideia é possibilitar a avaliação da influência que as diferenças presentes em cada modelo tem, por exemplo, no tempo necessário para oferecer uma resposta a partir de uma entrada do usuário.

1.3.2 Objetivos Específicos

- Explorar as funcionalidades que os LLMs podem oferecer.
- Desenvolver um programa que permita ao usuário enviar *feedbacks* das respostas produzidas pelos diferentes modelos.
- Possibilitar a análise dos resultados de desempenho a partir das avaliações recebidas e alguns outros critérios, como tempo de resposta e tamanho dos textos gerados.

2 Fundamentação Teórica

O capítulo atual busca elaborar e explorar os conceitos importantes para a construção do trabalho.

2.1 Redes Neurais

As RNA são peças fundamentais para a criação de modelos de aprendizado de máquina que permitem a resolução dos mais variados problemas, pertinentes ao cotidiano de muitas pessoas. Segundo Cardon, Müller e Navaux (1994), as redes neurais presentes no cérebro humano são a principal inspiração para as criadas artificialmente que, assim como as primeiras, se caracterizam por componentes menores, os neurônios receptores de estímulos. Estes, por sua vez, são capazes de passar informações adiante, sob a condição de que os valores recebidos por eles, quando processados, atinjam um limiar específico. De acordo com os mesmos autores, os pesos, participantes muito importantes no processamento das entradas, indicam a força de uma sinapse e a importância daquela informação para o resultado final. São esses os valores que devem ser atualizados para que a rede aprenda.

Além disso, dados precisam ser fornecidos para que uma RNA funcione. É graças a eles que o conhecimento, obtido a partir da identificação de características semelhantes entre exemplos do mesmo tipo, pode ser generalizado e, assim, permitir que novos casos não presentes no grupo de treinamento sejam definidos corretamente.

Uma rede pode aprender de diversas maneiras. Uma delas é através do método supervisionado, em que os dados utilizados para a fase de treinamento são rotulados, ou seja, apresentam a classificação esperada, a que deveria ser o resultado final fornecido pela rede. Enquanto isso, na aprendizagem não supervisionada, uma outra técnica, os dados disponíveis para treino não possuem informações adicionais sobre o tipo tratado.

Ambos os métodos citados apresentam vantagens e desvantagens próprias. De acordo com Balestriero et al. (2023), a aprendizagem supervisionada, por exemplo, é limitada pela quantidade de dados rotulados disponíveis para uso. O processo de classificar muitos exemplos para utilizar no treinamento é extenso, já que demanda tempo e esforço.

Como um contraposto a esse problema, o aprendizado auto-supervisionado se destaca e alcança avanços importantes, sobretudo para tarefas de linguagem. Essa abordagem dispõe da capacidade de, através de entradas não rotuladas, produzir

representações descritivas dos dados. Modelos desse tipo podem ser treinados a fim de prever palavras a partir de outras, o que contribui para que eles possam compreender, em certo nível, o relacionamento entre as partes da sentença analisada.

São diversas as inovações e aplicações possíveis através do uso das RNA. Uma das áreas mais beneficiadas por elas é a de PLN. As Redes Neurais Recorrentes (RNN, do inglês) foram responsáveis por uma grande parte do desenvolvimento desse ramo, já que uma de suas maiores vantagens é considerar estados anteriores dos neurônios artificiais para processar informações mais a frente. Esse diferencial é especialmente importante para a compreensão de linguagem porque as palavras dentro de uma sentença se relacionam entre si, ou seja, um termo no início de uma frase pode ter ligação com o último, por exemplo. Entretanto, uma barreira considerável apresentada pelas RNN é a dificuldade de paralelizar seu funcionamento (visto que o processamento é feito sequencialmente), uma operação muito almejada a fim de deixar o progresso da rede mais rápido.

Os *transformers* surgem como uma alternativa para o uso das redes recorrentes e ganham espaço em um domínio que busca a comunicação entre computador e ser humano, através de linguagem natural.

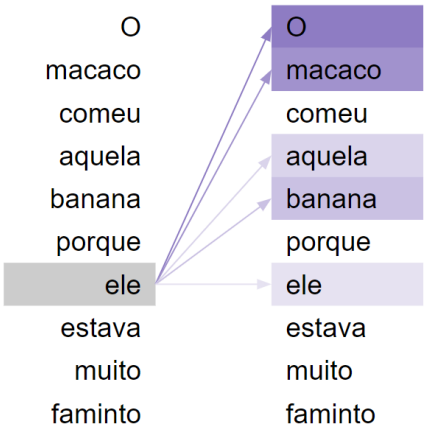
2.2 Transformers

A concepção dos *transformers* foi extremamente importante em um cenário marcado pela forte exploração das áreas de aprendizado de máquina e das redes neurais. Em pouco tempo, o uso de estruturas desse tipo contribuiu para que fosse possível atingir performances estado-da-arte em tarefas de linguagem, tamanho o avanço alcançado por pesquisas e estudos sobre o assunto. O surgimento do modelo teve como pontapé inicial a publicação do artigo *“Attention is all you need”*, em 2017, que explicava suas principais características, responsáveis por benefícios significativos. Segundo os autores do texto, Vaswani et al. (2017), os mecanismos de atenção se destacam por possibilitar que as dependências de uma sentença sejam identificadas, sem que a distância entre os termos atrapalhe. Os *transformers* usam esses recursos para que a paralelização do processamento dentro da rede seja possível.

Como descrito por Tunstall, Werra e Wolf (2022), através da atribuição de espécies de pesos para as partes de uma entrada em formato de texto, ou seja, ao distribuir atenção para cada uma delas, é possível reconhecer a importância dos fragmentos de uma frase e, também, discernir qual o grau de relação que eles possuem uns com os outros. Nas RNN, somente com os estados escondidos, o processamento sequencial pode não oferecer resultados tão apurados quanto estruturas que usam atenção.

Mais especificamente, *self-attention* é um conceito fundamental. De acordo com os autores do artigo que descreve o modelo, esse é o mecanismo de atenção responsável por relacionar as diferentes partes de uma sentença e, assim, obter uma representação da sequência de palavras como um todo. Na Figura 1, o funcionamento do mecanismo é demonstrado.

Figura 1 – Exemplo do funcionamento do mecanismo de *self-attention*



Fonte: Adaptada de Xie et al. (2021)

Com o advento de uma estrutura poderosa como os *transformers*, o progresso na área de PLN aumentou significantemente e as interações entre homem e máquina se tornaram ainda mais intuitivas.

2.3 Processamento de Linguagem Natural

Segundo Chowdhary (2020), uma linguagem pode ser definida como um conjunto de sentenças construídas através de um alfabeto finito. Apesar de serem baseadas em um número limitado de símbolos, as línguas apresentam alta complexidade, seja pelas diferenças culturais que variam de acordo com o local de origem ou especificidades do período histórico, já que certos termos podem deixar de ser utilizados, por exemplo. Esse é um dos motivos pelos quais surgem dificuldades no processo de interpretação da comunicação dos seres humanos.

Em um mundo digital, a quantidade de informações enviadas e mantidas pela internet é gigante e só tende a crescer. Por isso, de acordo com Banane e Erraissi (2022), os métodos utilizados para o PLN se tornam cada vez mais necessários, já que são capazes de processar, automaticamente, dados na forma de texto. Para os autores, a área de PLN tem como objetivo analisar e estudar a capacidade dos humanos e máquinas interagirem diretamente.

A interpretação de regras implícitas presentes na comunicação é uma das grandes dificuldades para o campo de compreensão das conversações naturais. São muitos os fatores que ditam as nuances de um diálogo, como a entonação, por exemplo.

Um outro obstáculo considerável é o fato de que o conhecimento de mundo, obtido através das experiências que o ser humano tem ao longo da vida, influencia diretamente no entendimento de uma expressão. Em uma frase, citar a palavra “cadeira” desperta a imagem do objeto graças a uma noção pré-existente no cotidiano de quem fala ou recebe a informação.

Além disso, a ambiguidade é um conceito que também interfere na maneira utilizada para interpretar uma sentença, já que uma mesma palavra, em situações diferentes, pode ter significados distintos que alteram completamente o sentido e a intenção do que é dito.

Grande parte dos problemas mencionados pode ser resolvido com a análise do contexto, dado que este fator revela muito sobre o que é enunciado dentro de uma frase. Uma palavra com múltiplos sentidos passa a ser compreendida com seu significado intendido a partir dos termos que se relacionam com ela.

Os esforços direcionados para gerar comunicação, mais intuitiva possível, entre homem e computador são a base do ramo de PLN, uma área de aprendizado de máquina. Ainda de acordo com Chowdhary (2020), as primeiras aplicações dessa área, contemporâneas a década de 1960, focavam na sintaxe da sentenças. Com o passar dos anos, a importância da semântica foi considerada.

As técnicas utilizadas a fim de executar tarefas relacionadas a linguagem foram variadas ao longo do tempo, como o uso de deduções e regras de inferência, condições e consequências, entre outras. A evolução dos métodos empregados levou a uma tendência mais atual: a utilização de *transformers* para formar estruturas que resolvam problemas de linguagem. A partir dessa ideia, os LLMs são idealizados.

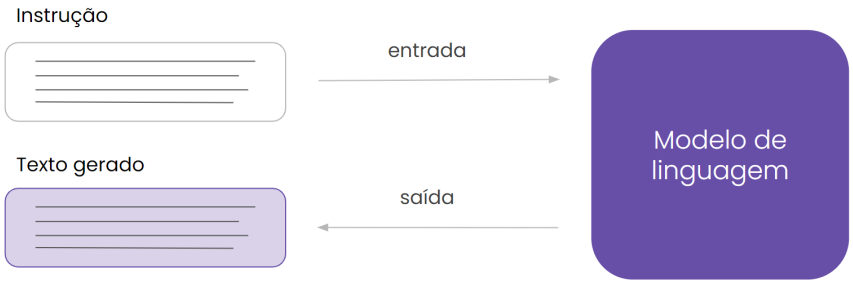
2.4 *Large Language Models*

Os modelos de linguagem são uma parte fundamental da área de PLN. Baseadas em distribuições probabilísticas, as estruturas que compõem os modelos são capazes de prever a palavra seguinte de uma sentença, ao considerar os termos anteriores a ela. Na Figura 2, uma das funções desses modelos é exemplificada.

Como uma versão mais poderosa dos modelos de linguagem, os LLMs surgem e apresentam desempenho surpreendente em diversas atividades que envolvem texto. Segundo Ozdemir (2023), é devido ao treinamento com um grande volume de dados que os modelos desse tipo podem compreender ainda melhor a complexidade das

línguas. Os grandes modelos usam, na maioria dos casos, os *transformers* em sua composição e, por isso, podem aproveitar a ótima interpretação de contexto que eles oferecem, graças aos mecanismos de atenção.

Figura 2 – Funcionamento de um modelo de linguagem



Fonte: Adaptada de Cohere (2022)

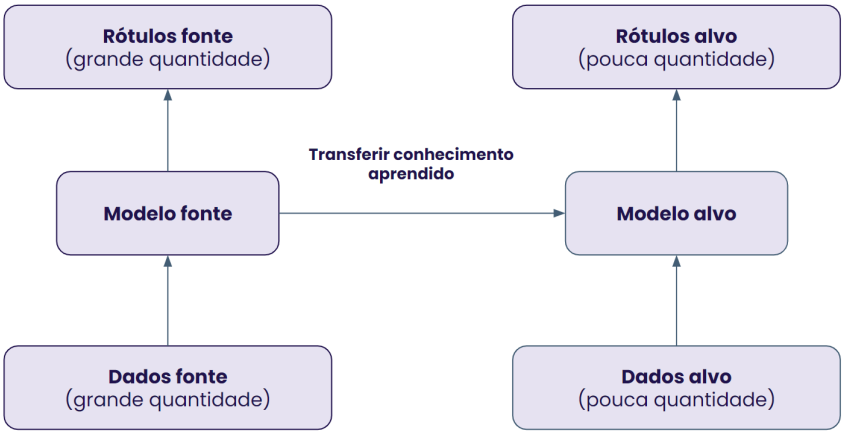
Os LLMs podem ser categorizados pela maneira como são treinados para completar uma sequência. Ainda de acordo com Ozdemir (2023), os modelos chamados *autoregressive* são definidos pela capacidade de completar uma sentença ao explorar os *tokens* anteriores. Os *tokens* podem ser considerados como pedaços da sequência analisada. Os modelos *autoencoding*, por sua vez, possuem a função de mascarar partes de uma frase e prever as palavras faltantes. Cada um dos tipos se destaca em objetivos diferentes: o primeiro, em tarefas de geração de texto e o segundo, em compreender linguagem. Mesmo que sejam distintos, todos os grandes modelos focam no reconhecimento de contexto.

Outras diferenças consideráveis entre os LLMs são provocadas pelas características utilizadas para o pré-treinamento. Todos eles usam grandes quantidades de dados para a fase de treino, entretanto, os corpos de textos escolhidos e as tarefas para as quais os modelos são treinados variam. Para se destacar no mercado, algumas empresas usam dados não públicos, enquanto outras optam por fontes como a Wikipedia.

Um ponto crucial para o sucesso dos grandes modelos é o conceito de *transfer learning*. Conforme dito por Ozdemir (2023), a ideia de transferir aprendizado, nesse contexto, consiste em treinar um LLM, que já foi pré-treinado, para realizar atividades mais específicas como classificação de texto, por exemplo. Dessa maneira, o conhecimento adquirido sobre as linguagens e os relacionamentos entre palavras diminui a necessidade de uma enorme quantidade de dados a fim de treinar os modelos para tarefas mais particulares, além de melhorar o desempenho. O tempo gasto para o processo também passa a ser menor. A Figura 3 indica a dinâmica possibilitada pelo

transfer learning.

Figura 3 – Representação do conceito de *transfer learning*



Fonte: Adaptada de Baheti (2021)

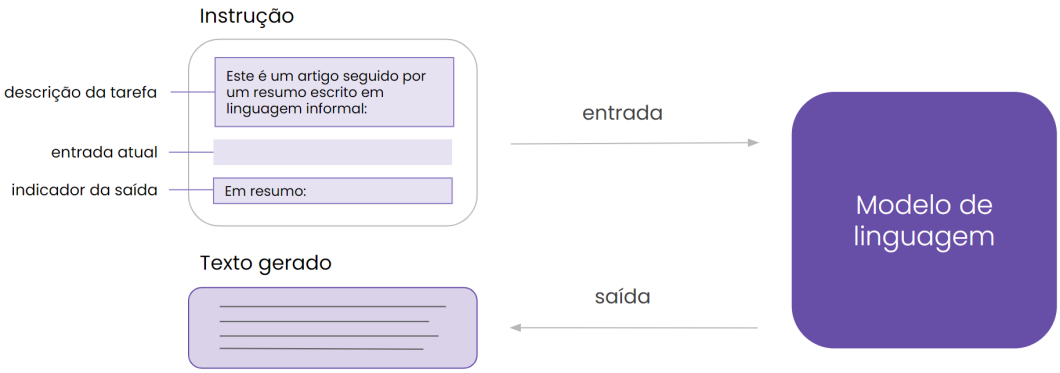
Entre as maneiras de obter os resultados esperados com o uso dos LLMs está a concepção de *prompts* (instruções) que explicitem, da melhor forma, a demanda do usuário.

2.5 *Prompt Design*

Ao utilizar um LLM, a grande motivação do usuário é obter as melhores respostas para suas requisições. Para que isso aconteça, as entradas dadas aos modelos devem ser as mais claras possíveis. De acordo com Jiang et al. (2022), o método conhecido como *prompt programming* consiste em utilizar instruções, em linguagem natural, para customizar um LLM a fim de que realize certas tarefas.

Enquanto isso, o conceito de *prompt design* se destaca por prezar pela construção das entradas de modo a atingir resultados mais precisos, através de estratégias como adição de detalhes, exemplos e contexto. Essa é uma ideia essencial para a eficácia dos modelos de linguagem baseados em instruções. A Figura 4 exemplifica o uso do conceito.

Figura 4 – Exemplo de uso de *prompt design*



Fonte: Adaptada de Cohere (2022)

3 Materiais e Métodos

Várias ferramentas foram utilizadas para a concepção e desenvolvimento do trabalho. Entre elas estão *softwares* que permitem a criação de protótipos, códigos, entre outros. Além disso, os modelos usados, disponibilizados por diferentes empresas, foram cruciais e também devem ser mencionados.

A implementação do projeto foi feita no formato de um site que reúne os principais tópicos discutidos, permite interações com o usuário e captura dados relevantes.

3.1 Materiais Gerais

Alguns meios foram escolhidos para possibilitar a construção do site como, por exemplo, os ambientes de desenvolvimento e as linguagens de programação. O hardware utilizado tem as especificações apresentadas no Quadro 1.

Software

- Sistema operacional: Windows 11
- Linguagens utilizadas: HTTML, CSS, PHP, javascript.

Hardware

Quadro 1 – Especificações do notebook usado

Marca	Lenovo
Modelo	Ideapad Gaming 3i i5-11300H
Memória RAM	8GB
Armazenamento	512GB SSD
Placa de Vídeo	NVIDIA GeForce GTX 1650 com 4GB

Fonte: Elaborado pelo autor.

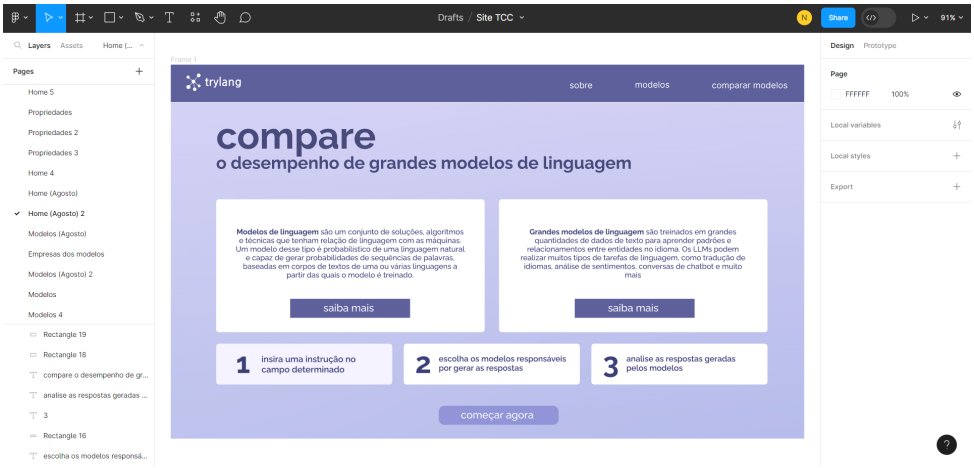
Os programas, importantes para propósitos distintos em diferentes etapas do projeto, são explorados a seguir.

3.1.1 Figma

O Figma é uma ferramenta conceituada, capaz de oferecer instrumentos para que os usuários criem designs que atendam as suas necessidades. Uma das grandes vantagens da aplicação é a possibilidade de produzir protótipos, que são muito úteis

no início de um projeto quando a direção criativa a ser tomada e os passos a serem seguidos ainda são incertos. Na Figura 5, a interface do *software* é apresentada.

Figura 5 – Interface do Figma



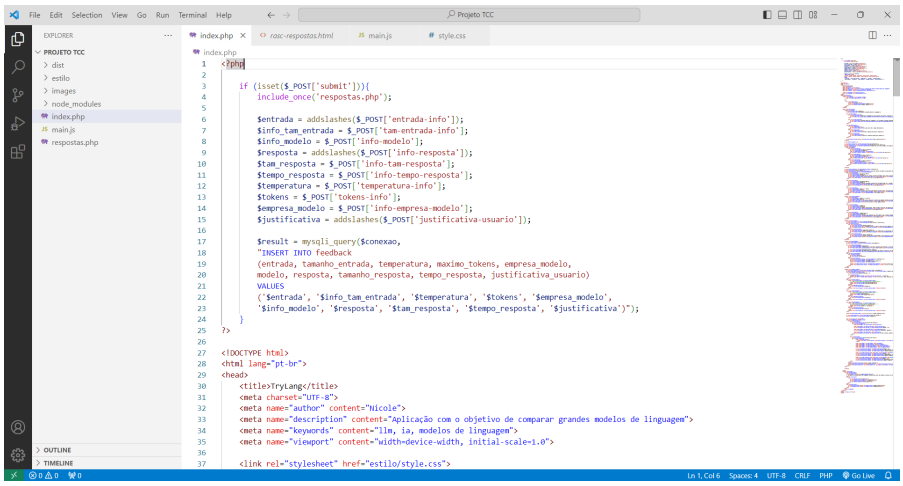
Fonte: Elaborada do autor

Segundo os próprios criadores do Figma (2011), a ferramenta se destaca por combinar facilidade, já que é um serviço acessível através de um navegador, com as funcionalidades de um programa instalado em um computador. Além disso, o planejamento de uma aplicação a ser desenvolvida passa a ser facilitado, devido a usabilidade do Figma, que possui controles intuitivos e tutoriais disponíveis na plataforma para sanar as dúvidas dos usuários.

3.1.2 Visual Studio Code

O Visual Studio Code (VSC) apresenta diferenciais responsáveis por colocar o editor de códigos em um espaço importante no mercado. Favorita de muitos programadores, a interface do VSC é simples e intuitiva, com as funcionalidades distribuídas de maneira a aproveitar bem o espaço disponível, de acordo com a documentação feita pela Microsoft (2015). A aplicação atrai usuários por possibilitar integração com o Git para controle de versões, por viabilizar o uso de diversas extensões que, no geral, poupam esforço e ajudam quem usa o programa, entre outras funções. Na Figura 6, a interface do VSC, utilizada para criação de códigos, é mostrada.

Figura 6 – Interface do Visual Studio Code

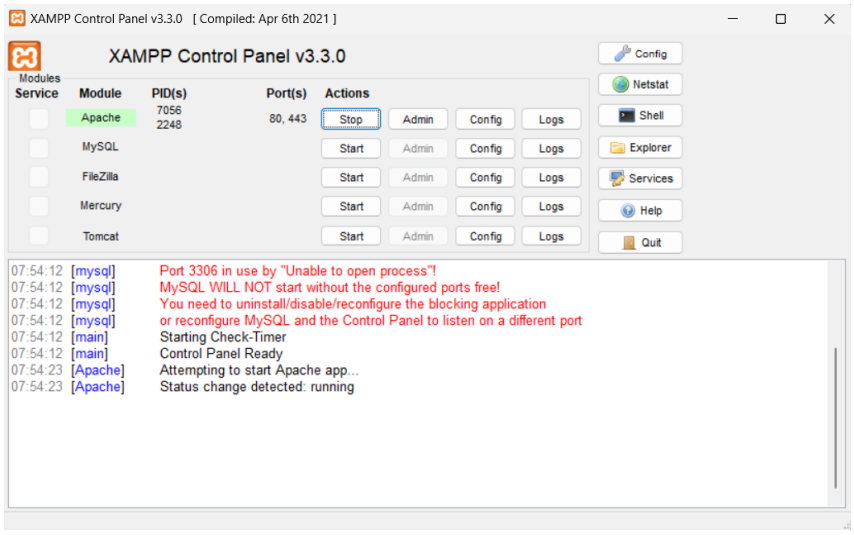


Fonte: Elaborada pelo usuário

3.1.3 XAMPP

O XAMPP, criado pela empresa Apache Friends, é um *software* gratuito e de código aberto. Segundo Agustin et al. (2018), a aplicação tem o papel de facilitar a instalação de pacotes e permitir a realização de testes dos sites criados pelo usuário, em um servidor local. Entre as opções disponíveis com a mesma função, o XAMPP é notável e seu nome, em formato de sigla, representa a compatibilidade com vários sistemas e os serviços oferecidos: Apache, MariaDB, PHP e Perl. A Figura 7 apresenta a interface do XAMPP e suas funcionalidades principais.

Figura 7 – Interface do XAMPP

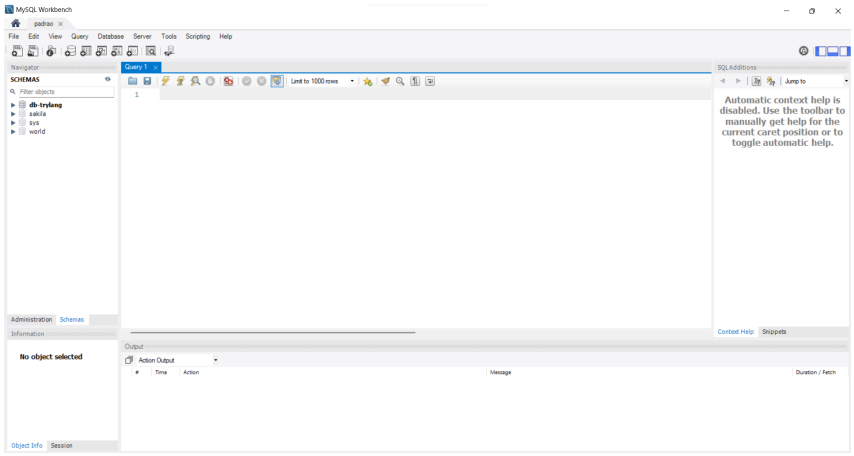


Fonte: Elaborada pelo autor

3.1.4 MySQL Workbench

Como uma ferramenta de banco de dados, o MySQL Workbench oferece uma interface visual capaz de intermediar as interações do usuário com o MySQL. De acordo com CodingDojo (2013), por ser um aplicativo gráfico, o Workbench torna muito mais fácil o processo de criação de bancos e de visualização de dados, além de viabilizar o uso de consultas SQL. Na Figura 8, é possível visualizar a tela principal do programa.

Figura 8 – Interface do MySQL WorkBench



Fonte: Elaborada pelo autor

3.2 Modelos

Para gerar respostas com base em pedidos dos usuários, alguns LLMs foram escolhidos. Entre os critérios usados para a seleção está a confiabilidade das empresas que disponibilizam os modelos, além da capacidade dos selecionados de seguirem instruções. Todos os designados são caracterizados por pertencerem a categoria *autoregressive*, explicada anteriormente (seção 2.4).

O intuito é permitir que o usuário compare modelos de diferentes empresas e decida, por si só, qual deles oferece o melhor resultado. Enquanto isso, a aplicação desenvolvida coleta a opinião de quem a usa e, a partir desses dados, pode ajudar a gerar um panorama maior das perfomances.

3.2.1 OpenAI

Os modelos criados pela OpenAI ganharam notoriedade nos últimos anos por apresentarem desempenhos impressionantes. Como afirma Dale (2021), um dos motivos para a relevância é a dificuldade de distinguir os textos produzidos por modelos dos escritos por humanos, apesar de serem suscetíveis a erros e alucinações.

O GPT-3, caracterizado por ser muito maior que seu predecessor (GPT-2) e treinado com 175 bilhões de parâmetros, foi lançado pela empresa em 2020. Segundo a documentação da OpenAI (2020), um dos modelos que compõem a linha GPT-3, o *text-davinci-003* é capaz de cumprir tarefas de linguagem com eficácia, além de ser competente ao seguir instruções. Enquanto isso, o *text-davinci-002* apresenta resultados semelhantes, porém, é treinado (mais especificamente) através de aprendizado supervisionado.

Os modelos mencionados foram os escolhidos para a implementação do site. O usuário pode decidir entre o *text-davinci-003* e o *text-davinci-002* para ser o representante da OpenAI na comparação de performance com os selecionados de outras empresas.

3.2.2 AI21 Labs

A AI21 Labs é uma empresa focada em inteligência artificial, criadora de modelos de linguagem com alta capacidade de compreender textos. Uma das divisões da companhia, a AI21 Studio, foi desenvolvida com o objetivo de ser um ambiente para que seus LLMs pudessem ser usados e implementados em aplicações dos usuários.

A linha de modelos *jurassic-2* se destaca por seu treinamento com uma grande quantidade de dados recentes, o que contribui para que a compreensão e a geração de textos sejam facilitadas. O *jurassic-2 ultra*, um dos escolhidos para compor o site do projeto, é considerado o mais poderoso da linha, tanto por ser o maior quanto por ser capaz de atender a tarefas mais complexas. O outro modelo selecionado é o *jurassic-2 mid*, ideal para tarefas com um menor grau de complexidade, enquanto ainda oferece eficiência, por produzir respostas em menos tempo.

3.2.3 Cohere

Com o crescimento da área de PLN, a Cohere, uma empresa orientada por pesquisas sobre aprendizado de máquina, também ganhou maior visibilidade. Seus modelos, baseados em *transformers*, foram produzidos com a intenção de serem aplicados em operações de negócios e, assim, aumentar o potencial dos empreendimentos.

Os modelos do tipo *command*, como previsto pelo nome, são equipados para responder a comandos dos usuários e, também, participar de conversações. O *command* é preparado para realizar tarefas de linguagem de maneira eficaz, além de oferecer respostas mais confiáveis. Já o *command-light*, a versão leve do primeiro modelo, é mais rápida e dispõe, praticamente, das mesmas capacidades.

3.3 Métodos

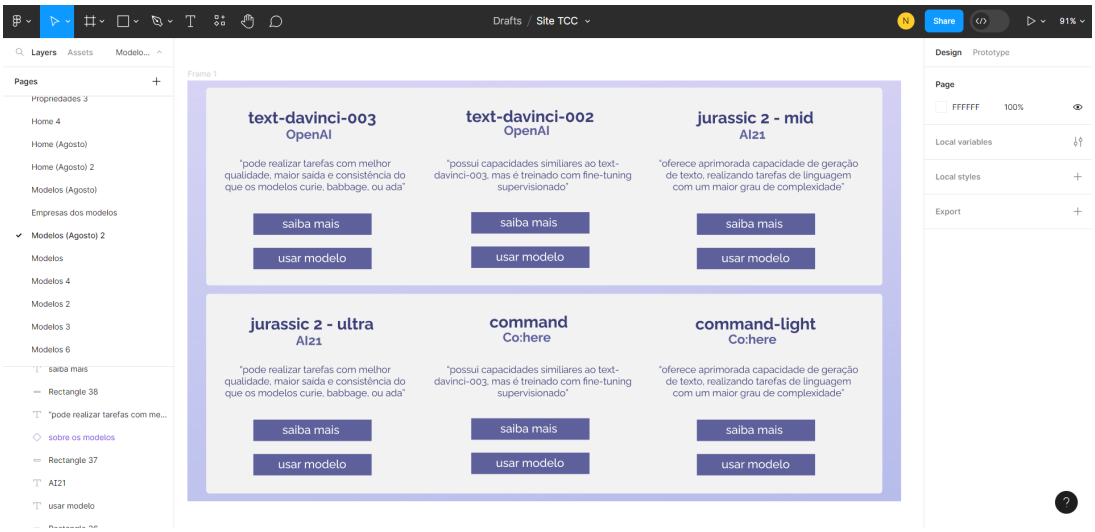
Os métodos empregados para a construção do projeto são apresentados a seguir. Eles variam desde o uso das linguagens escolhidas para o desenvolvimento até a interligação da aplicação com um banco de dados.

3.3.1 Desenvolvimento do Protótipo

O primeiro passo realizado durante a idealização do site foi a criação de um protótipo. Através dele, as estruturas e atributos visuais da aplicação puderam ser visualizados, o que facilitou o processo de desenvolvimento. As imagens criadas para compor o protótipo foram consultadas constantemente para que o projeto real ficasse o mais próximo possível do imaginado inicialmente.

Cada seção do site teve uma imagem destinada a representar como seu layout deveria ser, com as posições e cores já acertadas. Todos os módulos do protótipo foram produzidos através do Figma, como mostrado na Figura 9.

Figura 9 – Desenvolvimento do protótipo no Figma



Fonte: Elaborada pelo autor

3.3.2 Desenvolvimento da Aplicação

Depois do protótipo, a próxima fase cumprida foi o desenvolvimento da aplicação. Com os artifícios do HTML, a estrutura básica do site pôde ser feita e, com os do CSS, os estilos dos componentes foram alterados.

Assim que as primeiras etapas (estrutura e visual) foram finalizadas, um arquivo do tipo *javascript* foi criado para receber os comandos responsáveis por conferir atividades aos elementos da aplicação. Dessa maneira, os botões receberam funcionalidades

como, por exemplo, levar a páginas externas ou a partes específicas dentro do próprio site, ou selecionar o modelo desejado para as comparações. Além disso, a linguagem também foi usada para capturar os dados fornecidos pelo usuário, informações extremamente importantes para o funcionamento do projeto.

Para que os modelos pudessem receber requisições, as Application Programming Interface (API) foram necessárias. Através das plataformas das empresas, foi possível gerar as chaves essenciais para obtenção de acesso as APIs dos modelos. Cada uma delas foi atribuída a uma variável específica a fim de que fosse utilizada ao longo do código.

Um outro mecanismo fundamental implementado foi a conexão com um banco de dados, para que os dados processados pelo site fossem armazenados. Por meio da linguagem PHP, as variáveis criadas dentro do programa foram enviadas, no formato de um comando SQL, para preencher o banco. O MySQL Workbench, por sua vez, teve de ser configurado para receber as informações, ou seja, uma conexão, um banco e uma tabela foram criadas. A última foi montada com os campos correspondentes aos valores das variáveis que seriam obtidas. Nas Figura 10 e Figura 11, os códigos utilizado para esses propósitos são apresentados.

Figura 10 – Variáveis usadas no arquivo PHP

```php
$entrada = addslashes($_POST['entrada-info']);
$info_tam_entrada = $_POST['tam-entrada-info'];
$info_modelo = $_POST['info-modelo'];
$resposta = addslashes($_POST['info-resposta']);
$tam_resposta = $_POST['info-tam-resposta'];
$tempo_resposta = $_POST['info-tempo-resposta'];
$temperatura = $_POST['temperatura-info'];
$tokens = $_POST['tokens-info'];
$empresa_modelo = $_POST['info-empresa-modelo'];
$justificativa = addslashes($_POST['justificativa-usuario']);
```

Fonte: Elaborada pelo autor

Figura 11 – Envio das informações para o banco de dados

```php
$result = mysqli_query($conexao,
"INSERT INTO feedback
(entrada, tamanho_entrada, temperatura, maximo_tokens, empresa_modelo,
modelo, resposta, tamanho_resposta, tempo_resposta, justificativa_usuario)
VALUES
('$entrada', '$info_tam_entrada', '$temperatura', '$tokens', '$empresa_modelo',
'$info_modelo', '$resposta', '$tam_resposta', '$tempo_resposta', '$justificativa')");
```

Fonte: Elaborada pelo autor

A seguir, as seções produzidas para o site são apresentadas. A primeira, uma tela inicial representada pela Figura 12, foi idealizada com a intenção de introduzir o usuário aos conceitos necessários para a utilização do site. Caso seja do seu interesse saber mais sobre algum deles, botões foram inseridos para o direcionamento a páginas externas que explicam os conteúdos mais detalhadamente. Além disso, instruções básicas foram posicionadas na primeira seção para indicar o comportamento esperado de quem usa a aplicação. Um botão denominado “começar agora” foi adicionado para levar o usuário diretamente até a seção de comparações de modelos.

Figura 12 – Página inicial do site



Fonte: Elaborada pelo autor

O espaço da segunda seção, ilustrado pela Figura 13, foi dedicado as empresas que fornecem os modelos incorporados na aplicação. A ideia presente na tela inicial também foi aplicada aqui: se o usuário quiser saber mais sobre cada uma delas, pode ser redirecionado por meio dos links destacados no espaço de cada empresa.

Figura 13 – Seção das empresas dos modelos



Fonte: Elaborada pelo autor

A área dedicada aos modelos, uma das principais partes do site, contém breves descrições sobre eles, individualmente, e é indicada pela Figura 14. Ainda nessa seção, o usuário pode decidir quais dos modelos serão usados para as comparações. Para isso, foram dispostos botões que podem ser acionados a qualquer momento, tanto para a escolha dos LLMs, quanto para entender mais sobre algum deles.

Figura 14 – Seção dos modelos implementados



Fonte: Elaborada pelo autor

Com o objetivo de gerar verdadeiras comparações entre os modelos oferecidos por diferentes empresas, é possível selecionar apenas um de cada companhia e, dessa maneira, identificar os diferenciais de desempenho de cada um deles.

Em uma tentativa de guiar o usuário e aplicar os princípios de *prompt design*, uma seção específica foi criada e apresentada através da Figura 15. Por meio de dicas para a escrita de instruções, colocadas dentro de quadros para a apresentação no site, é possível melhorar a maneira com que os comandos são produzidos e, consequentemente, aprimorar as respostas geradas pelos modelos.

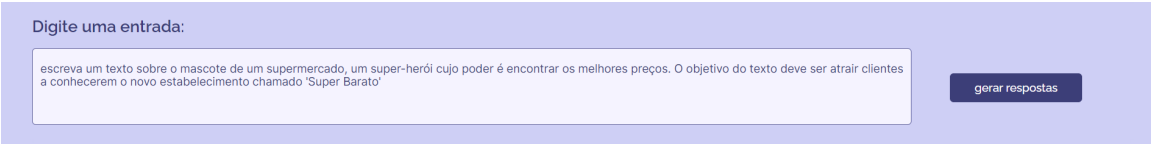
Figura 15 – Seção de dicas para as instruções



Fonte: Elaborada pelo autor

Quando o usuário clica em um dos botões do tipo “ver exemplo”, uma instrução já formulada previamente é posicionada no campo para inserção de entradas na seção de comparação de modelos, como mostrado pela Figura 16. A ideia por trás desse processo é permitir que o usuário veja, na prática, o efeito das dicas oferecidas e, também, disponibilizar sugestões de *prompts* caso sejam necessárias.

Figura 16 – Ação de um dos botões de dicas para instruções



Fonte: Elaborada pelo autor

Uma das características importantes para o uso dos LLMs é a temperatura atribuída aos modelos. Através dela, é possível controlar o nível de criatividade, já que é o valor dessa propriedade que define se o modelo deve sempre usar as palavras com maior probabilidade de acontecerem em uma sequência, ou se deve selecionar outras com chances menores. Quando a última opção é verdadeira, os resultados produzidos, a partir de uma mesma instrução, tendem a ser diferentes. Geralmente, os

valores atribuídos para a temperatura do modelo variam entre 0 e 1, de maneira que o menor número seja responsável por respostas sem variabilidade e, o maior, por textos diferentes entre si. Além disso, um outro conceito relevante é a quantidade máxima de *tokens* escolhida, por ser um valor que pode determinar o tamanho das respostas geradas, conforme a necessidade de quem faz a requisição ao modelo.

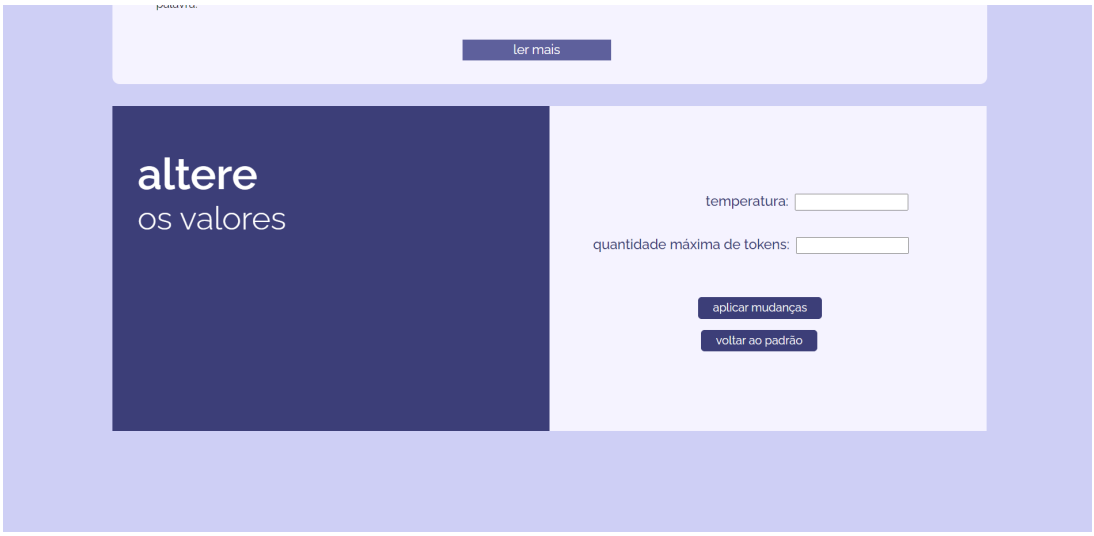
Para que o usuário pudesse alterar os valores das propriedades mencionadas, uma seção foi adicionada no projeto do site. Nela, estão contidas breves descrições dos conceitos, com links para direcionamento externo, e campos que capturam os números inseridos. A fim de efetuar as mudanças, o usuário pode aplicá-las com o uso de um botão e, quando elas já foram consolidadas, pode voltar aos valores definidos por padrão. A Figura 17 e Figura 18 demonstram o *layout* atingido.

Figura 17 – Seção de propriedades dos modelos



Fonte: Elaborada pelo autor

Figura 18 – Alterar propriedades



Fonte: Elaborada pelo autor

A parte principal do site e o centro de todas as operações é a responsável pelas comparações dos modelos. No campo determinado, o usuário pode escrever uma instrução de sua preferência e, caso não tenha escolhido os modelos na seção que detalha cada um deles, fazer isso nesse momento. Depois, ao apertar o botão “gerar respostas”, as requisições são feitas aos LLMs e as respostas geradas por eles são apresentadas nos espaços correspondentes. A Figura 19 apresenta a seção descrita.

Um botão, com a função de limpar as respostas já geradas e apresentadas nas caixas de texto, foi acrescentado, para casos em que é necessário produzir novos resultados para a mesma instrução.

Figura 19 – Seção de comparação dos modelos



Fonte: Elaborada pelo autor

Na visão do programador, o valor de entrada inserido dentro da caixa de texto é capturado e, assim como o valor da temperatura, quantidade máxima de *tokens* e nome do modelo escolhido, é passado como um parâmetro para a API da empresa, como mostrado pela Figura 20.

Figura 20 – Código da requisição para a API da empresa

```
// requisição openai
fetch("https://api.openai.com/v1/completions",{
    method: 'POST',
    headers: {
        Accept: "application/json",
        "Content-Type": "application/json",
        Authorization: `Bearer ${apiKey}`
    },
    body: JSON.stringify({
        model: modelo_openai,
        prompt: instrucao,
        max_tokens: tokens_openai,
        temperature: temperatura_openai
    })
})
```

Fonte: Elaborada pelo autor

Para calcular o tempo de resposta, uma funcionalidade nativa do *javascript* foi utilizada. Antes da requisição, a função *date* é atribuída a variável *start* e, assim, retorna o horário atual. Quando a API da empresa devolve a resposta gerada, uma nova variável, chamada *end*, recebe o mesmo comando usado no ínicio: *Date.now()*.

Figura 21 – Código do cálculo do tempo de resposta de um modelo

```
// início do cálculo do tempo de resposta
var start =  Date.now();

// requisição openai
fetch("https://api.openai.com/v1/completions",{
    method: 'POST',
    headers: {
        Accept: "application/json",
        "Content-Type": "application/json",
        Authorization: `Bearer ${apiKey}`
    },
    body: JSON.stringify({
        model: modelo_openai,
        prompt: instrucao,
        max_tokens: tokens_openai,
        temperature: temperatura_openai
    })
})

.then((response) => response.json())
.then((response) => {
    resposta_openai = response.choices[0]['text'];

    // fim do cálculo do tempo de resposta
    var end = Date.now();
    TempoOpenAI = (end - start)/1000;
    tamanho_resposta_openai = resposta_openai.length;

    texto_tempo_openai.innerHTML = `Tempo de resposta: ${TempoOpenAI} s`;
    texto_tamanho_openai.innerHTML = `Quantidade de caracteres: ${tamanho_resposta_openai}`;
    texto_modelo_openai.innerHTML = `Modelo: ${modelo_openai}`;
})
```

Fonte: Elaborada pelo autor

No cálculo final, os dois valores armazenados nas variáveis são subtraídos e o resultado, dividido por 1000, a fim de chegar em um valor em segundos. Dessa maneira, o tempo necessário para a produção do texto é obtido. O código implementado é exibido na Figura 21.

As respostas, depois de geradas, são apresentadas no espaço dedicado para a empresa do modelo, como indicado pela Figura 22. Isso é possível graças as funções do tipo *mostrarResposta*. Cada uma delas é encarregada de mostrar a resposta de uma das companhias. O código transforma a caixa de texto, um objeto HTML, em uma variável *javascript*, para que seu valor possa ser alterado. Assim, a resposta é passada por parâmetro quando a função é chamada, e atribuída a variável correspondente a caixa de texto. Esse processo é ilustrado pela Figura 23.

Figura 22 – Espaço dedicado para a resposta de uma empresa



Fonte: Elaborada pelo autor

Figura 23 – Código da função para mostrar a resposta

```
function mostrarRespostaOpenAI(resposta){
    var caixa_resposta = document.getElementById('caixa-resposta-openai');
    caixa_resposta.value = "";
    caixa_resposta.value = resposta;
}
```

Fonte: Elaborada pelo autor

A opinião dos usuários sobre as respostas geradas é uma parte essencial para o funcionamento da aplicação. É a partir desses dados que o panorama sobre os desempenhos dos modelos pode ser construído. A fim de que o usuário tenha todas as informações necessárias para decidir qual o melhor resultado (além da resposta em si), uma seção específica do site foi dedicada a elas.

As características das respostas, apresentadas no site, incluem o modelo escolhido, a quantidade de caracteres e o tempo de resposta. Além disso, algumas informações dos valores inseridos pelo usuário também são fornecidas. Entre elas estão a temperatura, a quantidade máxima de *tokens* e o tamanho da instrução. O último dado se torna útil para tarefas como a de sumarização, em que a saída deve ter um tamanho menor do que o texto inicial. Os valores são mostrados conforme indicado pela Figura 24.

Figura 24 – Seção de *feedback* do usuário



Fonte: Elaborada pelo autor

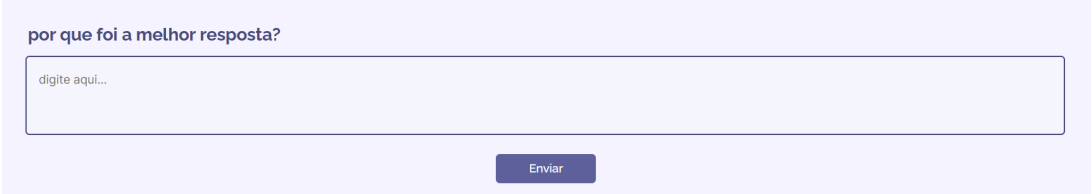
O usuário pode, ao analisar as informações mencionadas, escolher qual das respostas foi mais satisfatória, ou seja, qual delas atendeu aos requisitos estabelecidos por ele, como, por exemplo, rapidez e consistência. Um dos objetivos do projeto é oferecer a quem usa a aplicação uma maior quantidade de respostas para uma mesma

instrução. Dessa forma, o usuário dispõe de mais chances de encontrar o resultado adequado.

Além da escolha da melhor resposta, é possível inserir uma breve descrição que explique a decisão tomada, no espaço apresentado pela Figura 25. Todos os dados citados, desde as características de entrada e saída, até os inseridos pelo usuário, são enviados para um banco de dados.

Como uma medida para prevenir erros nos envios ao banco de dados, algumas restrições foram incluídas durante o desenvolvimento. Por exemplo, enquanto nenhuma resposta for gerada, o botão encarregado de enviar as informações de *feedback* do usuário fica bloqueado. O mesmo acontece quando o de limpar respostas é acionado. Essas são precauções impostas a fim de que o programa não tente enviar dados que ainda não existem.

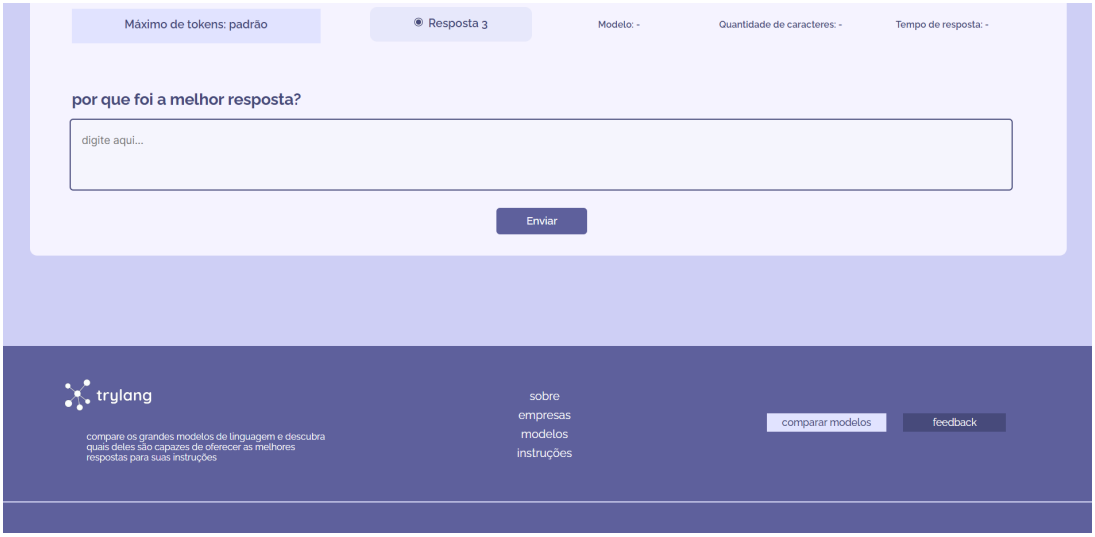
Figura 25 – Espaço para justificativa da escolha da resposta



Fonte: Elaborada pelo autor

Na última seção, alguns links e botões foram posicionados para facilitar o acesso e a navegação ao redor do site. Uma breve descrição sobre o projeto também foi adicionada. As adições são exibidas na Figura 26.

Figura 26 – Rodapé do site



Fonte: Elaborada pelo autor

Com isso, a construção da aplicação foi finalizada. Destacado como o propósito das decisões tomadas durante todo o processo está a criação de um ambiente intuitivo, que pudesse receber, de maneira amigável, pessoas que não conhecem tanto sobre o assunto e, ao mesmo tempo, atender as necessidades de quem já está mais familiarizado com as ideias usadas no site.

4 Análise dos Resultados

Como um meio de provar a relevância e capacidade do projeto desenvolvido, é apresentada, a seguir, a análise dos resultados obtidos.

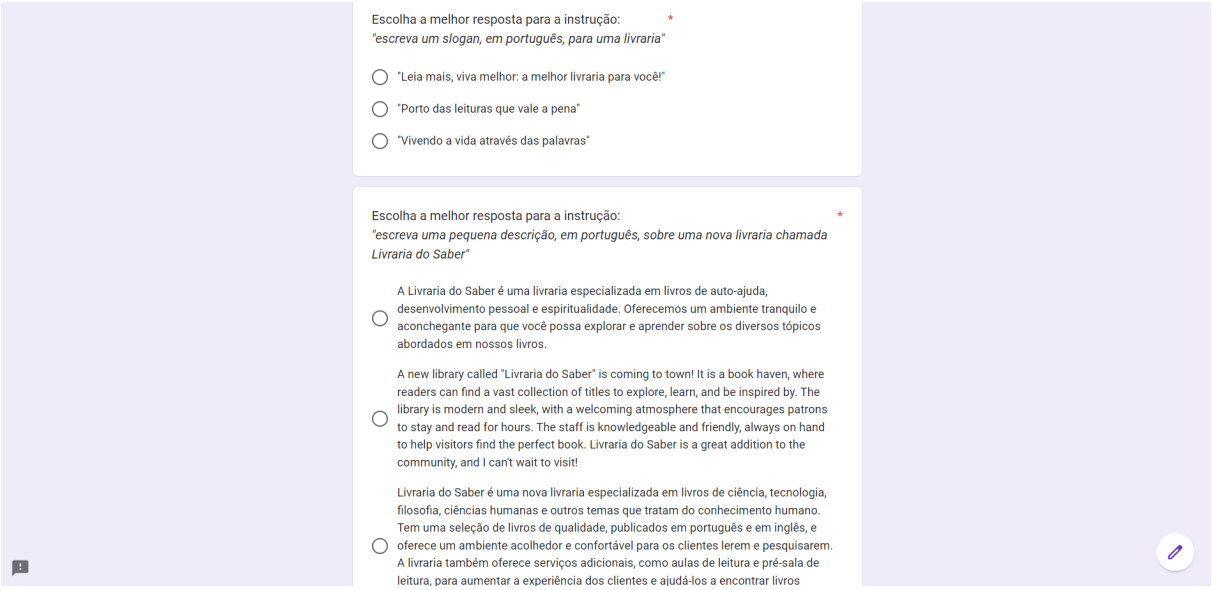
4.1 Experimento

A aplicação desenvolvida está, até o momento, hospedada localmente e, por isso, não pode ser divulgada amplamente para que a coleta de dados seja uma realidade. Entretanto, para ilustrar o funcionamento do projeto, uma pesquisa, feita através da plataforma Google Forms, foi divulgada para 60 pessoas, a fim de que elas devolvessem suas opiniões sobre as respostas geradas pelos modelos.

É importante ressaltar que, por meio do MySQL Workbench, é possível exportar os dados inseridos no banco de dados para um arquivo no formato CSV ou XML, por exemplo. Esse registro pode, por sua vez, ser o pontapé inicial para uma análise, já que existem programas de planilhas que são capazes de ler e interpretar esses tipos de arquivo e, consequentemente, as tuplas em questão. O Google Sheets é um deles e, portanto, também é qualificado para gerar gráficos que dizem a respeito dos dados coletados.

A partir dessas considerações, a pesquisa realizada para o projeto pode ser utilizada. Nela, duas perguntas do mesmo tipo foram inseridas, baseadas no funcionamento da aplicação, com alternativas determinadas pelas respostas geradas pelos modelos disponíveis, conforme mostrado pela Figura 27. As questões pediam que o usuário escolhesse o melhor resultado para uma instrução específica. A quantidade de perguntas foi definida pelo nível dos modelos a serem comparados, ou seja, a primeira foi destinada somente para os LLMs mais potentes de cada empresa, e a segunda, para os mais leves e menos poderosos.

Figura 27 – Perguntas do formulário utilizado



Fonte: Elaborada pelo autor

No total, foram obtidas 24 respostas. Depois de armazenadas, elas foram exportadas para uma planilha, assim como aconteceria com os dados armazenados no MySQL Workbench. Além disso, as outras informações que caracterizam cada uma das respostas (como tempo de resposta, por exemplo) foram adicionadas para que os gráficos pudessem ser elaborados. A Figura 28 e a Figura 29 apresentam o resultado desse processo.

Figura 28 – Parte da planilha correspondente as respostas para a primeira pergunta

entrada	tamanho_entrada	temperatura	maximo_tokens	empresa_modelo	modelo	resposta	tamanho_
escreva um slogan, em português, para uma livraria	50	0.7	200	AI21 Studio	j2-ultra	"Vivendo a vida através das palavras"	3
escreva um slogan, em português, para uma livraria	50	0.7	200	AI21 Studio	j2-ultra	"Vivendo a vida através das palavras"	3
escreva um slogan, em português, para uma livraria	50	0.7	200	AI21 Studio	j2-ultra	"Vivendo a vida através das palavras"	3
escreva um slogan, em português, para uma livraria	50	0.9	300	Co:here	command	"Porto das leituras que vale a pena"	3
escreva um slogan, em português, para uma livraria	50	0.7	200	AI21 Studio	j2-ultra	"Vivendo a vida através das palavras"	3
escreva um slogan, em português, para uma livraria	50	0.7	200	AI21 Studio	j2-ultra	"Vivendo a vida através das palavras"	3
escreva um slogan, em português, para uma livraria	50	0.5	2048	OpenAI	text-davinci-003	"Leia mais, viva melhor: a melhor livraria para você!"	5
escreva um slogan, em português, para uma livraria	50	0.7	200	AI21 Studio	j2-ultra	"Vivendo a vida através das palavras"	3
escreva um slogan, em português, para uma livraria	50	0.5	2048	OpenAI	text-davinci-003	"Leia mais, viva melhor: a melhor livraria para você!"	5
escreva um slogan, em português, para uma livraria	50	0.5	2048	OpenAI	text-davinci-003	"Leia mais, viva melhor: a melhor livraria para você!"	5
escreva um slogan, em português, para uma livraria	50	0.7	2048	AI21 Studio	j2-ultra	"Vivendo a vida através das palavras"	3
escreva um slogan, em português, para uma livraria	50	0.7	200	AI21 Studio	j2-ultra	"Vivendo a vida através das palavras"	3
escreva um slogan, em português, para uma livraria	50	0.7	200	AI21 Studio	j2-ultra	"Vivendo a vida através das palavras"	3
escreva um slogan, em português, para uma livraria	50	0.5	2048	OpenAI	text-davinci-003	"Leia mais, viva melhor: a melhor livraria para você!"	5
escreva um slogan, em português, para uma livraria	50	0.7	200	AI21 Studio	j2-ultra	"Vivendo a vida através das palavras"	3
escreva um slogan, em português, para uma livraria	50	0.7	200	AI21 Studio	j2-ultra	"Vivendo a vida através das palavras"	3
escreva um slogan, em português, para uma livraria	50	0.5	2048	OpenAI	text-davinci-003	"Leia mais, viva melhor: a melhor livraria para você!"	5
escreva um slogan, em português, para uma livraria	50	0.7	200	AI21 Studio	j2-ultra	"Vivendo a vida através das palavras"	3

Fonte: Elaborada pelo autor

Figura 29 – Parte da planilha correspondente as respostas para a segunda pergunta

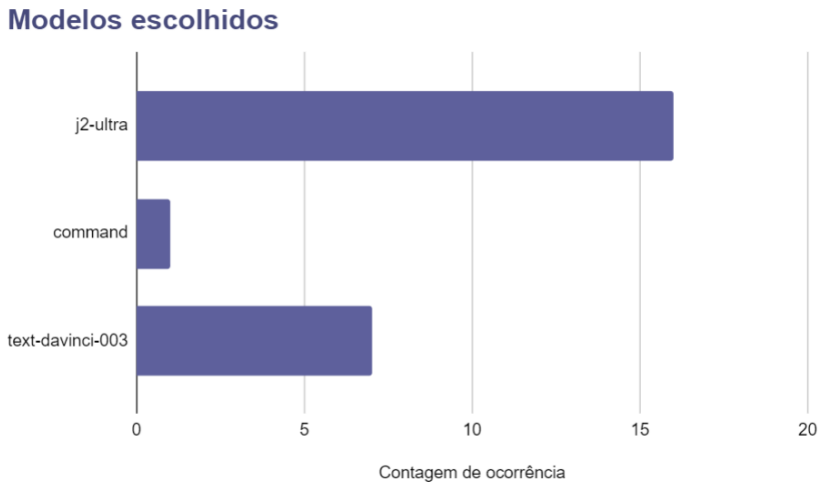
entrada	tamanho_entrada	temperatura	maximo_tokens	empresa_modelo	modelo	
"escreva uma pequena descrição, em português, sobre uma nova livraria chamada Livraria do Saber"	96	0.5	2048	OpenAI	text-davinci-002	A Livraria do Saber é uma livraria espec
"escreva uma pequena descrição, em português, sobre uma nova livraria chamada Livraria do Saber"	96	0.7	200	AI21 Studios	j2-mid	Livraria do Saber é uma nova livraria esp
"escreva uma pequena descrição, em português, sobre uma nova livraria chamada Livraria do Saber"	96	0.8	200	AI21 Studios	j2-mid	Livraria do Saber é uma nova livraria esp
"escreva uma pequena descrição, em português, sobre uma nova livraria chamada Livraria do Saber"	96	0.9	200	AI21 Studios	j2-mid	Livraria do Saber é uma nova livraria esp
"escreva uma pequena descrição, em português, sobre uma nova livraria chamada Livraria do Saber"	96	0.10	200	AI21 Studios	j2-mid	Livraria do Saber é uma nova livraria esp
"escreva uma pequena descrição, em português, sobre uma nova livraria chamada Livraria do Saber"	96	0.11	200	AI21 Studios	j2-mid	Livraria do Saber é uma nova livraria esp
"escreva uma pequena descrição, em português, sobre uma nova livraria chamada Livraria do Saber"	96	0.5	2048	OpenAI	text-davinci-002	A Livraria do Saber é uma livraria espec
"escreva uma pequena descrição, em português, sobre uma nova livraria chamada Livraria do Saber"	96	0.7	200	AI21 Studios	j2-mid	Livraria do Saber é uma nova livraria esp
"escreva uma pequena descrição, em português, sobre uma nova livraria chamada Livraria do Saber"	96	0.7	200	AI21 Studios	j2-mid	Livraria do Saber é uma nova livraria esp
"escreva uma pequena descrição, em português, sobre uma nova livraria chamada Livraria do Saber"	96	0.7	200	AI21 Studios	j2-mid	Livraria do Saber é uma nova livraria esp
"escreva uma pequena descrição, em português, sobre uma nova livraria chamada Livraria do Saber"	96	0.5	2048	OpenAI	text-davinci-002	A Livraria do Saber é uma livraria espec
"escreva uma pequena descrição, em português, sobre uma nova livraria chamada Livraria do Saber"	96	0.7	200	AI21 Studios	j2-mid	Livraria do Saber é uma nova livraria esp
"escreva uma pequena descrição, em português, sobre uma nova livraria chamada Livraria do Saber"	96	0.7	200	AI21 Studios	j2-mid	Livraria do Saber é uma nova livraria esp
"escreva uma pequena descrição, em português, sobre uma nova livraria chamada Livraria do Saber"	96	0.7	200	AI21 Studios	j2-mid	Livraria do Saber é uma nova livraria esp
"escreva uma pequena descrição, em português, sobre uma nova livraria chamada Livraria do Saber"	96	0.7	200	AI21 Studios	j2-mid	Livraria do Saber é uma nova livraria esp
"escreva uma pequena descrição, em português, sobre uma nova livraria chamada Livraria do Saber"	96	0.5	2048	OpenAI	text-davinci-002	A Livraria do Saber é uma livraria espec
"escreva uma pequena descrição, em português, sobre uma nova livraria chamada Livraria do Saber"	96	0.7	200	AI21 Studios	j2-mid	Livraria do Saber é uma nova livraria esp

Fonte: Elaborada pelo autor

4.2 Resultados e Discussões

Com base nas respostas obtidas, algumas análises dos dados recebidos se tornam possíveis e, dessa maneira, revelam a capacidade de modelos e como eles se destacam em certos aspectos. Para a instrução “escreva um slogan, em português, para uma livraria”, a representação dos modelos escolhidos, ou seja, quais deles forneceram os melhores textos, é apresentada na Figura 30.

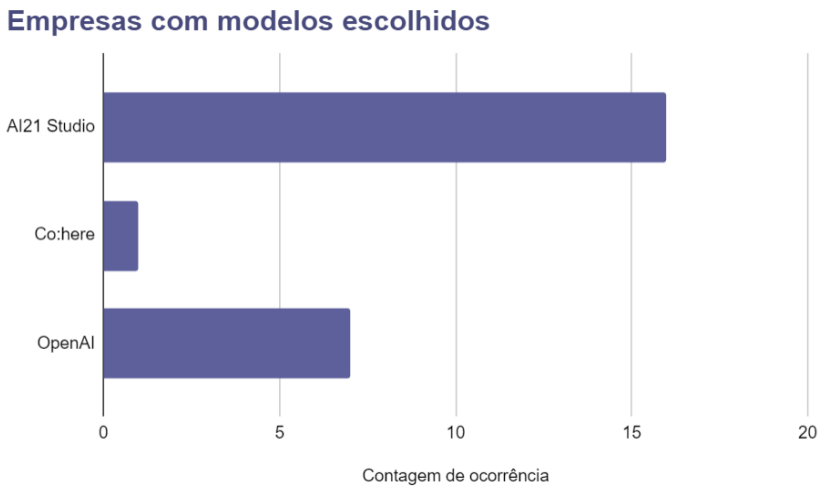
Figura 30 – Gráfico dos modelos escolhidos



Fonte: Elaborada pelo autor

Consequentemente, o gráfico que indica as empresas e suas frequências nas respostas do público é o apresentado pela Figura 31.

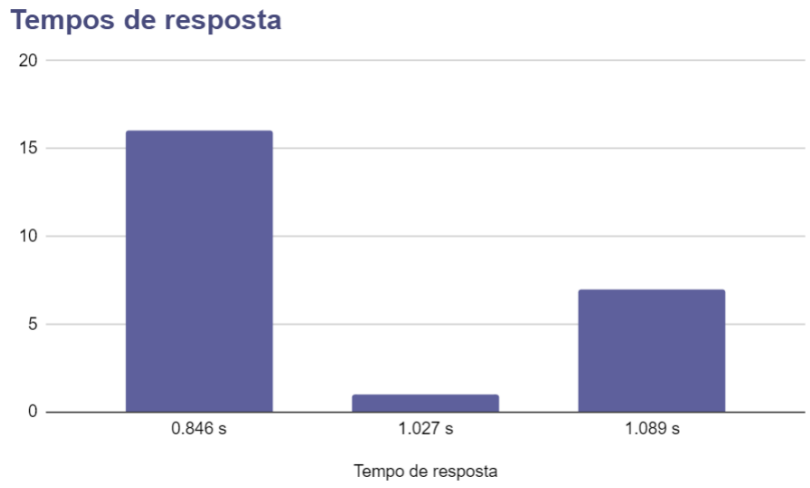
Figura 31 – Gráfico das empresas escolhidas



Fonte: Elaborada pelo autor

Além disso, o tempo necessário para cada modelo gerar a sua respectiva resposta também pode ser levado em consideração, por meio da Figura 32.

Figura 32 – Gráfico dos tempos de resposta (baseados na ocorrência)

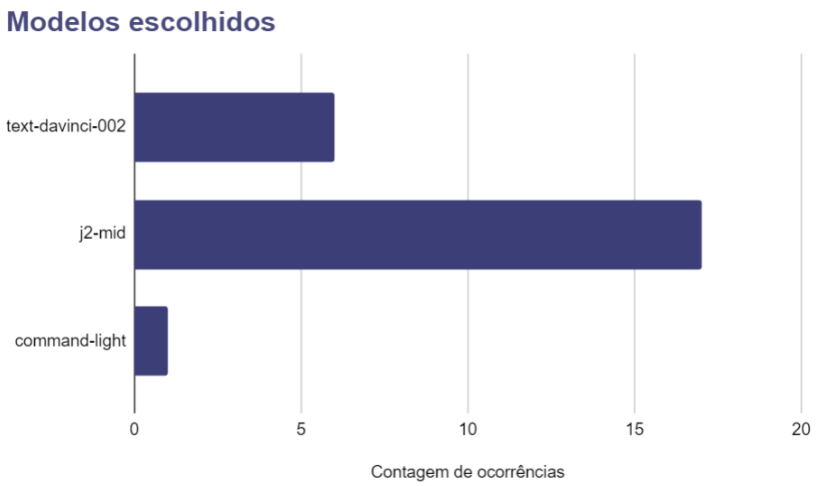


Fonte: Elaborada pelo autor

Com base nos dados mencionados, é possível perceber que o modelo criado pela AI21 Studio, o *j2-ultra*, foi o que obteve maior sucesso na geração de um texto que cumprisse as demandas da instrução. Não só sua resposta se destacou como a mais satisfatória, mas, também, o tempo demandado para produção do resultado foi o menor entre os 3 modelos. Nesse caso, o *j2-ultra* se destacou para geração de um texto curto (um slogan), já que demonstrou rapidez e eficácia.

Em relação a segunda pergunta incorporada na pesquisa, sua instrução pedia aos LLMs: "escreva uma pequena descrição, em português, sobre uma nova livraria chamada Livraria do Saber". A representação dos modelos escolhidos é exibida na Figura 33.

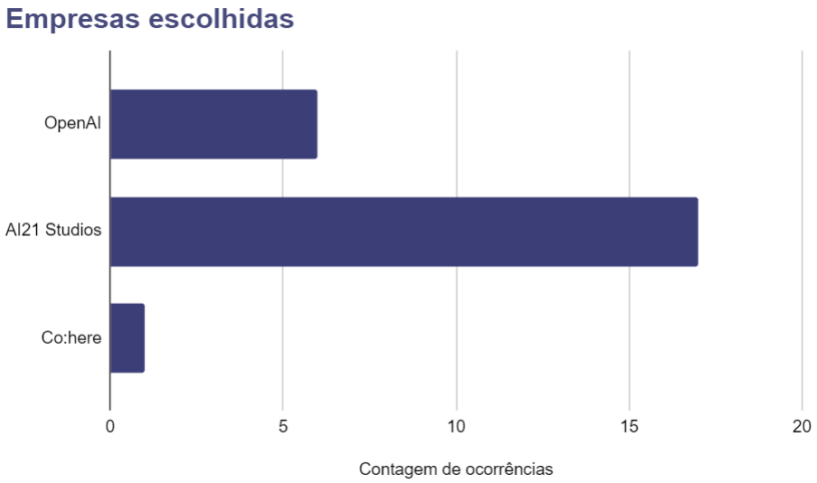
Figura 33 – Gráfico dos modelos escolhidos



Fonte: Elaborada pelo autor

A mesma ideia vale para as frequências das empresas dos modelos, também apresentadas através de um gráfico na Figura 34.

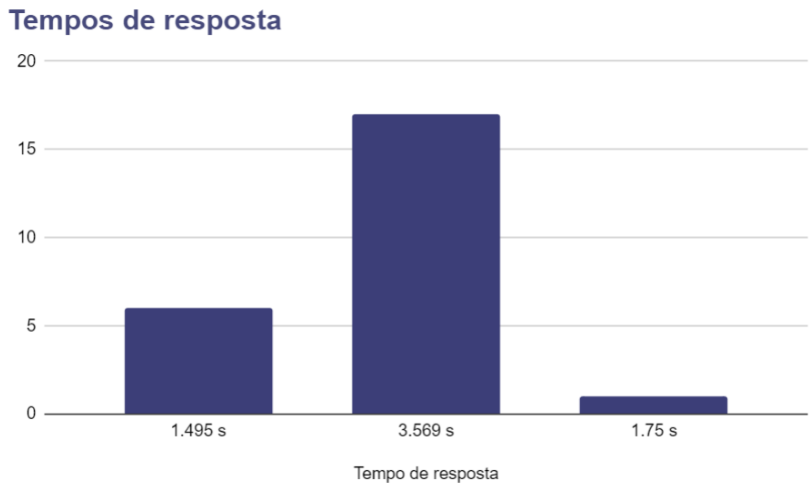
Figura 34 – Gráfico das empresas escolhidas



Fonte: Elaborada pelo autor

Os tempos de resposta também foram contabilizados para a segunda pergunta e são expressados na Figura 35.

Figura 35 – Gráfico dos tempos de resposta (baseados na ocorrência)



Fonte: Elaborada pelo autor

Dessa vez, o modelo da AI21 Labs, o *j2-mid*, não dispõe do menor tempo de resposta. Entretanto, a qualidade dos resultados é mantida, já que seu texto foi considerado o melhor entre todos os produzidos.

Com base na análise do gráfico dos tamanhos das respostas, a partir da Figura 36, é interessante notar que o resultado mais escolhido não é o que possui menor quantidade de caracteres.

Figura 36 – Gráfico dos tamanhos das respostas escolhidas



Fonte: Elaborada pelo autor

Esse fato é importante porque a instrução explicita claramente o tamanho desejado para o texto a ser gerado ao usar o termo “pequena descrição”. Entretanto,

ainda que existam respostas menores, a mais compatível com o requisitado foi a com maior quantidade de caracteres. Uma das explicações possíveis para esse acontecimento é a de que a qualidade do texto criado possa ultrapassar a relevância de outras necessidades do usuário, mesmo que estas estejam expostas em suas instruções.

5 Considerações Finais

Ao entender os conceitos iniciais que explicam o funcionamento dos LLMs, é possível compreender também o impacto causado por eles, em pouco tempo de existência. O desempenho impressionante dos modelos abriu portas para possibilidades de aplicações antes não imagináveis. O site desenvolvido pelo projeto busca ajudar, em menor escala, nas escolhas de modelos para certas tarefas, através da coleta de dados fornecidos pelos usuários.

A fim de que essa ideia fosse uma realidade, os requisitos essenciais foram atendidos, ou seja, a aplicação recebeu meios para enviar e receber informações, e para armazená-las de maneira segura. Além disso, os dados salvos podem ser utilizados de várias formas, a serem decididas por quem executa a análise. Todas essas decisões foram tomadas com o objetivo de oferecer uma opção viável para consultar modelos rapidamente e, ao mesmo tempo, contribuir para um estudo maior das capacidades de cada um deles.

Com os recursos básicos do projeto e com a adição de artifícios que podem melhorar a experiência do usuário, como textos explicativos, foi possível criar um ambiente interativo, capaz de atender as necessidades de quem busca explorar as vantagens proporcionadas pelos grandes modelos.

5.1 Trabalhos Futuros

Ao considerar a continuação do projeto, os seguintes passos podem ser observados:

- Hospedar o site na internet e disponibilizar a aplicação para uso
- Aumentar a quantidade de modelos utilizados
- Aumentar a quantidade de empresas, criadoras de LLMs, escolhidas para fazer parte do projeto
- Promover a coleta de dados em maior escala
- Analisar os dados coletados e gerar um panorama maior dos desempenhos dos modelos

Referências

AGUSTIN, F.; KURNIAWAN, H.; YUSFRIZAL, Y.; UMMI, K. Comparative analysis of application quality between appserv and xampp webserver using ahp based on iso/iec 25010: 2011. 2018. Disponível em: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8674345. Acesso em: 28 outubro 2023.

BAHETI, P. A newbie-friendly guide to transfer learning. 2021. Disponível em: https://www.v7labs.com/blog/transfer-learning-guide. Acesso em: 28 outubro 2023.

BALESTRIERO, R.; IBRAHIM, M.; SOBAL, V.; MORCOS, A.; SHEKHAR, S.; GOLDSTEIN, T.; BORDES, F.; BARDES, A.; MIALON, G.; TIAN, Y. et al. A cookbook of self-supervised learning. 2023. Acesso em: 28 outubro 2023.

BANANE, M.; ERRAISSI, A. A comprehensive study of natural language processing techniques based on big data. 2022. Disponível em: https://ieeexplore.ieee.org/abstract/document/9765270/. Acesso em: 28 outubro 2023.

CARDON, A.; MÜLLER, D. N.; NAVAUX, P. Introdução às redes neurais artificiais. 1994. Disponível em: https://www.academia.edu/download/60901380/intro_rna20191014-14353-doa3qp.pdf. Acesso em: 28 outubro 2023.

CHOWDHARY, K. Fundamentals of artificial intelligence. 2020. Disponível em: https://link.springer.com/content/pdf/10.1007/978-81-322-3972-7.pdf. Acesso em: 28 outubro 2023.

CODINGDOJO. Mysql workbench. 2013. Disponível em: https://static1.squarespace.com/static/503aea8ee4b0b543ed24ddbe/t/63c0a31d5b404e34cc53190f/1673569058486/Stack+4+-+Week+1+-+002+MySQL+Workbench.pdf. Acesso em: 28 outubro 2023.

COHERE. Prompt engineering. 2022. Disponível em: https://docs.cohere.com/docs/prompt-engineering. Acesso em: 28 outubro 2023.

DALE, R. Gpt-3: What's it good for? *Natural Language Engineering*, v. 27, 2021. Disponível em: https://www.cambridge.org/core/journals/natural-language-engineering/article/gpt3-whats-it-good-for/0E05CFE68A7AC8BF794C8ECBE28AA990. Acesso em: 28 outubro 2023.

FIGMA. About figma, the collaborative interface design tool. 2011. Disponível em: https://www.figma.com/about/. Acesso em: 28 outubro 2023.

FLECK, L.; TAVARES, M. H. F.; EYNG, E.; HELMANN, A. C.; ANDRADE, M. A. d. M. Redes neurais artificiais: Princípios básicos. *Revista Eletrônica Científica Inovação e Tecnologia*, v. 1, 2016. Disponível em: https://www.academia.edu/download/57830283/4330-15577-1-PB.pdf. Acesso em: 28 outubro 2023.

HÜLLERMEIER, E.; KRUSE, R.; HOFFMANN, F. Information processing and management of uncertainty in knowledge-based systems. v. 80, 2010. Disponível em: https://link.springer.com/chapter/10.1007/978-3-030-50146-4_3. Acesso em: 28 outubro 2023.

JIANG, E.; OLSON, K.; TOH, E.; MOLINA, A.; DONSBACH, A.; TERRY, M.; CAI, C. J. Promptmaker: Prompt-based prototyping with large language models. 2022. Disponível em: https://dl.acm.org/doi/pdf/10.1145/3491101.3503564. Acesso em: 28 outubro 2023.

KASNECI, E.; SESSLER, K.; KÜCHEMANN, S.; BANNERT, M.; DEMENTIEVA, D.; FISCHER, F.; GASSER, U.; GROH, G.; GÜNNEMANN, S.; HÜLLERMEIER, E. et al. Chatgpt for good? on opportunities and challenges of large language models for education. *Learning and individual differences*, v. 103, 2023. Disponível em: https://www.sciencedirect.com/sdfe/reader/pii/S1041608023000195/pdf. Acesso em: 28 outubro 2023.

KHAN, S.; NASEER, M.; HAYAT, M.; ZAMIR, S. W.; KHAN, F. S.; SHAH, M. Transformers in vision: A survey. *ACM computing surveys (CSUR)*, v. 54, 2022. Disponível em: https://dl.acm.org/doi/pdf/10.1145/3505244. Acesso em: 28 outubro 2023.

LIN, T.; WANG, Y.; LIU, X.; QIU, X. A survey of transformers. *AI Open*, 2022. Disponível em: https://www.sciencedirect.com/sdfe/reader/pii/S2666651022000146/pdf. Acesso em: 28 outubro 2023.

MICROSOFT. Documentation for visual studio code. 2015. Disponível em: https://code.visualstudio.com/docs. Acesso em: 28 outubro 2023.

OPENAI. Openai documentation. 2020. Disponível em: https://platform.openai.com/docs/models/overview. Acesso em: 28 outubro 2023.

OZDEMIR, S. Quick start guide to large language models: Strategies and best practices for using chatgpt and other llms. 2023. Acesso em: 28 outubro 2023.

TUNSTALL, L.; WERRA, L. V.; WOLF, T. Natural language processing with transformers. 2022. Disponível em: https://books.google.com/books?hl=pt-BR&lr=&id=nzxbEAAAQBAJ&oi=fnd&pg=PP1&dq=Natural+language+processing+with+transformers&ots=sTwmIWq3NB&sig=-10AmBD6LroEyoWvl3OV9ii5UPY. Acesso em: 28 outubro 2023.

VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, L. u.; POLOSUKHIN, I. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. Disponível em: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf. Acesso em: 28 outubro 2023.

XIE, H.; QIN, Z.; LI, G. Y.; JUANG, B.-H. Deep learning enabled semantic communication systems. *IEEE Transactions on Signal Processing*, v. 69, 2021. Disponível em: https://ieeexplore.ieee.org/abstract/document/9398576/. Acesso em: 28 outubro 2023.