UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS
DEPARTAMENTO DE COMPUTAÇÃO

LUCAS SILVA BARBOSA

UM *CHATBOT* ESPECIALIZADO PARA O CONTEXTO DA UNIVERSIDADE FEDERAL DE OURO PRETO

Ouro Preto, MG
2023

UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS
DEPARTAMENTO DE COMPUTAÇÃO

LUCAS SILVA BARBOSA

UM *CHATBOT* ESPECIALIZADO PARA O CONTEXTO DA UNIVERSIDADE FEDERAL DE OURO PRETO

Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal de Ouro Preto como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Prof. Dr. Pedro Henrique Lopes Silva

Ouro Preto, MG
2023

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE OURO PRETO
REITORIA
INSTITUTO DE CIENCIAS EXATAS E BIOLOGICAS
DEPARTAMENTO DE COMPUTACAO

FOLHA DE APROVAÇÃO

Lucas Silva Barbosa

Um Chatbot especializado para o contexto da Universidade Federal de Ouro Preto

Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal
de Ouro Preto como requisito parcial para obtenção do título de Bacharel em Ciência da Computação

Aprovada em 7 de Fevereiro de 2024.

Membros da banca:

Pedro Henrique Lopes Silva (Orientador) - Doutor - Universidade Federal de Ouro Preto
Jadson Castro Gertrudes (Examinador) - Doutor - Universidade Federal de Ouro Preto
Guilherme Augusto Lopes Silva (Examinador) - Mestre - Universidade Federal de Ouro Preto

Pedro Henrique Lopes Silva, Orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 7/02/2024.

Documento assinado eletronicamente por **Pedro Henrique Lopes Silva**, **PROFESSOR DE MAGISTERIO SUPERIOR**, em 09/02/2024, às 09:59, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0 , informando o código verificador **0665678** e o código CRC **BCE1BD90**.

R. Diogo de Vasconcelos, 122, - Bairro Pilar Ouro Preto/MG, CEP 35402-163
Telefone: 3135591692 - www.ufop.br

Dedico este trabalho à minha mãe, Jucirene, por ter me ensinado a ter todo o foco e perseverança necessários para chegar até aqui.

Agradecimentos

Agradeço, primeiramente, à minha mãe, Jucirene, por todo o apoio que me deu para chegar até aqui, além de ter me ensinado a não desistir. Ao meu irmão, Matheus, minha saudade diária. À Luísa, por todo o carinho, companheirismo, e por me escolher todos os dias. Você brilha mais do que qualquer outra luz em todo o universo. Aos meus irmãos de vida da República Aquarius, por todo o acolhimento e amizade desde que cheguei a esta que se tornou o meu eterno lar. Ao meu orientador, Prof. Dr. Pedro Henrique Lopes Silva, pela dedicação e paciência fundamentais para o enriquecimento e sucesso deste trabalho. Por fim, à Universidade Federal de Ouro Preto e ao Colegiado do Curso de Ciência da Computação pela excelência no ensino proporcionada ao longo da minha jornada acadêmica.

Trata-se sempre de vencer o caos por um plano secante que o atravessa.

DELEUZE e GUATTARI (1997)

Resumo

Este estudo aborda a concepção e avaliação de um sistema de *chatbot* contextualizado para responder a perguntas específicas da Universidade Federal de Ouro Preto (UFOP), notavelmente na Pró-Reitoria de Graduação (PROGRAD). O *chatbot*, um sistema de conversação automatizada, é integrado à *API* do *ChatGPT* e emprega e técnicas de indexação de documentos para proporcionar respostas eficientes e padronizadas a questões frequentes de estudantes. A pesquisa enfatiza o valor de tal recurso como suporte acadêmico, delineando abordagens de implementação e descrevendo a integração de informações de documentos disponibilizados por diversos setores da UFOP. A conclusão ressalta a relevância do *chatbot* para eficaz atendimento a consultas rotineiras, evidenciando seu potencial no contexto acadêmico e destacando áreas de aprimoramento para sua otimização futura. Destaca-se que o *chatbot* implementado alcançou uma acurácia global de 90,4%, o que reforça sua eficácia e confiabilidade na prestação de assistência aos estudantes da UFOP.

Palavras-chave: Chatbots. Deep Learning. Transformer.

Abstract

This study addresses designing and evaluating a contextualized chatbot system to answer specific questions at the Federal University of Ouro Preto (UFOP), notably at the Dean of Undergraduate Studies. Chatbot, an automatic chat system, is integrated with the API of ChatGPT and employs document indexing techniques to provide efficient and standardized answers to frequently asked questions from students. The research emphasizes the value of such a resource as academic support, outlining implementation approaches and describing the integration of information from documents made available by different sectors of UFOP. The conclusion highlights the relevance of chatbots for effectively responding to routine queries, highlighting their potential in the academic context, and highlighting areas for improvement for future optimization. Notably, the implemented chatbot achieved an overall accuracy of 90.4%, reinforcing its effectiveness and reliability in assisting UFOP students.

Keywords: Chatbots, Deep Learning, Transformer.

Lista de Ilustrações

Lista de Tabelas

Lista de Abreviaturas e Siglas

A.L.I.C.E.	*Artificial Linguistic Internet Computer Entity*
AIML	*Artificial Intelligence Markup Language*
ANN	*Artifical Neural Network*
API	*Application Programming Interface*
BLEU	*Bilingual Evaluation Understudy*
COCIC	Colegiado de Ciência da Computação
DECOM	Departamento de Computação
DINA	*Dinus Intelligente Assistance*
FAQ	*Frequently Asked Questions*
GPT	*Generative Pre-trained Transformer*
HTTP	*Hypertext Transfer Protocol*
JSON	*JavaScript Object Notation*
LLM	*Large Language Model*
LSA	*Latent Semantic Analisys*
MIT	*Massachusetts Institute of Technology*
PLN	Processamento de Linguagem Natural
PRACE	Pró-Reitoria de Assuntos Comunitários e Estudantis
PROGRAD	Pró-Reitoria de Graduação
PROPPI	Pró-Reitoria de Pesquisa, Pós-Graduação e Inovação
UDINUS	*Universitas Dian Nuswantoro*
UFOP	Universidade Federal de Ouro Preto

Sumário

1 Introdução

Chatbots são sistemas de conversação que podem fazer interações de bate-papo com humanos automaticamente. Eles são desenvolvidos para serem assistentes virtuais, proporcionando entretenimento para as pessoas, ajudando a responder perguntas, obter instruções de direção, servir como parceiros humanos em casas inteligentes, dentre vários outros propósitos (KHIN; SOE, 2020).

Segundo Adamopoulou e Moussiades (2020a), *chatbots* são importantes como assistentes virtuais porque podem economizar tempo e recursos automatizando tarefas que, de outra forma, exigiriam intervenção humana. Eles também podem fornecer uma experiência de usuário mais personalizada e eficiente. Além disso, um *chatbot* pode ser usado como um serviço de atendimento ao cliente que pode fornecer uma resposta rápida e otimizada (CHANDRA; SUYANTO, 2019).

Em determinados contextos empresariais e institucionais, é comum ocorrerem situações em que os funcionários responsáveis pelo atendimento ao público se veem obrigados a responder repetidamente a mesma dúvida, proveniente de diferentes indivíduos, apresentando a mesma resposta, porque muitas dessas dúvidas a serem respondidas por estes funcionários possuem respostas objetivas. Nesta situação, a introdução de um sistema de *chatbot* com respostas pré-determinadas poderia proporcionar benefícios em termos de economia de tempo e esforço tanto para os usuários quanto para os funcionários encarregados de fornecer essas respostas (SANTOSO et al., 2018). O contexto específico abordado neste trabalho diz respeito à Universidade Federal de Ouro Preto (UFOP), mais especificamente à Pró-Reitoria de Graduação (PROGRAD).

A PROGRAD, além de suas diversas responsabilidades, lida com as dúvidas dos estudantes de graduação relacionadas a vários assuntos dentro da universidade. Essas dúvidas incluem questões como localização de prédios e salas, processo para obter bolsas de auxílio para alimentação ou permanência, trancamento de disciplinas, estágios obrigatórios e não obrigatórios, bem como as regras para colação de grau, entre outras. Todos esses exemplos compartilham a característica de possuírem respostas padronizadas. Portanto, fica evidente a viabilidade de implementação de um sistema de *chatbot* para auxiliar na resposta a essas questões, o qual tem o potencial de reduzir o tempo e o esforço despendidos pelos funcionários ao responder perguntas que possuem respostas objetivas (GAO; JIANG, 2021).

O presente estudo aborda a implementação e avaliação de um sistema de *chatbot* contextualizado para responder a perguntas específicas relacionadas à Universidade Federal de Ouro Preto (UFOP), em especial à Pró-Reitoria de Graduação (PROGRAD). O sistema de *chatbot* foi desenvolvido visando automatizar respostas padronizadas a perguntas frequentes de estudantes de graduação, abrangendo áreas como localização de instalações, processos administrativos e regulamentos acadêmicos.

Por meio da integração com a API (*Application Programming Interface*) do *ChatGPT* e técnicas de indexação de documentos, o *chatbot* foi treinado para compreender e gerar respostas contextuais baseadas em informações contidas no Manual do Aluno da UFOP e em outros documentos relevantes. A implementação do *chatbot* tem como objetivo oferecer uma solução eficiente e acessível para o esclarecimento de dúvidas rotineiras dos estudantes, reduzindo o tempo e o esforço necessários para fornecer respostas padronizadas.

Foram conduzidos experimentos de teste de acurácia abrangendo uma variedade de perguntas relacionadas a diferentes intenções do *chatbot*. Os resultados revelaram que o *chatbot* alcançou um notável grau de precisão, atingindo taxas de acerto de 100% para algumas intenções específicas. Globalmente, o *chatbot* manteve uma acurácia consistente, registrando pelo menos 70% de acerto em todas as intenções avaliadas.

Entretanto, observou-se uma redução na acurácia para algumas intenções específicas, indicando a necessidade de aprimoramentos. Essa diminuição sugere a necessidade de melhorias nos dados utilizados durante o treinamento do *chatbot*, destacando a importância da coleta de informações mais abrangentes e detalhadas de diversas áreas da UFOP. O processo de aprimoramento dos dados pode ser fundamental para a otimização do desempenho do *chatbot*, tornando-o mais eficaz na abordagem de uma gama mais ampla de consultas acadêmicas.

1.1 Justificativa

A implementação de um sistema de *chatbot* para uso da Pró-Reitoria de Graduação (PROGRAD) da Universidade Federal de Ouro Preto (UFOP) apresenta diversas vantagens e benefícios. Atualmente, os funcionários da PROGRAD podem ter de lidar com perguntas repetitivas e questões com respostas padronizadas dos estudantes de graduação. Essas demandas consomem tempo e recursos humanos que poderiam ser direcionados para atividades mais complexas e estratégicas. Ao introduzir um *chatbot*, é possível automatizar o processo de atendimento e proporcionar respostas rápidas e precisas aos estudantes, reduzindo o esforço humano envolvido e aumentando a eficiência do serviço prestado pela PROGRAD.

1.2 Objetivos

O objetivo principal deste projeto é desenvolver um sistema de *chatbot* para uso da Pró-Reitoria de Graduação (PROGRAD) da Universidade Federal de Ouro Preto (UFOP). O *chatbot* terá como finalidade automatizar o atendimento e fornecer respostas precisas para as perguntas mais frequentes dos estudantes de graduação. Com ênfase na eficiência, o *chatbot* estará disponível 24 horas por dia, proporcionando assistência personalizada e precisa. Dessa forma, busca-se reduzir significativamente o tempo e o esforço investidos no atendimento das demandas dos estudantes.

Os objetivos específicos deste trabalho são:

- Utilizar a API do ChatGPT para integrar a funcionalidade de processamento de linguagem natural e reconhecimento de padrões ao *chatbot*, permitindo que ele compreenda adequadamente as perguntas dos estudantes e forneça respostas contextuais, considerando a diversidade de formas como as perguntas podem ser formuladas.
- Coletar documentos oficiais da PROGRAD para contextualizar o ChatGPT.
- Comparar os resultados com outros trabalhos na literatura.
- Documentar todo o processo de integração e uso da API do ChatGPT, criando um guia claro e detalhado para facilitar a manutenção e futuras atualizações do *chatbot*.

1.3 Organização do Trabalho

Este trabalho está estruturado em cinco capítulos distintos. No Capítulo 2, será apresentada a fundamentação teórica, onde serão explicados os conceitos essenciais que permeiam todo o projeto, além de serem abordados alguns trabalhos relacionados ao tema. O Capítulo 3 compreende o desenvolvimento do *chatbot*, detalhando o processo de criação e implementação do sistema. Os resultados obtidos, bem como os testes realizados, serão apresentados no Capítulo 4. Por fim, o Capítulo 5 engloba as conclusões do estudo, abordando também possíveis impedimentos e limitações encontradas, além de fornecer *insights* para futuros trabalhos nesta área.

2 Revisão Bibliográfica

Este capítulo tem como objetivo realizar uma exploração do conhecimento pré-existente e das pesquisas correlacionadas ao desenvolvimento de *chatbots*, bem como elucidar alguns conceitos empregados no decorrer deste trabalho, que são fundamentais para a sua compreensão.

2.1 Fundamentação Teórica

2.1.1 Evolução dos *chatbots*

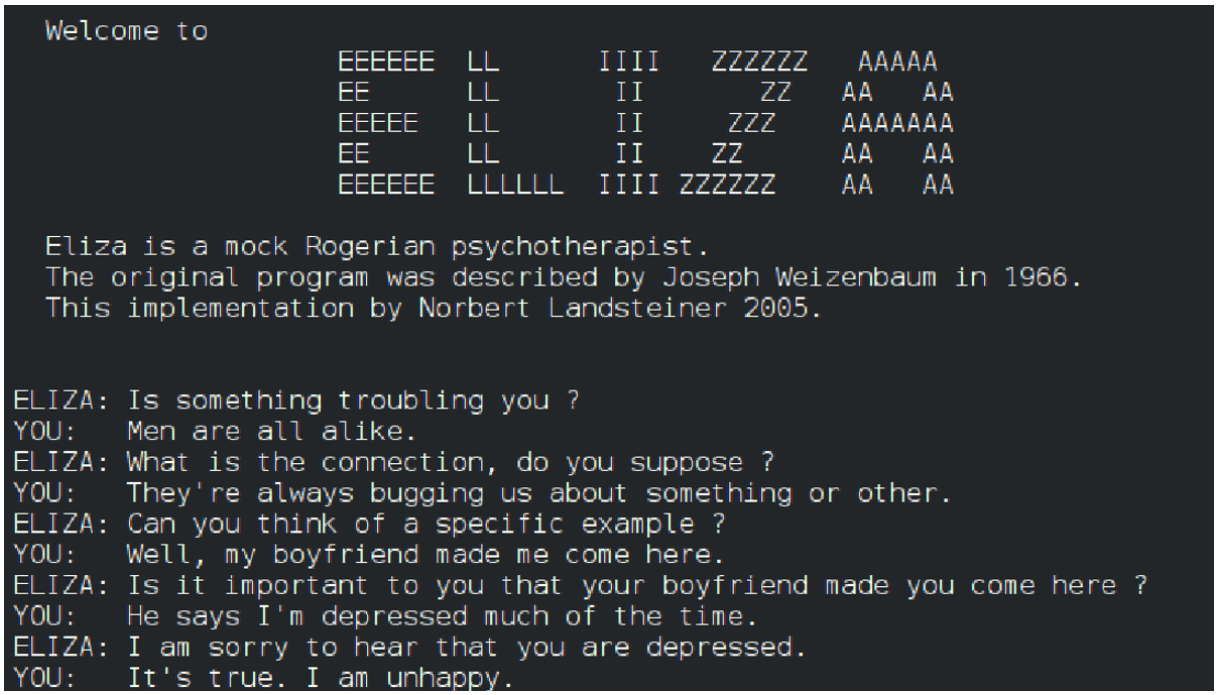
Os *chatbots* são programas de software concebidos para interagir com pessoas por meio de linguagem natural, desempenhando o papel de facilitadores na aquisição de informações por meio de sistemas de diálogo. Essas aplicações comunicam-se com os usuários em diversos domínios específicos, respondendo a suas consultas com declarações gerais de conversação. Inicialmente desenvolvidos com propósitos de entretenimento, com o intuito de simular conversas humanas, os *chatbots* evoluíram ao ponto de conquistar ampla utilização em diversos setores, abrangendo até mesmo o mundo corporativo. Uma das abordagens que vem ganhando crescente popularidade para a implementação de *chatbots* envolve o emprego do processamento de linguagem natural (PLN), uma técnica que permite ao computador processar textos que representam a linguagem humana (KARRI; KUMAR, 2020).

Um dos desafios mais significativos no campo da Inteligência Artificial reside na capacidade de viabilizar a comunicação entre máquinas e seres humanos por meio do uso da linguagem natural. Os sistemas conversacionais pioneiros, predecessores dos *chatbots* contemporâneos, foram concebidos com o propósito de emular o comportamento humano em diálogos baseados em texto. No entanto, esses sistemas eram limitados por regras pré-definidas, criadas manualmente, e por ambientes de interação restritos, o que os impedia de compreender conversas de forma profunda e abrangente (SHUM; HE; LI, 2018). Um exemplo notável é o sistema **ELIZA** (WEIZENBAUM, 1966).

Bhagwat (2018) aborda a progressão histórica dos *chatbots* desde sua origem com o pioneiro **ELIZA** em 1966, desenvolvido pelo **MIT**. Ao analisar as frases de entrada com base nas regras de decomposição acionadas por palavras-chave, o **ELIZA** gera respostas usando regras de remontagem associadas às regras de decomposição selecionadas (Figura 2.1). Além disso, o componente **SLIP** permite a marcação de palavras em um texto e sua posterior recuperação com base em suas *tags*, facilitando a identificação de palavras-chave. Por exemplo, a palavra-chave "MÃE?" em **ELIZA** pode ser identificada como substantivo e membro da classe "família" (WEIZENBAUM, 1966). Essa estrutura simples baseada em regras serviu de inspiração para muitos *chatbots* subsequentes. No entanto, escalar esses *chatbots* para lidar com volumes substanciais

de interações representa um desafio significativo (BHAGWAT, 2018).

Figura 2.1 – Exemplo de conversa com o *chatbot* **ELIZA**, que simula um psicoterapeuta.



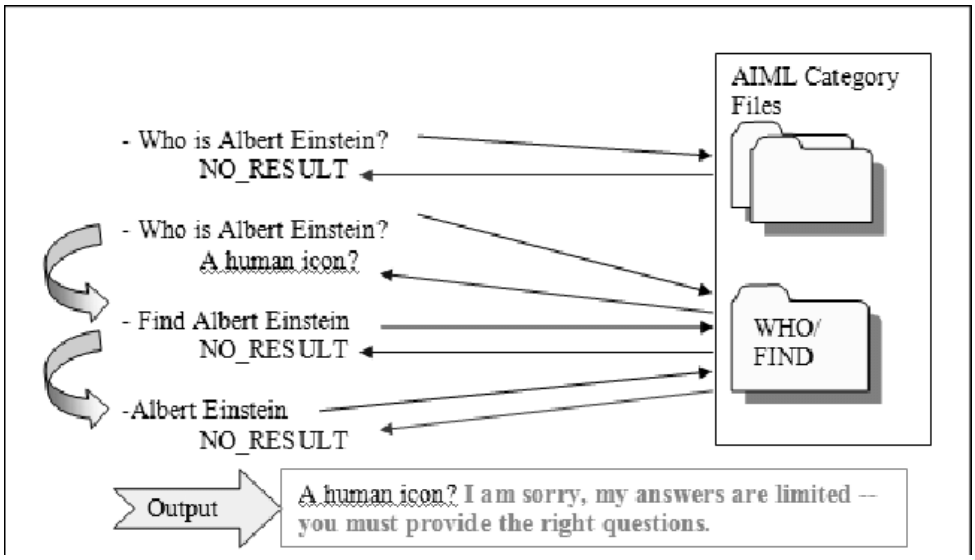
Fonte: Retirado de
<https://olhardigital.com.br/2023/04/13/pro/mae-dos-chatbots-robo-do-mit-fingia-ser-inteligente/>.
Acesso em 26/12/2023.

O estudo de Bhagwat (2018) também aborda a história do *chatbot* **A.L.I.C.E.** (*Artificial Linguistic Internet Computer Entity*), desenvolvido em 1995 pelo Dr. Richard Wallace. O **A.L.I.C.E.** foi criado utilizando AIML (*Artificial Intelligence Markup Language*), um dialeto do XML (*Extensible Markup Language*), desenvolvido pelo próprio Dr. Wallace, especificamente concebido para o desenvolvimento de *chatbots*. A arquitetura do **A.L.I.C.E.** funciona separando o mecanismo do *chatbot* — responsável por interagir com os usuários usando linguagem natural e fornecer respostas com base em um conjunto de regras de correspondência de modelos de padrões — e o modelo de conhecimento linguístico — codificado manualmente no formato AIML, contendo um grande conjunto de regras de correspondência de modelos de padrões que permitem ao *chatbot* fornecer respostas sem a necessidade de análise sofisticada de linguagem natural ou inferência lógica —, permitindo flexibilidade e adaptabilidade para que conhecimento de diferentes idiomas sejam conectados e reproduzidos. O mecanismo do *chatbot* interage com os usuários usando linguagem natural, não exigindo processamento sofisticado ou técnicas complexas de aprendizado de máquina (ABUSHAWAR; ATWELL, 2015).

A Figura 2.2 representa a interação entre um usuário e o **A.L.I.C.E.**. O usuário faz várias perguntas sobre Albert Einstein, mas o sistema não consegue fornecer resultados satisfatórios até que a pergunta seja formulada de maneira específica. As perguntas são: "*Who is Albert Einstein?*", "*Find Albert Einstein*", e simplesmente "Albert Einstein". Todas as três primeiras perguntas resultam em "*NO RESULT*". Uma segunda tentativa com a pergunta "*Who is Albert*

Einstein?" resulta na resposta "*A human icon?*". Há duas pastas rotuladas como "*AIML Category Files*" e "*WHO/ FIND*" no lado direito do diagrama, indicando categorias de respostas ou arquivos de dados. Outro *chatbot* mencionado em Bhagwat (2018) é o **Jabberwacky**, desenvolvido pelo programador britânico Rollo Carpenter. O **Jabberwacky** foi um dos primeiros *chatbots* a empregar inteligência artificial para aprender interações humanas de forma significativa. Sua capacidade de aprendizado contínuo permitia assimilar informações por meio de conversas e adquirir novos idiomas, conceitos e fatos apenas por meio de interações constantes. O **Jabberwacky** não tem regras predefinidas e depende muito do contexto e do *feedback* dos usuários.

Figura 2.2 – Diagrama de fluxo que representa a interação entre um usuário e o *chatbot* **A.L.I.C.E.**.



Fonte: Preez, Lall e Sinha (2009).

Existem duas tecnologias fundamentais para a implementação de *chatbots*: a abordagem de correspondência de padrões (*Pattern Matching*) e o aprendizado de máquina (*Machine Learning*). A primeira é caracterizada por sua simplicidade, velocidade e facilidade de implementação. No entanto, essa abordagem possui limitações ao se restringir a respostas predefinidas, resultando em falta de originalidade e incapacidade de lidar com variações significativas na entrada do usuário.

Por outro lado, a tecnologia de aprendizado de máquina permite que os *chatbots* utilizem algoritmos para aprender com as interações dos usuários. Essa abordagem demanda um grande volume de dados de treinamento, podendo ser computacionalmente cara. No entanto, oferece a vantagem de lidar com variações na entrada do usuário, capacidade para gerar novas respostas e aprimoramento contínuo de seu desempenho ao longo do tempo (ADAMOPOULOU; MOUSSIADES, 2020b). Para o presente trabalho, a tecnologia selecionada para implementação é a do aprendizado de máquina.

2.1.2 Processamento de Linguagem Natural

A história do processamento de linguagem natural (PLN) remonta à década de 1950, quando surgiu como a interseção entre inteligência artificial e linguística. O PLN trata computacionalmente os diversos aspectos da comunicação humana, como sons, palavras, sentenças e discursos, considerando formatos e referências, estruturas e significados, contextos e usos. Em sentido amplo, pode-se dizer que o PLN visa fazer o computador se comunicar em linguagem humana (GONZALEZ; LIMA, 2003). Atualmente, o PLN se baseia em vários campos muito diversos, exigindo que os pesquisadores e desenvolvedores da área de hoje ampliem significativamente sua base de conhecimento mental (NADKARNI; OHNO-MACHADO; CHAPMAN, 2011).

Gonzalez e Lima (2003) descrevem que o processamento de linguagem natural possui níveis de compreensão, sendo divididos em:

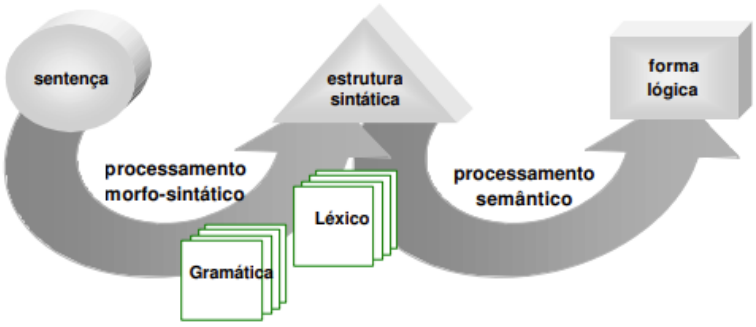
- **Nível fonético e fonológico**: Este nível trata da relação entre as palavras e os sons que elas produzem. Envolve a compreensão dos aspectos fonéticos e fonológicos da linguagem, como pronúncia e padrões de som.
- **Nível morfológico**: Nesse nível, o foco está na construção de palavras a partir de unidades primitivas de significado e como classificá-las em categorias morfológicas. Envolve a compreensão da estrutura e formação das palavras, incluindo prefixos, sufixos e palavras-raiz.
- **Nível sintático**: O nível sintático trata da relação entre as palavras em uma frase, onde cada palavra assume um papel estrutural específico. Envolve a compreensão da gramática e da sintaxe de um idioma, incluindo a estrutura das frases, a ordem das palavras e a formação de frases e sentenças.
- **Nível semântico**: O nível semântico se concentra na relação entre as palavras e seus significados, bem como na forma como esses significados são combinados para formar os significados das frases. Envolve a interpretação de palavras e frases, incluindo a desambiguação do sentido das palavras e a rotulagem de papéis semânticos.
- **Nível pragmático**: O nível pragmático envolve o uso de frases e sentenças em diferentes contextos, o que pode afetar seu significado. Envolve a compreensão dos aspectos pragmáticos da linguagem, como contexto, intenção e uso da linguagem em situações específicas.

O significado de uma frase, independente do contexto, pode ser representado por meio de sua forma lógica. A forma lógica codifica os possíveis sentidos de cada palavra e identifica as relações semânticas entre palavras e frases. Ao determinar essas relações semânticas, alguns

sentidos das palavras podem se tornar inviáveis e podem ser desconsiderados (GONZALEZ; LIMA, 2003).

A Figura 2.3 representa um modelo simplificado de como os sistemas de processamento de linguagem natural podem analisar e interpretar a linguagem humana. O processo começa com uma sentença que passa por processamento morfo-sintático, onde a sentença é analisada em termos de sua estrutura morfológica e sintática. Este processo envolve o uso de um Léxico e uma Gramática. A sentença então é transformada em uma estrutura sintática. A próxima etapa é o processamento semântico, onde a estrutura sintática é interpretada para determinar o significado da sentença. Finalmente, a sentença é transformada em uma forma lógica, que representa o significado da sentença de uma maneira que pode ser processada por um computador (GONZALEZ; LIMA, 2003).

Figura 2.3 – Diagrama que representa o processo de análise linguística.



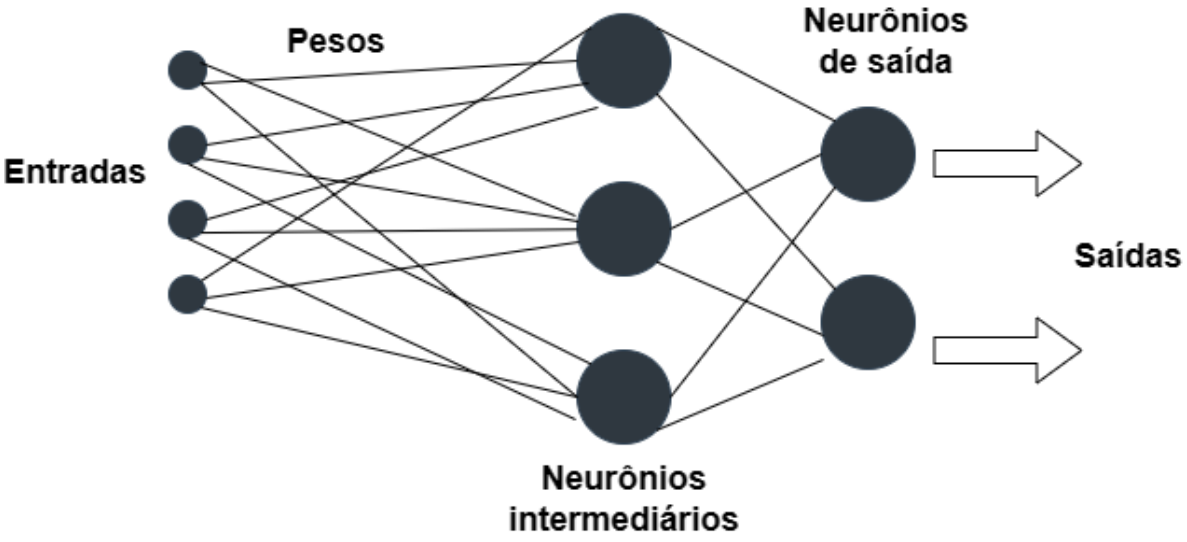
Fonte: Gonzalez e Lima (2003).

O PLN tem uma ampla variedade de aplicações, incluindo tradução automática (HAN; YIN, 2021), análise de sentimentos (CHRISTI; JAIN, 2020), reconhecimento de fala (CASTIL-LEJO, 2021) e resumo de texto (AWASTHI et al., 2021).

2.1.3 Rede Neural Artificial

As redes neurais artificiais são programas de computador desenvolvidos para emular o funcionamento do cérebro humano. Elas possuem a capacidade de aprender com exemplos e podem ser treinadas para realizar diversas tarefas, como reconhecimento de fala (HUSSAIN et al., 2021), previsão da estrutura secundária de proteínas (AKBAR; PARDASANI; KHAN, 2021), classificação de cânceres e predição de genes (MANIMEGALA; PRIYA; RANJANA, 2020). O processo de aprendizagem consiste em ajustar os pesos e limites da rede a cada novo exemplo apresentado, buscando aprimorar sua capacidade de classificação, como ilustrado na Figura 2.4 . Os neurônios recebem os sinais de entrada. O corpo do neurônio realiza a soma da multiplicação termo a termo entre os sinais de entrada e os pesos sinápticos que informam o quão relevante é cada uma das entradas no neurônio. O resultado da soma da multiplicação representa a saída do corpo celular artificial (KROGH, 2008).

Figura 2.4 – Exemplo de uma rede neural artificial.



Fonte: Adaptação de <https://docs.ufpr.br/~marianakleina/RNA.pdf>. Acesso em 29/07/2023.

Segundo Krogh (2008), para treinar uma rede, diversos algoritmos podem ser empregados, e existem programas disponíveis para criar redes neurais artificiais e treiná-las com dados específicos. No contexto das redes neurais artificiais, os pesos e limites referem-se aos parâmetros que são ajustados durante o processo de treinamento para melhorar a precisão da rede. Os pesos determinam a força das conexões entre os neurônios, enquanto os limites definem o ponto em que um neurônio irá disparar.

Além das redes neurais artificiais, um avanço significativo no campo da inteligência artificial é o conceito de *Deep Learning* (HA; TANG, 2021).

Para criar um *chatbot* eficaz na atualidade, pesquisadores e desenvolvedores precisam ter uma base sólida em aprendizado profundo (*Deep Learning*), um subconjunto do aprendizado de máquina em inteligência artificial (NGUYEN et al., 2021). Em sua pesquisa, Nguyen et al. (2021) integram modelos de aprendizado profundo aos *chatbots* para melhorar sua precisão e capacidade de entender e responder às consultas dos usuários.

2.1.4 *Deep Learning*

O aprendizado profundo (em inglês *Deep Learning*) é um subcampo do aprendizado de máquina (*Machine Learning*) que envolve o treinamento de redes neurais artificiais para aprender representações de dados. Essas redes são compostas por várias camadas de nós interconectados que processam dados de entrada e gradualmente aprendem a extrair recursos de alto nível. O aprendizado profundo tem sido bem-sucedido em uma ampla variedade de aplicações, incluindo visão computacional (VOULODIMOS et al., 2018), processamento de linguagem natural (DENG; LIU, 2018) e reconhecimento de fala (ARSENOVIC et al., 2017), devido à sua capacidade de aprender automaticamente representações complexas de dados (LECUN; BENGIO; HINTON,

2015a).

A tecnologia de aprendizado profundo funciona usando o algoritmo de retropropagação para ajustar os parâmetros internos do modelo em cada camada com base na representação na camada anterior. Isso permite que o modelo descubra estruturas complexas em grandes conjuntos de dados e aprenda padrões complexos que seriam difíceis de detectar usando métodos tradicionais de aprendizado de máquina (LECUN; BENGIO; HINTON, 2015b).

Além disso, o aprendizado profundo é, também, um subconjunto do aprendizado por representação (*Representation Learning*). O aprendizado por representação modela o cérebro humano, com neurônios cerebrais análogos às unidades de computação e a força das conexões entre os neurônios análoga aos pesos. A arquitetura de aprendizado profundo é semelhante a uma rede neural artificial (*Artificial Neural Network* - ANN), mas com mais camadas ocultas, o que nos permite modelar as funções mais complexas do nosso cérebro (BHAGWAT, 2018).

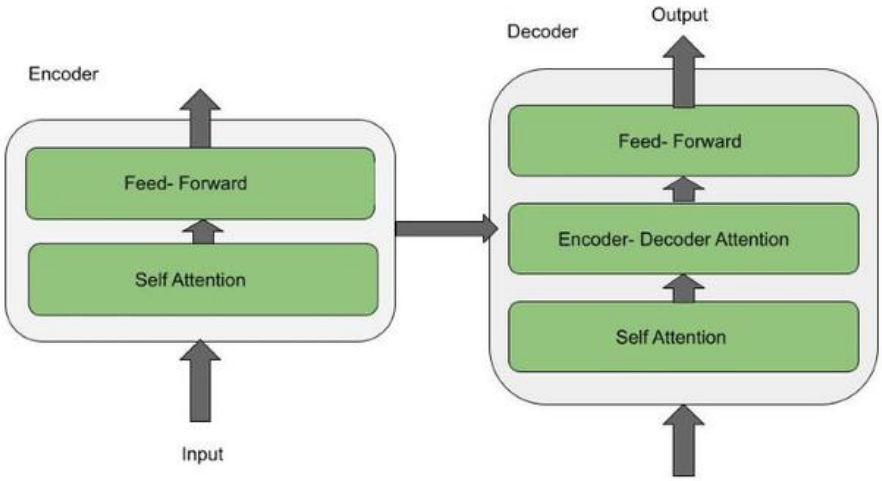
Uma das técnicas mais influentes e populares no aprendizado profundo para Processamento de Linguagem Natural são os modelos de linguagem baseados em *transformers* (MIN et al., 2021).

2.1.5 *Transformer*

O *transformer* é uma arquitetura de aprendizado profundo que tem sido amplamente adotada em vários campos, como processamento de linguagem natural, visão computacional e processamento de fala (LIN et al., 2022). Foi originalmente proposto como um modelo de sequência a sequência para tradução automática, mas trabalhos posteriores mostraram que modelos pré-treinados baseados em *transformers* podem alcançar desempenhos de estado-da-arte em várias tarefas (LIN et al., 2022).

A arquitetura de um *transformer* (Figura 2.5) consiste em um codificador e um decodificador, cada um dos quais é composto por várias camadas de autoatenção, que permitem que o modelo se concentre em diferentes partes da sequência ao gerar a saída, e redes neurais de *feedback*. A camada de autoatenção calcula as pontuações de atenção entre diferentes partes da entrada e as usa para ponderar a importância de cada parte na geração da saída. A camada de *feedback* aplica uma transformação não linear à saída da camada de autoatenção. O codificador processa a sequência de entrada e gera um conjunto de representações ocultas, enquanto o decodificador usa essas representações para gerar a sequência de saída (LIN et al., 2022).

Figura 2.5 – Arquitetura de um *transformer*. A arquitetura do codificador tem duas camadas: Autoatenção e *Feed Forward*. As entradas do codificador passam primeiro por uma camada de autoatenção e, em seguida, as saídas da camada de autoatenção são alimentadas para uma rede neural *feed-forward*. O decodificador tem tanto a camada de autoatenção quanto a de *feed-forward*, que também estão presentes no codificador, mas entre elas está uma camada de atenção que ajuda o decodificador a se concentrar em partes relevantes da frase de entrada.



Fonte: Retirado de <https://acervolima.com/introducao-aos-transformers/>. Acesso em 22/07/2023.

Uma implementação notável do aprendizado profundo que utiliza a arquitetura *transformer* é o GPT (*Generative Pre-Trained Transformer*) (TOPAL; BAS; HEERDEN, 2021).

2.1.6 GPT

O *Generative Pre-Trained Transformer*, ou GPT, é um modelo de processamento de linguagem natural. Ele é um modelo que tem em sua base *transformers*, com arquitetura idêntica à apresentada na seção anterior, e que usa aprendizado não supervisionado para gerar texto semelhante ao humano. O treinamento do modelo é realizado de forma prévia, onde ele é exposto a grandes quantidades de dados linguísticos, permitindo que o modelo aprenda os padrões estatísticos e as relações presentes na linguagem. Esses padrões e relações aprendidos são posteriormente utilizados pelo modelo para gerar novos textos de forma coerente e natural (KUBLIK; SABOO, 2022).

O GPT-1 foi o primeiro modelo de linguagem generativa pré-treinado que usou um decodificador de *transformer* e obteve melhorias significativas no desempenho com um simples ajuste fino. O GPT-2 melhorou o GPT-1 introduzindo informações de tarefas (que incluem tradução, classificação de texto, preenchimento de lacunas, entre outras tarefas de processamento de linguagem natural) durante o treinamento do modelo, usando mais dados de treinamento (40 GB versus 5 GB) e criando um modelo com uma escala de parâmetros maior (1,5 bilhão versus 117 milhões). O GPT-3 teve um aumento significativo, com um espaço de parâmetros expandido

de 175 bilhões e escala de dados de 45 TB (ZHANG; LI, 2021).

Segundo Zhang e Li (2021), o GPT-3 alcançou excelentes resultados em tarefas como geração de texto, adição matemática, geração de artigos de notícias, interpretação de vocabulário e redação de código.

O GPT-4 é o sucessor do GPT-3 e é considerado um modelo que apresentou avanços significativos no processamento de linguagem natural, que passou a aceitar entradas de imagem e texto e produzir saídas de texto, tornando-o mais versátil e capaz do que seus antecessores (OPENAI, 2023). Espera-se que o GPT-4 redefina o cenário do processamento de linguagem natural e desbloqueie novas possibilidades em vários domínios, pois representa a mais recente iteração dos modelos de linguagem da OpenAI (SINGH; SINGH, 2023).

O GPT-4 apresenta desempenho de nível humano em várias referências profissionais e acadêmicas, incluindo a aprovação em um exame simulado da Ordem dos Advogados com uma pontuação em torno dos 10% melhores candidatos, indicando uma melhoria significativa em sua compreensão linguística e capacidade de geração. Apesar disso, seus desenvolvedores alertam que o modelo não é totalmente confiável e pode "alucinar" fatos e cometer erros de raciocínio, o que pode ser problemático em contextos de alto risco. Deve-se tomar cuidado ao usar os resultados do modelo de linguagem, e protocolos específicos, como revisão humana, fundamentação em contexto adicional ou evitar totalmente usos de alto risco, devem ser seguidos (OPENAI, 2023).

A ampliação dos modelos de linguagem pode melhorar o desempenho independente de tarefas, às vezes até mesmo alcançando competitividade com abordagens anteriores de ajuste fino de última geração. Isso significa que os modelos de linguagem podem funcionar bem em uma variedade de tarefas de PLN sem exigir conjuntos de dados de ajuste fino específicos de milhares ou dezenas de milhares de exemplos (BROWN et al., 2020).

A progressão na capacidade de geração de texto possibilitou a origem do *chatbot* da OpenAI, o ChatGPT, o qual atualmente utiliza os modelos GPT-3.5 na versão gratuita e GPT-4 na versão por assinatura.

2.1.7 ChatGPT

O ChatGPT é o mais recente desenvolvimento no campo dos *chatbots*, que são sistemas inteligentes que usam modelos de processamento de linguagem natural para entender a linguagem humana (TAECHARUNGROJ, 2023). Trata-se de um *chatbot* altamente sofisticado baseado na tecnologia do modelo de linguagem GPT. Ele é capaz de atender a uma ampla variedade de solicitações baseadas em texto, incluindo responder perguntas simples e concluir tarefas mais avançadas, como gerar cartas de agradecimento e orientar pessoas em discussões difíceis sobre questões de produtividade. O ChatGPT aproveita seus extensos armazenamentos de dados e design eficiente para entender e interpretar as solicitações dos usuários e, em seguida, gera

respostas apropriadas em linguagem humana quase natural. (LUND; TING, 2023)

Segundo Sohail et al. (2023), a arquitetura do ChatGPT é dividida em três componentes principais:

- **Processamento de entrada**: esse componente pega a entrada do usuário e a processa para extrair informações relevantes. Ele usa técnicas de processamento de linguagem natural, como marcação de parte da fala e reconhecimento de entidades nomeadas, para entender a consulta do usuário.
- **Modelo GPT**: esse componente gera respostas à consulta do usuário com base na entrada processada na etapa anterior. O modelo GPT é um modelo de aprendizado profundo que foi pré-treinado em um grande corpus de dados de texto. Ele usa esse pré-treinamento para gerar respostas contextualmente relevantes para a consulta do usuário.
- **Geração de saída**: esse componente pega a resposta gerada pelo modelo GPT e a formata em um formato legível por humanos. Ele também executa tarefas de pós-processamento, como verificação ortográfica e correção gramatical, para garantir que a resposta seja precisa e gramaticalmente correta.

O ChatGPT se tornou amplamente usado em empresas, agências governamentais e organizações sem fins lucrativos devido à sua disponibilidade, conveniência, baixo custo e melhor experiência do usuário (TAECHARUNGROJ, 2023).

2.2 Trabalhos Relacionados

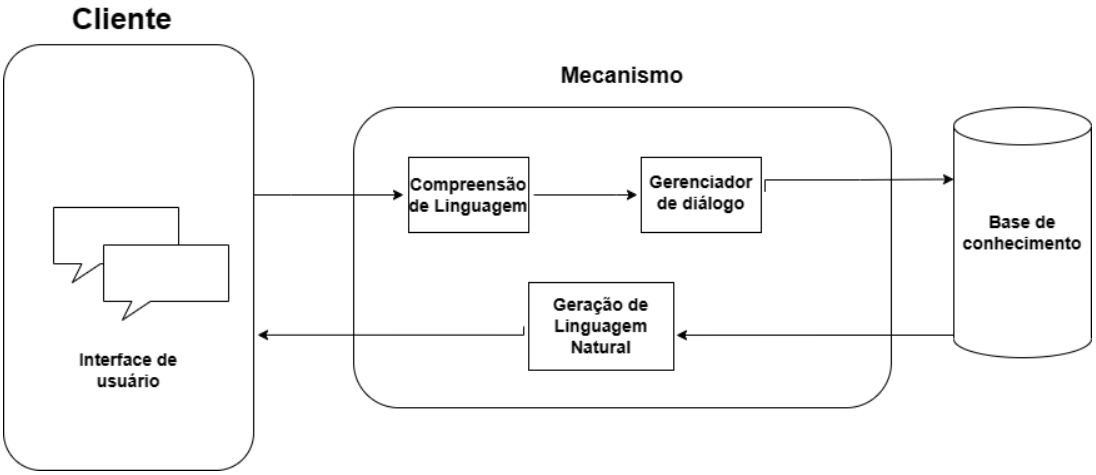
Atmauswan e Abdullahi (2022) apresentam o desenvolvimento de um *chatbot* inteligente para sistema de informação universitário usando abordagem de linguagem natural. Foi construído com uso do *Dialogflow*, uma plataforma de compreensão de linguagem natural usada para projetar e integrar uma interface de usuário conversacional em diversas aplicações. Também utilizou técnicas de processamento de linguagem natural. O *chatbot* obtém seu conhecimento das perguntas frequentes dos alunos sobre a Universidade Internacional de Albukhary, e os dados das perguntas são obtidos em colaboração com a Unidade de Bolsas de Estudo, o Departamento de Admissão e Gestão Acadêmica e a Unidade Internacional de Estudantes. O material é compilado em um conjunto de perguntas frequentes (em inglês *Frequently Asked Questions*, ou FAQ), incluindo informações sobre bolsas de estudo, informações sobre cursos, procedimentos de admissão e informações sobre vistos de estudante. Depois que o conjunto de dados é coletado, os autores determinam a intenção - o propósito ou objetivo da pergunta de um usuário - e a entidade - informações específicas dentro da pergunta de um usuário que são relevantes para a intenção - de cada pergunta coletada. Os *chatbots* usam intenções e entidades para gerar respostas a partir de texto semiestruturado, como perguntas frequentes, enquanto respondem a perguntas simples.

Foi implementado uma busca por respostas por meio de perguntas inseridas pelo usuário em uma janela de bate-papo ao vivo. A arquitetura do *chatbot* inclui componentes para compreensão de linguagem, compreensão de linguagem natural, gerenciamento de diálogos, base de conhecimento e geração de linguagem natural.

Com base na Figura 2.6, a arquitetura do *chatbot* usando processamento de linguagem natural consiste em vários componentes, segundo Atmauswan e Abdullahi (2022):

- O Componente de Compreensão de Linguagem analisa as solicitações do usuário para inferir intenções e entidades.
- O componente de Gerenciamento de Diálogos decide como proceder com base na melhor interpretação e gera respostas voltadas para o usuário.
- O componente da base de conhecimento formaliza os dados para oferecer gerenciamento de conversas e o componente Geração de Linguagem Natural gera respostas de texto em linguagem natural.

Figura 2.6 – Arquitetura do *chatbot* proposto por Atmauswan e Abdullahi (2022) para um *chatbot* para sistema de informação universitário usando abordagem de linguagem natural. O Componente de Compreensão de Linguagem analisa as solicitações do usuário para inferir intenções e entidades e, em seguida, o Gerenciador de diálogo decide como proceder com base na melhor interpretação. A base de conhecimento formaliza os dados para oferecer o gerenciamento da conversa e o componente de Geração de Linguagem Natural gera respostas de texto em linguagem natural.



Fonte: Adaptação de Atmauswan e Abdullahi (2022).

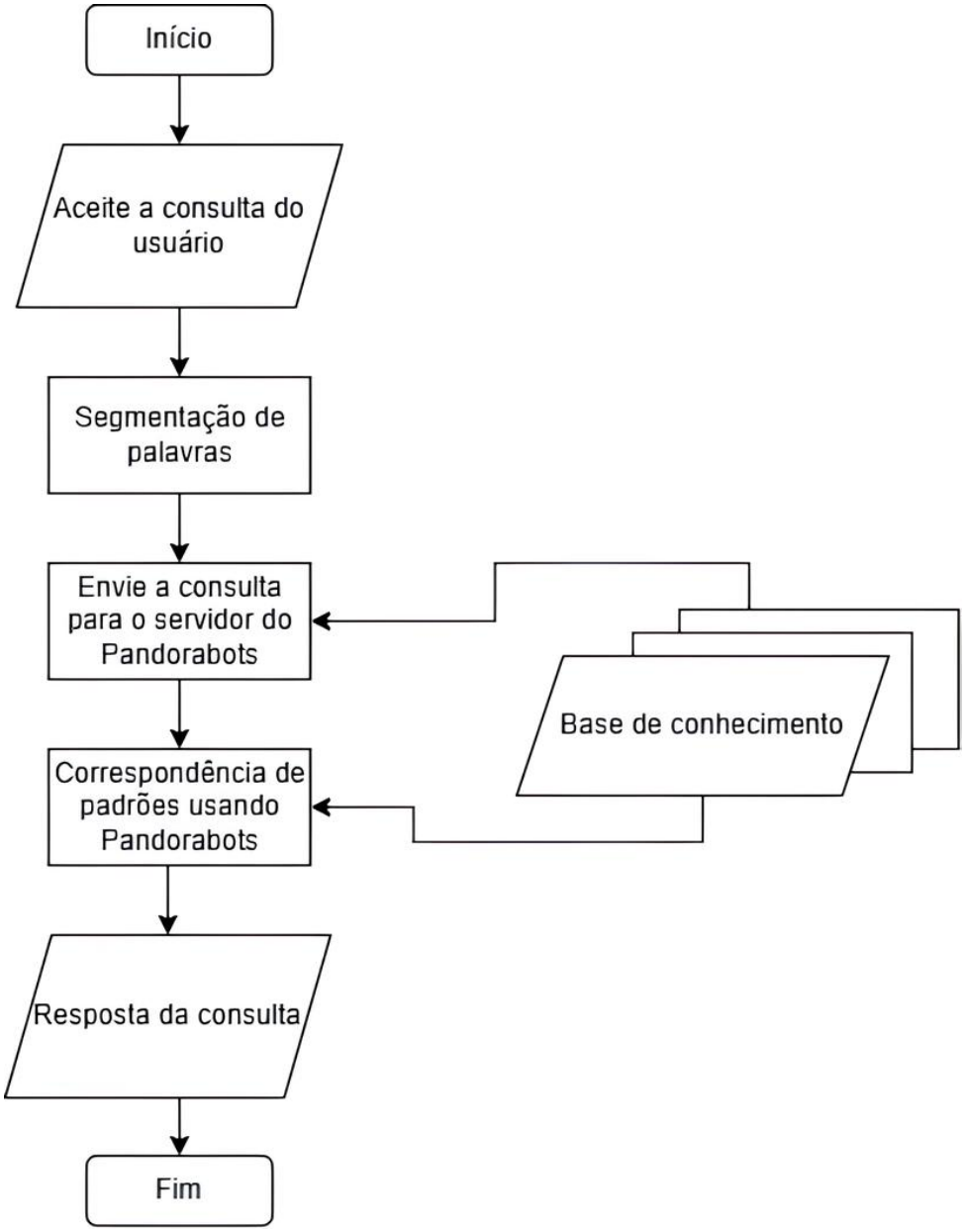
De acordo com Atmauswan e Abdullahi (2022), o uso do processamento de linguagem natural torna os *chatbots* mais fáceis de usar e confiáveis, bem como permite que os *chatbots* compreendam a consulta do usuário e respondam com mais precisão, levando a uma melhor experiência do usuário. Em contrapartida, são citadas na pesquisa as possíveis limitações devido à quantidade e qualidade de dados disponíveis. Se o *chatbot* não tiver dados suficientes para extrair dados, talvez não seja capaz de fornecer respostas precisas.

Khin e Soe (2020) utilizam a linguagem de marcação de inteligência artificial (em inglês *Artificial Intelligence Markup Language*, ou AIML) - usada para construir uma base de conhecimento para o desenvolvimento de um *chatbot* que fornece respostas eficientes e precisas para qualquer pergunta do usuário sobre informações da *University Of Computer Studies*, de Yangon, Mianmar, partindo da coleta de 970 pares de perguntas e respostas para definir o domínio de conhecimento fornecido ao *chatbot*. Deste modo, a partir dos dados coletados, os autores aplicam correspondência de padrões através de *tags* AIML para o fornecimento de respostas. O sistema proposto foi medido quanto à eficiência com três exemplos de diálogos: (i) em termos de categorias atômicas, que contém um único padrão e um único modelo, (ii) categorias padrão, que contém vários padrões e modelos, e (iii) categorias recursivas, que contém referências a outras categorias em seus modelos. O *chatbot* implementado consegue responder à maioria das perguntas do usuário corretamente, segundo os autores. Entretanto, foram encontradas respostas incompatíveis, devido a alguns padrões de entrada que não coincidem com o conhecimento do *chatbot*, indicando a necessidade de incluir mais dados para a base de conhecimento.

A Figura 2.7 apresenta um diagrama de fluxo que descreve o processo de aceitar e responder uma consulta do usuário usando a API do *Pandorabots*, uma plataforma online que permite aos desenvolvedores construir, hospedar e implantar *chatbots* com o uso de AIML. O processo começa com a aceitação da consulta do usuário, seguida pela segmentação de palavras e envio da consulta ao servidor *Pandorabots*. O servidor realiza uma correspondência de padrões usando a API e, em seguida, fornece uma resposta baseada na base de conhecimento disponível. O processo termina após fornecer a resposta à consulta.

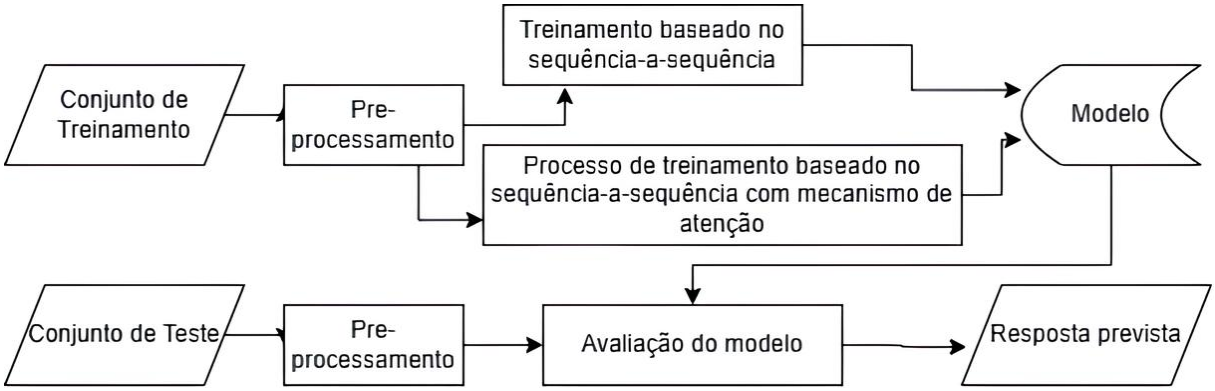
Chatbots podem ser grandes auxiliares na resolução do problema de atendimento ao cliente (CHANDRA; SUYANTO, 2019). Chandra e Suyanto (2019) utilizaram um pequeno conjunto de dados obtidos de conversas de admissão no aplicativo de mensagens instantâneas *Whatsapp* para desenvolver um *chatbot* baseado em um modelo de sequência-a-sequência para admissão na universidade usando um sistema de resposta a perguntas. O sistema consiste em duas etapas: treinamento e teste, conforme ilustrado na Figura 2.8. O conjunto de treinamento é um par de sentenças de entrada e alvo que alimentam o modelo sequência-a-sequência. As sentenças são pré-processadas por meio da transformação para letras minúsculas, remoção da pontuação e *tokenização*. Em seguida, o modelo é treinado com base no sequência-a-sequência sem e com mecanismo de atenção, usando uma taxa de aprendizado de 0.001. Isso produz um modelo treinado que é então avaliado na etapa de teste.

Figura 2.7 – Diagrama de fluxo do *chatbot* proposto por Khin e Soe (2020).



Fonte: Adaptação de Khin e Soe (2020).

Figura 2.8 – Diagrama de fluxo do *chatbot* proposto por Chandra e Suyanto (2019). O diagrama começa com conjuntos de treinamento e teste, passa pelo pré-processamento, treinamento e avaliação do modelo, e termina na resposta prevista.



Fonte: Adaptação de Chandra e Suyanto (2019).

O desempenho do modelo foi medido usando a pontuação *Bilingual Evaluation Understudy* (BLEU), uma métrica usada para avaliar a qualidade do texto gerado por máquina, comparando-o a uma ou mais traduções de referência geradas por humanos, com pontuação no intervalo de 0 a 100. Os resultados da pesquisa mostram que o modelo produz uma pontuação BLEU bastante alta de 41,04. Uma técnica de mecanismo de atenção — que permite que o modelo se concentre nas partes mais relevantes de uma frase de entrada para gerar a saída — usando frases invertidas melhora o modelo para fornecer um BLEU mais alto de até 44,68 (CHANDRA; SUYANTO, 2019).

Santoso et al. (2018) propuseram uma solução para auxiliar candidatos interessados em admissão na *Universitas Dian Nuswantoro* (UDINUS), visando fornecer informações de forma mais ágil do que aguardar uma resposta da equipe de admissão. Essa solução consistiu na criação de um *chatbot* desenvolvido *Dinus Intelligente Assistance* (DINA). Para construir o conhecimento do *chatbot*, os pesquisadores reuniram dados provenientes do livro de visitas da UDINUS, que continha perguntas e respostas relacionadas aos serviços de admissão da universidade.

A abordagem empregada para capacitar o *chatbot* envolveu a aplicação de técnicas de aprendizado de máquina, especificamente o método de *Forward Chaining* (AL-AJLAN, 2015), que possibilita a extração de padrões da base de conhecimento para oferecer respostas coerentes às perguntas dos usuários. O conhecimento do DINA foi estruturado a partir de um conjunto de Perguntas Frequentes, organizadas por categorias. A arquitetura do *chatbot* adotou uma abordagem modular, em que cada módulo era composto por uma base de conhecimento, um mecanismo de inicialização e a lógica necessária para tratar as requisições dos usuários.

Além disso, o processo de interação do *chatbot* envolvia a análise das entradas dos usuários por meio do *Dialogflow*, uma plataforma de gerenciamento de diálogos que utiliza processamento de linguagem natural para identificar as intenções subjacentes nas consultas dos usuários. Os resultados dos testes realizados pelos pesquisadores, usando dez amostras aleatórias de perguntas, indicaram que o *chatbot* DINA foi capaz de responder corretamente a oito delas.

Isso reforça a viabilidade do *chatbot* DINA como uma ferramenta útil para auxiliar os candidatos a encontrar informações relevantes sem depender exclusivamente da equipe de admissão, conforme destacado pelos autores.

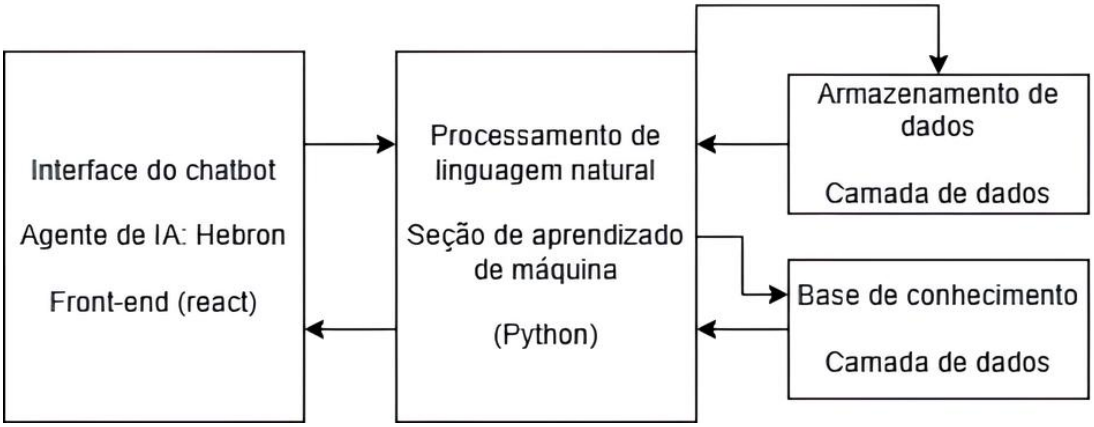
Também é possível utilizar mais de uma abordagem, como fizeram Ranoliya, Raghuwanshi e Singh (2017) ao fornecerem o design de um *chatbot* que responde eficientemente perguntas com base no conjunto de dados de perguntas frequentes usando a linguagem de marcação de inteligência artificial (em inglês *Artificial Intelligence Markup Language*, ou AIML) e a análise semântica latente (em inglês *Latent Semantic Analysis*, ou LSA). A implementação foi utilizando o *MyAdvisor*, um sistema de aconselhamento acadêmico baseado em *chatbot* projetado para emular cenários reais de aconselhamento entre orientadores e estudantes, com design do sistema guiado pela heurística de usabilidade, que é um conjunto de princípios usados para avaliar a usabilidade de um sistema. O *chatbot* proposto pode ser usado por qualquer universidade para responder perguntas frequentes a estudantes curiosos de forma interativa. Apesar de não apresentar resultados do estudo, os autores indicam que o *chatbot* pode lidar com questões gerais e baseadas em modelos usando AIML e LSA, respectivamente.

Oguntosin e Olomo (2021) descrevem o desenvolvimento de um *chatbot* de comércio eletrônico para o *Covenant University Community Mall* com dados de um servidor *MySQL* usado como banco de dados. Foram realizados testes quanto à funcionalidade e os resultados mostraram que o *chatbot* desenvolvido foi capaz de fornecer uma experiência de compra fácil, inteligente e confortável para a comunidade da universidade. O *chatbot* implementado usa processamento de linguagem natural para entender e manipular texto ou fala em linguagem natural, que utiliza fundamentos de aprendizado de máquina e aprendizado profundo (Figura 2.9). De acordo com os autores, foram utilizados os seguintes recursos na implementação do *chatbot* :

- Para o processamento de linguagem natural, o recurso usado é o *Spacy*, que é uma biblioteca e API (*Application Programming Interface*) em linguagem Python de código aberto que ajuda o *chatbot* a entender e traduzir os grandes volumes de textos que encontrará durante suas conversas com seus usuários-alvo, especialmente na estruturação gramatical de cada frase.

- *Recast.ai* é a API usada para treinar o *chatbot* com subdivisões, como as intenções do usuário, com expressões pré-programadas em cada fluxo de conversação de intenções e habilidades do *chatbot*.

Figura 2.9 – Diagrama de fluxo do *chatbot* proposto por Oguntosin e Olomo (2021). O diagrama começa com a interface do *chatbot*, o agente de IA utilizado (*hebron*) e o *front-end* implementado em *React*, que estão conectados à seção de aprendizado de máquina, com processamento de linguagem natural, implementado na linguagem *Python*, que, por sua vez, estão conectados à base de conhecimento e aos dados armazenados.



Fonte: Adaptação de Oguntosin e Olomo (2021).

Monteiro (2021) discute o desenvolvimento de um *chatbot* chamado Helena, projetado para auxiliar estudantes da área de Ciência da Computação da Universidade Federal de Ouro Preto com procedimentos universitários. O *chatbot* é capaz de fornecer respostas imediatas às dúvidas comuns dos alunos, potencialmente reduzindo a carga de trabalho do Conselho do Curso ao responder e-mails repetitivos. O desenvolvimento de Helena utilizou a plataforma *IBM Watson Assistant*, um sistema de computação cognitiva que utiliza inteligência artificial para processar e analisar grandes quantidades de dados, permitindo que ele compreenda e responda à linguagem humana de maneira semelhante à humana. Dessa forma, o *IBM Watson Assistant* foi usado para o processamento de linguagem natural manter o *chatbot* disponível e escalável para uma interação eficiente com os usuários. O *chatbot* foi treinado usando exemplos reais fornecidos pelo COCIC, garantindo que ele pudesse aprender com cenários autênticos. O código usado no desenvolvimento do *chatbot* é disponibilizado para a comunidade.

Apesar de suas vantagens, foram observados alguns pontos negativos no trabalho. Um deles foi a interface, que poderia ser melhorada para aprimorar a experiência do usuário. Além disso, foi destacada como uma limitação a dependência do *chatbot* de perguntas específicas para respostas corretas. Dependendo da pergunta, mesmo que esteja relacionada a um contexto dentro da base de conhecimento do *chatbot*, se não for formulada no formato textual esperado, o *chatbot* pode falhar em fornecer uma resposta precisa. Essa limitação evidencia a necessidade de refinamento nas capacidades de compreensão de linguagem natural do *chatbot* para melhor acomodar uma variedade mais ampla de entradas dos usuários.

A proposta do *chatbot* para a Pró-Reitoria de Graduação (PROGRAD) na Universidade Federal de Ouro Preto (UFOP) insere-se em um cenário dinâmico de desenvolvimento de *chatbots*, conforme evidenciado pela revisão da literatura. Diferentes abordagens foram exploradas nos

estudos revisados, empregando técnicas como AIML, LSA, processamento de linguagem natural, aprendizado de máquina e aprendizado profundo. A versatilidade dessas tecnologias destaca a variedade de métodos disponíveis para a implementação de *chatbots* em diversos contextos.

Ao integrar a API do *ChatGPT* e aplicar técnicas de indexação de documentos, o *chatbot* implementado neste trabalho para uso da PROGRAD na UFOP representa uma proposta de melhoria no campo, alinhando-se com a busca por eficiência e acessibilidade na resolução de dúvidas, como observado em outros projetos. A escolha estratégica de explorar avanços em processamento de linguagem natural, similar aos estudos revisados, evidencia o compromisso em aprimorar a compreensão contextual e a capacidade de resposta do *chatbot*.

3 Construção do *Chatbot*

Neste capítulo serão apresentadas as etapas para a construção de um *chatbot* com o uso da *API* da *OpenAI*. Serão apresentados em detalhes os módulos e bibliotecas específicos que foram empregados durante o desenvolvimento. Além disso, serão descritos minuciosamente os procedimentos adotados para a coleta de dados, incluindo as fontes utilizadas. A análise abordará, igualmente, o processo de indexação desses dados e sua aplicação no treinamento do *chatbot*, destacando a metodologia empregada para otimizar a eficiência do modelo.

3.1 Dados coletados

Foi conduzida uma análise minuciosa dos documentos presentes no site oficial da Pró-reitoria de Graduação (PROGRAD), um dos canais de disseminação de informações da Universidade Federal de Ouro Preto (UFOP). Dentro desse ambiente virtual, está disponível o Manual do Aluno[1], uma compilação abrangente de informações direcionadas aos discentes da instituição, abordando diversos aspectos relacionados à UFOP.

O Manual do Aluno é organizado em segmentos temáticos distintos (Figura 3.1), englobando áreas como as regulamentações internas da universidade, informações detalhadas sobre a graduação em si, orientações acerca do ensino a distância e ainda uma descrição abrangente da estrutura universitária. Essa disposição temática permite uma pesquisa facilitada e eficaz das informações necessárias.

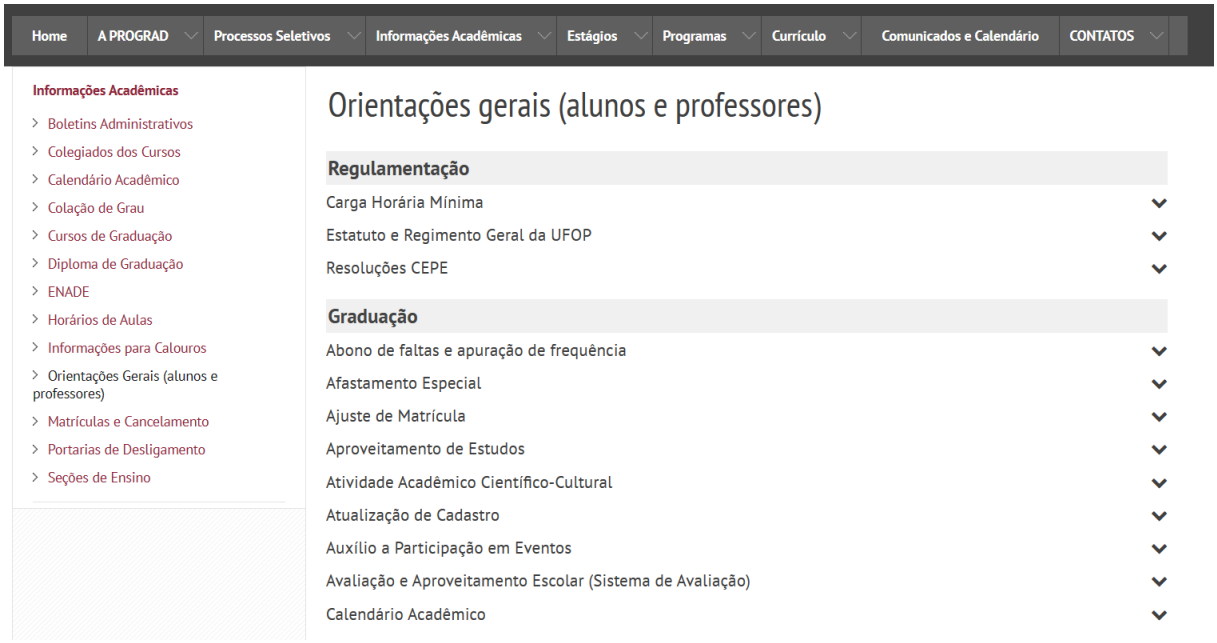
Além do Manual do Aluno, que foi utilizado para a primeira versão da implementação do *chatbot*, outros documentos relevantes foram empregados na coleta de dados para versões subsequentes. Esses documentos incluem:

- A regulamentação dos programas de assistência estudantil da PRACE[2] (Pró-reitoria de Assuntos Comunitários e Estudantis), o órgão responsável por proporcionar as condições de acesso e permanência aos estudantes, técnicos administrativos e docentes da Instituição, garantindo assim o bem-estar psicossocial de toda a comunidade acadêmica. Esse documento foi retirado do próprio site da PRACE.

- Dúvidas frequentes sobre iniciação científica, disponíveis no site da Pró-Reitoria de Pesquisa, Pós-Graduação e Inovação (PROPPI)[3] da universidade. Essas informações são

[1] Disponível em <https://www.prograd.ufop.br/orientacoes-gerais-alunos-e-professores>. Acesso em 02/12/2023.
[2] Disponível em <https://www.prace.ufop.br/assistencia-estudantil/bolsas/normas-e-regulamentos>. Acesso em 02/12/23.
[3] Disponível em <https://propp.ufop.br/pt-br/pesquisa/duvidas-frequentes>. Acesso em 02/12/2023.

Figura 3.1 – Manual do Aluno, escrito pela Pró-Reitoria de Graduação da Universidade Federal de Ouro Preto e disponibilizado no site da própria universidade. Possui informações gerais de orientações sobre a universidade organizadas em formato de tópicos.



Fonte: <https://www.prograd.ufop.br/orientacoes-gerais-alunos-e-professores>. Acesso em 07/08/2023.

essenciais para os alunos que desejam se envolver em atividades de pesquisa durante a graduação.

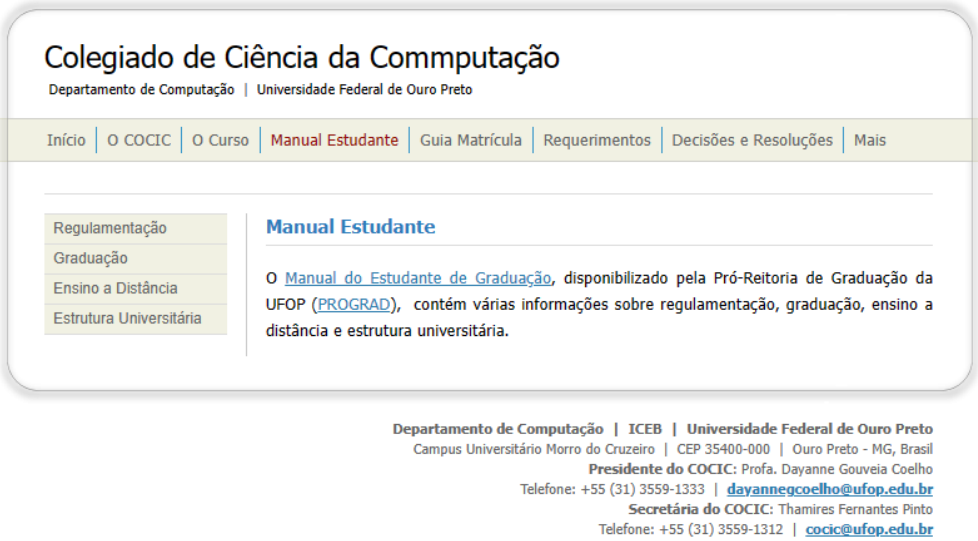
- Documento que dispõe sobre as normas de funcionamento do Programa de Monitoria da PROGRAD[4]. O Programa de Monitoria tem como objetivo contribuir para a melhoria do ensino de graduação na UFOP, promovendo a integração entre teoria e prática, bem como a cooperação acadêmica entre discentes e docentes.

- Legislação sobre estágio de estudantes[5], disponibilizada também no site da PROGRAD, com base na Lei Nº 11.788, de 25 de setembro de 2008. Essa legislação é fundamental para orientar os estudantes sobre as diretrizes e regulamentações relacionadas a estágios durante a graduação.

- O Manual do Aluno do curso de Ciência da Computação[6] (Figura 3.2), disponibilizado pelo COCIC (Colegiado de Ciência da Computação) em seu site. O COCIC é responsável pela coordenação didática das atividades que compõem o projeto pedagógico da graduação em Ciência da Computação, e seu manual fornece informações específicas para os alunos desse curso.

[4] Disponível em <https://www.prograd.ufop.br/nolink/monitoria>. Acesso em 02/12/2023.
[5] Disponível em <https://www.prograd.ufop.br/informacoes-academicas/estagio/legislacao>. Acesso em 02/12/2023.
[6] Disponível em <http://www.decom.ufop.br/cocic/manual-do-estudante/>. Acesso em 02/12/2023.

Figura 3.2 – Seção do Manual do Estudante de Graduação em Ciência da Computação da Universidade Federal de Ouro Preto.



Fonte: <http://www.decom.ufop.br/cocic/manual-do-estudante/>. Acesso em 29/12/2023.

Esses documentos, juntamente com o Manual do Aluno, contribuem para um acesso abrangente e eficiente às informações necessárias para os estudantes da UFOP, abordando diversos aspectos da vida acadêmica e regulamentações internas da universidade.

Devido à extensa abrangência e à riqueza de informações contidas nos documentos mencionados anteriormente, optou-se por considerá-los recursos ideais para a implementação do *chatbot*. Essa decisão foi baseada na intenção de oferecer aos usuários um canal de interação que forneça respostas precisas e bem fundamentadas, aproveitando o amplo conhecimento contido nos documentos listados.

3.2 Construção da aplicação

Para a implementação de um *chatbot* contextualizado, direcionado a questões sobre a Universidade Federal de Ouro Preto (UFOP), tornou-se necessário implementar uma aplicação que abarca diversas etapas. Essas etapas englobam a criação de um índice de documentos, realizando uma indexação criteriosa desses documentos para viabilizar consultas eficientes, além da interação com os usuários, permitindo-lhes realizar perguntas ao modelo treinado utilizando a *API* do *ChatGPT*.

A indexação dos documentos é realizada pelo *LlamaIndex*, que organiza e categoriza os dados de acordo com critérios específicos. Por exemplo, ele pode agrupar informações semelhantes juntas ou separar dados com base em certos atributos. O *LlamaIndex* também identifica pontos-chave nos documentos, chamados de nós relevantes, que correspondem à pergunta do cliente. Isso envolve um processo de análise e comparação detalhada para encontrar correspondências

entre a pergunta do cliente e os conteúdos indexados nos documentos.

Para a implementação do *chatbot*, foi utilizada a linguagem *Typescript*[7] com o *framework LLamaIndex.TS*[8], e a *API* do *ChatGPT*. A Figura 3.3 ilustra a arquitetura do *chatbot* implementado.

O processo começa quando um cliente submete uma pergunta. Essa pergunta é então direcionada ao *LlamaIndex*, que atua como um mecanismo de busca e comparação sofisticado. Ele compara a pergunta do cliente com um conjunto de documentos que foram previamente organizados e indexados. Esses documentos, que estão inicialmente em formato TXT, são transformados em formato *JSON*, que facilita a busca e a recuperação de informações.

A busca realizada pelo *LlamaIndex* é meticulosa e direcionada. O objetivo é identificar pontos-chave nos documentos, chamados de nós relevantes, que correspondem à pergunta do cliente. Isso envolve um processo de análise e comparação detalhada para encontrar correspondências entre a pergunta do cliente e os conteúdos indexados nos documentos.

Uma vez que os nós relevantes são identificados, o *LlamaIndex* formula uma consulta específica para cada nó. Essas consultas são projetadas para vasculhar os documentos e extrair informações que são diretamente relevantes para a pergunta original do cliente.

Depois de reunir essas informações, o *LlamaIndex* cria um *prompt*. Este *prompt* é uma espécie de direcionamento ou instrução que é transmitida para o *chat engine*, uma ferramenta do próprio *LLamaIndex*. O *chat engine* é responsável por interagir com um modelo avançado de linguagem, o ChatGPT.

O *chat engine* envia o *prompt* ao ChatGPT. O ChatGPT é um modelo de linguagem altamente avançado que é capaz de compreender contextos complexos e gerar respostas que são coerentes, naturais e detalhadas. Com base no *prompt* recebido, o ChatGPT formula uma resposta que é diretamente relevante e útil para a pergunta inicial do cliente. Essa resposta, uma vez gerada pelo ChatGPT, é então comunicada de volta ao *chat engine* e transformada em um fluxo legível de texto.

Por fim, a aplicação envia resposta ao cliente que originou a pergunta. Todo esse processo, desde a submissão da pergunta pelo cliente até a entrega da resposta, ocorre de maneira automatizada e eficiente.

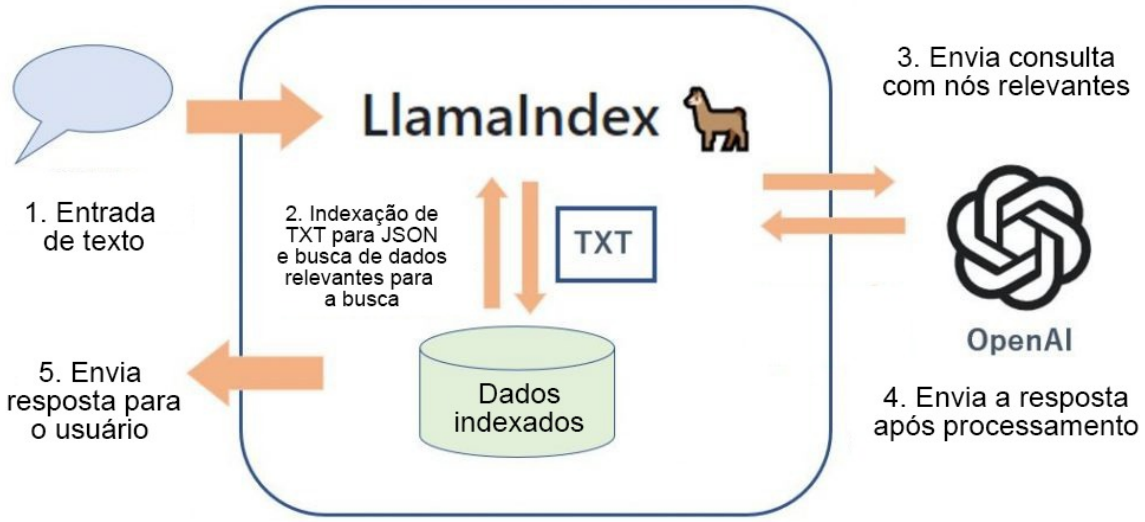
Os clientes são capazes de obter respostas precisas e contextualizadas para suas perguntas. Essas respostas são baseadas não apenas no conteúdo dos dados coletados, mas também nas capacidades de processamento de linguagem do ChatGPT. O ChatGPT é capaz de entender o contexto da pergunta do cliente e gerar uma resposta que é não apenas relevante, mas também detalhada e informativa.

Por exemplo, se um cliente perguntar “Quais são os cursos oferecidos pela UFOP?”,

[7] Disponível em <https://www.typescriptlang.org/>. Acesso em 12/01/2024.
[8] Disponível em <https://ts.llamaindex.ai/>. Acesso em 12/01/2024.

o *chatbot* poderá responder com uma lista detalhada de cursos, juntamente com informações adicionais sobre cada curso, como a duração do curso, o número de créditos necessários para a graduação, as oportunidades de bolsas de estudo disponíveis.

Figura 3.3 – Arquitetura da aplicação, um *chatbot* para o contexto da UFOP. O *Llamaindex* recebe as questões inseridas pelo usuário, recolhe dados relevantes para a consulta presentes no documento indexado, passa para o *chat engine* os documentos recuperados, que faz a consulta à *API* do *ChatGPT*, retornando uma resposta ao usuário.



Fonte: Adaptado de <https://recruit.gmo.jp/engineer/jisedai/blog/llamaindex-chatgpt-tuning/>. Acesso em 15/01/2024.

3.2.1 Linguagens e Bibliotecas

A implementação da aplicação envolveu a utilização do ambiente de execução de código *NodeJS*[9], uma plataforma que possibilita a execução autônoma de aplicações desenvolvidas em *JavaScript*, independentemente de um navegador (NODEJS, 2024). Este ambiente foi usado para a criação de um servidor *web* local, fornecendo a infraestrutura necessária para hospedar a aplicação e permitir interações através do protocolo HTTP.

Além disso, a escolha do *Express*[10], um *framework* que opera sobre o ambiente *NodeJS* em tempo de execução, desempenhou um papel crucial na implementação. O *Express* oferece soluções eficazes para o gerenciamento de rotas, requisições e respostas, simplificando a arquitetura do servidor e a manipulação de diferentes verbos HTTP (EXPRESSJS, 2024).

O principal dos componentes utilizados na implementação do *chatbot* é o *LLamaIndex.TS* (LIU, 2022), um *framework* desenvolvido em *Typescript*, que fornece uma interface central para conectar seus modelos de linguagem de grande escala (em inglês, *Large language model* ou LLM) com dados externos. Essa biblioteca funciona realizando o carregamento dos dados coletados,

[9] Disponível em <https://nodejs.org/en>. Acesso em 17/01/2024.
[10] Disponível em <https://expressjs.com/pt-br/>. Acesso em 17/01/2024.

seguido da criação de um índice a partir dos documentos fornecidos. Os dados são estruturados em representações intermediárias que são fáceis e eficientes para os LLMs consumirem (LIU, 2022).

O *LLamaIndex* usa sistemas de Geração Aumentada de Recuperação que combinam modelos de linguagem grandes com uma base de conhecimento privada. Geralmente consiste em duas etapas: a etapa de indexação e a etapa de consulta. O índice é construído usando classes e objetos fornecidos pela biblioteca (AWAN, 2023).

Para utilizar um grande contexto para uma consulta, *LLamaIndex* divide o documento de entrada em pedaços menores, chamados de índices. Depois disso, para cada índice, uma consulta é emitida para a LLM sendo utilizada. Para determinar quais índices são mais relevantes para a consulta, é gerado o *embedding* do *prompt* — a representação numérica do texto de entrada (*prompt*) em um espaço vetorial contínuo — e a similaridade de cosseno entre esse *embedding* e os *embeddings* de cada índice é calculada. Isso permite refinar a resposta com base no novo contexto, levando em consideração todos os índices relevantes para gerar a resposta final (ZIRNSTEIN, 2023).

A integração da biblioteca *llamaindex* no projeto proporciona as funcionalidades necessárias para carregar e indexar dados, identificar partes relevantes dos documentos para a tarefa de extração e executar consultas ao *ChatGPT*.

Para a implementação da interface de usuário do *chatbot*, optou-se pela utilização do *Next.js*[11], um renomado *framework* concebido para o *React*[12], uma biblioteca JavaScript desenvolvida e mantida pelo conglomerado *Meta*. Notoriamente reconhecido por sua robustez e flexibilidade, o *Next.js* proporciona uma estrutura eficiente para o desenvolvimento de aplicações web, especialmente aquelas que demandam interatividade em tempo real e uma experiência de usuário dinâmica (NEXTJS, 2024).

A escolha do *Next.js* como base para a implementação da interface do *chatbot* fundamentou-se na sua capacidade de simplificar a criação de páginas web de forma escalável e otimizada.

3.2.2 Leitura e indexação dos Dados Coletados

A indexação dos dados é uma etapa crucial que envolve a organização e estruturação dos documentos carregados, visando possibilitar consultas eficientes. Essa organização é de importância fundamental para o *chatbot* conseguir recuperar informações pertinentes e responder de forma precisa às perguntas dos usuários.

No contexto do projeto implementado, a tarefa de indexação é realizada pelo *LLamaIndex*. Nesse processo, o foco está na indexação dos dados coletados. Para efetuar a indexação, o projeto

[11] Disponível em <https://nextjs.org/>. Acesso em 16/01/2024.
[12] Disponível em <https://pt-br.legacy.reactjs.org/>. Acesso em 16/01/2024.

adota uma abordagem que começa com a leitura da base de conhecimento, utilizando a classe *SimpleDirectoryReader*.

Vale ressaltar que os documentos coletados (Figura 3.5) são previamente convertidos para o formato de arquivo de texto (*txt*). Após essa etapa, o *LLamaIndex* executa o processo de indexação, através da classe *VectorStoreIndex*, o que resulta na reestruturação do conteúdo do manual para um formato mais organizado e consultável. Nesse caso, o formato escolhido para armazenar as informações indexadas é o formato *JSON* (Figura 3.4), que é altamente flexível e eficaz para representar dados estruturados.

Figura 3.4 – Manual do Aluno da UFOP indexado após chamada da classe *VectorStoreIndex*.

```
{
   "docstore/data":{
      "4339035b-8884-4e9e-8d55-2130874be316":{
         "indexId":"4339035b-8884-4e9e-8d55-2130874be316",
         "nodesDict":{
            "fe17553d-4e5c-4e32-a416-125c4f3fd61a":{
               "id_":"fe17553d-4e5c-4e32-a416-125c4f3fd61a",
               "metadata":{

               },
               "excludedEmbedMetadataKeys":[

               ],
               "excludedLlmMetadataKeys":[

               ],
               "relationships":{
                  "SOURCE":{
                     "nodeId":"./data/manual_do_calouro_revisado_08_de_fevereiro_2019.txt",
                     "metadata":{

                     },
                     "hash":"Bkw76uZTAcHUW7wkqioodsg3WDfQ/bq4XxzClF5Aa9c="
                  },
                  "NEXT":{
                     "nodeId":"a3432055-982e-4a1b-a7a3-64e03eeb435d",
                     "metadata":{
```


Fonte: Elaborado pelo autor.

Assim, ao concluir a indexação, os documentos passam de um formato de arquivo de texto simples para um formato *JSON*, no qual as informações estão organizadas de maneira hierárquica e interligadas.

Durante o processo de indexação, a classe *ServiceContext* desempenha um papel fundamental ao ser responsável por diversas configurações e recursos essenciais ao longo desse procedimento. Dentre esses elementos, destacam-se o modelo de linguagem empregado, as dimensões da janela de contexto, e o número máximo de saídas, entre outros. Essa classe proporciona uma solução eficiente para a administração desses recursos e configurações, simplificando a forma de ajuste do comportamento do pipeline (SERVICECONTEXT, 2024).

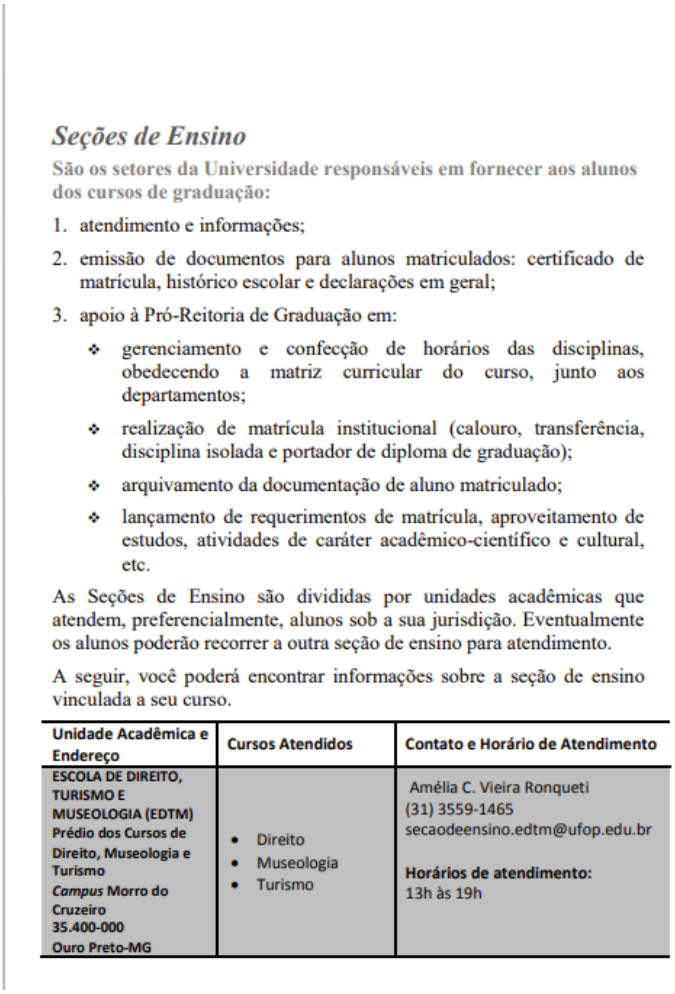
No âmbito deste trabalho, foi estabelecido que o tamanho padrão para um nó, representado por um fragmento de texto, é de 512 unidades, enquanto a sobreposição entre esses nós, ou seja,

os fragmentos de texto, é configurada como 20 unidades.

Além da *ServiceContext*, outro componente de relevância no processo de indexação é a classe *StorageContext*. Esta classe confere uma abstração fundamental para a gestão de armazenamento de nós, índices e vetores, sendo fundamental no processo de indexação por definir onde serão salvos os documentos indexados (STORAGECONTEXT, 2024).

Tanto o *ServiceContext* quanto o *StorageContext* são transmitidos como parâmetros para a inicialização do processo de indexação através da classe *VectorStoreIndex*. Ela processa o documento carregado, gera representações vetorizadas e organiza essas representações em uma estrutura que permite buscas rápidas. Essa etapa de indexação cria um mapeamento entre as consultas feitas pelo usuário e as partes relevantes dos documentos, permitindo que o *chatbot* encontre informações correspondentes de maneira eficiente.

Figura 3.5 – Uma página do Manual do Aluno, um dos documentos a ser indexado, disponibilizado no site da UFOP pela PROGRAD. Essa página em específico apresenta informações sobre as Seções de Ensino da universidade.



Seções de Ensino

São os setores da Universidade responsáveis em fornecer aos alunos dos cursos de graduação:

1. atendimento e informações;
2. emissão de documentos para alunos matriculados: certificado de matrícula, histórico escolar e declarações em geral;
3. apoio à Pró-Reitoria de Graduação em:
   - gerenciamento e confecção de horários das disciplinas, obedecendo a matriz curricular do curso, junto aos departamentos;
   - realização de matrícula institucional (calouro, transferência, disciplina isolada e portador de diploma de graduação);
   - arquivamento da documentação de aluno matriculado;
   - lançamento de requerimentos de matrícula, aproveitamento de estudos, atividades de caráter acadêmico-científico e cultural, etc.

As Seções de Ensino são divididas por unidades acadêmicas que atendem, preferencialmente, alunos sob a sua jurisdição. Eventualmente os alunos poderão recorrer a outra seção de ensino para atendimento.

A seguir, você poderá encontrar informações sobre a seção de ensino vinculada a seu curso.

Unidade Acadêmica e Endereço	Cursos Atendidos	Contato e Horário de Atendimento
ESCOLA DE DIREITO, TURISMO E MUSEOLOGIA (EDTM) **Prédio dos Cursos de Direito, Museologia e Turismo** ***Campus* Morro do Cruzeiro** **35.400-000** **Ouro Preto-MG**	• Direito • Museologia • Turismo	Amélia C. Vieira Ronqueti (31) 3559-1465 secaodeensino.edtm@ufop.edu.br **Horários de atendimento:** 13h às 19h

Fonte: <https://www.prograd.ufop.br/sites/default/files/manual_do_calouro_revisado_08_de_fevereiro_2019.pdf>. Acesso em 07/08/2023.

A integração com a API da OpenAI requer a utilização da variável de ambiente denominada **OPENAI_API_KEY**[13]. Essa variável é de importância crucial, visto que desempenha um papel essencial na autenticação e autorização das requisições feitas à API, podendo ser obtida no site da OpenAI.

3.2.3 Controlador do *chatbot*

Na camada de controle do *chatbot*, ao receber uma requisição HTTP, o processo inicial consiste na verificação da requisição recebida para determinar se esta inclui uma mensagem no corpo da solicitação ou se está vazia. Em seguida, é estabelecida a constante LLM, a qual estabelece a conexão com a OpenAI, utilizando o modelo GPT-4, escolhido por ser o modelo mais recente disponível.

Prosseguindo, é instanciado o *chat engine*, que inicia invocando a função responsável pela indexação dos dados coletados. Após esse procedimento, o *chat engine* recupera o texto relevante do índice, utilizando a mensagem fornecida pelo usuário. Este texto recuperado é então definido como contexto no *prompt* do sistema, o que viabiliza a geração de uma resposta à mensagem do usuário. Posteriormente, a resposta é formatada para um texto legível e encaminhada de volta ao solicitante da requisição ao *chatbot*.

3.2.4 Interação com usuário e geração de respostas

Foi desenvolvida uma interface interativa utilizando *Next.js*, que proporciona uma experiência de conversação contínua com o *chatbot*. A interface assemelha-se àquelas de *chatbots* renomados, como o ChatGPT. Nela, os usuários são convidados a formular suas perguntas em um campo dedicado, enquanto as perguntas enviadas e as respostas recebidas são exibidas em um quadro, criando um histórico interativo das interações, oferecendo aos usuários a possibilidade de revisitar o histórico de suas interações.

Esse histórico oferece aos usuários a capacidade de revisitar suas interações passadas. Essa funcionalidade contribui para criar uma experiência de conversação mais completa e personalizada.

Ao enviar uma pergunta, uma requisição é automaticamente acionada para o *chatbot* através da rota '*/api/chat*'. Essa rota, por sua vez, utiliza o *VectorStoreIndex* para buscar dados relevantes com base na pergunta formulada. A comparação é realizada através das representações vetoriais dos documentos presentes no índice.

Após a consulta ao índice, o *chatbot* utiliza o *ChatGPT* para gerar uma resposta contextualizada. O *ChatGPT* gera uma resposta que incorpora as informações contidas nos documentos indexados, oferecendo uma resposta à pergunta do usuário. A Figura 3.6 mostra a interface de

[13] Disponível em <https://platform.openai.com/account/api-keys>. Acesso em 01/12/2023.

usuário implementada, bem como o resultado para a pergunta “O que é a UFOP?” , uma pergunta simples relacionada à informações presentes nos documentos indexados.

É importante ressaltar que a interface de usuário do *chatbot* oferece aos usuários uma interface de interação amigável. Essa interface é projetada de forma a tornar a experiência do usuário mais intuitiva e eficiente. Além disso, a interface é estilizada de acordo com as diretrizes visuais e a identidade da UFOP, garantindo uma representação coesa da instituição.

Quando os usuários acessam o *chatbot*, eles são recebidos com instruções claras sobre como fazer perguntas ou solicitar informações. A interface permite que os usuários digitem suas perguntas em um campo de texto e, em seguida, enviem-nas para o *chatbot*. Após o envio da pergunta, os usuários recebem respostas instantâneas e contextualizadas, geradas pelo *ChatGPT* com base nas informações contidas nos documentos coletados.

Figura 3.6 – *Chatbot* implementado pelo próprio autor.

Fonte: <https://github.com/lucasbarbosa1/ufop-chatbot>. Acesso em 17/01/2024.

Para avaliar a eficácia da abordagem adotada, planeja-se realizar testes para avaliar a capacidade do *chatbot* em lidar tanto com perguntas simples quanto com as mais complexas. Serão consideradas respostas corretas aquelas que apresentem informações precisas e contextualmente relevantes, alinhadas com o conteúdo dos documentos fornecidos. Por outro lado, respostas incorretas serão aquelas que contenham informações imprecisas, irrelevantes ou ausentes em relação à pergunta formulada. A integralidade da implementação do presente projeto encontra-se disponível no seguinte endereço eletrônico: <https://github.com/lucasbarbosa1/ufop-chatbot>.

As perguntas a serem direcionadas ao *chatbot* serão as mesmas utilizadas para testar o HELENA (MONTEIRO, 2021). Dessa forma, a avaliação da implementação será conduzida por meio da comparação dos resultados obtidos com os do HELENA, considerando uma variedade de 12 intenções diferentes. Essas intenções abrangem diversos domínios dos dados coletados e exploram distintos níveis de complexidade, proporcionando uma avaliação abrangente da capacidade do *chatbot* em fornecer respostas precisas e pertinentes.

Para garantir uma avaliação abrangente, foram formuladas 20 perguntas para cada uma das 12 intenções distintas, totalizando 240 perguntas no conjunto de testes. Após a interação com o *chatbot*, as respostas foram cuidadosamente analisadas e classificadas como corretas ou incorretas, de acordo com os dados presentes nos documentos coletados. Em seguida, a acurácia foi calculada para cada intenção, dividindo o número total de perguntas corretamente respondidas pelo número total de perguntas feitas. Este método de avaliação proporcionou uma visão detalhada da eficácia do *chatbot* em lidar com diferentes níveis de complexidade e variados domínios de conhecimento.

4 Resultados

O presente capítulo aborda os experimentos conduzidos para avaliar a eficácia e o desempenho do *chatbot* contextualizado desenvolvido no contexto deste projeto, bem como seus resultados.

Com o objetivo de mensurar a precisão do *chatbot*, foram conduzidos testes de consultas, abrangendo 12 diferentes intenções, conforme detalhado na Tabela 4.1. Posteriormente, as respostas geradas foram comparadas com as fornecidas pelo *chatbot* HELENA[1]. Por meio desses experimentos, almeja-se proporcionar uma compreensão abrangente do comportamento do *chatbot* diante de variados tipos de perguntas, bem como avaliar sua adaptabilidade e precisão em comparação com o HELENA, especialmente ao lidar com variações de contexto.

Tabela 4.1 – Tabela de intenções.

Intenção
Abater Horas de ATV
Aproveitamento de Disciplinas
Calendário Acadêmico
Cancelamento de Matrícula
Carteira Estudantil
Certificado de Monitoria e Tutoria
Colação de Grau
Desligamento da Universidade
Estágio
Exame Especial
Horas Válidas de ATV
Trancamento de Disciplina

Fonte: Adaptado de Monteiro (2021).

4.1 Acurácia do *chatbot*

Para realizar os testes de acurácia, foram submetidas ao *chatbot* 20 perguntas para cada intenção, conforme ilustrado na Figura 4.1. Essas mesmas 20 perguntas foram utilizadas no teste de acurácia do HELENA, permitindo uma comparação direta. Após receber as respostas do *chatbot*, verificou-se se o resultado esperado foi adequadamente retornado.

[1] Disponível em <https://helena-assistant.github.io/>. Acesso em: 02/01/2024

Figura 4.1 – Perguntas enviadas para o *chatbot* da intenção ‘Abater Horas de ATV’.

```
"Como abater horas ATV?",
"Como aplico as horas de ATV?",
"Como cortar minhas horas de ATV?",
"Dar baixa nas horas de ATV.",
"Está sendo possível solicitar o lançamento de ATVs?",
"Estou com umas ATV's para lançar, referente ao meu estágio. Está sendo possível solicitar esse lançamento?",
"Gostaria de saber como está funcionando para cortar minhas horas de ATV durante esse período online?",
"Onde preciso entregar os documentos de ATV?",
"Entregar documentos de ATV",
"horas de atu",
"hrs de atu",
"hrs de atv",
"Como abter hrs de avt",
"quer saber sobre hrs de avt",
"quero saber sobre hrs de atv",
"onde entregar docs de atv",
"sobre horas de atv",
"dar baixa nas hrs de atv",
"como lancar minhas hrs de atv",
"quero sber mais sobre atv",
```

Fonte: retirado de <https://github.com/helena-assistant/gym/blob/master/data/data-mixed.js>. Acesso em 17/01/2024.

Os resultados obtidos são apresentados na Tabela 4.2, que representa a acurácia do *chatbot* implementado para cada intenção. Esse processo de avaliação permitiu uma análise comparativa dos desempenhos.

Tabela 4.2 – Acurácia de cada intenção.

Intenção	Acurácia
Abater Horas de ATV	70%
Aproveitamento de Disciplinas	80%
Calendário Acadêmico	80%
Cancelamento de Matrícula	90%
Carteira Estudantil	100%
Certificado de Monitoria e Tutoria	95%
Colação de Grau	95%
Desligamento da Universidade	100%
Estágio	100%
Exame Especial	95%
Horas Válidas de ATV	85%
Trancamento de Disciplina	100%

Fonte: Criado pelo autor.

Ao analisar a Tabela 4.2, destaca-se que quatro intenções alcançaram uma acurácia perfeita de 100% nas respostas, sendo elas ‘Carteira Estudantil’, ‘Desligamento da Universidade’, ‘Estágio’ e ‘Trancamento de Disciplina’. Este desempenho notável pode ser atribuído, provavelmente, à homogeneidade das perguntas, facilitando a adaptação do *chatbot* ao contexto dessas consultas específicas, bem como ao fato de muitas das questões serem básicas, cujas respostas são prontamente identificáveis no Manual do Aluno ou em algum outro documento coletado. No entanto, as intenções ‘Abater Horas de ATV’, ‘Aproveitamento de Disciplinas’ e ‘Calendário

Acadêmico' apresentaram menor acurácia, registrando respectivamente 70%, 80% e 80% de acertos.

A redução na precisão dessas intenções pode ser atribuída à diversidade de questões abordadas em torno desses tópicos, que carecem de um contexto uniforme entre as perguntas. Além disso, algumas perguntas não dispunham de informações claras nos documentos indexados, levando o *chatbot* a fornecer respostas genéricas, muitas vezes provenientes do ChatGPT, nos erros cometidos. Em alguns casos, o *chatbot* apresentou respostas confusas, indicando uma possível dificuldade em compreender totalmente as perguntas, mesmo quando as informações necessárias estavam disponíveis nos documentos indexados.

No que diz respeito às intenções 'Certificado de Monitoria e Tutoria', 'Colação de Grau', 'Exame Especial', Cancelamento de Matrícula' e 'Horas Válidas de ATV', elas demonstraram resultados satisfatórios, alcançando respectivamente 95%, 95%, 95%, 90% e 85% de acertos. Esses resultados apontam para a capacidade do *chatbot* em lidar efetivamente com uma gama diversificada de consultas.

Na próxima seção, que aborda a comparação dos resultados dos testes realizados com os resultados dos testes de acurácia do *chatbot* HELENA, será possível contextualizar melhor o desempenho do *chatbot* implementado no cenário de sistemas similares.

4.2 Comparação com testes de acurácia do HELENA

Monteiro (2021) conduziu testes de acurácia no *chatbot* implementado, abordando intenções semelhantes às apresentadas neste trabalho. No entanto, divergindo da abordagem adotada aqui, os testes do HELENA classificaram as respostas em categorias distintas, nomeadamente "Respondidas", "Não respondidas" e "Incorretas". Ao contrário, neste trabalho, as respostas foram classificadas apenas como "Corretas" ou "Incorretas". Para fins de comparação entre os resultados do *chatbot* implementado neste projeto e os do HELENA, optou-se por considerar que as respostas classificadas como "Não Respondidas" e "Incorretas" seriam tratadas de forma equivalente, ambas sendo consideradas "Incorretas".

O HELENA utilizou uma tabela de intenções que foi subdividida em códigos, atribuindo uma identificação única a cada uma das intenções, conforme ilustrado na Tabela 4.3. Essa abordagem de categorização facilitou a organização e análise dos resultados obtidos.

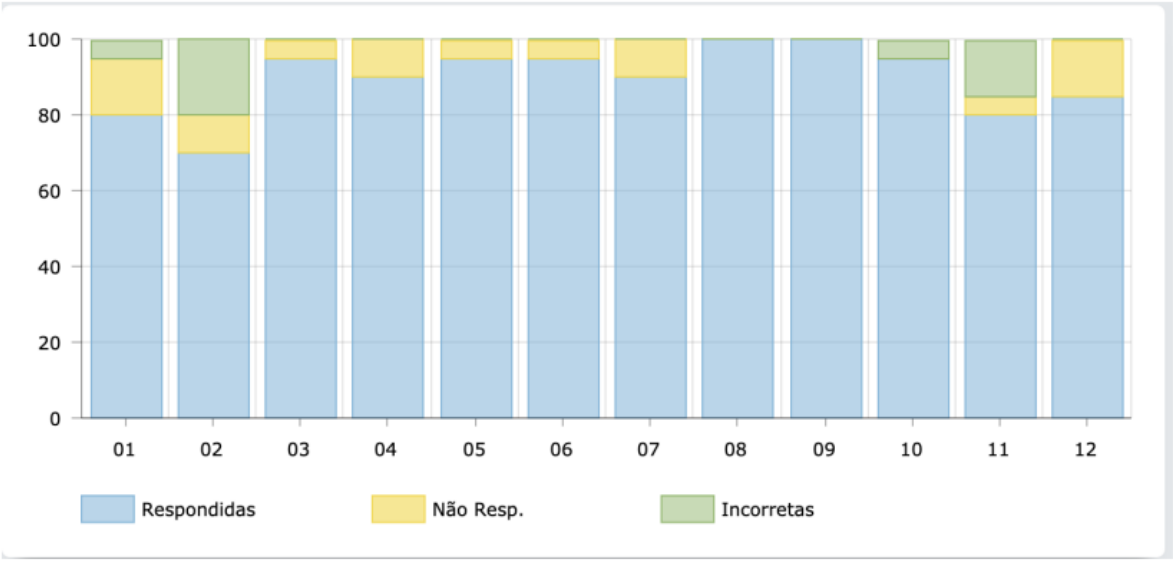
Tabela 4.3 – Mapeamento de intenções para teste de acurácia do *chatbot* HELENA.

Intenção	Código
Abater Horas de ATV	01
Aproveitamento de Disciplinas	02
Calendário Acadêmico	03
Cancelamento de Matrícula	04
Carteira Estudantil	05
Certificado de Monitoria e Tutoria	06
Colação de Grau	07
Desligamento da Universidade	08
Estágio	09
Exame Especial	10
Horas Válidas de ATV	11
Trancamento de Disciplina	12

Fonte: Monteiro (2021).

Os resultados desses testes, conduzidos pelo HELENA no primeiro experimento de acurácia, conforme descrito em (MONTEIRO, 2021), são apresentados de maneira detalhada na Figura 4.2. A análise comparativa será embasada nos resultados detalhados para cada intenção, possibilitando a identificação de semelhanças, divergências e tendências distintas. Ao contextualizar esses resultados, teremos uma compreensão mais completa do desempenho relativo de cada *chatbot*.

Figura 4.2 – Resultados obtidos no primeiro teste de acurácia do *chatbot* HELENA.



Fonte: Monteiro (2021).

Ao examinar os resultados obtidos pelo HELENA nos testes de acurácia, presentes na Figura 4.2 observa-se variações nas taxas de sucesso para diferentes intenções. Destaca-se que ‘Desligamento da Universidade’ e ‘Estágio’ alcançaram uma acurácia perfeita de 100%,

demonstrando uma capacidade consistente de resposta para essas categorias específicas. Por outro lado, 'Abater Horas de ATV' e 'Aproveitamento de Disciplinas' registraram taxas de acerto de 80% e 70%, respectivamente, indicando desafios na precisão dessas intenções no HELENA.

Comparando esses resultados com o *chatbot* implementado neste projeto (Tabela 4.4), pode-se identificar semelhanças e diferenças notáveis. Ambos os sistemas apresentaram acurácia perfeita de 100% para 'Desligamento da Universidade' e 'Estágio', indicando uma consistência notável na capacidade de resposta para essas consultas específicas. Entretanto, divergências surgiram em 'Abater Horas de ATV' e 'Aproveitamento de Disciplinas'. Enquanto o HELENA obteve 80% e 70% de acertos, respectivamente, o *chatbot* implementado alcançou 70% e 80% nessas mesmas intenções.

Tabela 4.4 – Comparação de acurácia nas respostas entre o *chatbot* implementado e o HELENA. Melhores resultados estão destacados em negrito.

Intenção	Acurácia *chatbot* implementado (%)	Acurácia HELENA (%)
Abater Horas de ATV	70	**80**
Aproveitamento de Disciplinas	**80**	70
Calendário Acadêmico	80	**95**
Cancelamento de Matrícula	**90**	**90**
Carteira Estudantil	**100**	95
Certificado de Monitoria e Tutoria	**95**	90
Colação de Grau	**95**	**95**
Desligamento da Universidade	**100**	**100**
Estágio	**100**	**100**
Exame Especial	**95**	**95**
Horas Válidas de ATV	**85**	80
Trancamento de Disciplina	**95**	85

Fonte: Criado pelo autor.

Além disso, ao analisar as outras intenções do teste, observa-se que 'Calendário Acadêmico' apresentou 80% de acerto no *chatbot* implementado e 95% no HELENA, indicando uma leve discrepância na precisão dessas respostas. 'Cancelamento de Matrícula' obteve 90% de acertos em ambos os sistemas, revelando uma consistência notável. 'Carteira Estudantil' registrou 100% de acertos no *chatbot* implementado e 95% no HELENA. 'Certificado de Monitoria e Tutoria' alcançou 95% de acerto no *chatbot* implementado e 90% no HELENA. 'Colação de Grau' obteve 95% de acertos em ambos os sistemas, demonstrando uma eficácia consistente. 'Exame Especial' registrou 95% de acertos tanto no *chatbot* implementado quanto no HELENA. 'Horas Válidas de ATV' apresentou 85% de acertos no *chatbot* implementado e 80% no HELENA. 'Trancamento de Disciplina' alcançou 95% de acertos no *chatbot* implementado e 85% no HELENA.

Essas divergências evidenciam nuances na abordagem e no desempenho específico de cada *chatbot* em relação a esses tópicos, estando diretamente associadas à quantidade de dados

utilizados para treinamento por cada *chatbot*, os quais continham informações sobre esses temas. O impacto da disponibilidade e qualidade dos dados de treinamento se reflete nas capacidades de compreensão e resposta de cada sistema, influenciando diretamente a acurácia observada em intenções específicas. Portanto, a análise dessas discrepâncias não apenas ressalta as variações de desempenho, mas também destaca a importância da qualidade e quantidade dos dados de treinamento na eficácia de *chatbots* em lidar com consultas variadas.

Além das análises específicas por intenção, é relevante considerar a média global de acurácia alcançada pelos dois sistemas. Em média, o *chatbot* implementado obteve uma acurácia de aproximadamente 90,4%, enquanto o HELENA registrou uma média de aproximadamente 89,5%. Esses resultados indicam que, de maneira geral, o *chatbot* implementado neste projeto apresentou uma eficácia ligeiramente superior na precisão das respostas em comparação com o HELENA.

Uma consideração adicional importante é que o *chatbot* implementado baseia-se no modelo de linguagem avançado do ChatGPT, que possui a capacidade de entender e gerar texto de maneira contextual e fluida. Essa abordagem permite ao *chatbot* manter conversas mais dinâmicas e lidar com perguntas fora do escopo predefinido, proporcionando uma experiência mais flexível e adaptável para o usuário. Por outro lado, o HELENA, sendo baseado em regras específicas, pode ter limitações ao lidar com consultas que não estejam estritamente dentro das categorias previamente estabelecidas.

5 Considerações Finais

5.1 Conclusão

Com base nos resultados consistentes dos testes, nos quais o *chatbot* implementado alcançou uma acurácia superior a 70% em todas as intenções, é viável afirmar que o sistema demonstra ser uma opção eficaz e promissora para atender às demandas da comunidade acadêmica da UFOP. Acredita-se que a acurácia global acima desse limiar é um indicativo positivo da capacidade do *chatbot* em fornecer respostas precisas em uma variedade de consultas.

A comparação entre os resultados do *chatbot* desenvolvido neste projeto e os do HELENA proporciona informações importantes sobre o desempenho desses sistemas diante de diversas intenções. Essa análise mais profunda destaca áreas de excelência e oportunidades de aprimoramento em ambos os *chatbots*. Notadamente, a consistência nas taxas de acerto em categorias como ‘ Desligamento da Universidade’ e ‘Estágio’ sugere que o *chatbot* implementado é capaz de oferecer respostas confiáveis em situações críticas.

É importante ressaltar que a queda de acurácia observada em algumas intenções esteve diretamente relacionada à qualidade dos dados de treinamento. Portanto, para otimizar ainda mais a eficácia do *chatbot*, é preciso um esforço contínuo na coleta de documentos mais detalhados e específicos sobre os diferentes segmentos da UFOP. Esse aprimoramento na base de dados permitirá ao *chatbot* compreender e responder com maior precisão às consultas da comunidade acadêmica, consolidando sua utilidade e confiabilidade.

5.2 Trabalhos Futuros

Considerando os resultados promissores e as áreas de aprimoramento identificadas durante a implementação e avaliação do *chatbot* para a comunidade acadêmica da UFOP, abrem-se perspectivas para futuras pesquisas e desenvolvimentos.

Primeiramente, a expansão contínua da base de dados é essencial. A coleta de documentos mais abrangentes e específicos, abordando diferentes procedimentos acadêmicos e setores da universidade, pode contribuir significativamente para aprimorar a precisão e a amplitude das respostas do *chatbot*.

Outra direção promissora é a integração de fontes externas. A exploração da interconexão do *chatbot* com fontes como sites oficiais da universidade, bases de dados acadêmicas e sistemas de informação pode enriquecer a compreensão do *chatbot* sobre eventos e procedimentos atuais, assegurando respostas atualizadas e precisas.

A incorporação de recursos multimídia também surge como uma possibilidade interessante. Adicionar suporte para imagens, à exemplo do *gpt-4*[1], pode aprimorar a interação, permitindo respostas mais contextualizadas e detalhadas em situações específicas.

Explorar essas direções nos trabalhos futuros não apenas refinaria a acurácia do *chatbot*, mas também contribuiria para a evolução contínua das soluções de assistência virtual em ambientes acadêmicos.

[1] Disponível em <https://openai.com/research/gpt-4>. Acesso em 12/01/2024.

Referências

ABUSHAWAR, B.; ATWELL, E. Alice chatbot: Trials and outputs. Computación y Sistemas, v. 19, 12 2015.

ADAMOPOULOU, E.; MOUSSIADES, L. Chatbots: History, technology, and applications. Machine Learning with Applications, v. 2, p. 100006, 2020. ISSN 2666-8270. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2666827020300062>.

ADAMOPOULOU, E.; MOUSSIADES, L. Chatbots: History, technology, and applications. Machine Learning with Applications, v. 2, p. 100006, 2020. ISSN 2666-8270.

AKBAR, S.; PARDASANI, K. R.; KHAN, F. Swarm optimization-based neural network model for secondary structure prediction of proteins. Network Modeling Analysis in Health Informatics and BioInformatics, Springer Vienna, v. 10, n. 1, p. 1–9, 2021.

AL-AJLAN, A. The comparison between forward and backward chaining. International Journal of Machine Learning and Computing, v. 5, p. 106–113, 04 2015.

ARSENOVIC, M.; SLADOJEVIC, S.; ANDERLA, A.; STEFANOVIC, D. Facetime — deep learning based face recognition attendance system. In: 2017 IEEE 15th International Symposium on Intelligent Systems and Informatics (SISY). [S.l.: s.n.], 2017. p. 000053–000058.

ATMAUSWAN, P.; ABDULLAHI, A. Intelligent chatbot for university information system using natural language approach. Albukhary Social Business Journal, v. 3, p. 6, 2022.

AWAN, A. A. LlamaIndex: Uma estrutura de dados para aplicativos baseados em LLMs (Large Language Models) | DataCamp. 2023. <https://www.datacamp.com/tutorial/llama-index-adding-personal-data-to-llms>. (Accessed on 08/11/2023).

AWASTHI, I.; GUPTA, K.; BHOGAL, P. S.; ANAND, S. S.; SONI, P. K. Natural language processing (nlp) based text summarization - a survey. IEEE, 2021.

BHAGWAT, V. A. Deep learning for chatbots. Master's Projects, San Jose State University, v. 1, p. 52, 2018.

BROWN, T. B.; MANN, B.; RYDER, N.; SUBBIAH, M.; KAPLAN, J.; DHARIWAL, P.; NEELAKANTAN, A.; SHYAM, P.; SASTRY, G.; ASKELL, A.; AGARWAL, S.; HERBERT-VOSS, A.; KRUEGER, G.; HENIGHAN, T.; CHILD, R.; RAMESH, A.; ZIEGLER, D. M.; WU, J.; WINTER, C.; HESSE, C.; CHEN, M.; SIGLER, E.; LITWIN, M.; GRAY, S.; CHESS, B.; CLARK, J.; BERNER, C.; MCCANDLISH, S.; RADFORD, A.; SUTSKEVER, I.; AMODEI, D. Language models are few-shot learners. v. 1, p. 63, 2020.

CASTILLEJO, S. P. Automatic speech recognition: can you understand me? Research-publishing.net, p. 121–126, 2021.

CHANDRA, Y. W.; SUYANTO, S. Indonesian chatbot of university admission using a question answering system based on sequence-to-sequence model. Procedia Computer Science, v. 157, p. 367–374, 2019. The 4th International Conference on Computer Science and Computational Intelligence (ICCSCI 2019) : Enabling Collaboration to Escalate Impact of Research Results for Society.

CHRISTI, J.; JAIN, G. Sentiment categorization through natural language processing. Journal of emerging technologies and innovative research, JETIR, v. 7, n. 6, p. 2344–2351–2344–2351, 2020.

DELEUZE, G.; GUATTARI, F. O que é a filosofia? [S.l.: s.n.], 1997. v. 2.

DENG, L.; LIU, Y. Deep learning in natural language processing. [S.l.]: Springer, 2018.

EXPRESSJS. 2024. Acessado em 17 de janeiro 2024. Disponível em: <https://expressjs.com/>.

GAO, Z.; JIANG, J. Evaluating human-ai hybrid conversational systems with chatbot message suggestions. ACM, p. 534–544, 2021.

GONZALEZ, M.; LIMA, V. L. S. de. Recuperação de informação e processamento da linguagem natural. PUCRS - Faculdade de Informática, 2003.

HA, D.; TANG, Y. Collective intelligence for deep learning: A survey of recent developments. arXiv: Neural and Evolutionary Computing, 2021.

HAN, R.; YIN, Y. Head-driven english syntactic translation model based on natural language processing. IEEE, 2021.

HUSSAIN, S.; NAZIR, R.; JAVEED, U.; KHAN, S.; SOFI, R. Speech Recognition Using Artificial Neural Network. [S.l.]: Springer, Singapore, 2021. 83-92 p.

KARRI, S. P. R.; KUMAR, B. S. Deep learning techniques for implementation of chatbots. 2020 International Conference on Computer Communication and Informatics (ICCCI), p. 1–5, 2020.

KHIN, N. N.; SOE, K. M. University chatbot using artificial intelligence markup language. 2020 IEEE Conference on Computer Applications(ICCA), p. 1–5, 2020.

KROGH, A. What are artificial neural networks? Nature, p. 195–197, 2008.

KUBLIK, S.; SABOO, S. Gpt-3: the ultimate guide to building NLP products with OpenAI API. [S.l.]: Packt Publishing, 2022. v. 1.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. Nature, v. 521, 2015.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. Nature, v. 521, p. 436–444, 2015.

LIN, T.; WANG, Y.; LIU, X.; QIU, X. A survey of transformers. AI Open, v. 3, p. 111–132, 2022. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2666651022000146>.

LIU, J. LlamaIndex. 2022. Disponível em: <https://github.com/jerryjliu/llama_index>.

LUND, B.; TING, W. Chatting about chatgpt: How may ai and gpt impact academia and libraries? Library Hi Tech News, v. 3, p. 26–28, 2023.

MANIMEGALA, R.; PRIYA, K.; RANJANA, S. Brain cancer classification using artificial neural network. v. 7, n. 9, p. 1476–1485, 2020.

MIN, B.; ROSS, H.; SULEM, E.; VEYSEH, A. P. B.; NGUYEN, T. H.; SAINZ, O.; AGIRRE, E.; HEINTZ, I.; ROTH, D. Recent advances in natural language processing via large pre-trained language models: A survey. CoRR, abs/2111.01243, 2021. Disponível em: <https://arxiv.org/abs/2111.01243>.

MONTEIRO, G. S. Helena: Um chatbot para auxÍlio dos discentes do decom em trÂmites universitÁrios. Universidade Federal de Ouro Preto, v. 1, p. 58, 2021.

NADKARNI, P. M.; OHNO-MACHADO, L.; CHAPMAN, W. W. Natural language processing: an introduction. Journal of the American Medical Informatics Association, v. 18, n. 5, p. 544–551, 09 2011. Disponível em: <https://doi.org/10.1136/amiajnl-2011-000464>.

NEXTJS. 2024. Acessado em 17 de janeiro de 2024. Disponível em: <https://nextjs.org/>.

NGUYEN, T. T.; LE, A. D.; HOANG, H. T.; NGUYEN, T. Neu-chatbot: Chatbot for admission of national economics university. Computers and Education: Artificial Intelligence, v. 2, p. 100036, 2021. ISSN 2666-920X. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2666920X21000308>.

NODEJS. 2024. <https://nodejs.org/>. Acessado em 17 de janeiro de 2024.

OGUNTOSIN, V.; OLOMO, A. Development of an e-commerce chatbot for a university shopping mall. Applied Computational Intelligence and Soft Computing, p. 14, 2021.

OPENAI. GPT-4 Technical Report. 2023.

PREEZ, S.; LALL, M.; SINHA, S. An intelligent web-based voice chat bot. In: . [S.l.: s.n.], 2009. p. 386 – 391.

RANOLIYA, B. R.; RAGHUWANSHI, N.; SINGH, S. Chatbot for university related faqs. 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), p. 1525–1530, 2017.

SANTOSO, H. A.; WINARSIH, N. A. S.; MULYANTO, E.; SARASWATI, G. W.; SUKMANA, S. E.; RUSTAD, S.; ROHMAN, M. S.; NUGRAHA, A.; FIRDAUSILLAH, F. Dinus intelligent assistance (dina) chatbot for university admission services. 2018 International Seminar on Application for Technology of Information and Communication, p. 417–423, 2018.

SERVICECONTEXT. 2024. Acessado em 17 de janeiro de 2024. Disponível em: <https://docs.llamaindex.ai/en/stable/module_guides/supporting_modules/service_context.html>.

SHUM, H.-Y.; HE, X.; LI, D. From eliza to xiaoice: Challenges and opportunities with social chatbots. Frontiers of Information Technology and Electronic Engineering, v. 19, 01 2018.

SINGH, S.; SINGH, N. GPT-3.5 vs. GPT-4, Unveiling OpenAI's Latest Breakthrough in Language Models. 2023.

SOHAIL, S. S.; FARHAT, F.; HIMEUR, Y.; NADEEM, M.; MADSEN, D. Øivind; SINGH, Y.; ATALLA, S.; MANSOOR, W. Decoding chatgpt: A taxonomy of existing research, current challenges, and possible future directions. Journal of King Saud University - Computer and Information Sciences, p. 101675, 2023. Disponível em: <https://www.sciencedirect.com/science/article/pii/S131915782300229X>.

STORAGECONTEXT. 2024. Acessado em 17 de janeiro de 2024. Disponível em: <https://docs.llamaindex.ai/en/stable/api_reference/storage.html>.

TAECHARUNGROJ, V. What can chatgpt do? analyzing early reactions to the innovative ai chatbot on twitter. Big Data and Cognitive Computing, v. 7, n. 1, 2023. Disponível em: <https://www.mdpi.com/2504-2289/7/1/35>.

TOPAL, M. O.; BAS, A.; HEERDEN, I. van. Exploring transformers in natural language generation: Gpt, bert, and xlnet. arXiv: Computation and Language, 2021.

VOULODIMOS, A.; DOULAMIS, N.; DOULAMIS, A.; PROTOPAPADAKIS, E. Deep learning for computer vision: A brief review. Computational Intelligence and Neuroscience, p. 13, 06 2018.

WEIZENBAUM, J. Eliza—a computer program for the study of natural language communication between man and machine. Commun. ACM, Association for Computing Machinery, New York, NY, USA, v. 9, n. 1, p. 36–45, jan 1966. ISSN 0001-0782. Disponível em: <https://doi.org/10.1145/365153.365168>.

ZHANG, M.; LI, J. A commentary of gpt-3 in mit technology review 2021. Fundamental Research, v. 1, n. 6, p. 831–833, 2021. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2667325821002193>.

ZIRNSTEIN, B. Extended context for instructgpt with llamaindex. 06 2023.

Anexos

ANEXO A – Perguntas utilizadas para testar os *Chatbots*

As perguntas são as mesmas usadas por Monteiro (2021) para testar o *Chatbot* Helena.

Abater Horas de ATV:

- Como abater horas ATV?
- Como aplico as horas de ATV?
- Como cortar minhas horas de ATV?
- Dar baixa nas horas de ATV.
- Está sendo possível solicitar o lançamento de ATVs?
- Estou com umas ATV's para lançar, referente ao meu estágio. Está sendo possível solicitar esse lançamento?
- Gostaria de saber como está funcionando para cortar minhas horas de ATV durante esse período online?
- Onde preciso entregar os documentos de ATV?
- Entregar documentos de ATV.
- horas de atu.
- hrs de atu.
- hrs de atv.
- Como abter hrs de avt.
- quer saber sobre hrs de avt.
- quero saber sobre hrs de atv.
- onde entregar docs de atv.
- sobre horas de atv.
- dar baixa nas hrs de atv.

- como lancar minhas hrs de atv.
- quero sber mais sobre atv.

Aproveitamento de Disciplinas:

- Aproveitar disciplina de outra faculdade.
- Como aproveitar as disciplinas de outra faculdade?
- Como aproveitar matérias?
- Tem algum jeito de aproveitar outras matérias?
- cmo aproveitar materias.
- como aproveitar disciplinas de outra facul.
- como fzr o aproveitamento.
- tem algum jeito de aproveitar outras materias.
- cmo aproveitar outras disciplinas.
- cmo aproveitar obrigtorias.
- cmo aprveitar discuplinas de outra faculdde?
- tem algma forma de aprovetar outras materias.
- aprvetamento de diciplinas.
- tem com aproveitars disciplinas.
- aprvtar materias.
- tem como usar horas de outra universidade.
- cmo usar hrs de outra faculdade.
- usr hras de outra faculdade.
- tem cmo usar hrs de outra universidade.
- aprov de disciplina.

Calendário acadêmico:

- Calendario academico.
- Onde posso acessar o calendário academico?
- Onde posso consultar o calendário academico?
- clendario academico.
- calendario acdemico.
- consultar o calendario academico.
- calendario das aulas.
- calendario de aulas.
- queria saber do calendario admico.
- clendario acdemico.
- Helena, vc pode me falar sobre o calendario academico?
- vc pode me passar mais informações sobre o calendario acadêmico?
- você sabe sobre o calendário academico?
- cmo posso consultar o calendario?
- e sobre o calendario academico.
- sobr o clendario da universidade.
- calendario da universidade.
- como podemos consultar o calendario academico.
- e sobr o calendario academico.
- sobre o calendário da facul.

Cancelamento de Matrícula:

- Cancelamento.
- cancelmton.
- Cancelamento de matricula.

- Como faz pra cancelar a matrícula.
- Gostaria de saber como faz o cancelamento de matrícula.
- Quero cancelar minha matrícula.
- cmo posso fazer para cancelar minhar matricula.
- canclemento de matricula.
- cnacelamento de maticula.
- cmo cancelar matricula.
- cancelmton de matricula.
- qro cancelar minha mtricula.
- gostria de saber como cancelar.
- gostaria de cancela a matricula.
- qro saber cmo faz pra cancelar matricula.
- cmo fzer pra cancelar minha matrícula.
- qro saber como poçu cancelar mina matrícula.
- clemanto de matrícula.
- consegue me informar sobre o cacelamento de matrícula.
- conseguiria mme dizer mais sobre o cancelamento de matrícula.

Carteira Estudantil:

- Carteirinha.
- Onde posso pedir a segunda via da carteirinha.
- Pra que serve a carteirinha.
- Pra que serve a carteirinha estudantil?
- Segunda via carteirinha.
- Segunda via da carteirinha de estudante.
- obter segunda via de carteirinha.

- carteirinha estudantil.
- como ter caeteirinha estudantil.
- catrinha estudantil.
- Ei helena, você sabe como posso obter a carteirinha.
- o q é a carteirinha.
- onde q posso pedir a carteirinha.
- o q é a catrinha estundatil.
- o que é a carteirinha de estudante.
- cartrinha d estudante.
- carteira de estudante.
- sobre a carteinha estudantil.
- cmo pedira carteirinha de estudante.
- como pedir a carteirinha d estudante.

Certificado de Monitoria e Tutoria:

- Como aplicar horas de monitoria.
- Como aplicar horas de tutoria.
- Como validar minhas horas de monitoria?
- Como validar minhas horas de tutoria?
- cmo vlidar as minhas hras de turorial.
- cmo validar as minhas hrs d tutoria.
- Helena, consegue me dizer algo sobre monitoria.
- Como validar minhas hrs de monitoria e tutoria.
- Como ter os certificados de monitoria e tutoria.
- cmo validar meus certificados de monotoria e tutoria.
- certificados de monitoria e tutoria.

- monitoria e tutoria.
- helena, validar as horas de tutoria.
- Olá helena, cmo posso validar minhas horas de monitoria.
- cmo obter os ceertificados de monitoria.
- pegar os certificados de tutoria.
- consegue me dizer sobre hras de monitoria e tutoria.
- monitoria/tutoria.
- cmo compensar hrs de monitoria.
- como compensar horas de monitoria.

Colação de grau:

- Colação de grau extraordinária.
- Estudante não apto a colação de grau.
- Não colação de grau.
- Pedir colação de grau.
- Problemas com colação de Grau.
- Protocolar o requerimento de colação de grau extraordinária.
- colação de grau.
- como solicitar a colação de grau extraordinaria.
- como posso saber se estou apto para colação.
- colassão de gral.
- Helena, gostaria de saber mais sobre colassaum de gral.
- colação de gral estraordinária.
- como posso pedir a colação extraordinária.
- colação extraodinária.
- solicitação de colação de grau extraordinária.

- solicitação de colação.
- como consigo solicitar uma colação extraodinária.
- cmo posso protocolar um requerimento de colassao.
- colassao estraodinaria.
- colasao extra-ordinaria.

Desligamento da Universidade:

- Como acontece o desligamento.
- Desligamento.
- Desligamento da universidade.
- Quando posso ser desligado.
- Quando posso ser desligado da universidade.
- Quando um aluno é desligado?
- quando sou desligado?
- o que é desligamento.
- quando uma pessoa é deligada?
- deligamento.
- o q é desligamento?
- desligamento da universidade.
- deligamento da universidade.
- quando sou desligado.
- posso ser desligado.
- tive três coeficientes abaixo de 3.0, posso ser desligado?
- quando meus coeficientes podem causar desligamento?
- fui reprovada em todas as materias esse perído, posso ser desligada?
- quais são as condições para desligamento da universidade?
- Quais são as condições para desligamento.

Estágio:

- Como funciona o plano de atividades.
- Documentação de Estágio.
- Documentação para contrato de estágio.
- Estágio não obrigatório.
- Para onde enviar a documentação de estágio?
- Preciso de um professo específico para assinar o contrato de estágio?
- Processo de aprovação de estágio.
- Processo de aprovação de estágio pela universidade.
- Quais documentos preciso para contrato de estágio?
- Quais são os documentos para estágio?
- Qual professor deve assinar o contrato de estágio?
- documentação de estágio não obrigatório.
- pode me informar sobre a documentação de estágio.
- docs de estgio.
- documentaçaõ de estágio.
- cmo obter a documentação de estágio.
- documentção de estágio.
- plano de atvs.
- Passei em um estagio qual é a documentação.
- o que preciso de documentação pra estágio.

Exame Especial:

- Como funciona o exame especial.
- Exame especial.
- Gostaria de saber mais sobre os exames especiais.

- Quando acontece o exame especial.
- Quando posso fazer o exame especial.
- esames especiais.
- quando acontecem os exames.
- quando devo fazer os exames especiais.
- qual é a condição para fazer exames especiais.
- o que é o esame especial.
- cmo é o exame espesial?
- o que é exame parcial total.
- o q exame total parcial.
- qndo devo fzer o exame espacial.
- quando acontecem os esames especiais.
- ei helena, me conte sobre os exames especiais.
- o professor pode não aplicar o exame especiais.
- como fazer fazer os exames especiais.
- fazr os ezames especiais.
- examis especiais.

Horas Válidas de ATV:

- Eu consigo abater horas de atv 100 com horas de trabalho como desenvolvedor CLT?
- Horas válidas de atv.
- Quais atividades são válidas para atv?
- Quais atividades são válidas para atv100?
- Quais horas são válidas para atv?
- Tutoria conta como pontos para receber ATV 100?
- O que posso validar como atv.

- o que são hrs válidas atv.
- Atividades de monitoria contam como atv?
- O q é valido como atv100.
- quais hrs podemos validar como atv?
- como posso validar como hrs de estágio.
- hras válidas cmo atv?
- o que posso contar cmo atv.
- quais hrs são válidas para atu.
- o que conta cmo hrs atv.
- quais atividades são válidas pra extracurriculares.
- fiz um curso, posso validar cmo atv.
- o que é valiado para contar cmo hras de avt.
- quantas horas preciso para cumprir atv100.

Trancamento de disciplina:

- Como trancar minha disciplina.
- Como trancar uma disciplina?
- Como trancar uma matéria?
- O que é trancamento de disciplina?
- trancamento.
- trancamento de disciplina.
- cmo fazer o trancamento de disciplina.
- tracamento de diciplina.
- cmo acontece por disciplina.
- quando q posso trancar uma disciplina.
- cmo trancar uma matéria.

- como trancar uma cadeira.
- posso trancar uma cadeira no final de período.
- trancamemnto de cadeira.
- cmo fazer o trancamento de uma diciplina.
- quando posso fazer o trancamento de cadeira.
- cmo posso trancar uma matéria.
- o que é o tracamento de diciplina.
- até quando posso trancar matrícula.
- helena, como é trancar matéria.