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

MARIA EDUARDA DE AZEVEDO SILVA

"ZoIA": UMA FERRAMENTA PARA DESCRIÇÃO AUTOMÁTICA DE IMAGENS NO NAVEGADOR

CAMPINA GRANDE - PB

2023

MARIA EDUARDA DE AZEVEDO SILVA

"ZoIA": UMA FERRAMENTA PARA DESCRIÇÃO AUTOMÁTICA DE IMAGENS NO NAVEGADOR

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

Orientador : Herman Martins Gomes

CAMPINA GRANDE - PB

2023

MARIA EDUARDA DE AZEVEDO SILVA

“ZoIA”: uma ferramenta para descrição automática de imagens no navegador

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

BANCA EXAMINADORA:

Herman Martins Gomes

Orientador – UASC/CEEI/UFCG

Eanes Torres Pereira

Examinador – UASC/CEEI/UFCG

Francisco Vilar Brasileiro

Professor da Disciplina TCC – UASC/CEEI/UFCG

Trabalho aprovado em: 15 de Maio de 2024.

CAMPINA GRANDE - PB

RESUMO

Para navegar na internet, pessoas com deficiência visual utilizam leitores de tela. A garantia do bom funcionamento dessas ferramentas está atrelada à responsabilidade dos desenvolvedores web em utilizar as tags de maneira correta e fornecer informações suficientes, como descrições de imagens, para garantir a acessibilidade e a inclusão desses usuários. Porém, o fornecimento de textos alternativos não é sempre garantido por diversos fatores, como, por exemplo, a ausência de descrições associadas às imagens assim como de requisitos e funcionalidades que busquem garantir a existência desse dado. Diante dessa problemática, este trabalho objetiva propor o "ZoIA", um plugin para navegadores baseado em Inteligência Artificial, que descreve e insere textos alternativos para imagens dispostas em páginas na internet. A análise comparativa entre a geração direta de descrições em português e a abordagem via sequência sugeriu que o ajuste do modelo para o idioma de interesse pode resultar em melhorias qualitativas no contexto, semântica e coesão gramatical das descrições automáticas. Neste trabalho buscou-se contribuir para a melhoria da interação e da experiência de pessoas com deficiência visual na utilização de aplicações web com o uso de estratégias do estado da arte da Inteligência Artificial.

“ZoIA”: a tool for automatic image captioning in web browsers

ABSTRACT

People with visual impairments use screen readers to browse the internet. Ensuring that these tools work properly is linked to the responsibility of web developers to use tags correctly and provide sufficient information, such as image descriptions, to ensure accessibility and inclusion for these users. However, the provision of alternative texts is not always guaranteed due to various factors, such as the absence of descriptions associated with images, as well as requirements and functionalities that seek to guarantee the existence of this data. Faced with this problem, this work aims to propose “ZoIA”, a browser plugin based on Artificial Intelligence, which describes and inserts alternative texts for images on web pages. The comparative analysis between the direct generation of descriptions in Portuguese and the approach via sequence suggested that adjusting the model for the language of interest can result in qualitative improvements in the context, semantics and grammatical cohesion of automatic descriptions. This work sought to contribute to improving the interaction and experience of visually impaired people when using web applications using state-of-the-art Artificial Intelligence strategies.

“ZoIA”: uma ferramenta para descrição automática de imagens no navegador

Maria Eduarda de Azevedo Silva
Departamento de Sistemas e Computação
Universidade Federal de Campina Grande
Campina Grande, Paraíba, Brasil
maria.silva@ccc.ufcg.edu.br

Herman Martins Gomes
Departamento de Sistemas e Computação
Universidade Federal de Campina Grande
Campina Grande, Paraíba, Brasil
hmg@computacao.ufcg.edu.br

Abstract

O bom funcionamento de leitores de tela, comumente utilizados por usuários com deficiência visual, está atrelado à responsabilidade dos desenvolvedores web em utilizar as *tags* de maneira correta e fornecer informações suficientes, como descrições de imagens, para garantir a acessibilidade e a inclusão desses usuários. Porém, o fornecimento de textos alternativos não é sempre garantido por diversos fatores, como, por exemplo, a ausência de descrições associadas às imagens, assim como de requisitos e funcionalidades que busquem garantir a existência desse dado. Diante dessa problemática, este trabalho propõe o “ZoIA”, um *plugin* para navegadores, fundamentado em Inteligência Artificial, que descreve e insere textos alternativos para imagens dispostas em páginas na internet. Uma análise comparativa entre a geração direta de descrições em português e a abordagem via sequência sugeriu que o ajuste do modelo para o idioma de interesse pode resultar em melhorias qualitativas no contexto, semântica e coesão gramatical das descrições automáticas. Neste trabalho buscou-se contribuir para a melhoria da interação e da experiência de pessoas com deficiência visual na utilização de aplicações web com o uso de estratégias do estado da arte da Inteligência Artificial.

Palavras-chave: Acessibilidade, tecnologias assistivas, Inteligência Artificial, descrição automática de imagens.

1 Introdução

A interação humano-computador passou por uma intensa transformação ao longo da evolução da digitalização dos mais diversos setores na sociedade. Ainda no início do milênio, estima-se que, nos Estados Unidos, cerca de 66% da população adulta já utilizava a internet de maneira regular, o que correspondia a um crescimento de 700% de usuários desde 1995 (Chiang et al. (2005)). O uso da internet para realização de atividades do dia a dia aumentou exponencialmente, compreendendo mais de 4 bilhões de usuários em todo mundo no ano de 2020 (Ritchie et al. (2023)), se tornando o principal meio de comunicação e de realização de tarefas nas mais diversas áreas.

Porém, apesar da grande transformação e adesão ao uso de meios digitais, sérios desafios ainda se apresentam no tocante à garantia de um ambiente acessível para todos os seus usuários, em especial pessoas com deficiência, que necessitam de recursos nem sempre assegurados para navegar na web.

1.1 Descrição do Problema

No recorte nacional, cerca de 3,4% da população brasileira, equivalente a quase 7 milhões de pessoas, apresentavam algum tipo de dificuldade ou deficiência visual, segundo dados da Pesquisa Nacional de Saúde (PNS) de 2019 [1]. Porém, apesar dessa parcela considerável, um estudo da BigDataCorp, em parceria com o Movimento Web para Todos[2], que analisou a estrutura de um conjunto de 30 milhões de sites brasileiros, concluiu que cerca de 95,78% dos websites ainda carecem de boas práticas de marcação HTML, necessárias para a boa performance de leitores de tela, e 84,21% não garantem a presença de textos alternativos para a descrição de imagens. Tais indicadores eviden-

[1] https://agenciadenoticias.ibge.gov.br/agencia-sala-de-imprensa/2013-agencia-de-noticias/releases/31445-pns-2019-pais-tem-17-3-milhoes-de-pessoas-com-algum-tipo-de-deficiencia

[2] https://mwpt.com.br/numero-de-sites-brasileiros-aprovados-em-todos-os-testes-de-acessibilidade-tem-queda-em-relacao-ao-ano-passado-e-e-ainda-menor-que-1/

ciam a urgência do desenvolvimento de estratégias e soluções que busquem garantir a acessibilidade do conteúdo na web por parte de pessoas com deficiência.

1.2 Direção de Solução

O constante avanço do estado da arte da Inteligência Artificial tem proporcionado uma série de aplicações focadas em tarefas de melhoria de acessibilidade. No contexto da descrição automática de imagens, propostas relacionadas ao uso de mecanismos de auto-atenção para a geração de legendas mais coerentes com base no contexto de uma dada imagem como entrada se mostram uma opção viável para a melhoria dos resultados da atividade (Liu et al. (2020)), aspecto que acabou ganhando força e popularização pelo advento das modernas arquiteturas de *transformers* (Vaswani et al. (2023)).

Em adição, a cultura Open Source[3] e sua grande capacidade de criar e engajar comunidades em torno de uma causa pode ser um agente de grande contribuição para a evolução de soluções de *software* que promovam uma melhoria significativa na usabilidade de páginas web por parte de pessoas com deficiência, tornando o ambiente da internet mais amigável e acessível.

1.3 Objetivos

Este trabalho objetiva propor o "ZoIA", um *plugin* para navegadores capaz de colher, processar e gerar descrições automáticas para imagens incorporadas no conteúdo das páginas web.

Em adição, experimentos de descrição automática de imagens para língua portuguesa foram realizados com o objetivo de comparar a geração de texto alternativo com base no ajuste de um *transformer* utilizando uma base de dados em língua portuguesa e a mesma tarefa realizada por uma série de modelos pré-treinados. Tal estudo visa observar se existe viabilidade e eficácia de uma técnica de refinamento para melhorar o teor semântico das descrições automaticamente geradas em detrimento de estratégias baseadas em geração intermediária e tradução.

1.4 Estrutura

A próxima seção trata de uma Contextualização (Seção 2) acerca das temáticas que permeiam a problemática e a solução. Na Seção 3, apresenta-se uma descrição do *plugin* desenvolvido, juntamente com sua arquitetura, módulos de API, Interface com o usuário e Integração da solução final. A Seção 4 contém uma apresentação e discussão dos experimentos realizados. Por fim, a síntese das propostas, resultados e contribuições é realizada na Conclusão (Seção 5) e Trabalhos Futuros (Seção 6) são evidenciados a partir das contribuições deste trabalho.

2 Contextualização

Esta seção apresenta uma série de tópicos fundamentais para a compreensão do problema de acessibilidade na web e dos métodos do estado da arte da Inteligência Artificial que impulsionaram os avanços no domínio de descrição automática de imagens.

2.1 Acessibilidade na Web

Na tentativa de criar padrões para a melhoria da acessibilidade das páginas web, uma série de recomendações foi organizada pelo *World Wide Web Consortium* (W3C), denominada de *Web Content Accessibility Guidelines* (WCAG)[4]. Tais diretrizes acomodam padrões que visam tornar o conteúdo da web mais acessível para uma série de usuários, sendo organizadas sob a ótica de quatro princípios: perceptibilidade, operabilidade, compreensibilidade e robustez.

O princípio de perceptibilidade busca assegurar que informações contidas na interface de usuário sejam apresentadas de maneira perceptiva para os mesmos, isto é, os usuários devem ser capazes de perceber o conteúdo em tela, independentemente de sua natureza textual. Por sua vez, o princípio de operabilidade tem como principal preocupação a garantia de que os componentes e a navegação das páginas sejam operáveis, ou seja, os usuários devem ser aptos a operar a interface para seu objetivo. Já a compreensibilidade abarca o fato de que as informações e as operações sobre a interface devem ser compreensíveis, isto é, o usuário deve ser capaz de compreender a usabilidade e a natureza das informações e das operações dispostas na interface. Por fim, a robustez vem para assegurar que o conteúdo possa ser interpretado por uma vasta variedade de agentes e tecnologias assistivas, ou seja, o conteúdo deve continuar sendo de bom acesso à medida que a tecnologia e os

[3] https://opensource.org/

[4] https://www.w3.org/TR/WCAG21/

próprios usuários, juntamente com suas necessidades, evoluem.

Análises em diferentes contextos realizadas com base nesses padrões indicam problemas recorrentes com a garantia, em especial, de critérios de aceitação relacionados à perceptibilidade e à robustez (Teixeira et al. (2019))(Yaokumah et al. (2015)), fato que impacta diretamente a experiência e a usabilidade de pessoas com deficiência visual ao navegarem por páginas na internet.

Em se tratando de sites em português brasileiro, o Movimento Web para Todos[5], em uma parceria firme com a BigDataCorp, realiza anualmente uma pesquisa de acessibilidade em páginas web ativas no país. A metodologia adotada pelos pesquisadores responsáveis utiliza de técnicas de identificação de recursos acessíveis com base em padrões de acessibilidade para links, imagens e formulários. Os resultados históricos desse estudo revelam que, com o passar do tempo, ao invés de haver evolução nos mecanismos para proporcionar sites mais acessíveis, houve, em geral, uma piora nos índices computados, tendo em vista o crescimento do número de páginas, incluindo os quesitos que mais estão vinculados à navegação de pessoas com deficiência visual (Imagens e Verificação de marcação do HTML).

2.2 Descrição Automática de Imagens

Durante a evolução dos meios midiáticos, muitos mecanismos de comunicação tiveram e continuam tendo como base conteúdos de origem imagética, desde os jornais impressos, passando pela televisão e chegando à atual era das redes sociais. Alguns recursos de acessibilidade para que pessoas com deficiência visual possam consumir conteúdos contendo imagens têm dependência de curadoria totalmente humana, como a audiodescrição presente no cinema e na televisão.

Porém, no contexto da internet, onde a gama de informações é drasticamente maior, existe uma dificuldade na garantia da existência e da geração de conteúdo alternativo para o consumo de dados de natureza visual, ocasionada por diversos fatores que vão desde a falta de responsabilidade dos desenvolvedores até a não consciência dos usuários finais quanto à importância desses artifícios para a usabilidade da web por pessoas com deficiência. Esse cenário se mostra favorável à adoção de mecanismos automatizados que possam gerar descrições alternativas de maneira a suprir minimamente a necessidade apresentada.

Uma das tarefas estudadas na Visão Computacional, denominada de *Image Captioning*, tem como principal objetivo a geração de descrições automáticas para imagens com base em aspectos relacionados à identificação de objetos, ações e relacionamentos entre elementos dentro de uma cena. Esse campo de estudo passou por grandes evoluções e foi bastante beneficiado pelo atual estado da arte da Inteligência Artificial.

Em abordagens iniciais, a descrição automática de imagens era tratada como um problema de recuperação da informação. Um motor de busca era projetado para receber como entrada uma imagem e pesquisar em um *corpus* de descrições previamente definidas aquela que melhor se adequava às características extraídas do dado visual. Essa estratégia, por mais que efetiva, apresentava um claro problema: não existia possibilidade das descrições geradas serem flexíveis ao conteúdo presente no dado, perdendo expressividade relacionada ao contexto.

Com isso, novas abordagens baseadas em redes neurais profundas começaram a ser aplicadas, buscando uma maior liberdade na descrição dos conteúdos. Uma estratégia do atual estado da arte que elevou o nível da qualidade das descrições automáticas foram as arquiteturas *Encoder-Decoder*. Nessa abordagem a tarefa de *Image Captioning* passa a ser percebida como um problema de tradução, no qual queremos traduzir uma imagem dada como entrada para um texto semanticamente coerente na saída. Um outro benefício desse método é a possibilidade de combinar diferentes arquiteturas de codificadores e decodificadores, podendo utilizar desde Redes Neurais Convolucionais (CNNs), para codificação das imagens, e Redes Neurais Recorrentes (RNNs), para decodificação em texto, até Transformers Visuais (ViTs) combinados com LLMs para a resolução da mesma tarefa (Sharma et al. (2020)).

2.3 Transformers Visuais

No contexto de aprendizagem profunda envolvendo diversas tarefas de Visão Computacional, as arquiteturas baseadas em Redes Neurais Convolucionais (CNNs) (LeCun et al. (1989)) têm historicamente se destacado. Porém, a grande

[5]https://mwpt.com.br/

adesão e ótimos resultados dos mecanismos de atenção das arquiteturas de *Transformers* no Processamento de Linguagem Natural, acabou por levantar questionamentos acerca da aplicação de métodos análogos em tarefas relacionadas a imagens.

Os modelos Transformers Visuais (ViT), inicialmente propostos por Dosovitskiy et al. (2021), fundamentam-se nas arquiteturas de *Transformers* aplicadas a tarefas de processamento de linguagem natural com modificações para o contexto de aplicação. Tais modelos apresentaram resultados muito próximos aos conseguidos por redes ResNet (He et al. (2015)), consideradas até então o estado da arte para tarefas como classificação de imagens.

As alterações realizadas da arquitetura original, própria para tratar de entradas textuais, foram principalmente para abarcar o contexto dos dados de imagem, recebendo e processando sequências bidimensionais ao invés de unidimensionais. Um dos maiores problemas enfrentados na aplicação de *Tranformers* em dados visuais é a complexidade do cálculo de atenção, que acaba por escalar de maneira a se tornar impraticável em sequências 2D. A maneira encontrada pelos autores para diminuir o custo dessas operações foi trabalhar com um mecanismo de atenção global, a partir da divisão da imagem de entrada em partições denominadas de *patches*. Essas subdivisões são organizadas sequencialmente e combinadas com *embeddings* posicionais por meio de uma projeção linear, que por sua vez são dados como entrada do codificador do *Transformer* proposto por Vaswani et al. (2023).

Após análises, os autores concluíram que modelos ViT, em comparação com as ResNets, representando a solução *baseline* para a tarefa de classificação de imagens, têm desempenho igual ou superior, mas elencaram para possíveis trabalhos e desafios futuros a aplicação e avaliação do método em outras tarefas da Visão Computacional. Contudo, recentes revisões acerca da aplicação de abordagens baseadas em *Vision Transformers* se mostram as mais promissoras para tarefas que envolvem a intersecção entre visão e linguagem, como a descrição de cenas (Ondeng et al. (2023)).

2.4 Grandes Modelos de Linguagem

O interesse humano na modelagem de linguagens naturais data bem antes dos grandes marcos históricos da Inteligência Artificial. Estudos relacionados à linguística, como *"Cours de Linguistique Générale"* (Desaussure (1965)), que teve sua primeira publicação ainda em 1916, acabaram por culminar em importantes gatilhos para a evolução do Processamento de Linguagem Natural como subárea do aprendizado de máquina.

Uma das primeiras grandes necessidades humanas em termos do processamento de linguagens se deu por parte do interesse de mecanismos para realizar traduções entre idiomas de maneira automatizada. Esse problema acabou servindo como um guia para o avanço das pesquisas na área de processamento de textos, encontrando um cenário favorável com a evolução das Redes Neurais Artificais, da aprendizagem profunda e da era do Big Data - culminando no atual estado da arte, baseado nas arquiteturas de *Transformers*, propostas para resolução desse exato problema.

Tais fatores proporcionaram o surgimento e a consolidação dos *Large Language Models* (LLMs), modelos de linguagem complexos baseados em *Transformers* e treinados em um conjunto massivo de dados textuais. Esses modelos alcançam a marca de milhões, bilhões e, recentemente, até trilhões de parâmetros, o que os dão uma alta capacidade e a característica de desenvolverem habilidades emergentes do treinamento, como aprendizado por contexto, raciocínio em múltiplas etapas e aptidão para seguir instruções (Minaee et al. (2024)). Essa propriedade possibilita o projeto de agentes de Inteligência Artificial de propósito geral, baseados em linguagem natural e com aplicabilidade nos mais diversos setores e contextos.

Pela sua alta capacidade, esses modelos também são aplicados em contextos de resolução de tarefas multimodais, isto é, que lidam com dados de diferentes naturezas, como a sintetização de imagens, áudios e vídeos a partir de *prompts* textuais ou a geração de descrições para imagens. Essa amplitude de aplicação faz com que o atual estado das LLMs corrobore com uma nova perspectiva para o futuro da Inteligência Artificial, proporcionando um comportamento cada vez mais próximo das capacidades humanas por parte das IAs generativas (Song et al. (2023)).

3 Ferramenta

Esta seção descreve decisões e funcionamento da ferramenta desenvolvida neste trabalho, passando

por aspectos de arquitetura, funcionalidades e integração. A implementação da proposta pode ser encontrada em um repositório do GitHub[6] aberto para recebimento de contribuições da comunidade.

3.1 Arquitetura

A ferramenta foi construída nos moldes da arquitetura disposta na Figura 1. Existem dois módulos principais que encapsulam as funcionalidades da solução: a interface de interação com o usuário e a API que integra e implementa as operações de administração e os modelos de Inteligência Artificial. Tais estruturas serão detalhadas em termos de tecnologias e funcionalidades nas subseções subsequentes.

3.2 API

A API da aplicação desempenha um papel fundamental ao proporcionar a base de interação entre a interface principal da aplicação e os modelos que possibilitam a descrição das imagens. Além disso, esse módulo implementa uma interface secundária, ligada diretamente ao serviço, responsável por gerenciar a administração do sistema em sua proposta. Nesta subseção, iremos descrever detalhes acerca da implementação e dos papéis desempenhados por este módulo para os diferentes tipos de usuário da solução.

A API do "ZoIA" foi pensada com a função principal de integrar uma sequência de modelos capaz de processar e gerar descrições automáticas em língua portuguesa para imagens, a partir do recebimento desses dados via uma requisição HTTP. Para atender a tal propósito, a linguagem de programação Python3 junto ao framework Flask[7], para construção de APIs REST, guiaram o desenvolvimento da aplicação. A escolha para o uso dessa linguagem é justificada principalmente pela modularização e o uso convencional das interfaces e bibliotecas que manipulam os modelos de Inteligência Artificial utilizados. Python apresenta uma vasta gama de bibliotecas e frameworks específicos para construção e experimentação de soluções baseadas em IA, com uma comunidade bastante ativa, aspectos que contribuem para o uso de funcionalidades avançadas para construção e treinamento de modelos que acompanham o estado da arte sem altos custos de implementação e a facilidade de manutenção das abordagens desenvolvidas.

Dentre os frameworks para construção de APIs REST na linguagem, o Flask foi escolhido pela sua simplicidade como ferramenta, tendo uma interface favorável para a rápida compreensão e criação de novos *endpoints*, mantendo a arquitetura do código e o fluxo de dados de maneira organizada e intuitiva. Além disso, a tecnologia também apresenta uma comunidade ativa, que favorece a implementação de pacotes de extensão para a implementação de estratégias de administração e autenticação robustas e escaláveis sem muitos custos de desenvolvimento, aproveitadas no escopo desse projeto. Por fim, outro ponto importante é a facilidade de manutenção que o framework proporciona, provendo uma arquitetura de fácil compreensão para aqueles que desejam manter instâncias independentes ou que almejam contribuir com a solução em geral dentro da cultura *Open Source*, acelerando os ciclos de desenvolvimento e construindo um espaço para a criação de uma comunidade ativa e engajada em torno do projeto.

Como já citado anteriormente, a API implementa uma estratégia de autenticação e administração do sistema. Essa estrutura contém um *frontend* simples, diretamente acoplado ao lado do servidor, roteado a partir da renderização de templates HTML manipulados via Python e suportados pelo próprio Flask. Tal interface compreende em um *dashboard* de administração, que pode ser acessado por usuários do tipo ADMIN (usuários com o poder de adicionar ou editar os papéis de outros usuários do sistema, sejam eles usuários finais ou também administradores) e SUPERADMIN (usuários capazes de realizar grandes operações de administração no sistema, como criação e edição de qualquer outro usuário). Os usuários finais do sistema, do tipo USER, devem também ser criados pelas entidades administradoras. Para a criação e persistência das representações desses usuários, foi necessário conectar a API a uma instância de banco de dados. Para facilidade no momento do desenvolvimento, o banco de dados escolhido e mantido por padrão foi o SQLite[8], dado que o mesmo pode ser criado na forma de um simples arquivo. Porém, em termos de implementação, a aplicação é capaz de acoplar e suportar instâncias mais robustas para serem escaladas em produção, aspecto garantido pelo uso do pacote de extensão Flask-

[6] https://github.com/MariaEduardaDeAzevedo/ZoIA
[7] https://flask.palletsprojects.com/en/3.0.x/
[8] https://www.sqlite.org/

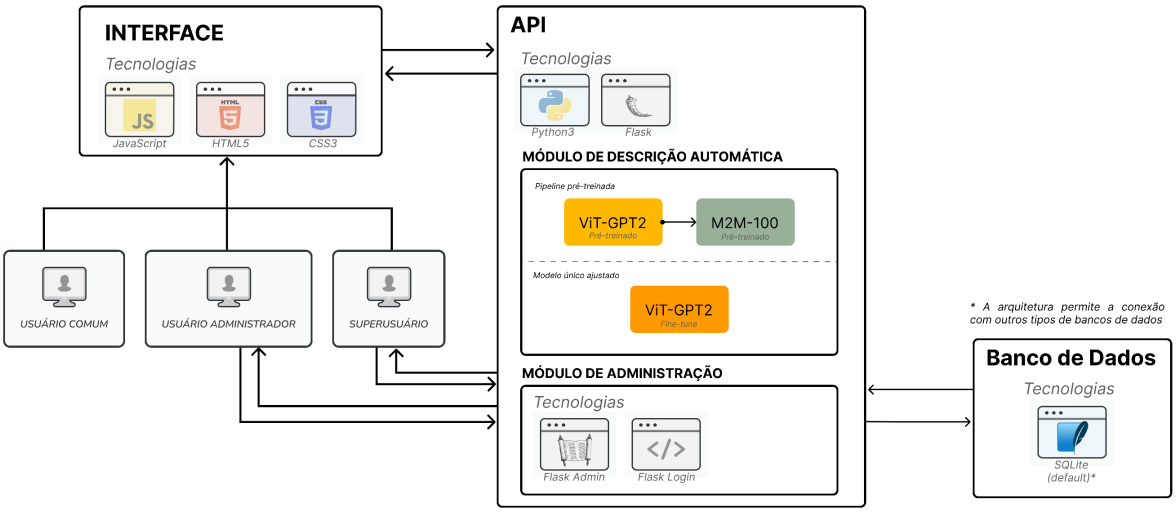

Figura 1: Arquitetura da ferramenta ZoIA.

SQLAlchemy[9]. Essa extensão facilita a integração do framework com o *toolkit* SQLAlchemy, popular ferramental que possibilita o mapeamento objeto-relacional (ORM) na linguagem Python, simplificando a realização de tarefas comuns em sistemas com bancos de dados, como definição de modelos, esquemas e operações de consulta, inserção, atualização e exclusão, também conhecido como o padrão CRUD.

Quando uma instância da API é implantada (localmente ou como um serviço), por default um SUPERADMIN é criado automaticamente, e pode ser acessado por credenciais padrões contidas nas configurações do projeto. Ao executar a API, o usuário pode fazer *login* nessa conta padrão acessando o formulário de *login* do *dashboard* da aplicação. Após se autenticar, o superadministrador, assim como os demais papéis que têm acesso à tal área, podem visualizar uma lista com os usuários cadastrados no sistema e informações a ele relacionadas, como horário de criação e edição, *status* de ativação, token de acesso do mesmo e o usuário responsável pela última operação sobre aquela entidade. O salvamento dessas informações e operações relacionadas dão à ferramenta uma capacidade de auditoria no processo de administração dos participantes do sistema.

No *dashboard* os administradores são também capazes de cadastrar novos usuários, a partir de um e-mail válido, e atribuir um papel ao mesmo, além de editar usuários já existentes no sistema. Ao criar o usuário, um token de acesso é gerado de maneira automática e aleatória e associado a sua conta. Esse token é utilizado como uma senha de acesso, sendo papel do administrador entrar em contato com o usuário cadastrado repassando essas credenciais para sua autenticação no sistema.

Já na parte do serviço responsável pela implementação das funcionalidades, são duas as estruturas básicas de *controller*: um primeiro que autentica os usuários na ferramenta e um segundo que implementa a funcionalidade de descrição de imagens.

O *controller* de autenticação do usuário gerencia o processo de autenticação dos utilizadores da ferramenta. Ele consiste em dois *endpoints* principais, integradas pela interface de interação principal da solução: um para *login* e outro para *logout*. O *endpoint* de *login* é responsável por receber as credenciais de autenticação, consistindo no e-mail cadastrado por uma entidade administradora e o token de acesso a ele vinculado, e verificar se correspondem a um usuário válido no sistema. Caso as credenciais sejam válidas, o *endpoint* gera um token JWT que é retornado ao cliente para autenticação subsequente na funcionalidade de gerar descrições automáticas para a imagem. Se as credenciais não forem válidas, o *endpoint* retorna uma mensagem e um código de erro adequados para a situação. Por outro lado, a rota de *logout* permite que os usuários encerrem sua sessão atual na aplicação. Quando um usuário desloga, o token de acesso associado a sua sessão é invalidado, impedindo que ele seja utilizado para

[9]https://flask-sqlalchemy.palletsprojects.com/en/3.1.x/

autenticação futura, o que garante a segurança da conta do usuário e protege contra acesso não autorizado.

No *controller* de descrição automática, é implementada uma única rota. Para integrá-la, o cliente precisa enviar o token JWT gerado durante o processo de autenticação no cabeçalho da requisição e, como corpo, uma lista de URLs ou de imagens codificadas em Base64. Ao longo desta rota, essas imagens são armazenadas em um diretório temporário do sistema operacional que executa a instância da API e são usadas como entrada para uma sequência de modelos capaz de gerar descrições em português para os dados. Na resposta, esse o *endpoint* retorna um objeto contendo uma lista de descrições, seguindo a ordem exata das imagens indicadas na requisição. Se não for possível gerar uma descrição para uma determinada imagem, uma mensagem padrão é enviada no índice correspondente, indicando a impossibilidade de descrevê-la. Na Figura 2 é possível observar um diagrama de sequência que descreve o fluxo seguido por essa rota da API em sua implementação.

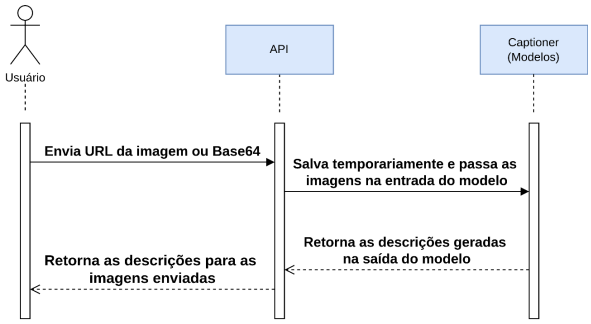

Figura 2: Diagrama de sequência representando fluxo da funcionalidade principal do sistema.

3.3 Sequência de Modelos

Para implementação de uma solução independente de treinamentos específicos para o contexto, isto é, para que seja possível gerar as descrições das imagens em língua portuguesa sem a necessidade de um ajuste fino, um encadeamento de modelos foi arquitetado para tal.

A sequência pensada apresenta um *transformer* do tipo ViT-GPT2 (NLP Connect (2022)), pré-treinado em toda base de dados *Microsoft Common Objects in Context*, popularmente conhecida como COCO (Lin et al. (2014)), em sua versão de 2017. Para complementar, foi acoplado à saída desse modelo o tradutor multilíngual M2M100 (Fan et al. (2020)), mantido e pré-treinado pela empresa Meta.

O modelo ViT-GPT2, uma combinação do Vision Transformer (ViT) e do GPT-2 (Generative Pre-trained Transformer 2), é uma arquitetura de rede neural *Encoder-Decoder* que integra elementos de visão computacional e processamento de linguagem natural. O ViT-GPT2 opera em duas etapas distintas: na primeira etapa, o ViT, originalmente desenvolvido para tarefas de visão computacional, processa a imagem de entrada dividindo-a em *patches* e os transformando em sequências de tokens. Em seguida, essas são passadas na entrada do GPT-2, uma LLM pré-treinada para geração de texto. O GPT-2 utiliza o contexto visual fornecido pelo ViT para aprimorar a sua saída, com o propósito de gerar descrições mais precisas e contextualizadas para as imagens de entrada. Essa abordagem combina a capacidade do ViT de entender o conteúdo visual com a habilidade do GPT-2 de gerar linguagem natural dotada de contexto, resultando em um modelo capaz de resolver bem a tarefa de descrição de imagens.

Outras arquiteturas que combinam um codificador visual com um decodificador em texto podem ser pensadas para a resolução do mesmo problema, combinando desde redes neurais do tipo CNN e RNN até diferentes arquiteturas *Transformer-based*. A escolha desse modelo específico está vinculada aos bons resultados produzidos por uma opção que é facilmente encontrada em respositórios *Open Source* e com a facilitada capacidade de integração, podendo gerar inferências de maneira simples até em domínios locais, o que corrobora para um bom custo benefício dentro da proposta da ferramenta.

Na segunda parte da sequência está acoplado um modelo de tradução multilingual: o M2M100. A sigla é dada para *Many-to-Many Multilingual Model 100*, indicando a possibilidade do modelo em gerar traduções entre 100 diferentes idiomas sem a necessidade de uma tradução intermediária para o inglês. Sua arquitetura é baseada em *transformers* do tipo *Sequence-to-Sequence*, também composta de um módulo codificador e outro decodificador, ambos compostos pela mesma estrutura de camadas. A estratégia de treinamento desse modelo se baseou em uma base de dados multilíngual, possuindo textos em 100 idiomas agrupados por famílias, abordagem que permite ao modelo aprender as nuances e sutilezas de cada

língua, resultando em traduções mais precisas e naturais mesmo em um cenário genérico.

A escolha do modelo de tradução foi realizada de maneira empírica e também com base em resultados previamente apresentados e documentados na literatura, com a testagem de opções que fossem *Open Source* e capazes de gerar traduções da língua inglesa para o português. Os resultados obtidos pelo M2M100 ao acoplá-lo à sequência se mostraram mais coerentes quando comparados empiricamente com o conteúdo das imagens de entrada.

O funcionamento do passo a passo para geração de uma descrição na estratégia apresentada pode ser visualizada no diagrama da Figura 3, no qual uma imagem é recebida como entrada do primeiro modelo, que gera a descrição em língua inglesa, que por sua vez é repassada para o segundo, capaz de gerar uma tradução Inglês-Português.

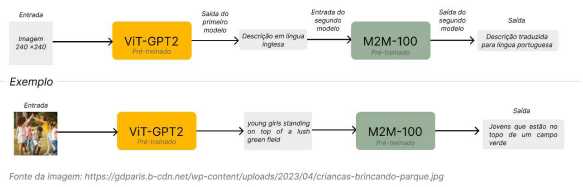

Figura 3: Diagrama da execução do sequência de modelos. O exemplo apresentado na figura foi executado em um caso real de uso da ferramenta.

3.4 Interface

O módulo da interface diz respeito ao *plugin* para navegadores desenvolvido para a interação do usuário com a ferramenta. Essa parte do sistema foi construída com tecnologias clássicas de desenvolvimento web (JavaScript, CSS3 e HTML5), com o propósito de ser de simples desenvolvimento, contribuição e manutenção.

A extensão é subdividida em dois tipos de interface: configuração e atalhos. A interface de configuração apresenta um componente visual em forma de *popup*, que permite ao usuário operar as configurações para o uso da ferramenta, sendo elas ativar ou desativar a operabilidade do *software*, se autenticar na ferramenta e configurar o atalho de disparo para gerar descrições automáticas.

O *popup* pode ser acessado pelo usuário ao clicar no ícone fixado da extensão na barra de tarefas do seu navegador. Após o clique, a primeira impressão do usuário com a interface é a descrita pela Figura 4. No centro da visualização, é possível encontrar o botão de ativação ou desativação da extensão (Figura 4-a). No cabeçalho desse estado existem ainda dois pontos de interação importantes: o botão de *login* (Figura 4-b), que redireciona para o formulário de autenticação na ferramenta, e o botão de configuração (Figura 4-c), indicado pelo ícone de engrenagem, que ao ser clicado redireciona o usuário para a tela de configuração de atalho.

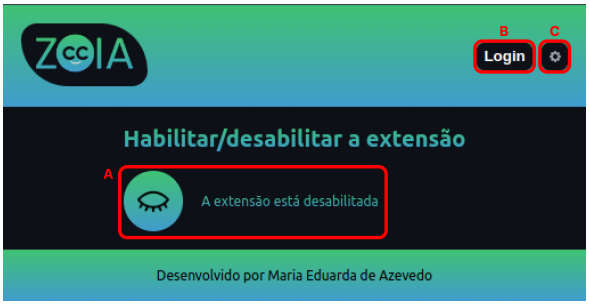

Figura 4: Estado inicial do *popup*. Em destaque é possível notar os pontos de interação, sendo eles a habilitar/desabilitar a extensão (a), ir para página de *login* (b) e ir para página de configurações (c).

Ao interagir com o botão de *login*, o usuário é imediatamente redirecionado para a tela representada pela Figura 5. Esse estado é referente ao formulário de autenticação do usuário na ferramenta. Para realizar se autenticar, o usuário precisa fornecer o endereço do *host* ao qual deseja se conectar (Figura 5-a), o e-mail (Figura 5-b) vinculado a sua conta e o código de acesso (Figura 5-c) fornecido por um administrador do *host*.

A ferramenta, em sua completude, foi desenvolvida para ter a capacidade de *self-hosting*, isto é, qualquer interessando, seja um usuário único ou um núcleo institucional, que deseje manter uma instância do sistema "ZoIA" pode fazê-lo de maneira completamente independente e disponibilizá-lo para o seleto grupo de usuários de interesse. Dessa forma, a interface de interação com o usuário também foi pensada para servir como cliente comum para qualquer *host* mantido por qualquer núcleo de administração permitindo a submissão da URL do servidor a que se deseja conectar, seja esse local ou hospedado em nuvem. Perceba que para tal o usuário deve estar previamente cadastrado no servidor de interesse por um dos administradores da instância do sistema em que se deseja conectar.

Ao finalizar o preenchimento do formulário, o usuário pode se autenticar na API ao clicar no

botão de *login* ((Figura 5-d). Essa ação envia uma requisição de autenticação para o servidor indicado pelo campo "Host", que retorna um token JWT, armazenado no *local storage* do navegador para ser passado no cabeçalho das demais requisições, afim de reconhecer o usuário que está utilizando aquela funcionalidade.

Figura 5: Formulário de *login* na ferramenta. Esse estado do *popup* é ativado após o usuário interagir com o botão de *login* no header do estado inicial da extensão.

Já a interface de atalhos consiste em um conjunto de *scripts* executados a partir da chamada de um atalho de teclado, disparado pelo usuário. Esses *scripts* são responsáveis por varrerem o conteúdo das páginas em busca da imagem selecionada, enviar uma requisição nos moldes do *endpoint* principal da API para gerar uma descrição alternativa e manipular o retorno, de maneira a injetar o resultado como texto alternativo do dado no corpo do HTML.

O atalho de disparo da requisição pode ser configurado pelo usuário no estado de configuração do *popup*, que pode ser acessado ao clicar no ícone da engrenagem (Figura 4-c). Ao acessar essa funcionalidade, é possível visualizar o estado da Figura 6. Nessa tela, o usuário é capaz de configurar uma sequência de teclas que será utilizada como atalho de disparo. Ao clicar em uma tecla, automaticamente o visor (Figura 6-a) irá imprimir o código da mesma junto a combinação do atalho formado. Caso o usuário deseje refazer uma determinada combinação, basta que o mesmo clique na tecla de "Backspace", apagando assim a última tecla registrada. Assim que finalizada a configuração, o usuário pode clicar no botão "Salvar" (Figura 6-b) para persistir a sua nova escolha de atalhos e imediatamente utilizar a nova combinação para disparar as requisições de geração de texto alternativo.

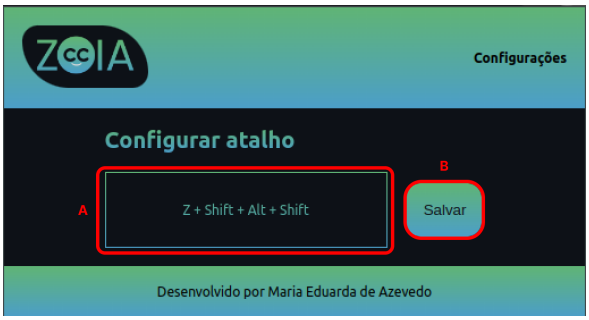

Figura 6: Tela de configuração de atalho de disparo da ferramenta. Esse estado pode ser acessado após o usuário clicar no ícone da engrenagem da tela inicial do *popup*.

3.5 Uso da Ferramenta

Com base nas partes descritas nas subseções anteriores, a sequência de ações que possibilita o uso da ferramenta é discutida a seguir. Um diagrama de atividades do fluxo principal de uso da ferramenta pode ser encontrado no Apêndice A.

Inicialmente, é necessário que um usuário com poder de cadastro (ADMIN ou SUPERADMIN) adicione o usuário, independente de seu papel, ao sistema. Após isso, a conta e o código de acesso são automaticamente criados para aquele cliente e devem ser repassados a ele pelos administradores, por meio de um contato externo, não vinculado à ferramenta.

Em paralelo o usuário pode instalar a extensão em seu navegador com uma *engine* do Google Chrome. Ao receber suas credenciais via o contato com o administrador, o mesmo poderá fazer *login* no *popup* da interface de configurações e se autenticar no *host* desejado com sua conta.

Uma vez ativada a atividade da extensão, para que os *scripts* da interface de atalhos possam escutar as ações de teclado desempenhadas sobre as páginas web, o usuário pode disparar o atalho para geração de uma descrição para uma imagem em foco na página. Após disparado, um *script* varre o website em busca do componente em foco e extrai o dado ali presente, o enviando via requisição HTTP para a API mantida pelo servidor cadastrado. Quando a resposta chega à extensão, um outro *script* é disparado para realizar a injeção da descrição no DOM da página, o colocando em

substituição do valor presente no atributo “alt” da *tag img* e também renderizando uma tarja com a descrição na posição superior à imagem (Figura 7).

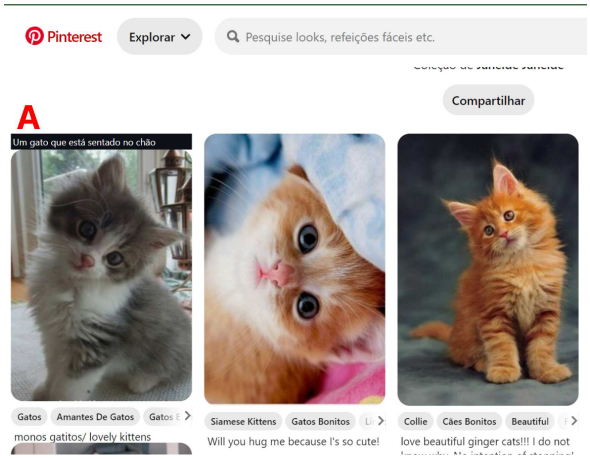

Figura 7: Exemplo de funcionamento visual da ferramenta na plataforma de imagens *Pinterest*.

Na Figura 8 é possível observar o efeito da injeção do *script* no HTML da página (Figura 8-a) ao inspecionar um elemento de imagem que foi foco de uma requisição (Figura 13 8-b). Isso faz com que os leitores de tela, já utilizados por pessoas portadoras de deficiências visuais, possam fazer a leitura do dado e automaticamente interpretar os dados imagéticos para o usuário final da solução, sem a adição de camadas de interpretadores de texto para áudio por parte do *plugin*.

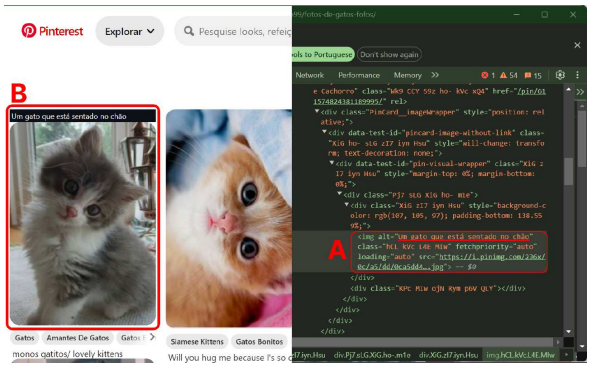

Figura 8: Exemplo de funcionamento dos scripts de injeção das descrições no DOMs da página web.

4 Experimentos

Na tentativa de resolver o problema de descrição automática de imagens para língua portuguesa em uma única fase, isto é, sem a necessidade do uso

Figura 9: Esquema de atuação dos scripts da extensão mediante as ações do usuário em um caso de uso da ferramenta.

de uma estratégia baseada na aplicação de uma sequência de modelos, viu-se uma oportunidade para realização de alguns experimentos. Tomando como base o processo de treinamento do ViT-GPT2 utilizado na sequência integrada à solução da ferramenta, uma replicação do experimento foi realizada com uma base de dados nativamente construída para descrições em português, com o objetivo de comparar os resultados por meio de um conjunto de métricas adequadas e avaliar a viabilidade de investimento e pesquisas voltadas para estratégias desse tipo.

4.1 Base de Dados

Para realização dos experimentos, a base de dados #PraCegoVer (Castro et al. (2022)) foi utilizada para treinar, validar e testar os resultados obtidos. Tal conjunto foi projetado por pesquisadores do Instituto de Computação da Universidade Estadual de Campinas (Unicamp), contendo uma série de imagens e suas respectivas descrições, em língua portuguesa, geradas por seres humanos, extraídas diretamente da rede social *Instagram* por meio de *web scraping*.

O *scraping* automatizado dos dados para a montagem da base foi possível por conta da aderência de instituições de diversos setores à campanha online “#PraCegoVer”, que inspira também o nome do conjunto. Essa iniciativa foi lançada no domínio das redes sociais no ano de 2012 pela então coordenadora de Educação Especial do estado da Bahia, Patrícia Silva de Jesus, especialista em acessibilidade para pessoas com deficiência visual. A ideia, que ainda perdura nos dias de hoje, sendo mantida pelos seus adeptos, diz respeito a descrever os conteúdos visuais postados nas redes, utilizando o espaço das legendas e comentários para tal e sinalizando com a *hashtag* correspondente. Dessa maneira, os autores do trabalho projetaram um *script* para coletar postagens do *Insta-*

gram que possuem a sinalização da *tag* da campanha em seu conteúdo textual.

Os autores aplicaram ainda uma série de processamentos para evitar duplicações dos dados e estruturar a base de maneira semelhante à MS COCO, utilizando opções alternativas de descrições para cada imagem, isto é, um mesmo dado pode possuir mais de uma descrição válida associada a ele no conjunto, o que contribui para a variabilidade e também para o aumento de dados da base. No final da coleta, esses foram capazes de coletar mais de 533 mil postagens de mais de 14 mil perfis distintos da rede social, e após o pós-processamento cerca de 230.400 imagens foram mantidas.

Para realizar análises comparativas, os pesquisadores dividiram a base de dados em subconjuntos de tamanhos distintos, de maneira a preservar as características do conjunto total para as diferentes dimensões. Os subconjuntos foram nomeados como #PraCegoVer-63k, contendo cerca de 63 mil exemplares divididos em treino, teste e validação, o #PraCegoVer-173k, com 173 mil exemplos e, em atualização mais recente, o #PraCegoVer-400k com 400 mil imagens e suas respectivas descrições. Neste experimento foi utilizado o subconjunto #PraCegoVer-63k, dados os desafios relacionados às limitações de *hardware* para realização do treinamento e do armazenamento da base.

As imagens possuem em totalidade compressão *jpg* e são referenciadas por meio de um identificador único, que as nomeia. Esse identificador também associa as imagens a suas descrições, por meio de uma estruturação de arquivos *JSON*, exemplificado na Figura 10. Essa maneira de estruturar a base de dados tem inspiração em outros conjuntos de mesmo propósito, sendo comum de ser observado na literatura (Lin et al. (2014), Young et al. (2014)).

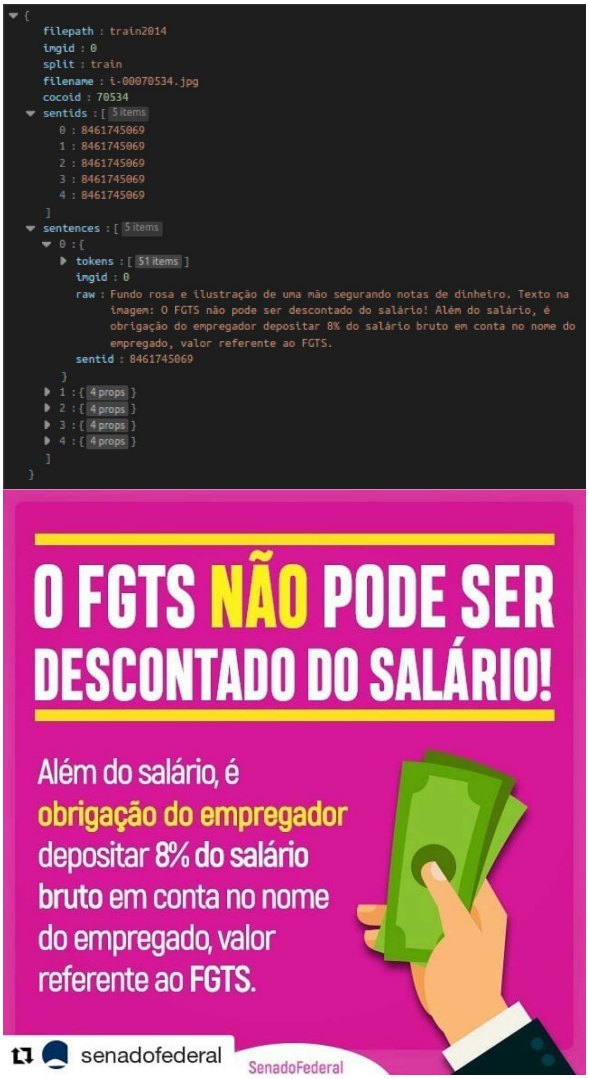

Figura 10: Exemplo de estruturação dos dados em JSON para uma imagem e suas respectivas descrições.

4.2 Metodologia

Inicialmente, para o desenvolvimento dos experimentos foi necessário adquirir a base de dados de seu repositório na plataforma Zenodo (dos Santos et al. (2023)). Para tal, houve a necessidade de baixar uma série de partições *tar.gz* (*tar.gz.part**) e uni-los em um único arquivo para então realizar a descompactação completa da base. Essa etapa demandou cerca de um dia para o download de todos os componentes do arquivo final e a sua descompactação.

Após a extração das imagens e dos arquivos de descrição, surgiu a necessidade de os armazenar em um repositório na nuvem, para facilitar o seu acesso em máquinas distintas, incluindo eventuais máquinas virtuais que viriam a ser utilizadas para a execução dos experimentos de treinamento. Para isso, foi escolhido o serviço de armazenamento de arquivos Google Drive para estocar as imagens e suas descrições. O trabalho manual de realizar o *upload* de todos os dados se mostrou praticamente inviável, dado o tamanho do conjunto de dados, sendo necessário o desenvolvimento de um *script* que, ao se conectar com a conta na plataforma, realizou o carregamento de maneira inteiramente automatizada. Esse processo foi realizado no intervalo de uma semana, na qual o *script* ficou em

operação em uma máquina local realizando o envio dos dados da base para um diretório privado da nuvem.

Em paralelo, a mesma arquitetura capaz de realizar descrições de imagens utilizada na construção da solução da sequência (ViT-GPT2) foi preparada para receber um treinamento de ajuste com os dados do conjunto de dados em português. Dado o contexto das amostras de descrição em língua portuguesa, foi necessário realizar uma modificação referente ao modelo pré-treinado, acoplando no lugar do GPT-2 para língua inglesa uma outra versão da mesma arquitetura, quantizada e pré-treinada em português (Guillou (2020)), mantendo o ViT original como codificador. Todos os modelos utilizados nos experimentos e no desenvolvimento foram adquiridos por meio de código aberto na plataforma Hugging Face[10], de mesmo modo que a biblioteca *transformers*[11] utilizada como interface para uso e manipulação das arquiteturas.

Após a aquisição completa dos dados, o subconjunto #PraCegoVer-63k foi isolado e pré-processado para facilitar sua manipulação. Antes do treinamento, as imagens foram redimensionadas para uma resolução de 224x224 pixels, tamanho de entrada requerido pela arquitetura. Além disso, os dados textuais foram submetidos a uma limpeza e normalização, envolvendo a remoção de caracteres especiais e a conversão das descrições para letras minúsculas, além da adequação das estruturas para serem passadas como entrada do modelo nos padrões da biblioteca utilizada.

Na configuração de treinamento do modelo, foi habilitada a opção *"predict_with_generate"*, permitindo a geração de previsões durante o treinamento. A estratégia de avaliação foi configurada para ser executada a cada época, utilizando como métrica a estratégia ROUGE (Lin (2004)). Quanto ao tamanho do *batch* de treinamento e avaliação por dispositivo, optou-se por um tamanho de lote igual a 4. Por fim, foi especificado um diretório de saída para armazenar os resultados do treinamento diretamente vinculada à conta do Google Drive. Esses argumentos foram selecionados visando a replicação do experimento realizado pela versão pré-treinada da arquitetura utilizada previamente na estratégia de sequência (Kumar (2022)), de maneira que a comparação dos resultados pudesse ser mais fidedigna.

Para a realização do treinamento, foi utilizado o ambiente virtual do Google Colab Pro[12], sendo necessária uma assinatura para sua utilização. Os experimentos foram executados em uma GPU Tesla V100, de 16GB de memória, em uma máquina virtual com 201.2GB de disco e 51GB de memória RAM.

Com a adequação dos dados, a aquisição e modificações dos modelos necessários e o ambiente configurado, o experimento foi devidamente executado, utilizando três épocas de treinamento, tal qual o experimento base levado em consideração. O modelo resultante foi automaticamente sendo salvo em um diretório apontado para o Google Drive e, após finalizado o treinamento, esse foi baixado para a execução da avaliação no conjunto de testes.

Para realizar a comparação das estratégias, predições para imagens do conjunto de teste foram geradas a partir das duas abordagens, isto é, para as mesmas entradas, gerou-se saídas utilizando a sequência de modelos e o modelo ajustado para descrições em português. Mais uma vez por conta das limitações de ambiente e hardware, não foi possível utilizar toda a partição de testes do subconjunto utilizado, sendo reservado uma parcela de 2132 imagens para a avaliação e comparação dos resultados. Em cada grupo de predições, calculou-se uma série de métricas, todas baseadas na comparação entre o texto gerado e um conjunto de descrições anotadas por seres humanos, utilizadas com frequência na literatura e descritas nos parágrafos subsequentes.

Algumas das métricas utilizadas foram projetadas para o contexto de tradução de textos entre idiomas distintos, como é o caso das estratégias ROUGE (Lin (2004)), BLEU (Papineni et al. (2002)) e METEOR (Banerjee and Lavie (2005)). Porém, é possível ressignificar a interpretação dessas para o contexto de descrição automática de imagens, sendo também significativas para mensurar a qualidade das estratégias para resolução do problema, dado que esse pode ser interpretado como uma tradução entre dados de formatos diferentes.

A métrica ROUGE foi originalmente projetada não apenas para tradução, mas também avaliação de sumarização. Essa métrica explicita o quão boa é a estratégia de geração com base na revocação,

[10]https://huggingface.co/

[11]https://huggingface.co/docs/transformers/pt/index

[12]https://colab.research.google.com/

isto é, quanto do conteúdo esperado é recuperado no conteúdo sintético. Tal interpretação direta não reflete bem a aplicação desse método no contexto de descrições automáticas, já que não há tentativa direta de recuperação de um conteúdo, porém, podemos interpretar a comparação como sendo o quão bem a descrição sintética conseguiu expressar o conteúdo esperado, tendo como base um conjunto de legendas de referência.

Já a métrica BLEU, quando aplicada ao contexto de descrições automáticas, tem como objetivo mensurar a proximidade entre uma descrição gerada automaticamente e um conjunto de anotações de referência, considerado de alta qualidade, construído por humanos especializados. O escore dessa métrica é composto pelas pontuações de precisão calculadas para segmentos individuais, em geral sentenças, comparando a descrição gerada e o conjunto de anotações de referência. Após computadas, a média dessas pontuações é calculada para os resultados obtidos em todo o *corpus*. A métrica BLEU foi uma das pioneiras na apresentação de alta correlação entre seus resultados e julgamentos humanos de qualidade, o que garantiu a sua popularização na automação de avaliações de tradução e na sua consequente incorporação para a avaliação do problema de descrição automática de imagens.

Com a finalidade de atender alguns pontos falhos da BLEU, a métrica METEOR foi proposta como uma alternativa. Essa pode ser descrita como uma espécie de *F-measure* entre os textos de referência (anotados por humanos) e o texto gerado, já que é baseada na média harmônica entre precisão e revocação, com essa última recebendo maior ponderação. Outros pontos interessantes dessa estratégia com relação às demais é a aplicação de *stemming* aos textos comparados e a capacidade de correspondência de sinônimos, não se prendendo apenas à associação direta de termos.

A métrica CIDEr-D (Vedantam et al. (2015)) foi escolhida por ter sido diretamente projetada para o contexto de descrição automática de imagens. Essa métrica busca avaliar uma descrição sintética com base no consenso humano. A estratégia também aplica normalizações, como *stemming* e lematização dos termos, além de realizar ponderação das palavras por meio de TF-IDF, de maneira a penalizar palavras que possam aparecer mais frequentemente em todo o *corpus* de descrições não sendo consideradas de alta importância para manutenção do contexto esperado com relação aos textos de referência. Por fim, a métrica utiliza a distância do cosseno para realizar a comparação entre descrição sintética e anotações de referência, levando em consideração a precisão e a revocação e combinando os resultados em uma única pontuação.

A Similaridade do Cosseno foi também utilizada como uma métrica base, de maneira a refletir a distância média entre as palavras que ocorrem na descrição automaticamente gerada e as anotações de referência dentro do espaço vetorial do *corpus* da base de dados. Em comparação com as demais métricas, essa não apresenta foco em aspectos como semântica e contexto, apenas na similaridade média com relação às ocorrências dos termos. Para sua computação, foram utilizadas as representações vetoriais (*embeddings*) da LLM BERTimbau (Souza et al. (2020)), versão ajustada do modelo BERT(Devlin et al. (2019) para língua portuguesa.

4.3 Resultados

Nesta seção são apresentados os resultados obtidos na avaliação das duas estratégias abordadas neste trabalho. Os resultados das métricas para a partição de testes reservada podem ser observados nas Tabelas 1 e 2, sendo respectivamente referentes à sequência de modelos e ao ajuste realizado com o treinamento na base de dados #PraCegoVer-63k. O conjunto de métricas utilizado tem por objetivo mensurar o nível de semelhança semântica entre as descrições sintéticas com as geradas por seres humanos.

Em primeira análise, é possível notar que os resultados obtidos para a sequência são inferiores àqueles obtidos para a versão ajustada com a base de dados, o que pode indicar, à primeira vista, uma estratégia promissora nesse sentido.

A métrica BLEU se mostrou significativamente maior quando a quantidade de n-gramas considerados no cálculo é pequena. Pela natureza dessa métrica, esse resultado indica que, para segmentos menores do texto, o modelo ajustado tende a gerar sentenças mais corretas do ponto de vista gramatical e estrutural do que a sequência de modelos. Quando a quantidade de palavras em cada grama aumenta, observamos que para ambas as estratégias o valor da métrica é equivalente em termos práticos, o que indica uma possível dificul-

dade das duas técnicas em manter a coerência gramatical à medida que as descrições crescem.

Já para a métrica ROUGE, considerando sequências de unigramas e bigramas, observa-se que existe uma evidente vantagem do modelo ajustado com relação à abordagem pré-treinada. O mesmo resultado se mostra para o ROUGEL, isto é, considerando a maior subsequência comum entre a predição e o valor de referência, o que indica que ao realizar o ajuste, o modelo foi capaz de gerar descrições que expressam o conteúdo esperado para extensões maiores do texto.

Observando o valor de METEOR para as duas abordagens, mais uma vez o modelo ajustado se sobressaiu. A métrica indicou que o ajuste do modelo com uma base de dados canonicamente em língua portuguesa aumenta os níveis de precisão e *recall* na comparação com descrições reais.

Em adição, houve também um melhor resultado da métrica CIDEr-D. O resultado dessa para o experimento de ajuste mostrou que o modelo apresentou uma tendência maior em gerar descrições que apresente componentes principais da imagem de entrada.

Por fim, a Similaridade do Cosseno, apesar de não ser uma métrica tão expressiva e robusta como as demais, ainda cumpriu com o padrão observado: o modelo ajustado pontuou melhor do que a sequência implantada na API. Os resultados expressaram que, para a versão ajustada, o modelo conseguiu gerar descrições com um vocabulário mais próximo do esperado.

As métricas obtidas nesse experimento se mostraram equivalentes a outros estudos realizados e documentados pelos autores da base #PraCegoVer com outras estratégias de descrições automáticas de imagens também do estado da arte (Castro et al. (2022)). Os valores aqui obtidos pertencem ao intervalo mediado pelo desvio padrão dos resultados nos *benchmarks* experimentados por eles para avaliar o *dataset* proposto.

Com isso, é possível concluir que os resultados mostraram que há uma tendência de melhoria das descrições com base em um contexto real do idioma quando existe um esforço de ajuste de um modelo para a geração de legendas já na língua de destino, sem a necessidade de uma camada de tradução. Apesar disso, em termos práticos, há uma evidência da dificuldade da geração automática de descrições, dado que ambas as estratégias obtiveram resultados relativamente baixos, o que indica que, apesar de performarem de maneira adequada, não generalizam muito bem para contextos muito amplos.

MÉTRICA	**ESCORE**
BLEU1	0.114573
BLEU2	0.009240
BLEU3	0.000851
BLEU4	0.000000
ROUGE1	0.052037
ROUGE2	0.006473
ROUGEL	0.048069
METEOR	0.018625
CIDEr-D	0.024334
Similaridade do Cosseno	0.687693

Tabela 1: Escores das métricas para a estratégia da sequência de modelos integrada à API.

MÉTRICA	**ESCORE**
BLEU1	0.182801
BLEU2	0.024657
BLEU3	0.005267
BLEU4	0.000839
ROUGE1	0.110517
ROUGE2	0.021052
ROUGEL	0.100769
METEOR	0.048889
CIDEr-D	0.049836
Similaridade do Cosseno	0.758465

Tabela 2: Escores das métricas para a estratégia de refinamento do modelo com *dataset* #PraCegoVer-63k.

5 Conclusão

Neste trabalho foram apresentados os desafios ainda presentes para assegurar uma boa qualidade de navegação web por parte de pessoas com deficiência, em especial, deficientes visuais, principalmente pela falta de artifícios de acessibilidade que dizem respeito às descrições e textos alternativos para conteúdos de imagem.

Alavancando técnicas do estado da arte da Inteligência Artificial, este trabalho propôs o "ZoIA": uma ferramenta construída para integrar navegadores web, capaz de se comunicar com um modelo de descrição automática e gerar e injetar no conteúdo do HTML legendas para componen-

tes de imagem a partir do controle do usuário. Para uma solução inicial e menos custosa, uma sequência de modelos foi pensada para realizar a geração de descrições em língua portuguesa, apresentando um módulo capaz de descrever os dados em língua inglesa e um segundo para traduzir para o português.

Com o objetivo de comparar estratégias plausíveis de serem integradas à arquitetura, um experimento de ajuste do modelo de descrição automática para língua portuguesa foi realizado e seus resultados devidamente comparados com a abordagem previamente adotada e integrada à ferramenta. Dos resultados, foi possível concluir que existe um indicativo de que estratégias baseadas na geração direta de legendas para um idioma alvo melhoram os atributos com relação às legendas sintéticas, porém um impasse relacionado à complexidade de resolução do problema ainda se mostra, dadas as baixas métricas obtidas para ambas as abordagens.

Dessa maneira, conclui-se que a proposição de ferramentais de *software* alimentados por Inteligência Artificial podem compor ideias transformadoras para contornar problemas observados em instâncias sociais, como a acessibilidade, objetivo que se buscou alcançar com o projeto da aplicação proposta. Embora o experimento com a abordagem direta de geração de legendas para o idioma alvo tenha mostrado melhorias em relação à geração com tradução, a complexidade relacionada ao problema permanece evidente, refletida nas métricas ainda aquém de cenários ideais. Portanto, é essencial continuar explorando e refinando estratégias para aprimorar a qualidade e a eficácia das soluções que visam contornar os problemas de acessibilidade na web, visando garantir uma experiência de navegação inclusiva e igualitária para todos.

6 Trabalhos futuros

Este trabalho, tanto em termos da ferramenta proposta quanto da condução dos experimentos, foi capaz de destacar contribuições futuras decorrentes de suas abordagens.

Como trabalho futuro inicial, a aplicação de testes e estudos de usabilidade com usuários reais para validação das implementações tomadas, tal qual levantamento de pontos de melhoria para aprimoramento da experiência do usuário, se mostra um passo essencial para a evolução dos estudos de interface para pessoas com deficiência, podendo utilizar o "ZoIA" como objeto de estudo para tal.

Outro aspecto diz respeito à evolução dos experimentos iniciados, desde sua replicação e comparação de resultados para bases de dados em outros idiomas, quanto na melhoria das técnicas utilizadas para o treinamento, visando melhorar os resultados aqui obtidos, tal qual o mascaramento de entidades nas descrições durante o pré-processamento da base e comparação com a reprodução do experimento em outras arquiteturas, tanto na sequência pré-treinada como na versão refinada.

Em suma, este trabalho estabelece uma base de ideias e objetos de estudo para futuras investigações na área de acessibilidade na web que busquem o desenvolvimento e a evolução de soluções mais eficazes e inclusivas. Ao abordar os desafios presentes no uso da internet e na automação de estratégias de geração de descrições de conteúdos de imagem para pessoas com deficiência visual, esperamos que este trabalho inspire e oriente pesquisadores e desenvolvedores a continuarem nos avanços em direção a uma web mais acessível e inclusiva para todos.

Agradecimentos

Expresso em primeiro lugar o meu amor e gratidão à minha mãe, Sandra Alves de Azevedo, por ter se doado tanto pela minha formação educacional e ter me apoiado em toda essa trajetória repleta de desafios. Igualmente agradeço aos meus familiares, meu pai Fábio e minha irmã Mariana, que acompanharam de perto minha evolução acadêmica, e àqueles que não puderam estar aqui em vida para compartilhar de tal felicidade: minha tia, Maria de Fátima Alves de Azevedo, que foi peça fundamental em minha formação como pessoa, e minhas avós Maria das Neves e Sebastiana.

Demonstro meu carinho aos meus amigos de longa data, João Vítor Diniz e Thamyres Lima, por estarem sempre ao meu lado, me apoiando e vibrando pelas conquistas alcançadas, assim como àqueles que fiz na minha trajetória acadêmica, em especial os que compartilharam comigo experiências de trabalho nos laboratórios em que passei e os de encontro em salas de aula (Pedro Lima, Mateus Aires, Emannuelly Melo, Helen Cavalcanti, Carmem Neri, alguns dentre uma extensa lista).

Estendo meu reconhecimento aos meus professores da academia pelo papel desempenhado em minha educação superior, em principal meu orientador, Herman Martins Gomes, pelo apoio ao tema deste trabalho e pela grande contribuição de suas disciplinas em minha formação acadêmica. De igual modo ovaciono meus professores dos ensinos básico e médio, responsáveis por um alicerce de conhecimento que foi capaz de me direcionar por caminhos mais desafiadores sem grandes dificuldades.

Deixo também registrado o meu agradecimento aos pesquisadores Gabriel Oliveira dos Santos, Esther Luna Colombinin e Sandra Avila, da Universidade Estadual de Campinas (UNICAMP), por terem contribuído com este trabalho cedendo acesso à base de dados #PraCegoVer.

Por fim, quero agradecer aos meus animais de estimação, Chico e Lola, que com inocente companhia me proporcionaram felicidade nos momentos em que necessitei de leveza.

Referências

Banerjee, S. and Lavie, A. (2005). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In Goldstein, J., Lavie, A., Lin, C.-Y., and Voss, C., editors, *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Castro, I. et al. (2022). pracegover: A large dataset for image captioning in portuguese. *Data*, 7(2):13.

Chiang, M. F., Cole, R. G., Gupta, S., Kaiser, G. E., and Starren, J. B. (2005). Computer and world wide web accessibility by visually disabled patients: Problems and solutions. *Survey of Ophthalmology*, 50(4):394–405.

Desaussure, F. (1965). *Course In General Linguistics*. McGraw-Hill, Inc., USA, 1 edition.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding.

dos Santos, G. O., Colombini, E. L., and Avila, S. (2023). #pracegover dataset.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021). An image is worth 16x16 words: Transformers for image recognition at scale.

Fan, A., Bhosale, S., Schwenk, H., Ma, Z., El-Kishky, A., Goyal, S., Baines, M., Celebi, O., Wenzek, G., Chaudhary, V., Goyal, N., Birch, T., Liptchinsky, V., Edunov, S., Grave, E., Auli, M., and Joulin, A. (2020). Beyond english-centric multilingual machine translation.

Guillou, P. (2020). Gportuguese-2 (portuguese gpt-2 small): a language model for portuguese text generation (and more nlp tasks...).

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition.

Kumar, A. (2022). The illustrated image captioning using transformers. *ankur3107.github.io*.

LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., and Jackel, L. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1:541–551.

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. page 10.

Lin, T., Maire, M., Belongie, S. J., Bourdev, L. D., Girshick, R. B., Hays, J., Perona, P., Ramanan, D., Doll'a r, P., and Zitnick, C. L. (2014). Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312.

Liu, M., Li, L., Hu, H., Guan, W., and Tian, J. (2020). Image caption generation with dual attention mechanism. *Information Processing Management*, 57(2):102178.

Minaee, S., Mikolov, T., Nikzad, N., Chenaghlu, M., Socher, R., Amatriain, X., and Gao, J. (2024). Large language models: A survey.

NLP Connect (2022). vit-gpt2-image-captioning (revision 0e334c7).

Ondeng, O., Ouma, H., and Akuon, P. (2023). A review of transformer-based approaches for image captioning. *Applied Sciences*, 13(19).

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Ritchie, H., Mathieu, E., Roser, M., and Ortiz-Ospina, E. (2023). Internet. *Our World in Data*. https://ourworldindata.org/internet.

Sharma, H., Agrahari, M., Singh, S. K., Firoj, M., and Mishra, R. K. (2020). Image captioning: A comprehensive survey. In *2020 International Conference on Power Electronics IoT Applications in Renewable Energy and its Control (PARC)*, pages 325–328.

Song, S., Li, X., Li, S., Zhao, S., Yu, J., Ma, J., Mao, X., and Zhang, W. (2023). How to bridge the gap between modalities: A comprehensive survey on multimodal large language model.

Souza, F., Nogueira, R., and Lotufo, R. (2020). BERTimbau: pretrained BERT models for Brazilian Portuguese. In *9th Brazilian Conference on Intelligent Systems, BRACIS, Rio Grande do Sul, Brazil, October 20-23 (to appear)*.

Teixeira, L., Eusébio, C., and Silveiro, A. (2019). Website accessibility of portuguese travel agents. In *2019 14th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–6.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2023). Attention is all you need.

Vedantam, R., Zitnick, C. L., and Parikh, D. (2015). Cider: Consensus-based image description evaluation.

Yaokumah, W., Brown, S., and Amponsah, R. (2015). Accessibility, quality and performance of government portals and ministry web sites: A view using diagnostic tools. In *2015 Annual Global Online Conference on Information and Computer Technology (GOCICT)*, pages 46–50.

Young, P., Lai, A., Hodosh, M., and Hockenmaier, J. (2014). From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.

A Diagramas adicionais

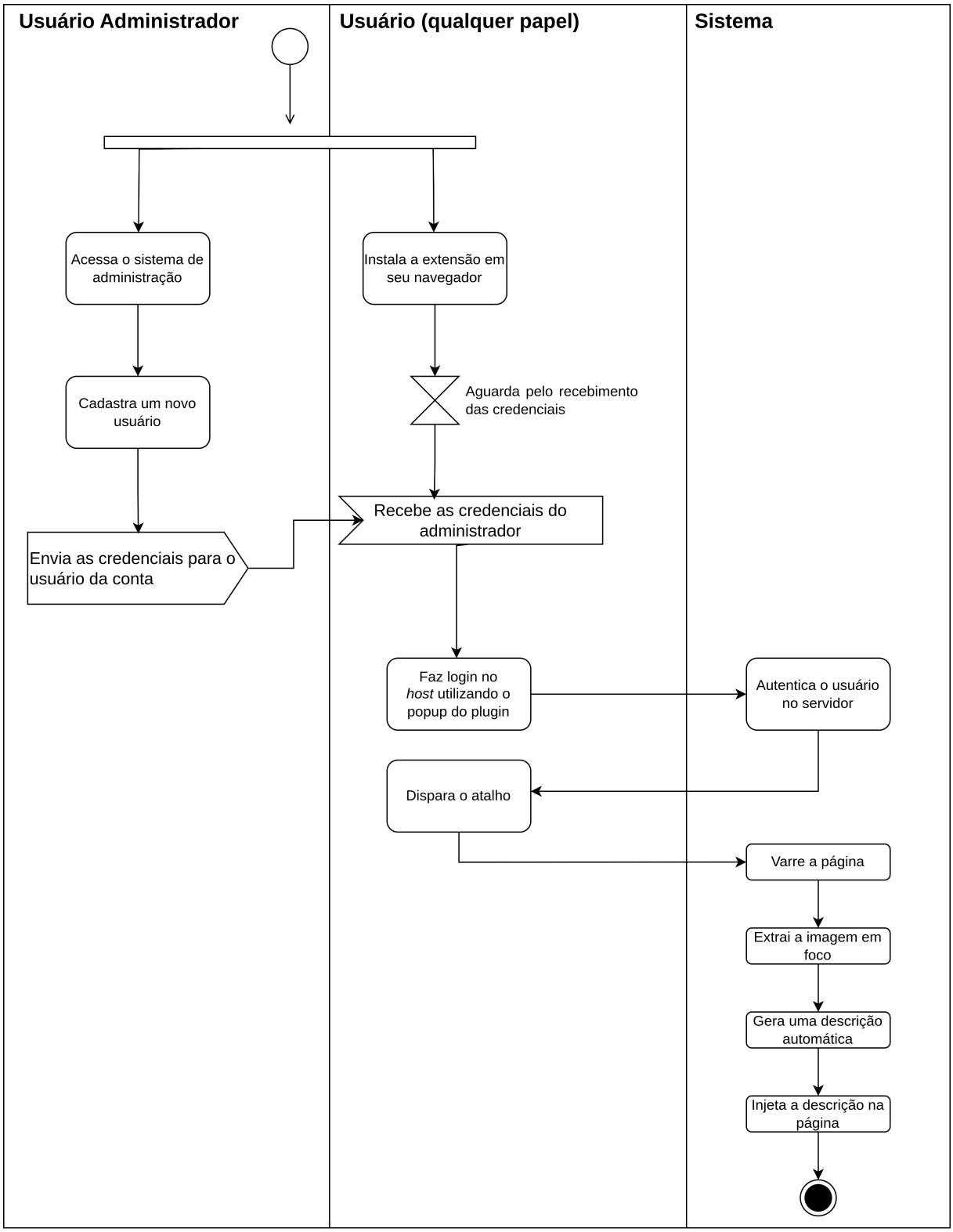

Figura 11: Diagrama de atividades do fluxo principal de uso da ferramenta.