



COLÉGIO TÉCNICO DE CAMPINAS
TÉCNICO DE DESENVOLVIMENTO DE SISTEMAS



VINÍCIUS DOS SANTOS ANDRADE

TraduzAI

A Personalized Solution for Effective and Fluent Translation in Different Contexts



VINÍCIUS DOS SANTOS ANDRADE

TraduzAI

A Personalized Solution for Effective and Fluent Translation in Different Contexts

Relatório do Trabalho de conclusão de curso apresentado ao Curso Técnico de Desenvolvimento de Sistemas do Colégio Técnico de Campinas - UNICAMP como requisito parcial para obtenção título de Técnico em Desenvolvimento de Sistemas.

Orientador: Prof. André Luís dos Reis Gomes de Carvalho

Co-orientador: Prof. Me. Sérgio Luiz Moral Marques

VINÍCIUS DOS SANTOS ANDRADE

TraduzAI

A Personalized Solution for Effective and Fluent Translation in Different Contexts

Trabalho de conclusão de curso apresentado ao
Curso Técnico de Desenvolvimento de Sistemas do
Colégio Técnico de Campinas - UNICAMP como
requisito parcial para obtenção título de Técnico
em Desenvolvimento de Sistemas.

Aprovado em 25/11/2024.

BANCA EXAMINADORA:

Orientador: Prof. André Luís dos Reis Gomes de Carvalho
Colégio Técnico de Campinas - UNICAMP

Coorientador: Prof. Me. Sérgio Luiz Moral Marques
Colégio Técnico de Campinas - UNICAMP

Ao papai do céu.
Aos meus pais, Siloede e Silvana.
Ao Arthur, meu irmão caçula.
Agradecimentos

AGRADECIMENTOS

Resumo

Este projeto tem como objetivo desenvolver um sistema de tradução automática que, além de traduzir de uma língua para outra, seja capaz de interpretar e traduzir jargões técnicos de diversas áreas, como medicina, psicologia, direito, matemática, química, física, programação e filosofia, entre outras ciências exatas e naturais, para uma linguagem cotidiana e acessível a pessoas não especializadas. O sistema utiliza o serviço da AWS Amazon Translate para realizar a tradução automática e a API da OpenAI para simplificar textos técnicos, assegurando que a tradução seja precisa e compreensível. A personalização do sistema permite que os usuários forneçam feedback sobre as traduções e simplificações, contribuindo para a melhoria contínua da qualidade e precisão.

A aplicação conta com uma interface gráfica intuitiva, desenvolvida em Python com Tkinter, onde os usuários podem inserir textos, importar arquivos e selecionar o idioma de destino. O sistema detecta automaticamente o idioma de origem e traduz o conteúdo, preservando a precisão técnica e transformando termos complexos em uma linguagem clara e acessível. Além disso, o projeto disponibiliza recursos de exportação em diferentes formatos, como TXT, PDF e DOCX, permitindo que empresas automatizem o atendimento a clientes em diversos idiomas e integrem a funcionalidade de tradução em suas plataformas.

O sistema avalia a legibilidade dos textos utilizando sete métricas: Índice de Flesch Reading Ease, Grau de Flesch-Kincaid, Índice SMOG, Índice de Coleman-Liau, Índice Automático de Legibilidade (ARI), Pontuação de Dale-Chall e BLEU Score. Essas métricas garantem que a linguagem seja adequada ao público-alvo e contribuem para a análise da clareza e compreensão dos textos traduzidos e simplificados.

O projeto integra o cálculo do BLEU Score (Bilingual Evaluation Understudy) por meio da estratégia de Back-Translation, uma métrica que avalia a qualidade das traduções automáticas ao compará-las com traduções de referência humanas. Essa integração possibilita o monitoramento contínuo da qualidade e a melhoria das traduções de forma objetiva.

A infraestrutura do sistema é baseada em serviços da AWS, como Amazon S3 para armazenamento de dados e Amazon CloudWatch para monitoramento e registro de eventos, garantindo escalabilidade, segurança e alta disponibilidade.

Palavras-chave: Tradução Automática, Jargões Técnicos, Deep Learning, Redes Neurais Recorrentes, Transformer, Amazon Translate, Personalização, Feedback do Usuário, Atendimento ao Cliente, Monitoramento de Qualidade.

Abstract

This project aims to develop an automatic translation system that, in addition to translating from one language to another, can interpret and translate technical jargon from various fields, such as medicine, psychology, law, mathematics, chemistry, physics, programming, and philosophy, among other exact and natural sciences, into everyday language accessible to non-specialists. The system uses the AWS Amazon Translate service for automatic translation and the OpenAI API to simplify technical texts, ensuring that the translation is both accurate and understandable. The system's customization allows users to provide feedback on translations and simplifications, contributing to continuous improvement in quality and accuracy.

The application features an intuitive graphical interface, developed in Python with Tkinter, where users can input text, import files, and select the target language. The system automatically detects the source language and translates the content while maintaining technical precision and transforming complex terms into clear and accessible language. Additionally, the project provides export features in various formats, such as TXT, PDF, and DOCX, enabling companies to automate customer service in multiple languages and integrate translation functionality into their platforms.

The system evaluates the readability of texts using seven metrics: Flesch Reading Ease, Flesch-Kincaid Grade Level, SMOG Index, Coleman-Liau Index, Automated Readability Index (ARI), Dale-Chall Readability Score, and BLEU Score. These metrics ensure that the language is appropriate for the target audience and contribute to the analysis of clarity and comprehension of the translated and simplified texts.

The project integrates the calculation of the BLEU Score (Bilingual Evaluation Understudy) through the Back-Translation strategy, a metric that assesses the quality of automatic translations by comparing them with human reference translations. This integration enables continuous quality monitoring and objective improvement of translations.

The system's infrastructure is based on AWS services, such as Amazon S3 for data storage and Amazon CloudWatch for event monitoring and logging, ensuring scalability, security, and high availability.

Keywords: Automatic Translation, Technical Jargon, Deep Learning, Recurrent Neural Networks, Transformer, Amazon Translate, Personalization, User Feedback, Customer Service, Quality Monitoring.

Sumário

Introdução	9
Desenvolvimento	19
Conclusão	20
Referências	21

Introdução

Sistema de Tradução Automática de Textos com Inteligência Artificial e Integração AWS para Contexto Multilíngue no Atendimento ao Cliente

Descrição do Projeto:

Desenvolva um sistema de tradução automática inteligente capaz de traduzir textos entre múltiplos idiomas, com foco na aplicação em ambientes de atendimento ao cliente multilíngue. O sistema utilizará modelos de Deep Learning, como Redes Neurais Recorrentes (RNNs) ou Transformer (ex: BERT, GPT), juntamente com os serviços da AWS, como Amazon Translate, para fornecer traduções precisas e contextualmente relevantes. O objetivo é melhorar a comunicação e a eficiência em cenários empresariais, onde a tradução rápida e acurada é crucial.

O sistema também incluirá uma interface web interativa que permitirá aos usuários inserir ou copiar textos para tradução instantânea, além de uma API RESTful para integração com outros sistemas de atendimento, como CRMs ou chats de suporte ao cliente. Serão utilizados serviços da AWS como Lambda, API Gateway, e S3 para o armazenamento de logs e dados traduzidos, garantindo escalabilidade e facilidade de manutenção.

Além da tradução automática, o sistema irá monitorar e medir a acurácia das traduções, utilizando técnicas de avaliação de qualidade de tradução e permitirá o ajuste fino dos modelos para melhorar a precisão em linguagens de nicho.

Tecnologias e Ferramentas Utilizadas:

1. Python (para a lógica do sistema e integração)
2. TensorFlow ou PyTorch (para criar e treinar modelos customizados, se necessário)
3. Amazon Translate (como serviço de tradução automática em tempo real)
4. AWS Lambda (para execução serverless de funções)
5. API Gateway (para disponibilizar APIs RESTful)
6. Amazon S3 (para armazenar logs, dados e arquivos traduzidos)
7. Amazon CloudWatch (para monitoramento de performance)
8. AWS IAM (para gerenciamento de permissões e segurança)
9. Docker (para facilitar o desenvolvimento e o ambiente de deploy)
10. React.js ou Flask (para o desenvolvimento da interface web interativa)
11. Jupyter Notebooks (para experimentação com modelos de Machine Learning)

Justificativa e Utilidade:

O projeto se torna útil para empresas que lidam com atendimento ao cliente em múltiplos idiomas e precisam oferecer suporte rápido e eficiente. O sistema pode ser integrado a plataformas de atendimento online, permitindo que os operadores de suporte traduzam automaticamente as mensagens dos clientes e respondam de forma rápida. Além disso, pode ser aplicado em sistemas de e-commerce, permitindo uma experiência de usuário melhorada para clientes de diferentes regiões.

Ao incorporar Deep Learning para ajuste fino e melhorar a precisão das traduções, o sistema também atenderá a setores mais específicos, como jurídico ou técnico, onde traduções literais podem não ser suficientes. Outro benefício será a possibilidade de monitorar as métricas de qualidade de tradução e de permitir ajustes no modelo.

Funcionalidades Principais:

1. **Tradução Multilíngue em Tempo Real:** Usar AWS Translate para fornecer traduções automáticas instantâneas entre múltiplos idiomas.
2. **Ajuste Fino com Deep Learning:** Usar modelos pré-treinados ou ajustar novos modelos utilizando dados de tradução de domínio específico (se necessário).
3. **API RESTful:** Desenvolver uma API REST que permite integração com plataformas de suporte ao cliente (ex: Zendesk, Salesforce).
4. **Interface Web Intuitiva:** Desenvolver uma interface web onde o usuário pode colar textos e obter traduções instantâneas.
5. **Monitoramento de Qualidade de Tradução:** Implementar uma métrica de acurácia e monitoramento (BLEU score, etc.).
6. **Logs e Armazenamento:** Armazenar traduções realizadas e logs de uso no Amazon S3 para análise posterior e auditoria.
7. **Escalabilidade Serverless:** Utilizar AWS Lambda e API Gateway para garantir que o sistema seja escalável sem gestão manual de servidores.
8. **Segurança e Autenticação:** Configurar permissões usando IAM e aplicar autenticação para limitar o acesso à API.
9. **Dashboard de Relatórios:** Criar um painel de controle com métricas de uso e qualidade, utilizando CloudWatch para monitorar o desempenho.

Cronograma com Sub Entregas (2 meses)

Semana 1: Planejamento e Configuração do Ambiente

- **Tarefa 1:** Planejar a arquitetura do sistema e documentar as funcionalidades principais.
- **Tarefa 2:** Configurar ambiente de desenvolvimento local e as credenciais para uso da AWS (IAM, S3).
- **Entrega 1:** Documento de Arquitetura do sistema com um fluxograma da integração entre os componentes (Frontend, Lambda, API Gateway, Translate).

Semana 2: Criação da API e Integração com AWS Translate

- **Tarefa 3:** Criar um endpoint básico de API REST usando API Gateway e Lambda.
- **Tarefa 4:** Integrar o endpoint com Amazon Translate para fornecer traduções simples.
- **Entrega 2:** API de Tradução Funcional, disponível para receber solicitações de tradução.

Semana 3: Interface Web Básica

- **Tarefa 5:** Criar a interface web com um campo para entrada de texto e botão de "Traduzir".
- **Tarefa 6:** Conectar a interface web com a API de tradução via AJAX ou Fetch.
- **Entrega 3:** Interface Web Funcional para teste e verificação de traduções.

Semana 4: Ajuste Fino e Melhorias de Qualidade

- **Tarefa 7:** Implementar métricas de qualidade para avaliar as traduções (ex: BLEU score).
- **Tarefa 8:** Criar logs básicos das traduções e armazená-los no Amazon S3.
- **Entrega 4:** Sistema de Monitoramento de Qualidade integrado e logs funcionando no S3.

Semana 5: Otimização e Personalização do Modelo

- **Tarefa 9:** Explorar ajustes finos nos modelos de tradução, utilizando TensorFlow ou PyTorch se necessário.
- **Tarefa 10:** Aplicar ajustes baseados em domínios específicos (atendimento ao cliente, e-commerce).
- **Entrega 5:** Modelo de Tradução Customizado com melhorias na acurácia para nichos específicos.

Semana 6: Integração com Serviços Externos

- **Tarefa 11:** Criar a documentação da API e configurar segurança de acesso (API Key, autenticação).
- **Tarefa 12:** Desenvolver um script de integração com uma plataforma de atendimento ao cliente.
- **Entrega 6:** API Segura e Integrada com um exemplo de integração com CRM.

Semana 7: Monitoramento e Painel de Controle

- **Tarefa 13:** Configurar monitoramento com CloudWatch e exibir métricas de uso no frontend.
- **Tarefa 14:** Desenvolver um painel de controle para visualizar as métricas (número de traduções, qualidade).
- **Entrega 7:** Dashboard de Monitoramento para o sistema, mostrando desempenho e métricas de qualidade.

Semana 8: Testes Finais e Ajustes

- **Tarefa 15:** Realizar testes de carga e de integração de ponta a ponta.
- **Tarefa 16:** Otimizar o desempenho e corrigir bugs encontrados.
- **Entrega 8:** Sistema Testado e Otimizado, pronto para uso com bom desempenho.

Semana 9: Entrega Final e Documentação

- **Tarefa 17:** Criar documentação final do sistema e dos endpoints da API.
- **Tarefa 18:** Realizar a entrega final do projeto e gerar um relatório completo das métricas e do desempenho do sistema.
- **Entrega 9:** Entrega Final do sistema funcionando com documentação completa

Tradução Automática de Textos com Deep Learning na AWS: Uma Abordagem Personalizada e Adaptativa

Problema: A tradução automática, apesar de amplamente disponível, ainda enfrenta desafios em lidar com nuances linguísticas, jargões específicos de domínio e estilos de escrita individuais. Usuários com necessidades de tradução recorrentes em áreas específicas, como tradução literária, técnica ou jurídica, frequentemente se deparam com traduções genéricas que não atendem às suas expectativas de precisão e fluidez.

Solução: Desenvolver um sistema de tradução automática de textos na AWS que permita a personalização e adaptação do modelo de tradução para domínios e estilos de escrita específicos. O sistema utilizará uma combinação de Amazon Translate para tradução base e modelos de Deep Learning, como Redes Neurais Recorrentes (RNNs) com atenção, treinados em conjuntos de dados customizados para aprimorar a qualidade da tradução em áreas de interesse do usuário.

Funcionalidades:

- 1. Interface Web Intuitiva:** Permitirá aos usuários fazer upload de textos, selecionar o domínio de especialidade (ex: jurídico, médico, literário) e o estilo de escrita desejado (ex: formal, informal).
- 2. Tradução Baseada em Amazon Translate:** O sistema utilizará o Amazon Translate como base para a tradução inicial, garantindo rapidez e cobertura ampla de idiomas.
- 3. Customização com Deep Learning:** Modelos RNN com atenção serão treinados em conjuntos de dados específicos de domínio e estilo, permitindo a adaptação da tradução às nuances e jargões do contexto.
- 4. Sistema de Feedback e Retreinamento:** Os usuários poderão fornecer feedback sobre a qualidade da tradução, que será utilizado para aprimorar os modelos de Deep Learning ao longo do tempo.
- 5. Gerenciamento de Glossários Personalizados:** Usuários poderão criar e gerenciar glossários com traduções específicas para termos e expressões relevantes ao seu domínio.
- 6. API para Integração com Outras Aplicações:** O sistema disponibilizará uma API REST para que outras aplicações possam integrar a funcionalidade de tradução personalizada.

Diferenciais:

1. **Foco na Personalização:** O sistema vai além da tradução genérica, permitindo a adaptação a domínios e estilos de escrita específicos.
2. **Combinação de Amazon Translate e Deep Learning:** Aproveita a robustez do Amazon Translate e o poder de customização do Deep Learning.
3. **Aprendizado Contínuo:** O sistema utiliza o feedback do usuário para aprimorar a qualidade da tradução ao longo do tempo.
4. **Interface Web e API:** Oferece flexibilidade de acesso e integração com outras plataformas.

Roteiro de Desenvolvimento (60 dias / 9 entregas):

Semana 1 (Entrega 1):

Configuração do Ambiente AWS: Criar conta AWS, configurar IAM, S3, Lambda e API Gateway.

Desenvolvimento da Interface Web Básica: Criar interface web com upload de arquivos e seleção de idioma (HTML, CSS, JavaScript).

Integração com Amazon Translate: Implementar a funcionalidade de tradução básica utilizando a API do Amazon Translate.

Semana 2 (Entrega 2):

Coleta e Preparação de Dados: Definir domínios de especialidade e coletar conjuntos de dados para treinamento (ex: textos jurídicos, artigos médicos).

Pré-processamento de Dados: Limpeza, tokenização e formatação dos dados para treinamento dos modelos de Deep Learning.

Escolha e Implementação da Arquitetura RNN: Escolher e implementar a arquitetura RNN com atenção (TensorFlow/PyTorch) para tradução.

Semana 3 (Entrega 3):

Treinamento Inicial do Modelo RNN: Treinar o modelo RNN com o conjunto de dados base para tradução em um domínio específico.

Avaliação do Modelo: Implementar métricas de avaliação da qualidade da tradução (ex: BLEU score).

Integração do Modelo RNN com a Interface Web: Permitir a escolha do modelo de tradução (Amazon Translate ou RNN) na interface.

Semana 4 (Entrega 4):

Implementação do Sistema de Feedback: Criar mecanismo para coleta de feedback do usuário sobre a qualidade da tradução.

Gerenciamento de Glossários: Implementar funcionalidade para criação e gerenciamento de glossários personalizados.

Integração do Glossário com o Modelo RNN: Adaptar o modelo RNN para utilizar o glossário durante a tradução.

Semana 5 (Entrega 5):

Treinamento do Modelo RNN com Feedback: Implementar o processo de retreinamento do modelo RNN com o feedback do usuário.

Aprimoramento da Interface Web: Adicionar funcionalidades para visualizar e editar o glossário, visualizar histórico de traduções e feedback.

Testes de Usabilidade: Realizar testes de usabilidade com usuários para coletar feedback sobre a interface e a qualidade da tradução.

Semana 6 (Entrega 6):

Implementação da API REST: Criar API REST para acesso às funcionalidades de tradução, feedback e gerenciamento de glossários.

Documentação da API: Criar documentação completa da API para facilitar a integração com outras aplicações.

Testes de Integração: Realizar testes de integração da API com uma aplicação exemplo.

Semana 7 (Entrega 7):

Otimização do Desempenho: Analisar e otimizar o desempenho do sistema, focando na velocidade de tradução e no consumo de recursos na AWS.

Implementação de Segurança: Implementar medidas de segurança para proteger os dados dos usuários e o acesso à API.

Testes de Carga: Realizar testes de carga para garantir a escalabilidade do sistema em cenários de alto tráfego.

Semana 8 (Entrega 8):

Preparação para Implantação: Criar scripts para automatizar a implantação do sistema na AWS.

Documentação do Sistema: Criar documentação completa do sistema, incluindo arquitetura, código-fonte, instruções de uso e manutenção.

Apresentação do Projeto: Preparar apresentação do projeto, destacando as funcionalidades, resultados e aprendizados.

Semana 9 (Entrega 9):

Implantação Final na AWS: Implantar o sistema em um ambiente de produção na AWS.

Testes Finais: Realizar testes finais no ambiente de produção para garantir o funcionamento correto do sistema.

Entrega do Projeto: Entregar o projeto completo, incluindo código-fonte, documentação e apresentação.

Desenvolvimento

Conclusão

Referências

TODO: buscar referências sólidas no GitHub

AWS. Full Stack Machine learning on AWS: Language Translation with AWS Amplify and Amazon Translate. [S.I.], 2023. 1 vídeo (12 min). Disponível em: <[link do vídeo]>. Acesso em: [23 de setembro de 2024].

AWS Tutorials. AWS Mini-Project-3: A Translator using AWS Translate and Boto3. [S.I.], 2023. 1 vídeo (8 min). Disponível em: <[link do vídeo]>. Acesso em: [23 de setembro de 2024].

[Nome do Canal/Autor]. Amazon Translate Service with AWS API Gateway and Python Lambda Function. [S.I.], 2023. 1 vídeo (15 min). Disponível em: <[link do vídeo]>. Acesso em: [23 de setembro de 2024].

ALURA. Formação: Técnicas de processamento de linguagem natural. [S.I.], 2023. 1 curso online (58 aulas). Disponível em: <[link do curso]>. Acesso em: [23 de setembro de 2024].

UDEMY. Deep Learning: NLP for Sentiment analysis & Translation 2024. [S.I.], 2023. 1 curso online ([número de aulas/horas]). Disponível em: <[link do curso]>. Acesso em: [23 de setembro de 2024].

UDEMY. Deep Learning: Natural Language Processing with Transformers. [S.I.], 2023. 1 curso online ([número de aulas/horas]). Disponível em: <[link do curso]>. Acesso em: [23 de setembro de 2024].

UDEMY. Voice-Over/Voice Acting happens Automatically - Amazon Polly. [S.I.], 2023. 1 curso online ([número de aulas/horas]). Disponível em: <[link do curso]>. Acesso em: [23 de setembro de 2024].

UDEMY. Translation happens Automatically - Using Amazon Translate. [S.I.], 2023. 1 curso online ([número de aulas/horas]). Disponível em: <[link do curso]>. Acesso em: [23 de setembro de 2024].