Pontifícia Universidade Católica do Rio de Janeiro
Departamento de Informática

Projeto final de programação

Classificador de Trechos Textuais em Construtos da Teoria Social Cognitiva

Wallace Albertini

Orientador: Marcos Kalinowski

Rio de Janeiro

Dezembro de 2024

Sumário

1 Breve descrição

Este programa foi desenvolvido como um classificador de trechos textuais em construtos da Teoria Social Cognitiva. Ele oferece uma solução integrada para pesquisadores e profissionais interessados em realizar análises contextuais com apoio de aprendizado de máquina e recuperação de informações.

1.1 Usuários Primários

Este programa foi projetado principalmente para:

- Pesquisadores e profissionais das áreas de psicologia, educação e análise de comportamento.
- Professores e estudantes interessados em métodos de análise qualitativa, especialmente em níveis de graduação e pós-graduação.

1.2 Natureza do Programa

- Prova de conceito: Este é um sistema funcional que demonstra a integração entre técnicas de machine learning, modelos de linguagem natural e ferramentas de recuperação de informações.
- Ferramenta utilitária: Atualmente, encontra-se em fase inicial de desenvolvimento e pode ser estendido para se tornar uma solução mais robusta.
- Ressalvas:
  - O sistema depende de conexões externas para serviços como a API do OpenAI, o que pode limitar seu uso em ambientes offline.
  - A classificação automática pode apresentar variações dependendo da qualidade dos dados de entrada e do treinamento dos modelos subjacentes.

2 Visão do Projeto

2.1 Cenário Positivo 1:

Julia é uma pesquisadora em psicologia comportamental interessada em categorizar transcrições de entrevistas qualitativas para uma análise temática. Ela utiliza o programa para

carregar um arquivo Excel contendo várias transcrições, pré-processa os dados e inicia o sistema de classificação. Com base na recuperação de informações relevantes e na interação com o modelo GPT, Julia recebe as classificações acompanhadas de justificativas claras. Ela usa essas informações para validar suas hipóteses e, ao final, utiliza as métricas de avaliação para verificar a precisão e consistência das classificações geradas. Julia conclui o estudo com dados bem organizados e insights valiosos para seu artigo científico.

2.2 Cenário Positivo 2:

Marcos, um professor universitário em educação, precisa ensinar aos alunos de graduação como analisar comportamentos observados em contextos educacionais. Ele usa o programa para apresentar exemplos reais de entrevistas e transcrições, permitindo que os alunos entendam como o modelo TSC pode ser aplicado em cenários práticos. Após classificar exemplos, Marcos revisa as respostas com os alunos, destacando erros e acertos do modelo. Ele também utiliza a função de recuperação de contextos relacionados para promover discussões sobre como diferentes fatores afetam o comportamento. Isso enriquece a aula e melhora a compreensão dos alunos sobre o tema.

2.3 Cenário Negativo 1:

Marina é uma estudante de psicologia que tenta usar o programa para analisar dados qualitativos, mas encontra problemas ao carregar o arquivo Excel. O programa retorna uma mensagem de erro indicando que o formato das colunas ou a nomenclatura do arquivo não está correta. Marina tenta corrigir o problema, mas como não possui experiência com manipulação de dados, fica frustrada e desiste temporariamente de usar o programa, optando por buscar ajuda com seu orientador ou um colega.

2.4 Cenário Negativo 2:

Mauro, um aluno de mestrado, utiliza o programa para categorizar exemplos de comportamento observados em uma empresa. Ele espera resultados rápidos e precisos, mas percebe que a classificação automática, em alguns casos, não reflete o contexto esperado, devido à natureza limitada do treinamento do modelo ou à falta de exemplos similares no banco de dados. Apesar de tentar refinar os resultados utilizando a função de recuperação de contextos, Mauro conclui que a ferramenta precisa de uma maior personalização para atender adequadamente às especificidades do seu campo de pesquisa.

3 Especificação de Requisitos Funcionais e Não-Funcionais do Software

3.1 Requisitos Funcionais:

RF1 O sistema deve ser capaz de carregar dados de planilhas Excel, com múltiplas abas, contendo trechos textuais e categorias e armazenar embeddings dos dados para consulta e recuperação de informações relevantes.

RF2 O sistema deve ser capaz de classificar os trechos textuais de acordo com categorias predefinidas da Teoria Social Cognitiva (TSC): Reciprocal Determinism, Observational Learning (Modeling), Reinforcement, Self-efficacy, Outcome Expectations, Behavioral Capability, e Environmental Factors.

3.2 Requisitos Não-Funcionais:

RNF1 Desempenho: O sistema deve ser capaz de realizar a análise em tempo razoável para entradas de tamanho moderado.

RNF2 Escalabilidade: O sistema deve ser projetado para suportar aumento de dados, com capacidade de carregar grandes arquivos Excel e processar múltiplas consultas simultâneas.

RNF3 Usabilidade: A interface de usuário deve ser simples, com feedback claro durante o processo de análise.

RNF4 Segurança: A chave de API para o modelo de linguagem (OpenAI) deve ser gerida de forma segura para evitar exposições indevidas.

4 Descrição ou Modelo de Arquitetura, Dados, Semântica ou Outra Dimensão Relevante do Software

4.1 Arquitetura

O sistema é dividido em vários módulos, cada um com uma responsabilidade específica:

1. Módulo de Carregamento de Dados (DataLoader):
    - Responsável por carregar dados de arquivos Excel.

- Extrai citações e categorias para posterior processamento.

2. Módulo de Pré-processamento (PreProcessor):
   - Responsável por realizar o pré-processamento das citações, incluindo a conversão de texto em embeddings (vetores).
   - Utiliza modelos como o all-MiniLM-L6-v2 para gerar embeddings a partir das citações.
3. Módulo de Recuperação de Dados (Retriever):
   - Utiliza o ChromaDB para armazenar e recuperar os embeddings das citações, realizando consultas para encontrar os dados mais relevantes com base nas entradas do usuário.
4. Módulo de Modelo de Linguagem (OpenAiModel):
   - Faz a consulta ao modelo GPT-4 da OpenAI para gerar respostas baseadas nos dados recuperados.
5. Módulo de Prompter:
   - Cria as mensagens de prompt que são enviadas ao modelo GPT-4 para garantir que ele produza uma resposta relevante e formatada corretamente.
6. Interface de Usuário (View):
   - Fornece a interface com o usuário utilizando o Streamlit, permitindo a interação do usuário com o sistema de forma simples e intuitiva.

4.2 Dados

- Entrada: Dados no formato de planilhas Excel, contendo trechos textuais (quotes) e suas respectivas categorias.
- Saída: A classificação das citações em uma das sete categorias da Teoria Social Cognitiva, com uma justificativa fornecida pelo modelo.

4.3 Descrição ou Modelo Funcional do Software:

O software segue um modelo funcional baseado no processo de consulta a um sistema de recuperação de dados e geração de resposta:

1. Carregamento de Dados: O Controller carrega os dados de um arquivo Excel, iniciando a base de dados para processamento.

2. Pré-processamento: Os trechos textuais e suas classificações são processadas e convertidas em embeddings que serão usados para consultas de similaridade.

3. Recuperação de Dados: Quando o usuário faz uma consulta, o sistema recupera os dados mais relevantes com base nos embeddings armazenados.

4. Classificação e Resposta: O modelo GPT-4 gera uma resposta classificada com base nos dados recuperados, e a interface exibe a resposta ao usuário, junto com uma justificativa.

4.4 Sobre o Código:

- Linguagem: Python 3.x.
- Bibliotecas:
  - Streamlit: Utilizada para criar a interface de usuário interativa.
  - Pandas: Para manipulação de dados em formato DataFrame, especialmente no carregamento e processamento das planilhas.
  - ChromaDB: Para armazenamento e consulta de embeddings em um banco de dados vetorial.
  - OpenAI: Para integração com o modelo GPT-4, utilizado para gerar respostas às consultas.
  - Langchain: Utilizado para criação e manipulação de mensagens de prompt para o modelo de linguagem.

4.5 Estratégia de Comentários em Linha:

- Os comentários são utilizados de forma a descrever a função de cada bloco de código.
- Cada classe e método possuem descrições breves sobre sua responsabilidade no fluxo do programa.

5 Manual de Utilização para Usuários Contemplados

5.1 Tarefa 1: Carregar Dados de Planilhas Excel

5.1.1 Guia de Instruções:

Para carregar os dados de uma planilha Excel, siga os seguintes passos:

1. Abra o sistema e insira o arquivo Excel com as planilhas desejadas.
2. Certifique-se de que o arquivo Excel contém as colunas com as citações e as categorias adequadas.
3. No campo apropriado, insira o nome do arquivo Excel (por exemplo, Análise Temática de Conteúdo.xlsx) e as planilhas a serem processadas.
4. O sistema irá automaticamente processar as planilhas e extrair as citações e categorias.

Alternativa: Caso o sistema não reconheça corretamente as planilhas ou colunas, verifique se os nomes estão corretamente especificados e se o formato das colunas corresponde ao esperado.

5.1.2 Exceções ou Potenciais Problemas:

- Se o arquivo Excel não estiver no formato adequado (colunas de citações e categorias ausentes ou mal formatadas):
  - O sistema exibirá um erro informando que a leitura da planilha falhou. Verifique o formato das colunas e reenvie o arquivo.
  - É porque: O arquivo Excel não está estruturado corretamente, conforme esperado pelo sistema. As colunas devem seguir o padrão de 'Quote' e 'Categoria'.

5.2 Tarefa 2: Inserir um Trecho Textual para Classificação

5.2.1 Guia de Instruções:

Para inserir uma citação e analisar sua classificação, siga os seguintes passos:

1. No painel principal do sistema, localize o campo de entrada onde você pode digitar uma nova citação.

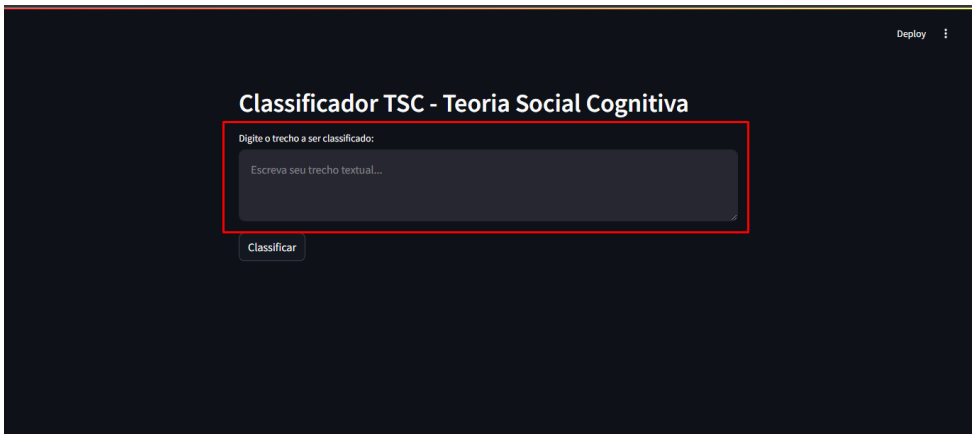

Figura 1: Tela sem trecho textual

2. Insira a citação desejada no campo de texto.

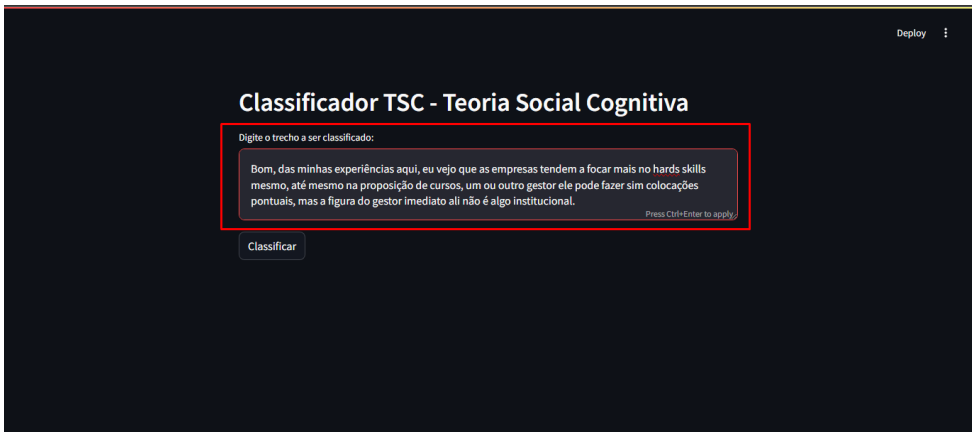

Figura 2: Texto a ser classificado

3. Clique no botão Classificar para iniciar o processo de classificação.

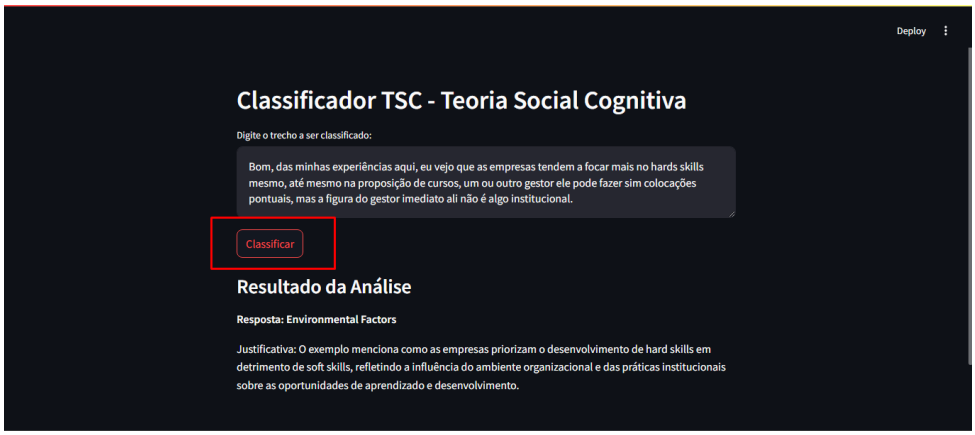

Figura 3: Botão classificar

4. O sistema exibirá a classificação do trecho textual na forma de uma categoria (por exemplo, Self-efficacy) e uma justificativa.

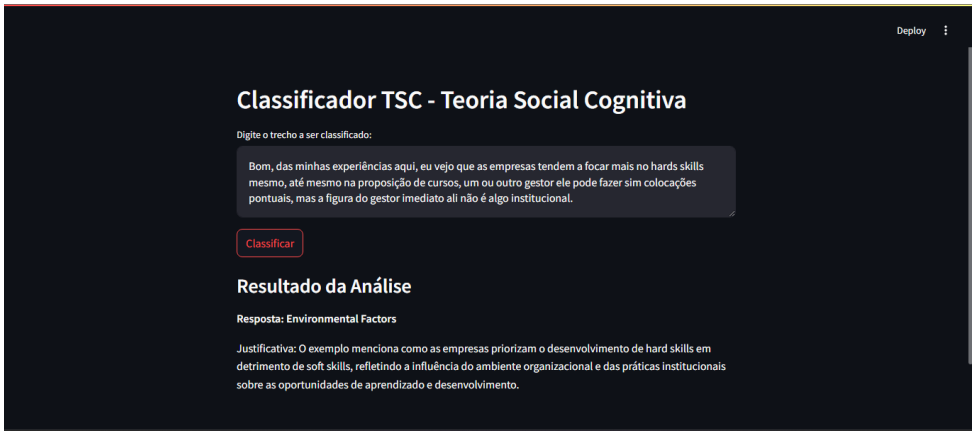

Figura 4: Trecho classificado

Alternativa: Caso você deseje inserir mais de uma citação para análise, o sistema permite o processamento de várias citações de uma vez. Basta repetir os passos para cada citação.

5.2.2 Exceções ou Potenciais Problemas:

- Se a citação inserida for muito curta ou incompleta:
  - O sistema pode não conseguir gerar uma análise precisa. Insira uma citação mais completa ou detalhada para melhores resultados.
  - É porque: O modelo de linguagem precisa de contexto suficiente para realizar a classificação corretamente.

- Se o sistema não conseguir classificar a citação:
  - O sistema pode informar que a citação não se encaixa nas categorias predefinidas. Tente reformular ou fornecer mais contexto para a citação.
  - É porque: O modelo de linguagem pode não ter informações suficientes para identificar a categoria com precisão.

6 Resultados Obtidos nos Testes

A seguir, são apresentados os resultados dos testes realizados utilizando o modelo GPT-4 Mini em diferentes cenários e configurações.

6.1 Resultados de Acurácia e F1 Score

Os valores de acurácia e F1 Score foram obtidos em diversos experimentos utilizando abordagens como RAG (Retrieval-Augmented Generation) e 1-shot, além de variações nas entradas de dados, como a limitação a apenas uma categoria no Excel. Os resultados estão resumidos abaixo:

- Acurácia: 0.5519, cenário: modelo GPT-4 Mini, limitando entradas no Excel a apenas uma categoria, RAG.
- Acurácia: 0.5221, cenário: modelo GPT-4 Mini, limitando entradas no Excel a apenas uma categoria, RAG.
- F1 Score: 0.4346, cenário: modelo GPT-4 Mini, com duas sheets e entradas limitadas no Excel a apenas uma categoria, RAG.
- F1 Score: 0.4175, cenário: modelo GPT-4 Mini, com duas sheets, entradas limitadas a uma categoria e adicionando significado às categorias no prompter do RAG, RAG.
- F1 Score: 0.3487, cenário: modelo GPT-4 Mini, com duas sheets e entradas limitadas no Excel a apenas uma categoria, RAG.
- F1 Score: 0.3312, cenário: modelo GPT-4 Mini, com duas sheets e adicionando significado às categorias no prompter do RAG, RAG.
- F1 Score: 0.3000, cenário: modelo GPT-4 Mini, com duas sheets, limitando entradas a apenas uma categoria, 1-shot.
- Acurácia: 0.4458, cenário: modelo GPT-4 Mini, RAG.

Esses resultados ilustram o impacto das diferentes configurações no desempenho do modelo, evidenciando o trade-off entre acurácia e F1 Score dependendo da abordagem utilizada.

7 Conclusão

Este trabalho apresentou o desenvolvimento de um sistema com o objetivo de classificar trechos textuais em construtos da Teoria Social Cognitiva (TSC) de maneira automatizada. O projeto envolveu a integração de diversas tecnologias e metodologias modernas, como o uso de modelos de linguagem natural avançados, vetorização de textos com embeddings, armazenamento e recuperação eficiente de dados utilizando ChromaDB, e a construção de uma interface interativa para facilitar a utilização por usuários finais.

Durante o desenvolvimento, destacaram-se os seguintes aspectos:

1. Automatização e Precisão: O uso de modelos de linguagem como o GPT-4 e ferramentas como LangChain permitiu automatizar a análise textual com elevado nível de precisão, demonstrando a viabilidade da abordagem para cenários reais.

2. Modularidade e Escalabilidade: A arquitetura modular adotada facilita a reutilização e a expansão do sistema para novas categorias de análise ou diferentes domínios.

3. Interatividade e Acessibilidade: A criação de uma interface amigável utilizando Streamlit possibilita que usuários com diferentes níveis de conhecimento técnico utilizem a ferramenta de forma intuitiva.

4. Documentação e Usabilidade: Foi produzida uma documentação completa, incluindo manuais técnicos e de usuário, garantindo que o sistema seja facilmente compreendido e adaptado por futuros desenvolvedores e usuários.