

Proposta de Projeto: Desenvolvimento Integrado para Soluções Distribuídas com Aplicação de Aprendizado de Máquina

Douglas Henrique Siqueira Abreu

Objetivo

Desenvolver uma extensão inovadora para navegadores que capitalize a potência do aprendizado de máquina em ambientes distribuídos. Nossa abordagem visa aprimorar a experiência dos usuários em aplicativos de mensagens instantâneas, como o Whatsapp, por meio de sugestões de respostas inteligentes.

Contextualização

Essa extensão utiliza algoritmos de processamento de linguagem natural, permitindo uma compreensão do contexto das mensagens em tempo real. Ao aplicar os conhecimentos adquiridos em sala de aula, este projeto não apenas oferece uma solução prática para otimizar as interações no aplicativo de troca de mensagens, mas também aprofunda a compreensão da equipe dos desafios reais enfrentados ao empregar o aprendizado de máquina em cenários distribuídos do mundo real.

Orientação e Gestão do Projeto

O componente curricular do Projeto Integrador 6 objetiva mobilizar e integrar competências para o desenvolvimento de projetos em equipe. Espera-se que estas equipes lidem com design arquitetural, programação de sistemas distribuídos para aplicações de sistemas inteligentes e/ou aplicativos full stack, e abranjam a sistematização dos processos pertinentes.

O professor orientador assume o papel de diretor das equipes. Nesta função, o orientador não fornece assistência técnica, mas guia as equipes sobre como alcançar os resultados desejados. Ele acompanhará a gestão do projeto, esclarecendo dúvidas sobre o escopo e orientando os grupos sobre a direção a seguir. Não é papel do professor orientador da disciplina programar com a equipe ou resolver problemas técnicos. As equipes devem ser autogerenciáveis em termos técnicos.

1 Definição do Problema

1. **Identificação:** Com o aumento do ritmo do dia-a-dia e da chegada de informações e troca de dados, surge a necessidade das pessoas conversarem de

maneira eficiente e facilitada.

2. **Justificativa:** A extensão é capaz de analisar as mensagens dos clientes em tempo real sugerindo respostas adequadas de forma imediata. Ao utilizar a solução, o usuário irá ter uma maior agilidade nas repostas do seu cotidiano, facilitando na comunicação entre pessoas e economizando tempo com a troca de mensagens rápidas. Além disso a ferramenta irá ajudar indivíduos que possuam limitações, sejam elas físicas ou cognitivas, pois as sugestões estariam disponíveis a um clique.

2 Design Arquitetura

1. **Estruturação:** A solução foi dividida em três componentes principais:
 - (a) Extensão (Chrome): Responsável por pegar a informação do whatsapp do usuário e mandar via HTTP para o backend. Além disso, também tem a responsabilidade de receber a mensagem (response) da API e disponibilizar para o user a opção de fazer a cópia do texto para colar no campo de digitação do app de mensagens.
 - (b) API: Receberá a mensagem e passará os dados para o terceiro módulo, além de receber a mensagem do terceiro módulo e enviar novamente a extensão, funcionando assim, como uma ponte entre a extensão e o componente (RASA).
 - (c) IA: Irá processar os dados em linguagem natural e retornar para o segundo módulo (API).
2. **Front-end e Back-end:**
 - (a) Front-end: O Front-end seria a extensão instalada no browser (Chrome), onde nela acontece a interação do usuário com o sistema. A extensão é feita em HTML, CSS e JavaScript (sem framework específico).
 - (b) Back-end: O Back-end está separado em duas principais partes, a API que será implementada com o framework de python FastAPI e o componente em python chamado RASA. Ele será responsável por realizar o processamento de linguagem natural.

3 Desenvolvimento de Sistemas Distribuídos

Sistema foi pensado em dois subsistemas (distribuídos), sendo um deles rodando no browser do usuário e o outro sendo um micro serviço que a princípio rodará localmente (no mesmo computador do usuário), mas será totalmente modulável, podendo assim ser colocados em sistemas de núvens ou servidores locais já que o próprio micro serviço ficará recebendo chamadas via web.

A vantagem de possuir sistema separados é que podemos utilizar as melhores ferramentas (linguagens, bibliotecas, etc) para cada ambiente.

4 Integração de Sistemas Inteligentes

1. **Seleção de Modelos:** A biblioteca RASA NLU (Natural Language Understanding), que é um componente chave do RASA, foi utilizada durante o desenvolvimento. O treinamento do modelo consistiu em fornecermos as mensagens e as categorias onde elas se encaixariam com um conjunto de respostas. Ou seja, um modelo de aprendizado de máquina supervisionado.
2. **Treinamento e Teste:** A coleta de dados basicamente se resumiu em troca de mensagens do integrantes da equipe, criando assim um banco de dados com perguntas e respostas relacionadas e orgânicas, já que foram geradas de conversas reais. Estes foram os dados utilizados para treinar o RASA a identificar o contexto de mensagens recebidas de uma conversa, e sugerir possíveis respostas para elas.

5 Implementação Full Stack

1. **Tecnologias:** As tecnologias elegidas para a construção do projeto foram:
 - Python 3: Devido a grande comunidade de ciência de dados e inteligência artificial, esta ferramenta se torna muito útil, trazendo facilidades como flexibilidade, legibilidade e bibliotecas voltadas ao data science.
 - Fast API: O FastAPI é ideal para construir APIs RESTful que precisam ser altamente performáticas. Ele lida bem com solicitações concorrentes e permite que você crie endpoints que processam e respondem a dados rapidamente.
 - HTML CSS E JAVASCRIPT: Juntas, as três tecnologias voltadas para o desenvolvimento web são a base de nossa extensão oferecendo uma interface simples e amigável ao usuário.
 - VsCode: Utilizada como nossa principal IDE, devido a suas facilidades para a escrita de código.
 - RASA: É uma plataforma open-source em Python para o desenvolvimento de chatbots e assistentes virtuais.
2. **Fluxo de Dados:** Detalhe como os dados fluirão entre o frontend, o backend e os sistemas de armazenamento, considerando também a interação com os modelos de aprendizado de máquina.
 - (a) Usuário irá interagir com uma interface desenvolvida para web.
 - (b) A interface sera uma extensão desenvolvida para navegadores web e enviará os dados coletados da página do usuário para uma API
 - (c) Após a API se comunicar com o backend, threads serão iniciadas para que seja implementada a programação paralela e a IA retorne suas respostas ao usuário.

6 Padrões de Projeto

O CleanCode foi um dos padrões de projeto que trouxemos para o nosso PI, com o intuito de aumentar legibilidade, facilidade na manutenção e melhoria da comunicação do time.

O SOLID foi outro caso, onde mesmo que não tenhamos usado Orientação a Objetos, focamos ao máximo em aplicar seus princípios. Como exemplo, o SRP (Princípio de Responsabilidade Unica), onde é estabelecido que todas as classes devem ter responsabilidades bem específicas. Como comentado, não foi utilizado POO, e por consequência classes, mas mesmo assim esse conceito foi aplicado nas funções da extensão.

Conclusão

A tomada de decisões bem fundamentadas foi uma constante ao longo do projeto. Desde a escolha da ideia do projeto, até a definição das estratégias de IA para a geração de respostas sugeridas, cada passo foi embasado em análises criteriosas e discussões colaborativas. O resultado do projeto, representa a capacidade da equipe de trabalhar em conjunto em prol de um único objetivo, exercitando não apenas qualidades técnicas, mais também soft skills desejadas por empresas do mercado de trabalho.