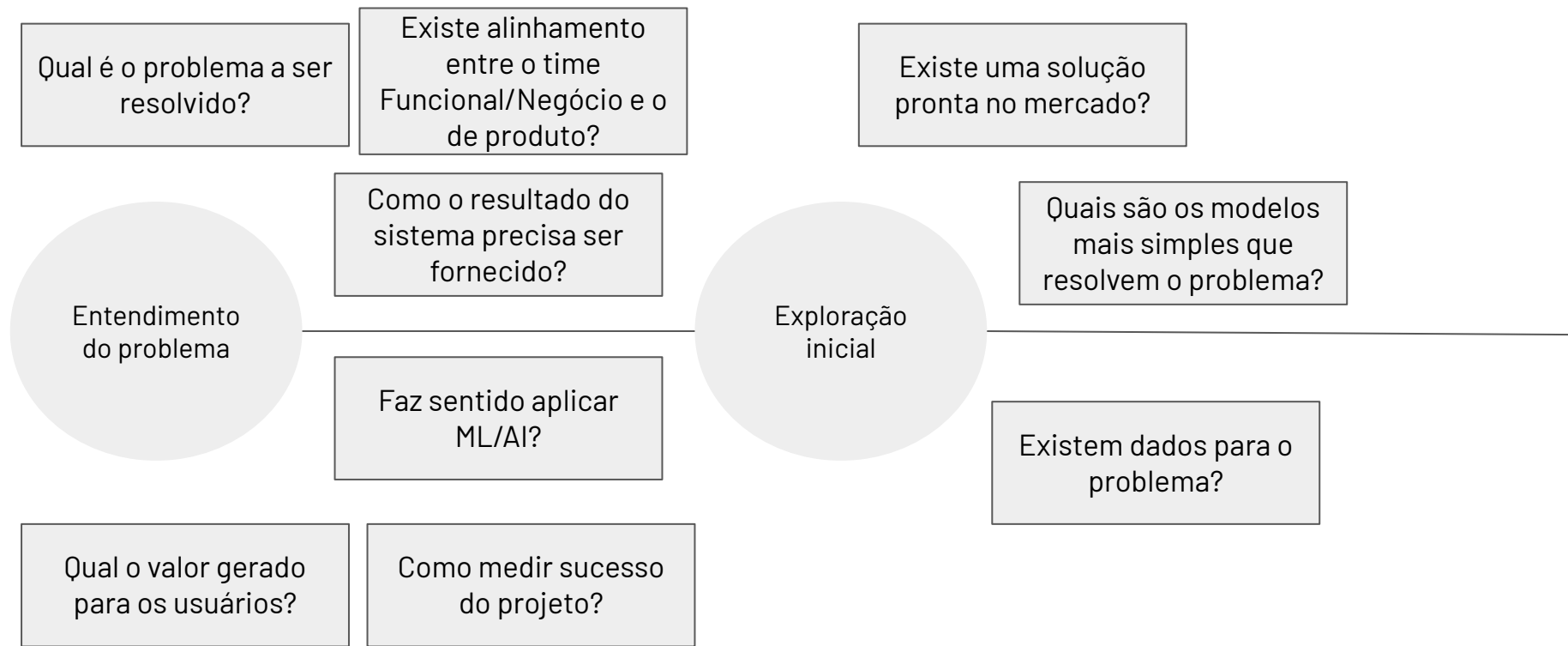


Voice-over

DS Challenge - Gabriel Pontes

Mapa mental de Resolução de problemas de DS



Análise de requisitos

Problema:

- Resolver o problema de **voice-over**: um problema de traduzir um áudio de uma língua para um áudio em outra língua – técnica muito utilizada em documentários. As vozes dos atores são gravadas sobre a faixa de áudio original que pode ser ouvida em segundo plano.

Entregáveis:

1. Transcrição em pt-br
2. Tradução em inglês
3. Sample do vídeo final em inglês com 3-5 minutos

Análise do vídeo sample

Em análise até 5 minutos de vídeo:

- Quantidades de locutores: 1
- Sampling rate:

Estudo dos dados

Antes mesmo de aplicar qualquer algoritmo de Machine Learning se faz necessário um estudo exploratório dos dados, com foco em entender a sua distribuição:

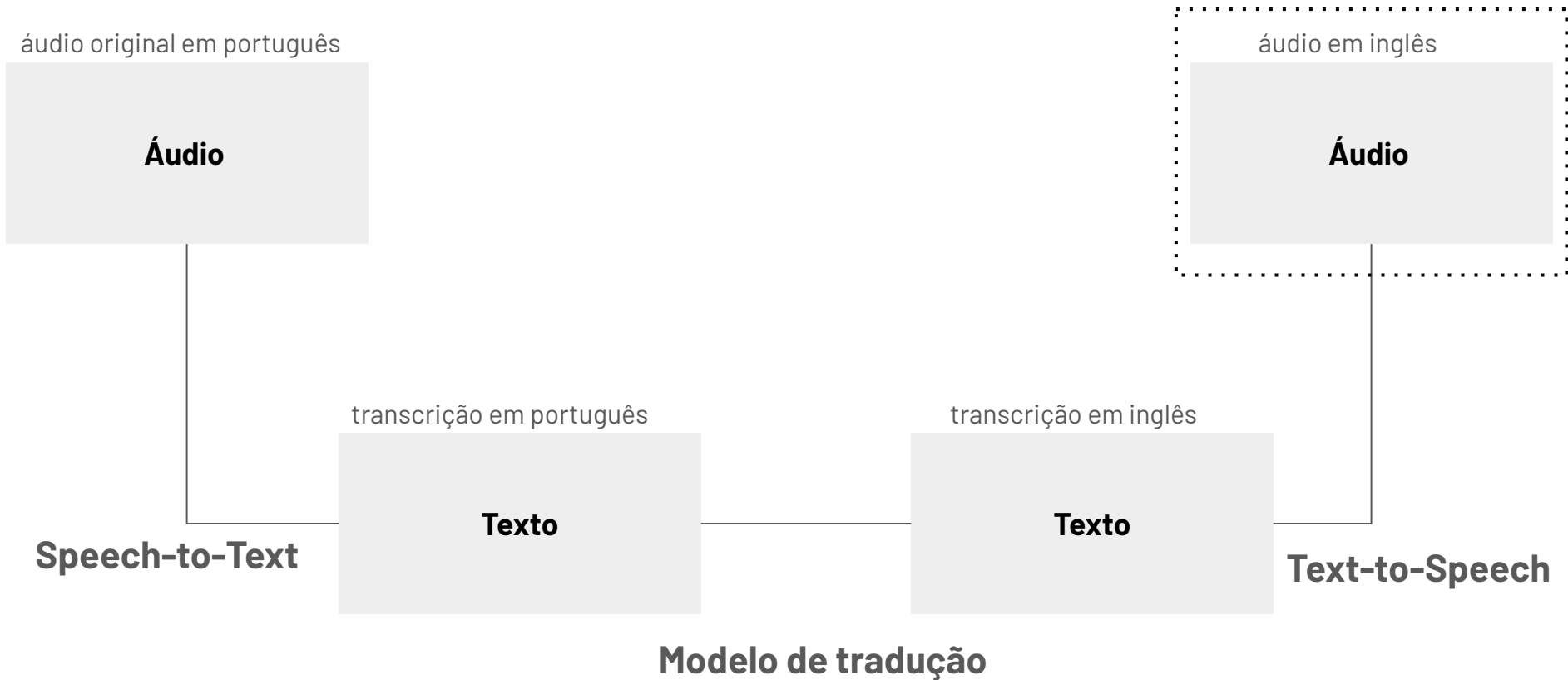
- Entender o domínio dos vídeos?
 - São podcasts? Vídeos de tutoriais? Vídeos de cursos?
- Os meus vídeos são muito longos? Qual a distribuição de tamanho desses vídeos?
 - Se os vídeos forem muito longos, será necessário alguma forma de quebrar o vídeo original em vídeos menores, sem prejudicar a performance de transcrição etc.
- Os vídeos apresentam mais de 1 interlocutor?
 - A presença de mais de 1 interlocutor pode prejudicar o trabalho dos algoritmos de ML.
- O conteúdo desses vídeos utiliza uma linguagem muito técnica ou é uma linguagem de contexto mais geral?
 - Isso pode ajudar a guiar se será necessário fine-tuning para os modelos especialistas.
-

Como o sistema será utilizado?

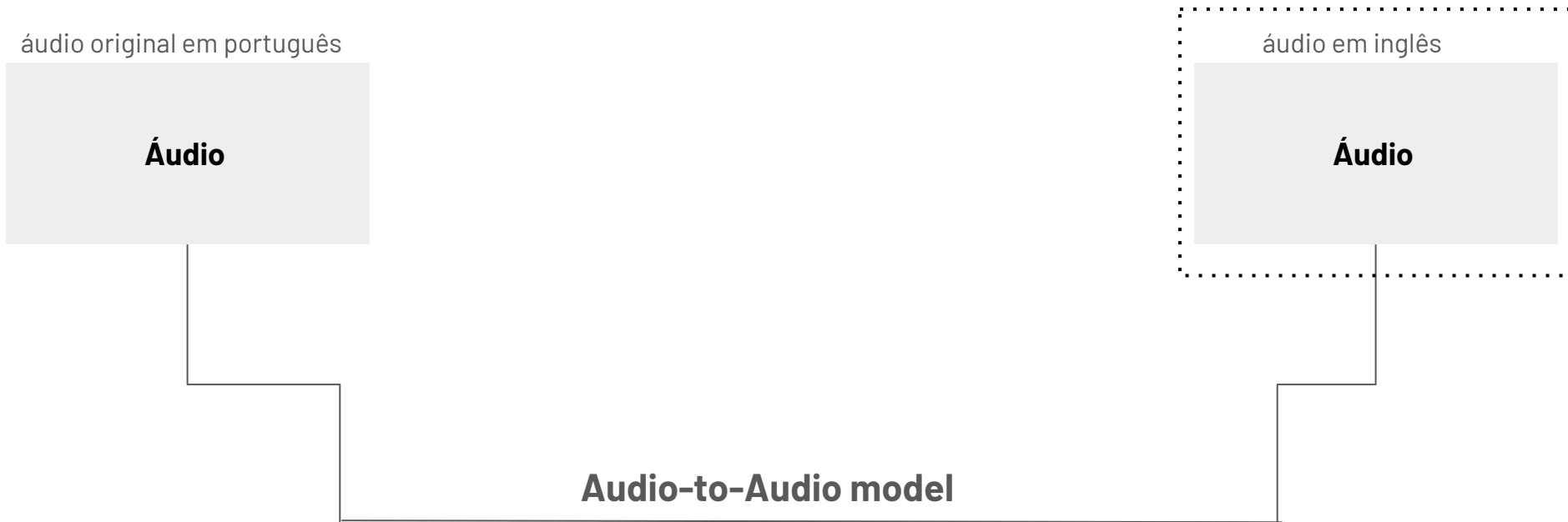
Durante a fase de design do sistema é importante levar em consideração os seguintes aspectos:

- O sistema será utilizado em batch ou tempo-real?
- Os erros dos modelos são críticos?

A arquitetura da solução



Solução Alternativa



Na solução anterior, os erros dos modelos vão sendo retropropagados um para o outro. Isso pode resultar em piora de performance. Como já existem modelos que trabalham com a modalidade audio-to-audio, pode ser interessante medir a sua performance.

A arquitetura da solução

3 modelos:

- Speech-to-Text
- Text-to-Tex / modelo de tradução
- Speech-to-Text / sintetizador de voz

Escolha dos modelos

Como existem muitos modelos para cada um dos cenários, acabei levando em consideração os seguintes pontos na escolha:

- Facilidade de utilização (tentei utilizar modelos que eu não precise manipular pré-processamentos aos dados)
- Custo de hardware (busquei os modelos mais baratos computacionalmente, tanto em termos de RAM como de GPU/CPU) tentando equilibrar com o custo de execução, afinal tempo também é uma variável importante.
- Se o modelo é open-source
- Se a sua licença open-source permite o uso comercial (apesar de ser uma PoC, é importante levar em consideração aspectos de viabilidade de produtização).
- A janela de contexto do modelo (quantidade de tokens de entrada).
- Validação qualitativa dos outputs dos modelos.

Num cenário de projeto real:

- Validar os modelos quantitativamente para cada uma das métricas de interesse.
- Latência dos modelos.

Modelos de Speech-To-Text (Speech Recognition)

Modelo	Companhia	Open Source	Pontos Fortes	Pontos Fracos
Whisper	OpenAI	Sim	ability to translate speech to text in over 60 languages	
DeepSpeech	Mozilla		a powerful model with high accuracy and good performance, suitable for real-time applications	requires significant computational resources for training and inference
Wav2Letter	Meta AI		known for its accuracy and efficiency	it currently supports fewer languages than DeepSpeech

Modelos de Translation pt->en

Modelo	Companhia	Open Source	Pontos Fortes	Pontos Fracos
MBart50	Facebook	Sim	Multilingual,	
T5	Google	Sim	Multilingual	Pode ser necessário fine-tuning para ter bons resultados

Modelos de TTS

Modelo	Companhia	Open Source	Pontos Fortes	Pontos Fracos
Massively Multilingual Speech (MMS): English Text-to-Speech	Facebook	Sim	ability to translate speech to text in over 60 languages	
Tacotron 2	DeepMind	Sim		Computacionalmente caro

Pré-processamento de áudio

Passo a passo:

- Resampling the audio data: sampling rate expected by a model
- Filtering the dataset: limiting the audio examples to a certain duration
- Converting audio data to model's expected input: extract input features

Como acabei utilizando o Whisper, não tive que me preocupar com nenhum pré-processamento.

Validação dos modelos

As métricas comumente utilizadas para validar os modelos utilizados na arquitetura sugerida:

Transcription model

Word Error Rate (WER)
Avaliação humana

Translation model

BLEU
METEOR
Avaliação humana

TTS model

Mel-cepstral distortion
(MCD)
PESQ (Perceptual Evaluation
of Speech Quality)
Avaliação humana

Abordagem híbrida

As métricas comumente utilizadas para validar os modelos utilizados na arquitetura sugerida:

- Pode ser interessante para o problema utilizar modelos pagos do Google/AWS/Microsoft/OpenAI para rotular dados e construir datasets para o treinamento dos modelos open source, já que esses modelos geralmente apresentam melhores resultados já porque os seus desenvolvedores apresentam uma quantidade massiva de dados.

Próximos passos

Em consequência do curto tempo disponível para o projeto acredito há componentes que podem ser melhorados:

- Testar LLM para tradução de textos.
- Explorar mais modelos de sintetização de voz, que respeite mais as pausas durante a fala.
- Investir mais tempo em técnicas de pré-processamento para dados de áudio e texto, com intuito de tornar o custo de execução o menor possível.
- Investigar quantos batchs um vídeo original deveria ser quebrado para aplicar as 3 tarefas em vídeos longos (como de 1h).
- Investigar possíveis vieses que podem surgir a partir dos sotaques brasileiros que são muitos variados e como isso impacta a qualidade dos resultados.
- Investigar os resultados dos modelos audio-to-audio.