



IME-USP

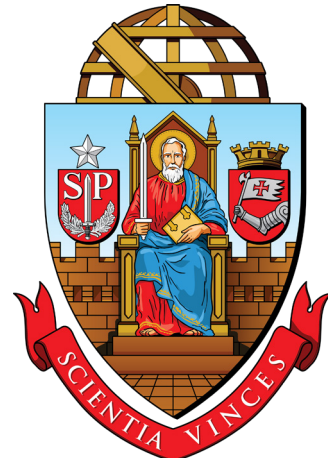
Desenvolvimento de um módulo de reconhecimento de voz para a *game engine* Godot

Leonardo Pereira Macedo

Orientador: Prof. Marco Dimas Gubitoso

Bacharelado em Ciência da Computação

Instituto de Matemática e Estatística | Universidade de São Paulo



Introdução

A área de jogos eletrônicos (video games) evoluiu muito desde o início da década da 70, quando começaram a ser comercializados. As principais causas estão relacionadas aos avanços em diferentes áreas da Computação.

Com o passar do tempo, surgiram as **game engines**: *frameworks* voltados especificamente para a criação de jogos, visando a facilitar o desenvolvimento e/ou algumas de suas etapas. Citamos *Unreal Engine*, *Unity* e *Godot* como exemplos.

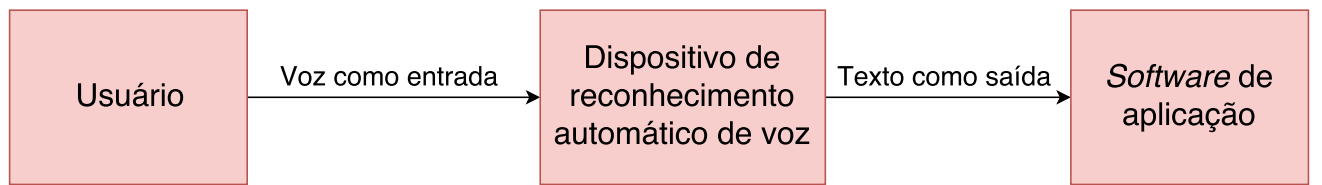
Também observam-se avanços significativos na área de reconhecimento de voz desde seus primeiros sistemas na década de 50. Esta tecnologia vem ficando cada vez mais integrada em nosso dia a dia, sendo usada para autenticação de usuário, buscas na Internet, etc.

Este trabalho visa a juntar ambos os temas ao desenvolver um módulo (“plugin”) de reconhecimento de voz para uma *game engine* em particular, *Godot*.

Reconhecimento de voz

Reconhecimento automático de voz é um campo que desenvolve técnicas para computadores captarem, reconhecerem e traduzirem a linguagem falada para texto; por isso também o nome *speech to text* (STT).

Um sistema genérico STT possui três componentes:



- O **usuário**: Codifica um comando através de sua voz. Em nosso contexto, corresponde ao jogador.
- O **dispositivo de STT**: Converte a mensagem falada para um formato interpretável. Equivale ao módulo de reconhecimento de voz que este trabalho busca produzir.
- O **software de aplicação**: Recebe a saída do dispositivo e realiza alguma ação. Corresponde a um jogo produzido em *Godot*.

Os principais termos de reconhecimento de voz incluem:

- Fluência**: A forma de comunicação com o sistema, podendo ser por palavras isoladas, conectadas ou fala contínua.
- Dependência do usuário**: Caracterizado pela existência ou não de treinamento feito pelo usuário para melhorar a acurácia do sistema.
- Vocabulário**: Palavras reconhecidas pelo sistema.
- Utterance**: Vocalização de uma ou mais palavras, possuindo um significado único ao computador.

Bibliotecas de reconhecimento de voz

Buscou-se uma biblioteca de reconhecimento de voz que possa ser usada no módulo. É importante que ela seja de código aberto e escrita em C/C++, características que *Godot* possui. O uso em jogos nos leva a desejar eficiência (rapidez) como uma característica essencial. Por fim, uma biblioteca configurável para diferentes línguas e sistemas operacionais seria desejável.

Encontramos três opções viáveis: *Kaldi*, *Pocketsphinx* e *HTK*. Apesar da primeira ser a mais eficiente, escolhemos usar a segunda pela sua maior leveza e facilidade de uso.

Pocketsphinx

Pocketsphinx, integrante do projeto *CMUSphinx*, é uma biblioteca de reconhecimento de voz desenvolvida pela *Carnegie Mellon University*.

Esta ferramenta considera que palavras são formadas por unidades menores chamadas **fonemas**. Usa-se o **Modelo Oculto de Markov** para melhores resultados: considera-se a fala como uma sequência de estados, que transitam entre si com certa probabilidade; os estados mais prováveis possuem uma melhor interpretação da voz.

- Destacamos três arquivos de configuração:
- O modelo acústico, composto por diversos arquivos que configuram detectores de fonemas;=.
 - O dicionário fonético, que mapeia palavras em fonemas. Por exemplo:

yellow Y EH L OW

- O arquivo de palavras-chave, que indica quais palavras do dicionário devem ser detectadas, de acordo com um limiar especificado. Por exemplo:

yellow /1e-6/

Godot

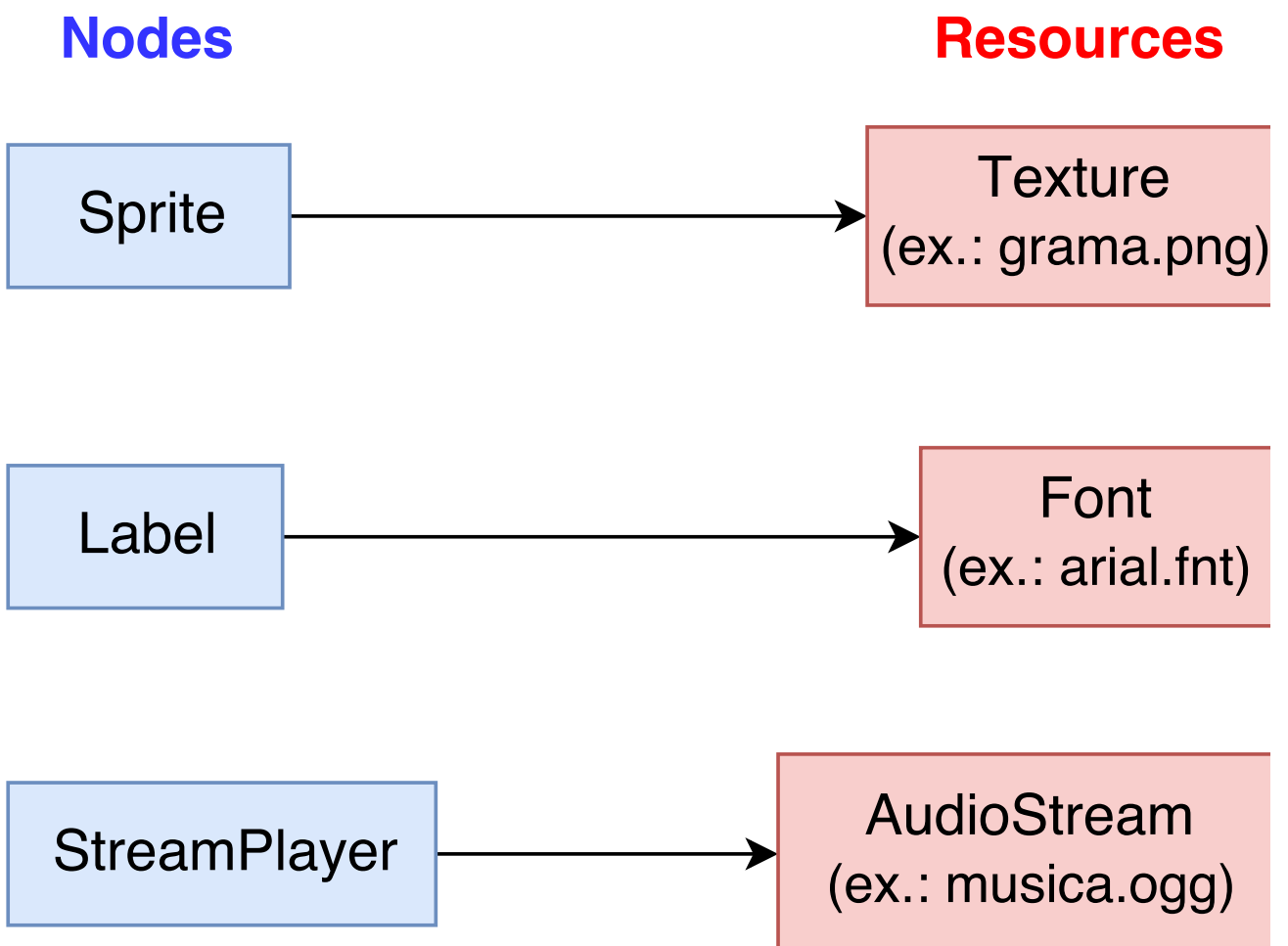


GODOT
Game engine

Godot é uma *game engine* criada por Juan Linietzky e Ariel Manzur em 2007, e cujo código código foi aberto ao público em 2014. Seu código fonte é escrito em **C++**, mas usuários do programa utilizam a linguagem nativa **GScript**, que permite programar com maior facilidade sem se preocupar com detalhes internos de implementação.

Dentre as classes mais importantes de sua arquitetura para este trabalho, destacamos:

- Object**: Classe base para todos os tipos não embutidos em *Godot*.
- Reference**: Herda de *Object*; implementa gerenciamento automático de memória.
- Resource**: Herda de *Reference*; funciona como um contêiner de dados.
- Node**: Herda de *Reference*; define um comportamento a ser usado em um jogo.

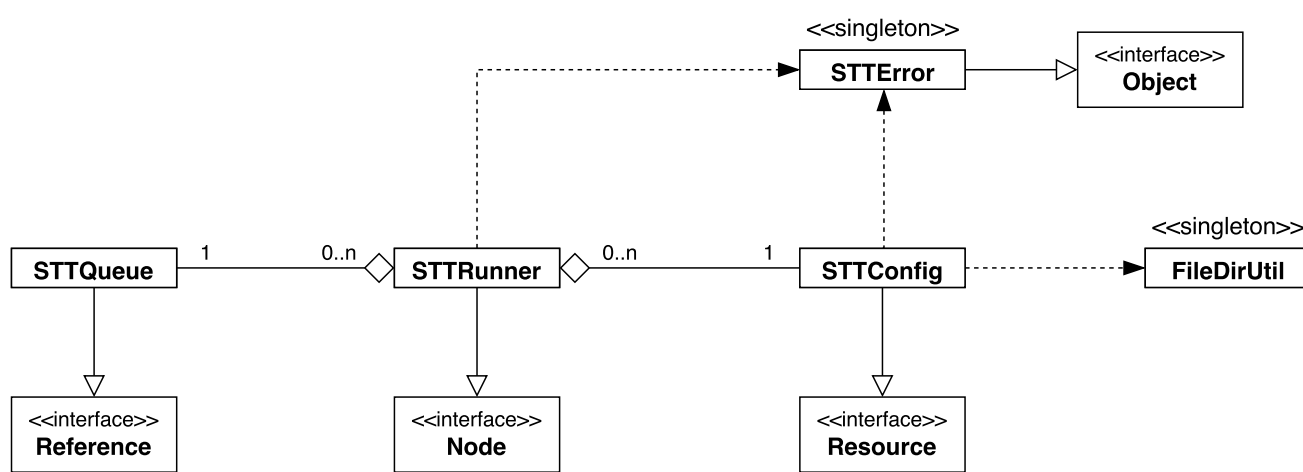


Módulo Speech to Text

Em relação a reconhecimento de voz, o módulo possui as seguintes características:

- Fluência**: Palavras conectadas.
- Dependência do usuário**: Sistema independente.
- Vocabulário**: Tipicamente pequeno.

Apresentamos a arquitetura do módulo *Speech to Text* a seguir:



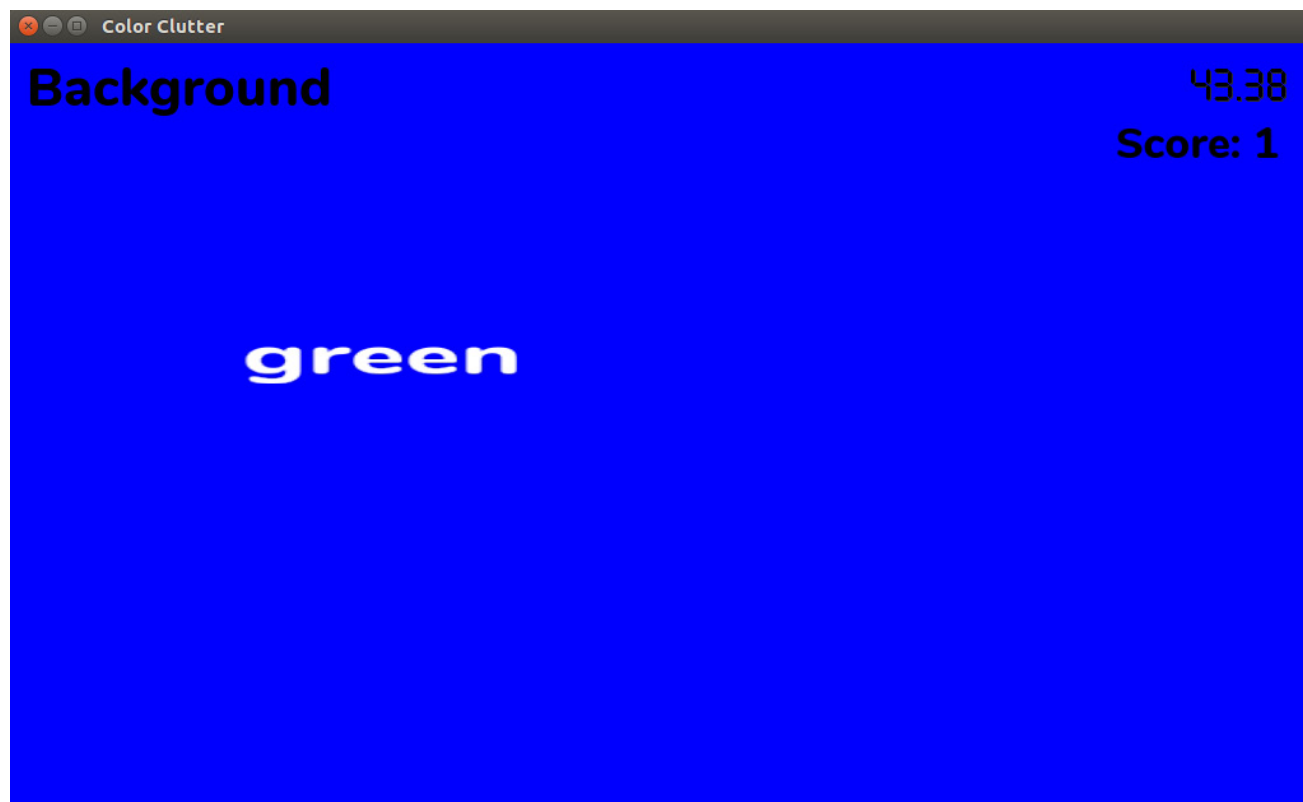
- As cinco classes implementadas são:
- STTConfig**: Controla arquivos de configuração de *Pocketsphinx*.
 - STTRunner**: Realiza o reconhecimento de voz paralelamente ao jogo, em uma thread.
 - STTQueue**: Contém uma fila para guardar palavras reconhecidas pelo *STTRunner*.
 - STTError**: Define constantes para possíveis erros ocorridos nas classes do módulo.
 - FileDirUtil**: Classe auxiliar para manipular arquivos e diretórios.

Jogo Color Clutter

Criamos um jogo simples, **Color Clutter**, para demonstrar o uso do módulo *Speech to Text*.

Uma típica tela do jogo consiste em um fundo totalmente preenchido com alguma cor X. Em alguma posição da tela, uma outra cor Y aparece escrita em um tom Z. O objetivo do usuário é falar a cor correta (X, Y ou Z), de acordo com o que é solicitado em uma legenda apresentada na tela.

No exemplo a seguir, o usuário deve dizer **blue** para prosseguir.



Referências

National Research Council. *Automatic Speech Recognition in Severe Environments*. The National Academies Press, 1984.

Kaldi. *About the Kaldi project*. URL: <http://kaldi-asr.org/doc/about.html>.

CMUSphinx. *About the CMUSphinx*. URL: <http://cmusphinx.sourceforge.net/wiki/about>.

HTK. *What is HTK?* URL: <http://htk.eng.cam.ac.uk>.

Linietzky, J. and Manzur, A. *Godot Engine*. URL: <https://godotengine.org>.