

SSC0601
Laboratório de Introdução à Ciência de Computação I
Engenharia de Computação

Projeto de Software

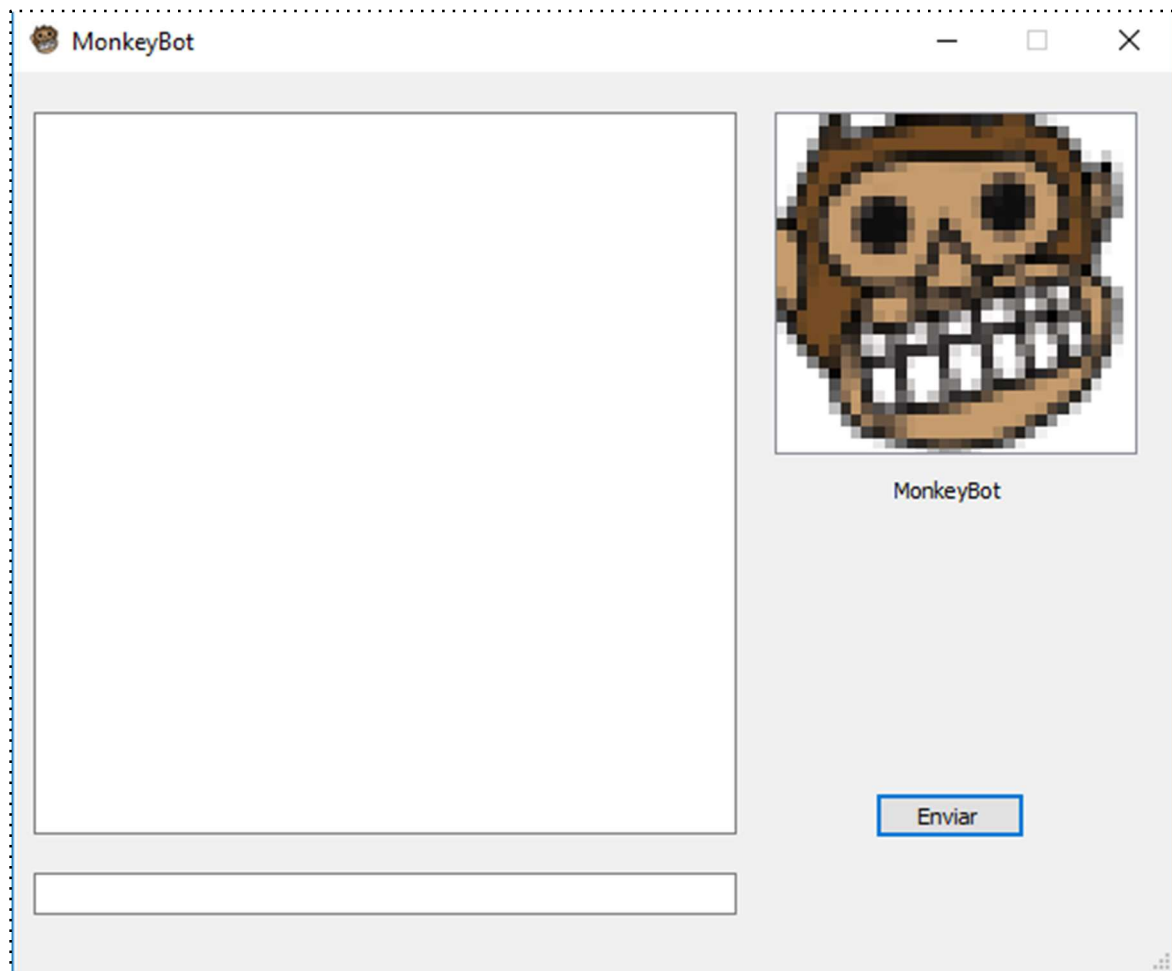


MonkeyBot

MonkeyBot: Um ChatBot que aprende

Nome: Breno Lívio Silva de Almeida **Nº USP:** 10276675

São Carlos
2017



Introdução

MonkeyBot é um programa de comunicação com uma espécie de inteligência artificial que consegue aprender respostas ao decorrer das conversas que essa realiza. Seu desenvolvimento foi na linguagem C++ utilizando do software Qt Creator. O fluxograma foi criado no programa DIA.

Objetivo

O objetivo do projeto é o de fazer uma inteligência artificial que não dependa de simplesmente frases pré-programadas em seu database, e sim que ela possa aprender com o tempo. Deve-se notar que uma inteligência artificial com frases pré-programadas a longo prazo tem a limitação do conhecimento, diferentemente de uma que aprende novas frases para o database e que a longo prazo ultrapassa o nível de conhecimento de uma IA que não aprende.

Descrição por texto

Envio -

O usuário entra em uma interface com um espaço para digitar. Ele deve digitar algo nesse espaço e digitar “Enter” ou simplesmente clicar no botão “Enviar”.

Análise da frase enviada e resposta consequente do Bot -

Após isso a frase será analisada e será verificado se ela já se encontra no database armazenado em um arquivo txt.

Deve-se saber que a estrutura desse arquivo está da seguinte forma:

```
FRASE ENVIADA  
RESPOSTA  
oi  
olá
```

Então, para a análise, deve-se analisar apenas as linhas ímpares.

A análise, talvez a parte mais complexa do programa, consiste em comparar a frase enviada pelo usuário com a frase já armazenada no database e funciona da seguinte maneira:

Digamos que temos a frase “como vai você?” como frase enviada e “vou bem” para resposta para essa palavra. O usuário digita “como vai você?” no espaço desejado e será comparado essa frase a cada uma das linhas ímpares (frases já enviadas). A comparação consiste em analisar caractere por caractere para verificar a compatibilidade entre as frases.

EXEMPLO (quando encontrada no database):

```
Usuário falou: ei, como vai você?  
Bot falou: Ah, eu vou bem...
```

A análise primeiramente começa assim:

Imagine o database

```
oi  
olá  
como vai você?  
vou bem
```

O usuário mandará uma frase e será feito testes com cada linha ímpar para ver se a frase dessa linha está contida na frase do usuário. Então o usuário mandou “ei, como vai você”, o programa começa na primeira linha ímpar “oi” e vê se tem uma letra ‘o’ precedida por espaço (se não estiver no começo da frase) em algum lugar.

Caso não achou, ele vai para a próxima linha ímpar, no caso “como vai você?” e verifica se há uma letra ‘c’, após achar, será verificado letra por letra. Há uma tolerância de erros: o usuário pode errar até 3 letras para uma frase de pelo menos 4 caracteres.

Se a frase não for encontrada, o Bot não entende e pergunta sobre a palavra enviada, o usuário então concede informação acerca da palavra, desde repetindo com outra palavra ou explicando seu significado.

EXEMPLO:

Usuário falou: bom dia

Bot perguntou: Hm? bom dia?

Usuário falou: é uma saudação quando está de manhã

No database fica

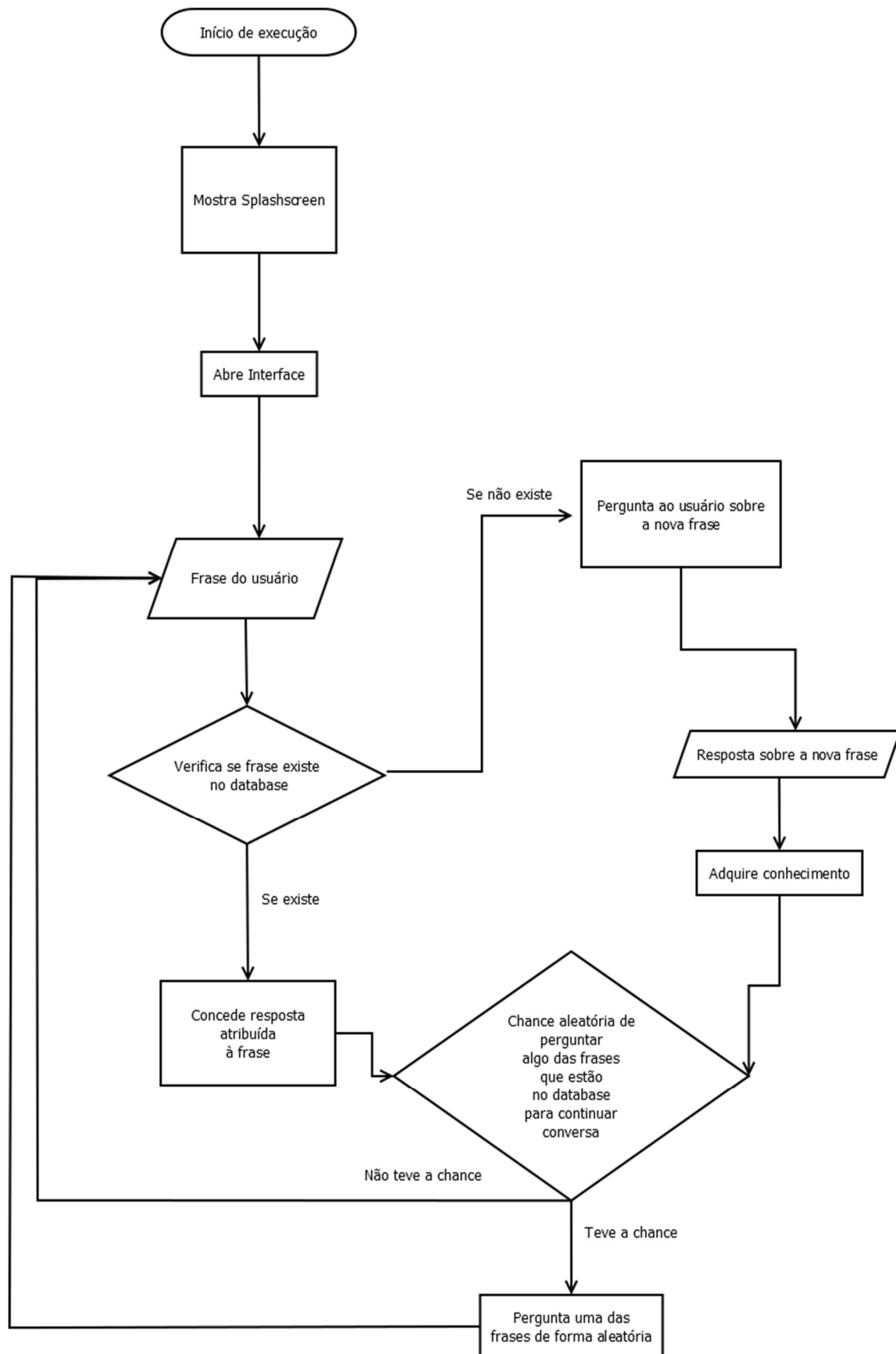
bom dia

é uma saudação quando está de manhã

No último exemplo, se o usuário agora falar bom dia, o Bot concede a resposta aprendida.

O Bot também pode fazer perguntas de forma aleatória usando frases anteriormente usadas pelo usuário.

Descrição gráfica (Fluxograma)



Conclusão e observações

A ideia de aprendizado do Bot é interessante, mas pode ser muito complexa. O próprio programa lida com vários casos, mas não abrange todos. Esses casos envolvem geralmente sintaxe da frase e outros problemas que poderiam ser solucionados com mais tempo de estudo.

Note que o Bot pode ser modelado a gosto do usuário, o que pode ser bom ou ruim, dependendo das intenções da pessoa que utiliza o software.

Um exemplo recente de ChatBot que aprende e foi corrompido pelos usuários:

<https://www.tecmundo.com.br/inteligencia-artificial/102782-tay-twitter-conseguiu-corromper-ia-microsoft-24-horas.htm>

O sistema de tolerância de erros pode ser prejudicial em alguns momentos da conversa, mas, em geral, é adequado para permitir que ela possa fluir entre o usuário e o Bot. Um detalhe que esqueci foi o de converter a frase da pessoa para uma frase com caracteres com letras apenas minúsculas, para assim possibilitar uma melhor comparação com as frases do database.