

INF1771 - INTELIGÊNCIA ARTIFICIAL

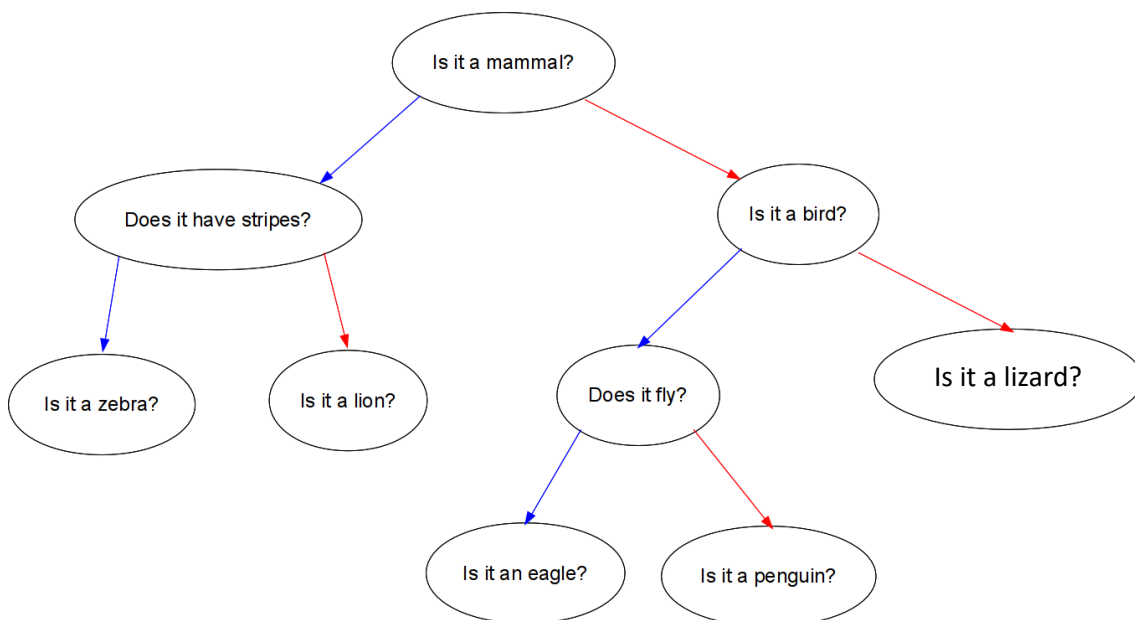
TRABALHO 2 – AGENTE LÓGICO – JOGO DOS ANIMAIS

Introdução

Na década de 80, existia um jogo para computadores chamado “jogo dos animais”. Esse jogo foi derivado de um programa de TV da década de 50 chamado “20 Questions”¹. Em essência, uma pessoa pensa em algo específico que a outra tentará adivinhar fazendo perguntas de sim / não. A mesma ideia também originou uma aplicação bastante famosa nos últimos anos chamada “Akinator”² onde o programa tenta adivinhar o que o usuário está pensando.

Para este trabalho, vamos criar uma versão do jogo dos animais em Prolog. O agente lógico deverá adivinhar qual é o animal (por exemplo, "Zebra") através da construção da base de conhecimento.

Na prática, o conhecimento é representado através de uma árvore de decisão binária onde cada nó representa uma pergunta, com a resposta sim ou não. A resposta do usuário determina para qual nó filho o agente deverá fazer a transição. Uma árvore para representar o conhecimento com vinte perguntas de "animais" seria semelhante a:



Observe que uma resposta "sim" sempre vai para o filho à esquerda (setas azuis), enquanto uma resposta "não" sempre vai para o filho à direita (setas vermelhas). As

¹ https://en.wikipedia.org/wiki/Twenty_Questions

² <https://pt.wikipedia.org/wiki/Akinator>

folhas da árvore são os palpites, portanto, devem representar um animal. Os nós internos são mais gerais e representam as decisões para separar as categorias de animais. Observe que todos os nós internos devem ter dois filhos, um para sim e outro para não!

Nesta tarefa, você implementará uma versão de aprendizado do agente lógico para o jogo dos animais: cada vez que se joga o jogo e o agente perde, ele deve aprender um novo animal. A base de conhecimento gerada durante a partida deve ser salva em disco para que se possa continuar com o novo conhecimento. Use o comando *listing* para listar a base de conhecimento e os comandos *tell/told* para salvar em disco.

Exemplos de Partida

```
Bem-vindo ao jogo dos animais. Vou tentar adivinhar qual está pensando.
```

```
É um mamífero? (s / n): n
```

```
Ele é um passaro? (s / n): s
```

```
Ele voa? (s / n): n
```

```
É um pinguim? (s / n): s
```

```
YAY! Adivinhei seu animal!
```

Exemplo quando o agente acerta o animal.

```
Bem-vindo ao jogo dos animais. Vou tentar adivinhar qual está pensando.
```

```
É um mamífero? (s / n): s
```

```
Tem listras? (s / n): n
```

```
É um leão? (s / n): n
```

```
Puxa! Eu não sei! Qual animal pensou? ornitorrinco
```

```
Qual pergunta devo fazer para distinguir 'ornitorrinco' de 'leão'?  
Ele ruge?
```

```
Agora digite qual a resposta certa para 'ornitorrinco' (s / n): n
```

```
Obrigado por me ensinar algo novo!
```

Exemplo quando o agente erra o animal e devemos ensinar algo novo.

Quando as novas questões são inseridas, o jogo precisa modificar a árvore do jogo.

Observe que a última pergunta respondida está em uma folha. Para expandir a árvore do jogo, a pergunta na folha ("É um leão?") precisa ser substituída pela nova pergunta sim / não ("Ele ruge?" ou "Ele põe ovos?"). Então, a velha questão dos animais ("É um leão?") e a nova questão dos animais ("É um ornitorrinco?") devem ser adicionadas como novos nós à árvore da base de conhecimento.

Perguntas Iniciais:

- P - É um mamífero?
- P - Tem listras?
- R - É uma zebra?
- R - É um leão?
- P - É um pássaro?
- P - Ele voa?
- R - É uma águia?
- R - É um pinguim?
- R - É um lagarto?

Forma de Avaliação:

Será avaliado se:

- (1) O trabalho atendeu a todos os requisitos especificados anteriormente;
- (2) Os algoritmos foram implementados e aplicados de forma correta;
- (3) O código foi devidamente organizado;
- (4) O trabalho foi apresentado corretamente em "sala de aula";

Recomendações Finais:

Muito do trabalho consiste em entender como representar os fatos, como definir o loop e como implementá-lo em Prolog.

Tenham dúvidas e façam as perguntas para que eu possa ajuda-los durante o desenvolvimento