

Portfólio 2 - Projeto utilizando conceito de banco de conhecimentos para solução de um problema

1 Introdução

Este documento tem como foco apresentar o problema a ser resolvido, a implementação e os resultados de um Projeto utilizando conceito de banco de conhecimentos para solução de um problema, desenvolvido na linguagem Python, para a disciplina de Inteligência Artificial.

Foram utilizados a IDE pycharm para implementar conceito de banco de conhecimentos para solução de um problema.

2 Problema

Um problema comum é o da verificação de instância, que procura determinar se um indivíduo específico, com base nos fatos conhecidos sobre ele, é membro de um determinado conceito ou categoria.

Imagine que você está construindo uma base de conhecimentos que precisa entender o mundo de forma lógica esta base armazena dois tipos de informação, primeiro fatos diretos sobre indivíduos específicos como "Sócrates é um Homem" e "Félix é um Gato", e o segundo, um conjunto de regras universais ou hierarquias, como "Todo Homem é Mortal" e "Todo Gato é um Mamífero", o problema da verificação de instância ocorre quando fazemos uma pergunta ao sistema que não pode ser respondida apenas olhando os fatos diretos.

Por exemplo, se perguntarmos "Sócrates é Mortal?", a base de conhecimento não encontrará esse fato exato armazenado o desafio é determinar se o sistema pode, usando seu conhecimento das regras, inferir logicamente a resposta.

3 Projeto utilizando conceito de banco de conhecimentos para solução de um problema: tableau

O nosso sistema precisa analisar um fato conhecido e aplicar a regra relevante nele para concluir se o indivíduo pertence à tal categoria, mesmo que esse fato nunca tenha sido afirmado explicitamente.

Para realizar essa inferência de maneira formal e garantir a correção lógica, é utilizado o algoritmo Tableau. Ele faz essa verificação de instância essencialmente testando a consistência da base de conhecimento, ao tentar provar que a afirmação oposta que o indivíduo não pertence ao conceito levaria a uma contradição.

3.1 Exemplo de uso

O código não olha somente os fatos diretos, ele executa um processo de raciocínio quando você pergunta se "Sócrates" é "Mortal", o código primeiro consulta sua base de fatos ABox e descobre que "Sócrates" é um "Homem" em seguida, ele usa esse fato como ponto de partida para explorar a base de regras TBox ele inicia uma busca, pegando o conceito "Homem" e procurando todas as regras que se aplicam a ele, o código encontra a regra "Todo Homem é Mortal" e, com base nisso, infere que "Mortal" é um novo conceito que também se aplica a "Sócrates".

Esse novo conceito inferido é adicionado a uma lista temporária de todos os tipos aos quais "Sócrates" pertence esse processo continua recursivamente por exemplo, se houvesse outra regra como "Todo Mortal é um SerVivo", o código a encontraria e adicionaria "SerVivo" à lista também após esgotar todas as regras aplicáveis em toda a hierarquia, o código gera uma lista completa de todos os conceitos, tanto diretos quanto inferidos, que descrevem "Sócrates" e por fim, ele simplesmente verifica se o conceito original da pergunta ("Mortal") está presente nessa lista completa se sim, ele retorna "Sim" caso contrário, retorna "Não".

```
==== CONFIGURANDO O TBOX (REGRAS) ====
[Regra Adicionada] REGRA: Todo 'Homem' é um 'Mortal'.
[Regra Adicionada] REGRA: Todo 'Homem' é um 'Mamifero'.
[Regra Adicionada] REGRA: Todo 'Gato' é um 'Mamifero'.
[Regra Adicionada] REGRA: Todo 'Mamifero' é um 'Animal'.
[Regra Adicionada] REGRA: Todo 'Peixe' é um 'Animal'.

-----
==== CONFIGURANDO O ABOX (FATOS) ====
[Fato Adicionado] INDIVÍDUO: 'Socrates' é um 'Homem'.
[Fato Adicionado] INDIVÍDUO: 'Felix' é um 'Gato'.
[Fato Adicionado] INDIVÍDUO: 'Nemo' é um 'Peixe'.

-----
--- Verificando Instância: 'Socrates' é um 'Homem'? ---
Raciocinador: Buscando todos os tipos de 'Socrates'...
Raciocinador: Tipos encontrados (diretos e inferidos) para 'Socrates': {'Mamifero', 'Mortal', 'Animal', 'Homem'}
Resultado: SIM. 'Homem' está na lista de tipos inferidos.

--- Verificando Instância: 'Socrates' é um 'Mortal'? ---
Raciocinador: Buscando todos os tipos de 'Socrates'...
Raciocinador: Tipos encontrados (diretos e inferidos) para 'Socrates': {'Mamifero', 'Mortal', 'Animal', 'Homem'}
Resultado: SIM. 'Mortal' está na lista de tipos inferidos.

--- Verificando Instância: 'Felix' é um 'Animal'? ---
Raciocinador: Buscando todos os tipos de 'Felix'...
Raciocinador: Tipos encontrados (diretos e inferidos) para 'Felix': {'Mamifero', 'Gato', 'Animal'}
Resultado: SIM. 'Animal' está na lista de tipos inferidos.

--- Verificando Instância: 'Felix' é um 'Mortal'? ---
Raciocinador: Buscando todos os tipos de 'Felix'...
Raciocinador: Tipos encontrados (diretos e inferidos) para 'Felix': {'Mamifero', 'Gato', 'Animal'}
Resultado: NÃO. 'Mortal' não pode ser inferido como um tipo para 'Felix'.

--- Verificando Instância: 'Nemo' é um 'Mamifero'? ---
Raciocinador: Buscando todos os tipos de 'Nemo'...
Raciocinador: Tipos encontrados (diretos e inferidos) para 'Nemo': {'Peixe', 'Animal'}
Resultado: NÃO. 'Mamifero' não pode ser inferido como um tipo para 'Nemo'.
```

Figura 1: Execução do algoritmo tableau