

Sistemas De Representação De conhecimento e Raciocínio

GRUPO 44

Ana Teresa Gião Gomes - A89536
Maria Quintas Barros - A89325
Maria Beatriz Araújo Lacerda - A89535
Francisco Franco - A89536
3º ano de MIEI

Abril de 2021



Figura 1: A89536
Ana Gomes

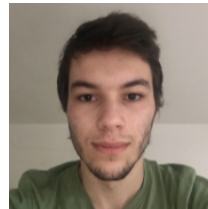


Figura 2: A89536
Francisco Franco



Figura 3: A89525
Maria Barros



Figura 4: A89535
Beatriz Lacerda

Conteúdo

1	Introdução	4
2	Preliminares	4
3	Sistema de Inferência	5
4	Análise do trabalho e resultados obtidos	5
4.1	Base de Conhecimento	5
4.2	Conhecimento Positivo e Conhecimento Negativo	5
4.3	Conhecimento Imperfeito	6
4.3.1	Conhecimento Incerto	6
4.3.2	Conhecimento Impreciso	7
4.3.3	Conhecimento Interdito	8
4.4	Manipulação de Invariantes	8
5	Conclusão	10

Resumo

O presente relatório diz respeito ao segundo exercício prático de grupo proposto na unidade curricular de Sistemas de Representação de Conhecimento e raciocínio da Universidade Do Minho. O objetivo deste trabalho é construir, utilizando a linguagem PROLOG, um mecanismo de representação de conhecimento na área da prestação de cuidados de saúde, mais especificamente na vacinação contra o COVID-19.

Ao longo deste relatório apresentaremos as estratégias adotadas para implementar conhecimento imperfeito no nosso sistema.

1 Introdução

No âmbito da unidade curricular de Sistemas de Representação de COhecimento e Raciocínio foi nos proposto adicionar ao exercício anterior representação de conhecimento imperfeito.

Deste modo, os objetivos para este exercício são:

- Representar conhecimento positivo e negativo;
- Representar casos de conhecimento imperfeito, pela utilização de valores nulos de todos os tipos estudados;
- Manipular invariantes que designem restrições à inserção e à remoção de conhecimento do sistema;
- Lidar com a problemática da evolução do conhecimento, criando os procedimentos adequados;
- Desenvolver um sistema de inferência capaz de implementar os mecanismos de raciocínio inerentes a estes sistemas.

2 Preliminares

Antes de abordar o trabalho proposto, o grupo procurou saber mais sobre este tema. Deste modo, focamos a nossa atenção nos vários tipos de armazenamento de informação e os seus pressupostos, obtendo dois:

- **Base De Dados**

Uma base de dados considera toda a informação que não se encontra referenciada nela como falsa, partindo, deste modo, de um Pressuposto de Mundo Fechado.

Parte também de um Pressuposto dos Nomes Únicos, dado que duas constantes diferentes representam duas entidades diferentes nesse mesmo universo de representação.

Para além disto, parte-se também do princípio que não existem mais objetos nesse universo para além daqueles já representados por constantes na base de dados (Pressuposto do Domínio Fechado).

- **Sistema de Representação de Conhecimento**

Um sistema de representação de conhecimento existe o Pressuposto de Mundo Aberto, isto é, existe a possibilidade de existirem factos verdadeiros para lá daqueles que se encontram referenciados na base de conhecimento.

Parte também o Pressuposto de Nomes Únicos, à semelhança das Bases de Dados.

Depois de comparar os dois sistemas possíveis sobre os quais uma linguagem de manipulação de informação poderá basear, chegamos à conclusão que os Sistemas de Representação de Conhecimento são menos limitantes, pois permitem-nos processar informação incompleta.

No que toca a conhecimento imperfeito, este pode assumir três formas distintas:

- **Incerto** Conhecimento acerca do qual não possuímos qualquer informação.
- **Impreciso** Conhecimento acerca do qual sabemos um intervalo de valores.
- **Interdito** Conhecimento que não será representado nem conhecido.

3 Sistema de Inferência

Visto que adicionamos a capacidade para representar o conhecimento incompleto ao nosso sistema temos agora 3 respostas possíveis a uma questão Q:

- **Verdadeira** Quando é possível provar que Q faz parte da base de conhecimento.
- **Falsa** Quando for possível provar a falsidade da questão Q.
- **Desconhecida** Quando não for possível provar que a questão Q é falsa ou verdadeira.

Para isto foi desenvolvido o meta-predicado "demo". O meta-predicado responsável por verificar a existência da questão Q na base de conhecimento é então o "não".

```
demo(Questao, falso) :- ~Questao.  
demo(Questao, verdadeiro) :- Questao.  
demo( Questao, desconhecido ) :-  
    nao( Questao ),  
    nao( ~Questao ).
```

```
nao( Questao ) :- Questao, !, fail.  
nao( _ ).
```

4 Análise do trabalho e resultados obtidos

4.1 Base de Conhecimento

No seguimento do exercício anterior a nossa base de conhecimento é composta por:

- Extensão do predicado utente:
Idutente, N^o Segurança Social, Nome, Data_Nasc, Email, Telefone, Morada, [Doenças_Crónicas], Profissão, CentroSaúde $\rightarrow V, F$
- Extensão do predicado centro de saúde:
Idcentro, Nome, Morada, Telefone, email $\rightarrow V, F$
- Extensão do predicado staff:
idstaff, Idcentro, Nome, email $\rightarrow V, F$
- Extensão do predicado vacinaçao_covid:
Staff, utente, Data, Vacina, Toma $\rightarrow V, F$

4.2 Conhecimento Positivo e Conhecimento Negativo

A representação do conhecimento positivo foi feita no exercício 1, sendo que esta é baseada nos factos. Exemplo de conhecimento positivo:

```
utente(1, 23456, 'Carlos', 24-04-1999, 'carlos@gmail.com', '966345781', 'Rua do  
    Falco', [Renite], 'Professor', 1).  
centro_saude(1, 'Centro de Sade de Barcelos', 'Rua Alta', '253234674', 'centro1@gmail.com').  
staff(1, 1, 'Rute Silva', 'rute@gmail.com').  
vacinacao_covid(1, 2, 6-3-2021, Vacina2, 1).
```

Quanto ao conhecimento negativo, este pode ser feito de duas formas: negação por falha ou negação forte.

Exemplo de negação forte:

```
%Um utente id 11 mas apenas sabemos que centro de saude nao e 1
utente(11, 56320, 'Jonas', 23-09-1984, 'jonas@gmail.com', '965874301', 'Rua do Gota', [],
'Dermatologista', centro_desconhecido).
?-utente(11, 56320, 'Jonas', 23-09-1984, 'jonas@gmail.com', '965874301', 'Rua do Gota', [],
'Dermatologista', 1).
```

Output:

```
?- demo(utente(11, 56320, 'Jonas', 23-09-1984, 'jonas@gmail.com', '965874301', 'Rua do
Gota', [], 'Dermatologista', 1),R).
R = falso .

?- demo(utente(11, 56320, 'Jonas', 23-09-1984, 'jonas@gmail.com', '965874301', 'Rua do
Gota', [], 'Dermatologista', 4),R).
R = desconhecido.
```

Negação Por Falha

O código abaixo representa o conhecimento negativo relativo aos utentes.

```
-utente(A,B,C,D,E,F,G,H,I,J) :- nao(utente(A,B,C,D,E,F,G,H,I,J)),
nao(excecao(utente(A,B,C,D,E,F,G,H,I,J))).
```

Foram utilizados como condições o "não" e a "exceção", visto que o "não" é representativo da negação por falha e o "exceção" representa o conhecimento imperfeito.

4.3 Conhecimento Imperfeito

4.3.1 Conhecimento Incerto

No caso do conhecimento incerto (valor nulo do tipo I), não possuímos qualquer informação sobre a informação desconhecida.

Para atingir isto, a informação desconhecida será um valor nulo, valor este que está associado a uma exceção. Deste modo, o conhecimento será representado como desconhecido.

```
%Utente de id 15 em que nao e conhecido do seu numero de telemovel
utente(15, 75203, 'Carolina Alves', 27-01-1998, 'carolina@gmail.com',
telemovel_desconhecido, 'Rua dos Gomes', [], 'Estudante', 2).

%Membro do staff em que nao e conhecido o centro de saude onde trabalha
staff(8, centro_desconhecido, 'Vasco Correia', 'vasco@gmail.com').

%Excecoes:
% Nmero de telemovel desconhecido
excecao(utente(A,B,C,D,E,F,G,H,I,J)) :- utente(A,B,C,D,E, telemovel_desconhecido, G,H,I,J).
%Centro desconhecido
excecao(staff(A,B,C,D)) :- staff(A, centro_desconhecido, C,D).
```

Exemplos do output:

```
?- demo(staff(8,1, 'Vasco Correia', 'vasco@gmail.com'),R).
R = desconhecido.

?- demo(staff(8,2, 'Vasco Correia', 'vasco@gmail.com'),R).
R = desconhecido.

?- demo(utente(15, 75203, 'Carolina Alves', 27-01-1998, 'carolina@gmail.com',
'965230147', 'Rua dos Gomes', [], 'Estudante', 2),R).
```

```
R = desconhecido.
```

```
?- demo(utente(15, 75203, 'Carolina Alves', 27-01-1998, 'carolina@gmail.com',  
            '975745200', 'Rua dos Gomes', [], 'Estudante', 2), R).  
R = desconhecido.
```

Tal como podemos verificar nos outputs acima, qualquer que seja o valor introduzido no número de telemóvel do utente ou no centro do staff, o resultado do meta-predicado demo é desconhecido.

4.3.2 Conhecimento Impreciso

Para o conhecimento impreciso (valor nulo tipo II), a informação é desconhecida dentro de um intervalo de valores. Assim, qualquer valor dentro do intervalo definido é considerado desconhecido, enquanto que os valores fora do intervalo são considerados falsos.

```
% um membro do staff de id 7 pode trabalhar no centro de saude 1 ou no centro 2  
excecao(staff(7,1, 'Cardoso', 'patricia@gmail.com')).  
excecao(staff(7,2, 'Cardoso', 'patricia@gmail.com')).  
  
% utente nasceu entre 1945 e 1955  
excecao(utente(14, 77777, 'Carolina Alves', 27-01-A, 'carolinaa@gmail.com', '968520362', 'Rua  
da Amargura', [], 'Reformada', 1)) :- A >= 1945 , A <= 1955.
```

Tal como podemos observar no código acima, o membro do staff pode trabalhar em dois possíveis centros, isto é, entre dois valores pontuais. Para além disto, temos a utente de ID 14 cujo ano de nascimento pertence ao intervalo [1945,1955]

Output:

```
?- demo(staff(7,1, 'Cardoso', 'patricia@gmail.com'), R).  
R = desconhecido.  
  
?- demo(staff(7,2, 'Cardoso', 'patricia@gmail.com'), R).  
R = desconhecido.  
  
?- demo(staff(7,5, 'Cardoso', 'patricia@gmail.com'), R).  
R = falso .
```

Tal como vemos no output, quando introduzimos o centro de saúde como 1 ou 2, recebemos como resultado desconhecido. Quando introduzimos o id do centro como 5, é nos retornado falso.

```
?- demo(utente(14, 77777, 'Carolina Alves', 27-01-2000, 'carolinaa@gmail.com',  
            '968520362', 'Rua da Amargura', [], 'Reformada', 1), R).  
R = falso .  
  
?- demo(utente(14, 77777, 'Carolina Alves', 27-01-1950, 'carolinaa@gmail.com',  
            '968520362', 'Rua da Amargura', [], 'Reformada', 1), R).  
R = desconhecido.
```

O mesmo se verificou no exemplo do utente. Quando introduzimos um valor no ano de nascimento fora do intervalo é nos retornado falso. Quando introduzimos, por exemplo, 1950 como ano de nascimento, é nos retornado desconhecido.

4.3.3 Conhecimento Interdito

Relativamente ao conhecimento interdito (valor nulo tipo III), este não deve ser conhecido ou especificado.

```
%utente 12 tem um nmero de segurana social que nao pode ser conhecido
utente(12, ssocial_desconhecido, 'Vicente', 21-08-1994, 'vicente@gmail.com',
    '965209421', 'Rua do Lado', [], 'Estudante', 1).

excecao(utente(A,B,C,D,E,F,G,H,I,J)) :-utente(A,ssocial_desconhecido,C,D,E,F,G,H,I,J).

nulo(ssocial_desconhecido).
```

Como o objetivo é que não seja permitido representar ou conhecer o número da segurança social do utente, atribuímos valor nulo a esse parâmetro. E visto que é conhecimento desconhecido é também necessário associar-lhe uma exceção, à semelhança de outros exemplos acima.

4.4 Manipulação de Invariantes

Para que seja possível inserir e remover conhecimento da nossa base, foi necessário a criação de vários invariantes.

A acrescentar aos invariantes desenvolvidos no primeiro exercício, que permitiam a inserção e remoção de conhecimento perfeito, foi necessário criar outros invariantes que nos permitem lidar com a problemática do conhecimento imperfeito.

Deste modo, criamos invariantes relativos ao conhecimento nulo interdito. A função destes invariantes é impedir a inserção de informação, visto que não é suposto ela ser conhecida em nenhum ponto.

```
%Invariante que nao permite a insercao de um numero de segurana social no utente
apresentado na seco anterior (ID = 12)
+utente(A,B,C,D,E,F,G,H,I,J) :: (resposta(ssocial_desconhecido,(utente(12,
    ssocial_desconhecido, 'Vicente', 21-08-1994, 'vicente@gmail.com', '965209421', 'Rua do
    Lado', [], 'Estudante', 1),nao(nulo(ssocial_desconhecido))),S), comprimento(S,N), N==0).
```

Outputs:

```
?- listing(utente).:- dynamic utente/10.

utente(1, 23456, 'Carlos', 24-4-1999, 'carlos@gmail.com', '966345781', 'Rua do Falco',
    [Renite], 'Professor', 1).
utente(2, 23906, 'Rui', 13-3-1987, 'rui@gmail.com', '989349235', 'Rua da Esquina', [],
    'Mdico', 1).
utente(3, 12396, 'Carlota', 20-9-2005, 'carlota@gmail.com', '963215637', 'Rua Alto Mar',
    [], 'Estudante', 1).
utente(4, 76590, 'Maria', 25-9-1935, 'maria@gmail.com', '934674129', 'Rua do Bairro', [],
    'Carpinteira', 1).
utente(5, 56851, 'Jorge', 29-5-1955, 'jorge@gmail.com', '962458965', 'Rua de S.Bento da
    Vrzea', [], 'Psicologa', 1).
utente(6, 65975, 'Alexandra', 18-2-2009, 'alexandra@gmail.com', '912358957', 'Rua Alto
    Mar', [], 'Enfermeira', 2).
utente(10, 75362, 'Lisandro', 5-6-2005, 'lisandro@gmail.com', '963030147', 'Rua Lisboa',
    [], 'Estudante', 2).
utente(7, 45210, 'Cristiana', 18-8-2012, 'cristiana@gmail.com', '932013663', 'Rua dos
    Pees', [Alergias, Bronquite], 'Tradutor', 2).
utente(8, 78563, 'Slvio', 14-11-2000, 'silvio@gmail.com', '932456901', 'Rua Antnio
    Cardodo', [], 'Eletricista', 2).
```



```

utente(9, 96203, 'Rafael', 3-10-1977, 'rafael@gmail.com', '963124032', 'Rua da Boa
  Vontade', [], 'Engenheira', 2).
utente(11, 56320, 'Jonas', 23-9-1984, 'jonas@gmail.com', '965874301', 'Rua do Gota', [],
  'Dermatologista', centro_desconhecido).
utente(12, ssocial_desconhecido, 'Vicente', 21-8-1994, 'vicente@gmail.com', '965209421',
  'Rua do Lado', [], 'Estudante', 1).
utente(15, 75203, 'Carolina Alves', 27-1-1998, 'carolina@gmail.com',
  telemovel_desconhecido, 'Rua dos Gomes', [], 'Estudante', 2).

true.

?- evolucao(utente(12, 12896, 'Vicente', 21-08-1994, 'vicente@gmail.com', '965209421', 'Rua
  do Lado', [], 'Estudante', 1)).
false.

?- evolucao(utente(17, 12896, 'Raquel', 02-09-1986, 'raquel@gmail.com', '962030147', 'Rua
  Silva', [], 'Professora', 2)).
true .

?- listing(utente).
:- dynamic utente/10.

utente(1, 23456, 'Carlos', 24-4-1999, 'carlos@gmail.com', '966345781', 'Rua do Falco',
  [Renite], 'Professor', 1).
utente(2, 23906, 'Rui', 13-3-1987, 'rui@gmail.com', '989349235', 'Rua da Esquina', [],
  'Mdico', 1).
utente(3, 12396, 'Carlota', 20-9-2005, 'carlota@gmail.com', '963215637', 'Rua Alto Mar',
  [], 'Estudante', 1).
utente(4, 76590, 'Maria', 25-9-1935, 'maria@gmail.com', '934674129', 'Rua do Bairro', [],
  'Carpinteira', 1).
utente(5, 56851, 'Jorge', 29-5-1955, 'jorge@gmail.com', '962458965', 'Rua de S.Bento da
  Vrzea', [], 'Psicloga', 1).
utente(6, 65975, 'Alexandra', 18-2-2009, 'alexandra@gmail.com', '912358957', 'Rua Alto
  Mar', [], 'Enfermeira', 2).
utente(10, 75362, 'Lisandro', 5-6-2005, 'lisandro@gmail.com', '963030147', 'Rua Lisboa',
  [], 'Estudante', 2).
utente(7, 45210, 'Cristiana', 18-8-2012, 'cristiana@gmail.com', '932013663', 'Rua dos
  Pees', [Alergias, Bronquite], 'Tradutor', 2).
utente(8, 78563, 'Slvio ', 14-11-2000, 'silvio@gmail.com', '932456901', 'Rua Antnio
  Cardodo', [], 'Eletricista', 2).
utente(9, 96203, 'Rafael', 3-10-1977, 'rafael@gmail.com', '963124032', 'Rua da Boa
  Vontade', [], 'Engenheira', 2).
utente(11, 56320, 'Jonas', 23-9-1984, 'jonas@gmail.com', '965874301', 'Rua do Gota', [],
  'Dermatologista', centro_desconhecido).
utente(12, ssocial_desconhecido, 'Vicente', 21-8-1994, 'vicente@gmail.com', '965209421',
  'Rua do Lado', [], 'Estudante', 1).
utente(15, 75203, 'Carolina Alves', 27-1-1998, 'carolina@gmail.com',
  telemovel_desconhecido, 'Rua dos Gomes', [], 'Estudante', 2).
utente(17, 12896, 'Raquel', 2-9-1986, 'raquel@gmail.com', '962030147', 'Rua Silva', [],
  'Professora', 2).

true.

```

5 Conclusão

Devido a restrições de tempo optamos por não explorar todas as possibilidades de evolução na nossa base tais como a inserção de conhecimento impreciso.

No entanto, numa perspetiva geral, consideramos que concluímos este trabalho com sucesso, dado que todos os requisitos propostos pela equipa docente foram cumpridos. Adicionalmente, este trabalho permitiu-nos, como grupo, aprofundar a nossa compreensão relativamente ao tema principal deste exercício: o conhecimento imperfeito.