

Typicality & ontology

Igor de Camargo e Souza Câmara

14 de Setembro de 2018

IME - USP

Segundo [1], há duas maneiras de entender o que significa *saber* um conceito.

1. Compreender as condições necessárias e suficientes que caracterizam o conceito. Essa noção vem sobretudo da filosofia.
2. Ser capaz de identificar elementos do conceito, ainda que imperfeitamente, baseando-se em heurísticas e produção de **protótipo**. Ideia que vem da psicologia e das ciências cognitivas.

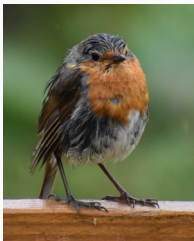
Um **protótipo** é uma instanciação abstrata de um conceito.

Um **protótipo** é uma instanciação abstrata de um conceito.

Pássaro \Rightarrow

Conceitos e protótipos

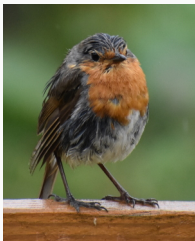
Um **protótipo** é uma instanciação abstrata de um conceito.



Pássaro \Rightarrow

Conceitos e protótipos

Um **protótipo** é uma instanciação abstrata de um conceito.



Pássaro \Rightarrow



Pássaro \nRightarrow

Conceito de pássaro provavelmente envolve uma série de características como: ter penas, voar, ter um determinado tamanho, construir ninho etc. Nem todas elas são necessárias para a caracterização do conceito de pássaro e também não são suficientes.

Conceitos e protótipos

Uma grande diferença entre o raciocínio a partir de protótipos e de conceitos “clássicos”: **composicionalidade**.

Conceitos “clássicos” são composicionais: $Pet\ fish = Pet \cap fish$.

Protótipos **não** são composicionais:

$$\begin{cases} Pet \Rightarrow small, warm, furry, have\ paws\ etc. \\ Fish \Rightarrow scaly, wet\ etc. \end{cases}$$

$Pet\ fish \Rightarrow \dots$

Conceitos e protótipos

Uma grande diferença entre o raciocínio a partir de protótipos e de conceitos “clássicos”: **composicionalidade**.

Conceitos “clássicos” são composicionais: $Pet\ fish = Pet \cap fish$.

Protótipos **não** são composicionais:

$\left\{ \begin{array}{l} Pet \Rightarrow \textit{small, warm, furry, have paws etc.} \\ Fish \Rightarrow \textit{scaly, wet etc.} \end{array} \right.$



$Pet\ fish \Rightarrow \dots$

Conceitos e protótipos

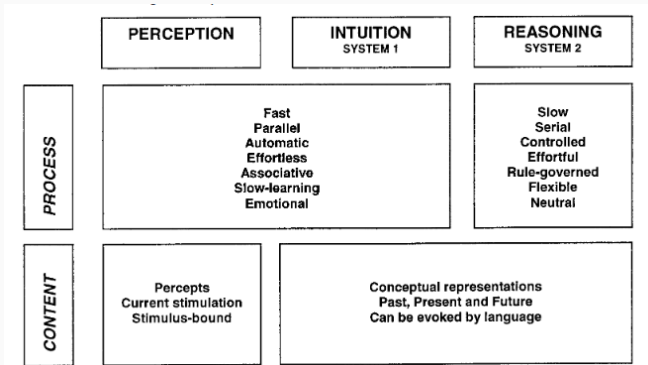
- Frixione [1] argumenta que mesmo em cenários onde há um conceito claro e o dominamos, é frequente usarmos a **tipicalidade** (o quanto um objeto se assemelha ao protótipo do conceito) para decidir instâncias particulares.
- Um exemplo é a água. Sabemos que, no nível molecular, água = H_2O . No entanto, para decidir se um objeto é ou não água, atentamos para outras propriedades, como ser líquido, incolor, inodoro etc.
- Isso pode ser útil para representação de sistemas baseados em conhecimento porque frequentemente nos depararemos com situações em que não é possível auferir as condições necessárias e suficientes para caracterizar um objeto. Por exemplo, não teremos como fazer uma análise química de um líquido para decidir se ele é ou não água.

Ideias para implementação

Mas como implementar essa outra maneira de entender a representação conceitual?

Frixione [1] sugere que olhemos para ensinamentos das ciências cognitivas. Em especial:

1. Distinção entre sistemas de raciocínio - dual process. Sistema 1 e 2.



[2]

2. Manter os efeitos prototípicos separados da representação de conceitos composicional.
 - 2.1 Representações composicionais e efeitos prototípicos exigem arquiteturas representacionais diferentes.

2. Manter os efeitos prototípicos separados da representação de conceitos composicional.
 - 2.1 Representações composicionais e efeitos prototípicos exigem arquiteturas representacionais diferentes.
3. Desenvolver representações de conceitos híbridas, prototípicas e baseadas em exemplos (*exemplar based*).
 - 3.1 Baseado em protótipos: conhecimento sobre categorias é guardado em protótipos - melhor representante de cada categoria. Protótipos são listas de valores associados a pesos.
 - 3.2 Baseado em exemplos: conceito é um conjunto de exemplos do conceito (i.e. gato é um conjunto de gatos vistos durante a vida).

Uma ideia é trabalhar com **linked data**, coordenando mais de um banco de dados.

Assim, haveria uma taxonomia com os conceitos clássicos usuais, baseada em DLs/Ontologias, e outra para dar conta do raciocínio prototípico (Frixione [1] sugere um *Prototype-Exemplar Learning Classifier*. PEL-C.)

Por exemplo: perguntar por uma fruta cítrica amarela (SPARQL).

```
SELECT? citrus
WHERE {?citrus :has colour : YELLOW .
}
```

Ser amarelo não é uma condição necessária para limão, mas amarelo constitui protótipo de limão ¹.

Um sistema híbrido poderia responder a *query* com um limão, portanto.

¹talvez não no Brasil...

Ching-man Au Yeung e Ho-fung Leung [3] propuseram um modelo formal para **typicality**, que eles diferenciam de **likeliness** (i.e. o quanto um objeto pertence a um conceito - o que permite graus de pertencimento, ao invés do par pertence/não pertence).

Um formalismo para Typicality

Vetor característico c_x (de um conceito x): vetor $(c_{x,1}, \dots, c_{x,n})$, com cada $c_{x,i} \in [0, 1]$ e n é o número total de propriedades.

Vetor de propriedades p_a de um individuo a - análogo ao anterior. Indica quanto o indivíduo a tem de cada propriedade.

Vetor prototípico t_x , de um conceito x é um vetor de números reais $(t_{x,1}, \dots, t_{x,n})$, onde cada $t_{x,i} \in [0, 1]$ é definido por:

$$t_x = \frac{1}{|S|} \sum_{s \in SU\{x\}} c_s$$

Onde S é o conjunto de *sub-conceitos* de x , c_s é a medida da propriedade naquele subconceito.

Ideia: quanto maior for o peso da propriedade nos subconceitos, maior será o peso no vetor prototípico.

Finalmente definimos **tipicalidade** de um objeto a para um conceito x como um número que resultante de uma função e que representa quanto a é considerado instância típica de x .

$$\tau_x : I \rightarrow [0, 1]$$

A função de tipicalidade obedece a 4 axiomas:

(A6) a tem tipicalidade máxima (1) para conceito x sse
 $t_{x,i} > 0 \rightarrow p_{a,i} = 1$ para todos i .

(A7) a tem tipicalidade mínima (0) para conceito x sse
 $t_{x,i} > 0 \rightarrow p_{a,i} = 0$ para todos i .

(A8) para conceito x e dois indivíduos a, b , se existe j tal que
 $t_{x,j} > 0, p_{a,j} > p_{b,j} \geq 0$ e para os demais $i \neq j, p_{a,i} = p_{b,i}$, então
 $\tau_x(a) > \tau_x(b)$.

(A9) Para dois conceitos, x, y e um indivíduo a . Se, para algum j tal que
 $t_{x,j} > t_{y,j} > 0$ vale que $p_{a,j} > 0$ e $t_{x,i} = t_{y,i}$ para os demais $i \neq j$, então
 $\tau_y(a) > \tau_x(a)$.

Um formalismo para Typicality

Uma função possível:

$$\tau_x(a) = \frac{p_a \cdot t_x}{\sum_{i=1}^n t_{x,i}}$$

Onde p_a é o vetor de características de a , t_x é o vetor do conceito x e $t_{x,i}$ denota cada uma das características do vetor t_x .

Um exemplo

Supor as características:

A	Animal	B	Has-Wings	C	Has-Feathers	D	Can-Fly
E	Eat-Seed	F	Has-Curved-Beak	G	Can-Sing	H	Can-Run

E os conceitos²:

Bird : [A]₁ [B]₁ [C]₁
Sparrow : [A]₁ [B]₁ [C]₁ [D]₁ [E]_{0.8}
Parrot : [A]₁ [B]₁ [C]₁ [D]₁ [F]₁
Robin : [A]₁ [B]₁ [C]₁ [D]₁ [G]_{0.8}
Ostrich : [A]₁ [B]₁ [C]₁ [H]_{0.9}

²O número à direita indica o quanto aquela característica tem força para o conceito em questão - é o número que aparece no vetor de características do conceito.

Um exemplo

Agora consideramos um indivíduo que é um avestruz (ostrich):
caracterizado por $p_o = (1, 1, 1, 0, 0, 0, 0, 0.8)$.

O vetor prototípico de pássaro é: $(1, 1, 1, 0.75, 0.25, 0.25, 0.25, 0.225)$.

Com tudo isso, calculamos a tipicidade dele em relação ao conceito de pássaro: $\tau_{Bird}(o) = 0.673$.

Ou seja, um avestruz não é tão representativo do conceito de pássaro.



M. Frixione and A. Lieto.

Representing and reasoning on typicality in formal ontologies.

In *Proceedings of the 7th International Conference on Semantic Systems*, pages 119–125. ACM, 2011.



D. Kahneman.

A perspective on judgment and choice: mapping bounded rationality.

American psychologist, 58(9):697, 2003.



C.-m. A. Yeung and H.-f. Leung.

Ontology with likeliness and typicality of objects in concepts.

In *International Conference on Conceptual Modeling*, pages 98–111. Springer, 2006.