

IMPLEMENTACE EFEKTIVNÍ SÉMANTICKÉ SÍTĚ PRO TVORBU BÁZE ZNALOSTÍ

Michal Hala



Obsah

- Knowledge Representation and Reasoning
- Transparentní intenzionální logika
- Communication and Artificial Reasoning with TIL Inference Machine
- Tell/Ask



Knowledge Representation and Reasoning (KRR)

- Cílem je vytvořit rozumně uvažující stroj -> **logický agent**
- Má 2 části
 - *Znalostní báze: uchovává fakta o světě ve formě formulí transparentní intenzionální logiky (TIL)*
 - *Inferenční mechanismus: umožňuje provádět operace nad znalostní bází -> zejména odpovídání na otázky*



Transparentní intenzionální logika (TIL)

- „**Transparent Intensional Logic** (frequently abbreviated as TIL) is a logical system created by [Pavel Tichý](#). Due to its rich *procedural semantics* TIL is in particular apt for the logical analysis of natural language. From the formal point of view, TIL is a hyperintensional, partial, typed lambda calculus.“ [wikipedia](#)
- „**Transparent Intensional Logic** is a logical theory developed with a view to logical analysis of sizeable fragments of primarily natural language.“
M. Duží, B. Jespersen and P. Materna: [Procedural Semantics for Hyperintensional Logic](#). Foundations and Applications of TIL. Springer, 2010.



TIL Příklad

- Věta: „Pes nese kost.“

- Formule v TILu:

$$\lambda w_1 \lambda t_2 (\exists i_3) (\exists x_4) (\exists i_5) ([\text{Does}_{w_1 t_2, i_5, [Imp_{w_1, x_4}]}] \wedge [\text{Pes}_{w_1 t_2, i_5}] \wedge x_4 = [\text{Nést, } i_3]_{w_1} \wedge [\text{Kost}_{w_1 t_2, i_3}])$$

- Ve světě w_1 a čase t_2 , existují individua i_3 a i_5 , a konstrukce x_4 pro něž platí:
 - i_5 vykonává x_4
 - i_5 má vlastnost *být psem*
 - x_4 je činnost *nesení* individua i_3
 - i_3 má vlastnost *být kostí*.



Communication and Artificial Reasoning with TIL InferenceMachine (CAT)

- Na FI vyvíjený KRR systém
- Skládá se z:
 - *Algoritmu normální translace (Normal Translation Algorithm, NTA): pro vstupní českou větu vystoupí jejímu významu ekvivalentní formuli v TIL*
 - *Inferenčního stroje pro TIL (TIL Inference Machine, TIM): buduje znalostní bázi a provádí inferenci*



Operace TIMu

- **Tell:** říci TIMu novou formuli v TILu, kterou by měl znát
- **Ask:** zeptat se TIMu, jestli zná odpověď na otázku



Tell 1

- Na vstup je poslána formule TIL v normálním tvaru
- Např. pro větu „Živí se planktonem.“ existuje ekvivalentní formule:

$$\lambda w_1 \lambda t_2 (\exists x_3) (\exists i_4) ([\text{Does}_{w_1 t_2}, \text{On}, [\text{Imp}_{w_1}, x_3]] \wedge [\text{plankton}_{w_1 t_2}, i_4] \wedge x_3 = [\text{živit}, i_4]_{w_1})$$

Zdrojový formát:

IV(w9/w)(IV(t10/t)([O(Exi/(o(o(o(o((ot)w))(o((ot)w)))))),IV(x7/(o(o((ot)w))(o((ot)w)))([O(Exi/(o(oi))),IV(i8/i)([O(and/(ooo)),[O(and/(ooo)),[[[O(Does/(((oi(o(o((ot)w))))t)w)),V(w9/w)],V(t10/t)],V(On/i),[[O(Imp/(((o(o((ot)w))(o(o((ot)w))(o((ot)w))))w)),V(w9/w)],V(x7/(o(o((ot)w))(o((ot)w)))]),[[[O(plankton-2/(((oi)t)w)),V(w9/w)],V(t10/t)],V(i8/i)],[O(=/((o(o(o((ot)w))(o((ot)w)))(o(o((ot)w))(o((ot)w)))))),V(x7/(o(o((ot)w))(o((ot)w)))]),[[[O(živit-0/(((o(o((ot)w))(o((ot)w)))w)i)),V(i8/i)],V(w9/w)]])])])



Tell 2

- TIM vytvoří z formule strom, obsahující 3 typy:
 - *Termy (mohou obsahovat termy, konstrukce, nebo hodnoty)*
 - *Konstrukce (charakterizované jménem a typem)*
 - *Hodnoty (charakterizované jménem a typem)*

```
Term {  
  Term {  
    Construction { name: plankton-2 , type: Type: (((oi)t)w) }  
    Value { name: w8 , type: Type: w }  
  }  
  Value { name: t9 , type: Type: t }  
}
```

Plankton_{w₁t₂}

$[[O(\text{plankton-2}/(((oi)t)w)),V(w_9/w)],V(t_{10}/t)]$



Tell 3

Reprezentace věty „Živí se planktonem“ v bázi znalostí (pretty-print bez typů)

```
Term {
  Construction { name: and }
  Term {
    Construction { name: and }
    Term {
      Term {
        Term {
          Construction { name: Does }
          Value { name: w8 }
        }
        Value { name: t9 }
      }
    }
    Value { name: On }
    Term {
      Term {
        Construction { name: Imp }
        Value { name: w8 }
      }
      Value { name: x10 }
    }
  }
}
```

```
Term {
  Term {
    Term {
      Construction { name: plankton-2 }
      Value { name: w8 }
    }
    Value { name: t9 }
  }
  Value { name: i11 }
}
Term {
  Construction { name: = }
  Value { name: x10 }
  Term {
    Term {
      Construction { name: živit-0 }
      Value { name: i11 }
    }
    Value { name: w8 }
  }
}
```



Ask 1

- Vstupem je Term (konjunkce Termů)
- TIM se snaží najít shody v bázi znalostí a popřípadě najít hodnoty odpovídající volným proměnným v otázce
- Rozlišujeme 2 typy otázek:
 - **Uzavřené:** zajímá nás, zda-li báze znalostí obsahuje term, který se shoduje s otázkou -> odpověď True/False
 - **Otevřené:** otázka obsahuje volnou proměnnou, kterou unifikujieme s podtermy shod v bázi znalostí



Ask 2

- Znalostní báze:

$\lambda w_1 \lambda t_2 (\exists x_3) (\exists i_4) ([\text{Does}_{w_1 t_2}, \text{On}, [\text{Imp}_{w_1}, x_3]] \wedge [\text{plankton}_{w_1 t_2}, i_4] \wedge x_3 = [\text{živit}, i_4]_{w_1})$

- Otázka 1:

$[\text{plankton}_{_}, _]$

Odpověď 1:

True

- Otázka 2:

$x :: [\text{plankton}_{_}, x]$

Odpověď 2:

i_4



THIS IS ALL



**THANKS FOR
YOU ATTENTION**

memegenerator.es

DĚKUJI ZA
POZORNOST

