Logika

Logika — rachunek kwantyfikatorów

Materiały pomocnicze do wykładu

dla

Studentek i Studentów

Informatyki

Wydział EAIiIB AGH

Antoni Ligęza

Materiały pomocnicze:

http://home.agh.edu.pl/~ligeza

Literatura

1. Mordechai Ben-Ari: Mathematical Logic for Computer Science (Logika matematyczna w informatyce). Springer-Verlag, London, 2001 (WN-T, Warszawa, 2005).
2. Kenneth A. Ross i Charles R. B. Wright: Discrete Mathematics (Matematyka dyskretna). WN PWN, 2013.
3. Antoni Ligęza: Logical Foundations for Rule-Based Systems. Springer-Verlag, Berlin, 2006. Wydawnictwo AGH, Kraków, 2005.
4. Michael R. Genesereth, Nils J. Nilsson: Logical Foundations of Artificial Intelligence. Morgan Kaufmann Publishers, Inc., Los Altos, California, 1987.
5. Zbigniew Huzar: Elementy logiki dla informatyków. Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 2007.
6. Stuart Russell, Peter Norvig: Artificial Intelligence. A Modern Approach. Pearson, 2010.
7. Marek Wójcik: Zasada rezolucji. Metoda automatycznego wnioskowania. PWN, Warszawa, 1991.
8. C. L. Chang and R. C. T. Lee: Symbolic Logic and Mechanical Theorem Proving. Academic Press, 1973.
9. Ronald J. Brachman and Hector J. Levesque: Knowledge Representation and Reasoning. Morgan Kaufmann, 2004.
10. Frank van Harmelen, Vladimir Lifschitz, Bruce Porter (Eds.): *Handbook of Knowledge Representation*. Elsevier B.V., Amsterdam, 2008. `http://ii.fmph.uniba.sk/~sefranek/kri/handbook/`

Zasoby sieciowe. Kurs w Stanford

Kurs logiki on-line Stanford:

`https://www.coursera.org/course/intrologic`

1. Wikipedia-pl: `http://pl.wikipedia.org/wiki/Logika_matematyczna`
2. Wikipedia-en: `http://en.wikipedia.org/wiki/Logic`
3. AI-Lab-Prolog: `http://ai.ia.agh.edu.pl/wiki/pl:prolog:prolog_lab`
4. EIS-KRR: `http://ai.ia.agh.edu.pl/wiki/pl:dydaktyka:krr:start`
5. ALI-home: `home.agh.edu.pl/~ligeza`
6. David Poole and Allen Mackworth: Artificial Intelligence. Foundations of Computational Agents. `http://artint.info/`
7. Ulf Nilsson and Jan Maluszynski: Logic, Programming and Prolog. `http://www.ida.liu.se/~ulfni/lpp/`

Ograniczenia logiki rachunku zdań

Przykłady zdań języka naturalnego nieformalizowalne w rachunku zdań:

Adam jest bratem Zygmunta. Jeżeli Adam jest bratem Zygmunta to Zygmunt jest bratem Adama.

Jeżeli blok A leży nad blokiem B, a B leży nad C, to A leży nad C

Istnieje prawdziwe szczęście.

Wszystko ma swoja cenę.

Każdy kogoś kocha.

Jeżeli X kocha Y to Y także kocha X.

Jeżeli jest przejazd z X do Y, a także z Y do Z, to jest przejazd z X do Z.

Jeżeli każdy kogoś kocha to każdy jest kochany przez kogoś.

Każdy Polak, który coś studiuje jest ambitny. Jan jest ambitny. Zatem Jan studiuje coś.

Kobieta zmienną jest.

Dobry fryzjer goli każdego, kto nie goli się sam.

Problemy — zadania — konstrukcje logiki

Problemy:

- Zmienne indywiduowe — definiujemy własności grup/klas/zbiorów obiektów,
- Kwantyfikatory — istnieje, istnieje dokładnie jeden, każdy, zawsze, wszędzie, wszystkie,...; ktoś, kiedyś, gdzieś, niektóre, większość,...
- Abstrakcja i uszczegóławianie — ograniczona indukcja i podstawienia; taksonomie,
- Obiekty strukturalnie złożone — `box(a)`, `book(autor,tytul,rok)`,
- Relacje — związki; predykaty,
- Ograniczenia — np. dysjunkcja,
- Wyrażenia warunkowe — implikacja,
- Wnioskowanie — stosowanie praw ogólnych do przypadków; anonimizacja - istnienie.

Ograniczenia logiki rachunku zdań

Język logiki rachunku zdań ma niską siłę ekspresji. Przykład:

Sokrates jest człowiekiem.

Każdy człowiek jest śmiertelny.

Sokrates jest śmiertelny.

```
man(plato).
man(socrates).

mortal(X):- man(X).
```

```
ojciec(jacek,wojtek).
ojciec(jacek,barbara).
ojciec(jan,ewa).
ojciec(jan,tomek).
ojciec(jan,jacek).
m(tomek).
m(jacek).
m(wojtek).
k(ewa).

brat(B,X) :-
   ojciec(P,X),
   ojciec(P,B),
   m(B),
   B \= X.
```

Alfabet i notacja

Definicja 1 *Relacja R to dowolny podzbiór iloczynu kartezjańskiego pewnej liczby zbiorów:*

$$R \subseteq X_1 \times X_2 \times \ldots \times X_n$$

Relacja jest zbiorem. Elementami reacji są n-krotki postaci $(x_1, x_2, \ldots, x_n)$. Zapis $R(x_1, x_2, \ldots, x_n)$ czytamy: zachodzi relacja R dla argumentów $x_1, x_2, \ldots, x_n$.

Dane sa parami rozłączne zbiory:

- C — zbiór stałych,
- V — zbiór zmiennych,
- F — zbiór symboli funkcyjnych,
- P — zbiór symboli predykatywnych.

Definicja 2 *Zbiór* termów $\mathtt{TER}$ *określony jest następująco:*

- *jeżeli c jest stałą, $c \in C$, to $c \in \mathtt{TER}$;*
- *jeżeli X jest zmienną, $X \in V$, to $X \in \mathtt{TER}$;*
- *jeżeli f jest n-arnym symbolem funkcyjnym, $f \in F$, oraz $t_1, t_2, \ldots, t_n$ są termami, to $f(t_1, t_2, \ldots, t_n) \in \mathtt{TER}$;*
- *elementy zbioru $\mathtt{TER}$ są generowane tylko regułami j.w.*

Liczbę n nazywamy arnością symbolu f. Zapisujemy:

$$f/n$$

Przykłady termów

Niech $a, b, c \in C$, $X, Y, Z \in V$, $f, g \in F$, oraz niech f i g będą symbolami funkcyjnymi o jednym ($f/1$) i dwóch argumentach ($g/2$).

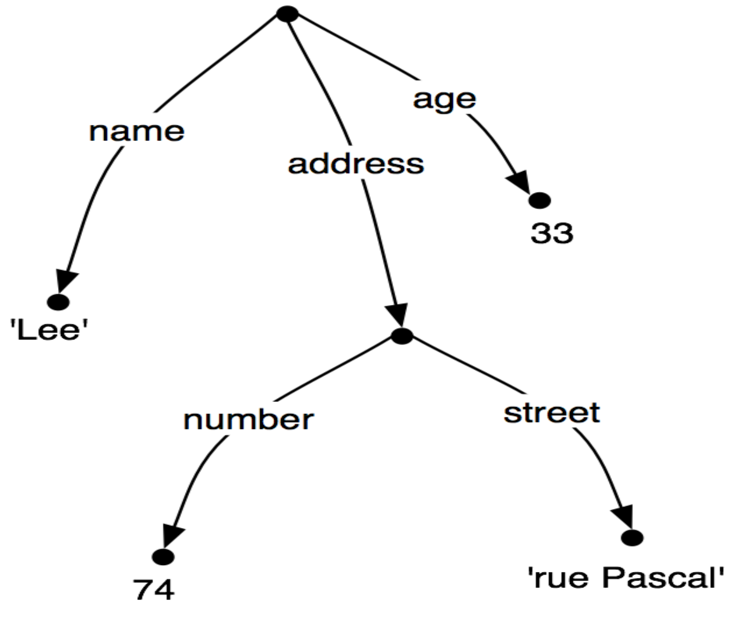
Następujące wyrażenia są termami:

- a, b, c;
- X, Y, Z;
- $f(a), f(b), f(c), f(X), f(Y), f(Z)$;
  $g(a, b), g(a, X), g(X, a), g(X, Y)$;
  $f(g(a, b)), g(X, f(X)), g(f(a), g(X, f(Z)))$.

Zbiór termów jest (dla jednej stałej i jednego symbolu funkcyjnego):

- nieskończony,
- przeliczalny.

Każdy term może być reprezentowany w postaci drzewa.



Rysunek 1: Struktura termu

Zastosowania termów

```
book (book_title,
      author(first_name,last_name),
      publisher_name,
      year_of_publication
     )
```

```
<book>
      <book_title> Learning XML </book_title>
      <author>
            <first_name> Erik </first_name>
            <last_name>  Ray  </last_name>
      </author>
      <publisher_name> O'Reilly & Associates, Inc. </publisher
      <year_of_publication> 2003 </year_of_publication>
</book>
```

```
book:
   title:     book_title
   author:    author_name
   publisher: publisher_name
   year:      year_of_publication
```

Zastosowania termów: c.d.

$$\frac{\frac{x}{y}}{\sqrt{1+\frac{x}{y}}},$$

```
\frac{
      \frac{x}{y}
     }
     {
      \sqrt{1+\frac{x}{y}}
     }
```

Listy, drzewa, funkcje, dane strukturalne, YAML,...

Formuły

Definicja 3 *Formuły atomiczne:* ATOM.
Jeżeli $p \in P$ *oraz* p/n*, a* $t_1, t_2, \ldots, t_n \in \mathrm{TER}$ *to:*

$$p(t_1, t_2, \ldots, t_n) \in \mathrm{ATOM}.$$

ATOM *jest zbiorem minimalnym spełniającym warunek j.w.*

Przykady formuł atomicznych:

- $p(a), p(b), q(a, a), q(a, c)$;
- $p(X), p(Y), q(X, X), q(X, Z)$;
- $p(f(a)), p(f(X)), q(f(g(a, b)), g(X, f(X))), q(g(f(a), g(X, f(Z))), a)$.

Termy a Formuły Atomiczne — czym się różnią?

Definicja 4 *Formuły:* FOR

- $ATOM \subseteq \mathrm{FOR}$*;*
- *if* Φ *is a formula,* $\Phi \in \mathrm{FOR}$*, then* $\neg(\Phi) \in \mathrm{FOR}$*;*
- *if* Φ *and* Ψ *are formulae,* $\Phi, \Psi \in \mathrm{FOR}$*, then* $(\Phi \wedge \Psi)$*,* $(\Phi \vee \Psi)$*,* $(\Phi \Rightarrow \Psi)$*,* $(\Phi \Leftrightarrow \Psi) \in \mathrm{FOR}$*;*
- *if* $\Phi \in \mathrm{FOR}$*,* X *denotes a variable, then* $\forall X(\Phi) \in \mathrm{FOR}$ *and* $\exists X(\Phi) \in \mathrm{FOR}$*;*
- *all the elements of* FOR *must be generated by applying the above rules.*

Notacja:

- kwantyfikator ogólny ograniczony: $\forall X \in D_X$, $\forall_{X \in D_X}$,
- istnieje dokładnie jeden element: $\exists ! X$,
- $\forall$ — uogólnienie koniunkcji: $\bigwedge$,
- $\exists$ — uogólnienie dysjunkcji: $\bigvee$,

Uogólnienie koniunkcji:

$$\forall X : p(X) \quad \stackrel{?}{\equiv} \quad p(a) \wedge p(b) \wedge p(c) \wedge \ldots$$

Uogólnienie dysjunkcji:

$$\exists X : p(X) \quad \stackrel{?}{\equiv} \quad p(a) \vee p(b) \vee p(c) \vee \ldots$$

Formuły bez zmiennych to tzw. instancje podstawowe.

Rola zmiennych, zmienne wolne

Zmienne pełnią następujące role:

- names/references to unknown objects — definiują dowiązania do kwantyfikatorów,
- placeholders — rezerwują miejsce dla obiektów (chwilowo nieokreślonych),
- coreference constraints — definiują ograniczenia koreferencyjne.

Zmienne w formula mogą być:

- związane — w zakresie/zasięgu kwantyfikatora,
- wolne — poza zasięgiem kwantyfikatora,
- wolne i związane; mówimy wówczas o *wystąpieniach zmiennej*.

Definicja 5 *Zmienne wolne (w formule):* $FV()$

- *if* $t \in V$ *then* $FV(t) = \{t\}$*;*
- *if* $t \in C$ *then* $FV(t) = \emptyset$*;*
- *if* $t = f(t_1, t_2, \ldots, t_n) \in \mathrm{TER}$ *then* $FV(t) = FV(t_1) \cup FV(t_2) \cup \ldots \cup FV(t_n)$*;*
- *if* $q = p(t_1, t_2, \ldots, t_n) \in ATOM$ *then* $FV(q) = FV(t_1) \cup FV(t_2) \cup \ldots \cup FV(t_n)$*;*
- $FV(\neg\Phi) = FV(\Phi)$*;*
- $V(\Phi \diamond \Psi) = FV(\Phi) \cup FV(\Psi)$ *for any* $\diamond \in \{\wedge, \vee, \Rightarrow, \Leftrightarrow\}$*;*
- $FV(\nabla X(\Phi)) = FV(\Phi) \setminus \{X\}$ *for* $\nabla \in \{\forall, \exists\}$*.*

Uniwersum, interpretacja i wartościowanie

Określenie semantyki wymaga zdefiniowania:

- D — niepusty zbiór uniwersum,
- I — interpretacja — mapowanie stałych, funkcji i predykatów na odpowiednie obiekty związane z D,
- v — wartościowanie — przypisanie wartości zmiennym wolnym.

Definicja 6 *Wartościowanie v:*

$$v\colon V \to D$$

Definicja 7 *Interpretacja I:*

- *for any constant $c \in C$, $I(c) \in D$;*
- *for any free occurrence of variable $X \in V$, $I(X) = v(X)$, where $v(X) \in D$;*
- *for any function symbol $f \in F$ of arity n, $I(f)$ is a function of the type*

  $$I(f)\colon D^n \to D;$$

- *for any predicate symbol $p \in P$ of arity n, $I(p)$ is a relation such that*

  $$I(p) \subseteq D^n;$$

- *for any term $t \in \mathrm{TER}$, such that $t = f(t_1, t_2, \ldots, t_n)$,*

  $$I(t) = I(f)(I(t_1), I(t_2), \ldots, I(t_n)).$$

Semantyka rachunku predykatów

Semantyka — przypisanie znaczenia. Dane:

- D — niepusty zbiór uniwersum (skończony, przeliczalny, nieskończony-nieprzeliczalny),
- I — mapowanie stałych, funkcji i predykatów na odpowiednie obiekty związane z D (elementy, funkcje, relacje),
- v — wartościowanie — przypisanie wartości zmiennym wolnym (lub formuły zamknięte).

Definicja 8 *1.* $\models_{I,v}$ $p(t_1, t_2, \ldots, t_n)$ *iff (if and only if)* $(I(t_1), I(t_2), \ldots, I(t_n)) \in I(p)$ *(recall that* $I(X) = v(X)$ *for any free variable* $X \in VAR$*;*

2. $\models_{I,v} \neg\Phi$ *iff* $\not\models_{I,v} \Phi$*;*

3. $\models_{I,v} \Phi \wedge \Psi$ *iff both* $\models_{I,v} \Phi$ *and* $\models_{I,v} \Psi$*;*

4. $\models_{I,v} \Phi \vee \Psi$ *iff* $\models_{I,v} \Phi$ *or* $\models_{I,v} \Psi$*;*

5. $\models_{I,v} \Phi \Rightarrow \Psi$ *iff* $\not\models_{I,v} \Phi$ *or* $\models_{I,v} \Psi$*;*

6. $\models_{I,v} \Phi \Leftrightarrow \Psi$ *iff* $\models_{I,v} \Phi$ *and* $\models_{I,v} \Psi$*, or,* $\not\models_{I,v} \Phi$ *and* $\not\models_{I,v} \Psi$*;*

7. $\models_{I,v} \forall X \Phi$ *iff for any* $d \in D$ *and any variable assignment* u *such that* $u(X) = d$ *and* $u(Y) = v(Y)$ *for any* $Y \neq X$*, there is* $\models_{I,u} \Phi$*;*

8. $\models_{I,v} \exists X \Phi$ *iff there exists* $d \in D$ *such that for variable assignment* u *defined as* $u(X) = d$ *and* $u(Y) = v(Y)$ *for any* $Y \neq X$*, there is* $\models_{I,u} \Phi$*.*

Komentarze i uwagi

Zwykle rozważamy formuły zamknięte — bez zmiennych wolnych.

Jeżeli w formule są zmienne wolne — to możemy potraktować je jak kwantyfikowane uniwersalnie albo egzystencjalnie. Przykład: czy $p(X) \vee \neg p(Y)$ uznajemy za tautologię?. Dlatego najlepiej unikać zmiennych wolnych w formułach.

Dla uniknięcia nieporozumień — dokonujemy konsekwentnego przemianowania (zmiany nazw) zmiennych, tak aby każde pojedyncze związanie zmiennej kwantyfikatorem odnosiło się do wystąpień tej zmiennej tylko w obrębie zakresu/obszaru działania tego kwantyfikatora.

Definicja 9 *Definicja logicznej konsekwencji:*

A formula H is a logical consequence *of formulae $\Delta_1, \Delta_2, \ldots, \Delta_n$ if and only if for any interpretation I satisfying $\Delta_1 \wedge \Delta_2 \wedge \ldots \wedge \Delta_n$ for some variable assignment v there exists a variable assignment u, such that H is satisfied under interpretation I and variable assignment u.*

Pytanie: ile jest możliwych interpretacji dla następujących formuł:

- p,
- $p \wedge q$, $p \vee q$, $p \Rightarrow q$,
- $p(a)$,
- $p(f(a)))$,
- $\forall X : p(X)$,
- $\exists X : p(X)$.

Interpretacja Herbranda

Definicja 10 *Uniwersum Herbranda:*

Let $H_0 = C(\Delta)$, i.e. H_0 contains all the constants occurring in some set of formulae Δ (if $C(\Delta) = \emptyset$ then one defines H_0 in such a way that it contains a single arbitrary symbol, say $H_0 = \{c\}$).

Now, for $i = 0, 1, 2, \ldots,$ let $H_{i+1} = H_i \cup \{f(t_1, t_2, \ldots, t_n) \colon f \in F(\Delta)$ and $t_1, t_2, \ldots, t_n \in H_i\}$ (where the arity of f is n). Then H_∞ is called the Herbrand Universe *of Δ.*

Definicja 11 *Baza Herbranda:*

*Let Δ be a set of formulae and let **H** be the Herbrand universe of Δ. A set $B_H = \{p(h_1, h_2, \ldots, h_n) \colon h_1, h_2, \ldots, h_n \in$ **H**$, p \in P(\Delta)\}$ (where the arity of p is n) is called the* Herbrand base *or the* atom set *of Δ.*

Definicja 12 *Interpretacja Herbranda:*

*Let Δ be a set of formulae and let **H** be the Herbrand universe of Δ. Any interpretation I_H is called a* Herbrand interpretation *(**H**-interpretation) if the following conditions are satisfied:*

- *for any constant $c \in$ **H**, $I_H(c) = c$;*
- *for any n-ary functional symbol $f \in F(\Delta)$, and any $h_1, h_2, \ldots, h_n \in$ **H**,*

$$I_H(f) \colon (h_1, h_2, \ldots, h_n) \to f(h_1, h_2, \ldots, h_n).$$

Twierdzenie Herbranda

Twierdzenie 1 *Twierdzenie Herbranda I: Zbiór klauzul S (formuła w postacji CNF, wszystkie zmienne kwantyfikowane uniwersalnie) jest niespełnialna wtw. gdy jest on niespełnialny przy każdej interpretacji Herbranda (nie ma modelu).*

Twierdzenie 2 *Twierdzenie Herbranda II: Zbiór klauzul S (formuła w postacji CNF, wszystkie zmienne kwantyfikowane uniwersalnie) jest niespełnialna wtw. gdy istnieje skończony i niespełnialny zbiór S' instancji podstawowych zdań z S.*

Reguły przekształcania formuł FOPC

Oznaczenie: $\Phi[X]$ — jawne zdefiniowanie wystąpienia zmiennej X w formule Φ.

Reguły bazowe dla kwantyfikatorów:

- $\forall X \Phi[X] \wedge \Psi \equiv \forall X (\Phi[X] \wedge \Psi)$,
- $\forall X \Phi[X] \vee \Psi \equiv \forall X (\Phi[X] \vee \Psi)$,
- $\exists X \Phi[X] \wedge \Psi \equiv \exists X (\Phi[X] \wedge \Psi)$,
- $\exists X \Phi[X] \vee \Psi \equiv \exists X (\Phi[X] \vee \Psi)$.

Odpowiedniki praw De Morgana:

- $\neg(\forall X \Phi[X]) \equiv \exists X (\neg\Phi[X])$,
- $\neg(\exists X \Phi[X]) \equiv \forall X (\neg\Phi[X])$.

Reguły dystrybucji kwantyfikatorów:

- $\forall X \Phi[X] \wedge \forall X \Psi[X] \equiv \forall X (\Phi[X] \wedge \Psi[X])$,
- $\exists X \Phi[X] \vee \exists X \Psi[X] \equiv \exists X (\Phi[X] \vee \Psi[X])$.

Uzupełniające reguły dystrybucji kwantyfikatorów wykorzystujące przemianowanie zmiennych:

- $\forall X \Phi[X] \vee \forall X \Psi[X] \equiv \forall X \Phi[X] \vee \forall Y \Psi[Y] \equiv \forall X \forall Y (\Phi[X] \vee \Psi[Y])$,
- $\exists X \Phi[X] \wedge \exists X \Psi[X] \equiv \exists X \Phi[X] \wedge \exists Y \Psi[Y] \equiv \exists X \exists Y (\Phi[X] \wedge \Psi[Y])$.

Pozostałe reguły przekształceń równoważnych — analogiczne do rachunku zdań

- $\neg\neg\phi \equiv \phi$ — prawo (eliminacji) podwójnej negacji,
- $\phi \wedge \psi \equiv \psi \wedge \phi$ — przemienność koniunkcji,
- $\phi \vee \psi \equiv \psi \vee \phi$ — przemienność dysjunkcji,
- $(\phi \wedge \varphi) \wedge \psi \equiv \phi \wedge (\varphi \wedge \psi)$ — łączność koniunkcji,
- $(\phi \vee \varphi) \vee \psi \equiv \phi \vee (\varphi \vee \psi)$ — łączność dysjunkcji,
- $(\phi \vee \varphi) \wedge \psi \equiv (\phi \wedge \psi) \vee (\varphi \wedge \psi)$ — rozdzielność koniunkcji względem dysjunkcji,
- $(\phi \wedge \varphi) \vee \psi \equiv (\phi \vee \psi) \wedge (\varphi \vee \psi)$ — rozdzielność dysjunkcji względem koniunkcji,
- $\phi \wedge \phi \equiv \phi$ — idempotencja koniunkcji (pochłanianie),
- $\phi \vee \phi \equiv \phi$ — idempotencja dysjukcji (pochłanianie),
- $\phi \wedge \bot \equiv \bot$, $\phi \wedge \top \equiv \phi$ — prawo identyczności,
- $\phi \vee \bot \equiv \phi$, $\phi \vee \top \equiv \top$ — prawo identyczności,
- $\phi \vee \neg\phi \equiv \top$ — prawo wyłączonego środka,
- $\phi \wedge \neg\phi \equiv \bot$ — prawo sprzeczności,
- $\neg(\phi \wedge \psi) \equiv \neg(\phi) \vee \neg(\psi)$ — prawo De Morgana,
- $\neg(\phi \vee \psi) \equiv \neg(\phi) \wedge \neg(\psi)$ — prawo De Morgana,
- $\phi \Rightarrow \psi \equiv \neg\psi \Rightarrow \neg\phi$ — prawo kontrapozycji,
- $\phi \Rightarrow \psi \equiv \neg\phi \vee \psi$ — zasada eliminacji implikacji.

Postać normalna

Definicja 13 *Definicja postaci normalnej formuły:*

Formuła α jest w postaci normalnej *wtw, gdy jest ona postaci*

$$(Q_1X_1)\ldots(Q_nX_n)(\mu),$$

gdzie (Q_kX_k) (dla $1 \leq k \leq n$) to albo $(\forall X_k)$ albo $(\exists X_k)$, zaś μ - formuła bez kwantyfikatorów.

Metoda sprowadzania danej formuły do postaci normalnej:

1. wyeliminować spójniki $\Leftrightarrow$ i $\Rightarrow$ za pomocą przekształceń równoważnych,
2. wprowadzić znak negacji bezpośrednio symbole atomowe za pomocą prawa podwójnej negacji oraz praw De Morgana (również dla kwantyfikatorów),
3. przesunąć kwantyfikatory do przedrostka za pomocą reguł dystrybucji kwantyfikatorów.

Przykład:

Sprowadź $\exists Z \forall X((r(Z) \wedge p(X)) \Rightarrow \exists Y q(X,Y))$ do postaci normalnej.

$\exists Z \forall X((r(Z) \wedge p(X)) \Rightarrow \exists Y q(X,Y)) \equiv \exists Z \forall X(\neg(r(Z) \wedge p(X)) \vee \exists Y q(X,Y)) \equiv$
$\exists Z \forall X((\neg r(Z) \vee \neg p(X)) \vee \exists Y q(X,Y)) \equiv \exists Z \forall X \exists Y(\neg r(Z) \vee \neg p(X) \vee q(X,Y))$

Postać standardowa Skolema

Definicja 14 *Definicja postaci standardowej Skolema:*

Formuła α jest w postaci standardowej Skolema *wtw, gdy jest w postaci normalnej, w jej przedrostku nie ma kwantyfikatorów szczegółowych, zaś jej matryca jest w postaci CNF.*

Metoda sprowadzania danej formuły do postaci standardowej Skolema:

1. sprowadzić formułę do postaci normalnej,
2. sprowadzić matrycę μ do postaci CNF,
3. przekształcić przedrostek $(Q_1X_1)\ldots(Q_nX_n)$, powtarzając następujące czynności aż do usunięcia wszystkich kwantyfikatorów szczegółowych:
   - jeśli w przedrostku przed $Q_rX_r = \exists_r X_r$ nie występują kwantyfikatory ogólne, to w matrycy μ wymienimy wystąpienia X_R na nową stałą np. c, która nie występuje w μ oraz usuwamy $\exists_r X_r$ z przedrostka,
   - jeśli w przedrostku przed $Q_rX_r = \exists_r X_r$ występuje jeszcze s kwantyfikatorów ogólnych $(\forall_{j_1} X_{j_1})\ldots(\forall_{j_s} X_{j_s})$, gdzie $(1 \leq j_1 \leq j_s \leq r)$ to w matrycy μ wymienimy wystąpienia X_R na nowy symbol funkcyjny s-argumentowy np. $f(X_{j_1},\ldots,X_{j_s})$, który nie występuje w μ oraz usuwamy $\exists_r X_r$ z przedrostka.

Przykład:

$\exists Z \forall X \exists Y(\neg r(Z) \vee \neg p(X) \vee q(X,Y)) \equiv \forall X(\neg r(c) \vee \neg p(X) \vee q(X, f(X)))$,

gdzie c jest stałą zaś f jest symbolem funkcyjnym.

Postać klauzulowa $\{\neg r(c) \vee \neg p(X) \vee q(X, f(X))\}$

KRR — logika — co i jak?

- dowodzenie twierdzeń, weryfikacja logicznej konsekwencji:

$$\Delta \models H;$$

- prowadzenie rozumowań — wywód:

$$\Delta \vdash H;$$

- badanie spełnialności (SAT); poszukiwanie modelu:

$$\models_I H;$$

- badanie niespełnialności:

$$\not\models_I H;$$

- badanie zupełności – weryfikacja tautologii:

$$\models H$$

- badanie poprawności reguł wnioskowania:

$$(\Delta \vdash H) \quad \longrightarrow \quad (\Delta \models H)$$

- badanie zupełności reguł wnioskowania:

$$(\Delta \models H) \quad \longrightarrow \quad (\Delta \vdash H)$$

Dowodzenie twierdzeń

Twierdzenie 3 *Jeżeli $\Delta_1, \Delta_2, \ldots \Delta_n$ są formułami logicznymi (nazywanymi aksjomatami), formuła Ω (nazywana hipotezą lub konkluzją) jest ich logiczną konsekwencją wtw. gdy formuła $\Delta_1 \wedge \Delta_2 \wedge \ldots \Delta_n \Rightarrow \Omega$ jest tautologią.*

Twierdzenie 4 *Jeżeli $\Delta_1, \Delta_2, \ldots \Delta_n$ są formułami logicznymi (nazywanymi aksjomatami), formuła Ω (nazywana hipotezą lub konkluzją) jest ich logiczną konsekwencją wtw. gdy formuła $\Delta_1 \wedge \Delta_2 \wedge \ldots \Delta_n \wedge \neg\Omega$ jest sprzeczna.*

Problem dowodzenia twierdzeń ma postać: mając dane aksjomaty $\Delta_1, \Delta_2, \ldots \Delta_n$ uznane za prawdziwe wykazać prawdziwość hipotezy Ω. Tak więc należy wykazać, że:

$$\Delta_1 \wedge \Delta_2 \wedge \ldots \Delta_n \models \Omega$$

Metody dowodzenia twierdzeń:

- sprawdzanie wszystkich możliwych interpretacji (wada: duża złożoność obliczeniowa),
- dowód wprost – korzystając z aksjomatów i reguł dowodzenia generujemy nowe formuły aż do uzyskania formuły Ω,
- dowodzenie tautologii – korzystając z Tw.1 dowodzimy, że formuła $\Delta_1 \wedge \Delta_2 \wedge \ldots \Delta_n \Rightarrow \Omega$ jest tautologią,
- dowód nie wprost – to dowód twierdzenia przeciwstawnego, równoważnego danemu. Polega na dowodzeniu twierdzenia postaci $\neg\Omega \Rightarrow \neg(\Delta_1 \wedge \Delta_2 \wedge \ldots \Delta_n)$.
- dowód przez sprowadzenie do sprzeczności; korzystają z Tw.2, polega na wykazaniu sprzeczności formuły: $\Delta_1 \wedge \Delta_2 \wedge \ldots \Delta_n \wedge \neg\Omega$.

Ważniejsze reguły wnioskowania (Fitch)

- AND Introduction (AI):

$$\frac{\phi_1, \ldots, \phi_n}{\phi_1 \wedge \ldots \wedge \phi_n}$$

- AND Elimination (AE):

$$\frac{\phi_1 \wedge \ldots \wedge \phi_n}{\phi_i}$$

- OR Introduction (OI):

$$\frac{\phi_i}{\phi_1 \wedge \ldots \wedge \phi_n}$$

- OR Elimination (OE):

$$\frac{\phi_1 \vee \ldots \vee \phi_n, \phi_1 \Rightarrow \psi, \ldots \phi_n \Rightarrow \psi}{\psi}$$

- Negation Introduction (NI):

$$\frac{\phi \Rightarrow \psi, \phi \Rightarrow \neg\psi}{\neg\phi}$$

- Negation Elimination (NE):

$$\frac{\neg\neg\phi}{\phi}$$

- Implication Introduction (II):

$$\frac{\phi \vdash \psi}{\phi \Rightarrow \psi}$$

- Implication Elimination (IE):

$$\frac{\phi, \phi \Rightarrow \psi}{\psi}$$

- Equivalence Introduction (EI),
- Equivalence Elimination (EE)

reguły dla kwantyfikatorów (Fitch)

Universal Introduction (UI)

$$\frac{\Phi}{\forall X : \Phi}$$

Universal Elimination (UE)

$$\frac{\forall X : \Phi[X]}{\Phi[t]}$$

gdzie $t \in \mathrm{TER}$.

Existential Introduction (EI)

$$\frac{\Phi[t]}{\exists X : \Phi[X]}$$

Existential Elimination (EE)

$$\frac{\exists X : \Phi[X], \quad \forall Y : (\Phi[Y] \Rightarrow \Psi)}{\Psi}$$

gdzie: zmienna Y nie występuje w formule Ψ.

Dowodzenie Metodą Rezolucji

1. Zamiast dowodzić, że:

$$\{\Delta_1, \Delta_2, \ldots, \Delta_n\} \models H$$

   dowodzimy niespełnialności

$$\{\Delta_1, \Delta_2, \ldots, \Delta_n\} \cup \{\neg H\}$$

2. Wszystkie kwantyfikatory przesuwamy przed formułę.
3. Formułę bez kwanytyfikatorów przekształcamy do postaci CNF.
4. Eliminujemy kwantyfikatory egzystencjalne, zastępując zmienne termami; pozostają tylko kwantyfikatory ogólne (można je pominąć, gdyż wszystkie pozostałe zmienne są kwantyfikowane ogólnie).
5. W efekcie dostajemy zbiór zdań — jest to tzw. S-postać.
6. Stosując metodę rezolucji wyprowadzamy zdanie puste.

Reguła rezolucji dla zdań $C_1 = \phi \vee q_1$ oraz $C_2 = \varphi \vee \neg q_2$; σ jest podstawieniem unifikującym (mgu):

$$\frac{\phi \vee q_1, \varphi \vee \neg q_2}{\phi\sigma \vee \varphi\sigma}$$

Reguła faktoryzacji:

$$\frac{C}{C\theta}$$

Reguła faktoryzacji jest niezbędna dla przypadków typu:

$$\{p(X) \vee p(Y), \neg p(U) \vee \neg p(V)\}$$

Paradoks Fryzjera

There is a barber who was ordered to shave anyone who does not shave himself. Should he shave himself or not?

Formalizacja problemu w FOPC:

A. $\forall X \neg \mathrm{shaves}(X, X) \;\Rightarrow\; \mathrm{shaves}(\mathrm{barber}, X)$ — anyone who does not shave himself is shaved by the barber.

B. $\forall Y \mathrm{shaves}(\mathrm{barber}, Y) \Rightarrow \neg \mathrm{shaves}(Y, Y)$ — anyone who is not shaved by the barber shaves himself.

Transformacja do S-postaci:

- $C_1 = \mathrm{shaves}(X, X) \vee \mathrm{shaves}(\mathrm{barber}, X)$,
- $C_2 = \neg \mathrm{shaves}(\mathrm{barber}, Y) \vee \neg \mathrm{shaves}(Y, Y)$.

Niech $\theta = \{X/\mathrm{barber}, Y/\mathrm{barber}\}$.

$$C_1\theta = \mathrm{shaves}(\mathrm{barber}, \mathrm{barber})$$

$$C_2\theta = \neg \mathrm{shaves}(\mathrm{barber}, \mathrm{barber})$$

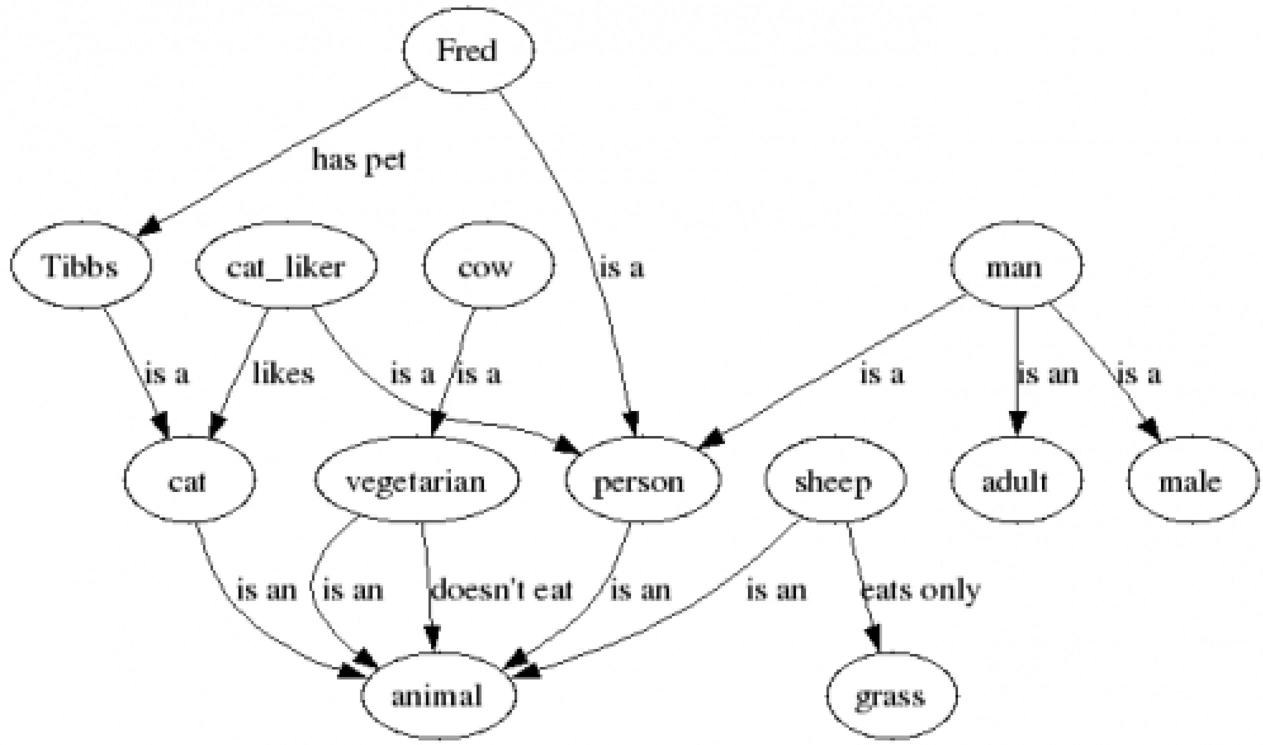
W wyniku rezolucji mamy:

$$\frac{\mathrm{shaves}(\mathrm{barber}, \mathrm{barber}), \neg \mathrm{shaves}(\mathrm{barber}, \mathrm{barber})}{\perp}.$$

Metoda tablic semantycznych

Wykład 10.

Logiki Deskrypcyjne



Rysunek 2: Example semantic net

Logiki Deskrypcyjne

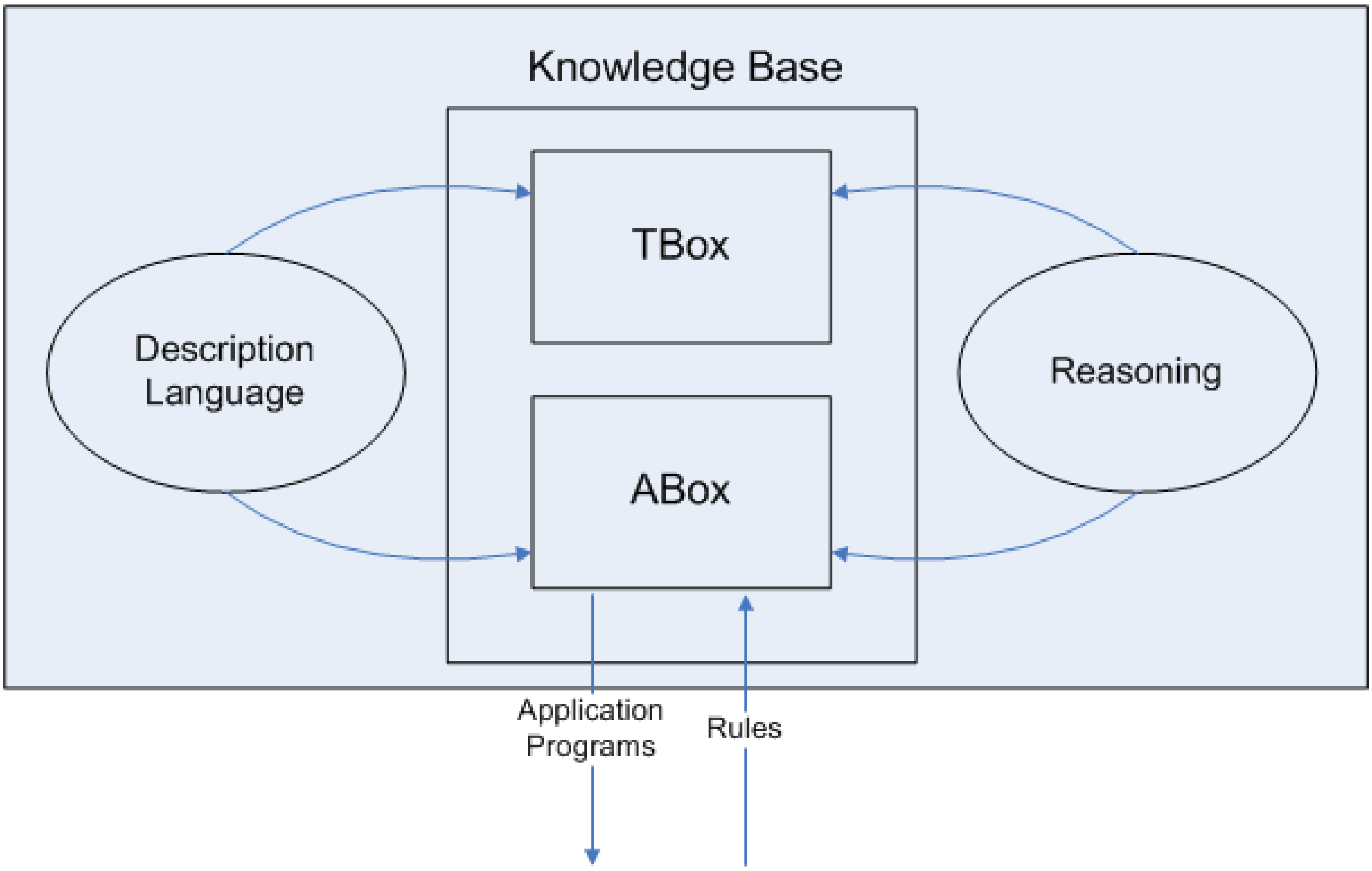
Logiki deskrypcyjne (opisowe) — formalny sposób zapisu wiedzy (taksonomie, relacje, ograniczona kwantyfikacja: ontologie; umożliwiają ograniczone wnioskowanie.

- Logiki deskrypcyjne (opisowe) (ang. Description Logics, DL) są rodziną formalizmów reprezentacji wiedzy.
- Elementami reprezentacji są pojęcia (klasy) i instancje oraz role (relacje) (obiekty ((Nazwy w nawiasach używane są zwykle w ontologiach zapisanych w języku OWL, opartym na formalizmie DL))).
- Logiki opisowe są koncepcyjnie powiązane z sieciami semantycznymi (ang. semantic networks) i ramami (ang. frames), jednak w przeciwieństwie do nich, przez swoje powiązanie z logiką pierwszego rzędu, posiadaję formalnie zdefiniowaną semantykę i zapewniają możliwość automatycznego wnioskowania.
- Intuicyjnie można powiedzieć, że logiki opisowe łączą paradygmat obiektowy (ramy, sieci semantyczne) z logiką (rachunek predykatów, logika 1. rzędu).

Wybrane fragmenty wiedzy zapisane w logice opisowej:

- $Fred : person$, $Tibbs : cat$, $(Tred, Tibbs) : has\_pet$
- $man \equiv person \sqcap adult \sqcap male$, $cat\_liker \equiv person \sqcap \exists likes.cat$
- $(cat\_liker, cat) : likes$, $(sheep, grass) : eats\_only$
- $cat \sqsubseteq animal$, $sheep \sqsubseteq animal \sqcap \forall eats.grass$

Logiki Deskrypcyjne: Taksonomie i Relacje



Rysunek 3: Komponenty bazy DL

Zastosowanie logik opisowych

Reprezentacja własności instancji (obiektu):

- przynależność obiektu do klasy (ang. concept assertions), np:
  - $Fred : person$ - Fred jest osobą
  - $Tibbs : cat$ - Tibbs jest kotem
- relacja między dwoma obiektami:
  - $(Fred, Tibbs) : has\_pet$ - Fred ma zwierzę, którym jest Tibbs

Definicje i własności pojęć:

- definicje pojęć (warunki konieczne i wystarczające), np.
  - $man \equiv person \sqcap adult \sqcap male$ - Mężczyzna to dorosła osoba rodzaju męskiego
  - $cat\_liker \equiv person \sqcap \exists likes.cat$ - Miłośnik kotów to osoba, która lubi (jakiegoś) kota
- relacje między pojęciami (klasami)
  - $(cat\_liker, cat) : likes$ - (każdy) miłośnik kotów lubi (jakiegoś) kota
  - $(sheep, grass) : eats\_only$ - (każda) owca je tylko trawę
- aksjomaty
  - $cat \sqsubseteq animal$ (każdy) kot jest zwierzęciem (hierarchia pojęć)
  - $sheep \sqsubseteq animal \sqcap \forall eats.grass$ owce to zwierzęta, które jedzą tylko trawę (warunek konieczny, ale nie wystarczający)

Bazowy język DL

Podstawowy język DL

- W języku logiki opisowej tworzymy opisy — formalizujemy ontologie.
- Podstawowe elementy języka to: atomicze pojęcia i atomiczne role.
- Złożone opisy tworzy się indukcyjnie za pomocą konstruktorów.
- Poszczególne języki DL różnią się między sobą zbiorem dopuszczalnych konstruktorów
- Najprostszy język to AL (ang. Attributive Language)

Składnia DL/AL

- atomiczne pojęcia $(A, B, ...)$
- atomiczne role $(R, S, ...)$
- opisy $(C, D, ...)$; mogą nimi być:
  - A - pojęcie atomiczne
  - $\top$ - //top concept//, pojęcie uniwersalne oznaczające 'wszystko'
  - $\bot$ - //bottom concept//, pojęcie puste, oznaczające 'nic'
  - $\neg A$ - negacja
  - $C \sqcap D$ - koniunkcja
  - $\forall R.C$ - kwantyfikator uniwersalny: "dla każdego"
  - $\exists R.\top$ - kwantyfikator egzystencjalny/szczegółowy: "istnieje"

Semantyka DL/AL

Semantyka DL/AL — Semantyka zdefiniowana jest poprzez //interpretację// składającą się z:

- dziedziny intepretacji: $\Delta^{\mathcal{I}}$ - niepustego zbioru, na który mapowane są symbole i relacje
- funkcji interpretacji, która przypisuje:
  - każdemu atomicznemu pojęciu zbiór: $A \to A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
  - każdej atomicznej roli relację binarną: $R^{\mathcal{I}} \to \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$

Konstruktor	Składnia	Semantyka
pojęcie atomiczne (atomic concept)	A	$A^{\mathcal{I}} = A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
atomiczna rola (atomic role)	R	$R^{\mathcal{I}} = R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
pojęcie uniwersalne (universal concept)	$\top$	$\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$
pojęcie puste (bottom concept)	$\bot$	$\bot^{\mathcal{I}} = \emptyset$
(**atomic** negation)	$\neg A$	$(\neg A)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus A^{\mathcal{I}}$
koniunkcja/przecięcie (intersection)	$C \sqcap D$	$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$
ograniczenie wartości (value restriction)	$\forall R.C$	$(\forall R.C)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \vert \forall b, (a,b) \in R^{\mathcal{I}} \to b \in C^{\mathcal{I}}\}$
ograniczony kwantyfikator egzystencjalny (limited existential quantification)	$\exists R.\top$	$(\exists R.\top)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \vert \exists b, (a,b) \in R^{\mathcal{I}}\}$

Przykłady formalizacji

Przykład:

- pojęcia atomiczne: //Person//, //Female//, //Elephant// ++|uwaga: dopóki nie zapisze się tego explicite, pomiędzy pojęciami nie występują żadne relacje. Są to po prostu oznaczenia jakichś zbiorów++
  - osoba rodzaju żeńskiego: $Person \sqcap Female$
  - słonica: $Elephant \sqcap Female$
  - osoba, które nie jest rodzaju żeńskiego: $Person \sqcap \neg Female$
- atomiczna rola: //hasChild//
  - osoba, która ma (jakieś) dziecko/dzieci: $Person \sqcap \exists hasChild.\top$
  - osoba, której wszystkie dzieci są rodzaju zeńskiego $Person \sqcap \forall hasChild.Female$
  - osoba bezdzietna $Person \sqcap \forall hasChild.\bot$

Rodzina języków DL

Poszczególne języki DL rozróżniamy poprzez konstruktory, które dopuszczają.

Przykładowe konkstruktory:

- $\mathcal{U}$ - suma : $(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$
- $\mathcal{E}$ - pełny kwantyfikator egzystencjalny : $(\exists R.C)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} | \exists b, (a,b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}\}$
- $\mathcal{N}$ - ograniczenia liczbowe:
- $(\geq nR)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} || \{b \mid (a,b) \in R^{\mathcal{I}}\} |\geq n\}$
- $(\leq nR)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} || \{b \mid (a,b) \in R^{\mathcal{I}}\} |\leq n\}$
- $\mathcal{C}$ - negacja : $(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$

Używające powyższych konstruktorów języki nazywają się odpowiednio:

- $\mathcal{ALU}$
- $\mathcal{ALE}$
- $\mathcal{ALN}$
- $\mathcal{ALC}$ -> odpowiada podzbiorowi logiki pierwszego rzędu ograniczonemu do formuł z dwoma zmiennymi

Powiązanie z innymi rachunkami (logicznymi)

Większość logik opisowych jest podzbiorem logiki pierwszego rzędu:

- nazwy pojęć — predykaty unarne
- relacje (atomiczne) — predykaty binarne
- pojęcia — formuły z jedną wolną zmienną

Formuły logiki opisowej można intuicyjnie interpretować poprzez analogię do algebry zbiorów.

Przykład użycia	Składnia DL	Składnia FOL	Algebra zbiorów
Mężczyzna: **dorosły i osoba i rodzaju męskiego**	$C_1 \sqcap \ldots \sqcap C_n$	$C_1(x) \wedge \ldots \wedge C_n(x)$	$C_1 \cap \ldots \cap C_n$
Gazeta to **dziennik lub czasopismo**	$C_1 \sqcup \ldots \sqcup C_n$	$C_1(x) \vee \ldots \vee C_n(x)$	$C_1 \cup \ldots \cup C_n$
Wszystko, co jedzą wegetarianie to **nie mięso**	$\neg C$	$\neg C(x)$	C^c (dopełnienie zbioru)
Kraje UE to: **Niemcy, Francje, ..., Polska**	$\{x_1\} \sqcup \ldots \sqcup \{x_n\}$	$x = x_1 \vee \ldots \vee x = x_n$	$\{x_1\} \cup \ldots \cup \{x_n\}$
Każde zwierzę, które ma starsza pani **to kot**	$\forall P.C$	$\forall y.P(x,y) \rightarrow C(y)$	$\pi_Y(P) \subseteq C$
Właściciel psa **ma jakiegoś psa**	$\exists P.C$	$\exists y.P(x,y) \wedge C(y)$	$\pi_Y(P) \cap C \neq \emptyset$ ($sqcap$ - projekcja)
Rozsądny mężczyzna spotyka się z **maksymalnie 1 kobietą równocześnie** ;-)	$\leq nP$	$\exists^{\leq n} y.P(x,y)$	$card(P) \leq n$ (card - liczność zbioru)
Miłośnik zwierząt **ma minimum 3 zwierzaki**	$\geq nP$	$\exists^{\geq n} y.P(x,y)$	$card(P) \geq n$
Dziecko **to to samo co** młoda osoba	$C \equiv D$	$\forall x.C(x) \leftrightarrow D(x)$	$C \equiv D$
Każda foka jest zwierzęciem	$C \sqsubseteq D$	$\forall x.C(x) \rightarrow D(x)$	$C \subseteq D$