

Kategorialny Parser Składniowo-Semantyczny „ENIAM”

definicja reprezentacji semantycznej

Wojciech Jaworski

1 Wprowadzenie

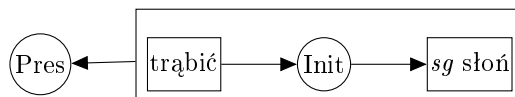
Znaczenie wypowiedzi reprezentowane jest w systemie dwupoziomowo. Najpierw za pomocą grafu semantycznego opisującego występujące w tekście pojęcia oraz związki pomiędzy nimi a następnie w postaci formuły logiki pierwszego rzędu rozszerzonej o predykat metajęzykowy i dodatkowe kwantyfikatory.

Następne trzy sekcje niniejszego dokumentu opisują poglądowo poszczególne elementy formalizmu, po nich następuje teoriomodelowa definicja.

Grafy semantyczne inspirowane są pracami Johna Sowy „Knowledge Representation: Logical, Philosophical, and Computational Foundations”

2 Grafy i formuły

Formuły naszego języka reprezentacji znaczenia wyrażamy graficznie w formie grafów semantycznych. Przykładowo dla zdania *Słoń trąbi* uzyskamy graf:



W powyższym grafie pudełka reprezentują byty, o których jest mowa. Występuje zatem obiekt *słoń* i zdarzenie *trąbić*. Symbol *sg* określa licznosc obiektów jako dokładnie 1. Występujące w pudełkach napisy to typy bytów, są nimi pochodzące ze Słownosieci sensy.¹

¹Formalnie sensy w Słownosieci identyfikują leksemy zaopatrzone w numery rozpoznające poszczególne znaczenia, np *słoń 1* i *trąbić 1*. Aby zwiększyć czytelność prezentacji będziemy jednak konsekwentnie pomijać te numery.

Kółeczka zawierają nazwy ról tematycznych opisujących powiązania pomiędzy bytami. Init wskazuje na to, że *słoń* jest inicjatorem *trąbienia*, a Pres na to, że zdarzenie jest równoczesne z czasem jego wypowiedzenia. Źródłem informacji o rolach tematycznych argumentów czasowników (a w przyszłości również rzeczowników, przymiotników i przysłówków) jest słownik walencyjny Walenty.

Zewnętrzna ramka to kontekst. Reprezentuje on sytuację, czyli podzbiór czasoprzestrzeni, w którym istnieją byty wskazane przez pojęcia w pudełkach i zachodzą wymienione w kółeczkach związki pomiędzy nimi.

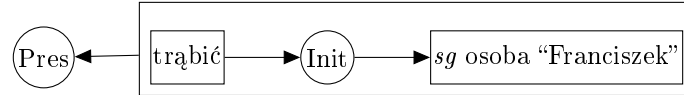
Powyższy graf jest równoważny formule logicznej

$$\exists(x, \text{DSCR}(x, \exists(s, \text{TYPE}(s, \text{słoń}) \wedge |s| = 1,$$

$$\exists(t, \text{TYPE}(t, \text{trąbić}) \wedge \text{INIT}(t, s))), \text{PRES}(x)),$$

w której każdy z obiektów jest identyfikowany przez zmienną. Predykat $\text{TYPE}(x, t)$ przypisuje typ t do zmiennej x , to znaczy, stwierdza, że zbiór bytów wskazywanych przez x należy do sensu t . Ze zmiennymi zawsze wiążemy zbiory bytów. Pozwala to w prosty sposób zdefiniować licznosc. Predykaty INIT i PRES odpowiadają rolom tematycznym. Z kolei predykat metajęzykowy DSCR wyraża kontekst.

Standardowo każdej jednostce leksykalnej (leksemowi lub wyrażeniu wielosłownemu) zawartej w zdaniu odpowiada pudełko, a relacji składniowej kółeczko. Pudełka zawierają sensy jednostek leksykalnych. Mogą też zawierać ich liczbę i/lub kwantyfikację. Zaś w przypadku nazw własnych nazwę i typ nazwy własnej np *Franciszek trąbi*:



Formuła odpowiadająca temu zdaniu ma postać

$$\exists(x, \text{DSCR}(x, \exists(o, \text{TYPE}(o, \text{osoba}) \wedge \text{HASNAME}(o, \text{'Franciszek'}) \wedge |o| = 1,$$

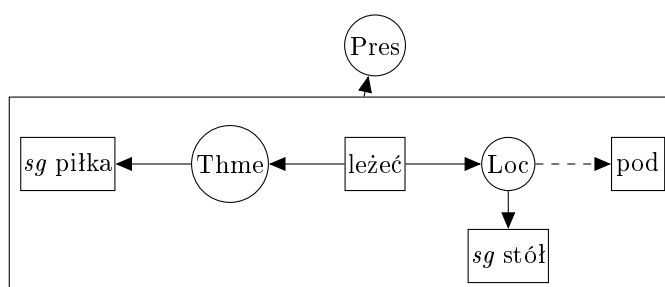
$$\exists(t, \text{TYPE}(t, \text{trąbić}) \wedge \text{INIT}(t, o))), \text{PRES}(x)).$$

Nazwy własne reprezentujemy za pomocą predykatu $\text{HASNAME}(x, \text{'name'})$, który wiąże nazwę 'name' ze zbiorem bytów oznaczonym przez x . W grafach semantycznych nazwy własne (np. *Franciszek*) umieszczamy w cudzysłowach. Nazwy własne nie definiują typu obiektu, tylko identyfikują obiekt poprzez podanie przypisanej mu etykiety.

Poszczególne elementy obu formalizmów reprezentacji znaczenia opiszemy szczegółowo w następnych rozdziałach.

3 Składnia powiązań między pojęciami

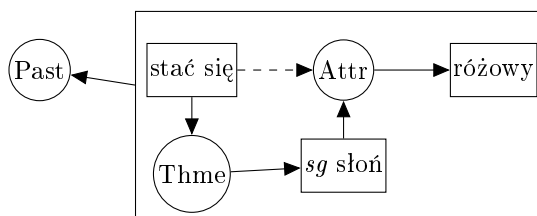
Związki pomiędzy pojęciami reprezentujemy za pomocą kuleczek zaopatrzonych w strzałki wskazujące charakter udziału każdego z pojęć w związku. W notacji logicznej kuleczkom odpowiadają predykaty a strzałki wskazują ich argumenty. Kierunek strzałek w relacjach jest określony tak, by prowadziły od pojęcia stanowiącego nadrzędnik w drzewie zależności składniowych do pojęcia stanowiącego podrzędnik w tym drzewie². Zabieg ten ma na celu zwiększenie czytelności grafów. Występuje 5 układów strzałek. Poniższy graf semantyczny dla zdania *Piłka leży pod stołem* prezentuje trzy z nich:



Są to:

- jednoargumentowa rola Pres, strzałka wskazująca argument prowadzi do kółeczka;
- dwuargumentowa rola Thme, strzałka wchodząca do kółeczka jest pierwszym argumentem, a strzałka wychodząca drugim;
- trójargumentowa rola Loc, strzałka wchodząca do kółeczka jest pierwszym argumentem, przerywana strzałka wychodząca drugim, a ciągła wychodząca trzecim.

Zdanie *Słoń stał się różowy* prezentuje przykład czwartego typu:

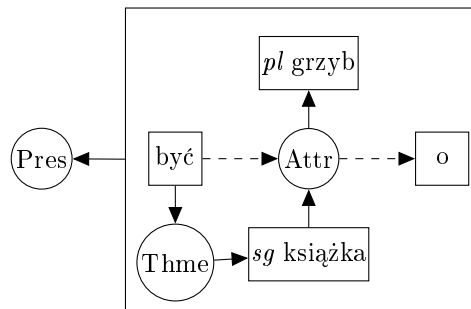


Jest to:

²Jest to ogólna reguła, która ma zastosowanie, gdy relacja semantyczna wiąże pojęcia odpowiadające słowom połączonym związkiem składniowym. W konkretnych sytuacjach mogą się pojawiać wyjątki od niej.

- parametryzowana przez zdarzenie dwuargumentowa rola Attr, gdzie ciągła strzałka wchodząca do kółeczka jest pierwszym argumentem, wychodząca wychodząca drugim, a przerywana wchodząca wskazuje parametryzujące zdarzenie.

Piąty typ występujący w zdaniu *Książka jest o grzybach* stanowi połączenie typu trzeciego i czwartego:



Mamy tu:

- parametryzowaną przez zdarzenie trójargumentową rolę Attr, strzałka wchodząca do kółeczka jest pierwszym argumentem, przerywana strzałka wychodząca drugim, ciągła wychodząca trzecim, a przerywana wchodząca wskazuje parametryzujące zdarzenie.

4 Role tematyczne w „Walenty”

Zbiór ról tematycznych jest rozszerzeniem systemu ról tematycznych ze słownika walencyjnego „Walenty”. Poniżej prezentujemy listę ról wraz z ich definicjami wziętymi z „Zasad opracowania ram semantycznych w słowniku walencyjnym Walenty” autorstwa Anny Andrzejczuk i Elżbiety Hajnicz. Skróty nazw ról są inspirowane skrótami zaproponowanymi przez Johna Sowę w książce „Knowledge Representation”. Role występujące w systemie Walentego dzielimy na podstawowe opisujące obligatoryjnych uczestników sytuacji:

Initiator (Init) ktoś, kto lub coś, co inicjuje czynność wskazaną przez predykat akcji;

Stimulus (Stim) coś lub ktoś, kto swoją cechą, właściwością wywołuje reakcję w Experiencerze;

Theme (Thme) ktoś lub coś, kto lub co podlega działaniu Initiatora, jakiegoś samoistnemu procesowi lub znajduje się w jakimś stanie;

Experiencer (Expr) ktoś odczuwający uczucia, emocje, przeżywający różne stany psychiczne, mimowolnie odbierający bodźce zmysłami;

Factor (Fctr) ktoś, kto choć bezpośrednio nie wykonuje danej czynności, to przyczynia się w sposób czynny do jej wykonania, ma wpływ na sytuację, wspomaga, a nawet umożliwia jej zaistnienie;

Instrument (Inst) coś, dzięki czemu możemy wykonać jakąś czynność, co pełni funkcję wspomagającą, ułatwiającą wykonanie akcji;

Recipient (Rcpt) ktoś, ku komu jest skierowana czynność, ale kto tej czynności nie podlega, do kogo jest skierowana wypowiedź, z myślą o kim podejmowana jest akcja;

Result (Rslt) coś, co jest efektem końcowym akcji, na którą wskazuje predykat, co powstaje w jej wyniku.

Oprócz tego rozważane są role uzupełniające, które służą do reprezentacji charakteru okoliczności sytuacji:

Measure (Meas) przypisujemy argumentom, które informują w jakim stopniu została wykonana czynność czy jaka jest jej wielkość, a dokładniej podają miarę tej czynności;

Location (Loc) określa miejsce, w którym dzieje się akcja czy też skąd lub dokąd zmierza uczestnik sytuacji;

Path (Path) to trasa, ścieżka, droga, czy to w sensie fizycznym, metafizycznym czy metaforycznym;

Time (Time) to czas, w którym coś się dzieje;

Duration (Dur) określa czas trwania sytuacji, na którą wskazuje predykat;

Condition (Cond) określa okoliczności, bez zaistnienia których nie doszłoby do zaistnienia akcji; informuje o przyczynie lub warunkach zaistnienia sytuacji;

Purpose (Purp) określa cel, który przyświeca sytuacji, na którą wskazuje predykat; gdyby Initiatorowi nie przyświecał ten cel, nie doszłoby do danej sytuacji; określa też nasze marzenia, życzenia, oczekiwania.

Attribute (Attr) jako jedyna z ról uzupełniających określa jednego z uczestników sytuacji; przypisujemy ją cechom, właściwościom ludzi, zwierząt, przedmiotów, zjawisk, procesów, a także stanom, emocjom itp;

Manner (Manr) określa, w jaki sposób została wykonana określona czynność;

Powyższy zbiór ról modyfikowany jest rolami pomocniczymi — atrybutami ról. Role pomocnicze służą rozróżnieniu uczestników sytuacji, którzy mają tę samą rolę podstawową lub uzupełniającą.

Role główną opatrujemy atrybutami Foreground, Background, gdy w ramie semantycznej istnieją dwa argumenty mające przypisaną tę rolę. Podstawą do przypisania tych atrybutów, jest równorzędność argumentów opatrzonych tą rolą.

Role główną opatrujemy atrybutami Source, Goal wówczas, gdy oba argumenty pełnią taką samą rolę w zdaniu, ale sytuacja wskazuje kierunek wykonywanej czynności czy procesu. Source (Src) to początek, źródło procesów, czynności, a Goal (Goal) to cel, do którego się dąży lub etap końcowy procesu, czynności. Wyjątek stanowi tu rola Location, gdzie Location Source informuje o miejscu początkowym czynności a Location Goal o miejscu końcowym.

W naszym systemie znaczenie roli Measure jest rozszerzone na rzeczowniki i oznacza ona miarę dla pojęć wyrażonych przez rzeczowniki niepoliczalne. Oprócz niej wprowadzamy rolę Count wyrażającą liczebność odniesienia rzeczowników policzalnych.

Jednoargumentowe role Past, Pres i Fut wskazują związek czasu wypowiedzi z czasem, do którego się ona odnosi. Z kolei rola Successor (Succ) wyraża następstwo czasowe sytuacji.

5 Model

Zdefiniujemy teraz model, w którym będziemy wyrażać formalną semantykę naszych języków reprezentacji znaczenia. Semantykę zdefiniujemy wprost dla formuł logicznych, następnie w rozdziale ?? przeniesiemy jej definicję na grafy semantyczne.

Dla osób nieobeznanych z logiką formalną: Model możemy sobie wyobrazić jako strukturę danych, która zawiera precyzyjny opis tego co znajduje się w świecie, informacje o wszystkich bytach i relacjach między nimi. Grafy pojęć i formuły logiczne stanowią teorię opisującą ten model. Model również opisujemy za pomocą pewnego formalizmu (języka), więc na pierwszy rzut oka treść tego rozdziału może wydawać się „tłumaczeniem z polskiego na nasze”. Tym niemniej istnieje fundamentalna różnica między modelem a opisującą go teorią: model wyraża dokładnie to co jest, a teoria zawiera zbiór zdań, które w tym modelu są prawdziwe. Zatem kompletny opis modelu

jest (często nieskończonym) zbiorem predykatów stwierdzającym co w modelu jest prawdziwe. W opisie tym nie ma miejsca na negację, alternatywę, czy implikację. Z kolei od teorii wymagamy jedynie tego by była zgodna z modelem, czyli prawdziwa, nie musi ona być kompletna.

Model \mathcal{M} jest to uniwersum, czyli zbiór wszystkich bytów, oraz zależne od czasu relacje i funkcje zdefiniowane na tych bytach. Niech x będzie elementem uniwersum modelu, który jest w relacji r z elementem y w chwili t . Fakt ten będziemy oznaczać $x.r(t) = y$. Notacja ta odzwierciedla przyjęte przez nas rozumienie funkcji i relacji jako cech (atrybutów) bytów. Tak więc r jest cechą, atrybutem x , a y jest wartością tej cechy. Jeśli dany atrybut nie zmienia swojej wartości w przeciągu sytuacji opisywanej przez model argument czasowy pomijamy w notacji, tzn. zamiast $x.r(t) = y$ napiszemy $x.r = y$.

Wśród elementów uniwersum znajdują się

- liczby,
- interwały, czyli odcinki czasu,
- zdarzenia i stany,
- przedmioty,
- napisy,
- sytuacje,
- algorytmy znaczeniowe.

Przykładowe atrybuty to

beg wiążący interwał z punktem w czasie będącym jego początkiem,

end wiążący interwał z punktem w czasie będącym jego końcem,

len wskazujący długość interwału.

time oznaczający czas zdarzenia/stanu,

loc oznaczający miejsce zdarzenia/stanu/przedmiotu,

name wiążący byt z napisem będącym jego nazwą.

Interpretacja jest funkcją, która wiąże symbole języka z elementami modelu. Interpretacja stałej a to element uniwersum, który wskazuje ta stała. Element ten oznaczany jako $a^{\mathcal{M}}$. Przyjmujemy ponadto, że interpretacją napisu jest on sam, czyli $s^{\mathcal{M}} = s$, gdy s jest napisem.

Zmienne w przeciwieństwie do stałych nie mają interpretacji tylko wartościowanie. Wartościowanie jest funkcją przypisującą zmiennym elementy uniwersum. Wartościowanie pozwala zdefiniować kwantyfikatory, np formuła $\exists(x, \varphi(x))$ jest spełniona w modelu \mathcal{M} , gdy istnieje wartościowanie v przypisujące zmiennej x element uniwersum taki, że $\varphi(x)$ jest prawdziwe w modelu \mathcal{M} .

Interpretacją predykatu p jest relacja oznaczana $p^{\mathcal{M}}$. Na przykład interpretacją predykatu `HASNAME` jest relacja `HASNAME` ^{\mathcal{M}} . W naszym modelu przyjmujemy, że `HASNAME` ^{\mathcal{M}} = *name*. Oznacza to, że dla zmiennej x i napisu s , formuła `HASNAME`(x , ' s ') jest prawdziwa wtw. gdy $v(x).name = s^{\mathcal{M}} = s$.

Prawdziwość formuły φ w modelu \mathcal{M} przy wartościowaniu v wyrażamy za pomocą relacji konsekwencji semantycznej ' \models '. Na przykład ostatnie zdanie z poprzedniego akapitu możemy zapisać jako:

$$(\mathcal{M}, v \models \text{HASNAME}(x, 's')) \iff v(x).name = s^{\mathcal{M}} \iff v(x).name = s$$

Predykat `TYPE` wiąże ze sobą zbiór bytów oraz sens ze Słownosieci. Interpretacjami słownosieciowych sensów są *algorytmy znaczeniowe*, które dla danego bytu rozstrzygają, czy można o nich prawdziwie orzec przynależność do sensu. Algorytm znaczeniowy zwraca stałe **True** oraz **False**. Możemy teraz zdefiniować predykat `TYPE`:

$$(\mathcal{M}, v \models \text{TYPE}(x, t)) \iff t^{\mathcal{M}}(v(x)) = \text{true}$$

Aby uprościć notację zamiast pisać $t(x) = \mathbf{True}$ będziemy pisać $t(x)$, czyli pozornie utożsamimy wartości zwracane przez algorytmy znaczeniowe ze wartościami logicznymi. Ponadto zakładamy, że interpretacja danego sensu jest identyczna w każdym modelu, czyli że słowa mają ustalone znaczenia wspólne dla wszystkich użytkowników języka. Odzwierciedlimy ten fakt w notacji oznaczając algorytmy znaczeniowe pogrubioną czcionką i pomijając wskazanie modelu. Na przykład zamiast pisać $\text{słó}\overset{\mathcal{M}}{\text{ń}}$ napiszemy **słóń**. Zatem powyższą definicję zapiszemy w skrócie jako:

$$(\mathcal{M}, v \models \text{TYPE}(x, t)) \iff \mathbf{t}(v(x))$$

6 Wprowadzanie modelu

Semantyka języka reprezentacji znaczenia budowana jest na zbiorze możliwych światów. Przyjmujemy, że każdy mówca prezentuje pewien model świata, w którym interpretuje swoje wypowiedzi i każdy akt komunikacji jest utworzeniem nowego modelu. Model ten nazywamy modelem wypowiedzi. Gdy ktoś coś komunikuje, nie znaczy to, że treść komunikatu jest faktycznie prawdziwa, ani tego, że nadawca wierzy w jej prawdziwość, a jedynie, że przedstawia model, w którym ona jest prawdziwa. Dzięki temu, że każdy mówca przedstawia oddzielny model wypowiedzi, reprezentacja semantyczna dłuższego tekstu, w którym różne osoby wypowiadają sprzeczne opinie, zdaje z tego sprawę, sama nie będąc kontradiktoryczna.

W sytuacji, w której mówca mówi o tym, co sądził lub co powiedział ktoś drugi, pierwszy symuluje w swoim modelu wypowiedzi model drugiego. Może się w tym mylić i wtedy niezgodność zachodzi między pierwszym modelem a stanem faktycznym świata (tzw. światem aktualnym). Niezależnie od tego, referowana myśl też może być prawdziwa lub nieprawdziwa – za to odpowiada już drugi model.

Poszczególne modele mają strukturę hierarchiczną. Na przykład zdanie

Jaś wierzy, że Marysia kłamie, że go nienawidzi. (1)

interpretujemy następująco:

- Na poziomie narracji mamy model \mathcal{A} , w którym zachodzi sytuacja a , czyli *Jaś wierzy, że b* .
- \mathcal{B} jest modelem, w którym zachodzi sytuacja b , czyli *Marysia kłamie, że c* .
- \mathcal{C} jest modelem, w którym zachodzi sytuacja c , czyli *Marysia nienawidzi Jasia*.

Zdanie (1) reprezentowane z użyciem metapredykatu DSCR będzie miało w ogólności następującą postać (reprezentowana jest jedynie zmiana kontekstu):

$$\begin{aligned} & \text{Narrator zakomunikował, że } a \\ & \quad \wedge \text{DSCR}(a, \text{Jaś wierzy, że } b) \\ & \quad \wedge \text{DSCR}(b, \text{Marysia kłamie, że } c) \\ & \quad \wedge \text{DSCR}(c, \text{Marysia nienawidzi Jasia})) \end{aligned} \tag{2}$$

Z racji tego, że modele mają strukturę hierarchiczną, model wewnętrzny powinien należeć do uniwersum modelu zewnętrznego. Były odpowiadające

takim modelom nazywamy sytuacjami. Niech φ będzie formułą, a s stałą, której interpretacją w \mathcal{M} jest sytuacja. Wtedy

$$(\mathcal{M}, v \models \text{DSCR}(s, \varphi)) \iff (v(s).model \models \varphi)$$

Atrybut *model* wiąże sytuację, z opisującym ją modelem.

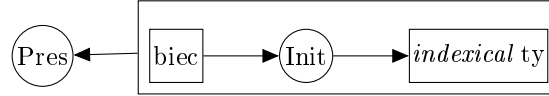
7 Wprowadzanie bytów do dyskursu

W języku logiki pierwszego rzędu wzmiankowane byty stanowią odniesienia zmiennych, a same zmienne wprowadzane są przez kwantyfikatory. Przy czym dostępne są dwa rodzaje kwantyfikacji: uniwersalna w której wartościowanie wprowadzanej zmiennej przebiega po wszystkich elementach uniwersum spełniających restrykcję oraz egzystencjalna, przy której zmienna wartościowana jest jednym z bytów spełniających restrykcję.

W przypadku grafów semantycznych zmienne nie są jawnie wskazane, tym niemniej zachodzi konieczność określenia w jaki sposób obiekty wskazywane przez pudełka są związane z uniwersum.

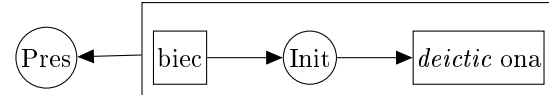
Domyślnie przyjmujemy kwantyfikację egzystencjalną, czyli np. zdanie *Słoń trąbi* stwierdza o istnieniu *słonia* i istnieniu zdarzenia *trąbienia*.

Z kolei dla zdania *Ty biegniesz* otrzymamy reprezentację z pojęciem okazjonalnym (indexical), który wiąże słowo *ty* z uczestnikami komunikacji, czyli odwołuje się do uniwersum na metapoziomie.



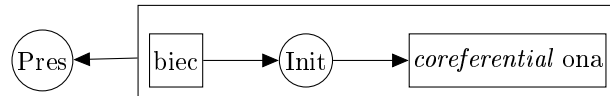
$$\begin{aligned} &\exists(x, \text{DSCR}(x, \exists(s, \text{TYPE}(s, \text{ty}(\#i)), \\ &\exists(t, \text{TYPE}(t, \text{biec}) \wedge \text{INIT}(t, s))))), \text{PRES}(x)), \end{aligned}$$

a dla zdania *Ona biegnie* otrzymamy reprezentację z pojęciem deiktycznym



$$\begin{aligned} &\exists(x, \text{DSCR}(x, \exists(s, \text{TYPE}(s, \text{ona}(\#d)), \\ &\exists(t, \text{TYPE}(t, \text{biec}) \wedge \text{INIT}(t, s))))), \text{PRES}(x)), \end{aligned}$$

lub koreferencyjnym



$$\begin{aligned} & \exists(x, \text{DSCR}(x, \exists(s, \text{TYPE}(s, \text{ona}(\#c)), \\ & \exists(t, \text{TYPE}(t, \text{biec}) \wedge \text{INIT}(t, s))))), \text{PRES}(x)), \end{aligned}$$

W powyższych trzech przypadkach nie następuje wprowadzenie nowego bytu do dyskursu. Zmienne wskazujące odniesienia pojęć okazjonalnych, deiktycznych i koreferencyjnych są wiązane przez algorytmy znaczeniowe tych pojęć z bytami wskazanymi przez ich dodatkowe argumenty.

Zaimki wnoszą informację o osobie, liczbie i rodzaju, która jest istotna przy ustalaniu ich referenta. Z tego względu lematyzujemy je do form uwzględniających te informacje (*ja, wy, one, która*), ewentualnie uzupełniając np. *ja-m*.

Operator wprowadzający pojęcie okazjonalne definiujemy następująco: niech x będzie sytuacją, którą modeluje \mathcal{M} , tzn. $x.model = \mathcal{M}$, wtedy

$$\#i^{\mathcal{M}} = x$$

, czyli

$$(\mathcal{M}, v \models \text{TYPE}(s, \text{ty}(\#i))) \iff \mathbf{ty}(x)(v(s))$$

Algorytmy znaczeniowe dla okazjonalnych zaimków *ja* i *ty* możemy zdefiniować następująco:

$$\begin{aligned} \mathbf{ja} &\equiv \lambda x \lambda y \ x.speaker = y \\ \mathbf{ty} &\equiv \lambda x \lambda y \ x.hearer = y \end{aligned}$$

Argument deiktyczny to kontekstowo zdefiniowany zbiór x zawierający obiekty, na które wskazywał mówca

$$\#d^{\mathcal{M}} = x$$

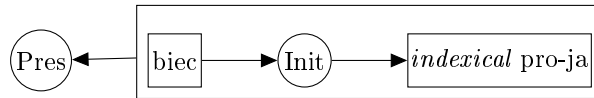
Z kolei argument koreferencyjny to kontekstowo zdefiniowany zbiór x zawierający obiekty, które były wzmiankowane w wypowiedzi

$$\#c^{\mathcal{M}} = x$$

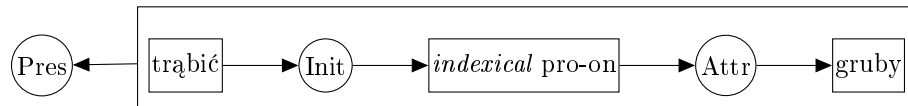
Algorytm znaczeniowy zaimka *ona* nie zależy od tego czy argument jest koreferencyjny, czy deiktyczny:

$$\mathbf{ona} \equiv \lambda x \lambda y \ y \in x \wedge y.gender = f$$

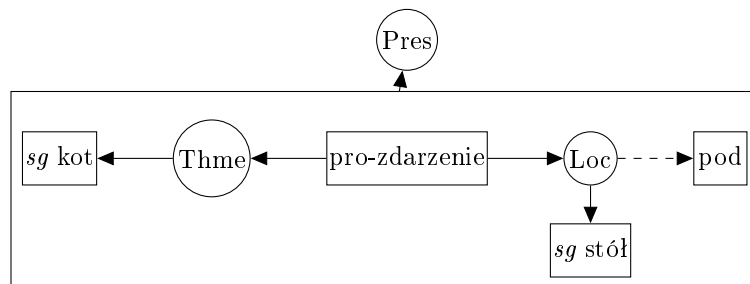
Kiedy podmiot nie jest dany w sposób jawny, jest reprezentowany za pomocą wariantu niemego zaimka *pro* np. *Biegnę.*:



Zaimek *pro* używany jest również, kiedy przedmiot jest wprowadzany poprzez podanie swojej cechy, co ma miejsce we frazach rzeczownikowych, których głową jest przymiotnik np. *Gruby trąbi*.



Kolejnym zastosowaniem niemego zaimka są równoważniki zdań. Dodajemy do nich *pro-zdarzenie*, które wiąże poszczególne elementy zdania, np. *Kot pod stołem*



W dalszym toku tekstu wprowadzimy jeszcze pojęcia interrogatywne (interrogative), występujące przy zadawaniu pytań, jak również kwantyfikację uniwersalną (przebiegającą po wielu bytach).

8 Definicje podstawowych ról tematycznych

ważyć

kosztować-2 Perski dywan kosztował Kalasantego 100.000zł. Thme Recipient Measure

Jan zaaranżował Marysi spotkanie Init Recipient Thme

węszyć-2 Kot węszył za myszami Expr Stimulus

golić-2 Kalasanty golił brodę u cyrulika Thme F Thme B Factor

Cyrulik ogolił Kalasantemu brodę brzytwą. Init Thme F Thme B Inst

słuchać

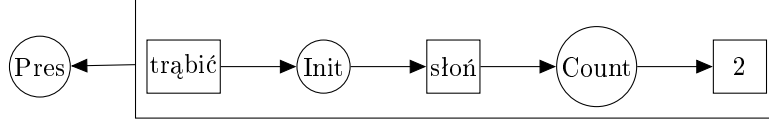
aranżować

9 Liczebność i miara

Odniesieniem rzeczowników policzalnych są kolekcje bytów. Atrybuty kolekcji są jednocześnie atrybutami wszystkich indywiduów wchodzących w jej

skład. W szczególności jednoelementowa kolekcja jest tożsama z indywiduum, które się na nią składa.

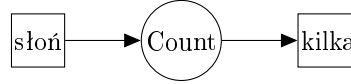
Rola Count wskazuje liczebność odniesienia rzeczowników policzalnych np: *Dwa słonie trąbią, kilka słoni*



$$\exists(x, \text{DSCR}(x, \exists(s, \text{TYPE}(s, \text{słoń}) \wedge \exists(d, \text{TYPE}(d, \text{dwa}), \text{COUNT}(s, d)), \exists(t, \text{TYPE}(t, \text{trąbić}) \wedge \text{INIT}(t, s))))), \text{PRES}(x)),$$

lub równoważnie

$$\exists(x, \text{DSCR}(x, \exists(s, \text{TYPE}(s, \text{słoń}) \wedge |s| = 2, \exists(t, \text{TYPE}(t, \text{trąbić}) \wedge \text{INIT}(t, s))))), \text{PRES}(x)),$$



$$\text{TYPE}(s, \text{słoń}) \wedge \text{TYPE}(k, \text{kilka}) \wedge \text{COUNT}(s, k),$$

Liczebniki odnoszą do liczb i reprezentujemy je jako zmienną desygnującą liczbę oraz algorytm znaczeniowy sprawdzający należenie jej do zbioru wyznaczonego przez intensję słowa. Liczebniki określone reprezentujemy za pomocą liczb, a nieokreślone (np *kilka*, *trochę*) za pomocą leksemów. Algorytmy znaczeniowy dla pojęć *2* i *kilka* mają postać

$$\mathbf{2} \equiv \lambda x \, x = 2$$

$$\mathbf{kilka} \equiv \lambda x \, x > 2 \wedge x < 10$$

Predykat COUNT zdefiniujemy następująco:

$$(\mathcal{M}, v \models \text{COUNT}(x, y)) \iff v(x).count = v(y)$$

Atrybut *count* określa liczebność kolekcji.

Wprowadzamy też funkcję $|\cdot|$ zwracającą liczebność kolekcji:

$$|s| = n \iff \text{COUNT}(s, n)$$

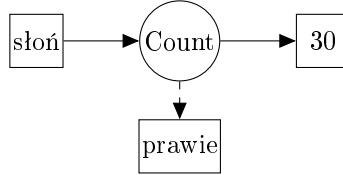
Dla liczebników określonych wprowadzamy skrót notacyjny: zamiast wprowadzać zmienną odnoszącą do liczby podajemy wartość tej liczby. Zatem zamiast pisać

$$\exists(d, \text{TYPE}(d, \text{dwa}), \text{COUNT}(s, d))$$

napiszemy

$$|s| = 2$$

Liczebniki mogą być modyfikowane przez operatory adnumeratywne np. *prawie 30 słoń*.



$$\text{TYPE}(s, \text{słoń}) \wedge \text{TYPE}(p, \text{prawie}) \wedge \text{COUNT}(s, p, 30)$$

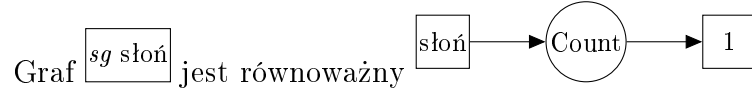
Trójargumentowy predykat COUNT definiujemy następująco:

$$(\mathcal{M}, v \models \text{COUNT}(x, r, y)) \iff v(r)(v(x).count, v(y))$$

Odniesieniem leksemu *prawie* jest elacja dwuargumentowa spełniająca algorytm znaczeniowy

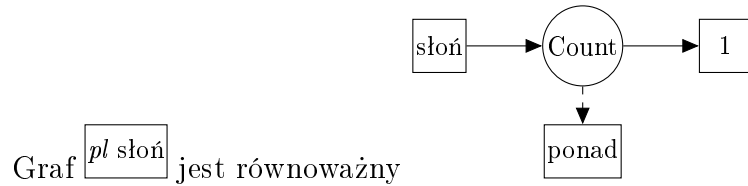
$$\mathbf{prawie} \equiv \lambda r \, r = \{(x, y) : x < y \wedge |x - y| < c\}, \text{ dla jakiegoś ustalonego } c$$

Liczność rzeczowników, które nie mają jej jawnie zadanej przez liczebnik wnioskujemy na podstawie liczby gramatycznej i zapisujemy za pomocą uproszczonej notacji. Symbol *sg* określa licznosc obiektów jako dokładnie 1.



$$\text{ i odpowiada mu formuła } \text{TYPE}(s, \text{słoń}) \wedge |s| = 1$$

Oprócz niego stosujemy symbol *pl* na określenie liczności większej niż 1.



Poyższym grafom odpowiada formuła

$$\text{TYPE}(s, \text{słoń}) \wedge \text{TYPE}(p, \text{ponad}) \wedge \text{COUNT}(s, p, 1)$$

Leksem *ponad* definiujemy jako

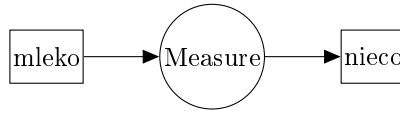
$$\mathbf{ponad} \equiv \lambda r \, r = \{(x, y) : x > y\}$$

Zgodnie z tą definicją powyższa formuła jest równoważna następującej:

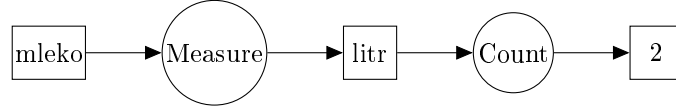
$$\text{TYPE}(s, \text{słóń}) \wedge |s| > 1$$

Rzeczowniki plurale tantum nie wnoszą informacji o liczności. Dla innych części mowy liczność uznajemy za nieokreśloną, gdy nie jest jawnie wskazana.

Pojęcia do których odnoszą się rzeczowniki niepoliczalne są oznaczane symbolem *mass*, a ich miara jest wyrażana predykatem MEASURE. Miara może być wyrażona za pomocą liczebników np. *nieco mleka* lub pojemników np. *dwa litry mleka*



$$\text{TYPE}(m, \text{mleko}) \wedge \text{TYPE}(n, \text{nieco}) \wedge \text{MEASURE}(m, n),$$



$$\text{TYPE}(m, \text{mleko}) \wedge \text{TYPE}(l, \text{litr}) \wedge |l| = 2 \wedge \text{MEASURE}(m, l).$$

10 Role przestrzenne

Rola Location (Loc) wskazuje położenie przedmiotu, czy zdarzenia. Role Location Source (Loc Src), Location Goal (Loc Goal), Path informują o obecności i kierunku ruchu. Predykaty odpowiadające tym rolom definiujemy w wersji dwuargumentowej, w której drugi argument jest miejscem

$$(\mathcal{M}, v \models \text{LOC}(x, y)) \iff \forall t \, v(x).loc(t) \sqsubset v(y)$$

$$(\mathcal{M}, v \models \text{LOCSRC}(x, y)) \iff \exists p \, v(x).loc(v(x).beg) \sqsubset v(y)$$

$$(\mathcal{M}, v \models \text{LOC GOAL}(x, y)) \iff \exists p \, v(x).loc(v(x).end) \sqsubset v(y)$$

$$(\mathcal{M}, v \models \text{PATH}(x, y)) \iff \exists t \, v(x).loc(t) \sqsubset v(y)$$

oraz w wersji trójargumentowej

$$(\mathcal{M}, v \models \text{LOC}(x, r, y)) \iff \forall t \exists p \, v(x).loc(t) \sqsubset p(t) \wedge r(p(t), v(y).loc(t))$$

$$(\mathcal{M}, v \models \text{LOCSRC}(x, r, y)) \iff$$

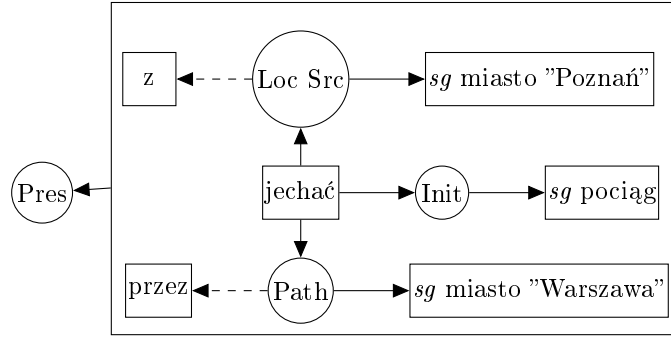
$$\exists p \, v(x).loc(v(x).beg) \sqsubset p \wedge r(p, v(y).loc(v(x).beg))$$

$$\begin{aligned}
(\mathcal{M}, v \models \text{LOC GOAL}(x, r, y)) &\iff \\
\exists p \, v(x).loc(v(x).end) \sqsubset p \wedge r(p, v(y).loc(v(x).end)) \\
(\mathcal{M}, v \models \text{PATH}(x, r, y)) &\iff \exists t \exists p \, v(x).loc(t) \sqsubset p \wedge r(p, v(y).loc(t))
\end{aligned}$$

gdzie p jest parametryzowanym przez czas miejscem, a r jest relacją pomiędzy miejscami. Uogólnione zawieranie zbiorów $X \sqsubset Y$ zachodzi wtw. X jest podzbiorem otoczki wypukłej Y .

Dwuargumentowe wersje predykatów są wykorzystywane zazwyczaj, gdy miejsce wskazane jest przez przysłówkę (np *tam*, *stąd*), a wersje trójargumentowe są używane, gdy relacje przestrzenne określane są za pomocą wyrażeń przyimkowych. Wtedy x jest odniesieniem nadrzędnika wyrażenia przyimkowego, r odniesieniem przyimka, a y odniesieniem podrzędnika przyimka.

Przykładowa interpretacja zdania *Z Poznania jedzie pociąg przez Warszawę.*:



$$\begin{aligned}
&\exists(x, \text{PRES}(x) \wedge \text{DSCR}(x, \\
&\quad \exists(z, \text{TYPE}(z, z), \\
&\quad \exists(p, \text{TYPE}(p, \text{miasto}) \wedge \text{HASNAME}(p, \text{'Poznań'}) \wedge |p| = 1, \\
&\quad \exists(j, \text{TYPE}(j, \text{jechać}), \\
&\quad \exists(o, \text{TYPE}(o, \text{pociąg}) \wedge |o| = 1, \\
&\quad \exists(r, \text{TYPE}(r, \text{przez}), \\
&\quad \exists(w, \text{TYPE}(w, \text{miasto}) \wedge \text{HASNAME}(w, \text{'Warszawa'}) \wedge |w| = 1, \\
&\quad \text{INIT}(j, o) \wedge \text{LOCSRC}(j, z, p) \wedge \text{PATH}(j, r, w)))))))))
\end{aligned}$$

11 Role czasowe

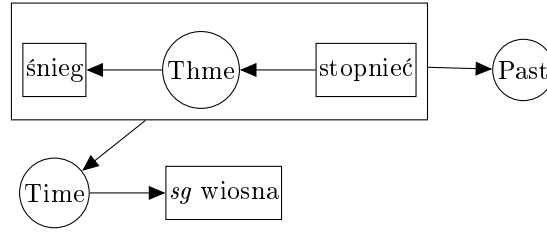
Predykat TIME wyraża czas zdarzenia lub sytuacji. Występuje, podobnie jak role przestrzenne, występuje w wersji dwu i trójargumentowej.

$$(\mathcal{M}, v \models \text{TIME}(x, y)) \iff \begin{cases} v(x).time = v(y) & \text{gdy } y \text{ jest interwałem} \\ v(x).time = v(y).time & \text{wpp.} \end{cases}$$

$$(\mathcal{M}, v \models \text{TIME}(x, r, y)) \iff \begin{cases} r(v(x).time, v(y)) & \text{gdy } y \text{ jest interwałem} \\ r(v(x).time, v(y).time) & \text{wpp.} \end{cases}$$

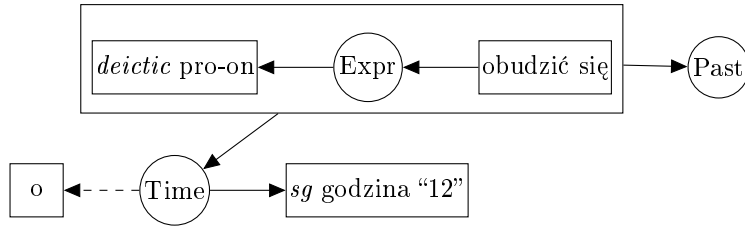
Czy nie powinno być $v(r)(v(x).time, v(y))$ zamiast $r(v(x).time, v(y))$?
Albo $\langle v(x).time, v(y) \rangle \in v(r)$?

Śnieg stopniał wiosną.



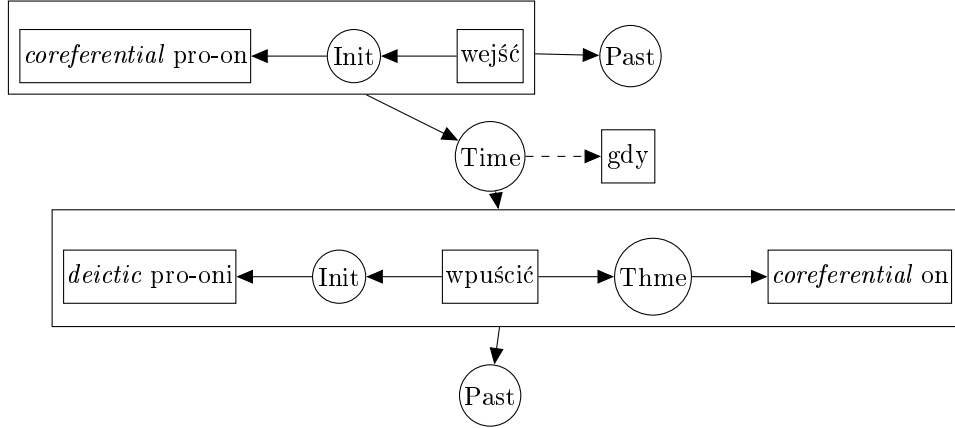
$$\exists(w, \text{TYPE}(w, \text{wiosna}) \wedge |w| = 1, \exists(x, \text{PAST}(x) \wedge \text{TIME}(x, w) \wedge \text{DSCR}(x, \exists(s, \text{TYPE}(s, \text{śnieg}), \exists(t, \text{TYPE}(t, \text{stopnieć}), \text{THME}(t, s))))))$$

Obudził się o 12.



$$\begin{aligned} &\exists(o, \text{TYPE}(o, o), \exists(g, \text{TYPE}(g, \text{godzina}) \wedge \text{HASNAME}(g, '12') \wedge |w| = 1, \\ &\quad \exists(x, \text{PAST}(x) \wedge \text{TIME}(x, o, g) \wedge \text{DSCR}(x, \\ &\quad \exists(p, \text{TYPE}(p, \text{pro-on}(\#d)), \exists(b, \text{TYPE}(b, \text{obudzić się}), \text{THME}(b, p)))))) \end{aligned}$$

Wszedł, gdy go wpuścili:

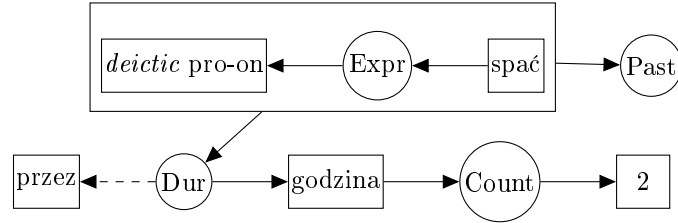


Predykat DUR wyraża czas trwania zdarzenia lub sytuacji.

$$(\mathcal{M}, v \models \text{DUR}(x, r, y)) \iff r(v(x).time.end - v(x).time.beg, v(y))$$

Wartością zmiennej y w powyższym wzorze jest długość interwału.

Spał przez dwie godziny.



$$\exists(r, \text{TYPE}(r, \text{przez}), \exists(g, \text{TYPE}(g, \text{godzina}) \wedge |g| = 2,$$

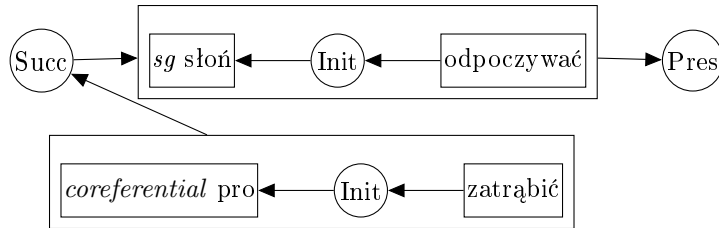
$$\exists(x, \text{PAST}(x) \wedge \text{DUR}(x, r, g) \wedge \text{DSCR}(x,$$

$$\exists(p, \text{TYPE}(p, \text{pro-on}(\#d)), \exists(s, \text{TYPE}(s, \text{spać}), \text{EXPR}(s, p))))))$$

Predykat SUCC opisuje następstwo czasów, bądź sytuacji

$$(\mathcal{M}, v \models \text{SUCC}(x, y)) \iff \begin{cases} v(x).end \leq v(y).time.beg & \text{gdy } x \text{ jest interwałem} \\ v(x).time.end \leq v(y).beg & \text{gdy } y \text{ jest interwałem} \\ v(x).time.end \leq v(y).time.beg & \text{wpp.} \end{cases}$$

Słoń odpoczywa zatrąbiwszy.



$$\begin{aligned}
& \exists(x, \text{PRES}(x) \wedge \text{DSCR}(x, \\
& \exists(s, \text{TYPE}(s, \text{słón}) \wedge |s| = 1, \exists(o, \text{TYPE}(o, \text{odpoczywać}), \text{INIT}(o, s)))) \wedge \\
& \exists(y, \text{SUCC}(y, x) \wedge \text{DSCR}(y, \\
& \exists(p, \text{TYPE}(p, \text{pro}(\#c)), \exists(z, \text{TYPE}(z, \text{zatrąbić}), \text{INIT}(z, p))))))
\end{aligned}$$

Czasy gramatyczne wyrażają relację pomiędzy czasem opisywanej sytuacji a czasem wygłoszenia komunikatu. Niech s będzie sytuacją, w której wygłoszono komunikat o treści x . Wtedy role PAST, PRES i FUT możemy zdefiniować następująco:

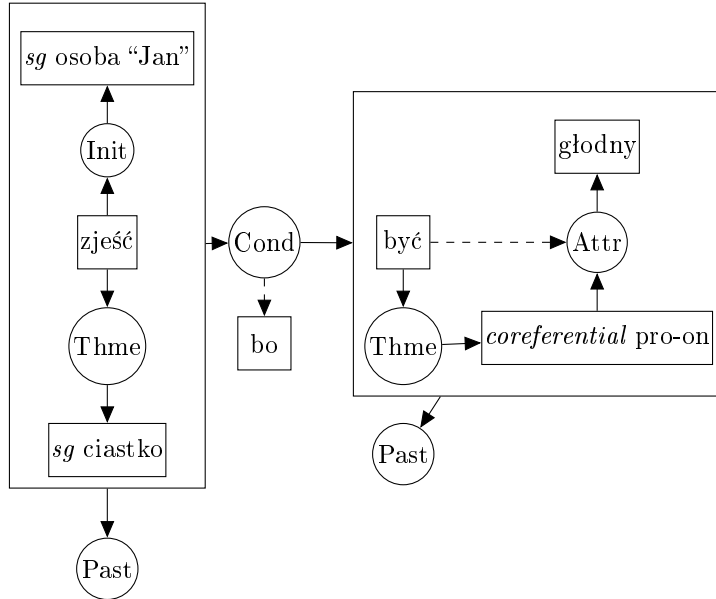
$$\begin{aligned}
(\mathcal{M}, v \models \text{PRES}(x)) & \iff v(x).time \cap s.time \neq \emptyset \\
(\mathcal{M}, v \models \text{PAST}(x)) & \iff v(x).time.end < s.time.beg \\
(\mathcal{M}, v \models \text{FUT}(x)) & \iff v(x).time.beg > s.time.end
\end{aligned}$$

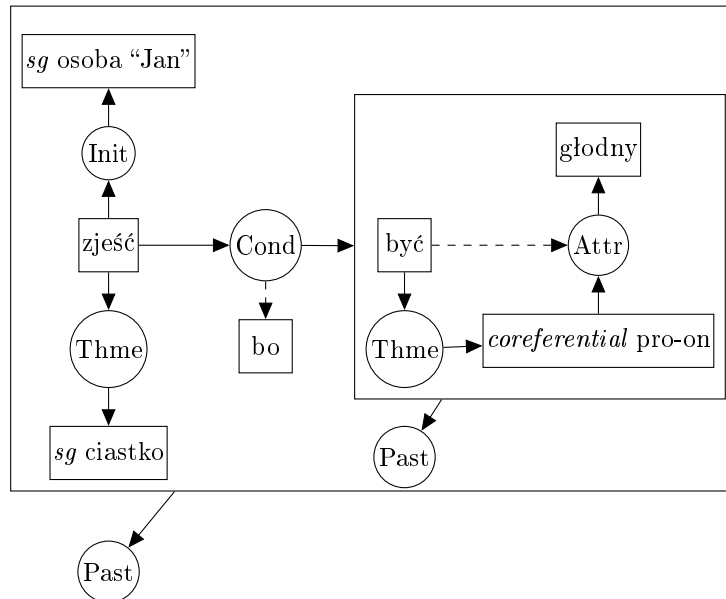
12 Przyczyna i cel

Role Condition i Purpose wiążą zdarzenia i sytuacje z sytuacjami będącymi ich przyczyną bądź celem.

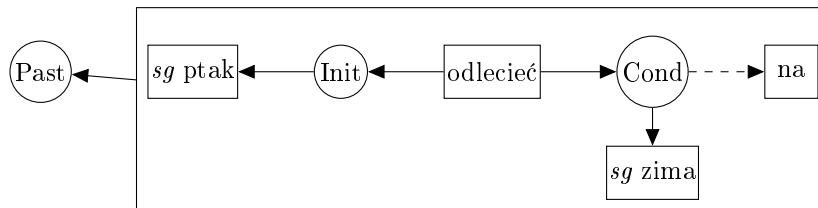
$$\begin{aligned}
(\mathcal{M}, v \models \text{COND}(x, r, y)) & \iff v(r)(v(x).cond, v(y)) \\
(\mathcal{M}, v \models \text{PURP}(x, r, y)) & \iff v(r)(v(x).purp, v(y))
\end{aligned}$$

Role Condition i Purpose mogą wiązać zdarzenie bądź sytuację, np *Jan zjadł ciastko, bo był głodny*

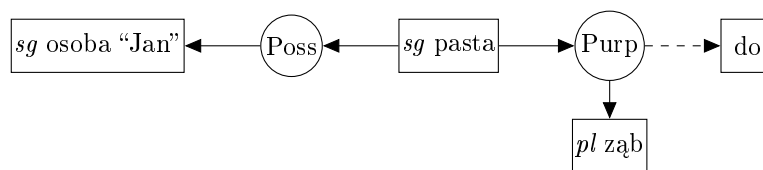




Przyimek *na* w jednym ze swoich znaczeń bierze określenie czasu i wiąże ten czas z celem czynności. *Ptak odleciał na zimę*



Rola Purpose użyta z rzeczownikami oznacza przeznaczenie obiektu, na przykład *pasta do zębów Jana* jest reprezentowana jako



Pojęciem, które wiąże relacja Condition (typ semantyczny pojęć *bo*, *ponieważ*, *z powodu*) to pojęcie CZEMU, lub *przyczyna*. *przyczyna* to rola jaką pełni sytuacja.

13 Cechy

Zgodnie z konwencją przyjętą w Walentym rozróżniamy cechy uczestników sytuacji i sposób wykonania czynności określając pierwsze z nich za pomocą predykatu ATTR a drugie za pomocą predykatu MANR.

Np *intensywnie różowy słoń*

$$\text{TYPE}(i, \text{intensywnie}) \wedge \text{TYPE}(r, \text{różowy}) \wedge \text{TYPE}(s, \text{słoń}) \wedge \text{MANR}(r, i) \wedge \text{ATTR}(s, r)$$

Definicje obu predykatów są identyczne:

$$(\mathcal{M}, v \models \text{ATTR}(x, y)) \iff v(y).prtc = v(x)$$

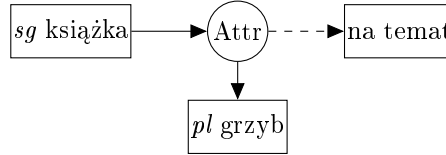
$$(\mathcal{M}, v \models \text{MANR}(x, y)) \iff v(y).prtc = v(x)$$

a algorytm znaczeniowy dla przysłówka jest taki sam jak dla odpowiadającego mu przymiotnika

$$\text{różowy} \equiv \lambda x x.prtc.color = x \wedge ???(x)$$

$$\text{intensywnie} \equiv \lambda x x.prtc.natezenie = x \wedge ???(x)$$

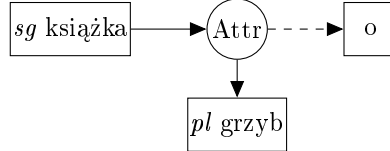
W przypadku wyrażeń przyimkowych opisujących cechy rzeczowników stosujemy rolę Attr, np.: *książka na temat grzybów, książka o grzybach*:



$$\text{TYPE}(k, \text{książka}) \wedge \text{TYPE}(t, \text{na temat}) \wedge \text{TYPE}(g, \text{grzyb}) \wedge \text{ATTR}(k, t, g)$$

$$(\mathcal{M}, v \models \text{ATTR}(x, a, y)) \iff v(a).prtc = v(x) \wedge v(a).val = v(y)$$

$$\text{na temat} \equiv \lambda x x.prtc.topic = x.val \wedge ???(x)$$



na temat jest przyimkiem złożonym traktowanym jako pojedyncza jednostka leksykalna, natomiast *o* w znaczeniu użytym w powyższym przykładzie jest jego synonimem. Przyimki złożone wskazują w swojej treści nazwę cechy, której dotyczą.

14 Relacja posiadania

Predykat $\text{POSS}(x, y)$ symbolizuje relację, w której y jest właścicielem x , np. *piłka chłopca*:

$$\text{TYPE}(p, \text{piłka}) \wedge \text{TYPE}(t, \text{chłopiec}) \wedge \text{POSS}(p, t)$$

Relację POSS definiujemy jako atrybut poss w modelu:

$$(\mathcal{M}, v \models \text{POSS}(x, y)) \iff v(x).poss = v(y)$$

15 Relacja uczestnictwa

Predykat $\text{PRTC}(x, y)$ symbolizuje relację, w której znaczenie x definiuje konkretny atrybut wiążący x z y , a powiązanie to jest ontologicznie konieczne. Na przykład we frazie *ojciec Józefa* byt oznaczony jako *ojciec* musi mieć dziecko, a byt oznaczony jako *Józef* musi mieć ojca. Przy czym, zarówno *Józef* jak i jego *ojciec* nie muszą istnieć (żyć) w czasie do którego odnosi się wypowiedziana fraza. Formalnie predykat PRTC definiujemy jako:

$$(\mathcal{M}, v \models \text{PRTC}(x, y)) \iff v(x).\text{prtc} = v(y)$$

Powiązanie x z y za pomocą konkretnej relacji zależy od znaczenia x , dlatego jest częścią jego algorytmu znaczeniowego (tak jak ma to miejsce przy podstawowych rolach tematycznych). Składniowo relacja PRTC jest wyrażana przez modyfikator rzeczownikowy w dopełniaczu.

Przykładowo frazie *ojciec Józefa* będzie odpowiadać formuła logiczna

$$\text{TYPE}(x, \text{ojciec}) \wedge \text{HASNAME}(y, \text{'Józef'}) \wedge \text{PRTC}(x, y)$$

Cecha y do której odnosi x jest wyznaczona przez algorytm znaczeniowy pojęcia *ojciec*:

$$\text{ojciec} \equiv \lambda x \ x.\text{prtc}.\text{ojciec} = x \wedge \text{mężczyzna}(x)$$

Czy odnośną cechą y -a jest posiadanie ojca? Trudno się tego domyślić na podstawie tego egzemplifikującego algorytmu.

Koniunkt $\text{mężczyzna}(x)$ jest zupełnie zbędny, każdy ojciec jest mężczyzną, równie dobrze można dodać $\text{istota_żywa}(x)$

Wtedy

$$\begin{aligned} (\mathcal{M}, v \models \text{TYPE}(x, \text{ojciec}) \wedge \text{HASNAME}(y, \text{'Józef'}) \wedge \text{PRTC}(x, y)) &\iff \\ v(x).\text{prtc}.\text{ojciec} = v(x) \wedge \text{mężczyzna}(v(x)) \wedge & \\ \wedge v(x).\text{prtc} = v(y) \wedge v(y).\text{name} = \text{'Józef'} & \\ \implies v(y).\text{ojciec} = v(x) \wedge \text{mężczyzna}(v(x)) \wedge v(y).\text{name} = \text{'Józef'} & \end{aligned}$$

Brakująca relatywizacja

$\text{PRTC}(x, y)$ musi być relatywizowane do zadanego algorytmu znaczeniowego przysługującego x -owi. Dlaczego: Weźmy zdanie „Jan III to syn Jana II, wnuk Jana I”. Zgodnie z przedostatnią linią powyższej równoważności (koniunkt pierwszy) reprezentacja tego zdania zawiera jako podformułę $v(j3).\text{prtc} = v(j2) \wedge v(j3).\text{prtc} = v(j1)$, a zatem Jan II i Jan I musieliby być samą osobą.

Funkcja one-to-many / pole .ojciec

Traktowanie tak jak funkcji relacji bycia ojcem $v(y).ojciec = x$ działa przypadkiem dla *ojciec*, bo każdy ma jednego biologicznego ojca, ale już dziadków ma się dwóch, więc dla *dziadek* (i większości innych) jest już potrzebny porządnny zapis relacyjny, np.

$$v(y).dziadek \ni v(x)$$

$$v(y).dziadek = \{v(x)\}$$

$$\text{lub normalnie } dziadek(v(y), v(x))$$

Funkcja one-to-many / pole .prtc

Traktowanie jak funkcji relacji uczestnictwa *.prtc* działa przypadkiem w jedno-jednoznaczny uwikłaniu $v(x).prtc.ojciec = v(x)$, bo każdy ojciec jest jedynym ojcem swoich dzieci, ale tak naprawdę sam term $v(x).prtc$ jest niezdefiniowany, ponieważ nie zawsze da się wyznaczyć jedyną osobę, dla której x jest ojcem. Dlatego zamiast $v(y).prtc = v(x)$ potrzebny jest porządnny zapis relacyjny np.

$$v(y).prtc \ni v(x)$$

$$v(y).prtc = \{v(x)\}$$

$$\text{lub normalnie } prtc(v(y), v(x))$$

Z kolei algorytm dla pojęcia *dziadek* może mieć postać:

$$\mathbf{dziadek} \equiv \lambda x x.\underline{prtc}.ojciec.ojciec = x \wedge \mathbf{mężczyzna}(x)$$

Co z

$$\mathbf{dziadek} \equiv \lambda x x.\underline{prtc}.matka.ojciec = x \wedge \mathbf{mężczyzna}(x)$$

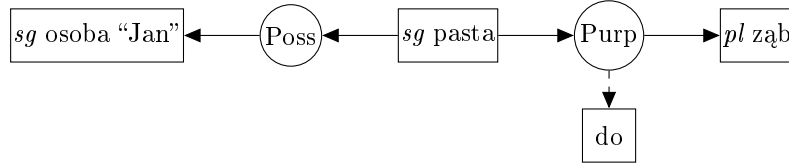
W powyższych przykładach pojęcie składniowo nadrzędne określało cechę pojęcia podrzędnego. Są sytuacje, w których pojęcie podrzędne jest komponentem nadrzędnego, np *zbiór osób*, czy *sekwencja zdarzeń*, w którym słowo *zbiór* ma algorytm:

$$\mathbf{zbiór} \equiv \lambda x x.\underline{prtc} = x.elementy \wedge \dots$$

Osobny przypadek stanowią czynności, wydarzenia i czyny, dla których predykat PRTC jest może zostać uszczegółowiony do jednej z podstawowych rolach tematycznych, np we frazie *powrót Michała* mamy rolę INIT, a we frazie *sprzedaż samochodu* rolę THME.

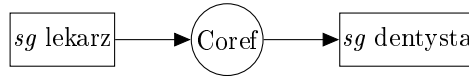
16 Relacja quasi-posiadania i identyczności

Przydawki przyimkowe domyślnie traktujemy tak jak argumenty czasownika. Na przykład *pasta do zębów Jana* jest reprezentowana jako



17 Relacja identyczności

Relacja identyczności komunikowana jest przez apozycję oraz przydawkę rzeczoną i zachodzi między odniesieniami obu wyrażeń. Apozycja (dwa rzeczowniki uzgodnione pod względem przypadku) wyraża dwa określenia tego samego obiektu. Może to być typ i nazwa, albo dwa różne typy, np. *lekarz dentysta* zapiszemy jako

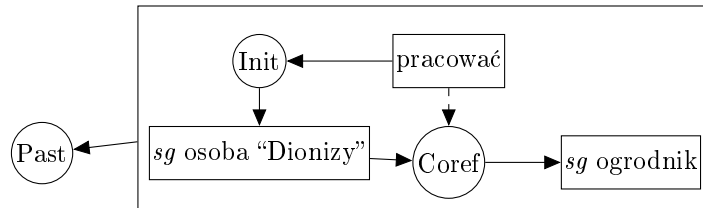


$$\text{TYPE}(l, \text{lekarz}) \wedge \text{TYPE}(d, \text{dentysta}) \wedge \text{COREF}(l, d)$$

Koreferencyjność definiujemy jako

$$(\mathcal{M}, v \models \text{COREF}(x, y)) \iff v(x) = v(y)$$

Relacja identyczności może zachodzić również pomiędzy podrzędnikami czasownika, np. w zdaniu *Dionizy pracował jako ogrodnik*. Walentym zjawisko to oznaczone jest rolą Attribute.



Przerywana strzałka pomiędzy *pracować* a relacją “=” oznacza, że relacja identyczności pomiędzy *Dionizym* a *ogrodnikiem* jest parametryzowana przez zdarzenie *pracować*.

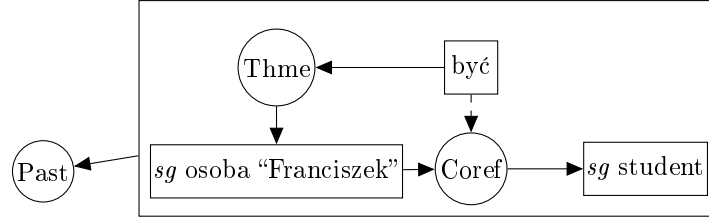
$$\text{TYPE}(d, \text{osoba}) \wedge \text{HASNAME}(d, \text{'Dionizy'}) \wedge \text{TYPE}(p, \text{pracować}) \wedge$$

$$\wedge \text{TYPE}(o, \text{ogrodnik}) \wedge \text{INIT}(d, p) \wedge \text{COREF}_p(d, o)$$

Koreferencyjność parametryzowaną przez zdarzenie, podobnie jak wszystkie inne parametryzowane przez zdarzenie relacje, zachodzą w zmiennym w czasie stopniu w trakcie zdarzenia. Atrybut *deg* wskazuje w jakim stopniu zachodzi relacja, a notacja $[\varphi]_{a.deg}$ oznacza, że w chwili t relacja φ jest prawdziwa w stopniu $a.deg(t)$.

$$(\mathcal{M}, v \models \text{COREF}_a(x, y)) \iff [v(x) = v(y)]_{v(a).deg}$$

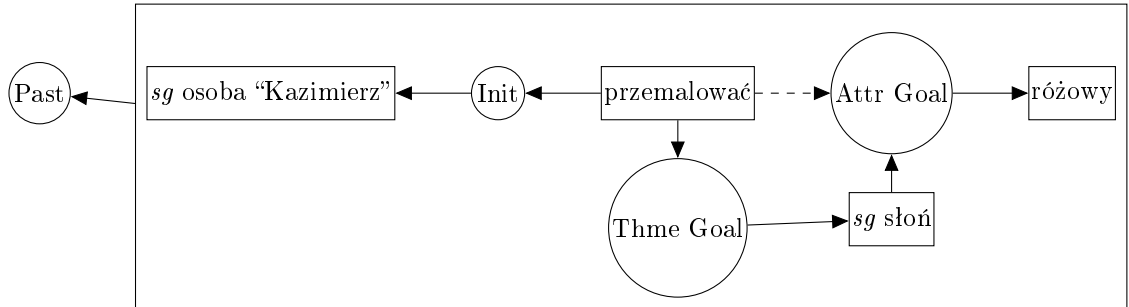
Analogicznie reprezentujemy konstrukcje predykatywne w zdaniach *Franciszek był studentem*



$$\begin{aligned} & \text{TYPE}(f, \text{osoba}) \wedge \text{HASNAME}(f, \text{'Franciszek'}) \wedge \text{TYPE}(b, \text{być}) \wedge \\ & \wedge \text{TYPE}(s, \text{student}) \wedge \text{THME}(b, f) \wedge \text{COREF}_b(f, s) \end{aligned}$$

18 Cechy zmieniające się

Cechy danego przedmiotu mogą być składniowo wyrażone przez podrzędnik czasownika. Słownik walencyjny Walenty sygnalizuje takie sytuacje nadając podrzędnikowi wyrażającemu cechę rolę Attribute oraz czyniąc go kontrolowanym przez podrzędnik będący nosicielem cechy, np *Kazimierz przemalował słoń na różowo*:

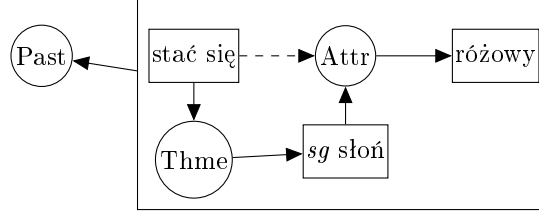


Przerywana linia łącząca *przemalować* i *Attr Goal* oznacza że relacja *Attr Goal* zmienia się w czasie trwania sytuacji w miarę postępów procesu malowania i nabierania przez słońce cechy bycia różowym.

$$\text{TYPE}(k, \text{osoba}) \wedge \text{HASNAME}(k, \text{'Kazimierz'}) \wedge \text{TYPE}(p, \text{przemalować}) \wedge$$

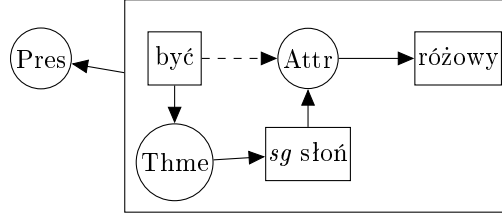
$$\begin{aligned}
& \wedge \text{TYPE}(s, \text{słoń}) \wedge \text{TYPE}(r, \text{różowy}) \wedge \text{INIT}(p, k) \wedge \\
& \wedge \text{THMEGOAL}(p, s) \wedge \text{ATTRGOAL}_p(s, r) \\
& (\mathcal{M}, v \models \text{ATTRGOAL}_a(x, y)) \iff [v(y).prtc = v(x)]_{v(a).deg_goal}
\end{aligned}$$

Analogicznie zinterpretujemy zdanie *Słoń stał się różowy*

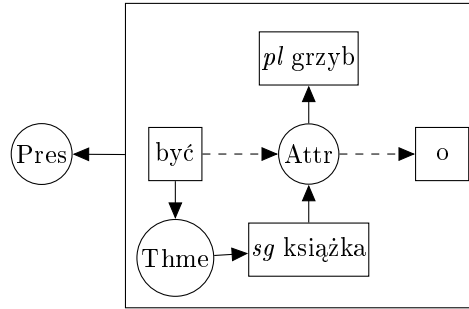


$$\begin{aligned}
& \text{TYPE}(t, \text{stać się}) \wedge \text{TYPE}(s, \text{słoń}) \wedge \text{TYPE}(r, \text{różowy}) \wedge \\
& \wedge \text{THME}(t, s) \wedge \text{ATTR}_t(s, r) \\
& (\mathcal{M}, v \models \text{ATTR}_a(x, y)) \iff [v(y).prtc = v(x)]_{v(a).deg}
\end{aligned}$$

Zdania z czasownikiem *być* stanowiące o cechach interpretujemy analogicznie do powyższego np. *Słoń jest różowy*, *Książka jest o grzybach*.



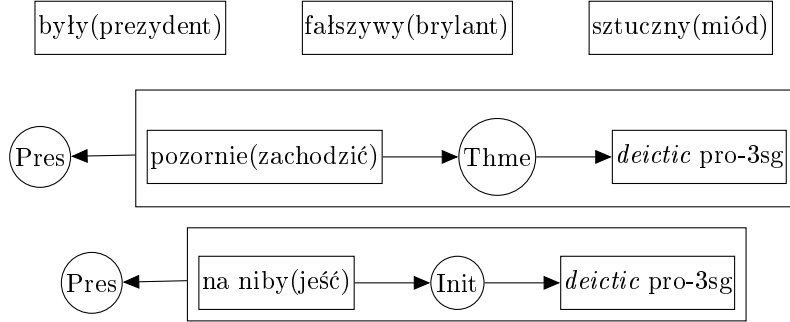
$$\begin{aligned}
& \text{TYPE}(b, \text{być}) \wedge \text{TYPE}(s, \text{słoń}) \wedge \text{TYPE}(r, \text{różowy}) \wedge \\
& \wedge \text{THME}(b, s) \wedge \text{ATTR}_b(s, r)
\end{aligned}$$



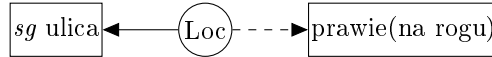
$$\begin{aligned}
& \text{TYPE}(b, \text{być}) \wedge \text{TYPE}(k, \text{książka}) \wedge \text{TYPE}(o, o) \wedge \text{TYPE}(g, \text{grzyb}) \wedge |g| > 1 \wedge \\
& \wedge \text{THME}(b, k) \wedge \text{ATTR}_b(k, o, g)
\end{aligned}$$

19 Modyfikatory nieintersektywne

Funkcja modyfikacji nieintersektywnej zachodzi między wyrażeniem określającym a jego nieintersektywnym określnikiem, który może być przymiotnikiem (*były prezydent*, *fałszywy brylant*, *sztuczny miód*), przysłówkiem (*pozornie zachodzi*) lub wyrażeniem przyimkowym (*je na niby*).

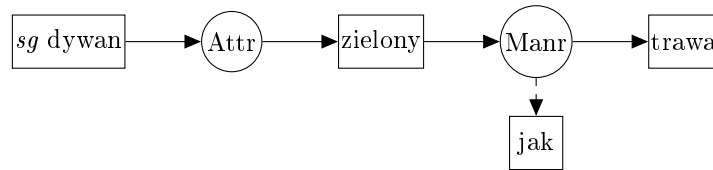


Przyimki mogą być też modyfikowane nieintersektywnie np. *prawie na rogu ulicy*



20 Stopniowanie przymiotników i przysłówków oraz konstrukcje porównawcze, relacyjne i wprowadzające kolejność

dywan zielony jak trawa,



$$\text{TYPE}(d, \text{dywan}) \wedge \text{TYPE}(z, \text{zielony}) \wedge \text{TYPE}(j, \text{jak}) \wedge \text{TYPE}(t, \text{trawa}) \wedge \\ \text{ATTR}(d, z) \wedge \text{MANR}(z, j, t)$$

$$(\mathcal{M}, v \models \text{MANR}(x, a, y)) \iff v(a).prtc = v(x) \wedge v(a).val = v(y)$$

$$\text{zielony} \equiv \lambda x x.prtc.color = x \wedge ???(x)$$

$$\mathbf{jak} \equiv \lambda x x.prtc.prtc = x.val$$

Wtedy

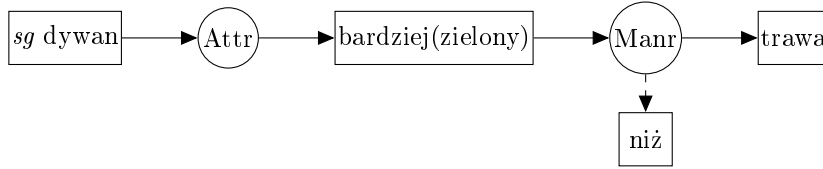
$$(\mathcal{M}, v \models \text{TYPE}(z, \text{zielony}) \wedge \text{TYPE}(j, \mathbf{jak}) \wedge \text{MANR}(z, n, t) \iff$$

$$v(z).prtc.color = v(z) \wedge ???(v(z)) \wedge v(j).prtc.prtc = v(j).val \wedge$$

$$v(j).prtc = v(z) \wedge v(j).val = v(t) \implies$$

$$v(z).prtc.color = v(z) \wedge v(z).prtc = v(t) \implies v(t).color = v(z)$$

Zakładamy, że oba sposoby stopniowania przymiotników i przysłówków (przez dodanie *bardziej* lub *najbardziej* oraz przez dodanie afiksów) są równoważne semantycznie, np. *najweselszy* i *najbardziej wesóły* znaczą to samo. Dlatego traktujemy wszystkie wystąpienia stopnia wyższego i najwyższego jakby były analityczne. Np. *dywan zieleńszy niż trawa*,



$$\text{TYPE}(d, \text{dywan}) \wedge \text{TYPE}(z, \text{bardziej(zielony)}) \wedge \text{TYPE}(n, \text{niż}) \wedge \text{TYPE}(t, \text{trawa}) \wedge \\ \text{ATTR}(d, z) \wedge \text{MANR}(z, n, t)$$

$$(\mathcal{M}, v \models \text{MANR}(x, a, y)) \iff v(a).prtc = v(x) \wedge v(a).val = v(y)$$

$$\mathbf{zielony} \equiv \lambda x x.prtc.color = x \wedge ???(x)$$

$$\mathbf{bardziej(a)} \equiv \lambda x \exists y \mathbf{a}(y) \wedge \mathbf{a}(x) \wedge x.degree > y.degree \wedge x.comp = y$$

$$\mathbf{niż} \equiv \lambda x x.prtc.comp.prtc = x.val$$

Wtedy

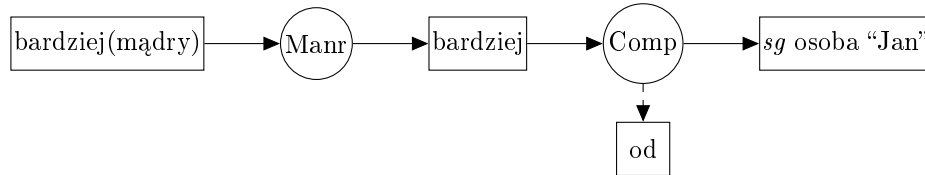
$$(\mathcal{M}, v \models \text{TYPE}(z, \text{bardziej(zielony)}) \wedge \text{TYPE}(n, \text{niż}) \wedge \text{MANR}(z, n, t) \iff$$

$$\exists y \mathbf{zielony}(y) \wedge \mathbf{zielony}(v(z)) \wedge v(z).degree > y.degree \wedge v(z).comp = y \wedge$$

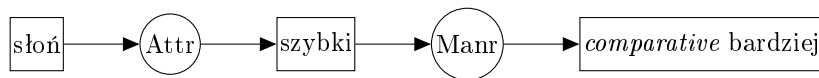
$$v(n).prtc.comp.prtc = v(n).val \wedge v(n).prtc = v(z) \wedge v(n).val = v(t) \implies$$

$$\exists y \mathbf{zielony}(y) \wedge \mathbf{zielony}(v(z)) \wedge v(z).degree > y.degree \wedge v(t).color = y$$

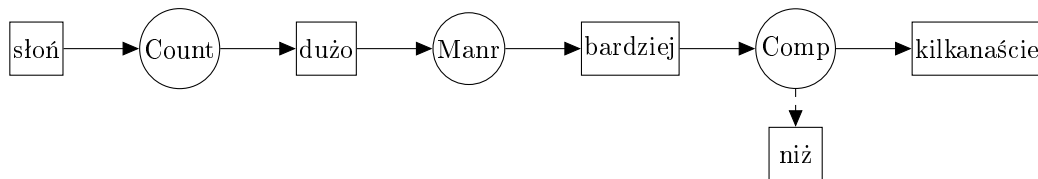
mądrzejsza od Jana



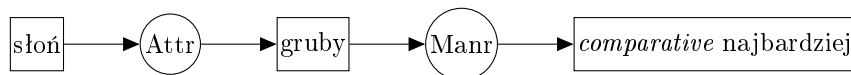
Comparative (Comp) oznacza argument porównawczy. Argument ten może pozostać niejawny, np. *szybszy słoń*



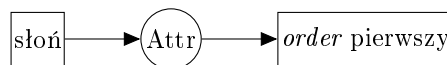
Liczność zadana przez konstrukcję porównawczą: *więcej niż kilkanaście słoń*



W przypadku stopnia najwyższego występuje niejawny argument porównawczy, np. *najgrubszy słoń*

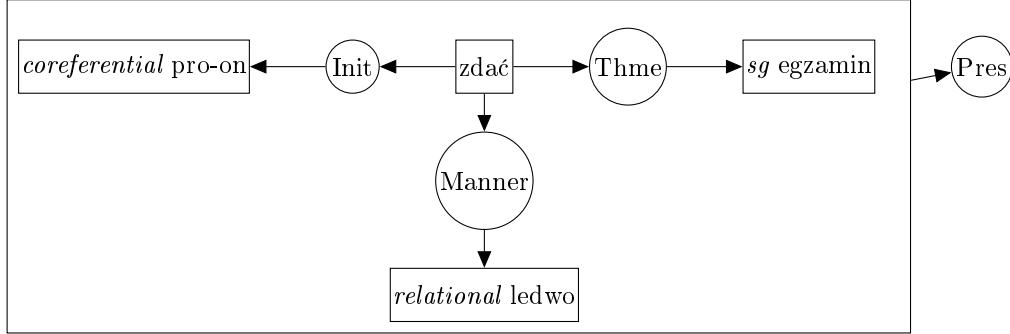


Leksemy takie jak *pierwszy*, *drugi*, *ostatni*, *kolejny*, *jeszcze*, *już* mają niejawny, zależny od kontekstu argument porządkowy, np. *pierwszy słoń*:



Znaczenie leksemów relacyjnych zależy w pewnym ustalonym zakresie od znaczenia nadrzędnika. Fakt ten jest reprezentowany przez argument relacyjny (*relational*), np. *ledwo znaczące prawie nie* w zdaniu *Ledwo zdał*

egzamin



Inne leksemy relacyjne: *dużo, sporo, ledwo, niedużo, już, jeszcze, aż*.

21 Sytuacje, procesy i relacje czasowe

Każdy proces wymieniowy w zdaniu umieszczamy w osobnym kontekście sytuacyjnym. W kontekstach sytuacyjnych uczestnicy istnieją a relacje między nimi zachodzą przez cały czas trwania sytuacji, chyba że uczestnik jest połączony relacją z procesem. W takiej sytuacji może on zostać w trakcie procesu stworzony, czyli zaistnieć dopiero na jego końcu, może powstawać stopniowo, może też przestać istnieć. Z kolei relacje wychodzące z procesów mogą się zmieniać w trakcie procesu. Natomiast relacje czasowe przypisane zdarzeniu dotyczą każdego uczestnika sytuacji a czas zdarzenia przysługuje całej sytuacji.

Na potrzeby reprezentacji za pomocą grafów semantycznych przyjmujemy, że relacje czasowe wiążą czas z sytuacjami, a znajdujące się w kontekstach pojęcia i relacje uznamy za fluenty niejawnie przez ten czas parametryzowane.

22 Czasowniki wyrażające stan

Słoń waży tonę.

$\text{TYPE}(s, \text{słoń}) \wedge \text{TYPE}(w, \text{ważyć}) \wedge \text{TYPE}(t, \text{tona}) \wedge \text{THME}(w, s) \wedge \text{MEASURE}(w, t)$

$\text{ważyć} \equiv \lambda x x.\text{theme.waga} = x.\text{measure}$

Waga słonia wynosi tonę.

$\text{TYPE}(w, \text{waga}) \wedge \text{TYPE}(s, \text{słoń}) \wedge \text{TYPE}(y, \text{wynosić}) \wedge \text{TYPE}(t, \text{tona}) \wedge$

$$\begin{aligned} & \wedge \text{PRTC}(w, s) \wedge \text{THME}(y, w) \wedge \text{MEASURE}(y, t) \\ \mathbf{waga} & \equiv \lambda x x.\text{prtc.waga} = x \wedge \mathbf{jednostka\ masy}(x) \\ \mathbf{wynosić} & \equiv \lambda x x.\text{thme} = x.\text{measure} \end{aligned}$$

Waga słonia jest równa tonie.

$$\text{TYPE}(w, \text{waga}) \wedge \text{TYPE}(s, \text{słoń}) \wedge \text{TYPE}(b, \text{być}) \wedge \text{TYPE}(r, \text{równy}) \wedge \text{TYPE}(t, \text{tona})$$

Promień koła jest równy połowie jego średnicy. Nazywam się Kalasanty.

$$\mathbf{nazywać\ się} \equiv \lambda x x.\text{thme.name} = x.\text{thme2}$$

Piłka leży pod stołem.

$$\mathbf{leżeć} \equiv \lambda x x.\text{thme.loc} = x.\text{loc}$$

Piłka jest pod stołem.

$$\mathbf{być} \equiv \lambda x x.\text{thme.loc} = x.\text{loc}$$

23 Czasowniki wyrażające zmianę stanu

Słoń schódł Waga słonia staje się równa tonie.