



Politechnika Wroclawska

# Reprezentacja wiedzy i wnioskowanie

Dariusz Banasiak

Katedra Informatyki Technicznej

Wydział Elektroniki

**AI** to nauka o komputerowych modelach wiedzy umożliwiających rozumienie, wnioskowanie i działanie.

Inteligentne zachowanie uzależnione jest od posiadanej wiedzy.

W sztucznej inteligencji przez reprezentację wiedzy rozumie się połączenie struktur danych oraz procedur interpretacyjnych operujących na tych danych. Procedury te, właściwie użyte powinny prowadzić do inteligentnego zachowania systemu.

Wiedza może mieć różny charakter. Ogólnie, wyróżnia się następujące rodzaje wiedzy:

- **wiedza o obiektach**, opisująca cechy i właściwości obiektów fizycznych np. *Ptaki mają skrzydła, Kanarek jest ptakiem, Kanarek jest żółty* itd.
- **wiedza o zdarzeniach**, związana jest z dynamiczną naturą świata (oprócz informacji o samym zdarzeniu opisuje również następstwa przyczynowe oraz sekwencję czasową zdarzeń) np. *Rano padał deszcz, Chłopiec poszedł do szkoły, Samochód potrącił pieszego*

- **umiejętności**, czyli wiedza jak wykonywać określone zadanie (ten typ wiedzy jest bardzo trudny lub niemożliwy do przekazania w sposób werbalny np. jazda samochodem, pisanie na klawiaturze, lepienie garnków itd.)
- **meta-wiedza**, czyli wiedza o samej wiedzy (może służyć do oceny wiarygodności uzyskanych danych uwzględniając np. niedoskonałość percepcji człowieka lub zastosowanych metod pomiarowych, także wiedza ogólna wykorzystywana np. w procesie wnioskowania)
- **wierzenia** (przekonania), czyli prawdziwe lub fałszywe, z góry powzięte nastawienia, które mogą wpływać na proces wnioskowania.

Dwa główne podejścia do reprezentacji wiedzy:

- deklaratywne
- proceduralne.

W podejściu deklaratywnym wiedza z danej dziedziny jest zbiorem specyficznych faktów, a korzystanie z wiedzy polega na stosowaniu do tego zbioru ogólnych procedur manipulacji faktami (np. reprezentacja logiczna).

W reprezentacjach deklaratywnych następuje wyraźne oddzielenie wiedzy z danej dziedziny od sposobu wykorzystania tej wiedzy w procesie wnioskowania.

W podejściu proceduralnym zakłada się, że przeważającą część wiedzy o świecie stanowią informacje o procesach i działaniach.

"Wiedzieć" jest równoważne z "wiedzieć jak": znajomość danego pojęcia sprowadza się głównie do umiejętności manipulacji tym pojęciem.

W reprezentacji proceduralnej wiedza zawarta jest w procedurach (podprogramach), które wiedzą jak należy się zachować w określonej sytuacji.

### Przykład

Reprezentacja wiedzy dotyczącej kolejności liter w alfabecie

Podejście deklaratywne:

Fakty: A stoi przed B, B stoi przed C, C stoi przed D, ...

Reguła: **JEŻELI** (*I1* stoi przed *I2*) i (*I2* stoi przed *I3*)  
**TO** (*I1* stoi przed *I3*)

Podejście proceduralne:

Procedura, która porównuje kolejność odpowiednich liter w alfabecie (np. wykorzystując ich kody ASCII)

Wybór określonego podejścia w dużej mierze zależy od rodzaju wiedzy, którą chcemy reprezentować.

**Metody deklaratywne** służą do reprezentowania wiedzy o charakterze statycznym np. charakterystyka obiektów, trwałe relacje między nimi, zależności logiczne między faktami itd.

**Metody proceduralne** są przydatne do reprezentowania wiedzy o charakterze dynamicznym np. ciągi akcji, procesy, algorytmy i reguły postępowania.



Ze sposobem reprezentacji wiedzy ściśle związane są zagadnienia jej wykorzystania. Należy rozpatrzeć trzy aspekty wykorzystania wiedzy:

- gromadzenie nowej wiedzy,
- wydobywanie z bazy wiedzy faktów dotyczących danego problemu,
- rozumowanie (wnioskowanie) przy użyciu faktów w poszukiwaniu rozwiązania.

**Gromadzenie wiedzy** polega na dodawaniu do bazy wiedzy nowych faktów. Proces ten powinien uwzględniać związki nowych faktów z już istniejącą wiedzą (klasyfikacja wiedzy) eliminując np. wielokrotne powtarzanie tych samych faktów (integracja wiedzy).

**Wydobywanie wiedzy** nie jest zagadnieniem prostym w przypadku systemów informatycznych. Dla dużych baz wiedzy (zawierających znaczną liczbę faktów) wybór faktów koniecznych do rozwiązania danego problemu nie jest zagadnieniem łatwym. Stosuje się w tym celu np. grupowanie faktów powiązanych ze sobą (lumping).

**Wnioskowanie** jest konieczne wówczas, gdy system nie jest w stanie wprost znaleźć rozwiązania określonego problemu.

Założmy, że baza wiedzy zawiera następujące fakty:

*Ptak ma skrzydła*

*Kanarek jest ptakiem*

Odpowiedź na pytanie: „*Czy kanarek ma skrzydła?*” wymaga wnioskowania – połączenia ze sobą dwóch faktów i wyciągnięcia odpowiedniego wniosku.

Rozróżnia się różne typy wnioskowania: formalne, proceduralne, przez analogię, przez uogólnienie.

W systemach sztucznej inteligencji stosowane są różne metody reprezentacji wiedzy. Do najważniejszych z nich należy zaliczyć:

- reprezentacje logiczne (rachunek zdań, rachunek predykatów, rachunek klauzul), w których fakty zapisywane są w postaci formuł, a wnioskowanie opiera się na uniwersalnych metodach dowodzenia twierdzeń np. rezolucji
- sieci semantyczne (semantic networks), mają one postać etykietowanego grafu skierowanego, w którym wierzchołki reprezentują obiekty, pojęcia, wartości itd., natomiast krawędzie wyrażają relacje między wierzchołkami

- reprezentacje typu (**obiekt, atrybut, wartość**), w których pojedynczy fakt zapisywany jest w postaci podanej trójki, a bazę wiedzy stanowi zbiór takich trójek;
- reguły produkcji, w metodzie tej bazę wiedzy stanowi zbiór faktów oraz reguł typu:

**IF warunek THEN konkluzja**

(metoda często stosowana w systemach ekspertowych)

- ramy (frames), stanowią połączenie metod deklaratywnych i proceduralnych; rama składa się ze szczelin (slots) opatrzonych nazwami, których zadaniem jest przechowywanie pewnej części wiedzy o obiekcie reprezentowanym przez ramę (ze szczeliną mogą być związane również procedury)

- zależności pojęciowe (Conceptual Dependency), służą do reprezentowania znaczenia zdań języka naturalnego w postaci specjalnej sieci. Wyróżnia się w nim sześć klas pojęć: pojęcia nominalne, akcje, lokalizacje, czasy, modyfikatory akcji i modyfikatory obiektów
- scenariusze (scripts), są to struktury opisujące przedmioty, role, warunki początkowe, rezultaty i ciągi scen, których opis bazuje na metodzie zależności pojęciowych (pojedynczy scenariusz może opisywać np. przebieg procesu konsumowania posiłku w restauracji, w którym rolami są klient, kelner, właściciel restauracji, kucharz, natomiast kolejne sceny to: wejście klienta do restauracji, zamówienie dania, konsumpcja, wyjście klienta z restauracji).

Język naturalny nie jest precyzyjny, w związku z czym niewłaściwe używanie logiki może prowadzić do wyprowadzania fałszywych stwierdzeń lub uznania za prawdziwe stwierdzeń, które nie wynikają z przesłanek.

### Przykład 1

**P1:** Wszyscy ludzie są śmiertelni

**P2:** X jest człowiekiem

**W:** X jest śmiertelny

### Przykład 2

**P1:** Jakież samochody terkoczą

**P2:** Mój samochód jest jakimś samochodem

**W:** Mój samochód terkocze

### Rachunek zdań

Za pomocą rachunku zdań dany problem można opisać w postaci zdań. Zdania te nie są analizowane pod względem treści, ale każde zdanie posiada określoną wartość prawdy (np. 1 – prawda, 0 – fałsz).

Zdania oznaczane są określonymi symbolami (np. dużymi literami alfabetu A, B, C, ..).

Zdania można łączyć za pomocą spójników logicznych:

- $\neg$  (negacja),  $\wedge$  (koniunkcja),  $\vee$  (suma),
- $\Rightarrow$  (implikacja),  $\Leftrightarrow$  (równoważność).

**W ten sposób można tworzyć wyrażenia logiczne (formuły).**



## Reprezentacja wiedzy

Na podstawie prawdziwości zdań atomowych można również formułom przypisać wartość prawdy:

<b>P</b>	<b>Q</b>	<b><math>\neg P</math></b>	<b><math>P \wedge Q</math></b>	<b><math>P \vee Q</math></b>	<b><math>P \Rightarrow Q</math></b>	<b><math>P \Leftrightarrow Q</math></b>
0	0	1	0	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	0	0
1	1	0	1	1	1	1

*Paweł jest mężczyzną*

**P**

*Paweł jest mężczyzną, a Anna jest kobietą*

**$P \wedge Q$**

*Paweł jest mężem Anny lub Paweł jest mężem Sylwii*

**$R \vee S$**

*Gdy Paweł jest mężem Anny, to Paweł nie jest mężem Sylwii*

**$R \Rightarrow \neg S$**

Literał to formuła w postaci pojedynczego symbolu zmiennej (np.  $A$ ) lub zmiennej zaprzeczonej (np.  $\neg B$ ).

### Koniunkcyjna postać normalna (CNF)

Zdanie jest w postaci CNF (ang. Conjunctive Normal Form) jeżeli ma formę:

$$A_1 \wedge A_2 \wedge A_3 \wedge \dots \wedge A_n$$

w której każda klauzula  $A_i$  ma formę:

$$B_1 \vee B_2 \vee B_3 \vee \dots \vee B_m \quad (B_j \text{ są literałami})$$

### Dysjunkcyjna postać normalna (DNF)

Zdanie jest w postaci DNF (ang. Disjunctive Normal Form) jeżeli jest alternatywą klauzul, z których każda jest koniunkcją literałów:

$$(B_{11} \wedge B_{12} \wedge \dots B_{1k1}) \vee \dots \vee (B_{n1} \wedge B_{n2} \wedge \dots B_{nkn})$$

Równoważności przydatne do przekształcania formuł logicznych:

$$A \Leftrightarrow B \equiv (A \Rightarrow B) \wedge (B \Rightarrow A)$$

$$A \Rightarrow B \equiv \neg A \vee B$$

$$\neg (A \wedge B) \equiv \neg A \vee \neg B$$

$$\neg (A \vee B) \equiv \neg A \wedge \neg B$$

$$\neg \neg A \equiv A$$

$$A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$$

### Rachunek predykatów

Rachunek zdań nie jest wystarczająco bogaty, aby za jego pomocą wyrazić niektóre fakty. Rozpatrzmy wyrażenie arytmetyczne postaci:

$$x > y$$

Wyrażenie to nie jest ani prawdziwe, ani fałszywe. Jego prawdziwość zależy od wartości zmiennych  $x$  i  $y$ .

Rachunek predykatów jest rozszerzeniem rachunku zdań przez wprowadzenie kwantyfikatorów:

- $\forall x$  („dla każdego  $x$ ”)
- $\exists x$  („istnieje takie  $x$ , że”)

Rachunek predykatów jest bardzo ważnym narzędziem do reprezentacji wiedzy (programowanie w logice np. Prolog).

**Wyrażenie  $W(x)$** , w którym występuje zmienna  $x$  i które staje się zdaniem prawdziwym lub fałszywym, gdy w miejsce  $x$  podstawimy wartość zmiennej  $x$  nazywamy **funkcją zdaniową** lub **predykatem**.

Predykat składa się z nazwy i dowolnej liczby argumentów (termów). Termami mogą być stałe (numeryczne lub alfanumeryczne), zmienne oraz wyrażenia (np. kolejne predykaty).

W wyniku podstawienia stałych za zmienne otrzymujemy zdania prawdziwe lub fałszywe.

W zagadnieniach reprezentacji wiedzy predykat definiuje się jako własność obiektu lub relację między obiektami.

W przypadku zdań języka naturalnego jest to ta część zdania, w której coś się orzeka o podmiocie lub opisuje pewną właściwość lub relację.

*Piotr jest kierowcą*

kierowca(Piotr)

*Piotr lubi Ewę*

lubi(Piotr, Ewa)

*Jan jest ojcem Piotra*

ojciec(Jan, Piotr)

*Ewa lubi wysokich mężczyzn*

wysoki(X)  $\wedge$  mezczyzna(X)  $\Rightarrow$   
lubi(Ewa, X)

Ogólnie strukturę predykatu możemy przedstawić w następującej postaci:

$\text{nazwa\_predykatu}(\text{arg}_1, \text{arg}_2, \dots, \text{arg}_n)$ .

Predykaty mogą być połączone spójnikami logicznymi sumy  $\vee$  oraz iloczynu  $\wedge$ , znakiem implikacji  $\Rightarrow$  oraz znakiem równoważności  $\Leftrightarrow$ .

Jeżeli argumentami predykatów są wyłącznie stałe i zmienne to predykaty takie nazywamy predykatami pierwszego rzędu. Jeżeli jednym z argumentów predykatu jest predykat pierwszego rzędu to nazywany jest on predykatem drugiego rzędu, itd. (predykat trzeciego rzędu zawiera jako argument predykat drugiego rzędu).

### Sieci semantyczne

Po raz pierwszy sieci semantyczne zostały zastosowane jako model ludzkiej pamięci (Quillian, 1968).

Ogólnie, **siecią semantyczną** nazywamy każdy graf skierowany, którego wierzchołki i krawędzie są etykietowane (przypisano im znaczenie wyrażone słowami języka naturalnego lub innymi symbolami).

**Wierzchołki** mogą w sposób dowolny przedstawiać ogólne stany rzeczy (obiekty, zdarzenia, pojęcia abstrakcyjne itp.).

**Krawędzie** określają relacje między wierzchołkami, mogące wyrażać dowolne treści semantyczne.



Sieci semantyczne znalazły szerokie zastosowanie do reprezentacji wiedzy. Zaliczane są do deklaratywnych metod reprezentacji wiedzy.

Następna plansza przedstawia przykład sieci semantycznej opisującej fragment świata ludzi i zwierząt.

<b>AKO</b>	– bycie podklasą określonego obiektu
<b>ISA</b>	– bycie instancją, konkretnym egzemplarzem z określonej klasy
<b>POSS</b>	– posiadanie pewnych elementów jako własnych składowych
<b>OWN</b>	– posiadanie pewnych obiektów jako przedmiotu własności
<b>CAN</b>	– umiejętność wykonywania określonych czynności
<b>FEATURE</b>	– posiadanie pewnych cech
<b>COLOR</b>	– posiadanie określonego koloru
<b>AGE</b>	– bycie w określonym wieku
<b>BROTHER</b>	– posiadanie brata



### Metody wnioskowania

Ważnym elementem poszukiwania rozwiązania danego problemu (np. odpowiedzi na pytanie dotyczące pewnej dziedziny wiedzy) jest wnioskowanie.

Wnioskowanie to proces polegający na wyprowadzeniu (zgodnie z prawami logiki określonymi przez reguły wnioskowania) ze zdań uznanych za prawdziwe (przesłanek) nowych twierdzeń (wniosków).

Proces wnioskowania można zdefiniować w następujący sposób. W bazie wiedzy znajdują się fakty oraz reguły. Na ich podstawie należy udowodnić określoną hipotezę stanowiącą cel wnioskowania.

### Wnioskowanie wstecz

**Wnioskowanie wstecz** polega na wykazaniu prawdziwości hipotezy głównej na podstawie prawdziwości przesłanek.

Jeżeli prawdziwość przesłanki nie wynika bezpośrednio z faktów zawartych w bazie wiedzy, to przesłankę tę traktujemy jako nową hipotezę i próbujemy wykazać jej prawdziwość.

Jeżeli w ten sposób wykażemy prawdziwość wszystkich przesłanek określonej reguły, to konkluzja tej reguły również będzie prawdziwa.

Jest ona następnie wykorzystywana w procesie wnioskowania do udowodnienia kolejnych reguł.

### Przykład

Dane są fakty P, Q i R oraz następujące reguły:

$$\mathbf{R1: } P \wedge Q \Rightarrow S$$

$$\mathbf{R2: } R \Rightarrow T$$

$$\mathbf{R3: } S \wedge T \Rightarrow U$$

$$\mathbf{R4: } S \wedge R \Rightarrow V$$

Należy udowodnić hipotezę, że zdanie V jest prawdziwe.

## Rozwiązanie:

- Krok1:** czy V jest zawarte w bazie? -> NIE
- Krok2:** czy V stanowi konkluzję reguły w bazie? -> TAK (reguła R4)
- Krok3:** czy S jest zawarte w bazie? -> NIE
- Krok4:** czy S stanowi konkluzję reguły w bazie? -> TAK (reguła R1)
- Krok5:** czy P jest zawarte w bazie? -> TAK
- Krok6:** czy Q jest zawarte w bazie? -> TAK  
(udowodniono regułę R1, S jest prawdziwe)
- Krok7:** czy R jest zawarte w bazie? -> TAK  
(udowodniono regułę R4, V jest prawdziwe)
- Krok8:** STOP (cel został osiągnięty)

**Fakty:** P, Q, R

**R1:**  $P \wedge Q \Rightarrow S$

**R2:**  $R \Rightarrow T$

**R3:**  $S \wedge T \Rightarrow U$

**R4:**  $S \wedge R \Rightarrow V$

### Metoda rezolucji

Ważną metodą wnioskowania w systemach reprezentacji wiedzy jest metoda rezolucji (dowodzenie przez zaprzeczenie). Jest ona stosowana w języku PROLOG.

Metoda rezolucji sprowadza się do wykazania, że negacja zdania (które uważamy za prawdziwe) powoduje zaprzeczenie wszystkich faktów potwierdzających prawdziwość tego zdania.

Metoda ta jest procesem iteracyjnym. W każdym kroku porównywane są wyrażenia zawierające predykaty wzajemnie zanegowane.

Po eliminacji sprzecznych predykatów (np.  $P$  i  $\neg P$ ) powstaje nowe zdanie (tzw. resolvent).

Aby zastosować metodę rezolucji predykaty należy przedstawić w postaci kanonicznej, która nie zawiera znaków implikacji oraz spójników logicznych AND ( $\wedge$ ).

Korzystając z praw logicznych można zastąpić:

➤  $A \Rightarrow B$  przez  $\neg A \vee B$

➤  $\neg(A \wedge B)$  przez  $\neg A \vee \neg B$  (prawo de Morgana).

Dla faktów i reguł z poprzedniego przykładu otrzymamy:

**F1:** P

**F2:** Q

**F3:** R

**R1:**  $\neg P \vee \neg Q \vee S$

**R2:**  $\neg R \vee T$

**R3:**  $\neg S \vee \neg T \vee U$

**R4:**  $\neg S \vee \neg R \vee V$



# Metody wnioskowania

Przebieg wnioskowania ilustruje poniższy rysunek:

