

Sztuczna Inteligencja

Reprezentacja wiedzy I

Wstęp i logika klasyczna

Włodzisław Duch

Katedra Informatyki Stosowanej UMK

Google: Wlodzislaw Duch

[Strona wykładów](#)

Co było i będzie



- Szukanie.
 - Programy oparte na szukaniu
 - Szukanie i ludzkie myślenie
-
- Reprezentacja wiedzy – wstęp
 - Logiczna reprezentacja wiedzy
 - Logika rozmyta i przybliżona
 - Inne formy reprezentacji wiedzy

Reprezentacja wiedzy

AI była utożsamiana z **inżynierią wiedzy** (knowledge engineering).

- Czym jest wiedza? Jak ją reprezentować?
- Jaką z niej korzystać? Jak podejmować decyzje?

Jednym z 3 głównych kierunków filozofii jest *epistemologia* (jakie są dwa pozostałe? Też ważne dla AI).

Czasopisma naukowe na temat inżynierii wiedzy:

- Data & Knowledge Engineering – Elsevier Journal
- Knowledge Engineering Review, Cambridge Journal
- The Intern. J. of Software Engineering & Knowledge Engineering
- IEEE Transactions on Knowledge and Data Engineering
- Expert Systems: The Journal of Knowledge Engineering

Dostępne materiały

[10 Free Must-read Books on AI](#) (KDD Nuggets)

- Większość książek tu rekomendowanych dotyczy uczenia maszynowego.

Historia AI: [The Quest for Artificial Intelligence](#): *A History of Ideas and Achievements*, Nils J. Nilsson (2009, 707 pp). Książka w PDF.

David L. Poole and Alan K. Mackworth, *Artificial Intelligence: Foundations of Computational Agents*, 2nd ed, Cambridge University Press 2017, pp. 820

[A curated list of AI courses, books](#), video lectures and papers.

- [Knowledge-Based AI](#): Cognitive Systems, darmowe wykłady na Georgia Tech.

[Wykład prof. K. Grąbczewskiego](#) zawiera około 100 slajdów na temat logiki.

Rodzaje wiedzy

- Obiekty, przedmioty – najprostsza wiedza?
Opis za pomocą przestrzeni cech, zbioru przykładów lub ich uogólnień - prototypów.

Trudności ze zrozumieniem kategoryzacji przez ludzi.

- Zdarzenia, zmienność w czasie i przestrzeni.
- Umiejętności – wiedza niewerbalna.
Rozróżnienie wiedzy proceduralnej i deklaratywnej.
- Meta-wiedza, czyli wiem, że coś wiem (nic nie wiem).
- Wierzenia, przekonania lub ich brak.

Reprezentacja wiedzy

Używanie wiedzy

- Gromadzenie wiedzy, klasyfikacja i integracja
- Wydobywanie (akwizycja) wiedzy

Rozumowanie

- Formalne rozumowanie
- Proceduralne rozumowanie
- Rozumowanie przez analogię
- Rozumowanie przez uogólnianie
- Meta-rozumowanie

Reprezentacja wiedzy cd.

- „Ziarnistość” reprezentowanej wiedzy – ile szczegółów?
- Modularność wiedzy – jak łatwo zmodyfikować?
- Reprezentacje deklaratywne – trudności z reprezentacją sekwencji działań.
- Reprezentacje proceduralne – trudności z modyfikacją.

Teoria reprezentacji wiedzy jest słabo rozwinięta, w ekonomii zarządzanie wiedzą jest dużym działem, UE lansowała hasło „gospodarki opartej na wiedzy”.

Sposób reprezentacji wiedzy przez mózgi – niejasny, ale sporo wiemy.

Reprezentacja stanów

Najprostsza reprezentacja wiedzy:

- Opis stanu bazy danych, problemu, obiektu.
- Operatory pozwalające na transformację stanu
- Opis problemu i możliwości jego transformacji.

Reprezentacja proceduralna

Stan bazy: opis statycznej sytuacji.

Opis serii czynności wymaga procedury postępowania.

- Przykład: procedura rozbrojenia pocisku;
 operacja chirurgiczna.

Procedury często zapisywane są w postaci programu poddającego serii transformacji jakiś obiekt.

Zalety: procedury upraszczają proces szukania, wiedza nie jest oddzielona od sposobu jej użycia.

Podobne efekty osiągamy wprowadzając makrooperatory w przestrzeni stanów; odpowiada to wyuczaniu się nowych skojarzeń, porcjowaniu wiedzy (chunking).

Wady: zupełności i spójności zwykle nie da się osiągnąć.

Reprezentacja logiczna

Reprezentacja logiczna - wstęp

Logika – długie tradycje, „prawa myślenia” Boole’a.

Notacja logiczna pozwala wyrazić wiedzę przez deklaracje.

Notacja logiczna pozwala prowadzić rozumowania.

Czasami taka rep. jest naturalna, np. szukanie w Google:

Duch AND neural:

$$\begin{aligned} & \{p \in \text{WebPages} \mid \text{contain}(p, \text{Duch}) \ \& \ \text{contain}(p, \text{neural})\} \\ = & \{p \in \text{WebPages} \mid \text{contain}(p, \text{Duch})\} \cap \\ & \{p \in \text{WebPages} \mid \text{contain}(p, \text{neural})\} \end{aligned}$$

Znaczenie $A \wedge B$ wynika ze znaczenia składników i jest niezależne od kontekstu.

Rachunek zdań

Zdania mogą być prawdziwe lub nie, reprezentują fakty.

Zdania stanowią zbiór, do którego należą również zdania otrzymane przez zastosowanie operatorów logicznych:

- \sim nieprawda
- \wedge koniunkcja, i
- \vee alternatywa
- \rightarrow implikacja

- Do zdań można stosować kwantyfikatory:
 - \forall dla wszystkich
 - \exists istnieje

Rachunek zdań pozwala na wnioskowanie na kilka sposobów.

Logika predykatów

W logice zdań nie da się np. zapisać:

„Wszystkie koty są czarne”.

Potrzebna jest logiczna reprezentacja stwierdzeń o obiektach, ich własnościach, relacjach i funkcjach, dopuszczająca kwantyfikatory.

- Logika predykatów to logika stwierdzeń mających za argumenty obiekty, np. *ja*, *człowiek*, *kartka*.
- Predykaty mają argumenty i wartość logiczną.
- Predykat *jest-czerwony(x)*, *większy-od(x,y)*, *lżejszy-od(x,y)*
- Predykat *isa*, czyli „jest członkiem”.
- Logika predykatów dopuszcza kwantyfikatory.

Reprezentacja logiczna



$\exists x, \text{Ptak}(x)$, czyli istnieje przynajmniej jedno takie x ,
że $\text{Ptak}(x)=T$.

„Każdy ptak ma skrzydła” można zapisać jako:

$\forall x, \text{Ptak}(x) \Rightarrow \text{MaSkrzydła}(x)$

Wnioskowanie:

z prawdziwych faktów \Rightarrow nowe, prawdziwe fakty.

$\forall x. \text{Wróbel}(x) \Rightarrow \text{Ptak}(x)$ to

$\forall x. \text{Wróbel}(x) \Rightarrow \text{MaSkrzydła}(x)$

Reguły wnioskowania nie zależą od konkretnej wiedzy.

Jeśli $P \rightarrow Q$, oraz $Q \rightarrow R$, to $P \rightarrow R$

Tablice Logiczne

- Tablice wartości logicznych, tablice prawdy lub matryce logiczne:

zbadaj wszystkie wartości zmiennych logicznych.

Dla n zmiennych 2^n kombinacji – zadanie NP-trudne.

P	Q	$\sim P$	$\sim P \vee Q$	$P \rightarrow Q$	$\sim P \vee Q = (P \rightarrow Q)$
T	T	F	T	T	T
T	F	F	F	F	F
F	T	T	T	T	T
F	F	T	T	T	T

Co jeśli $n=100$? Trzeba sprawdzić $> 10^{30}$ możliwości.

Tablice przydają się w stosunkowo prostych przypadkach.

Tablice semantyczne

- Popularne w systemach dowodzenia twierdzeń pod nazwą Tableaux
- W korzeniu jest zaprzeczenie konkluzji dowodzonego twierdzenia.
- Utwórz graf zamieniając znane fakty logiczne na gałęzie.
- Sprawdź, czy we wszystkich gałęziach pojawią się sprzeczności, to oznacza, że formuła w korzeniu jest sprzeczna.
- Metoda stosowana dla różnych typów logiki, np. logiki modalnej.

$A \wedge B$ A B	$A \vee B$ ----- A B		
$A \rightarrow B \equiv \sim A \vee B$ $\sim(A \wedge B) \equiv \sim A \vee \sim B$ $\sim(A \vee B) \equiv \sim A \wedge \sim B$	$A \rightarrow B$ $\equiv \sim A \vee B$ ----- $\sim A$ B	$\sim(A \wedge B)$ $\equiv \sim A \vee \sim B$ ----- $\sim A$ $\sim B$	$\sim(A \vee B)$ $\equiv \sim A \wedge \sim B$ $\sim A$ $\sim B$

Rozumowanie Logiczne – Dedukcja

- Dedukcja:
kojarz fakty, stosuj reguły oderwania:
modus ponens ($p \Rightarrow q, p=T$ to $q=T$)
modus tolens ($p \Rightarrow q, q=F$ to $p=F$)
podstawienia and/or, eliminacji and/or,
używaj rezolucji, czyli spróbuj dojść do sprzeczności by pokazać tautologię.

$$\alpha \vee \beta \wedge \neg \beta \vee \gamma \Rightarrow \alpha \vee \gamma$$

Klauzula = zbiór formuł logicznych.

- Klauzule Horna to stwierdzenia logiczne, składające się z koniunkcji zdań prostych (atomowych):

$$P_1 \wedge P_2 \dots \wedge P_n \Rightarrow Q$$

- Dzięki metodzie rezolucji możliwe jest wnioskowanie w czasie wielomianowym (Robinson 1965) !

Logika Pierwszego Rzędu

Równość obiektów można zdefiniować w sensie predykatów:

$$X=Y \text{ jeśli } \forall P \text{ mamy } P(X)=P(Y).$$

Mamy nie tylko fakty (zdania), ale obiekty i ich ogólne własności.

Obiekty: ludzie, domy, kolory, liczby, filmy ... to elementy zbiorów.

Relacje, własności i funkcje to podzbiory.

Obiekty mogą być argumentami funkcji lub operatorów, zwracają inne obiekty.

Funkcja(obiekt_1) = obiekt_2 , lub Operator: obiekt_1 = obiekt_2

Literał: zdanie lub predykat, najmniejsza jednostka mająca sens, symbol lub słowo, np. cyfra, łańcuch znaków, polecenie języka, tablica.

Term w AI: wyrażenie logiczne stosujące się do obiektu.

Zdania atomowe: wyrażenia bez spójników, implikacji, zaprzeczeń, nie ma żadnych właściwych podformuł.

FOL

Logika pierwszego rzędu (FOL), znana jest też jako rachunek predykatów pierwszego rzędu, lub rachunek kwantyfikatorów:

Kwantyfikatory $\exists \forall$ mogą być stosowane tylko do obiektów, a nie do zbioru obiektów, predykatów, funkcji, relacji.

W FOL można zapisać: dla każdej liczby rzeczywistej istnieje od niej większa

$$\forall x \in \mathbb{R} \exists y \in \mathbb{R}, x < y,$$

Albo: nie ma czegoś takiego, co interesuje wszystkich

$$\neg(\exists x)(S(x) \wedge (\forall y)(I(y) \implies C(y, x)))$$

Ale nie można zapisać:

- **każdy zbiór** liczb rzeczywistych ma kres górny (tu potrzebny jest rachunek predykatów co najmniej drugiego rzędu).
- dla każdej funkcji z X na Y... (gdyż funkcja jest podzbiorem $X \times Y$), FOL nie uwzględnia kategorii, czasu, zdarzeń.
Do tego potrzebujemy inne logik: temporalne, modalne, deontyczne.

Własności Logiki Pierwszego Rzędu

Logika pierwszego rzędu pozwala na formalizację większości matematyki.

- Nie można w niej dowieść fałszywego twierdzenia.
- Wszystkie prawdziwe twierdzenia mają dowód.
- Logika pierwszego rzędu ma ograniczone możliwości ekspresji.
- Równość predykatów: jeśli dla wszystkich x mamy $f(x)=g(x)$ to $f=g$.

Zdefiniujmy 3 predykaty:

$W(x) = T$ jeśli x jest elementem wiedzy;

$I(y)=T$ jeśli y jest istotą inteligentną;

$Z(y,x) = T$ jeśli y jest zainteresowany x

Co oznacza zdanie: $\neg \exists x (W(x) \wedge (\forall y) (I(y) \Rightarrow Z(y,x)))$

Logiki wyższego rzędu (HOL, Higher-Order Logic) są trudne w implementacji i nie ma dla nich efektywnych mechanizmów dowodzenia.

Przykłady stwierdzeń FOL

- Jeśli samochód należy do Karola, to jest on zielony.
 $\text{posiada}(\text{Karol}, \text{auto-1}) \rightarrow \text{kolor}(\text{auto-1}, \text{zielony})$
 $\forall_x \text{auto}(X) \wedge \text{posiada}(\text{Karol}, X) \rightarrow \text{kolor}(X, \text{zielony})$
- Mirek gra na gitarze lub na skrzypcach
 $\text{gra_na_inst}(\text{Mirek}, \text{gitara}) \vee \text{gra_na_inst}(\text{Mirek}, \text{skrzypce})$
- Niektórzy ludzie lubią żmije.
 $\exists X (\text{człowiek}(X) \wedge \text{lubi}(X, \text{żmija}))$
 $\exists X (\text{człowiek}(X) \wedge \forall Y (\text{żmija}(Y) \rightarrow \text{lubi}(X, Y)))$

Często zapis logiczny nie jest jednoznaczny.

Russel & Norvig, Roz. 9 „Inference in FOL”, dokładnie opisuje mechanizmy wnioskowania w FOL.

Wykład prof. K. Grąbczewskiego zawiera około 100 slajdów na temat logiki.

CNF, postać normalna

Szybki rzut oka na FOL.

- Literał: zdanie p lub zaprzeczenie $\neg p$

Klauzula: koniunkcja literałów.

Postać normalna (conjunctive normal form, CNF): koniunkcja klauzul.

1. Eliminowany jest operator \Rightarrow stosując równoważność $p \Rightarrow q$ oraz $\neg p \vee q$;
2. Rozwijane są wyrażenia z negacją
np. $\neg (p \vee q)$ przechodzi w $\neg p \wedge \neg q$
3. Zmienne opisywane przez różne kwantyfikatory otrzymują różne nazwy.
4. Wszystkie kwantyfikatory przesuwają się na początek zachowując porządek.

CNF, postać normalna cd.

5. **Skolemizacja**: pozwala na usunięcie wszystkich kwantyfikatorów $\exists y$, zastępując je funkcjami, np.
 $\forall x: Osoba(x) \rightarrow \exists y Serce(y) \wedge Ma(x,y)$
 $\forall x: Osoba(x) \rightarrow Serce(F(x)) \wedge Ma(x, F(x))$
6. Pozostają tylko uniwersalne kwantyfikatory \forall ; można je opuścić.
7. Przekształcić na koniunkcję wyrażeń zawierających tylko alternatywy.
8. Każdy człon koniunkcji stanowi osobną klauzulę, wszystkie muszą być spełnione.
9. Zmienne każdej z klauzuli otrzymują swoje odrębne nazwy.

Wynik: postać **CNF** wyrażenia logicznego,
czyli **zbiór klauzul, które są alternatywą literałów.**

Procedura unifikacji

Unifikacja: algorytm sprawdzania równoważności stwierdzeń rachunku zdań, próba sprowadzenia ich do tej samej formy przez podstawienie unifikującej wartości, np.

$$\text{Unif}(\text{gra}(x, \text{flet}), \text{gra}(\text{Jan}, y)) = \{x/\text{Jan}, y/\text{flet}\}$$

$$\text{Unif}(\text{Lubi}(x, \text{Syn}(\text{Szef}(\text{Piotr}))), \text{Lubi}(\text{Piotr}, \text{Syn}(y))) = \{x/\text{Piotr}; y/\text{Szef}(\text{Piotr})\}$$

- Kiedy dwa stwierdzenia są równoważne?
- Zmienne w predykatkach komplikują zadanie

$$\text{gra_na_inst}(X, Y) \rightarrow \text{gra_na_inst}(\text{Mirek}, \text{skrzypce})$$

po podstawieniu $X=\text{Mirek}, Y=\text{skrzypce}$

- Czy zdania $p(X, X)$ i $p(Y, Z)$ są jednakowe?

Podstaw X/Y – czyli X za Y ; Podstaw X/Z – czyli X za Z

Czasami konieczne jest podstawienie funkcji za zmienną

$$\text{człowiek}(X) \rightarrow \text{śmiertelny}(X); \text{człowiek}(\text{ojciec}(\text{Jana}))/X$$

$$\text{człowiek}(\text{ojciec}(\text{Jana})) \rightarrow \text{śmiertelny}(\text{ojciec}(\text{Jana}))$$

Metoda rezolucji

Metoda rezolucji: iteracyjny dowód przez sprzeczność;
podstawowa metoda dowodzenia stosowana w Prologu.

Rezolucja: $(P \vee Q) \wedge (\neg Q \vee R) \Rightarrow (P \vee R)$

Zamieniamy negację $\neg p$ (dowodzonego zdania) na klauzule, czyli zbiór literałów połączonych spójnikiem \wedge)

- wybieramy parę klauzul z aksjomatów i negacji zdania p
- dokonujemy rezolucji jednej pary klauzul, zamieniając ją na pojedynczą;
 - Jeśli mamy $(q \vee \neg q)$ to możemy opuścić takie wyrażenie.
 - Jeśli klauzula jest pusta to mamy sprzeczność,

np. wyrażenie $q \wedge \neg q$ daje pustą klauzulę

Chcemy znaleźć przypadek w którym nie da się utrzymać, że wszystkie klauzule są jednocześnie prawdziwe, czyli doprowadzić do sprzeczności.

Prosty przykład

$\forall x, \forall c, \forall h, \quad \text{kolor}(x, c) \wedge \text{masa}(x, h) \rightarrow \text{przedmiot}(x).$

Obiekt x jest przedmiotem jeśli ma kolor i masę.

$\text{kolor}(\text{złoto}, \text{żółty}), \text{masa}(\text{złoto}, \text{ciężkie}).$

Forma CNF:

$A_1: \forall x, \forall c, \forall h, \neg \text{kolor}(x, c) \vee \neg \text{masa}(x, h) \vee \text{przedmiot}(x)$

$A_2: \text{kolor}(\text{złoto}, \text{żółty}); A_3: \text{masa}(\text{złoto}, \text{ciężkie}).$

Dowiedź: $\text{przedmiot}(\text{złoto})$

Zakładamy $\neg \text{przedmiot}(\text{złoto})$ i szukamy sprzeczności.

Rezolucja A_1, A_2 daje $\forall x, \forall h, \neg \text{masa}(x, h) \vee \text{przedmiot}(x)$

Rezolucja A_3 , z wynikiem daje $\text{przedmiot}(\text{złoto})$

To jest sprzeczne z założeniem.

QED.

Przykład zastosowania: QA3

QA3, Question and Answer (Green 1969)

Program odpowiada na pytania, np.

Czy niebieska substancja może być siarczkiem żelaza?

Zapis faktów:

FeS jest siarczkiem, substancją ciemnoszarą, kruchą.

Predykat siarczek-żelaza(), zmienna FeS.

Własności FeS: prawdziwe jest

$\text{siarczek-żelaza(FeS)} \wedge \text{substancja(FeS)} \wedge \text{ciemnoszary(FeS)} \wedge \text{kruchy(FeS)} \wedge \dots$

Pytanie sprowadza się do dowodu zaprzeczenia:

$\neg \exists X. \text{niebieski}(X) \wedge \text{siarczek-żelaza}(X)$

Konieczny jest sprawny algorytm szukania.

Przykład zastosowania: STRIPS

STRIPS (Fikes, Hart, Nilsson, SRI 1972)

Mając robota w punkcie A i skrzynki w punktach B, C, D, zbierz skrzynki razem.

Sytuacja opisana jest następująco:

$AT(\text{Robot}, A)$

$AT(\text{Box1}, B) \wedge AT(\text{Box2}, C) \wedge AT(\text{Box3}, D)$

Polecenie (cel) brzmi

$\exists X. AT(\text{BOX1}, X) \wedge AT(\text{BOX2}, X) \wedge AT(\text{BOX3}, X)$

Dowód konstruktywny \Leftrightarrow sekwencji czynności rozwiązujących zadanie.

- Stosowany jest rachunek predykatów pierwszego rzędu oraz analiza celów i środków.

Wady i zalety rep. logicznej

FOL (First Order Logic, Filman i Weyhrauch 1976)

Sprawdza interakcyjnie dowody w zakresie logiki pierwszego rzędu.

Istnieją nieliczne programy wychodzące poza FOL.

Zalety reprezentacji logicznej:

- spójność - wszystkie fakty to stwierdzenia logiczne,
- zupełność - można wywnioskować wszystkie prawdziwe stwierdzenia dające się wyrazić w ramach FOL,
- oddzielenie części epistemologicznej od dedukcyjnej.

Wady:

- niektóre formy wiedzy trudno jest wyrazić za pomocą reprezentacji logicznej, konieczna jest dyskretyzacja na niezbyt liczne elementy;
- problemy z eksplozją kombinatoryczną;
- niewielka efektywność wnioskowania dla logik wyższego rzędu (metoda rezolucji pomaga dla FOL).

Logika i zdrowy rozsądek

Logika klasyczna dostarcza teorii rozumowania opartej na „oczywistych” założeniach, ale:

- W realnej sytuacji wiedza jest niekompletna, niepewna, ciągle uzupełniana, zmienna w czasie (logika sytuacyjna).
- Logika może być niemontoniczna: dodanie nowej wiedzy może falsyfikować istniejące konkluzje. Nowe fakty mogą być sprzeczne ze starymi.
- Stwierdzenia mogą być prawdziwe, fałszywe lub nieokreślone.
- Założenia domyślne (default logic) dostarczają wnioski z braku faktów.
- Potrzebna jest metoda rozstrzygania konfliktów jeśli wnioski są sprzeczne.

Jest wiele innych logik, np. logiki temporalne i dynamiczne (zmiennosc obiektów w czasie), modalne (rozdzielajace mozliwosc/koniecznosc), logiki epistemiczne, logiki wielomodalne, logiki deontyczne (wymagane, zezwolone), logiki programow, oraz logiki przyblizone i rozmyte, o ktorych warto wiedziec wiecej ze wzgledu na szerokie zastosowanie w analizie i modelowaniu danych.

