Pravdivost, dokazatelnost. Důkazové metody a systémy.

Luboš Popelínský

E-mail: popel@fi.muni.cz
http://nlp.fi.muni.cz/uui/

Obsah:

- Pravdivost v interpretaci a logická pravdivost
- Axiomatické systémy pro výrokovou logiku
- Formální systémy
- Normální formy
- DPLL

Logický agent

```
function KB-AGENT(percept)  # vrací akci
# globální: KB – báze znalostí; t – číslo, na začátku 0
    tell (KB, make_percept_sentence(percept, t))
    action ← ask(KB, make_action_query(t))
    tell (KB, make_action_sentence(action, t))
    t ← t+1
    return action
```

Pravdivost v interpretaci

Formule A je pravdivá v interpretaci I (Interpretace I splňuje formuli A), jestliže po substituci *za výrokové symboly* vrací hodnotu TRUE

Dokážeme dosazením z I do A a vyhodnocením (valuací) logických spojek.

viz Animace

Silnější vlastnost: pravdivost ve všech interpretacích

Logická pravdivost I

Formule je pravdivá (logicky pravdivá), jestliže je pravdivá ve všech interpretacích.

pravdivá formule == tautologie, anglicky též a valid formula

Důkaz (logické) pravdivosti formule: model checking; ukážeme, že formule je pravdivá ve všech interpretacích.

Vytvoříme pravdivostní tabulku a pro každou interpretaci ověříme, že formule je v této interpretaci pravdivá.

Složitost n výrokových symbolů, tj. 2^n intrepretací

Existuje efektivnější způsob?

Logická pravdivost II

Příklad: $p \Rightarrow (q \Rightarrow p)$

Příklad: $p \vee (\neg p \wedge r)$

Příklad: $p \vee (\neg p \wedge r) \vee \neg p$

Příklad: $(p \vee q) \wedge (\neg p \vee \neg r)$

např. znegování formule a důkaz nesplnitelnosti? $==$ důkaz sporem

Formule s implikací a důkaz sporem

$P \Rightarrow Q$

- ověření tautologií tvaru implikace metodou protipříkladu:
  $\Rightarrow$ je nepravdivá pouze pro pravdivý předpoklad a nepravdivý důsledek. Pro tuto variantu – za předpokladu nepravdivosti důsledku – pro příslušné interpretace ověříme (ne)pravdivost předpokladu.

- příklad: $p \Rightarrow (q \Rightarrow p)$
  - předpoklad: p pravdivá, $q \Rightarrow p$ nepravdivá
  - jediná možnost nepravdivosti $q \Rightarrow p$:
    q pravdivá, p nepravdivé
  - spor s předpokladem pravdivosti p

$==$ důkaz sporem, proof by refutation or proof by contradiction.

$\alpha \models \beta$ právě když je formule $\alpha \land \neg\beta$ nesplnitelná.

Axiomatický systém

- jazyk: stejný jako jazyk výrokové logiky; primárně používáme systém spojek $\{\Rightarrow, \neg\}$, ostatní spojky jsou chápány jako zkrácené zápisy: $A \wedge B =_{df} \neg(A \Rightarrow \neg B), A \vee B =_{df} \neg A \Rightarrow B$, $A \Leftrightarrow B =_{df} (A \Rightarrow B) \wedge (B \Rightarrow A)$
- axiomy (resp. schémata axiomů; A, B, C jsou formule):
  A$_1$ $A \Rightarrow (B \Rightarrow A)$
  A$_2$ $(A \Rightarrow (B \Rightarrow C)) \Rightarrow ((A \Rightarrow B) \Rightarrow (A \Rightarrow C))$
  A$_3$ $(\neg B \Rightarrow \neg A) \Rightarrow (A \Rightarrow B)$
- odvozovací (inferenční) pravidlo modus ponens (MP) (pravidlo odloučení): jsou-li z axiomů dokazatelné (odvoditelné) formule A a $A \Rightarrow B$, pak je dokazatelná i B. Zapisujeme též

$$\frac{A \qquad A \Rightarrow B}{B}$$

Příklad

- důkaz A: konečná posloupnost formulí, jejíž každý člen je axiom nebo důsledek MP, jehož předpoklady jsou mezi předchozími členy, a poslední člen je formule A. Je-li A dokazatelná, píšeme $\vdash A$.
- příklad: dokažte $\vdash A \Rightarrow A$ (vpravo jsou komentáře k jednotlivým krokům)

1. $\vdash A \Rightarrow ((A \Rightarrow A) \Rightarrow A)$	$\mathbf{A_1}$
2. $\vdash (A \Rightarrow ((A \Rightarrow A) \Rightarrow A)) \Rightarrow ((A \Rightarrow (A \Rightarrow A)) \Rightarrow (A \Rightarrow A))$	$\mathbf{A_2}$
3. $\vdash (A \Rightarrow (A \Rightarrow A)) \Rightarrow (A \Rightarrow A)$	MP(1,2)
4. $\vdash A \Rightarrow (A \Rightarrow A)$	$\mathbf{A_1}$
5. $\vdash A \Rightarrow A$	MP(3,4)

Vlastnosti uvedného axiomatického systému

- Věta (korektnost a úplnost): A je dokazatelná právě tehdy, když je pravdivá, tj. $\vdash A \Leftrightarrow \models A$
  Důkaz: $\Rightarrow$ (korektnost): ověříme, že axiomy jsou tautologie a jsou-li předp. MP tautologie, pak i důsledek je tautologie (tabulky, věta o implikaci)
  $\Leftarrow$ (úplnost): složitější, na základě pomocných tvrzení (lemma o neutrální formuli a lemma o odvození z atomických komponent)
  Pozn.: věta vystihuje vztah mezi syntaxí a sémantikou výr. logiky
- rozhodnutelnost: neexistuje systematická procedura (jde spíše o "hádání"jednotlivých kroků důkazu), nevhodné pro strojové zpracování. Dokazování lze zjednodušit pomocí dokazatelnosti z předpokladů a syntaktické věty o dedukci ($T, A \vdash B \Leftrightarrow T \vdash A \Rightarrow B$).
- axiomy jsou nezávislé (žádný nelze odvodit ze zbývajících dvou)

Formální systémy

Axiomatický systém je příkladem deduktivního systému (nebo též formálního systému).

Formální systém

- postavený výhradně na syntaktické bázi: jazyk logiky neinterpretujeme, provádíme s ním pouze syntaktické manipulace – důkazy
- cíl: získat formální teorii jako souhrn dokazatelných formulí – teorémů
- formální systém tvoří
  - jazyk + formule (symbolický jazyk výrokové logiky) – bez interpretací
  - odvozovací pravidla – operace na formulích umožňující konstrukce důkazů
  - případně axiomy – výchozí tvrzení přijímaná bez důkazu; (axiomy spolu s odvozovacími pravidly tvoří dedukční systém)
- formální systémy lze rozdělit na
  - axiomatické (méně pravidel)
  - předpokladové (méně axiomů)

Požadované vlastnosti formálních systémů

- korektnost (bezespornost): je dána výběrem axiomů a odvozovacích pravidel; systém je korektní, nelze-li v něm zároveň odvodit tvrzení i jeho negaci. Ve sporném systému lze odvodit cokoliv. Vyžadována vždy. (Sémantická korektnost: existuje alespoň jeden model.)
- úplnost: připojením neodvoditelné věty k úplnému systému se tento stane sporným. Nevyžadována vždy – úplné jsou pouze velmi jednoduché teorie. (Sémantická úplnost: každé tvrzení pravdivé ve std. interpretaci lze odvodit.)
- rozhodnutelnost: existence algoritmu pro ověření dokazatelnosti libovolné formule. V axiom. systémech podmíněna úplností; zpravidla splněna pouze pro určité třídy formulí.
- nezávislost axiomů: nezávislý axiom nelze odvodit z ostatních axiomů; závislý axiom může být vypuštěn z dané soustavy axiomů

Další axiomatické systémy

Для исчисления высказываний могут быть построены аксиоматизации и с одной единственной схемой аксиом. Так, например, если за примитивные связки принять $\neg$ и $\supset$, то при единственном правиле вывода — modus ponens — достаточной оказывается схема аксиом:

$$[(((\mathscr{A} \supset \mathscr{B}) \supset (\neg\mathscr{C} \supset \neg\mathscr{D})) \supset \mathscr{C}) \supset \mathscr{E}] \supset [(\mathscr{E} \supset \mathscr{A}) \supset (\mathscr{D} \supset \mathscr{A})]$$

(Мередит [1953]).

Другим примером такого рода может служить система Никода [1917], в которой употребляется единственная связка | (дизъюнкция отрицаний), имеется единственное правило вывода, по которому $\mathscr{C}$ следует из $\mathscr{A}$ и $\mathscr{A} \mid (\mathscr{B} \mid \mathscr{C})$, и единственная схема аксиом

$$(\mathscr{A} \mid (\mathscr{B} \mid \mathscr{C})) \mid \{[\mathscr{D} \mid (\mathscr{D} \mid \mathscr{D})] \mid [(\mathscr{E} \mid \mathscr{B}) \mid ((\mathscr{A} \mid \mathscr{E}) \mid (\mathscr{A} \mid \mathscr{E}))]\}.$$

Дальнейшие сведения из этой области, в том числе и исторический обзор, можно найти в книге Чёрча [1956].

Introduction to Mathematical Logic by Elliott Mendelson, 1964

Gentzenovský systém (kalkul sekventů)

- příklad pravidlového systému formální logiky: pouze nástin, nikoliv úplná definice
- další typ výrazů formálního jazyka: sekventy (sekvence)
  $A_1, A_2, \ldots, A_n \vdash B$,
  kde A_i, B jsou formule, $\vdash$ je symbol odvoditelnosti (dokazatelnosti). Posloupnost na levé straně chápeme jako konečnou množinu formulí (budeme ozn. Γ) – nezáleží na pořadí, lze vynechat duplicity, může být prázdná.
- jediný axiom: $\Gamma, A \vdash A$

Gentzenovský systém: pravidla

- obecné schéma pravidel: $\dfrac{\text{předpoklad}_1 \qquad \dots \qquad \text{předpoklad}_n}{\text{závěr}}$
- pravidla zavedení a eliminace předpokladu:

$$\frac{\Gamma \vdash A}{\Gamma, A \vdash A} \qquad \frac{\Gamma, A \vdash B \qquad \Gamma, \neg A \vdash B}{\Gamma \vdash B}$$

- řada pravidel pro spojky (uvedeme pouze některá na ukázku)
  zavedení a eliminace $\vee$:

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \qquad \frac{\Gamma, A \vdash C \qquad \Gamma, B \vdash C \qquad \Gamma \vdash A \vee B}{\Gamma \vdash C}$$

  zavedení a eliminace $\wedge$:

$$\frac{\Gamma \vdash A \qquad \Gamma \vdash B}{\Gamma \vdash A \wedge B} \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A} \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B}$$

Gentzenovský systém: důkazy

- důkaz sekventu: strom, v jehož kořeni je dokazovaný sekvent, v listech axiomy a každý uzel (závěr) se svými přímými následníky (předpoklady) představuje instanci některého z pravidel systému
- je-li dokázaný sekvent tvaru $\vdash A$, pak formuli A nazýváme teorém (odvoditelná resp. dokazatelná formule)
- systém je korektní a úplný (platí $\vdash A \Leftrightarrow \models A$)
- příklad: důkazu sekventu $\vdash p \vee \neg p$:

$$\dfrac{\dfrac{\neg p \vdash \neg p}{\neg p \vdash p \vee \neg p} \qquad \dfrac{p \vdash p}{p \vdash p \vee \neg p}}{\vdash p \vee \neg p}$$

(použito pravidlo eliminace předpokladu a dvakrát pravidlo zavedení $\vee$)

Logický agent

function KB-AGENT(*percept*) *# vrací akci*
 # globální: KB – báze znalostí; t – číslo, na začátku 0
    tell (*KB*, **make_percept_sentence**(*percept*, *t*))
    action ← **ask**(*KB*, **make_action_query**(*t*))
    tell (*KB*, **make_action_sentence**(*action*, *t*))
    $t \leftarrow t{+}1$
    return *action*

Jak se liší uvedené důkazové metody od znalostního agenta a v čem se podobají ?

Logická pravdivost III

Příklad 1: $p \Rightarrow (p \vee r)$

Příklad 2: $p \vee (\neg p \wedge r)$

Příklad 3: $p \vee (\neg p \wedge r) \vee \neg p$

Příklad 4: $(p \vee q) \wedge (\neg p \vee \neg r)$

všechny formule 2. – 4. jsou v normální formě.

Normální formy

- Věta o reprezentaci: každou n-ární pravdivostní funkci lze reprezentovat formulí výrokové logiky $A(p_1, p_2, \ldots, p_m)$ obsahující pouze spojky $\neg, \wedge$ a $\vee$, kde $m \leq n$.
- Jak zkonstruovat k funkci tuto formuli (základ důkazu věty): nechť je funkce reprezentována standardní tabulkou. Jsou-li všechny funkční hodnoty rovny 0, je reprezentující formulí libovolná kontradikce, například $p_1 \wedge \neg p_1$. Jinak pro každý řádek, v němž je funkční hodnota rovna 1, vytvoříme konjunkci $K_i = {}^i p_1 \wedge {}^i p_2 \wedge \cdots \wedge {}^i p_n$, kde pro $j = 1, 2, \ldots, n$

$$
{}^i p_j = \begin{cases} p_j & \text{je-li v } i\text{-tém řádku hodnota } j\text{-tého argumentu } 1 \\ \neg p_j & \text{je-li v } i\text{-tém řádku hodnota } j\text{-tého argumentu } 0 \end{cases}
$$

Disjunkce D všech konjunkcí K_i reprezentuje danou pravdivostní funkci.

Příklad

Příklad: určete formuli reprezentující následující pravdivostní funkci:

x	y	$f(x,y)$	K_i	D_i
1	1	1	$K_1 = p \wedge q$	
1	0	0		$D_2 = \neg p \vee q$
0	1	1	$K_3 = \neg p \wedge q$	
0	0	1	$K_4 = \neg p \wedge \neg q$	

- atomické formule a jejich negace == literály. Elementární konjunkcí nad $p_1, p_2, \ldots, p_n$ nazveme každou konjunkci, v níž se každý z těchto symbolů vyskytuje jako literál právě jednou. Úplnou normální disjunktivní formou (úndf) nad týmiž symboly nazveme každou disjunkci vesměs různých elementárních konjunkcí.
- podobně úplná normální konjunktivní forma (únkf) bude konjunkcí všech disjunkcí v sloupci D_i
- $(p \wedge q) \vee (\neg p \wedge q) \vee (\neg p \wedge \neg q)$ je v úplné normální disjunktivní formě
- $(\neg p \vee q)$ je v úplné normální konjunktivní formě

DPLL (Davis-Putnam-Logemann-Loveland Procedure)

základ většiny prakticky používaných důkazových nástrojů

pracuje s formulí v konjunktivní normální formě

Pravidla

UNSAT	If F contains $\Box$, then F is unsatisfiable.
SAT	If F is empty set $\{\}$, then F is satisfiable.
MULT	If a literal occurs more than once in a clause, then all but one can be deleted.
SUBS	A clause in F can be deleted, if it is a superset of another clause in F.
UNIT	An element $\neg$ L of a clause in F can be deleted,

if F contains $\{L\}$.

Pravidla

TAUT A clause can be deleted, if it contains a literal and its complement.

PURE A clause can be deleted, if it contains L and $\neg$ L does not occur in F.

SPLIT

If F is semantically equivalent to a formula of the form

$\{\ \{C_1 \vee L\}, ..., \{C_k \vee L\}, \{C_{k+1} \vee \neg L\}, ..., \{C_m \vee \neg L\},$
$C_{m+1}, ..., C_n\}$

where neither L nor $\neg L$ occur in C_i, 1 $\leq$ i $\leq$ n,

then replace F by the CNF of

$\{C_1, ..., C_k, C_{m+1}, ..., C_n\} \vee \{C_{k+1}, ..., C_m, C_{m+1}, ..., C_n\}$.

Pravidla 2

UNSAT $\{ \square, C_1, ..., C_n \} \equiv \{\square\}$
MULT $\{L, L, L1, \ldots, Lm\} \equiv \{L, L1, \ldots, Lm\}.$
SUBS $\{ \{L1, \ldots, Lm\}, \{L1, \ldots, Lm, \ldots Lk\}, C1, \ldots, Cn \}$
$\equiv \{ \{L1, \ldots, Lm\}, C1, \ldots, Cn\}.$
UNIT $\{\{L1, \ldots, Lm, L\}, \{\neg L \}\} \equiv \{L1, \ldots, Lm\}, \{\neg L \} \}$
TAUT $\{ \{ L1, \ldots, Lm, L, \neg L\}, C1, \ldots, Cm \} \equiv \{C1, \ldots, Cm \}.$

Pravidla 3

PURE $\{\{p, \neg q\}, \{\neg r\}\} \not\equiv \{\{p, \neg q\}\}$,
but $\{C1, \ldots, Cm, \{L, L1, \ldots, Ln\}\}$ is unsatisfiable
iff $\{C1, \ldots, Cm\}$ is unsatisfiable, where $\neg$ L
does neither occur in Ci, $1 \leq i \leq$ nor in Lj, $1 \leq j \leq n$.

SPLIT $\{\{p, r\}, \{\neg r\}, \{q\}\} \not\equiv \{\{p\}, \{q\}\} \vee \{\square, \{q\}\}$,
but the rule preserves unsatisfiability.

Vlastnosti pravidel

- Let F be a formula in CNF and F' be obtained from F by applying a rule.
- MULT, SUBS, UNIT and TAUT are equivalence preserving, i.e., $F' \equiv F$.
- They are polynomial simplification rules.
- PURE and SPLIT are unsatisfiability preserving,i.e.,
  F is unsatisfiable iff F' is unsatisfiable.

DPLL algoritmus

1. Input F. Transform F into CNF.
2. Apply the rules MULT, SUBS, UNIT, TAUT, PURE and SPLIT until SAT or UNSAT become applicable.
3. If SAT is applicable then terminate with **F is satisfiable**.
4. If UNSAT is applicable then terminate with **F is unsatisfiable**.

Remarks

- The algorithm always terminates.
- Whenever the application of a rule in step 3 yields a formula H, then H is unsatisfiable iff F is unsatisfiable.
- The algorithm is sound and complete.

DPLL: An Example

$\{\ \{p1,\ p2\},\ \{p4,\ \neg p2,\ \neg p3\},\ \{\ \neg\ p1,\ p3\},\ \{\neg p4\}\ \}$

Initialization

$\{\ \{p1,\ p2\},\ \{\neg p2,\ \neg p3\},\ \{\ \neg\ p1,\ p3\},\ \{\neg p4\ \}\}$

UNIT wrt $\{\neg p4\}$

$\{\ \{p1,\ p2\},\ \{\neg p2,\ \neg p3\},\ \{\ \neg\ p1,\ p3\}\ \}$

PURE wrt $\neg p4$

$\{\ \{p2\},\ \{\neg p2,\ \neg p3\}\}\ \vee\ \{\{p3\},\ \{\neg p2,\ \neg p3\ \}\ \}$

SPLIT wrt p1.

Příklad

$\{ \{p2\}, \{ \neg p3\}\} \vee \{\{p3\}, \{\neg p2 \} \}$

UNIT wrt $\neg$p2 in 1st disjunct and UNIT wrt $\neg$p3 in 2nd one.

$\{ \{ \neg p3\}\} \vee \{ \{\neg p2 \} \}$

PURE wrt p2 in 1st disjunct and PURE wrt p3 in 2nd one .

$\{ \} \vee \{ \}$

PURE wrt $\{ \neg\ p3\}$ in 1st dis. and PURE wrt $\{\neg p2 \}$ in 2nd.

SAT (satisfiable for both branches)

Shrnutí

Víme,

- co je logická pravdivost
- co jsou axiomatické systémy
- co jsou formální důkazové systémy
- jak se liší od logického agenta
- co jsou normální formy
- DPLL