Kognitívna mapa bludiska

Michal Malý

KAI FMFI UK

24. február 2011

Prehľad

1. Učenie posilňovaním
   - Porovnanie s ostatnými metódami
   - Definícia
   - Príklady
   - Metódy
   - Vybraný podproblém: tvorba modelu
   - Predošlý výskum
   - Môj prístup
2. Gramatická indukcia / Inferencia automatu
   - Formálne jazyky – rýchly prehľad
   - Definícia
3. Riešenie
   - Metóda
   - SAT solver
   - Výsledok
   - Ďalší výskum

Módy učenia

na rozdiel od:

- s učiteľom [supervised] – „trénovanie” – ukazujeme pár (vstup, správny výstup)
- samo-organizácia [unsupervised] – vznik štruktúry napr. na základe štatistických vlastností

rieši **učenie posilňovaním** úlohy, kde:

- správny vstup nie je k dispozícii
- niekedy sa nedá povedať ani, či výstup akcia bola správna alebo nie

Všeobecná definícia

RL je učenie čo robiť – ako namapovať akcie k situáciám – aby sa maximalizoval vstupný signál. Učenému nie je povedané, aké akcie má vykonať, ale miesto toho musí skúšaním zistiť, ktoré akcie prinášajú najväčšiu odmenu. V najzaujímavejších a najnáročnejších prípadoch akcia nemusí ovplyvniť len najbližšiu odmenu, ale aj časovo nasledujúcu situáciu a cez ňu všetky nasledujúce odmeny. Tieto dve charakteristiky – učenie cez pokus a omyl, a oneskorená odmena, sú dve najdôležitejšie charakteristiky RL. – R. S. Sutton and A. G. Barto.: Reinforcement Learning: An Introduction (aj online `http://www.cs.ualberta.ca/~sutton/book/the-book.html`)

Príklady

- obrátené kyvadlo [inverted pendulum]
- balansovanie metly :) [pole balancing]
- labyrint [maze]

Ako to urobiť?

Hocijako, ale používa sa napr.:

- dynamické programovanie
- Monte Carlo
- Temporal Difference Learning
- Q-Learning
- SARSA

Charakteristika

- policy
- reward function
- value function
- niekedy: model of environment

Môj vybraný podproblém – motivácia

- mnohé metódy nepoužívajú model sveta (stotožňujú možné stavy prostredia s perceptuálnym priestorom)
- ak používajú, potrebujú ho mať daný
- čo keď model nemáme / nechceme dať?

Príklad: Bludisko

- nepoznáme cesty
- nechceme preddefinovať agenta na konkrétne bludisko
- podobnosť so všeobecným problémom modelovania sveta: križovatky sú situácie, smery sú rozhodnutia

Predošlý výskum

- perceptuálny aliasing (Whitehead, Ballard, 1992) $\rightarrow$ algoritmus Lion (detekcia nekonzitentnosti a reset hodnoty akcie na nulu)
- *„Uvažujme úlohu balenia darčeka, ktorá zahŕňa 4 kroky: otvoriť krabicu, vložiť darček, zatvoriť ju, a zalepiť. Agent, ktorý je vedený len jeho aktuálnym vizuálnym vnemom nedokáže úlohu splniť, pretože ak má pred sebou zatvorenú krabicu, nevie, či darček je už vo vnútri, a teda sa nevie rozhodnúť, či má krabicu zalepiť alebo otvoriť."* (Lin, Mitchell, 1992)
- agregácia stavov (Singh, Jaakkola, 1995)
- delenie stavov (McCallum, 1993)

Môj prístup

- minimálny model (princíp minimálnej popisnej dĺžky (MDL) – Occamova britva)
- gramatická indukcia (inferencia automatu)

Definícia (Abeceda)

Abeceda je konečná neprázdna množina znakov (symbolov, tokenov).

Definícia (Slovo)

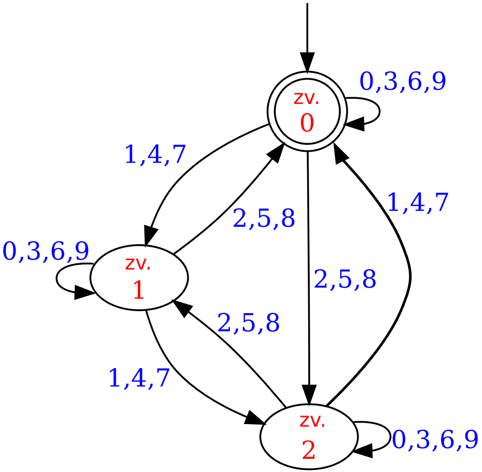
Slovo je konečná postupnosť symbolov z abecedy.

Definícia (Jazyk)

Formálny jazyk je množina slov.

Príklad:
abeceda: $\Sigma = \{a, b\}$
slová: *abba*,*aaaabbbb*
jazyk: $L = \{\varepsilon, ab, ba, aabb, abba, baba, bbaa, ...\}$ = tie slová, kde počet *a*-čok a *b*-čok v slove je rovnaký

Konečný automat



Automat na deliteľnosť troma.

Napr. „2047": prejde stavmi: zv.0, zv.2, zv.2, zv.0, zv.1

Gramatika

Definícia (Terminálny symbol)

Terminály – symboly, z ktorých sa skladajú slová (na výstupe).

Definícia (neterminálny symbol)

Neterminály – pomocné symboly pri odvodzovaní, nesmú sa objaviť na výstupe.

Definícia (Gramatika)

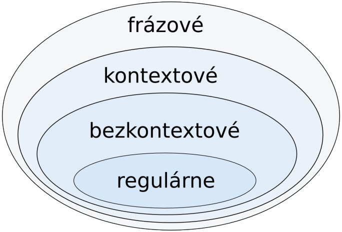
Gramatika je určená množinou neterminálov, množinou terminálov, množinou odvodzovacích pravidiel a štartovacím symbolom.

Gramatika – príklad

terminály: $T = \{1, 2, 3, \ldots 9, +, -, (,)\}$
neterminály: $N = \{V, C\}$
štartovací symbol: V
pravidlá:
$V \rightarrow C$
$V \rightarrow V + V$
$V \rightarrow V - V$
$V \rightarrow (V)$
$C \rightarrow 1,\ C \rightarrow 2,\ \ldots,\ C \rightarrow 9$

Vieme vyrobiť slovo $1 - (4 + 5)$ napríklad takto:
$V \Rightarrow V - V \Rightarrow C - V \Rightarrow C - (V) \Rightarrow C - (V + V) \Rightarrow C - (C + V) \Rightarrow$
$C - (C + C) \Rightarrow 1 - (C + C) \Rightarrow 1 - (4 + C) \Rightarrow 1 - (4 + 5)$

Chomského hierarchia



1. regulárne gramatiky ≡ konečný automat (nevedia $a^n b^n$)
2. bezkontextové gramatiky ≡ nedeterministický zásobníkový automat (nevedia $a^n b^n c^n$)
3. kontextové gramatiky ≡ nedeterministický lineárny automat (nevie EXPSPACE-hard problémy)
4. frázové gramatiky (neobmedzené, typu 0) ≡ Turingov stroj (vie všetko, čo je „intuitívne vypočítateľné”)

Gramatická indukcia / Inferencia automatu

Definícia

Gramatická indukcia je spôsob, ako odvodiť formálnu gramatiku z množiny vzoriek – pozorovaní.

Definícia

Inferencia automatu je spôsob, ako odvodiť automat z množiny vzoriek – pozorovaní.

Príklad: daná množina $\{27264, 4491, 23022, 18066, 3822, 7758, 14178, \ldots\}$.
Čo to je?

Použitá metóda: inferencia minimálneho konečného automatu/modelu

- konečný počet stavov, prechody, pozorovanie v stave
- spĺňa doterajšie pozorovania (skúsenosť)
- minimálny počet stavov

Metóda: SAT solver

- logický výraz s množstvom neznámych premenných
  $(a \vee b \vee \neg c) \wedge (\neg a \vee b \vee c)$
- je splniteľná? (je možné také dosadenie hodnôt premenným, aby logická hodnota výrazu *true*)
- aké dosadenie hodnôt ju spĺňa?

Použitie SAT solvera na nájdenie min. automatu

- automat zadefinujeme cez logické predikáty
- zapíšeme obmedzenia dané pozorovaniami ako sériu logických výrazov
- zapíšeme všeobecné obmedzenia
- prevedieme predikáty na premenné a do formátu súboru pre SAT solver

Formalizácia

- v stave s dostáva agent pozorovanie $o \quad = \quad obs(o, s)$
- na akciu a sa zo stavu s dostaneme do $s' \quad = \quad tr(s, a, s')$
- v čase t sme boli v stave $s \quad = \quad pos(t, s)$

Obmedzenia:

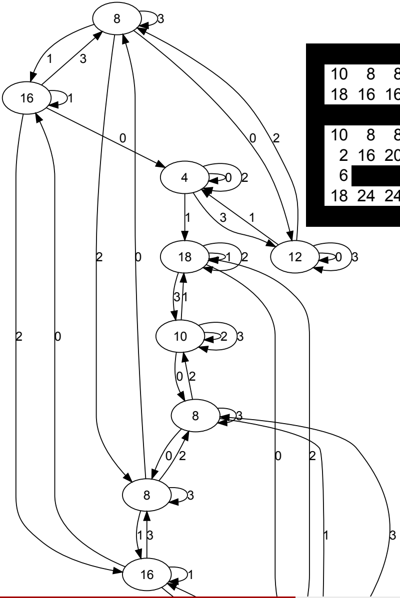
- v jednom stave máme najviac jedno pozorovanie:
  $\forall s \in \mathcal{S} : \forall o, o' \in \mathcal{O}, o \neq o' : \neg obs(o, s) \vee \neg obs(o, s')$
- a aspoň jedno: $\forall s \in \mathcal{S} : \bigvee_{o \in \mathcal{O}} obs(o, s)$
- Podobne pre predikát tr:
  $\forall s, s', s'' \in \mathcal{S}, a \in \mathcal{A}, s' \neq s'' : \neg tr(s, a, s') \vee \neg tr(s, a, s'')$ a tiež
  $\forall s \in \mathcal{S}, a \in \mathcal{A} : \bigvee_{s' \in \mathcal{S}} tr(s, a, s')$
- a pre predikát pos: $\forall 0 \leq t \leq T, s, s' \in \mathcal{S} : \neg pos(t, s) \vee \neg pos(t, s')$a
  $\forall 0 \leq t \leq T : \bigvee_{s \in \mathcal{S}} pos(t, s)$

Formalizácia (pokr.)

Obmedzenie, že model vyhovuje pozorovaniam:

- $\forall 0 \leq t \leq T, s \in \mathcal{S} : obs(s, o_t) \vee \neg pos(t, s)$
- $\forall 0 \leq t \leq T - 1, s, s' \in \mathcal{S} : tr(s, a_t, s') \vee \neg pos(t, s) \vee \neg pos(t + 1, s')$

Výsledok



10	8	8	8	12		14
18	16	16	16	4		6
				6		6
10	8	8	24	4		6
2	16	20		22		6
6						6
18	24	24	24	24	24	20

Možné rozšírenia

- silnejší formalizmus
- online budovanie modelu (nie nutne minimálne)
- štrukturované pozorovanie (-tica)

Koniec

Otázky?