

Tentamen Representatie & Zoeken

9 April 1996

Antwoorden

N.B: AI studenten maken vraag 1 t/m/ 5. De overige studenten maken slechts vraag 1 t/m 4.

Beoordeling: Alle vragen 1 t/m 5 tellen voor 20 punten. De eindscore is het gemiddelde van de te maken vragen.

Vraag 1: Zoeken

Vraag 1a: a,d,c,b,j,i,h,g,f,e,p,o,n.

Vraag 1b: a,d,j,p,i,o,v,u,i,c,h,n.

Vraag 1c: a, a,d,c,b, a,d,j,i,c,h,g,b,f,e a,d,j,p,i,o,c,h,n.

Vraag 1d: a,d,c,b,j,i,p,h,g,n.

Vraag 1e: Het algorithm is een A* algorithm. Het heet een A algorithme omdat de evaluatie functie de som is van de reeds afgelegde afstand plus de geschatte afstand ($f(n) = g(n) + h(n)$). Om A* te heten moet de geschatte afstand tot een doel-knoop overal kleiner-of-gelijk zijn aan de werkelijke minimale afstand tot een doel-knoop. Deze eigenschap blijkt op te gaan voor alle knopen in de boom. Het algorithm mag dus een A* algorithme genoemd worden.

Vraag 2

Vraag 2a

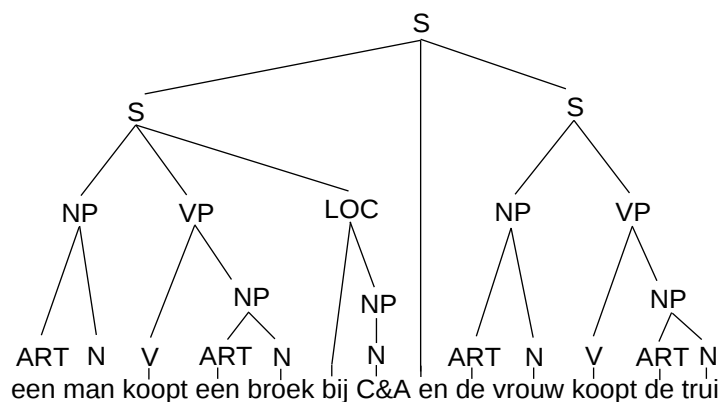


Figure 1:

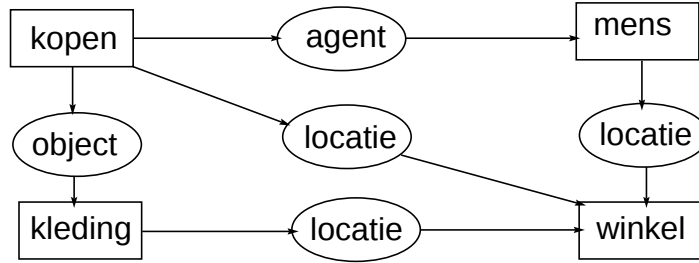


Figure 2:

Vraag 2b

Vraag 2c

Bijvoorbeeld: “Een broek koopt een C&A bij vrouw”. Deze zin volgt wel de gegeven grammatica, maar past niet in het case-frame van opgave 2b.

Vraag 3

S is de meest specifieke beschrijving van de wensen van de klant, en G is de meest algemene beschrijving. Het algoritme gaat nu door de volgende stappen:

- We beginnen met S zo specifiek mogelijk, en G zo algemeen mogelijk:
 $G = p(X, Y, Z)$, $S = \emptyset$.
- Na het eerste positieve voorbeeld moet S wat gegeneraliseerd worden:
 $S = \{p(man, lezen, lang)\}$.
- Na het negatieve voorbeeld moet G specifieker worden:
 $G = \{p(man, Y, Z), p(X, sport, Z), p(X, lezen, Z), p(X, Y, lang)\}$.
- Na het tweede positieve voorbeeld moet S nogmaals gegeneraliseerd worden:
 $S = \{p(man, Y, Z)\}$.
- Als laatste kunnen we nu elementen uit G verwijderen die niet met S in overeenstemming zijn. Dit leidt tot: $G = p(man, Y, Z)$.

Met andere woorden, de klant is geïnteresseerd in een man, en zijn hobbies en lengte doen er niet zoveel toe.

Vraag 4

Vraag 4a

De volledige invoer/uitvoer-tabel voor dit netwerk is:

invoer			uitvoer
I1	I2	I3	O1
0	0	0	1
1	0	0	0
0	1	0	0
0	0	1	0
1	1	0	0
1	0	1	0
0	1	1	0
1	1	1	1

Hieruit blijkt dat de functie van het netwerk is om te beslissen of op de invoer alleen 1-en of alleen 0-en staan. In die gevallen is de uitvoer 1. In alle andere gevallen (de invoer is een mengsel van 1-en en 0-en), is de uitvoer 0.

H1 gaat aan precies wanneer de invoer alleen uit 1-en bestaat.

H2 gaat aan precies wanneer de invoer minstens één 1 bevat.

Vraag 4b

Het gegeven netwerk is *geen* perceptron, omdat de invoer-neuronen niet direct aan de uitvoer-neuronen zijn verbonden. Het netwerk bevat een zogenaamde “hidden layer” tussen de invoer- en de uitvoer-laag.

Vraag 5

Als invoer neemt het EBL algoritme (i) een algemene domein-theorie, (ii) een doel-concept dat geleerd moet worden, (iii) een doel-vocabulair waarin het te leren doel-concept moet worden beschreven, en (iv) één voorbeeld van het te leren concept.

Als eerste stap bepaalt het EBL algoritme een verklaring waarom het voorbeeld inderdaad tot het doel-concept behoort. Deze verklaring moet geheel geformuleerd zijn in termen van het doel-vocabulair.

Als tweede stap neemt het EBL algoritme deze verklaring en verwijdt daaruit alle elementen die specifiek zijn voor dat ene voorbeeld. Deze stap heet generalisatie.

Het gevolg is een algemene beschrijving in termen van het doel-vocabulair waarom iets wel of niet tot het doel-concept zou behoren. Deze algemene beschrijving van het doel-concept in termen van het doel-vocabulair is de uitvoer van het EBL algoritme.