

Tentamen IN2205 Kennissystemen

4 Juli 2008, 14:00–17:00

- Dit tentamen heeft 9 meerkeuzevragen in totaal goed voor 18 punten en 7 open vragen met in totaal 53 punten.
- Wat betreft de meerkeuzevragen:
 - Er is voor iedere vraag telkens maar één goed antwoord mogelijk.
- Wat betreft de open vragen:
 - Geef antwoord in correct Nederlands of Engels en schrijf leesbaar (gebruik eerst kladpapier).
 - Geef geen irrelevante informatie. Dit kan leiden tot puntenaftrek.

Vraag:	10	11	12	13	14	15	16	Totaal:
Punten:	10	7	10	11	7	4	4	53

- Het gebruik van boek of aantekeningen tijdens dit tentamen is niet toegestaan.
- Het gebruik van een rekenmachine is toegestaan.
- Voordat je je antwoorden inlevert, controleer of op ieder blaadje je naam en studienummer staat en geef het aantal ingeleverde bladen aan op (tenminste) de eerste pagina.
- De tentamenstof bestaat uit hoofdstukken 1,2,3,4,6,7,8 en 9 uit *Artificial Intelligence* van Michael Negnevitsky, behalve secties 3.4, 3.5, 8.2, 9.1 en 9.7, het artikel *Swarm Smarts* van Eric Bonabeau *et. al.* en de lecture notes over Bayesian Reasoning en Swarm Intelligence.
- Uiteraard komen in één tentamen niet alle onderwerpen aan bod. Trek daarom op basis van dit tentamen geen conclusies over stof die nooit getoetst wordt.
- Totaal aantal pagina's: 8.

Succes!

Multiple-choicevragen

- (2 punten) In expert systemen wordt backward chaining gebruikt om ...
 - te bepalen welke consequent de grootste kans heeft.
 - alle mogelijk conclusies die uit de feiten in de database volgen af te leiden.
 - te bepalen of een bepaalde hypothese vanuit de data en regels bewezen kan worden.**
 - te bepalen welke regel als eerste mag vuren als meerdere regels tegelijk kunnen vuren.
- (2 punten) Hoe 'leert' een regelgebaseerd kennissysteem?
 - We passen de redeneeralgoritmen in de *database* aan.
 - We passen de redeneeralgoritmen in de *inference engine* aan.
 - We passen de regels in de *knowledge base* aan.**
 - We passen de regels in de *explanation facility* aan.
- (2 punten) Welke van onderstaande expressies komt overeen met de doorsnede (AND) van fuzzy sets A en B ?
 - $\mu_{A \cap B}(x) = \max [1 - \mu_A(x), \mu_B(x)]$.
 - $\mu_{A \cap B}(x) = \mu_A(x) + \mu_B(x) - \mu_A(x) \times \mu_B(x)$.
 - $\mu_{A \cap B}(x) = 1 - \max [1 - \mu_A(x), 1 - \mu_B(x)]$.**
 - $\mu_{A \cap B}(x) = 1 - \min [\mu_A(x), \mu_B(x)]$.

Antwoord: gebruik $\neg(A \cup B) = \neg A \cap \neg B$:

$$\mu_{A \cap B}(x) = 1 - \max [1 - \mu_A(x), 1 - \mu_B(x)] \quad (1)$$

$$= \min [1 - (1 - \mu_A(x)), 1 - (1 - \mu_B(x))] \quad (2)$$

$$= \min [\mu_A(x), \mu_B(x)] \quad (3)$$

- (2 punten) Stel we hebben de fuzzy sets A en B met membershipfuncties:

$$\mu_A(x) = (0.8/1, 0.2/2, 0.7/3, 0.1/4, 0.9/5, 0.2/6, 1.0/7, 0.0/8, 1.0/9) \quad (4)$$

$$\mu_B(x) = (1.0/1, 0.9/2, 0.8/3, 0.7/4, 0.6/5, 0.5/6, 0.4/7, 0.3/8, 0.2/9) \quad (5)$$

Wat is de membershipfunctie van de fuzzy set (*niet A*) of B als we gebruik maken van de standaardoperatoren?

- $\mu(x) = (1.0/1, 0.9/2, 0.8/3, 0.9/4, 0.6/5, 0.8/6, 0.4/7, 1.0/8, 0.2/9)$.
 - $\mu(x) = (1.0/1, 1.0/2, 0.7/3, 0.8/4, 0.8/5, 0.9/6, 0.7/7, 1.0/8, 0.7/9)$.
 - $\mu(x) = (0.0/1, 0.1/2, 0.2/3, 0.1/4, 0.4/5, 0.2/6, 0.6/7, 0.0/8, 0.8/9)$.
 - $\mu(x) = (1.0/1, 0.9/2, 0.8/3, 0.7/4, 0.9/5, 0.5/6, 1.0/7, 0.3/8, 1.0/9)$.
- (2 punten) Als $\mu_{mooiweer}(x)$ de membershipfunctie is van de fuzzy set *mooi weer*, dan is ...
 - $\mu(x)^{\frac{1}{2}}$ de membershipfunctie van *erg mooi weer*.
 - $\mu(x)^2$ de membershipfunctie van *erg mooi weer*.**
 - $1 - \mu(x)^2$ de membershipfunctie van *erg mooi weer*.
 - $1 - 2[\mu(x)]^2$ de membershipfunctie van *erg mooi weer*.

6. (2 punten) Welk soort training wordt gebruikt voor een Bidirectional Associative Memory (BAM)?
- A. Unsupervised learning.
 - B. Supervised learning.**
 - C. Reinforcement learning.
 - D. Hebb learning.
7. (2 punten) Bij een self-organizing map (Kohonen netwerk) ...
- A. wordt de verbinding tussen neuronen die tegelijk geactiveerd zijn sterker.
 - B. worden fouten in de invoer gecorrigeerd door het netwerk.
 - C. komt de positie van het output neuron in het netwerk overeen met een eigenschap van de invoer.**
 - D. wordt de foutgradient bepaald door de daadwerkelijke uitvoer te vergelijken met de gewenste uitvoer.
8. (2 punten) In een Genetic Programming systeem wordt cross-over toegepast op de LISP S-expressie $(+(*AA)(*BB))$. Een mogelijk resultaat van cross-over kan zijn:
- A. $(+(*AB)(*BA))$.**
 - B. $(-(*AA)(*BB))$.
 - C. $(+(*AA)B)$.
 - D. $(*(+AA)(+BB))$.
9. (2 punten) Om bij Genetische Algoritmen meer zekerheid te verkrijgen dat het globale maximum gevonden is en geen lokaal maximum, kunnen we kijken hoe stabiel het resultaat blijft als we een aantal parameters variëren. Welke parameters zijn dit?
- A. De crossoverkans p_c en de mutatie kans p_m .
 - B. De fitnessfunctie en de populatiegrootte.
 - C. De crossoverkans p_c en de fitnessfunctie.
 - D. De mutatiekans p_m en de populatiegrootte.**

Open vragen

10. Geef voor ieder van de onderstaande vraagstukken aan welke van de in dit vak behandelde oplossingsmethoden je zou gebruiken, motiveer je antwoorden.

- (a) (2 punten) De motor van een auto aan hand van het rijgedrag op ieder moment zo afstellen dat het benzineverbruik optimaal is.

Antwoord: Fuzzy logic: Deze kennis valt goed in regels uit te drukken. Het is echter niet noodzakelijk om voor elke combinatie van de continue inputwaarden een aparte regel te hebben.

- (b) (2 punten) Automatisch herkennen van gebarentaal.

Antwoord: Een neurale netwerk: dat kan omgaan met variatie in gebaren en met ontbrekende data. Systeem kan van voorbeelden leren.

- (c) (2 punten) Het plannen van de wedstrijden bij de Olympische spelen. Bedenk daarbij dat sommige onderdelen beslag leggen op dezelfde stadions, banen e.d., dat sommige sporters aan meerdere onderdelen meedoen en voldoende rust moeten krijgen tussen wedstrijden en dat het totale aanbod van wedstrijden zo gelijkmatig mogelijk over twee weken verdeeld moet zijn.

Antwoord: Dit is een complex optimaliseringsprobleem. Genetische algoritmen of eventueel particle swarm optimization.

- (d) (2 punten) Het automatisch sorteren van de favorieten in je internetbrowser zo dat links die naar pagina's met soortgelijke inhoud verwijzen dicht bij elkaar of bij elkaar in een mapje komen te staan.

Antwoord: Kohonen neurale netwerk of ant-based sorteren. In beide gevallen worden inputs automatisch gegroepeerd bij soortgelijke inputs zonder dat vooraf categorieën bepaald hoeven te worden.

- (e) (2 punten) Beantwoorden van deze vraag (dus bepalen welke op college behandelde techniek het meest geschikt is om een gegeven probleem op te lossen).

Antwoord: Expert systeem. Het gaat hier om kennis die een expert (jullie) toepast om het probleem op te lossen. De kennis die nodig is voor het beantwoorden van deze vraag is goed in regels uit te drukken.

11. Stel, we hebben een knowledge base met de volgende regels:

1. IF a THEN b
2. IF q THEN r
3. IF b AND c AND d THEN x
4. IF x AND y THEN z
5. IF d AND m THEN g
6. IF f THEN y
7. IF r AND f THEN p
8. IF l THEN m
9. IF p THEN c
10. IF a AND c THEN l

In de database komen de volgende feiten voor: **a, d, f, q**.
We gaan uit van een seriële (depth-first) verwerking van regels.

- (a) (4 punten) Laat zien hoe de inference engine aantoonst of doel **z** waar is. Geef aan welke feiten op de stack worden geplaatst en hoe de database verandert.

Antwoord: Database: a, d, f, q.

Doelen: z.

Stack: 4. Doelen x, y, z.

Stack: 3, 4. Doelen: b, c, (d), x, y, z.

Regel 1 vuurt, voegt b toe aan database.

Stack: 3, 4. Doelen: c, (d), x, y, z.

Stack: 9, 3, 4. Doelen: p, c, (d), x, y, z.

Stack: 7, 9, 3, 4. Doelen: r, (f), p, c, (d), x, y, z.

Regel 2 vuurt, voegt r toe aan de database.

f staat al in de database, regel 7 vuurt (van de stack af), voegt p toe aan de database.

Hierdoor kan regel 9 vuren (van de stack af), voegt c toe aan de database.

d staat al in de database, regel 3 vuurt (van de stack af) en voegt x toe aan de database.

Stack: 4. Doelen y, z.

Regel 6 vuurt, voegt y toe aan de database.

Hierdoor kan regel 4 vuren (van de stack af), voegt z toe aan de database.

Doel vervuld.

- (b) (3 punten) Wanneer maken expertsystemen gebruik van conflict resolution? Beschrijf twee methoden voor conflict resolution.

Antwoord: Conflict resolution wordt gebruikt om te bepalen welke regel mag vuren als er in een cycle meerdere regels zijn die kunnen vuren. Methoden voor conflict resolution:

- stoppen als het doel bereikt is. Regels met dezelfde antecedent maar een andere conclusie vuren niet.
- prioriteiten aan de regels toevoegen, de regel met de hoogste prioriteit mag vuren.
- Longest matching strategy. Specifiekere regels, d.w.z. regels met een langere antecedent mogen vuren.
- Data most recently entered. Voor ieder feit in de database wordt aangegeven wanneer dit is toegevoegd aan de database. De regel die de meest recent toegevoegde data gebruikt vuurt.

12. Beschrijf hoe we fuzzy inferentie kunnen doen met behulp van de Mamdani methode. Behandel achtereenvolgens de volgende onderwerpen, laat waar mogelijk plaatjes of formules zien.

Antwoord: Bij deze opgave wil ik plaatjes als op bladzijde 108 van het boek zien, voorzien van de nodige uitleg.

- (a) (2 punten) Fuzzificatie.

Antwoord: Fuzzificatie bepaald voor crisp inputs in welke mate deze behoren tot de juiste fuzzy sets. Een input heeft meestal een membership in meerdere sets. Voor drie fuzzy sets, *klein*, *gemiddeld* en *lang* zou een lengte van 182cm bijvoorbeeld de volgende waarden kunnen hebben: $\mu_{klein} = 0.1$, $\mu_{gemiddeld} = 0.6$, $\mu_{lang} = 0.4$. Geef dit weer in een plaatje of geef formules.

- (b) (2 punten) Regel evaluatie.

Antwoord: Voor alle regels worden de inputmembership waarden verkregen in (a) worden toegepast op de antecedenten. Als er meerdere antecedenten zijn worden deze gecombineerd met fuzzy operatoren, zoals *max* voor *OR* en *min* voor *AND* (alternatieven zijn *probor* en het product voor *AND*). De antecedenten worden toegepast op de consequenten doormiddel van clipping of scaling. Je hoeft alleen clipping uit te leggen. Het noemen van scaling geeft extra punten. Laat plaatjes of formules zien.

- (c) (2 punten) Aggregatie.

Antwoord: De consequenten van de regels verkregen in de vorige stap worden gecombineerd doormiddel van 'sommatie'. Over het algemeen wordt hiervoor de *max* operator gebruikt. Het resultaat is 1 fuzzy set. Geef hiervan een plaatje.

- (d) (2 punten) Defuzzificatie.

Antwoord: Defuzzificatie levert een crisp output uit de fuzzy set verkregen bij (c). Hiervoor berekenen we in de Mamdani methode het zwaartepunt (Centre of Gravity). Geef de formule van bladzijde 111 in het boek. Deze formule bevat integralen, in de praktijk benaderen we deze door sommatie te gebruiken en een aantal waarden in te vullen.

- (e) (2 punten) Waarin verschilt Sugeno inferentie van Mamdani inferentie?

Antwoord: In de vorm van de output functies in de regel consequent. Mamdani gebruikt fuzzy sets. Sugeno gebruikt functies van de crisp inputvariabelen. De aggregatiestap bij Sugeno komt neer op het berekenen van een gewogen gemiddelde, dit is over het algemeen efficiënter.

13. Multilayer feedforward neural networks zijn de meest gebruikte neurale netwerken. De gewichten van een dergelijk netwerk kunnen geleerd worden met behulp van de volgende regel:

$$w_{jk}(p+1) = w_{jk}(p) + \alpha \times y_j(p) \times \delta_k(p). \quad (6)$$

- (a) (4 punten) Leg uit hoe een feedforward neural network is opgebouwd en hoe het werkt. Behandel (minimaal) de netwerkstructuur, de inputs, de outputs en de activatiefunctie.

Antwoord: 1 input laag, 1 of meer verborgen lagen (hidden layers) en 1 output laag. Ieder neuron in een laag is verbonden met ieder neuron in de voorgaande laag. Aan de verbinding tussen neuronen *i* en *j* hangt een reel gewicht w_{ij} . Inputs zijn reële getallen of binaire waarden die op de inputs nodes gezet worden. De nodes in de hidden layers en de output laag berekenen hun activatie (waarde) aan hand van de waarden van hun inputs met behulp van een activatiefunctie. Meerdere activatiefuncties zijn mogelijk, het meest gebruikt is de sigmoidfunctie:

$$y_j = \frac{1}{1 + e^{-X_j}} \quad (7)$$

waarbij X_j de gewogen som van de inputs is die vergeleken wordt met een threshold waarde θ_j :

$$X_j = \sum_{i=1}^n x_i w_{ij} - \theta_j \quad (8)$$

. De threshold kan ook gerepresenteerd worden als een gewicht w_{0j} dat verbonden is met constante input met waarde -1.

- (b) (4 punten) Leg uit (in woorden, je hoeft geen formules te geven maar dit mag wel) hoe leren in een multilayer neural network werkt, geef daarbij aan waar de termen in bovenstaande leerregel (6) voor staan.

Antwoord: Leren in een multilayer neurale netwerk gebeurt aan de hand van voorbeelden die een voor een worden aangeboden aan het netwerk. Voor iedere input wordt eerst de output van het netwerk berekend. Deze output wordt vergeleken met de gewenste output. Dit levert een fout $e_k(p)$ op voor ieder output neuron k . Aan hand van deze fout wordt de error gradient $\delta_k(p)$ bepaald (door de afgeleide van de activatie functie van neuron k te vermenigvuldigen met de fout.)

De gewichten w_{jk} van neuron k worden vervolgens bijgesteld volgens bovenstaande regel. Het nieuwe gewicht hangt dus af van de huidige waarde van het gewicht, van de error gradient en van de output $y_k(p)$ van neuron k in iteratie p . De verandering in het gewicht wordt gewogen met de learning rate α , een getal tussen 0 en 1.

Aan hand van de outputwaarden in de output layer en de error gradienten in de output layer worden de error gradienten van de neuronen in de laatste verborgen laag berekend. De gewichten worden wederom bijgesteld met behulp van bovenstaande formule. Dit proces herhaalt zich tot de input laag bereikt is. De fout propageert dus van achter naar voren door het netwerk.

- (c) (2 punten) De *generalized delta rule* voegt aan leerregel (6) nog een momentum term toe:

$$w_{jk}(p+1) = w_{jk}(p) + \beta \Delta w_{jk}(p) + \alpha \times y_j(p) \times \delta_k(p). \quad (9)$$

Wat is het nut van deze term?

Antwoord: De momentum term neemt ook de vorige verandering van het gewicht mee gewogen met een constante $\beta \in [0, 1)$. Het momentum stabiliseert het leerproces. Als de veranderingen constant in dezelfde richting zijn (een eenvoudige zoekruimte) wordt het leerproces versneld, als de veranderingen in het gewicht steeds tegengesteld zijn (en de zoekruimte kennelijk veel pieken en dalen bevat) worden de leerstappen kleiner zodat het algoritme nauwkeuriger zoekt.

- (d) (1 punt) Welke functies kan een multilayer neural network wel/niet leren?

Antwoord: Multilayer neurale netwerken met een verborgen laag kunnen alle continue functies van de inputs leren. Netwerken met twee verborgen lagen kunnen ook discontinue functies representeren. (Dit is een theoretisch resultaat, het zegt niet hoeveel neuronen in die lagen moeten zitten).

14. Genetische Algoritmen en Evolution Strategies zijn twee vormen van Evolutionary Computation.

- (a) (4 punten) Waarin verschillen Genetische Algoritmen en Evolution Strategies?

Antwoord:

Genetische algoritmen maken gebruik van cross-over en mutatie, evolution strategies alleen van mutatie. Genetische algoritmen coderen een oplossing in een chromosoom (meestal in een bitstring),

mutatie is dan een bitflip. Evolution strategies werken direct op een vector van parameters en gebruiken een normale verdeling om mutaties te realiseren. De variant van evolution strategies die in het boek beschreven wordt, genereert slechts een nakomeling per iteratie en vervangt de ouder alleen door deze nakomeling als deze een hoger fitness heeft. Je zou hier ook het automatisch aanpassen van de mutatieparameters kunnen noemen.

- (b) (3 punten) Volgens sommigen valt ook Particle Swarm Optimization onder Evolutionary Computation. Wat zijn de overeenkomsten en verschillen tussen Particle Swarm Optimization en Evolutionary Computation?

Antwoord: In beide gevallen is er sprake van een populatie van oplossingen (chromosomen / particles) en bevat het zoekproces een stochastische component. Beide methoden maken gebruik van een fitnessfunctie. Bij Evolutionary computation wordt deze echter direct gebruikt om de fitness van een oplossing te bepalen, terwijl Particle Swarm Optimization fitness alleen gebruikt om te bepalen wat de beste globale of lokale oplossing is. In tegenstelling tot chromosomen worden particles nooit verwijderd en beschikken zij over een geheugen.

15. (4 punten) Een postbedrijf moet dagelijks een groot aantal brievenbussen legen. Per stad heeft het hiervoor een aantal auto's beschikbaar. Om de kosten laag te houden wil het bedrijf de routes langs alle brievenbussen in de stad zo kort mogelijk houden. Leg uit hoe het postbedrijf de routes kan plannen met behulp van ant-based algorithms.

Antwoord:

Dit is een travelling salesman probleem waarbij we de kortste route langs alle brievenbussen in de stad moeten vinden. Een ant-based algorithm doet dit door willekeurig 'mieren' op pad te sturen over een netwerk representatie van de stad waarin de knopen brievenbussen voorstellen en de verbindingen routes van de ene brievenbus naar de andere. Als een mier een volledige route langs alle brievenbussen heeft afgelegd, keert hij terug op zijn schreden en geeft aan ieder 'verbinding' tussen twee brievenbussen een gewicht dat omgekeerd proportioneel is aan de lengte van de route. Het totale gewicht van een verbinding is de som van de gewichten die alle mieren er aan toekennen. Het gevolg hiervan is dat verbindingen die deel uitmaken van kortere routes een hoog gewicht krijgen terwijl verbindingen die deel uitmaken van een langere route een laag gewicht krijgen.

Dit proces wordt een aantal iteraties herhaald, waarbij de mieren gevoelig zijn voor de gewichten op de verbindingen, hoe hoger het gewicht hoe groter de kans dat de mieren deze verbinding zullen volgen.

Na een aantal iteraties zullen de meeste mieren gebruik maken van de kortste paden door het netwerk.

16. (4 punten) De heer Holmes gaat 25 dagen per jaar op vakantie. Als hij niet op vakantie is zit hij 70% van zijn werkdag achter zijn bureau en beantwoordt hij in 80% van alle gevallen zijn e-mail binnen een dag. Als hij officieel vakantie heeft, komt Holmes toch nog wel eens naar zijn werk. Gemiddeld zit hij in zijn vakantie 1% van de (werk)tijd achter zijn bureau. Holmes beantwoordt tijdens zijn vakantie zijn e-mail minder snel: slechts 30% krijgt binnen een dag een antwoord. Stel nu dat Dr. Watson Holmes niet achter zijn bureau aantreft en Holmes ook zijn e-mail niet beantwoordt. Wat is dan de kans dat Holmes op vakantie is (je mag bij deze opgave even vergeten dat er schrikkeljaren bestaan).

Antwoord:

We zijn op zoek naar: $P(V = true | B = false, M = false)$.

Volgens de regel van Bayes:

$$P(V = true|B = false, M = false) = \frac{P(B = false, M = false|V = true)P(V = true)}{\sum_V (P(B = false, M = false|V)P(V))} \quad (10)$$

Als we veronderstellen dat de snelheid waarmee Holmes zijn mail beantwoord (M) en tijd die hij achter zijn bureau zit (B) onafhankelijk zijn gegeven dat hij op vakantie is of niet (V) dan wordt dit:

$$\alpha \times P(B = false|V = true) \quad P(M = false|V = true)P(V = true) = \quad (11)$$

$$\alpha \times (1 - 0.01) \quad (1 - 0.3)25/365 = 0.04747 \quad (12)$$

Om α te kunnen berekenen moeten we ook kijken naar het geval dat Holmes niet op vakantie is:

$$P(B = false|V = false)P(M = false|V = false) \quad P(V = false) \quad (13)$$

$$(1 - 0.7)(1 - 0.8) \quad (365 - 25)/365 = 0.05589 \quad (14)$$

dan volgt:

$$\alpha = \frac{1}{0.04747 + 0.05589} = 9.67492 \quad (15)$$

en dus

$$P(V = true|B = false, M = false) = 9.67492 \times 0.04747 = 0.45927 \quad (16)$$

Einde tentamenopgaven.

Controleer voor de zekerheid of je alle vragen hebt beantwoord. Het zouden er 16 moeten zijn.