Linköpings Universitet
Institutionen för Datavetenskap
Patrick Doherty

Tentamen
TDDC17 [1]Artificial Intelligence
11 January 2005 kl. 8-12

Points:

The exam consists of exercises worth 29 points.
To pass the exam you need 15 points.

Auxiliary help items:

Hand calculators.

Directions:

You can answer the questions in English or Swedish.
Use notations and methods that have been discussed in the course.
In particular, use the definitions, notations and methods in appendices 1-3.
Make reasonable assumptions when an exercise has been under-specified.
Begin each exercise on a new page.
Write only on one side of the paper.
Write clearly and concisely.

Jourhavande: Per Nyblom, 0705-31 97 31. Per will arrive for questions around 10 o' clock.

[1]This exam should also be used for those students who have registered for TDDA13 or TDDA58.

1. Consider the following theory (where x, y and z are variables and $\mathsf{history}$, $\mathsf{lottery}$ and john are constants):

$$\forall x([Pass(x, \mathsf{history}) \wedge Win(x, \mathsf{lottery})] \Rightarrow Happy(x)) \tag{1}$$

$$\forall x \forall y([Study(x) \vee Lucky(x)] \Rightarrow Pass(x, y)) \tag{2}$$

$$\neg Study(\mathsf{john}) \wedge Lucky(\mathsf{john}) \tag{3}$$

$$\forall x(Lucky(x) \Rightarrow Win(x, \mathsf{lottery})) \tag{4}$$

   (a) Convert formulas (1) - (4) into clause form. **[1p]**

   (b) Prove that $Happy(\mathsf{john})$ is a logical consequence of (1) - (4) using the resolution proof procedure. **[2p]**

   - Your answer should be structured using a resolution refutation tree (as used in the book).
   - Since the unifications are trivial, it suffices to simply show the binding lists at each resolution step.

2. Alan Turing proposed the Turing Test as an operational definition of intelligence.

   (a) Describe the Turing Test using your own diagram and explanations.**[2p]**

   (b) Do you believe this is an adequate test for machine intelligence? Justify your answer.**[1p]**

3. The Situation Calculus is a logical formalism for representing action and change.

   (a) Give a concrete example of the Frame Problem in the Situation Calculus using logical formulas.**[2p]**

   (b) Describe a solution to the problem using the example.**[1p]**

   (c) Is it possible to use the Situation Calculus for planning? If so, explain how. If not, why not?**[1p]**

4. Consider the following example:

   Aching elbows and aching hands may be the result of arthritis. Arthritis is also a possible cause of tennis elbow, which in turn may cause aching elbows. Dishpan hands may also cause aching hands.

   (a) Represent these causal links in a Bayesian network. Let ar stand for "arthritis", ah for "aching hands", ae for "aching elbow", te for "tennis elbow", and dh for "dishpan hands". **[2p]**

   (b) Given the independence assumptions implicit in the Bayesian network, write the formula for the full joint probability distribution over all five variables? **[2p]**

   (c) Compute the following probabilities using the formula for the full joint probability distribution and the probabilities below:

   - $P(ar \mid te, ah)$**[1p]**
   - $P(ar, \neg dh, \neg, te, ah, \neg ae)$**[1p]**
   - Appendix 2 provides you with some help in answering these questions.

Table 1: probabilities for question 5.

$$P(ah \mid ar, dh) = P(ae \mid ar, te) = 0.1$$
$$P(ah \mid ar, \neg dh) = P(ae \mid ar, \neg te) = 0.99$$
$$P(ah \mid \neg ar, dh) = P(ae \mid \neg ar, te) = 0.99$$
$$P(ah \mid \neg ar, \neg dh) = P(ae \mid \neg ar, \neg te) = 0.00001$$
$$P(te \mid ar) = 0.0001$$
$$P(te \mid \neg ar) = 0.01$$
$$P(ar) = 0.001$$
$$P(dh) = 0.01$$

5. A* search is the most widely-known form of best-first search. The following questions pertain to A* search:

   (a) Explain what an *admissible* heuristic function is using the notation and descriptions in (c). **[1p]**

   (b) Suppose a robot is searching for a path from one location to another in a rectangular grid of locations in which there are arcs between adjacent pairs of locations and the arcs only go in north-south (south-north) and east-west (west-east) directions. Furthermore, assume that the robot can only travel on these arcs and that some of these arcs have obstructions which prevent passage across such arcs.

   The *Mahattan distance* between two locations is the shortest distance between the locations ignoring obstructions. Is the Manhattan distance in the example above an admissible heuristic? Justify your answer explicitly. **[2p]**

   (c) Let $h(n)$ be the estimated cost of the cheapest path from a node n to the goal. Let $g(n)$ be the path cost from the start node n_0 to n. Let $f(n) = g(n) + h(n)$ be the estimated cost of the cheapest solution through n.

   Provide a general proof that A* using tree-search is optimal if $h(n)$ is admissible. If possible, use a diagram to structure the proof. **[2p]**

6. Use the table below and the information in appendix 3 to answer the following questions pertaining to version spaces and the candidate elimination algorithm.

   The following table shows a list of examples for a hypothetical credit-scoring application which might be used by a bank or credit-card company to filter prospective customers as to their credit worthiness. The target function is "person x is creditworthy".

Example	*Age*	*Mortage*	*Default*	*Length Employed*	*Surplus %*	*Creditworthy*
1	18-60	No	No	1-5	No	Yes
2	18-60	Yes	No	1-5	No	Yes
3	< 18	No	No	< 1	No	No
4	18-60	No	Yes	1-5	No	No
5	18-60	No	No	> 5	No	Yes

Figure 1: Positive and negative training examples for Target Attribute Creditworthy

   In the problem above *Creditworthy* is described in terms of the five attributes: Age, Mortage (for house), Default (on loan), Length Employed, and Surplus (cash).

   (a) Is the Candidate-Elimination algorithm a supervised or unsupervised learning algorithm? Why?**[1p]**

   (b) Use the Candidate-Elimination algorithm to compute the version space containing all hypotheses from H (the space of hypotheses) that are consistent with the observed sequence of training examples in the table above for the target attribute *Creditworthy*. **[4p]**

   - Initialize the algorithm with $G_0 = \{\langle ?, ?, ?, ?, ? \rangle\}$ and $S_0 = \{\langle \emptyset, \emptyset, \emptyset, \emptyset, \emptyset \rangle\}$
   - Provide a hand trace of the Candidate-Elimination algorithm learning from the training examples in the table above. In particular, show the specific and general boundaries of the version space after it has processed the first training example, the second training example, etc.

7. The following question pertains to adverserial search. Consider the game tree in Figure 2 in which static scores are all from the first players point of view. (Triangle is maximizer. Upside-down triangle is minimizer.)

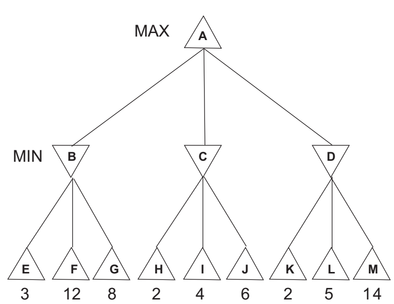

Figure 2: A Minimax Game Tree

   (a) In the game tree in Figure 2, what nodes would not need to be examined using the alpha-beta procedure? Justify your answer by describing the α/β values in the nodes of the tree and why branches would be cutoff based on this.**[3p]**

Appendix 1

Converting arbitrary wffs to clause form:

1. Eliminate implication signs.
2. Reduce scopes of negation signs.
3. Standardize variables within the scopes of quantifiers (Each quantifier should have its own unique variable).
4. Eliminate existential quantifiers. This may involve introduction of Skolem constants or functions.
5. Convert to prenex form by moving all remaining quantifiers to the front of the formula.
6. Put the matrix into conjunctive normal form. Two useful rules are:
   - $\omega_1 \vee (\omega_2 \wedge \omega_3) \equiv (\omega_1 \vee \omega_2) \wedge (\omega_1 \vee \omega_3)$
   - $\omega_1 \wedge (\omega_2 \vee \omega_3) \equiv (\omega_1 \wedge \omega_2) \vee (\omega_1 \wedge \omega_3)$
7. Eliminate universal quantifiers.
8. Eliminate $\wedge$ symbols.
9. Rename variables so that no variable symbol appears in more than one clause.

Appendix 2

A generic entry in a joint probability distribution is the probability of a conjunction of particular assignments to each variable, such as $P(X_1 = x_1 \wedge \ldots \wedge X_n = x_n)$. The notation $P(x_1, \ldots, x_n)$ can be used as an abbreviation for this.

The chain rule states that any entry in the full joint distribution can be represented as a product of conditional probabilities:

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid x_{i-1}, \ldots, x_1) \tag{5}$$

Given the independence assumptions implicit in a Bayesian network a more efficient representation of entries in the full joint distribution may be defined as

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid \mathit{parents}(X_i)), \tag{6}$$

where $\mathit{parents}(X_i)$ denotes the specific values of the variables in $\mathit{Parents}(X_i)$.

Recall the following definition of a conditional probability:

$$\mathbf{P}(X \mid Y) = \frac{\mathbf{P}(X \wedge Y)}{\mathbf{P}(Y)} \tag{7}$$

The following is a useful general inference procedure:

Let X be the query variable, let $\mathbf{E}$ be the set of evidence variables, let $\mathbf{e}$ be the observed values for them, let $\mathbf{Y}$ be the remaining unobserved variables and let α be the normalization constant:

$$\mathbf{P}(X \mid \mathbf{e}) = \alpha \mathbf{P}(X, \mathbf{e}) = \alpha \sum_{\mathbf{y}} \mathbf{P}(X, \mathbf{e}, \mathbf{y}) \tag{8}$$

where the summation is over all possible $\mathbf{y}$'s (i.e. all possible combinations of values of the unobserved variables $\mathbf{Y}$).

Appendix 3

Begin:Candidate-Elimination Learning Algorithm

Initialize G to the set of maximally general hypotheses in H
Initialize S to the set of maximally specific hypotheses in H

For each training example d, do

- If d is a positive example
  - Remove from G any hypothesis inconsistent with d
  - For each hypothesis s in S that is not consistent with d
    - Remove s from S
    - Add to S all minimal generalizations h of s such that
      - h is consistent with d, and some member of G is more general than h
    - Remove from S any hypothesis that is more general than another hypothesis in S
- If d is a negative example
  - Remove from S any hypothesis inconsistent with d
  - For each hypothesis g in G that is not consistent with d
    - Remove g from G
    - Add to G all minimal specializations h of g such that
      - h is consistent with d, and some member of S is more specific than h
    - Remove from G any hypothesis that is less general than another hypothesis in G

End: Candidate-Elimination Learning Algorithm

Hypothesis representation: Let each hypothesis be a vector of n constraints, specifying the values of each of the n attributes in a problem. For each attribute, the hypothesis will either

- indicate by a "?" that any value is acceptable for this attribute,
- specify a single required value for the attribute
- indicate by a "$\emptyset$" that no value is acceptable.

Definition 1
A hypothesis h is **consistent** with a set of training examples D if and only if $h(x) = c(x)$ for each example $\langle x, c(x)\rangle$ in D.[2]

$$Consistent(h, D) \equiv (\forall \langle x, c(x)\rangle \in D) h(x) = c(x).$$

Definition 2
The **general boundary** G, with respect to hypothesis space H and training data D, is the set of maximally general members of H consistent with D.[3]

$$G \equiv \{g \in H \mid Consistent(g, D) \wedge (\neg \exists g' \in H)[(g' >_g g) \wedge Consistent(g', D)]\}$$

Definition 3
The **specific boundary** S, with respect to hypothesis space H and training data D, is the set of minimally general (i.e., maximally specific) members of H consistent with D.

$$S \equiv \{s \in H \mid Consistent(s, D) \wedge (\neg \exists s' \in H)[(s >_g s') \wedge Consistent(s', D)]\}$$

[2]Here, x is an example, h is a hypothesis returning true or false and c is the target concept returning true or false.
[3]$h >_g h'$ states that h is strictly more general than h'. $h_j \geq_g h_k$ if and only if: $\forall (x \in X)[(h_k(x) = 1) \rightarrow (h_j(x) = 1)]$.