

COMP422/522-07B Test

10 October 2007

First name: _____ Last name: _____

ID Number: _____

Instructions

1. Write your name and ID number into the spaces provided above.
2. There are five questions of equal value.
3. All questions have to be answered.
4. Time allowed is 50 minutes.
5. Write your answers in the spaces provided. Do not use your own paper! There is a blank page at the end of the test that you can use. You can get extra paper from us if necessary.

DO NOT TURN THE PAGE UNTIL YOU
ARE ASKED TO DO SO!

Question 1: Models

Given the following denials, fact, and general rules, compute TWO different models:

```
<- student(X), vampire(X)
<- student(X), professor(X)
<- female(X), male(X)
being(dracula) <-
clever(X) ; student(X) <- being(X)
female(X) ; male(X) ; vampire(X) <- being(X)
student(X) ; professor(X) ; vampire(X) <- being(X)
```

Question 2: Different representations for Learning

Discuss the relationship between boolean learning and attribute-value learning. Specifically answer the following questions:

1. How can one represent a boolean learning problem using an attribute-value representation?
2. Can one also represent attribute-value learning problems using a boolean representation?
3. Is any information lost when transforming either way?

To illustrate your answer with an example, assume a boolean version of the play_tennis problem using the following boolean variables:

hot, windy, rainy, humid, play

Use the following version of the play_tennis problem (in attribute-value representation using the Weka format) to answer part 2:

```
@attribute temperature numeric
@attribute forecast {sunny,overcast,showers,rain}
@attribute windspeed numeric
@attribute play_tennis {true,false}
```

Question 3: Properties of Quality Criteria

Show that the accuracy criterion $acc(\text{clause}, pos_{\text{covered}}, neg_{\text{covered}}) \geq \text{threshold}$ is neither monotonic nor anti-monotonic. [HINT: To show that some property does not hold, it is sufficient to provide a counter-example]

Question 4: Version Spaces

Assuming the following background knowledge:

`{ male(john), female(mary), female(kim), parent(john,kim), parent(mary,kim) }`

and assuming rules for $father(X, Y)$ can be constructed using any subset of the following conditions:

`{ parent(X,Y), parent(Y,X), male(X), female(X), female(Y) }`

show what S and G (the specific and the general border) are after processing the following examples in the given order:

1. `father(john,kim)` POSITIVE
2. `father(mary,kim)` NEGATIVE
3. `father(john,john)` NEGATIVE

HINT:

`G0 = { father(X,Y) <- }`

`S0 = { father(X,Y) <- parent(X,Y), parent(Y,X), male(X), female(X), female(Y) }`

Question 5: Bottom clause

Using the following background knowledge:

```
father(X,Y) <- parent(X,Y),male(X)
mother(X,Y) <- parent(X,Y),female(X)
married(john,mary) <-
```

compute the bottom clause for:

```
pos <- parent(john,kim),parent(kim,ben),male(john),female(kim),male(ben)
```

HINT: when the clause does not contain variables, there is no need for skolemization/de-skolemization.

Extra space for answering questions