CS 662
Artificial Intelligence Programming
Mid-term 2 Review Sheet

Topics

- First-order logic
  - Definitions – models, quantifiers, variables
  - Converting English sentences into FOL
  - Inference – forward and backward chaining
  - Unification and substitution
  - Removing quantifiers / skolemization
  - Converting into CNF
  - Resolution
- Probability & Uncertainty
  - Axioms of probability
  - Joint probability distributions
  - Independence and Conditional Independence
  - Bayes Rule
- Machine learning
  - General terms and definitions
    - Learning task
    - Instances and attributes
    - Classification
    - Hypothesis space
  - Accuracy measures (precision, recall, accuracy)
  - Data handling issues
  - Decision trees:
    - using them to classify an instance
    - building them from data
    - Entropy and information gain
- Natural language processing
  - Why NLP is challenging (ambiguity, for example)
  - Definitions – parse tree, grammar
  - Top-down parsing
  - Bottom-up parsing
  - Chart parsing including active and inactive edges
  - Bag of Words model
  - Statistical NLP
    - N-grams
    - Smoothing
    - Probabilistic CFGs (PCFG)
  - Applications – Information Extraction
- Information retrieval
  - Searching versus browsing
  - TFIDF

    - Vector models and Cosine similarity
- Advanced knowledge representation
    - Semantic networks
    - Frames
    - Ontologies and Description Logic (very high level)
    - Applications – semantic web
- Planning
    - PDDL – representation
    - Progression and regression planning
    - Planning Graphs and GRAPHPLAN

Book sections:

- Ch 8; minimal coverage of 8.3 and 8.4
- Ch 9; in 9.4, only 9.4.1
- Ch 10, but not 10.4 or 10.5
- Ch 12: only 12.1, 12.2, and 12.5
- Ch 13; can skim 13.6
- Ch 18: only 18.1, 18.2, and 18.3
- Ch 22: 22.1, 22.3 (but stop at PageRank)
- Ch 23: 23.1, 23.2 stopping at 23.2.1

Types of questions

Here are the types of questions I tend to ask; more detailed examples are listed below:

- Define terms in your own words. True/false/explain for given definitions.
- Give the setup for a problem, such as parsing or planning.
- Sketch the execution of an algorithm such as decision trees.
- Work problems (various topics, see below for examples)
- Give pseudocode describing an algorithm; fix broken pseudocode
- Compare different algorithms and their strengths/weaknesses.
- Explain why an algorithm has the characteristics it does. For example, what does the agenda versus the chart do in chart parsing?
- Time and space complexity for different algorithms.

I won’t typically ask:

- Write syntactically correct Python code. (I may ask you to write pseudocode, but I'll be interested in program correctness, not whether every bracket and colon is in the right place)
- Multiple-choice questions.
- "Trick" questions - the point of this exam is to test your knowledge of the topics we've covered. I may ask you hard questions, but I won't ask things that are deliberately misleading or deceptive (unless I make a mistake, for which I will then be lenient in grading or hopefully notice it during the exam itself and correct or drop the question)

Detailed examples

1. Definition Questions. Define in your own words or “True / False / Explain”. If it is true, mark it true. If it is false, correct the statement so that it is true. Note: Adding ``not'' or otherwise negating the sentence is not acceptable. You must change the facts in the sentence if it is false.

   Sample question: The Turing test is a test of whether a computer program is rational

   Bad Answer, no credit: The Turing test is **not** a test of whether a computer program is rational

   Good Answer: The Turing Test is a test of whether a computer program is indistinguishable from a human.

   Any of the terms in the topic list above is fair game for a definition or brief explanation question. For example: Define Bayes Rule or state the difference between progression and regression planning.

2. Given a scenario, what AI domain/general approach would be appropriate.
3. Given a semantic network, what characteristics would be inherited by a given object.
4. Given a decision tree, and an unlabeled example, what class would the tree output.
5. Given a set of training data, what attribute would a decision tree learning algorithm split on first (using Information Gain)?
6. Problems such as those in HW4.
7. Given an English sentence and a logic sentence, say whether they have the same semantics, and fix them if they’re wrong.

   - For example: All students who like pizza like donuts and
   ∀x student(x) ^ likes(x,Pizza) ^ likes(x,Donuts)
   These do not have the same meaning. A fix to the logic would be:
   ∀x student(x) ^ likes(x,Pizza) → likes(x,Donuts)

- Filling out the numbers for slide 23 of the probability lecture