6.824, Artificial Intelligence

Fall 1994

Lectures Wednesday and Friday, 1:00 — 2:30, 3-442

	office	**phone**	**email**
David McAllester, Lecturer	NE43-412	3-6599	dam@ai.mit.edu
Milos Hauskrecht, TA	NE43-421	3-2351	milos@medg.lcs.mit.edu
Jean Darling, Course Secretary	NE43-403	3-4294	darling@ai.mit.edu

Course Description: This course covers technical material relevant to the construction of intelligent machines. The material includes search and planning algorithms; computer game-playing algorithms; knowledge representation languages; machine inference algorithms; formal models of natural language syntax and semantics; statistical inference algorithms; and the PAC formalism for concept learning. Philosophical issues of artificial intelligence will also be discussed. The course assumes prior exposure to the formal analysis of algorithms.

The grading in this course will be 40% based on the homework, 30% based on a three hour midterm exam and 30% based on a final during finals week. There will be ten problem sets. There will not be any programming assignments.

There is no official text for this course. Technical material will be given in specially prepared notes and published papers. Notes and papers containing the technical content of the course will be handed out at lecture. All handouts will be available outside of NE43-403. Postscript versions of all homework assignments and all other material for which we have electronic source will be available in a 6.824 locker on athena.

Students may discuss the problem sets, but each student is responsible for writing their own solution — no student may copy other students work and no joint solutions are allowed. In particular you may not show your solution to any other student, even if that student has already completed his or her own solution. This prevents copying and also prevents one student from checking or correcting other students solutions. You may discuss solutions as long as you do not discuss verbatum wording. Explaining a solution to another student is allowed but the wording of each student's solution must be written by that student.

There are 27 lectures in this course. The planned content of each lecture is listed below. The midterm will be a three hour evening exam on Tuesday Oct. 27 in 37-212 (the same room as lecture). The midterm will be on the material covered in the first five problem sets.

There will be an electronic mailing list for course announcements. Be sure to include an email address (if you have one) on the sign up sheet at the first day of lecture.

Lectures in 6.824, Artificial Intelligence

Introduction

1. **Introduction to Intelligent Computation** General discussion of artificial intelligence and its relationship to cognitive psychology, linguistics, philosophy, and other branches of computer science. Overview of the course.

Learning

2. **Natural Language Syntax and Semantics** The categorial approach to the syntax and semantics of natural language.

3. **Lexical Acquisition** This lecture will describe Jeff Siskind's work on learning the meanings of words from utterances in semantic contexts.

4. **Concept Learning** Mitchell's learning algorithm. PAC learning. Examples.

5. **Concept Learning II, VC dimension** The Vapnik-Chevronenkis dimension of a concept space as a criterion for learnability.

6. **Neural Networks** Neural Networks as function approximators. Convergence theorems for linear networks. Backpropagation. VC dimension of neural networks.

7. **Uncertainty and Bayesian Inference** Bayesian inference as a model of inductive and abductive inference. The problem of prior probabilities. The contrast with distribution free techniques. Minimum description length as a source of prior probabilities.

8. **Speech Perception** Basic Signal Processing. Hidden Markov models. The Viterbi Algorithm.

9. **Learning HMMs** The Baum-Welch reestimation technique.

10. **Bayesian Techniques in Vision** Edge detection. Markov random fields. Regularization.

11. **Learning Grammars** Stochastic grammars. The inside/outside algorithm. Using MDL to optimize Grammar size.

12. **Learning Probability Distributions** This lecture will discuss the MDL principle, its relationship to prior probabilities, and a PAC view of MDL.

Reasoning

13. **Reasoning About Probabilities** Computing conditional probabilities in Bayesian networks. The "back and forth" algorithm.

14. **Inference Rules** Inference as a model of computation. Inference rules and descriptive complexity theory.

15. **Semantics** Soundness and Completeness of inference rules.

16. **Constraint Satisfaction Problems** Introduction to constraint satisfaction search problems (CSPs). Using inference to accelerate search.

17. **Simulated Annealing and GSAT** Introduction to simulated annealing with applications to vision, protein folding, and circuit layout problems. The GSAT simplification of simulated annealing.

18. **Genetic Algorithms** Introduction to genetic algorithms with applications to a variety of optimization problems including concept learning. Discussion of "artificial life".

19. **Graph Search** The Dijkstra shortest path algorithm. The A* algorithm. Iterative deepening.

20. **STRIPS Planning** The STRIPS model of planning. Forward chaining and backward chaining sound and complete planners.

21. **Nonlinear Planning Algorithms** Systematic nonlinear planning (SNLP). Using inference to accelerate search in planning.

22. **Lifting** Lifting ground planners. Lifting as constraint posting and constraint propagation. Unification. The general lifting transformation and conditions under which it is valid.

23. **Games I** Min-max search. The concept of a strategy. α-β search. The best case behavior of α-β search. The relationship between games and and/or trees.

24. **Games II** The recent success of selective search. The fundamental concepts of conspiracy theory. Deep Thought's singular extension technique. Conspiracy theory. Informal principles of game search.

25. **Resolution Theorem Proving** Inference with clauses. The clausal representation of planning problems. The ground resolution rule. Lifted resolution.

26. **Knowledge Representation** Concept languages. The expressiveness/tractability tradeoff. Semantically universal languages. Relationship to current formal methods in computer science.

Conclusion

27. **Conclusion** Review of the course with an emphasis on philosophical issues. What will it take to build a computer that fully mimics human intelligence? When will it be done?