CS 3600-B: Introduction to Artificial Intelligence

Spring 2017

Lecture: MW 3:00-4:30, Howey L3

Instructor: David Byrd
Email: dave@imtc.gatech.edu
Office: TSRB 305A
Office Hours: MW 4:30-5:30 (after class) in CCB Commons

Lead TA: Oleg Filatov (ofilatov3@gatech.edu)

Undergrad TAs: Arjun Garg (arjun.garg@gatech.edu), Jason Gibson (jgibson37@gatech.edu), Greg Johnston (greg.johnston@gatech.edu), Pravan Kalaga (pkalaga@gatech.edu), Vishvak Saivenkat Murahari (vmurahari3@gatech.edu), Tejasvi Nareddy (tnareddy3@gatech.edu), Chris Waites (cwaites3@gatech.edu)

Grad TAs: Shagun Jhaver (jhaver.shagun@gatech.edu)

Syllabus

1. General Information

Introduction to Artificial Intelligence is a three-credit undergraduate course emphasizing the building of agents, environments, and systems that can be considered as acting intelligently. In particular, you will learn about the methods and tools that will allow you to build complete systems that can interact intelligently with their environment by learning and reasoning about the world.

2. Objectives

There are three primary objectives for the course: To provide a broad survey of AI; To develop a deeper understanding of several major topics in AI; To develop the design and programming skills that will help you to build intelligent artifacts.

In practice, you should develop enough basic skills and background that you can pursue any desire you have to learn more about specific areas in IS, whether those areas are planning, knowledge representation, machine learning, vision, robotics or whatever. In particular, this class provides a useful foundation for a number of courses involving intelligence systems, including Machine Learning (CS4641), Machine Learning for Trading (CS4803-MLT), Knowledge-Based AI (CS4634), Computer Vision (CS4495), Robotics and Perception (CS4632), Natural Language Understanding (CS4650) and Game AI (CS4731).

3. Prerequisites

Someone once said that the trick to doing AI is coming up with a good representation. That's not quite all there is to it, but it's close enough, so to succeed at this class, you should know a bit about data structures and algorithms. At the very least, you will have to be able to read pseudocode and understand basic algorithms as they are presented to you.

Someone else one defined AI as finding fast algorithms for NP-hard problems. Again, that's not quite all there is to it, but it's not too far from the truth, so it also turns out that a familiarity with (or at least a lack of abject fear over) some basic theory helps to situate many of the algorithms.

As the semester continues, it turns out that a familiarity with basic probability theory will also be very useful; however, we will spend some time on that in class in order to refresh your memory. Finally, you should feel pretty comfortable programming on your own. All projects will be implemented in Python. We will spend no time explaining languages in class; at this point in your career you've been exposed to several programming language and are expected to be able to readily acquire new programming language skills.

4. Resources

Required Text: Artificial Intelligence: A Modern Approach, Third Edition (the **blue** book) by Russell & Norvig, 2010.

Readings. The textbook for the course is the third edition of *Artificial Intelligence: A Modern Approach* by Russell and Norvig. There are significant differences between it and the first two editions, so be sure to have the right edition. We will follow the textbook quite closely (although time will not permit us to cover all of the chapters), so it is imperative that you have a copy of the book. We may occasionally use supplemental readings as well, but those will be provided for you.

Web. We will use the class web page (actually this GitHub repo) for the schedule, projects, and homework exercises. We will use TSquare for project submission and critical announcements. We will use a course management site called Piazza for general questions and discussions.

5. Grading

Practice Homeworks: Suggested homework assignments will be offered but not graded. Think of them as exam preparation (but generally harder than the exam questions). They do not contribute to your course grade.

Programming Assignments: There will be 4 graded projects throughout the semester; these will be worth a total of 50% of your final grade.

Exams: There will be a mid-term and a final exam, worth 20% and 30% of your final grade, respectively.

Class Participation: Participation means attending classes, participating in class discussions, asking relevant questions, volunteering to provide answers to questions, and providing constructive criticism and creative suggestions that improve the course or the textbook. Participation will be 0% of your grade; class participation to determine whether your grade can be lifted in case you are right on the edge of two grades.

Final Course Grades: The course grade boundaries will follow the traditional levels: 90.00+ == A, 80.00+ == B, etc. Because the distribution of grades in CS 3600 tends to be top-heavy already (# As > # Bs > # Cs > # Ds > # Fs), you should not expect a curve. For the same reason, there is no automatic upward rounding of grades. However, participation will be considered in these cases At most, participation can result in: grade = ceil(grade). (i.e. <= 89.00 will never become an A, even with excellent participation)

6. Plagiarism

You must abide by the academic honor code of Georgia Tech.

You must not collaborate, copy, or even glance at another student's work during exams.

You may collaborate without limitation on any ungraded homework assignments.

All code, images, and write-ups for graded projects must be yours alone. Project collaboration is liited to the "whiteboard level". You may discuss general approaches and algorithms, including high-level pseudocode. You should not share code or line-by-line pseudocode with other students. Giving and receiving code are equal violations of the honor code. Both will be referred to OSI.

DO go to the TAs for help on the programming projects. The TAs will know the limits of acceptable help, so you can safely collaborate with them without concern.

Do not store your programming assignment solutions on any *public* source repository (*e.g.* github.com). If another student finds and submits your published code, you are also guilty of an academic violation.

Do not turn in code you found on the web. I also use GitHub and StackOverflow...

There are no group projects in this class.

7. Course Policies

I reserve the right to modify the syllabus as necessary during the course. Changes will be minimized and you will be notified as early as possible.

Official communication for the class will come in lecture and/or via your Ga Tech e-mail. Piazza is used for discussion, but it is not an official communication channel. If you have an important issue to discuss with the instructor or a TA, send it via Ga Tech e-mail, *not* via a private Piazza post.

Student responsibilities: Be aware of the deadlines posted on the schedule. Read your GT email every day. Start work on projects when they become available.

Grade contests: If you believe there is a grading error in any project or exam, please communicate with the grading TA or instructor by e-mail within one week of the assignment being returned to you. You must have a very specific issue with a compelling argument as to why your grade is incorrect. Example compelling argument: "The TA took 10 points off because I did not cover topic X in my report, but on page 3 you can clearly see that I did." Example not compelling argument: "I think I should have gotten more points." Your assignment will be completely re-evaluated and your revised grade could be higher or lower.

Grade contest note: For all projects, the complete autograder is available to you alongside the project. Points will *not* be awarded if your code fails during grading, even for a "simple" mistake. You have the autograder. Use it and code carefully.

Late policy: Assignments are due at 11:55PM Eastern Time on the assignment due date. Assignments turned in after 11:55PM are considered late. Late assignments are not accepted without *prior* instructor approval.

Exam scheduling: Exams will be held on specific days at specific times. If there is an emergency or other issue that requires changing the date of an exam for you, you will need to have it approved by the Dean of Students. You can apply for that here: http://www.deanofstudents.gatech.edu (under Resources -> Class Absences)

Tentative Schedule (subject to change)

Week	Date	Topic	Due
1	1/9	Intro to AI, Agents	
	1/11	Agents and Environments	
2	1/16	Holiday	
	1/18	Agents, Problem Solving / Search	Project 0
3	1/23	Problem Solving / Search	
	1/25	Informed Search	
4	1/30	Constraint Satisfaction Problems	
	2/1	CSP	Project 1 (Search) due 2/5
5	2/6	Logic	
	2/8	Reasoning with Uncertainty	
6	2/13	Probability	
	2/15	Probability, Bayes Nets	
7	2/20	Bayes Nets	
	2/22	Midterm Review	
8	2/27	Midterm	
	3/1	Midterm Solutions, Dynamic Bayes Nets	
9	3/6	Dynamic Bayes Nets	Project 2 (CSP) due 3/5
	3/8	Dynamic Bayes Nets, Local Search	
10	3/13	Local Search, Optimization	
	3/15	Optimization	(*DROP DAY*)
11	3/20	Spring Break	
	3/22	Spring Break	
12	3/27	Learning	Project 3 (Bayes Nets) due 3/26

Week	Date	Topic	Due
	3/29	Learning	
13	4/3	Learning	
	4/5	Learning, Deep Learning	
14	4/10	Deep Learning	
	4/12	Deep Learning, Decision Making	
15	4/17	Decision Making	
	4/19	Decision Making	
16	4/24	"Final Instruction Days" - Exam Review	Project 4 (Learning) due 4/25
	4/26	Reading Period - no class	
	4/28	Final Exam @ 2:50pm	