

Final Project Guidelines

CS321: Artificial Intelligence, Winter 2017

We've covered a lot of material in AI, but there's still a great deal more out there. The final project is a chance to deepen your understanding of the material and apply it in an interesting area. You can choose to do essentially anything inside the area of AI, so long as I approve it in advance. You can build on material we do in class, or you can use it as an opportunity to learn about subject matter we don't have time to cover.

You are strongly encouraged to have some connection to machine learning, our final topic for the course. You can look at the topics on the schedule to get a sense of a couple machine learning areas we'll be talking about in class, and chapter 18 in Russel and Norvig is also a good source. The machine learning component could be the main component of the project, or might play a smaller part. For example, you might be mainly focused on how to apply search techniques to a new application area, but include a small machine learning portion to determine good feature weights for a heuristic. We'll start talking about machine learning on Monday, February 20. If you choose to do a project that does not include anything about machine learning, you'll want to be sure that it is sufficiently different from things you have implemented in a previous project; please consult with me if you have questions on what's different enough.

The main goals of the project are as follows:

- Demonstrate that you can characterize some application as involving a particular kind of AI problem.
- Design a solution to the problem that applies what you've learned in a way that is different than what you've done in previous homework assignments.
- Evaluate how well your solution works, and address questions about why it does or does not work well.
- Communicate your work so that someone who is familiar with AI can understand your approach.

You're welcome to choose any project you want as long as (a) I approve it, and (b) it either includes a machine learning component or significantly extends or differs from the homework projects (both superficially and in terms of algorithmic choices - i.e., if you wanted to do something MCTS-focused, you would need to go beyond simply applying MCTS to a different game). Here's a list of some projects that have worked well in the past:

- Computer game players, with machine learning for an evaluation function: Go, Poker, Cribbage, Othello.
- A simplified version of Parameterized Poker Squares (see [here](#) and come talk to me if you're interested in this one).
- Pac-Man Capture the Flag (see [here](#))
- Decision tree spam filter
- Comparison of the performance of several classification algorithms on a dataset and analysis of errors (see [UC Irvine machine learning repository](#) for a good source of possible datasets)

Here are some projects that haven't worked well in the past:

- Too complicated games: Risk, Yahtzee, Scrabble, Chess, Dominion, Battle Simulation (typically, this results in getting too focused on making the game work and not having enough time to do AI)
- "My really cool idea that I'm telling you, it's awesome, but I just can't really explain it"

Partnerships

You are **highly encouraged** to do the final project in partners; if you would like a partner but don't have someone specific in mind, post on Piazza to find someone to work with (check out the "Search for Teammates!" post) or email me if you'd like me to help facilitate finding you a partner. You may do the project in a group of three **if you make sure the project is large enough in scope to make sense for three people**. Groups of 3 will be expected to produce more extensive projects than groups of two, while individuals or groups of two will be expected to do projects of similar scope to one another (i.e., an individual project will be graded the same as if it had been done in a group of two).

Proposal

Your proposal should be no more than 1 page in length, and should describe your proposed project. The more specific you are, the more helpful my feedback will be. At a minimum, you'll want to describe what problem you're addressing, what kind of AI techniques you'll use, and how you'll know if your approach works well (i.e., how you'll evaluate your project). To ensure you have a solid foundation for your project, you should include two resources that you've read about your proposed approach. If you have a machine learning component, then one of those resources may be your textbook. The other can be another textbook, a paper, or a well-sourced webpage (typically not wikipedia). You shouldn't be looking for code (just like other assignments, you shouldn't be looking at other people's code, and you may not use other people's code), but trying to get a conceptual idea of the algorithm. It's okay if the part of the proposal related to machine learning is a little vague; if you aren't involving machine learning, make sure to it is clear how your project significantly differs from what we've implemented in class. If you have specific concerns about any part of the project, include them in your proposal - then I can give you feedback/advice! List the resources you used at the end of your proposal. Additionally, make sure that your proposal indicates who is working on it - only one of you and your partner need to turn it in, as long as you include both your names if working in a partnership.

Deliverables

There are four deliverables for the final project:

- A report describing your project. See below for details.
- The code for your project.
- A file named `readme.txt` that describes how to run your project, including where to download any relevant data if applicable. Specifically, how do I invoke your project from the command line, and are there any command line options I should know how to use? This documentation may be brief, but should enable me to interact with your program and see the project that you've described in your report.
- The partner survey on Moodle reporting how your partnership went (turn in after the project itself). This is only required if you worked with one or more partners. This survey **must** be turned in to receive any credit for a project completed with others.

Both the report and your code will contribute to your grade, and projects that do not include one portion (e.g., code but no report) will not receive credit.

Final project report

The final project report is where you'll communicate the results of your work. It should be no more than six pages and in PDF form. Your report may include figures or tables to communicate your approach and

results. Your report may be shorter than six pages - the important thing is that it clearly communicates your work (including an evaluation of your work) and how that work fits in to AI strategies for solving problems.

Your report should start with an introduction section. This section should set up what the problem you faced was - e.g., building an agent that can play Cribbage - and the way you approached that problem - e.g., treating Cribbage as a zero-sum, adversarial game. Your description of your approach should link to some larger AI context, demonstrating your ability to relate your project to ideas from class. The introduction should also give me a sense of what made your problem challenging and how you tackled that challenge. For example, Cribbage might be challenging because the opponent's hand can't be observed, or because you have to balance points in your hand versus points in the crib. I might handle the latter challenge by trying several different evaluation functions and comparing the results.

The next section of your report will give me more details about what you did. Here, you'll describe your strategy more specifically. For a search problem, for instance, you'd give a description of the state space and how you searched through it. This section should provide enough detail for the reader to understand the precise algorithms you were using, and for a coding-capable reader to be able to implement your solution. This is likely one of the longest parts of your paper, and may be broken down into several sections or subsections. Even if some of the algorithms you used are things covered in class, you should describe them in your report; imagine that your reader is an intelligent computer scientist who may have temporarily forgotten their AI knowledge, and thus needs you to briefly explain relevant algorithms and problem setups.

The third section of your report should describe some empirical results of your project: How did you measure how well your solution performed? What situations does it do well in, and when does it do poorly? If you explored several options (which often makes for a more effective project, such as comparing two feature sets for an evaluation function), how well did each option perform? If relevant, what data did you use to test your algorithm? Can you relate features of the data and/or your algorithm to your project's performance? Your project might be computationally intense - if so, how did you optimize it, and how long does it take to run? What is the computational complexity of the algorithms you're using? There are a number of different directions this portion of the report could take, but all reports must include an evaluation of some sort.

Finally, you'll end your report with a discussion section that sums up what you did and points to some future directions: how would you improve on what you did? What ideas do you have for alternative approaches for this problem, or what other problems could use the same approach as you took?