2018

Model AI Assignments 2018

Todd W. Neller  
Gettysburg College

Zack Butler  
Rochester Institute of Technology

Nate Derbinsky  
Northeastern University

See next page for additional authors

Follow this and additional works at: https://cupola.gettysburg.edu/csfac

Part of the Artificial Intelligence and Robotics Commons, and the Science and Mathematics Education Commons

Share feedback about the accessibility of this item.


This is the publisher's version of the work. This publication appears in Gettysburg College's institutional repository by permission of the copyright owner for personal use, not for redistribution. Cupola permanent link: https://cupola.gettysburg.edu/csfac/47

This open access conference proceeding is brought to you by The Cupola: Scholarship at Gettysburg College. It has been accepted for inclusion by an authorized administrator of The Cupola. For more information, please contact cupola@gettysburg.edu.
Model AI Assignments 2018

Abstract
The Model AI Assignments session seeks to gather and disseminate the best assignment designs of the Artificial Intelligence (AI) Education community. Recognizing that assignments form the core of student learning experience, we here present abstracts of seven AI assignments from the 2018 session that are easily adoptable, playfully engaging, and flexible for a variety of instructor needs. Assignment specifications and supporting resources may be found at http://modelai.gettysburg.edu.

Keywords
Artificial Intelligence, education, student learning

Disciplines
Artificial Intelligence and Robotics | Computer Sciences | Science and Mathematics Education

Comments
Presented at the Eighth Symposium on Educational Advances in Artificial Intelligence (EAAI-18) in New Orleans, LA on February 3-4, 2018.

Authors
Todd W. Neller, Zack Butler, Nate Derbinsky, Heidi Furey, Fred Martin, Michael Guerzhoy, Ariel Anders, and Joshua Eckroth

This conference proceeding is available at The Cupola: Scholarship at Gettysburg College: https://cupola.gettysburg.edu/csfac/47
Model AI Assignments 2018

Todd W. Neller
Gettysburg College
tneller@gettysburg.edu

Zack Butler
Rochester Institute of Technology
zjb@cs.rit.edu

Nate Derbinsky
Northeastern University
n.derbinsky@northeastern.edu

Heidi Furey
Manhattan College
h furye01@manhattan.edu

Fred Martin
University of Massachusetts Lowell
fred_martin@uml.edu

Michael Guerzhoy
University of Toronto
guerzhoy@cs.toronto.edu

Ariel Anders
Massachusetts Institute of Technology
aanders@csail.mit.edu

Joshua Eckroth
Stetson University
jeckroth@stetson.edu

Abstract

The Model AI Assignments session seeks to gather and disseminate the best assignment designs of the Artificial Intelligence (AI) Education community. Recognizing that assignments form the core of student learning experience, we here present abstracts of seven AI assignments from the 2018 session that are easily adoptable, playfully engaging, and flexible for a variety of instructor needs. Assignment specifications and supporting resources may be found at http://modelai.gettysburg.edu.

Go For a Walk! Pedestrian-Friendly A* with Learned Cost Functions

Zack Butler

This assignment is a two-part project based on path planning for pedestrians in an urban environment. In the first part of the project, students use real-world data collected from OpenStreetMap and digital elevation data to build an A* path planner. In particular, they need to consider the distance as well as elevation gained or lost on each edge, and provide a cost function and admissible heuristic derived from their ad-hoc estimates and/or published pedestrian models. Extensions use OpenStreetMap metadata to consider planning for bicycles and cars. In the second part of the project, the students in the course walk through the environment to collect actual data on the time taken to follow different paths. This data is shared to the entire class to build an overall data set. Students then perform regression over the features in their cost function using this data set to develop a new cost function. They can also compare the cost function derived from only their individual data to that derived from the overall class data set, and as an extension to the assignment, compare different types of regression algorithms.

Solve a Maze via Search

Nate Derbinsky

Copyright © 2018, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.
Understanding How Neural Networks Recognize Faces
Michael Guerzhoy

In this assignment, students build a feedforward neural network for face recognition using TensorFlow. Students then visualize the weights of the neural networks they train. The visualization allows students to understand feedforward one-hidden layer neural networks in terms of template matching, and allows students to explore overfitting. Using a framework such as TensorFlow allows for the students to be able to run a variety of experiments in order to obtain interesting visualizations.

Understanding How Recurrent Neural Networks Model Text
Michael Guerzhoy

In this assignment, students are introduced to Recurrent Neural Networks (RNNs) and explore the mechanism that allows RNNs to model English text character-by-character. Students learn to think of an RNN as a state machine. The assignment was originally used in a third-year neural networks and machine learning course, and could be adapted to an Intro AI course.

Recurrent Neural Networks have recently been shown to be remarkably effective at modelling complex time-series, including English text, character by character. Realistic-seeming “fake” English text can be generated using RNN models. Many students and practitioners tend to view RNNs as black boxes. This assignment forces students to understand how RNNs model complex time-series by having students explain the same properties of the outputs generated by the model in reference to the specific weights of the model.

In this assignment, students are provided with a “vanilla” RNN model that was trained on a corpus of Shakespeare play. The students are asked to write code to generate “fake” text from the model, and to then explain several properties of the text. Students are encouraged to find new interesting properties of the model for bonus marks.

Robot Juggling
Ariel Anders

In this assignment students learn how to control a robot to juggle a ball. To achieve this goal, students will program a velocity controlled robot, such that it causes the ball to bounce with some desired periodic motion. Specifically, students will implement a hybrid controller that uses a mirror control law within the framework of a 2D physics simulator.

Similar to many control assignments, the theoretical learning objectives deal with analyzing system responses and stability criterion. Additionally, designing and implementing the controller will give students a chance to translate theoretical knowledge into real-world practice. This assignment differentiates from most introductory control assignments because the dynamics of robot juggling is highly nonlinear, due to the collisions with the robot.

This assignment is tailored to Engineering Sophomore or Junior undergraduate students, with minimal programming experience. The assignment is implemented with Python, using readily accessible packages from Anaconda Cloud. Another highlight of this assignment is that it is free, open source, and capable of running across popular operating systems, like Windows, Mac, and Ubuntu.

Biductive Computing: Several Variants of a Universal Paradigm
Joshua Eckroth

This Model AI Assignment allows students to practice with logic programming and constraint programming in Prolog and ProbLog using a paradigm we call “biductive computing,” i.e. supporting both deductive and abductive inferences with the same code. This assignment includes four variants of biductive computing: database querying, planning, parsing, and probabilistic reasoning. In each variant, we describe a computational problem in a real-world domain, explain the biductive aspects of the desired implementation, provide test cases to measure correct solutions, and suggest possible enhancements. Each assignment variant may be used in an introductory AI course and, in the case of the database querying variant, may serve as a first assignment in Prolog.

References