COMP 6721: Artificial Intelligence

Fall 2019

Course Outline

Instructor: Zixi Quan, TBA, zixi.quan@concordia.ca

Office hours: Mondays & Wednesdays, 15:00–16:30 (or just email me)

Course schedule: Mondays & Wednesdays, 11:45–13:00, H-820

Lab schedule:

section FI: Monday 9:35-11:35 H849 with Andres Lou (lourios.andres@gmail.com)
section FJ: Wednesday 16:15-18:15 H907 with Pouria Chalangari (pouria.chalangari@gmail.com)
section FK: Wednesday 18:30-20:30 H843 with Farhood Farahnak (farhood.farahnak@gmail.com)

Recommended Book:

Stuart Russell and Peter Norvig, *Artificial Intelligence: A Modern Approach*, Prentice Hall.

Course Objectives: The purpose of the course is to provide a broad technical introduction of the core concepts of Artificial Intelligence (AI). Topics include: state-space search (uninformed and informed/heuristic search), adversarial search, machine learning, natural language processing and deep learning.

Web Page: Many resources for the course will be available on Moodle, accessible through the www.myconcordia.ca portal. Students should regularly consult the Moodle page for up-to-date information on the course, including slides, handouts, and exact course schedule.

Evaluation Scheme:

2 Projects ($2 \times 15\%$)	30%
1 Midterm exam	30%
1 Final exam	40%

Note: 1. The required programming language of two projects is **Python**.
2. While we encourage discussion among students, each student must independently and separately prepare and hand in their assignments. The maximum is two students allowed to finish each project. Students should be aware of the University's academic code of conduct, especially the sections concerning cheating, plagiarism, and the possible consequences of violating this code. Please see the details from https://www.concordia.ca/content/dam/common/docs/policies/official-policies/Academic-Code-Conduct-2015.pdf.
3. There is no fixed, a priori relationship between the numerical percentage and the final letter grades for this course. To pass the course, you must at least pass the projects and the final exam. Usually a score of 50% is required. There are no make-ups/alternates for missed exams or projects deadlines.

Tentative Schedule

Week	Dates	Topic	Chapter	Assignment
1	Sep 3 - Sep 8	**Introduction of AI**	1, 26	
2	Sep 9 - Sep 15	**State-Space Search**	3, 4.1.1	
3	Sep 16 - Sep 22	**Adversarial Search**	5.1-5.4	Project1 available
4	Sep 23 - Sep 29	**Machine Learning**	18.1-18.2	
5	Sep 30 - Oct 6	**Machine Learning Algorithms**		
6	Oct 7 - Oct 13	**Machine Learning Algorithms**		
		Neural Networks	19.1-19.6	
7	Oct 14 - Oct 20			Project1 due
	Mon Oct 14	**Thanksgiving – No class**		
	Wed Oct 16[1]	**Midterm Exam**		
8	Oct 21 - Oct 27	**Neural Networks**		Project2 available
		Natural Language Processing	22.1-22.3	
9	Oct 28 - Nov 3	**NLP: N-gram Models**	23.1, 24.7	
10	Nov 4 - Nov 10	**NLP: Syntax and Semantics**	23	
11	Nov 11 - Nov 17	**Deep Learning**		
12	Nov 18 - Nov 24	**Deep Neural Networks**		Project2 due
13	Nov 25 - Dec 2[2]	**Catch-Up and/or Review**		

Please note:
1. All the projects are due on the **Monday** of the due week.
2. In the event of extraordinary circumstances beyond the University's control, the content and/or evaluation scheme in this course is subject to change.
3. **Midterm Exam:** There will be one in-class midterm exam which will take place during regular lecture times. The term test will occur in week 7 of the term. There are no make-up midterm exams.
4. **Final Exam:** The final examination will last three hours, and will be administered during the examination period at the end of the term. The final examination covers all the materials seen during the term.
Note: The final exam is scheduled by the Exams Office. Do not make any other plans until the official Exam Schedule is Announced.

[0]The date of Midterm may change, as a classroom cannot be booked yet.
[1]Make-up day for Oct. 14 Thanksgiving Day

Graduate Attributes

As part of either the Computer Science or Software Engineering program curriculum, the content of this course includes material and exercises related to the teaching and evaluation of graduate attributes. Graduate attributes are skills that have been identified by the Canadian Engineering Accreditation Board (CEAB) and the Canadian Information Processing Society (CIPS) as being central to the formation of Engineers, computer scientists and information technology professionals. As such, the accreditation criteria for the Software Engineering and Computer Science programs dictate that graduate attributes are taught and evaluated as part of the courses. The following is the list of graduate attributes covered in this course, along with a description of how these attributes are incorporated in the course.

1. **Knowledge-base:** State space search (uninformed and informed/heuristic search, adversarial search), natural language processing and machine learning techniques.
2. **Use of Engineering Tools:** Determine appropriate programming languages, software libraries and other resources to develop programs that put into practice the foundations of artificial intelligence as taught in the lectures.
3. **Individual and Team Work:** Implementation of the projects and individual exams.
4. **Communications Skills:** Deliver projects as reports and/or oral presentations.