LING83800: METHODS IN COMPUTATIONAL LINGUISTICS II

SPRING 2021

CUNY GRADUATE CENTER

Instructor: Prof. Kyle Gorman
Practicum leader: Lucas Ashby
Lecture: Monday 4:15-6:15, the Cloud
Practicum: TBA, the Cloud
Office hours: Tuesday 2-4, the Cloud (by request)

SYNOPSIS

This course is the second of a two-semester series introducing computational linguistics and software development. The intended audience are students interested in speech and language processing technologies, though the materials will be beneficial to all language researchers.

OBJECTIVES

Using the Python programming language, students will learn core algorithms used to build speech and language technologies, and best practices for evaluation and basic statistical analysis.

MATERIALS

There is no textbook, but some readings will be assigned. Students are strongly encouraged to bring a laptop computer to the practicum. In some cases, the practicum may be held in the Computational Linguistics Laboratory (7400.13) currently under construction.

ASSIGNMENTS

Assignments will take the form of a small software development project accompanied by a write-up describing the general approach taken and any challenges encountered. Students will often be able to verify the technical correctness of their code by running provided tests. Students will also be graded on the readability of their code, the quality of documentation and the write-up. We will use GitHub Classroom for assignment turn-in.

The final assignment will be an open-ended project which will either extend earlier projects, or build and evaluate a speech and language technology system. Students are encouraged to conceive of projects relevant to their research interests. Students should discuss project plans with the instructor during office hours to confirm that it is both feasible and of appropriate scope.

GRADING

80% of students’ grades will be derived from the assignments; the remaining 20% will be reserved for participation and attendance. Assignments must be submitted on time or will receive a 0 grade (barring a documented emergency).

ACCOMODATIONS

The instructor will attempt to provide all reasonable accomodations to students upon request. If you believe you are covered under the Americans With Disabilities Act, please direct accomodations requests to Matthew G. Schoengood, Vice President for Student Affairs.
ATTENDANCE

Students are extended to attend all lectures and practica. If you are absent for any reason, please contact the instructor in advance with a brief explanation. The instructor reserves the right to tie grades to attendance records. The instructor and teaching assistant are not responsible for reviewing materials missed to absence. Assignments must be submitted on time or will receive a 0 grade (barring a documented emergency).

INTEGRITY

In line with the Student Handbook policies on plagiarism, students are expected to complete their own work. However, a student is permitted to collaborate with another student during the coding phase of an assignment so long as they: do not share lines of code with each other, mutually disclose their collaboration in their write-ups, and do not collaborate at all on their write-ups. The instructor reserves the right to refer violations to the Academic Integrity Officer.

RESPECT

For the sake of the privacy, students are not permitted to record lectures. Students are expected to be considerate of your peers and to treat them with respect during discussions.

SCHEDULE

(Please note that this is subject to change.)

<table>
<thead>
<tr>
<th>Date</th>
<th>Assignment due</th>
<th>Topic</th>
<th>Reading</th>
</tr>
</thead>
<tbody>
<tr>
<td>2/1</td>
<td></td>
<td>Syllabus and motivations</td>
<td>Hovy &amp; Spruit; Resnik &amp; Lin</td>
</tr>
<tr>
<td>2/8</td>
<td></td>
<td>Git &amp; GitHub</td>
<td>Chacon &amp; Straub ch. 1.1-3.2, 6.1-6.3</td>
</tr>
<tr>
<td>2/15</td>
<td></td>
<td>No class (President's Day)</td>
<td></td>
</tr>
<tr>
<td>2/22</td>
<td>HW1 due</td>
<td>Unit testing</td>
<td>unitest documentation</td>
</tr>
<tr>
<td>3/1</td>
<td>HW2 due</td>
<td>Command-line tools</td>
<td>Church</td>
</tr>
<tr>
<td>3/8</td>
<td>HW3 due</td>
<td>Sets, languages, and relations</td>
<td>Partee et al. ch. 1, 2, Hopcroft et al. ch. 1.5</td>
</tr>
<tr>
<td>3/15</td>
<td>HW4 due</td>
<td>Regular languages, regular expressions, and rational relations</td>
<td>Hopcroft et al. ch. 3-3.1, 3.3</td>
</tr>
<tr>
<td>3/22</td>
<td>HW5 due</td>
<td>Finite automata</td>
<td>Gorman &amp; Sproat ch. 1-1.4, 2, 3; Roark &amp; Sproat ch. 1, Eisenstein ch. 9; Hopcroft et al. ch. 2, 3.2; Jurafsky &amp; Martin ch. 2</td>
</tr>
<tr>
<td>3/29</td>
<td></td>
<td>No class (spring break)</td>
<td></td>
</tr>
<tr>
<td>4/5</td>
<td>HW6 due</td>
<td>Probability theory</td>
<td>Manning &amp; Schütze ch. 2</td>
</tr>
<tr>
<td>4/12</td>
<td>HW7 due</td>
<td>Language models</td>
<td>Gorman &amp; Sproat ch. 1.5-1.6; Roark &amp; Sproat ch. 6.1, Manning &amp; Schütze ch. 6</td>
</tr>
<tr>
<td>4/19</td>
<td>HW8 due</td>
<td>Hidden Markov models</td>
<td>Jurafsky &amp; Martin appendix A, Bird et al. ch. 5, Jelinek ch. 2; Manning &amp; Schütze ch. 9</td>
</tr>
<tr>
<td>4/26</td>
<td>HW9 due</td>
<td>Context-free grammars</td>
<td>Eisenstein ch. 10; Jurafsky &amp; Martin ch. 12</td>
</tr>
<tr>
<td>------</td>
<td>---------</td>
<td>-----------------------</td>
<td>---------------------------------------------</td>
</tr>
<tr>
<td>5/3</td>
<td>HW10 due</td>
<td>Generative classifiers</td>
<td>Bird et al. ch. 6.1-3, 6.5-6.9; Jurafsky &amp; Martin ch. 4</td>
</tr>
<tr>
<td>5/10</td>
<td>HW11 due</td>
<td>Discriminative classifiers</td>
<td>Breiman; Collins</td>
</tr>
<tr>
<td>5/17</td>
<td>HW12 due</td>
<td>scikit-learn tutorials 1, 2, 3</td>
<td>Freund &amp; Schapire; Ng &amp; Jordan</td>
</tr>
</tbody>
</table>

**REFERENCES**


[wellformedness.com/courses/LING83800/](http://wellformedness.com/courses/LING83800/)