COGS 123, CSE 173
Computational Cognitive Neuroscience

University of California, Merced

Spring, 2014

Instructor
David C. Noelle, Ph.D.
dnoelle@ucmerced.edu
262A Social Sciences & Management Building
(209) 228-4101
Office Hours: Tuesdays, 3:00 P.M. – 3:50 P.M., and by appointment

Teaching Assistant
William B. St. Clair
wst.clair@ucmerced.edu

Class Meetings
Tuesdays & Thursdays, 1:30 P.M. – 2:45 P.M., in 217 Kolligian Library
Fridays, 9:00 A.M. – 11:50 A.M., in 138 Science & Engineering Building
Fridays, 12:30 P.M. – 3:20 P.M., in 138 Science & Engineering Building

Overview

Cognitive neuroscience is concerned with the question of how the mass of interconnected neural cells that make up the brain can give rise to complex mental processes and intelligent behavior. Computer simulations of cognitive processes, informed and constrained by our knowledge of brain anatomy and physiology, can play a central role in this research endeavor by testing the feasibility of theories of brain function, uncovering hidden implications of such theories, and demonstrating how relatively simple biological mechanisms can work together to form powerful information processing systems which exhibit human-like patterns of performance. The generation, analysis, and testing of such computer simulations is the focus of computational cognitive neuroscience.

This course will introduce a collection of computer simulation techniques useful for investigating a variety of cognitive phenomena involving perception, action, learning, and memory. This introduction will include discussions of how such methods can leverage our growing neuroscientific knowledge.

This course is formatted to support interdisciplinary inquiry, with the backgrounds of students expected to vary broadly across the range of such disciplines as computer science, cognitive science, psychology, and neuroscience, as well as other related fields. The learning of both classic and contemporary methods for cognitive modeling will be
facilitated by readings, presentations by class participants as well as by the instructor, computer simulation exercises, a term project, and ample interaction and discussion between attendees.

Learning Outcomes

Students who successfully complete this course will have acquired an introductory understanding of the basic principles of computational cognitive neuroscience (CCN) modeling (including extensions to parallel distributed processing (PDP), connectionist, and artificial neural network models). By the end of the term, students will be able to relate the formal properties of such models to known biological mechanisms as well as to behavioral phenomena, and they will possess intellectual tools for modifying such models in the light of new psychological and neuroscientific findings. The knowledge acquired by the successful student will encompass a variety of broad topics, including: the mathematical and computational properties of CCN models, a survey of common modeling techniques, methods for extending traditional PDP techniques to incorporate relevant biological details, the use of CCN simulation software, methods for studying cognition through computational modeling and analysis, CCN approaches to perception and motor control, CCN models of language use and language learning, neural and cognitive development from a CCN perspective, and the role of CCN models in exploring the impact of brain damage and neurodegenerative disease.

These student learning outcomes are expected to be valuable to students pursuing a variety of undergraduate degree programs at the University of California, Merced. Specifically, with regard to the Cognitive Science B.S. and B.A. degree programs, these student learning outcomes contribute to a number of program learning outcomes, including increasing student knowledge of landmark findings and theories in cognitive science, developing student appreciation for formal and computational approaches in cognitive science, and improving the ability of students to argue for or against the use of computational models to address various scientific questions. With regard to the Computer Science and Engineering B.S. degree program, the student learning outcomes outlined in the previous paragraph contribute to a number of the computer science program learning outcomes, including practice applying knowledge of computing and related mathematics, practice designing, implementing, and evaluating computer-based systems, practice communicating technical information to an interdisciplinary audience, exposure to current computing techniques, and exposure to examples of trade-offs in software system design. In summary, students enrolled in a variety of undergraduate education programs should find this course useful for meeting their program’s objectives.

Resources

Meetings

Course participants will meet every Tuesday and Thursday of this semester according to the schedule outlined at the end of this document. Lecture and discussion meetings will take place from 1:30 P.M. to 2:45 P.M. in Room 217 of the Kolligian Library Building. These class meetings will consist of reviews of material available in course readings, introductions to new material, and discussions of related topics. Reviews will frequently be led by the instructor, but will sometimes involve presentations by other participants. The intended structure of these meetings includes substantial flexibility, allowing them to adapt to the interests and concerns of the participants.

In addition to these lecture and discussion meetings, course participants are expected to attend one laboratory session each week of the semester. Two such sessions are scheduled on a weekly basis, with both being held in Room 138 of the Science & Engineering Building. Laboratory sections are scheduled for 9:00 A.M. to 11:50 A.M. and for 12:30 P.M. to 3:20 P.M. on Fridays. These laboratory sessions will allow students to execute and explore computer simulations of human behavior and brain function in the company of the teaching assistant and fellow students. Computer simulation exercises will be conducted by individual participants during these laboratory meetings, and these exercises act as a central component of the learning process.
Web Site

Online materials for this class will be disseminated through the UCMCROPS system. This course management system may be accessed at:

http://ucmcrops.ucmerced.edu/

Enrolled students should be provided with access to a section of UCMCROPS specifically reserved for this class, labeled “S14-COMP COG NEURO”. The instructor should be promptly informed if such access is not appropriately granted.

This web site will be used to announce updates to the class schedule, as well as to distribute class materials. Students are required to obtain regular access to this resource, and they are strongly advised to consult it frequently (e.g., daily).

Readings

The primary source of expository readings for this class will be:


In addition to providing explanatory text, this book will also be used to guide the sequencing of topics discussed in this class, and computer simulation exercises will be drawn almost exclusively from its pages.

This online textbook is an updated and reduced version of the following previously published book:


While somewhat older, this second textbook contains more extensive material on some topics. Hardcopies of this MIT Press book are available from the campus bookstore. Electronic copies of the chapters in this tome are also available from the online CogNet service. While access to CogNet typically requires a subscription to the service, it is available to you, free of charge, through the university library web site ([http://library.ucmerced.edu/](http://library.ucmerced.edu/)).

While these two books provide rather comprehensive coverage of topics relevant to this course, they do not always offer thorough explanations of alternative or competing approaches to the modeling enterprise. Participants interested in obtaining a slightly wider view of the range of research methods employed in this field are encouraged to supplement their reading with other sources, including the following recommendations.

There is a collection of three seminal tomes in the world of PDP modeling, produced by a team of researchers at the University of California, San Diego in the early 1980s. Even after almost 30 years, the PDP volumes remain an excellent introduction to the use of connectionist techniques to understand psychological phenomena:


These books are still in print at the MIT Press (with the exception of the third volume, which contains exercises). Also, electronic copies of the chapters of these volumes are offered by CogNet.
A great collection of classic papers in the field of artificial neural network modeling may be found in:


This is also an MIT Press book, and an electronic version is also provided by the CogNet service.

An excellent contemporary text on computational neuroscience, focusing a bit more on neurobiological phenomena than behavioral phenomena, is:


This book also appears in electronic format on the CogNet service.

**Software**

Reading about computational modeling mechanisms is rarely sufficient to develop a deep understanding of them. In hopes of fostering a deeper understanding, class participants will conduct small simulation experiments throughout the semester using the *emergent* software package. This software system provides tools for the design, execution, and analysis of CCN models utilizing a wide variety of computational mechanisms and methods. The guiding text of this course provides extensive examples and exercises using *emergent*, and versions of these exercises will act as critical tools for communicating key concepts.

The *emergent* system is an open source software package available free of charge from the following World Wide Web site:

http://grey.colorado.edu/emergent/

While *emergent* is fully extendible through the incorporation of user written C++ code, no knowledge of computer programming is needed in order to make basic use of the system. A graphical user interface, involving pull-down menus, buttons, and the like, is provided, and this is the standard manner in which modelers interact with the *emergent* tools.

The *emergent* programs were originally developed for computers running Unix®-like operating systems, and support continues to be most robust on these platforms. The package has been ported, however, to Microsoft® Windows®, There has also been a port to Mac® OS X, and much of the recent development on *emergent* has been performed on the Mac® platform.

Computer simulation exercises using *emergent* draw extensively from the main text to be used in this class. While all of these exercises offer opportunities for learning, the instructor will identify a subset as particularly relevant. These exercises will help the student investigate general computational modeling mechanisms using this particular package of simulation tools. The main text offers much guidance for conducting these simulations, as does online documentation for the exercises, but there are additional sources of information on the fruitful use of the *emergent* software. Primary among these is the online documentation available from the *emergent* World Wide Web site. Further information on obtaining, installing, and using *emergent* may be found on the the official *emergent* web site.

**Laboratory**

The *emergent* software system has been installed on the computers that will be available during class laboratory sessions. During early meetings in this laboratory, students will be guided in the use of *emergent* in this operating environment. The computers in this facility run the Ubuntu Linux® operating system, which includes a standard graphical user interface for common operations. Students will be expected to learn, with the assistance of the teaching team during laboratory meetings, to navigate this environment sufficiently well to complete assigned computer simulation exercises. Documentation discussing use of this environment may be found at https://help.ubuntu.com/.
While scheduled laboratory sessions are expected to be of sufficient length to allow course participants to complete the regular exercises that will be assigned during the semester, additional access to emergent may be needed for students to complete their term projects and to conduct other self-guided investigations into computational cognitive neuroscience. Thus, students are encouraged to install emergent on computers located in their usual work spaces, either in their offices or in their homes, allowing them to explore CCN modeling methods in the comfort of their standard work environments.

Expectations & Evaluations

Background

Course participants are expected to have some background knowledge in cognitive science, computer science, psychology, or neuroscience and a basic understanding of the most fundamental concepts of differential calculus, linear algebra, and statistics. Computer programming skills may be useful to some students as they conduct their term projects, but such skills are certainly not required. As previously mentioned, the emergent software system which will be used in this course allows for the design and control of simulations through a rich “point and click” interface. Students are only expected to be sufficiently familiar with an appropriate computing environment so as to be able to perform such tasks as editing and managing files and manipulating objects in a window-based graphical environment.

The University of California, Merced, is committed to ensuring equal educational opportunities for students with disabilities. An integral part of this commitment is the coordination of specialized academic support services through the Disability Services office. Students with a physical or learning disability may ask Disability Services to assist in communicating this fact to the instructor so that appropriate accommodation may be provided. Absent notification, the instructor may assume that no such accommodation is sought.

Participation

Studying the neural basis of cognition using computational models is a challenging interdisciplinary endeavor requiring familiarity with notions from artificial intelligence, statistics, psychology, and neuroscience. Thus, students may find this class both rewarding and demanding. Mastery of the course materials will require extensive reading, puzzling through unfamiliar concepts, active participation in classroom discussions, learning new mathematical formalisms, many hours of hands-on experience building and analyzing computer simulations, and a willingness to view human cognition in new ways.

Specific readings will be suggested as appropriate for each class meeting, and participants will be expected to have studied those readings prior to gathering, so as to promote thoughtful questions and knowledgeable discussion. Students will be expected to contribute constructively to discussions, bringing to bear both insights into the material at hand and relevant knowledge acquired in other contexts.

This class is structured so as to leverage the variety and wealth of knowledge that the participants will bring to this learning enterprise. Interaction during class meetings should be seen as an important goal. Participants are strongly encouraged to investigate connections between course topics and their own fields of study and to bring insights from such explorations with them to class meetings. Indeed, such contributions will be expected.

Computational methods for cognitive neuroscience are best learned through active experimentation. Thus, a number of exercises will be assigned to help students acquire an understanding of CCN concepts and techniques and to aid in the evaluation of their understanding. These exercises will require substantial time and effort to complete, and they will involve the use of the emergent simulation software. Student solutions to these assignments will be evaluated by the teaching team, and appropriate feedback will be given.

Near the end of the term, each enrolled student will be expected to prepare a short research report (about 10–15 pages) describing the student’s use of computational modeling techniques to advance our understanding of some phenomenon of the cognitive neurosciences. This final paper should provide the student with an opportunity to extend the course material in a direction of personal interest and should demonstrate an understanding of key CCN concepts in some interesting way. Minimally, term projects should extend a computational model presented during the semester or analyze a presented model in a new way. In addition to producing written reports describing the work conducted...
as part of these term projects, all enrolled students will also be asked to share their findings with the other course
participants through brief presentations to be held during the course’s final meetings.

Class participants are expected to embrace the course material with earnest effort, to contribute constructively
to the learning of other students, and to always behave ethically and with civic concern. Students should come to
every class meeting prepared to discuss relevant topics. Exercises are to be completed by their respective due dates.
The ideas and contributions of others should be appropriately cited. (This includes ideas and contributions garnered
from readings, online resources, presentations, conversations, and any other source.) Students are expected to bring
educational obstacles to the instructor’s attention as early as possible, so that such problems may be promptly resolved.

Learning can be greatly facilitated by interactions between class participants, and these interactions are encour-
aged. Students should feel free to discuss lecture topics, readings, project ideas, and even exercise assignments with
each other. The actual completion of project work and exercises, however, should be conducted on an individual basis.
All assignments submitted for evaluation should reflect the understanding and effort of the individual participant. If
there is ever any doubt concerning the propriety of a given interaction, it is the student’s responsibility to approach
the instructor and clarify the situation prior to the submission of work results. Also, helpful conversations with fellow
students, or any other person (including members of the teaching team), should be explicitly mentioned in submitted
assignments. Failure to appropriately cite sources is one form of plagiarism, and it will not be tolerated!

Evaluation
The teaching team will provide comments on all assigned work submitted in a timely manner for evaluation. Those
students who are to receive grades for this course will have their work assessed roughly as follows:

<table>
<thead>
<tr>
<th>Assignment</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exercises #1 – #11</td>
<td>5% each</td>
</tr>
<tr>
<td>Project Oral Report</td>
<td>5%</td>
</tr>
<tr>
<td>Project Written Report</td>
<td>30%</td>
</tr>
<tr>
<td>Class Participation</td>
<td>10%</td>
</tr>
</tbody>
</table>

Student performance will be evaluated in comparison to that of other students, both past and present. Class participa-
tion will be closely monitored throughout the term.

Exercises will be assigned each Friday and will be due one week from that Friday, at 9:00 A.M. on that day. Late
assignments which arrive in the instructor’s hands before 9:00 A.M. on the day after a due date (Saturday) will be
evaluated and will receive 90% of the credit for the assignment. Late assignments which arrive in the instructor’s
hands before 9:00 A.M. on the subsequent day (Sunday) will be evaluated and will receive 80% of the credit for the
assignment. Assignments which are submitted later than this will not be evaluated, and no credit will be given.

Final written project reports will be due to the instructor by 11:30 A.M. on Friday, May 16th.

Schedule
In the schedule that appears on the following pages, a check mark (√) identifies an assigned reading for the date in
question. Other readings may be considered supplementary and optional. Those marked with a diamond (♦) may
be found in the CECN textbook, in the PDP volumes, in Neurocomputing, in Theoretical Neuroscience, or in online
emergent documentation. Other papers, marked with a heart (♥) do not appear in any of these sources and will not
be provided by the instructor.
Introduction

January 21: Introduction

- Syllabus Distributed

January 23: Historical Background

√ CCN, Chapter 1.
♦ CECN, Forward & Chapter 1.
♦ CECN, Appendix A.

Neural Activity

January 28: Individual Neurons — The Membrane Potential

√ CCN, § 2.1–2.4.1.
♦ CECN, Chapter 2.
♦ Theoretical, § 5.1–5.5.
January 30: Individual Neurons — The Action Potential

- Exercise #1 Specification Distributed On Friday, January 31
  ✓ CCN, § 2.4.2–2.4.3, 2.5–2.9 (§ 2.7 is optional).
  ◊ Theoretical, § 7.1–7.4.


February 04: Distributed Representations

- CCN, § 3.1–3.3.
- CECN, § 3.1–3.4.

February 06: Inhibition & Attractor Dynamics

- Exercise #1 Due At 9:00 A.M. On Friday, February 07
- Exercise #2 Specification Distributed On Friday, February 07
  ✓ CCN, § 3.4–3.7.
  ◊ CECN, § 3.5–3.8.
  ◊ Theoretical, § 7.5.

February 11: Mathematics Review

◊ Theoretical, § A.1, A.3, & A.5.
◊ Theoretical, § 8.1–8.2.

Synaptic Plasticity

February 13: Unsupervised Learning & Self-Organization

• Exercise #2 Due At 9:00 A.M. On Friday, February 14
• Exercise #3 Specification Distributed On Friday, February 14

√ CCN, § 4.1–4.2.2.
√ CCN, § 4.5 (Hebbian Learning).
√ CCN, § 4.5 (STDP).
◊ CECN, Chapter 4.
◊ Theoretical, § 10.1–10.2.
February 18: The Delta Rule

√ CCN, § 4.2.3–4.2.3.1.
◊ CECN, § 5.1–5.5.

February 20: The Generalized Delta Rule

• Exercise #3 Due At 9:00 A.M. On Friday, February 21
• Exercise #4 Specification Distributed On Friday, February 21
√ CCN, § 4.5 (Backpropagation).
◊ CECN, § 5.6.
◊ Theoretical, § 8.4.
February 25: Special Network Architectures


◊ CECN, § 6.5–6.6.


February 27: Leabra Learning

- *Exercise #4 Due At 9:00 A.M. On Friday, February 28*

- *Exercise #5 Specification Distributed On Friday, February 28*

√ CCN, § 4.2.3.2–4.7.

◊ CECN, § 5.7–5.11.

◊ CECN, § 6.1–6.4.
Large-Scale Functional Brain Organization

March 04: Cognitive Architecture

✓ CCN, Chapter 5.
✓ CECN, Chapter 7.

March 06: Designing Simulation Projects

• Exercise #5 Due At 9:00 A.M. On Friday, March 07
• Exercise #6 Specification Distributed On Friday, March 07
✓ CECN, Appendix B.

Perception & Action

March 11: Visual Perception

✓ CCN, § 6.1–6.3.
✓ CECN, § 8.1–8.4.

March 13: Visual Attention

• Exercise #6 Due At 9:00 A.M. On Friday, March 14
• Exercise #7 Specification Distributed On Friday, March 14
✓ CCN, § 6.4–6.7.
✓ CECN, § 8.5–8.8.
March 18: Reinforcement Learning

√ CCN, § 7.1–7.2.
◊ CECN, § 6.7–6.9.
◊ Theoretical, Chapter 9.

March 20: Motor Control

• Exercise #7 Due At 9:00 A.M. On Friday, March 21
• Exercise #8 Specification Distributed On Friday, March 21
√ CCN, § 7.3–7.6.

March 25: Spring Break

• No Meeting

March 27: Spring Break

• No Meeting

Memory

April 01: Priming

√ CCN, § 8.1–8.1.1, 8.2–8.3.
◊ CECN, § 9.1–9.2.
April 03: Episodic Memory

- **Exercise #8 Due At 9:00 A.M. On Friday, April 04**
- **Exercise #9 Specification Distributed On Friday, April 04**

✓ CCN, § 8.1.2–8.1.8, 8.4–8.6.
◇ CECN, § 9.3.

Language

April 08: Reading, Morphology, & Mental Rules

✓ CCN, § 9.1–9.3.
◇ CECN, § 10.1–10.5.
April 10: Semantics, Syntax, & Sentence Processing

- Exercise #9 Due At 9:00 A.M. On Friday, April 11
- Exercise #10 Specification Distributed On Friday, April 11

√ CCN, § 9.4–9.9.
♦ CECN, § 10.6–10.9.

Higher-Level Cognition

April 15: Cognitive Control

√ CCN, § 10.3–10.4.
♦ CECN, § 9.4–9.9.
♦ CECN, § 11.1–11.3.

April 17: Cognitive Flexibility & Working Memory

- Exercise #10 Due At 9:00 A.M. On Friday, April 18
- Exercise #11 Specification Distributed On Friday, April 18

♦ CECN, § 11.4–11.8.


April 22: Cognitive Development


April 24: Computational Modeling Challenges

Exercise #11 Due At 9:00 A.M. On Friday, April 25

CECN, Chapter 12.


April 29: Project Oral Reports

May 01: Project Oral Reports

May 06: Project Oral Reports

May 08: Project Oral Reports

May 16: Written Project Reports Due

Reports must be in the instructor’s hands by 11:30 A.M. on this day.