Human-directed Approaches to Computer Systems Problems

Syllabus

Web Page
http://www.cs.northwestern.edu/~pdinda/hdsys

Instructor
Peter A. Dinda
Technological Institute, Room L463
847-467-7859
pdinda@northwestern.edu
Office hours: Thursdays, 2-4pm or by appointment
I also respond quickly via email
I am often available via Google Talk (pdinda at gmail dot com)
There is a course newsgroup (see below)

Location and Time
We will meet in TECH MG28 for the first week
For the next 9 weeks, we will meet in FORD 3-340 (conference room)
Tuesdays and Thursdays, 9:30-10:50am

Prerequisites
There are no formal prerequisites for this course at this time, although the following are noted:

• Basic knowledge of computer systems, to the level of EECS 213 is helpful. We will review the hardware/software stack in the first week of classes
• Some experience with human-computer interaction, such as in EECS 330, is helpful.
• Programming skills in an imperative language such as C, C++, Perl, or Python will helpful. Programming will be needed for some of the possible projects.
• Some familiarity with GUI programming on Windows, Unix/Linux, Mac, Java, or a toolkit is helpful. GUI programming will be needed for some of the possible projects.
• Familiarity with basic statistics is assumed.
As this is the first iteration of this course, I will try to adapt the course to the skills of the students.

Readings

Readings for the course will be in the form of research papers and other materials. For the most part, these materials are available online. I will hand out paper copies of those that are not. A separate reading list is provided. It also lists several reference texts that may be helpful.

Objectives, framework, philosophy, and caveats

Recently, human-centered approaches have made significant inroads well outside of the traditional area of human-computer interaction. Examples include CAPTCHAs in computer security, and “distributed human computation” for solving AI-complete problems such as in the Google Image Labeling Game.

This research-oriented course, suitable for graduate students and advanced undergraduates, explores the application of human-centered approaches to solving problems in systems software, networks, and architecture. The design and engineering decisions made at these layers of computing are typically considered quite independently of end-users. However, these layers have dramatic effects on the user experience. Work at Northwestern and elsewhere demonstrates there is massive potential in applying human-centered approaches to computer systems problems. This course will explain the state of the art, how to evaluate it, and will give students the opportunity to expand it.

The overall thinking and motivations behind this course are given in much more depth in the following position paper:


We will explore the following topics in this course:

- Review of computer systems software and hardware, and their impact on the user experience
- Human-centered computing as applied to systems
- Results outside of systems: CAPTCHAs, distributed human computation, games with a purpose, etc
- Results within systems: user-driven power management, user-driven scheduling, speculative remote display, user comfort with resource borrowing, distributed scheduling games, others
- How interface issues in systems are different from those in applications
- User-driven specification and user-driven search
• Role of learning
• Open applications of concepts
• Open research questions

When a student completes this course, he or she should be able to:
• Understand the concept of human-centered computing as applied to computer systems
• Understand the aspects of computer systems that impact the user experience
• Formalize computer systems problems that may be aided through human-centered computing
• Give examples of user-driven specification and user-driven search
• Understand the issues of interface design that arise in systems that are different from or in addition to those in applications
• Design and develop human-centered approaches to systems problems
• Understand how to evaluate, and how to judge the evaluation of, human-centered approaches to systems problems
• Have familiarity with the existing literature on the application of human-centered computing to computer systems, and how to evaluate it
• Have experience in a research project in the area, and how to document it in an effective conference paper and presentation.

This course is structured as a graduate research seminar, with undergraduates welcome. It combines extensive reading, presentation, and discussion of original research papers and a quarter-long research project in the area that leads to a conference-style paper and presentation.

Almost all of the readings for the course will be in the form of research papers, and similar materials. We will generally read several papers or equivalent materials each week, covering fundamental ideas and important recent results. Each paper will be formally presented to the group by a student and then discussed in a round-table manner. Before each class, each student is expected to write up his/her comments on the course discussion group for access by all class members.

Student projects, which can be done individually or in groups, should strive to be original research, conducted with guidance from me. Each project will lead to a conference paper and a presentation. I want students to be very ambitious!

**Course Discussion Group**

The course web page includes a sign-up form and URL for the course discussion group. This is a Google Group that is accessible only to the members of the class.
Before each class, students are expected to read the papers, and post a summary and comments to the course discussion group. These posts, which will be accessible to everyone in the class, are required, and will form part of your grade.

Tools

In addition to standard resources (TLab, VLab, Wilkinson Lab, Pocket PCs), students will have access to other tools from my research lab on an as-needed basis, and I will purchase other tools if needed.

Project

Over the course of the quarter, you will apply what you learn to a project of your choice, and then document your project in a high quality conference-style paper, similar to those you’ll be reading, and a 20-25 minute presentation. If you’re an undergraduate or a new graduate student, this process will expose you to doing research, and documenting its results in a professional way. It is not uncommon for good work in a graduate class like this to lead to a paper submission to a high quality workshop or conference, and/or to a longer-term research project.

Students can work individually or in groups. I will expect a project proposal, and weekly progress reports in addition to the final paper and presentation. Ideally, we will meet once a week to discuss your project.

Exams

There will be no exams

Grading

40 % Project, including proposal and weekly progress reports
20 % Project paper and presentation
20 % In-class paper presentations of papers
20 % General classroom participation and comments on discussion group

Schedule

Week 1 Introduction

Tuesday, 1/8 Introduction
Review of how computer systems work

Efficient Reading of Papers in Science and Technology:
http://www.cs.columbia.edu/~hgs/netbib/efficientReading.pdf  (very short!)
EECS 213 textbook (Bryant and O’Hallaron, *Computer Systems: A Programmer’s Perspective*) may be handy if you’ve never taken a systems course before


Also take a look at Lampson, *Hints for Computer System Design* if you haven’t read it before.

Thursday, 1/10  Introduction - the overall ideas in the class


Week 2  Outside of Systems

Tuesday, 1/15  CAPTCHAs


Also be sure to check out captcha.net and recaptcha.net

Thursday, 1/17  Human Computation


Week 3  Measuring the User

Tuesday, 1/22  Measuring the Desktop User


A. Komatsubara, *Psychological upper and lower limits of system response time and user's preference on skill level*, HCI International 1997

**PROJECT PROPOSAL DUE AT END OF THIRD WEEK**

Thursday, 1/24  Measuring the Web User


Week 4  Modeling the User

Tuesday, 1/29  Workload modeling and prediction


Thursday, 1/31  Workload modeling and prediction


P. Gorniak, D. Poole, *Predicting Future User Actions by Observing Unmodified Applications*, AAAI 2000

Week 5  Speculation

Tuesday, 2/5  Speculative execution and display
“Wild and Crazy Ideas” Session (WACI)

J. Lange, P. Dinda, S. Rossoff, *Experiences with Client-based Speculative Remote Display*, DRAFT

Thursday, 2/7  (likely out of town at V3VEE board meeting)

SLACK TIME FOR NOW

**Week 6**  User-driven Scheduling

Tuesday, 2/12  Resource Management


Thursday, 2/14  Resource Management


**Week 7**  User-driven Power Management

Tuesday, 2/19  Power management


Thursday, 2/21  Power management


L. Yan, L. Zhong, N. Jha, User-perceived Latency based Dynamic Voltage Scaling for Interactive Applications, DAC 2005

Week 8  Evaluating User-based Systems

Tuesday, 2/26  Example user-oriented evaluations of systems


Thursday, 2/28  Challenges in user studies

Reading: TBD

Week 9  Security and Visualization

Tuesday, 3/4  Visualizing hashes and keys


Thursday, 3/6  Visualizing computer systems and security

Genetic Art for Intrusion Detection Project: http://ga-ids.cs.northwestern.edu

Week 10 Brainstorming/Crazy Ideas

Tuesday, 3/11
Thursday 3/13

Is there a grand unified abstraction for what users want?
Do emotions matter for empathic systems software?

Your questions here and paper suggestions here…


PROJECT DUE LAST DAY OF FINALS WEEK

Project Presentations In Finals Week Final Exam Slot