Spring 2021 – CS Special Topics

1. CS 494* – Eriksson – Safe and Efficient Systems Programming in Rust
2. CS 494* – Hummel (BioE/CS) – Wearable Technologies
3. CS 494* – Kshemkalyani – Parallel and Distributed Processing
4. CS 494* – Pina – Principles of Concurrent Programming
5. CS 594 – Di Eugenio – Natural Language Processing for Social Media
6. CS 594 – Miao – Secure Computation
7. CS 594 – Tang – Deep Learning for Computer Vision
8. CS 594 – Vamanan – Advanced Topics in Wireless Networking Research

*CS Undergraduate students must submit a modification of major to use the class as a technical elective.

Spring 2021 – CS Special Topics

CS 494– Safe and Efficient Systems Programming in Rust

- Instructor: Jakob Eriksson
- Meeting time: TR 11a-12:15pm (synchronous)
- CRN: 42279/42290

Course Description:
This course explores safe low-level systems programming, in the context of the Rust programming language. Here, systems programming refers to the development of systems software, such as operating systems, garbage collectors and memory managers, debuggers, network protocols and more. Since the 1970's, C, and its cousin C++, have been the preferred language for systems programming. Rust is a relatively young language, and is one of an only very small number of candidates to eventually replace C/C++ in this role.

While Rust is much appreciated by its users, having been voted "most loved language" on Stackoverflow four years in a row, it is also well known to have a steep learning curve. In this class, we face this learning curve together, in the context of several pieces of key systems software built in Rust.

Topics Covered (tentative):
Syntax
Mutability
Ownerships and Borrowing
Structs
Enums and Pattern matching
Crates, Packages and Modules
Writing Tests
Common Collections
Error Handling
Generics and Traits
Lifetimes
Closures
Smart Pointers, Drop and Memory Management Concurrency: Send, Sync and Mutual Exclusion
Concurrency: Message Passing with Channels Unsafe Rust

Assignments and evaluation:
There will be 6 programming assignments, accompanied by 6 small midterms, as well as a comprehensive final exam.

Prerequisites:
CS361 or equivalent (for grad students), or instructor consent.

Spring 2021 – CS Special Topics

CS 494– Wearable Technologies

- Instructor: Joe Hummel (CS) and Hananeh Esmailbeigi (BioE)
- Meeting time: W 10a-12:50pm (synchronous)
- CRN: 42271/43270

Course Description:
This course, taught primarily by BioE, is a lab-based course in the design and construction of wearable devices. Students work in teams of 4, 2 from CS and 2 from BioE, and use HW and SW skills to design, construct, and analyze wearable devices and the collected data. The class is run like a true lab, where class meets once/week for 3 hours, and class time is devoted mostly to project work. If necessary, projects are completed outside of class time.

Coursework:
3 assigned projects + a final project of team's own design. Each project involves HW and SW integration, some circuit design, data analysis, and a team report. Team work required.

Prerequisites:
CS 251 and 261, and CS 362 if possible.

Spring 2021 – CS Special Topics

CS 494– Parallel and Distributed Processing

- Instructor: Ajay Kshemkalyani
- Meeting time: MW 4:30-5:45pm (synchronous)
- CRN: 42280/42281
- Web site: https://www.cs.uic.edu/~ajayk/c494sp21.html

Method of Instruction:
The class will consist of lectures on the foundations of parallel and distributed processing.

Narrative Description:
This class explores the foundations of parallel and distributed computing. Emphasis will be laid on parallel and distributed algorithms, for basic topics ranging from graph algorithms to mutual exclusion, deadlock detection, and predicates detection algorithms, to design of communication algorithms for elementary primitives such as multicast, broadcast, scatter, gather, all-to-all personalized communication on distributed systems as well as clusters based on specific topologies. Basic concepts such as scalability of systems (i.e., combination of architecture plus algorithm) will also be explored. As a case study, the Message-Passing Interface (MPI) will be introduced, and students will be exposed to programming using MPI on UIC's Extreme cluster. Ultimately, the goal is to foster an understanding of the various ways in which concurrent actions in the system and the lack of global knowledge in the system can affect the design of parallel and distributed algorithms.

Student Evaluation and Deliverables:
There will be two midterms. Students will be expected to complete about 2 or 3 programming assignments using MPI (Message-Passing Interface) on UIC's Extreme cluster. There will also be a theory-based project/term paper.

Prerequisites:
Programming skills amenable to programming the UIC Extreme cluster, in a language such as C, is desired. CS 401 is suggested but not required.

Course Outline:

- Distributed graph algorithms (for traversals, spanning trees, collaborative exploration, routing, MIS, synchronizers, etc) *(3 weeks)*
- Mutual exclusion algorithms, deadlock detection algorithms, termination detection algorithms, leader election algorithms *(3 weeks)*
- Predicate detection algorithms *(1 week)*
- Total order and causal order multicast *(1 week)*
- Basics of cluster design *(1.5 weeks)*
- Algorithms for implementing basic communication primitives on topologies *(2 weeks)*
- Scalability analysis of parallel systems *(1 week)*
- Search algorithms and dynamic load balancing for discrete optimization *(1.5 weeks)*

CS 494– Principles of Concurrent Programming

- Instructor: Luis Pina
- Meeting time: MW 3:00-4:15pm (synchronous)
- CRN: 42278/42289

This course examines the theory and practice of writing correct concurrent programs. Subjects covered include multiprocessor architecture, mutual exclusion, wait-free and lock-free synchronization, memory models (Javascript, Python, Java, TSO, x86, ARM), language support for concurrent programming (spin locks, monitors, thread pools, event loops), concurrency models (shared memory, message passing, actors, fork-join), and transactional synchronization. The course illustrates the concepts with practical applications through fundamental concurrent data structures, such as lists and queues. The course will also explain mechanisms to debug concurrent programs: Deadlock detection, data race detection, partial order reduction, among others.

This course will be focused on the foundations and basic principles of concurrent programming: What machines need to provide to ensure that concurrent programs behave as expected. At the end of the course, students will have a basic understanding of both the foundations and the practice of concurrent programming.

Spring 2021 – CS Special Topics

CS 594– Natural Language Processing for Social Media

- Instructor: Barbara Di Eugenio
- Meeting time: TR 12:30-1:45pm (synchronous)
- CRN: 33648

Course Description:

In the last 15 years, online social networking has revolutionized communication (e.g., Facebook was founded in 2004, and Twitter, in 2006); social media has had an enormous impact on society, and on our personal and professional lives. Since most of this data uses language (and increasingly pictures and videos), natural language processing (NLP) is well positioned for social media data processing. However, traditional NLP methods do not work well for social media, which is often much more informal and "free flow" than traditional documents; in addition, much social media is asynchronous like written communication, but engenders interaction like a face-to-face conversation does. Orthogonally to these difficulties, traditional NLP tools themselves are currently being retooled with the help of deep learning.

This course will provide students with scientific foundations in NLP algorithms, approaches and evaluation techniques as they relate to social media language; insight into the many remaining research issues; and hands-on experience in processing social media language.

Materials for the course will come from a variety of sources that have become recently available. The interest in NLP for social media is attested by a book Farzindar and Inkpen (2020), part of the Synthesis Lectures on Human Language Technologies series by Morgan & Claypool, now in its third edition, the last of which just appeared in 2020; a special issue of the Computational Linguistics journal, from December 2018 Benamara et al. (2018); a series of 8 workshops on NLP for Social Media of the Association for Computational Linguistics (ACL), the 8th of which occurred in 2020. Papers on NLP for social media now regularly appear in all the major NLP conferences (ACL, NAACL, EMNLP, LREC), and journals (CL, TACL, NLE).

Course Outline (Tentative):

Please note: the references to the sections and chapters in the Farzindar and Inkpen (2020) book will be supplemented by additional references and research papers on the topic at hand. Farzindar and Inkpen (2020) is not a textbook, but rather, a succinct compendium of research on the specific topic.

Week	Topic	Readings
	Weeks 1-7: Foundations	
Week 1	Introduction to social media analysis	Ch. 1 Farzindar and Inkpen (2020)
Week 2	Data collection, Ethics & corpora	Ch. 5 Farzindar and Inkpen (2020)
Week 3	Tokenizers, POS taggers	Sec 2.3, 2.4 Farzindar and Inkpen (2020)
Weeks 4-5	Parsers, NER	Sec 2.5, 2.6 Farzindar and Inkpen (2020)
Week 6	Event Detection	Sec 3.4 Farzindar and Inkpen (2020)
Week 7	Opinion Mining, Sentiment Analysis	Sec 3.4 Farzindar and Inkpen (2020)
Week 8	Context Interpretation	Papers in Benamara *et al.* (2018)
Week 9	Multimodality	Research Papers

Spring 2021 – CS Special Topics

	Weeks 10-14: Applications	Research papers throughout
Week 10	Political discourse & Media monitoring	
Week 11	Hate speech, bullying, bias	
Week 12	Rumor detection and "fake news"	
Week 13	Applications for Health and Medicine	
Week 14	Multilingual applications	
Week 15	Project Presentations	

Course Work:

1. Reading, discussions, presentations of research papers
2. Two-three small assignments to try out existing tools
3. Course project

Book and readings:

Recommended: Farzindar and Inkpen (2020). 80% research papers taken from a variety of sources, including the major NLP conferences (ACL, NAACL, EMNLP, LREC), and journals (CL, TACL, NLE).

Prerequisite / Corequisite:

421 or 521 (or another equivalent NLP class); 582 Information Retrieval, 583 Data Mining and Text Mining; Deep Learning for NLP (offered so far as 594, approved as a permanent course in Spring 20 with the number, 533).

References:

Farah Benamara, Diana Inkpen, and Maite Taboada, editors. *Special issue on language in social media: Exploiting discourse and other contextual information, volume 44 of Computational Linguistics*. MIT Press, 2018.

Atefeh Farzindar and Diana Inkpen. *Natural language processing for social media (Third Edition), volume 46 of Synthesis Lectures on Human Language Technologies*. Morgan & Claypool Publishers, 2020.

CS 594– Secure Computation

- Instructor: Peihan Miao
- Meeting time: R 6:30-9pm (synchronous)
- CRN: 42282

Course Description:
Nowadays, vast amounts of individual information are collected by numerous entities, which fuels a wide range of personalized services. However, since the collected data may contain sensitive private information, due to regulatory compliance or corporate policies, securely computing on the collected data in a privacy-preserving yet efficient way becomes significantly important.

This course is an introduction to secure computation, which achieves the above goal, providing great promise and potential to facilitate research in various domains including healthcare, biomedicine, education, artificial intelligence, etc. Though the topic of secure computation grew out of the cryptography community, there is now increasing interest in this area from a lot of other communities.

We will cover both the theoretical foundations and practical implementations of secure computation. Students will be able to understand cryptography research papers and initiate research studies in this area.

Main Topics:

- Secure two-party and multi-party computation (MPC), including both theoretical approaches and practical implementations.
- Oblivious RAM and efficient secure RAM computation.
- Secure MPC design from advanced cryptographic tools such as fully homomorphic encryption (FHE), indistinguishability obfuscation (iO), and blockchain.

Grading:

- Regular attendance and active participation in class 20%
- Subscribe lecture notes (in latex) 30%
- Final project presentation and report 50%

Prerequisites:
CS 401 or equivalent.
No prior background in cryptography is required. However, mathematical maturity is required.

CS 594– Deep Learning for Computer Vision

- Instructor: Wei Tang
- Meeting time: Sat 1:00-3:30pm (synchronous)
- CRN: 33792

Course Description:
Computer vision is an important technology today because of the need to understand a large number of images and videos to improve human welfare. Although many traditional models have been developed for computer vision applications, recent deep learning methods have achieved remarkable results in almost all computer vision tasks. The purpose of this course is to introduce students to deep learning research applied to computer vision. The course will cover topics such as convolutional neural networks, recurrent neural networks, long short-term memory networks, generative adversarial networks, and attention models. Through lectures, projects and presentations, students will learn the latest techniques in applying and designing deep neural networks to solve practical computer vision problems. Students will also understand the research breadth and specific interests of the computer vision community.

Topics:
The course aims at exposing students to recently developed deep learning methods to solve computer vision problems that can lead to various emergent applications.

1 Introduction to computer vision and deep learning (1 week)
• Definition, motivation, challenges and applications of computer vision
• A brief history of deep learning and AI

2. Convolutional neural networks for image recognition (4 weeks)
• Neural networks and back-propagation
• Convolutional neural networks
• Applications: image classification, detection, segmentation, pose estimation, face recognition

3. Recurrent neural networks for video recognition (3 weeks)
• Recurrent neural networks
• Long short-term memory networks
• Applications: action recognition, image captioning and text recognition

4. Generative adversarial networks for image generation and synthesis (3 weeks)
• Generative adversarial networks
• Applications: image enhancement and restoration, image synthesis, and image translation

5. Attention models (3 weeks)
• Spatial attention, channel attention, self-attention and non-local networks
• Applications: image and video recognition, vision and language, and synthesis

Grading:
Project Presentation: 50%
Project Report: 50%

Prerequisites:
Linear algebra and calculus, computer vision

Spring 2021 – CS Special Topics

CS 594– Advanced Topics in Wireless Networking Research

- Instructor: Balajee Vamanan
- Meeting time: TR 2:00-3:15pm (synchronous)
- CRN: 33649

Course Description:
People are starting to rely more on wireless networks (e.g., Wi-Fi, cellular) for their everyday tasks. Further, recent trends such as Internet-of-Things (IoT), and Connected Cars rely heavily on wireless networks. However, the dynamics of the communication in a wireless medium are qualitatively different and necessitate optimizations across the networking stack. This course exposes graduate students to various wireless networks, protocols, scheduling mechanisms used, and network infrastructures (e.g., cellular base stations).
The course is beneficial for graduate students who are pursuing research in networks and systems. Students will have an opportunity to work with network emulators such as PhantomNet, which uses a software-defined radio, to implement their own protocol enhancements on a real network. The key takeaway is an understanding of the trade-offs involved in designing protocols and systems for wireless networks by getting exposed to recent research developments in this area.

Method of Instruction:
This class will involve lectures, paper discussions, and a collaborative research project. The class will be divided into modules (e.g., Wi-Fi, Cellular, 5G). Each module will begin with lectures, followed by a discussion of recent, impactful papers. Students will be asked to present papers, lead a discussion, and address shortcomings. The collaborative project is an opportunity to identify a well-defined problem under my guidance and work toward a workshop-level paper.

Student Deliverables:
Students will be evaluated on three aspects: (1) summarizing papers before the actual paper discussion, (2) presenting papers, and (3) course project. Student will be expected to summarize papers under discussion and provide their opinion about novelty, evaluation, and weakness. Each student will also be expected to present 1–2 papers and lead the discussion. The course project will be evaluated based on the idea, methodology, thoroughness of evaluation, and the quality of the project report.

Prerequisites:
• For students who have not done CS450 or equivalent, instructor consent is required.
• Working knowledge of programming in C, C++ is required.

Exams:
There will be no exams for this course.

Spring 2021 – CS Special Topics

Back to List

Tentative Topics

- Wi-Fi networks
    - Different 802.11 protocol variants
- Cellular networks
    - 3G
    - 4G, WiMax, LTE
    - 5G
- Interaction between transport layer protocols and wireless media
    - BBR, TCP-NICE
- Software-defined networking and Network Virtualization in cellular networks
- Internet of Things
- Ad-hoc Mesh networks