Spring 2018
CS Special Topics Classes

Click on topic to go description of class.

1. CS 491 - Network Security
    - Instructor: Prof. Venkatakrishnan
    - Meeting time: TR 2-315
    - CRNs: 41113 (ugrad) & 41122 (grad)
    - Notes: Undergrads have to submit modification of major to use this as technical elective

2. CS 594 - Cyber-Physical Systems: Correctness and Safety
    - Instructors: Prof. Sistla & Prof. Zefran
    - Meeting time: TR 2-315
    - CRN: 33649

3. CS 594 - Geometric Algorithms for Data Analysis
    - Instructor: Prof. Sidiropolous
    - Meeting time: MW 4-515
    - CRN: 34724

4. CS 594 – Web Security and Privacy
    - Instructor: Prof. Polakis
    - Meeting time: MW 5-615
    - CRN: 33648

5. CS 594 – Advanced Computer Networking
    - Instructor: Prof. Vamanan
    - Meeting time: TR 330-445
    - CRN: 33792

6. CS 594 – Advanced Machine Learning
    - Instructor: Prof. Zhang
    - Meeting time: TR 5-615
    - CRN: 38551
    - Notes: This class will become CS 512, so will count as a regular CS 5xx class

7. CS 594 – Artificial Intelligence: Methods and Applications
    - Instructor: Prof. Petrov
    - Meeting time: R 630-0900
    - CRN: 41289

CS491: Network security

Instructor: Venkat Venkatakrishnan

Topics:

I. Introduction
- Overview and logistics
- Network Security threats

II. Principles
- Mathematical Foundations
- Principles of Communication Security
  - o secret-key cryptography
  - o public-key cryptography
  - o secure hash functions and random numbers
  - o digital signatures

III. Network security infrastructure
- Trust and Trust Failures
- Routing and Naming: DNSSEC, S-BGP
- Public Key Infrastructure: X.509
- Authentication (Kerberos)

IV. Security of Protocols
- Security issues with the TCP/IP suite
- IPSec, SSH
- Wireless security

V. Network Security Applications
- E-mail security (PGP, S/MIME)
- Cloud Security
- TLS

VI. Other Network Defenses
- Anonymity, traffic analysis resistance and Tor
- Firewalls
- Network Intrusion Detection
- Distributed Denial of service (DDoS)

VII. Advanced Topics
- Botnets
- Elliptic Curve Cryptography

VIII. Wrap-up

Textbook:

William Stallings, Network Security Essentials: Applications and Standards (6th Edition), ISBN: 013452733X.

Prerequisites:
CS450 (introduction to networking, or equivalent) is required as a pre-requisite or co-requisite, or instructor consent.

Evaluation:
There will be one final exam. Several programming assignments as well as written homework will be part of the continuous evaluation.

- Programming Assignments 45%
- Other homework – 20%
- Final - 35%

CS/ECE 594 Special Topics
Cyber-Physical Systems: Correctness and Safety

Instructors: Prasad Sistla and Miloš Žefran

Course Description Cyber-physical systems (CPSs) involve computation, communication and control. Their applications include automotive systems, energy, robotics, medical devices and transportation. The course will introduce students to techniques for ensuring correct and safe functioning of cyber-physical systems. It will focus on formal methods for modeling CPSs and their safety properties, and static as well as run-time verification of such systems.
The course brings together techniques and tools from dynamical systems, automata theory, formal verification and control. The purpose of the course is to introduce the students to advanced research on cyber-physical systems while giving them a more general background on modeling of stochastic systems with discrete and continuous state, as well as formal verification and state estimation for such systems. Throughout the course, application examples will be used to motivate the topics studied. The students will also be introduced to challenges in guaranteeing safety for systems that rely on machine learning where explicit models of system operation are not available, and data-driven approaches to system modeling.

Course Organization. The first 7 weeks of the course will be lecture-based; students will present various topics based on provided literature for the remainder of the semester. They will be asked to submit report on their presentations. The course may also include some projects. The final grades will be based on the presentations, reports and/or projects together with class participation.

Class meetings Twice a week for 75 minutes each class.

Prerequisites No perquisites. Open to all graduate students in CS and ECE and others with permission of the instructors..

Readings Selected papers in the research area.

Overlap with other courses It may have minor overlap with CS545 (Formal Methods in Concurrent and Distributed Systems). No other overlap with other courses.

Course Outline

List of Topics	Hours
Introduction and basic mathematical tools	**9**
Hidden Markov Chains	3
Automata and Temporal Logic	3
Belief Propagation	3
Hybrid automata	**9**
Semantics	3
Composition	3
Probabilistic Hybrid Automata	3
Verification of Hybrid Automata	**9**
Model Checking based approaches	3
Approaches based on theorem provers	3
Combined approaches	3
Runtime Verification	**9**
Accuracy Measures	3
Monitorability	3
Monitor Design	3
Research Challenges	**9**
Decision-Theoretic Monitoring	3
Machine Learning and Safety	3
Data-Driven Models	3
Total	45

CS 594 Geometric Algorithms for Data Analysis

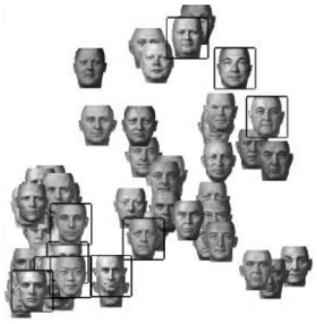
Instructor: Anastasios Sidiropoulos

Method of instruction: The instruction will be based on the following main components:

- During the first half of the course, the instructor will present various fundamental methods and ideas from computational geometry and discuss their applications in data analysis. Any necessary prerequisites will also be discussed during this time.
- During the second half of the course, the students will read and present research papers.
- The students will work on a project of their interest that incorporates ideas discussed in the class. The students will have the option to either conduct original research or experimentally evaluate prior work. The project will be performed in teams of 1–3 students. The students will be encouraged to start thinking about possible research topics early in the semester. The instructor will hold frequent meetings with each team to guide their progress.

Narrative description: Complex data sets arise in a plethora of application domains from measurements of various physical processes, history of financial transactions, logs of user activity in a network, and so on. The analysis of such data sets is therefore a task of increasing importance for science and engineering. Even though in many applications there is an abundance of such raw inputs, extracting meaningful information can often be a major computational challenge. In most cases, this difficulty is due to the lack of a useful representation of the data.

Over the recent years, geometric methods have become an indispensable tool for overcoming this difficulty. The reason behind this development is the fact that a data set endowed with pairwise similarities can be naturally interpreted as a geometric space. Such data sets include DNA sequences, statistical distributions, collections of news articles, and so on. Under this interpretation, several important data analytic questions can be understood as geometric computational problems. For example, the problem of classification can be expressed as geometric partitioning. Similarly, the problem of fitting a model to a set of measurements can be thought of as interpolation in some appropriate geometric space.



In these contexts, the main algorithmic challenges occur in high-dimensional, or more generally, complex metric spaces. Contemporary Computational Geometry aims at addressing the above challenges via the design of efficiently algorithmic methods for the analysis of these spaces.

Goal: In this course, the students will be exposed to algorithmic methods from computational geometry for the analysis of high-dimensional and complex data sets. Emphasis will be given on understanding the state of the art of these methods, and on developing intuition about which methods are appropriate in various application contexts.

Student deliverables: The students will have to read all the papers, and they will be expected to

actively participate in all the lectures. Furthermore, each student will present at least one research paper to the class. For the final project, the students will have to submit a proposal of their selected topic within the first half of the course, a final report at the end of the class, and they will be asked to give a brief presentation on their findings.

Class meetings: There will be two 75' meetings per week.

Prerequisites: The course will be accessible to students with a wide range of backgrounds, including both theoretical and applied areas of computer science, as well as mathematics. Some familiarity with algorithms will be assumed, equivalent to a CS 401-level course.

Exams: There will be no exams.

Readings: Selected research papers from the following tentative list of topics:

- *Compression:* How to compress a data set, by introducing only a small error (for example, distance oracles, graph spanners, dimensionality reduction, and so on).
- *Sketching:* How to summarize a large data set, so that relevant information can be recovered from a small sketch (for example, via coresets).
- *Similarity search:* How to find similar elements in a large data set (for example, approximate nearest-neighbor search, locality sensitive hashing, and so on).
- *Metrical simplification:* How to approximate a "complicated" metric space by a simpler one (for example, embedding arbitrary metric spaces into random trees).
- *Clustering:* How to partition a large data set into a few classes of closely-related elements (for example, k-means, k-median, k-center, spectral clustering, and so on).
- *Outlier detection:* How to remove irrelevant/erroneous elements from a data set.
- *Metric learning:* How to transform a geometric data set, so that it agrees with the opinions of experts, and how to use this transformation for prediction.

The discussion of the above topics will include motivational examples from various application domains, such as visualization, machine learning, bioinformatics, statistics, networking, and streaming/massively parallel computing.

Overlap with other courses: To the best of my knowledge, this course does not have a significant overlap with any other CS 594 courses from recent years.

The problem of clustering is discussed in CS 412 and in CS 583 by prof. Liu. However, in the proposed course, clustering will be covered from the point of view of algorithm design. Several different clustering objectives will be compared in terms of their computational difficulty, and the emphasis will be on presenting the currently best-known approximation algorithms. This treatment of the subject aims to complement the discussions in more applied courses.

The problem of dimensionality reduction is discussed in CS 412. In the proposed course, this topic will be covered at a greater depth, and as a specific instance of a more general "compression" theme, which includes methods such as graph spanners, distance oracles, and so on.

The topic of prediction is covered in great depth in the course CS 594 by prof. Ziebart (Fall 2013). In the proposed course, prediction will only be mention in passing, as a motivational example for the methods of metric learning.

Title: Web Security and Privacy

Method of Instruction:
This class will consist of two components:

- Reading and discussing previous research. Students will read, analyze, and discuss recent and/or influential security and privacy papers. A variety of topics will be discussed within the realm of web/mobile security and privacy, with a focus on online social networks. Readings will be tweaked/assigned based on students' interests.
- Research project. Students will be required to undertake a research project of their interest, relevant to any of the general themes of the course. The project will consist of original research, and will be in groups of 2 or 3 students (depending on student enrollment). I will schedule meetings with each project group to help them choose a problem of appropriate scope. There will be a mid-term status review (for any necessary course correction), a final presentation, and a project report.

Narrative Description:
As online services have become an integral part of daily life, with an ever-increasing part of our professional, social and personal life involving the Internet, it is of paramount importance to ensure the security and privacy of our online activities. Modern web services, especially online social networks, offer an expansive set of functionality that has redefined the World Wide Web, and altered our perception of privacy in the digital era. As these services have a massive user base and contain vast amounts of personal and sensitive user information, they have become highly profitable targets for online miscreants. The constant rise in attacks, the increasing exposure of private information and activities to untrusted third parties, and the sophistication of modern adversaries, have rendered the need for effective countermeasures more critical than ever.

Goal: In this class, students will become familiarized with state of the art security and privacy research and the techniques employed, and will actively try to demonstrate the shortcomings of proposed or deployed systems. Students will be guided into assuming the role of the adversary and questioning the design choices and assumptions of existing systems and technologies, with the goal of identifying real problems and exploring practical countermeasures.

Student Deliverables:

- *Reading*. Students will be required to read all papers. Each student will present at least once, and students will be required to participate in the discussion and critique the presentations of others. To ensure that students read all assigned papers, they will be required to turn in a summary for each paper before class. Students will be graded based on their presentation as well as their participation during Q&A.
- *Research*. Students will have to submit a research proposal at the end of the 2nd or 3rd week, a project status update at the end of the 8th week, and a final write-up

towards the end of the course (structured as a research paper). Each team will also do a 20-minute presentation of their project in class.

Class meeting: 75 minutes meeting *twice* per week which includes final project presentations.

Prerequisites
There are no explicit prerequisites, however a background in concepts of (network) security and web technologies is recommended. Students that have participated in active research activities are strongly encouraged.

Exams
There will be no exams for this course.

Readings
Selected research papers from a variety of relevant topics. Tentative topics include

- Bot prevention techniques, and state-of-the-art attacks for bypassing them
- Identity theft in online social networks (OSNs)
- Detecting and preventing Phishing/Spam in OSNs
- Detecting Sybil attacks in OSNs
- De-anonymization attacks in OSNs
- Privacy in location-based services: attacks and defenses
- Cookie and session management / hijacking
- Web tracking
- Cross site scripting, cross site request forgery, clickjacking
- Web authentication / Single Sign On services

594 Special Topics Proposal

Advanced Computer Networking

Method of Instruction:

This class will involve lectures, paper discussions, a midterm programming assignment and a collaborative research project.

- ❏ **Lectures***:* The first six-eight weeks of classes will consist of regular lectures. These lectures will provide context for each subtopic and set the tone for paper discussions that follow.
- ❏ **Paper discussions***:* We will discuss a handful of influential papers per topic, driven primarily by students. Every student will present one or more papers, depending on the class size. I will provide a list of papers and allow students to pick papers of their interest across topics on a first-come-first-serve basis.
- ❏ **Midterm programming assignment**: The goal of this assignment is to expose graduate students to ns-3 simulator which is widely used in the networking community. Students will implement simple protocol extensions and evaluate their implementations with community-standard workloads.
- ❏ **Projects***:* Students would collaborate and explore an open research problem. I will schedule meetings with each project group to help them choose a problem of appropriate scope. There will be a mid-term status review (for any necessary course correction), a final presentation, and a project report.

Narrative Description:

Networks are critical for diverse computing environments, from power-constrained IoT devices to compute-intensive datacenters. Various aspects of networking (e.g., topology, routing, congestion control) are *carefully* designed to match computational needs of the underlying applications to the platforms' constraints (e.g., energy, cost). While battery-operated bluetooth devices employ simpler protocols in ad-hoc mesh networks to enable monitoring and sensing, datacenters require sophisticated transport protocols in bandwidth-intensive clos networks that provide near-optimal bisection bandwidth to process vast amounts of data (e.g., Web Search). Therefore, it is fascinating to compare and contrast design choices across the spectrum from *minimal* ad-hoc networks and Internet to more powerful datacenters and Interconnection networks.

Goal

The course builds on top of our existing networking course, “CS450: Introduction to Networking“ and focuses on the finer aspects of network design. While CS450 introduces the design of Internet, this course would enable students to *question* Internet’s design by understanding the design choices of other networks’ (e.g., datacenters, Interconnection networks, IoT networks). It is intended for students who wish to pursue research in systems (networking, distributed systems, or architecture).

Student Deliverables

- ❏ *Paper Reading*: Students will be required to read and understand all papers. Each student will present one or more papers, depending on class size. Students will be graded based on their presentation as well as their participation during Q&A.
- ❏ *Midterm Programming Assignment:* Students will present four page project report (two column) describing their implementation and present quantitative evaluations.
- ❏ *Research Project*: Students will be required to submit a project proposal at the end of the 4th week, a mid-term project status report at the end of the 10th week, and a final presentation (15 mins + 5 mins) and report at the end of the course.

Class meeting

75 minutes meeting *twice* per week which includes final project presentations.

Prerequisites

For students who have not done CS450 or equivalent, instructor consent is required.

Exams

There will be no exams for this course.

Readings

Lectures and papers from a variety of relevant topics. Some of the tentative topics are

- ❏ The end-to-end design principle
- ❏ Internet

- ❏ Interconnection networks
- ❏ Datacenter networks
- ❏ Ad-hoc Mesh networks
- ❏ Flow and congestion control
- ❏ Software-defined Networking (SDN)
- ❏ Network virtualization

Overlap with other courses

To my knowledge, this course does not overlap with any of our existing 594s. Here is a list of 594s from previous semesters (these are the ones that I could gather from Santhi):

- Advanced topics in Software Engineering - Prof. Mark Grechanik
- Advanced Computer Security - Prof. Steve Checkoway
- Advanced and Persistent Threats
- The art and science of Dialogue-Based Systems
- Introduction to Security
- Web Security and Privacy
- Empirical Analysis: Deriving Sound Insights from Data

As you can see from this list, none of our previous 594s delve into networking.

The only course with which the proposed course has some overlap, although only by about 5%, is CS450. However, CS450 introduces basic networking concepts *only* in the context of Internet. In contrast, this course explores recent developments in a broader context (from datacenters to interconnection networks).

Department of Computer Science, University of Illinois at Chicago

Spring 2018

CS 594 — Advanced Machine Learning

CRN: 38551

Course Syllabus

Time: TR 5 – 6:15 PM
Classroom: TBD
URL: via Blackboard / Piazza (to be set up)

Instructor: Xinhua Zhang
Office: 1237 SEO
Phone: 312.413.2416
E-mail: zhangx@uic.edu (preferred)
Office Hours: TBD
Contact the instructor by email if you would like to meet at other time.
TA: TBD

Prerequisites

MATH 310 or MATH 320, CS 412 (which in turn prerequires CS 251; and IE 342 or STAT 381); or consent of the instructor. For graduate students, these prerequisites are only advisory. A self-evaluation will be posted on Blackboard by Week 1.

Course Goals:

- Students will be able to have an in-depth understanding of the principle and characteristics of advanced machine learning task settings (e.g., structured prediction, distributed optimization, deep learning for complex data).
- Students will be able to scale machine learning techniques to big datasets, by leveraging new structures in the data and new computational tools that emerge even after the completion of the course.
- Students will be able develop and analyze novel problem formulations and machine learning techniques that adapt to data analysis problems emerging in new applications.

Restrictions

Restricted to students in the following colleges/schools: Engineering or Graduate College. Capped to 32 students. Students who have taken CS 594 (Advanced Machine Learning) in Fall 2016 are NOT eligible for registration.

Credit Hours

4 graduate hours

Textbooks (Required)

[Mur] Kevin P. Murphy. Machine Learning: A Probabilistic Perspective. MIT Press, 2012. Free book available electronically via UIC library

[BV] Stephen Boyd and Lieven Vandenberghe. Convex Optimization. Cambridge University Press, 2004. PDF available at https://web.stanford.edu/~boyd/cvxbook

[GBC] Ian Goodfellow, Yoshua Bengio, and Aaron Courville, Deep Learning. MIT Press, 2016 Free book at: http://www.deeplearningbook.org

Also used will be recent research papers and excerpts of relevant background material from available textbooks, and supplemental notes for specific topics.

Tentative Schedule

1. (6 hours) Bayesian networks – naïve Bayes, hidden Markov models, general Bayesian networks, and topic models.
2. (2 hours) Undirected networks – Markov/Conditional Random Fields;
3. (5 hours) Variational inference – loopy BP, expectation propagation;
4. (3 hours) Sampling and simulation – rejection sampling, importance sampling, Markov chain Monte Carlo, particle filters;
5. (3 hours) Learning algorithms – Expectation maximization, contrastive divergence;
6. (2 hours) Structured prediction – Structured SVMs and kernel methods;
7. (2 hours) Learning theory – PAC learning and VC-dimension;
8. (9 hours) Convex and stochastic/parallel optimization;
9. (8 hours) Deep learning (*2): auto-encoder, deep generative models, understanding the properties and philosophy of non-convex optimization, and generalization performance.

Midterm: 75 minutes.
Project presentation: 3 hours

Grading and evaluation (tentative)

	%	date /deadline	Coverage
Assignment 1	16 %		A project on graphical model inference and sampling
Mid-term	10 %		
Assignment 2	27 %		A project on conditional random fields and parallel optimization using MPI on UIC's ACER cluster
Course Project	27 %		Deep learning project (open)
Final exam	20 %		cumulative

Assignments and course project are in groups of up to 3 students. Midterm and final exams are individual.

Assignments 1 and 2 aim to develop students' ability to apply machine learning tools to different data analysis problems and perform thorough experimentation, benchmarking, and empirical analysis. Tasks will be specified in Assignments 1 and 2.

Mid-term and final exams will assess students' understanding of different machine learning tasks and techniques, especially their theoretical properties and underlying principles.

Course projects, including reports and presentations, aim to develop students' ability to create novel machine learning techniques for different real-data analysis problems by leveraging their specific structures and computational resources at hand.

The final exam will be comprehensive and cumulative.

1. The mid-term will be **75 minutes** with **closed book**. It will be given in the classroom during class time. Therefore, **no make-ups** will be given. Partial grading will be used.

   The mid-term covers everything from the first **seven** weeks (tentatively).

2. In Assignment 1, you will implement a variety of probabilistic inference methods on a variety of Bayesian networks and run these algorithms on some simple networks. You will play with their performance and draw conclusions. You may choose to use any language and the grading will be based on your results.

3. In Assignment 2, you will build a classifier which recognizes "words" from images. This is a great opportunity to pick up practical experiences that are crucial for successfully applying machine learning to real world problems, and evaluating their performance with comparison to other methods. Although you may use any language of your choice, we will only provide utility code in Matlab/Octave.

   In this assignment, you will also have a great opportunity to learn parallel optimization. We will use PETSc which is in C/C++ and uses MPI for message passing. All low-level constructs have been encapsulated into high level linear algebra commands and optimization solvers (PETSc package: http://www.mcs.anl.gov/petsc). You will run experiments on UIC's

cluster: ACER. If your own machine gets multiple cores (highly likely) AND you are happy to install MPI and PETSc by yourself (caution: highly challenging for newbies, but we will provide some helpful guides), you may do experiments on your own machine.

4. The course project will be on deep learning. You will have the opportunity to design your own project. All teams will have 10 minutes to present their projects orally in the two sessions on week 15. Only *initial* results will be needed for oral presentation, and the detailed results can be submitted in the final report later.

All the five evaluations will be graded out of 100, and their weighted sum will be used to determine the final letter grade (A/B/…) **by curving**.

Communication, feedback, and discussion:

1. The web page on Blackboard will contain all materials relevant to the class, syllabus, assignments, lecture notes etc. You can also see you own grades.

2. For general announcements and notifications, I will send email to the whole class via Blackboard. Please check your email frequently, especially around deadlines (homework and exams).

3. Piazza will be used for Question and Answer. The system is highly catered to getting you help fast and efficiently from classmates and myself. Rather than emailing questions to me, I encourage you to post your *technical* questions on Piazza. If you have any problems or feedback for the Piazza developers, email team@piazza.com.

   Find our class page at: https://piazza.com/uic/spring2018/cs594atml/home (to be set up)

4. If you have any personal or non-technical questions, please send an email directly to the instructor (zhang@uic.edu) or the TA (if instructed to do so, e.g. dispute of grading).

General Policies on homework (project) and exams

1. **Late submission**s: Late submissions will not be accepted in any case, unless there is a **documented** personal emergency. Arrangements must be made with the instructor as soon as possible after the emergency arises, preferably well before the homework due date.

   Advice: If for whatever reason you don't manage to finish an assignment, hand in what you have. Partial credit will be given.

2. **Statute of limitations**: **Three weeks!** No grading questions or complaints — no matter how justified — will be listened to **three** weeks after the item in question has been returned.

3. **Group collaboration**: all members of each group should make nearly the same level of contribution to each project. So in a group of three, each member contributes 1/3 to Assignment 1, 1/3 to Assignment 2, and 1/3 to the course project. It is **not** allowed that one works on Assignment 1 alone, one on Assignment 2 alone, and one on the course project. All

members should be responsible for the whole submission of the team, not only his/her own contributed part. You are supposed to understand the work of your teammates inside out, and be able to answer questions when asked. If one member plagiarized, then **all members** of the team will receive the **same** penalty.

Policy on Academic Integrity

Academic dishonesty will not be tolerated. Please see the CS department policy below on the topic; this policy specifies penalties for violations.

What is academic dishonesty? To hand in any work which is not 100% the student's creation, unless you are explicitly allowed to do so. Thus:

Exams: All work on all exams must be individually performed.

Projects: no team may give any other team any portion of their solutions or code, through any means. Students are not allowed to show each other any portions of code or homework, unless they are on the same team.

Important Note: almost every semester somebody is caught red-handed and as a consequence fails the class. Isn't it better to get a B or a C than an F?

CS department policy on academic dishonesty

The CS Department will not tolerate cheating by its students. The MINIMUM penalty for any student found cheating will be to receive an F for the course and to have the event recorded in a department and/or College record. The maximum penalty will be expulsion from the University. Cheating includes all the following, though this is not a complete list:

- Copying or any other form of getting or giving assistance from another student during any test, quiz, exam, midterm, etc.
- Plagiarism–turning in writing that is copied from some other source.
- Obtaining solutions to homework by posting to the Internet for assistance, purchasing assistance, obtaining copies of solutions manuals for instructors, and obtaining copies of previous year's homework solutions.
- Computer programs: Any time you look at another student's code, it is cheating. (Exception: If you are EXPLICITLY told that you may do so by the instructor.)

For computer programs, if for some reason we cannot determine who copied from whom, we may, at our discretion, give failing grades to both students.

It is the responsibility of all engineering and computer science professionals to safeguard their company's "trade secrets." An employee who allows trade secrets to be obtained by competitors will almost certainly be fired. So, YOU are responsible for making sure that your directories have permissions set so that only you can read your files, for being sure to log out at the end of working in the computer lab, etc.

CS 594 - Artificial Intelligence: Methods and Applications
Instructor – Dr. Plamen Petrov
Meeting time – Thursdays, 630-9 pm
CRN – 41289

Class details:

Course Objectives:
The aim of this course is to introduce students to a system view of Artificial Intelligence by connecting the core concepts that make up an AI system and bringing an advanced understanding of how AI systems work. The course will expose students to some state of the art technology implementation of core AI methods and technologies and allow students to perform hands on experimentations. The course will touch upon the following three core components of a typical AI System: Intelligent Natural Language User Interaction, Knowledge Modeling and Representation, and Probabilistic Reasoning. The course will provide examples of those areas using the following two state-of-the-art AI platform:

- IBM Watson: https://www.ibm.com/watson/developer/
- BayesiaLab: http://www.bayesia.com/

Students will be provide free academic access to these two AI platforms and will be able to experiment for the hands on learning of the concepts.
Students will learn how to develop an AI-based conversational agent that interacts with a human by applying text classification, intent detection, entity detection and Information Extraction, and Dialog Management, then retrieves appropriate information or performs reasoning tasks in a knowledge domain using probabilistic inference using Bayesian networks. The primary goal of the course is to first equip students with good understanding and intuition of how different Artificial Intelligence methods work together to implement intelligent systems, and then let students perform in depth research and experimentation in a relevant area of their interest.

Method of Instruction:
This class will consist of three components:

- Lecture and tutorials. The instructor will introduce advanced concepts and demonstrate sample implementations of the concepts. The presented material will be based on select chapters from books and research papers.
- Reading and discussing previous research. Students will read, analyze, and discuss recent and/or influential AI papers. A select set of topics will be discussed within the realm of Artificial Intelligence based systems, with a focus on intelligent conversational interaction within a knowledge domain. Readings will be adjusted and assigned based on students' interests.
- Research project. Students will be required to undertake a research project of their interest, relevant to any of the general themes of the course. The project will consist of original research, and will be in groups of 2 or 3 students (depending on student enrollment). I will schedule meetings with each project group to help them choose a problem of appropriate scope. There will be a mid-term status review (for any necessary course correction), a final presentation, and a project report. The students will have the option to have either more experimental or more theoretical focus of their research project, but in either case demonstrable, working, and relevant code must be included in the project.

Student Deliverables:

- Reading. Students will be required to read all papers. Each student will present at least once, and students will be required to participate in the discussion and critique the presentations of others. To ensure that students read all assigned papers, they will be required to turn in a summary for each paper before class. Students will be graded based on their presentation as well as their participation during Q&A.
- Research. Students will have to submit a research / project proposal at the end of the 3rd or 4th week, a project status update at the end of the 9th week, a final write-up towards the end of the course (structured as a research paper), as well as working code demonstrating. Each team will also do a 20-minute presentation of their project in class.

Class meeting: 150 minutes meeting once per week which includes final project presentations.

Prerequisites

Students are required to have taken and passed a graduate course in Artificial Intelligence and / or Natural Language Processing. Students are expected to be familiar with concepts of Artificial Intelligence, Machine Learning, Natural Language Processing. Students that have participated in active research activities are strongly encouraged. Students must be comfortable developing non-trivial programs in either Python or Java. (Using other programming languages for the projects will be considered on a case by case basis). The course expects that working code will be produced by students.

Exams

There will be no exams for this course.

Readings

Selected research papers from a variety of relevant topics. Tentative topics include

- Conversational interfaces
- Dialog managers
- Concept, entity and relationship extraction
- Bayesian networks for knowledge representation
- Probabilistic inference with graphical models
- Learning Bayesian network structure from data
- Causal Identification and Estimation
- Natural Language Understanding
- Question Answering
- Artificial Intelligence as a Service (AIaaS)
- Working with and extending leading AI Platforms (IBM Watson, BayesiaLab)