CS6963 Lecture #1

Robert Ricci

January 13, 2015

• Why I’m doing this class
  – Networking is a (mostly) highly empirical field
  – Convincing people of the value of work includes evaluating it
    * True both in academia and industry
  – Taking a very “systems” approach to networking (empirical rather than analytic)
  – I want you to be able to:
    * Do very thorough evals of your own work
    * Recognize weaknesses in the evals of others
  – Everything, all the time, is an experiment
    * So you be prepared to conduct it well and to learn

• Evaluating systems
  – Start with a claim and/or questions: some examples
    * Thesis statement
    * Claims found in abstract or introduction of a paper
    * Advertising claims
    * …etc.
  – Papers, etc. that have no evaluatable claim are weak
    * Falsifiability (testability): Would it be possible to make an observation that proves it false?
  – Provide evidence to support or refute that claim
    * What do you evaluate?
    * Where do you stop? (can’t evaluate everything)
  – Scoping claims is critical
    * Under what assumptions / conditions does the claim hold?
    * This feeds into your evaluation plan
    * This is critical for comparing different work—it may be possible to prove each one better under different conditions, you have to be explicit about what the assumptions are
  – Full-system evaluation: How the system under test interacts with the world
    * Boundary of system under test: what is inside, and therefore being evaluated, and what is outside, and therefore not
    * Evaluating that system in a reasonable setting under reasonable workload
    * Micro vs. macro tests
• In this class
  – Use about 1/3 of Raj Jain’s book — old, but not outdated
  – Start with written homeworks
  – Shift to paper reading and analysis
  – Finish up with lab work on real testbeds (built here)
  – Including a detailed analysis of your own devising
  – Readings most days for the first half, taper off later
  – One week for HW, 1.5 for paper analysis, 2 for labs
  – Homeworks due at midnight after class so that you can ask questions
  – You’ll give at least two talks
  – Guest lectures as case studies
  – We’ll send some time reproducing results from literature and evaluating things that they didn’t

• Class logistics
  – Syllabus things
    * Grading assignments, %ages
    * Readings due at the start of class
    * Assignments due: at midnight after class
    * Cheating policy
    * Late policy: have to tell me the day before except in exceptional cases, give me access to repo
  – Everything in git — want you to get in the habit of keeping everything and organizing
    * I won’t grade you on your git history, but I’ll judge you
    * You can submit merge requests against the main class repo
  – Will put all assignments, including reading, in Canvas
  – Schedule in GitLab, but what’s in Canvas is official
  – Help available on Canvas
  – Can everyone bring laptops to class?

• What’s everyone working on (eye towards final project)?

• Git basics
  – Objects and DAG of commits
  – Relationship between repositories
    * Multiple remotes
  – Where files are stored (FS, index, pack)
  – How to check status - eg. have you checked in everything, pushed?
  – How to submit homework for this class
  – Where to get more help (gits@flux.utah.edu, Chacon book, etc.)

• For next time
  – HW #1 posted in Canvas for bootstrapping
    * There’s a step where you have to wait for me, so do it right now!
  – Read Chapter 1, including page 1, through the end of 1.2