

# Lessons Learned from My PhD Career

# Summary of my research topics

- Undergraduate
  - Hacking a web browser for PLT analysis
- PhD 1.5 years
  - Measuring, analyzing, and optimizing mobile performance
- PhD 1.5 years
  - AI on smartphones
- PhD 1.5 years
  - AI on IoT cameras

# Research Areas

- Which community you are working or will work with?
  - A community means 2-3 conferences, a group of senior people
- Know the most popular topics in your community
  - Understand at least 80% papers from this community – Lin Zhong from Yale
  - Who are the most active guys in each sub area?

# Research Topics

- Old vs. Hot
- Data-driven vs. Goal-driven
- Safe vs. Risky
- Focused vs. Broad
  - Learn some new stuff from each project
- Convince yourself first, then others
- Sharing with others before writing any code
  - Negative feedbacks are needed
- Keep your “stupid” research ideas in a doc

# Papers

- Treat every project (paper) seriously
  - Push your co-authors because it's so important
- Submission matters more than acceptance
  - Live with rejection and complain less
- Never let yourself give up a submission if it's planned
  - But make it as good as you can
- Equal time spent on (1) initializing the idea (2) design and implementation (3) writing, revising, and submitting
  - Every stage needs to be iterated for many times

# Reading

- Make it enjoyable and relaxing time
  - Be critical, but more importantly, find the shining ideas
  - Learn the techniques and writing
  - Two questions: what's the problem to solve? What's the key idea?
- Walk through the papers on your target conference each year

# Collaboration

- Be responsible as the project executor
  - Be responsive and reply fast
  - Deliver on your promise
  - Keep everyone on the same page
  - Bring out new stuff based on your hands-on experience
- Push everyone based on their schedules
- Always summarize the todo after each meeting

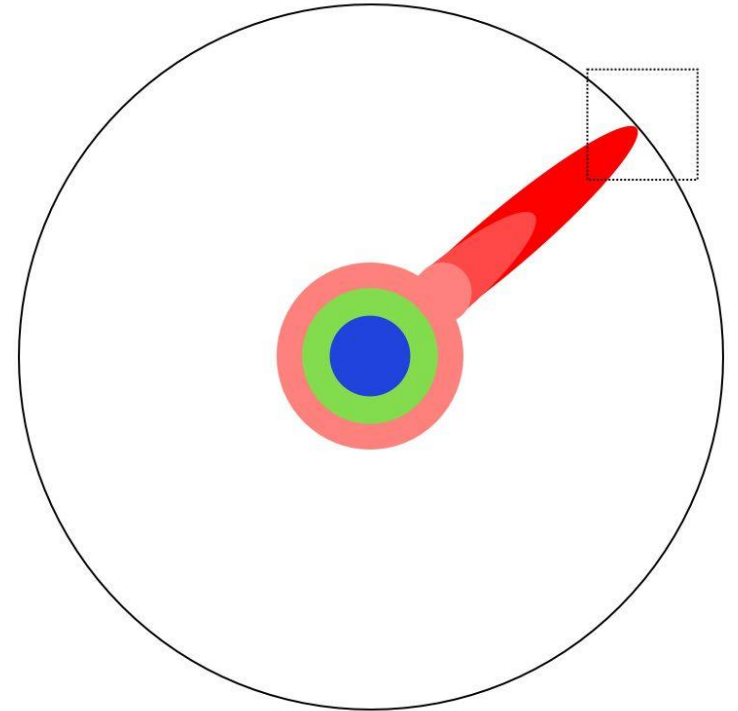
# System Research

- System research is about taste
- Systems are built for real apps & scenarios & workloads
  - Or at least promising in the future
- Keep eyes on the emerging techniques and hardware
- System research makes trade-offs
- Working Code Trumps All Hype – Satya from CMU



# MISC

- Have a big picture
- Know what others are working on
  - Colleagues, roommates, industry
  - Even other research communities
- PhD is a long journey
  - Get strong mentally and physically
  - Work hard in a sustainable way



# A walk through of my papers published during PhD and what I've learned from them

- [TMC'17] ShuffleDog: Characterizing and Adapting User-Perceived Latency of Android App
  - **Learn how to manage large codebase. Deliver the paper whatever happens.**
- [WWW'17] AppHolmes: Detecting and Characterizing App Collusion among Third-Party Android...
  - **Be curious of what is happening in our real life. Can we turn them into research ideas?**
- [MobiCom'18] DeepCache: Principled Cache for Mobile Deep Vision
  - **Be fast – timing is super important. Writing is everything.**
- [IMWUT'18] DeepType: On-Device Deep Learning for Input Personalization Service with...
  - **Harness the big data from industry. Solve a real problem.**
- [WWW'19] A First Look at Deep Learning Apps on Smartphones
  - **Measurement is always easy and attractive. Follow your intuition of what others may care.**
- [MobiSys'20] Approximate Query Service on Autonomous IoT Cameras
  - **Figures are important.**
- [USENIX ATC'21] Video Analytics with Zero-Streaming Cameras
  - **Keep submitting. Novel papers need strong motivations.**

# A few thoughts after one-year faculty experience

- Working hard is the only way to success
- 99% research work is useless; 99% researchers are valuable
- It's okay to lower down your expectation of a work during progress
  - Pursue solving a fundamental problem at first
  - Work on a (top conference) paper acceptance at least
  - It's enough to make the top conf. reviewers appreciate your work
- Push your advisor
- Papers mean (almost) everything to PhDs