

Computer Science Research

CS 197 & 197C | Stanford University | Sean Liu & Lauren Gillespie

Slides adapted from previous iterations of the course: Michael Bernstein & Jingyi Li.



Computer Science Research

CS 197 & 197C | Stanford University | Sean Liu & Lauren Gillespie

Slides adapted from previous iterations of the course: Michael Bernstein & Jingyi Li.

**What is computer
science research?**

Seeking a Better Way to Find Web Images

By JOHN MARKOFF NOV. 19, 2012

STANFORD, Calif. — You may think you can find almost anything on the Internet.

But even as images and video rapidly come to dominate the Web, search engines can ordinarily find a given image only if the text entered by a searcher matches the text with which it was labeled. And the labels can be unreliable, unhelpful (“fuzzy” instead of “rabbit”) or simply nonexistent.

To eliminate those limits, scientists will need to create a new generation of visual search technologies — or else, as the Stanford computer scientist [Fei-Fei Li](#) recently put it, the Web will be in danger of “going dark.”

Now, along with computer scientists from Princeton, Dr. Li, 36, has built the world’s largest visual database in an effort to mimic the human vision system. With more than 14 million labeled objects, from obsidian to orangutans to ocelots, the database has become a vital resource for computer vision researchers.

Stanford Researcher Finds Lots of Leaky Web Sites

By SOMINI SENGUPTA OCTOBER 11, 2011 6:32 PM 6

The Web is porous. Remarkable information trickles in from everywhere. It also sometimes spills out without its users knowing exactly where or how.

Take for instance these findings, released on Tuesday by computer scientists at Stanford University. If you type a wrong password into the Web site of The Wall Street Journal, it turns out that your e-mail address quietly slips out to seven unrelated Web sites. Sign on to NBC and, likewise, seven other companies can capture your e-mail address. Click on an ad on HomeDepot.com and your first name and user ID are instantly revealed to 13 other companies.

These findings, [released](#) by the Center for Internet and Society at Stanford Law School, are among the leaks found on 185 top Web sites. They serve to buttress what privacy advocates have long

CLOUD COMPUTING

Making Cloud-Computing Systems More Efficient

By QUENTIN HARDY MARCH 6, 2014 7:00 AM 3

Email

Share

Tweet

Save

More



Christos Kozyrakis, professor of electrical engineering and computer science at Stanford University, headed the creation of management software called Quasar.



GIVE THE TIMES

Ms. Valentine and Mr. Bernstein wanted to take the concept further. They created a platform, Foundry, in which the process of assembling and running a temporary organization could be automated, without so much as a phone call.



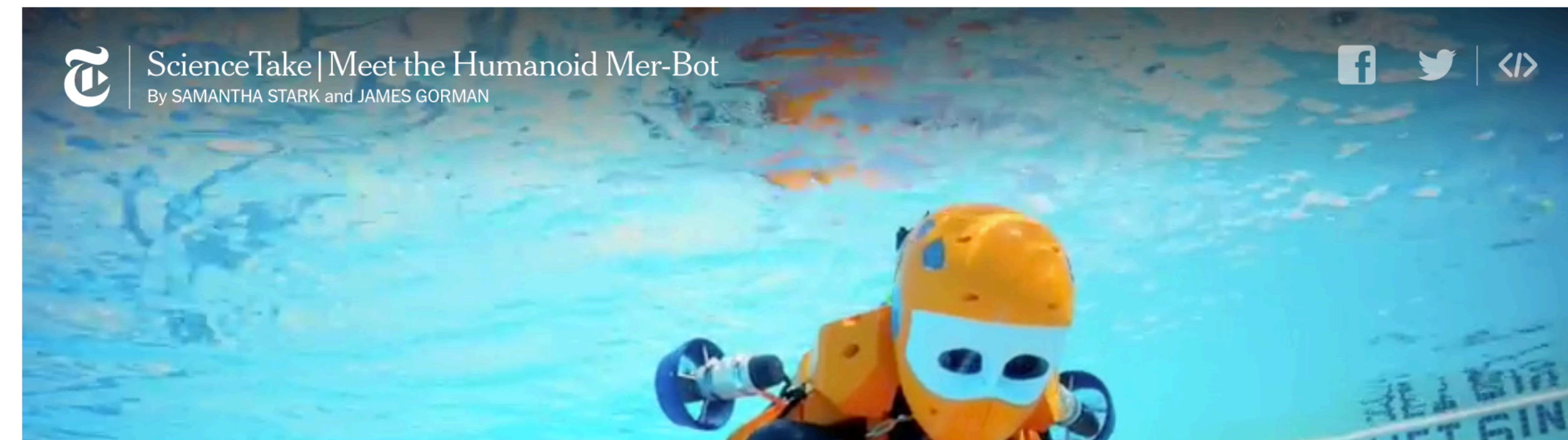
SCIENCE

SCIENCETAKE

OceanOne, a Mer-Bot Dive Buddy With a ‘Friendly Face’



ScienceTake | Meet the Humanoid Mer-Bot
By SAMANTHA STARK and JAMES GORMAN



**What will this course
achieve?**

Your experience in this course

Work on bleeding-edge topics now, rather than in two years

Fundamental research skills: read and write academic papers, formulate research questions, hands-on practice

Onramp to research in the department, and to research and advanced development in industry

197 vs. 197C

Both will have lecture here on Tues 10:30-11:20am

197

- Prereq: 106B
- Designed for any undergraduate who wants to try out CS research!
- Get matched to a research project, mentored by your CA
- Section: Thurs 10:30-11:20am.

197C

- Prereq: 106B + commitment from an **external** mentor for weekly meetings
- Designed for CURIS, PURE, and students with ongoing research projects
- No section. One weekly meeting with your research mentor, one weekly check-in meeting with course staff

Today

What is research, vs. industry?

How does this course work?

Research mindset

Course Staff!

197 & 197C
Instructor

197 & 197C
Co-Instructor

197
CA

197
CA



Sean Liu

Lauren Gillespie

Akshat Nigam

Yujie Tao

Sections: **AI & Sustainability**

AI & CompBio

**Human-Computer
Interaction**

Sean Liu



2015: BS, UC Berkeley EECS

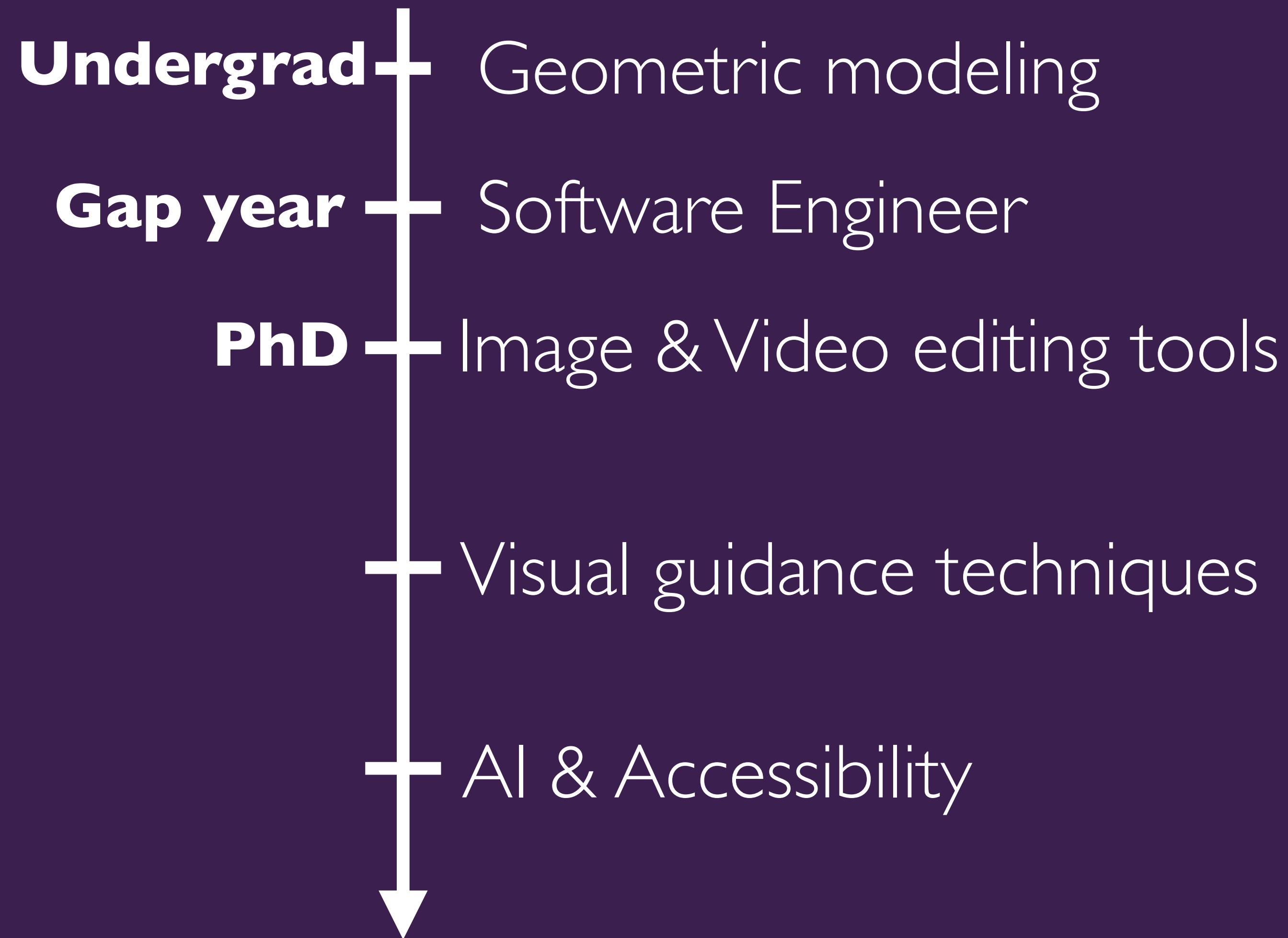
Present: final-year PhD student,
Stanford CS

Pronouns: she/her

Research Area: Computer
Graphics, HCI

(Call me Sean)

Sean's research history



Lauren Gillespie



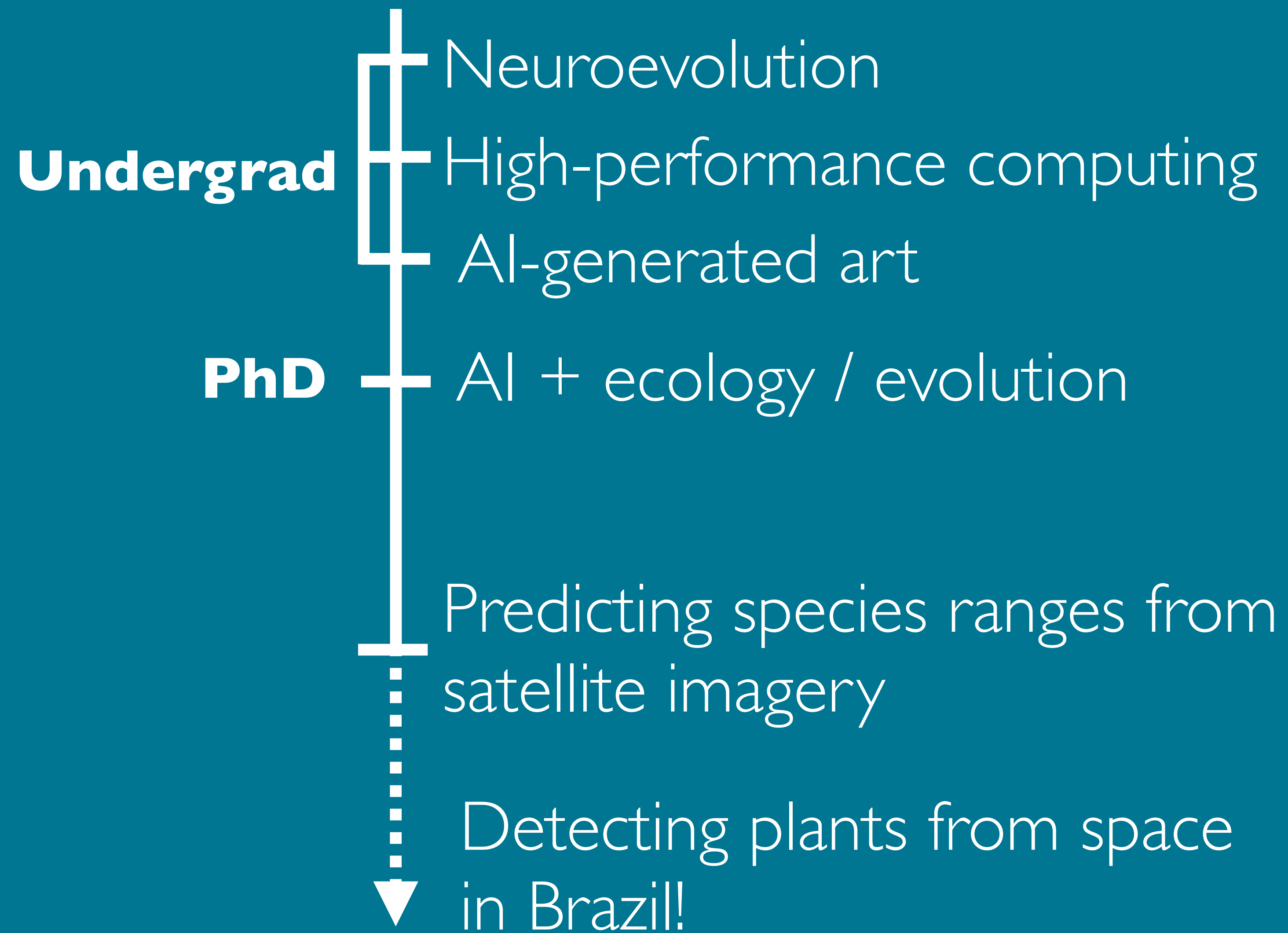
2019: BS, Southwestern University
Computer Science, Chemistry

Present: fourth year PhD
student, Stanford CS

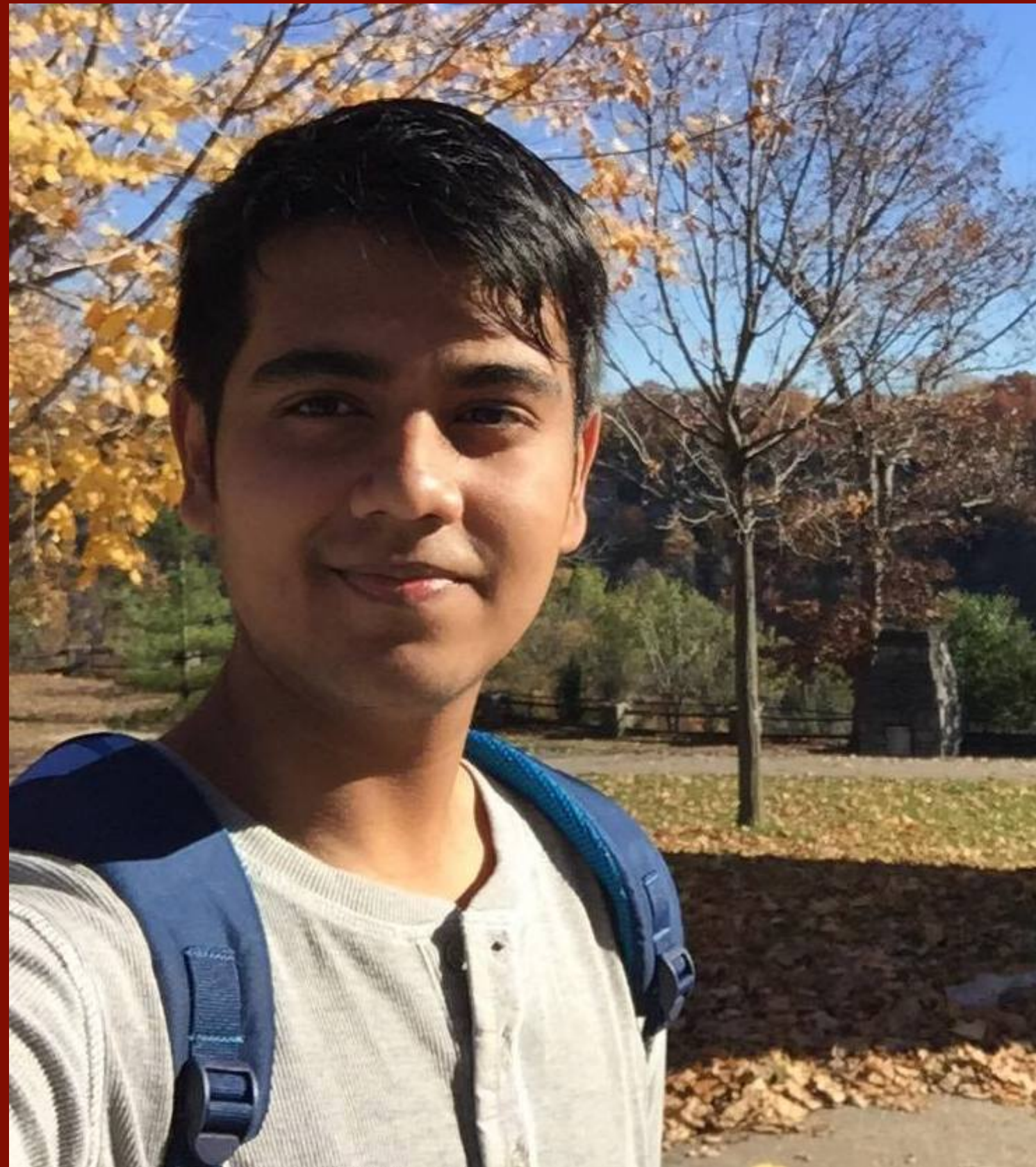
Pronouns: she/her

Research Area: AI for
Sustainability, Computer Vision

Lauren's research history



Akshat Nigam



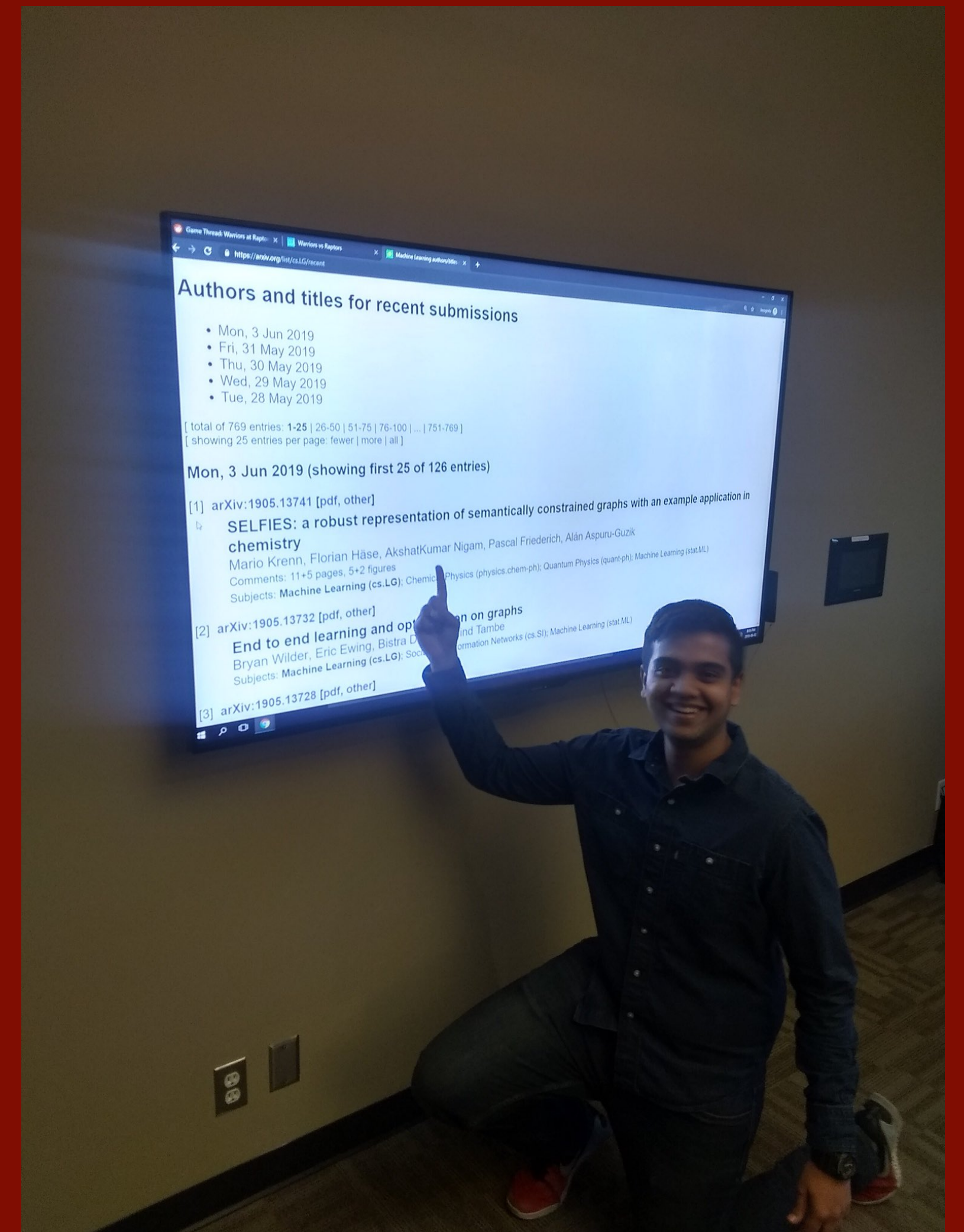
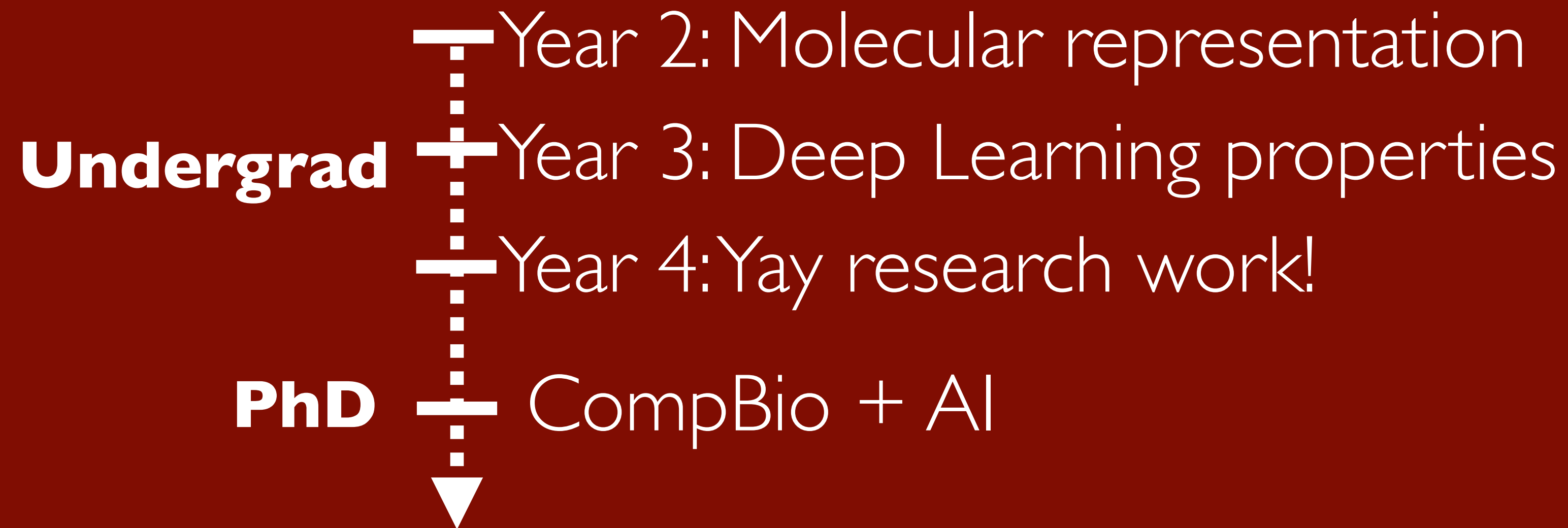
2020: University of Toronto

Present: Second year PhD student, Stanford CS

Pronouns: he/him

Research Area: Machine learning, Drug Discovery, Comp-Bio

Akshat's research history



Yujie Tao



2020: BA UNC-Chapel Hill
Journalism & Computer Science

2021: MS UChicago
Computer Science

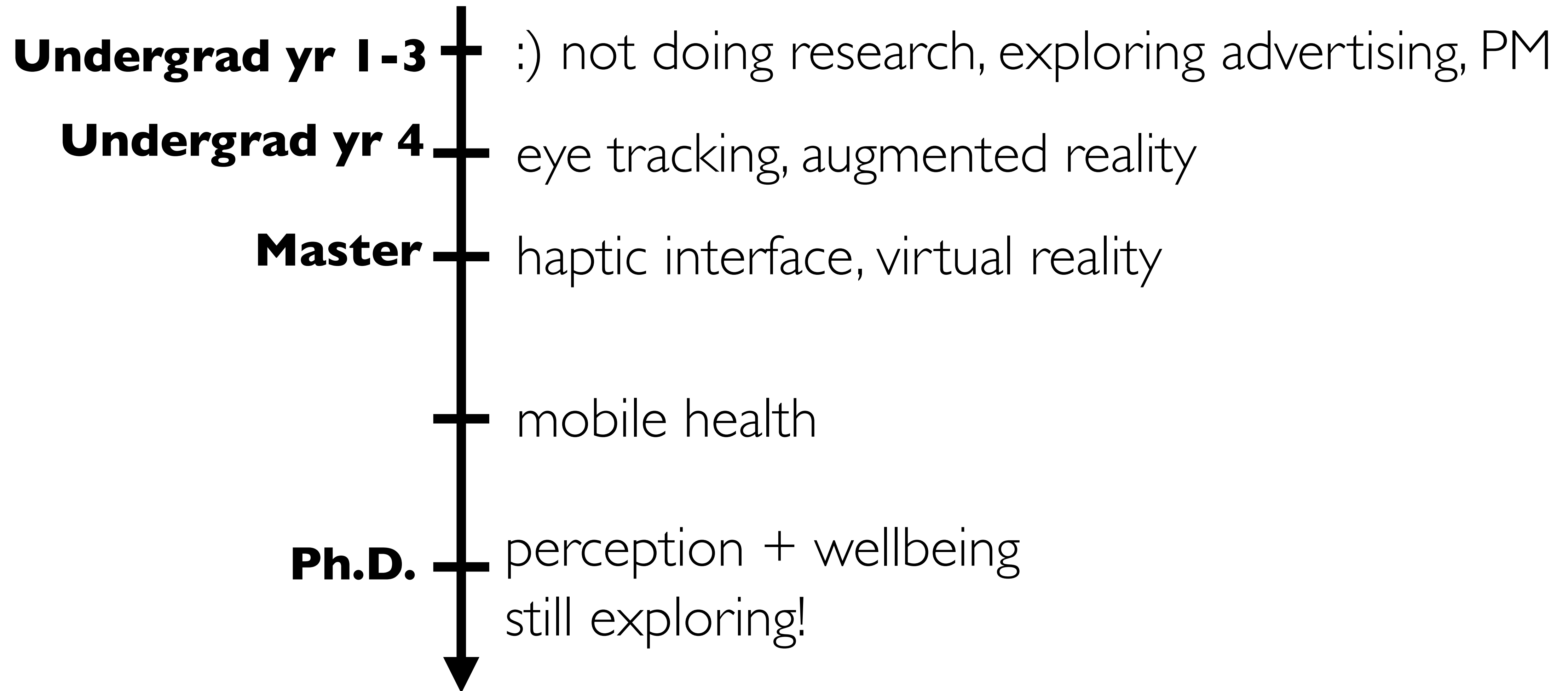
Present: first-year PhD student, Stanford CS

Pronouns: she/her

Research Area: Human Computer
Interaction



Yujie's research history



Computer science research

What is the goal of research?

Why has it driven major innovations in computing?

What separates research from advanced development?

A Tale of Three Turing Awards

Hennessy and Patterson: RISC


Computer architecture was increasing in complexity, in order to enable more and more advanced computation.

Everyone thought that increasingly powerful processors needed increasingly complicated instruction sets to take advantage of them.

☰ The New York Times 👤

GIVE THE TIMES

Computer Chip Visionaries Win Turing Award



Dave Patterson, right, and John Hennessy in the early 1990s. The men won the Turing Award for their pioneering work on a computer chip design that is now used by most of the tech industry. Shane Harvey

By **Cade Metz**

March 21, 2018

f t 📌

SAN FRANCISCO — In 1980, Dave Patterson, a computer science professor, looked at the future of the

Hennessy and Patterson: RISC


“No, let’s do it this way instead:” have a very simple instruction set. That way you can compare performance, optimize, and prevent errors.

This became known as Reduced Instruction Set Computer (RISC). It led to a sea change in architectures, and the founding of multiple major silicon valley companies.

☰ The New York Times 👤

GIVE THE TIMES

Computer Chip Visionaries Win Turing Award



Dave Patterson, right, and John Hennessy in the early 1990s. The men won the Turing Award for their pioneering work on a computer chip design that is now used by most of the tech industry. Shane Harvey

By **Cade Metz**

March 21, 2018

f t 📖

SAN FRANCISCO — In 1980, Dave Patterson, a computer science professor, looked at the future of the

Engelbart: interactive computing

When computers originated, they were used for, well, computing: calculating mathematical functions.

This meant that computers were seen as most appropriate for slow, batch interaction, shared by entire teams.



The screenshot shows the top portion of a New York Times article. At the top left is a hamburger menu icon. The page title "The New York Times" is centered at the top, with a user profile icon on the right. Below the title is a "GIVE THE TIMES" link. The article's byline reads "DOUGLAS C. ENGELBART, 1925-2013". The main title is "Computer Visionary Who Invented the Mouse" in a large, bold, serif font. Below the title is the author's name "By John Markoff". The date "July 3, 2013" is on the left, and social media sharing icons for Facebook, Twitter, a bookmark, and a comment bubble are on the right. The first paragraph of the article begins with "Douglas C. Engelbart was 25, just engaged to be married and thinking about his future when he had an epiphany in 1950 that would change the world." The second paragraph starts with "He had a good job working at a government aerospace laboratory in California, but he wanted to do something more with his life, something of value that might last, even outlive him. Then it came to him. In a single stroke he had what might be safely called a complete vision of the information age." The third paragraph begins with "The epiphany spoke to him of technology's potential to expand human intelligence, and from it he spun out a career that indeed had lasting impact. It led to a host of inventions that became the basis for the Internet and the modern personal computer." The final paragraph starts with "In later years, one of those inventions was given a warmhearted name, evoking a small, furry creature".

☰ The New York Times 

GIVE THE TIMES

DOUGLAS C. ENGELBART, 1925-2013

Computer Visionary Who Invented the Mouse

By John Markoff

July 3, 2013    

Douglas C. Engelbart was 25, just engaged to be married and thinking about his future when he had an epiphany in 1950 that would change the world.

He had a good job working at a government aerospace laboratory in California, but he wanted to do something more with his life, something of value that might last, even outlive him. Then it came to him. In a single stroke he had what might be safely called a complete vision of the information age.

The epiphany spoke to him of technology's potential to expand human intelligence, and from it he spun out a career that indeed had lasting impact. It led to a host of inventions that became the basis for the Internet and the modern personal computer.

In later years, one of those inventions was given a warmhearted name, evoking a small, furry creature

Engelbart: interactive computing

“No, let’s do it this way instead:”

computing should be used as a tool for thought. We must move from batch-style computing to interactive computing.

His result was the “Mother of All Demos”: mouse, hypertext, bitmapped screens, collaborative software, and more.

This led to Xerox Star. Steve Jobs saw it, was wow’ed, and infused the ideas into the Mac.



The screenshot shows a mobile view of a New York Times article. At the top, the New York Times logo is centered, with a hamburger menu icon on the left and a user profile icon on the right. Below the logo, there is a "GIVE THE TIMES" link. The article title is "Computer Visionary Who Invented the Mouse" by John Markoff, dated July 3, 2013. The article text describes Engelbart's work and his invention of the mouse, mentioning his work at a government aerospace laboratory and his stroke. The article also mentions that his work led to Xerox Star and the Mac.

☰ The New York Times ⓘ

GIVE THE TIMES

DOUGLAS C. ENGELBART, 1925-2013

Computer Visionary Who Invented the Mouse

By [John Markoff](#)

July 3, 2013    

Douglas C. Engelbart was 25, just engaged to be married and thinking about his future when he had an epiphany in 1950 that would change the world.

He had a good job working at a government aerospace laboratory in California, but he wanted to do something more with his life, something of value that might last, even outlive him. Then it came to him. In a single stroke he had what might be safely called a complete vision of the information age.

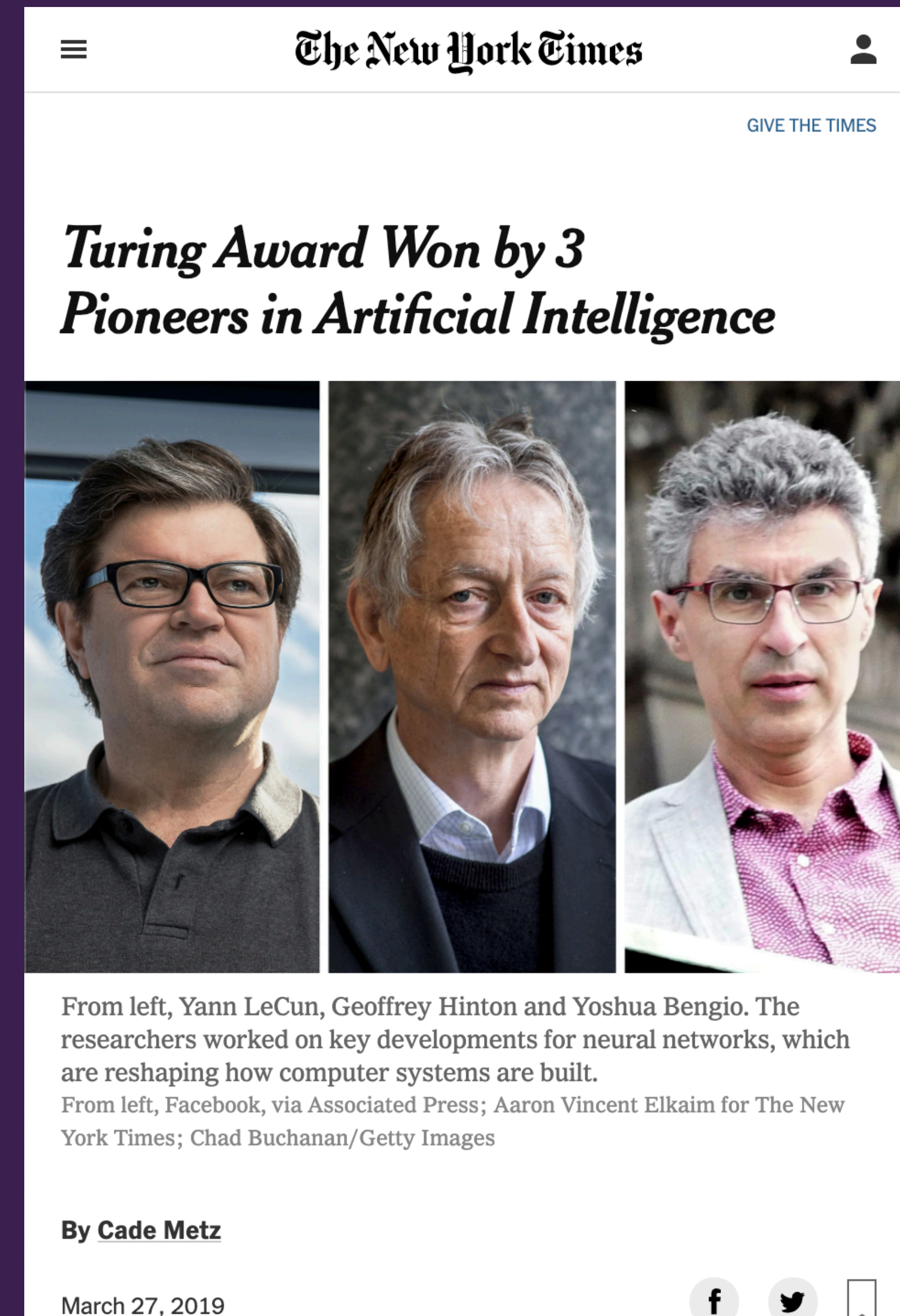
The epiphany spoke to him of technology’s potential to expand human intelligence, and from it he spun out a career that indeed had lasting impact. It led to a host of inventions that became the basis for the Internet and the modern personal computer.

In later years, one of those inventions was given a warmhearted name, evoking a small, furry creature

LeCun, Hinton, Bengio: deep learning

The idea of neural networks had been around for fifty years, but unsuccessful. Major AI figures had trashed it, even proving that early versions had very limited expressiveness.

Instead, machine learning was based on other models, for example the support vector machine and graphical models. Neural networks did not perform well.



The image shows a screenshot of a news article from The New York Times. At the top, the newspaper's logo is visible on the left, a hamburger menu icon on the far left, and a user profile icon on the far right. Below the logo is a "GIVE THE TIMES" link. The main headline reads "Turing Award Won by 3 Pioneers in Artificial Intelligence" in a bold, serif font. Below the headline are three vertical portrait photographs of the award winners: Yann LeCun on the left, Geoffrey Hinton in the center, and Yoshua Bengio on the right. Underneath the photos is a short paragraph of text, followed by a byline "By Cade Metz" and the date "March 27, 2019". At the bottom right of the article preview are icons for Facebook, Twitter, and a bookmark.

The New York Times

GIVE THE TIMES

Turing Award Won by 3 Pioneers in Artificial Intelligence

From left, Yann LeCun, Geoffrey Hinton and Yoshua Bengio. The researchers worked on key developments for neural networks, which are reshaping how computer systems are built.
From left, Facebook, via Associated Press; Aaron Vincent Elkaim for The New York Times; Chad Buchanan/Getty Images

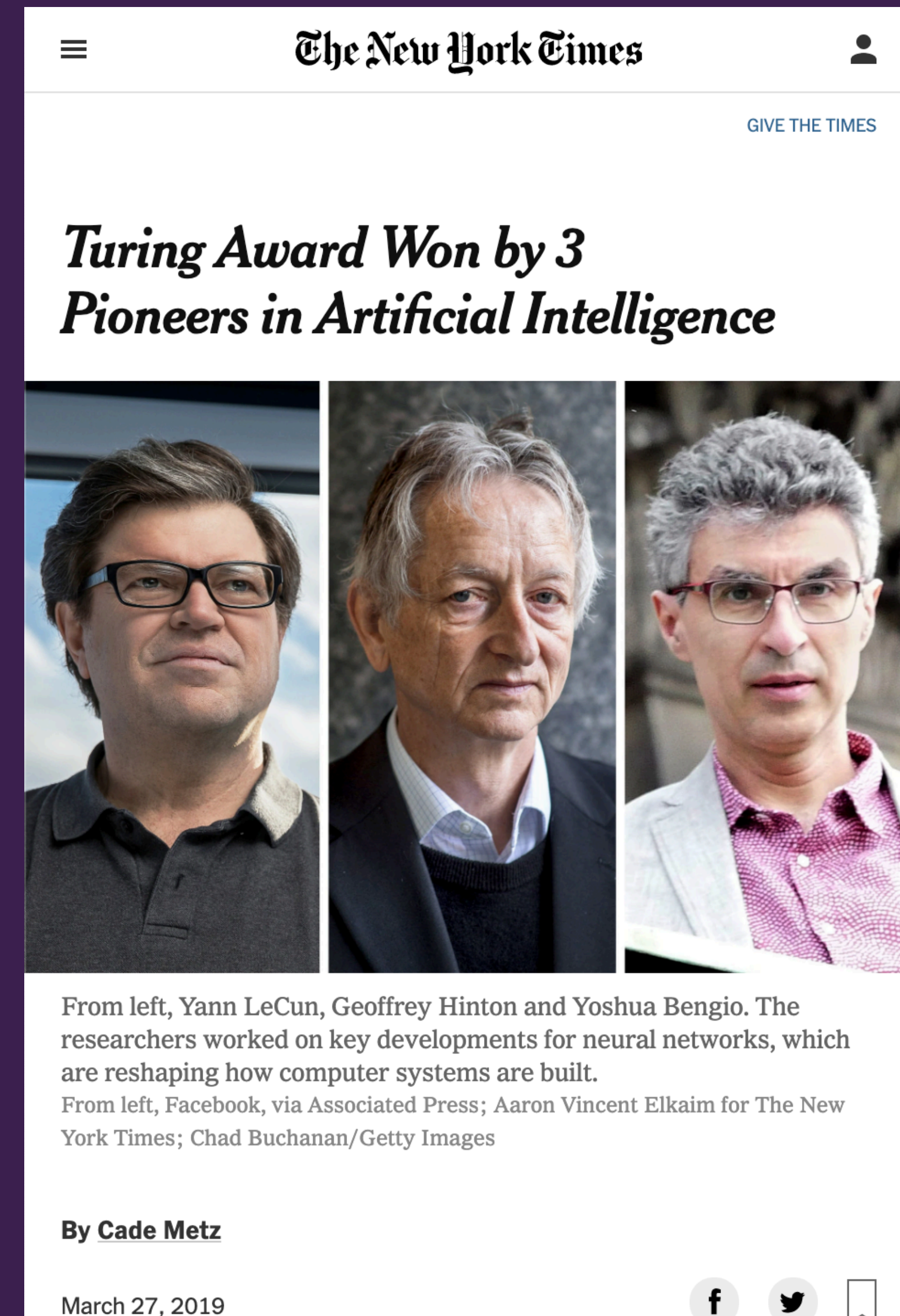
By **Cade Metz**

March 27, 2019

LeCun, Hinton, Bengio: deep learning

“No, let’s do it this way instead:” these networks learn extremely complex functions, so they need much more data than existing machine learning approaches, GPUs to train, and algorithms to enable them to learn more effectively.

Around 2010, these models began smashing records in speech and image recognition. They are now foundational to ML.






The screenshot shows the top portion of a New York Times article. At the top, the New York Times logo is centered, with a hamburger menu icon on the left and a user profile icon on the right. Below the logo, there is a "GIVE THE TIMES" link. The main headline reads "Turing Award Won by 3 Pioneers in Artificial Intelligence" in a bold, serif font. Below the headline are three side-by-side portrait photographs of the award winners: Yann LeCun on the left, Geoffrey Hinton in the middle, and Yoshua Bengio on the right. Underneath the photos is a short paragraph of text, followed by a byline "By Cade Metz" and the date "March 27, 2019". At the bottom right of the article preview, there are icons for Facebook, Twitter, and a bookmark.

The New York Times

GIVE THE TIMES

Turing Award Won by 3 Pioneers in Artificial Intelligence



From left, Yann LeCun, Geoffrey Hinton and Yoshua Bengio. The researchers worked on key developments for neural networks, which are reshaping how computer systems are built.
From left, Facebook, via Associated Press; Aaron Vincent Elkaim for The New York Times; Chad Buchanan/Getty Images

By **Cade Metz**

March 27, 2019

Not all research wins Turing Awards. But...

It all follows this same formula —

An implicit assumption: Industry and other researchers all thought one way about a problem

“No, let’s do it this way instead:” The researcher offered a new perspective that nobody had ever considered or made feasible before. They proved out their idea as the better approach.

And now, a definition.

Research introduces a fundamental **new idea** into the world.

Examples:

Simple instruction sets for complex computer architecture

Computing that is interactive, not batch

Algorithms needed to make deep learning effective

These ideas did not exist in any mature or well-articulated way before their creators developed them.

If the idea is already in the world, for example published by someone else, it is not considered **novel**, and thus not research.

Seeking a Better Way to Find Web Images

By JOHN MARKOFF NOV 19, 2012

Before: small computer vision datasets

After: huge computer vision dataset, and algorithms to utilize it

STANFORD, Calif. — You may think you can find almost anything on the Internet.

But even as images and video rapidly come to dominate the Web, search engines can ordinarily find a given image only if the text entered by a searcher matches the text with which it was labeled. And the labels can be unimpressive, unhelpful, or even nonexistent.

To eliminate those limits, scientists will need to create a new generation of visual search technologies. In this case, as the title of a computer scientist Fei-Fei Li recently put it, the Web will be in danger of “going dark.”

Now, a team of computer scientists from Princeton, Dr. Li, 38, has built the world’s largest visual database in an effort to mimic the human vision system. With more than 14 million labeled objects, from obsidian to orangutans to ocelots, the database has become a vital resource for computer vision researchers.

Before: programmers manually reserve resources for cloud computing

After: programmers provide needs, software allocates resources

CLOUD COMPUTING

Making Cloud Computing Systems More Efficient

By OLIVER SHARKEY MARCH 14, 2014 10:09 AM

Email

Share

Save

More



Dr. Kravakis, an associate professor of engineering and computer science at the University of California, Berkeley, headed the creation of management software called Quas...

Before: crowdsourcing is for workflows

After: crowdsourcing is for organizations

Mr. Valentine and Mr. Bernstein wanted to take the concept further. They created a platform, Foundry, in which the process of assembling and running a temporary organization could be automated, without so much as a phone call.



Stanford Researcher Finds Lots of Leaky Web Sites

By SAMANTHA STANK AND JAMES GORMAN OCTOBER 12, 2011 10:00 AM

Before: we think web tracking is isolated to the intended site

After: it's much leakier than we realized

The Web is porous. Remarkable information trickles in from everywhere. It also sometimes spills out without its users knowing exactly where or how.

Take for instance these findings, released on Tuesday by computer scientists at Stanford University. If you type a wrong password into the website of the White House, it turns out that your e-mail address quietly slips out to seven unrelated Web sites. Sign on to NBC and, likewise, seven other companies can capture your e-mail address. Click on an advertisement on oneDerma.com and your first name and user ID are instantly revealed to 13 other companies.

These findings, released by the Center for Internet and Society at Stanford Law School, are among the deals found on 185 top Web sites. They serve to buttress what privacy advocates have long

SCIENCETAKE

OceanOne, a Mer-Bot Dive Buddy With a 'Friendly Face'

Before: underwater robots should look and feel like boats

After: humanoid underwater robotics

ScienceTake | Meet the Humanoid Mer-Bot
By SAMANTHA STANK and JAMES GORMAN

Facebook Twitter



Research creates industry



PageRank algorithm



Stanford University Network
workstation (SUNet)



Computer graphics
architectures



Online education



Computer virtualization

Industry and research

Industry vs. research

What makes other start-ups and industry different than research?

If the core idea already exists, but needs to be refined in order to see success...it might be important, but it's not research.



Industry vs. research

Companies can and do engage in development that is research...

MapReduce and Spanner at Google

Kinect at Microsoft

...but typically companies are working to scale out ideas that exist.



Landay, 2000s:
activity sensing

Credit because he
developed the concept
and popularized it



Apple, 2010s:
Apple Watch

Credit because they
engineered it to work and
launched it

**(An incomplete list of)
research areas in
computer science**

Flavors of CS research

Computer science is field held together by a shared phenomenon of interest: computing.

This sets it apart from some other fields, which are drawn together by a shared theory or shared methodology. While this is a simplification, it is a helpful first cut:

Psychology: methodology of randomized experiment

Sociology: shared theories — functionalist perspective, conflict perspective, symbolic interactionist perspective

Math: methodology of formal proof

Anthropology: methodology of participant observation

Architecture

Artificial intelligence

Computational biology

Computer graphics

Computer security

Computer systems

Computer vision

Data science

Education

Human-computer interaction

Machine learning

Natural language processing

Networking

Operating/distributed systems

Programming systems/verification

Robotics

Theory

Topic: artificial intelligence

Architecture

Artificial intelligence

Computational biology

Computer graphics

Computer security

Computer systems

Computer vision

Data science

Education

Human-computer interaction

Machine learning

Natural language processing

Networking

Operating/distributed systems

Programming systems/verification

Robotics

Theory

Topic: computer systems

Architecture

Artificial intelligence

Computational biology

Computer graphics

Computer security

Computer systems

Computer vision

Data science

Education

Human-computer interaction

Machine learning

Natural language processing

Networking

Operating/distributed systems

**Programming systems/
verification**

Robotics

Theory

Topic: theory

Architecture

Artificial intelligence

Computational biology

Computer graphics

Computer security

Computer systems

Computer vision

Data science

Education

Human-computer interaction

Machine learning

Natural language processing

Networking

Operating/distributed systems

**Programming systems/
verification**

Robotics

Theory

Method: engineering

Architecture

Artificial intelligence

Computational biology

Computer graphics

Computer security

Computer systems

Computer vision

Data science

Education

Human-computer interaction

Machine learning

Natural language processing

Networking

Operating/distributed systems

Programming systems/ verification

Robotics

Theory

Method: probability and modeling

Architecture

Human-computer interaction

Artificial intelligence

Machine learning

Computational biology

Natural language processing

Computer graphics

Networking

Computer security

Operating/distributed systems

Computer systems

Programming systems/verification

Computer vision

Robotics

Data science

Theory

Education

Method: formal reasoning and proof

Architecture

Artificial intelligence

Computational biology

Computer graphics

Computer security

Computer systems

Computer vision

Data science

Education

Human-computer interaction

Machine learning

Natural language processing

Networking

Operating/distributed systems

**Programming systems/
verification**

Robotics

Theory

Method: design

Architecture

Artificial intelligence

Computational biology

Computer graphics

Computer security

Computer systems

Computer vision

Data science

Education

Human-computer interaction

Machine learning

Natural language processing

Networking

Operating/distributed systems

Programming systems/verification

Robotics

Theory

Method: empirical measurement and hypothesis testing

Architecture

Artificial intelligence

Computational biology

Computer graphics

Computer security

Computer systems

Computer vision

Data science

Education

Human-computer interaction

Machine learning

Natural language processing

Networking

Operating/distributed systems

Programming systems/verification

Robotics

Theory

Research mindset

Research is different than your usual coursework.

Coursework tends to be very clearly defined. Research tends to be exploratory and iterative.

You probably won't know the right answer: if we knew whether it was going to work, it wouldn't be research.

“I like” from research:

“The free-form structure of our project”

“The freedom to choose the questions and methods I find interesting”

“The independence I got in establishing a research direction”

“That I have had the opportunity to do a lot of self guided research and reading. I feel very free to shape parts of my learning and research experience.”

Research is a new and different skill. Embrace and navigate through the uncertainty.

“I wish” from research:

“That there was more structure or well-defined expectations.”

“I had a clearer idea or more deliverables and felt the barrier of being unfamiliar with certain parts of the project or coming on late less.”

“I had been able to narrow my scope a little earlier”

How this course works

Learning goals

Understand the major research topics currently active in your area. Be able to read a research paper and perform a literature review in that area.

Apply vectoring and velocity skills for navigating the open-ended nature of research.

Design and execute an appropriate evaluation of your method.

Write a paper and engage in the peer review process.

Learning goals

I97:

- Execute a first research project at the scale that can be submitted to a workshop or work-in-progress at a top-tier conference.

I97C:

- Onramp for your CURIS / PURE / ongoing research project

Is this course right for me?

CS197 is the best fit if you're...

Just starting your research journey

Done with CS 106B

CS197C is the best fit if you're...

Starting out on research with a {graduate student, post-doc, research staff, faculty} mentor

Hoping for extra guidance and support through your project

It's not the best fit if you're...

Looking for an advisor for your own research idea

Just looking for research units (connect with a lab and take CS 195)

197 & 197C: Assignments

Assignments offer waypoints in support of the project.

Assignment 1: learning about the project area, and learning how to read a paper

Assignment 2: literature review

Assignment 3: introduction

&etc.

All details are at cs197.stanford.edu and cs197c@cs.stanford.edu

I 97: Research project

This class is structured around a quarter-long research project. The project is completed in groups of 3-4 within a section.

TAs will offer project options tailored to each section and the students' interests within the section. These projects are designed to be accessible to you, of interest to the research community, and achievable within the timeline of the course.

“I have my own idea!”: mention it to your TA. We are unlikely to bend given those goals of accessibility, broader research interest, and achievability, but want to hear your ideas — it's possible!

197: Groups and projects

Form project teams and align with topics in section during Week 2.

Each section has two beginner-level, one advanced project.

After you join a section, your CA will match you to a project from a prepared list of options.

You will have some freedom to evolve the shape of that project...

...but we chose it to scope your project to something we know we can advise well, and that we think you can finish by the end of the quarter.

197C: CURIS / PURE / ongoing project

Small research milestone

- Can be preliminary work for upcoming CURIS internship
- Can be one “vector” to reduce risk in for current, ongoing project
- In weeks 1-2, work with your mentor to determine the milestone (<30 hours of work total, based on 3-unit courseload)
 - more info to come!

197 & 197C: Grading

Grades will be returned as ✓-, ✓, ✓+

- ✓- means “needs improvement”

- ✓ means “solid work”

- ✓+ means “wow, nice job!”

Rough benchmark: if you're getting ✓ on every assignment, you're on track for an A in the class

197 & 197C: Responding to TA

Your assignments are building toward a full research paper

Your TA will give feedback on each submission. To encourage mastery learning, we expect that you will address that feedback in your final version (final talk, final paper)

Look for feedback to be tagged as **required**, **recommended**, or **FYI**

If a section of the final paper improves on the original submission (e.g., Introduction assignment was a ✓, but the Introduction in the final paper is a ✓+), we will upgrade the original assignment grade

Conversely, final submissions that do not address **required** feedback will be penalized in the final talk and final paper grades

Attendance Policy

197:

Required weekly attendance:

Lecture

Team-only meeting

Section; Advising meetings with CA

197C:

Required weekly attendance:

Lecture

Check-in meeting with staff

Research meeting with mentor

One excused absence without penalty: must notify CA in advance and receive an acknowledgement.

No limit on absences due to uncontrollable circumstances (e.g., emergencies): must email staff ASAP.

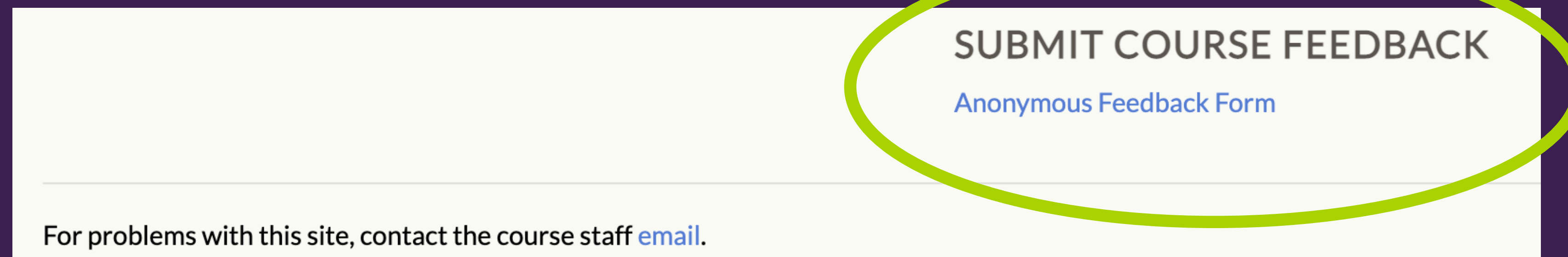
Feedback: CS197 & 197C

Option 1: High-resolution course evaluation

- You will get pinged by email three times at random dates this quarter to answer a few short questions.

Option 2: Submit anonymous feedback anytime

- Course website: scroll to bottom for link to Google form



Please give us feedback! We want to improve the course!

What after 197 & 197C?

Our goal is for CS 197 to be an onramp for you to research in Computer Science. We will:

- Perform outreach to faculty in CS or at Stanford to help introduce you so you can work on research projects after demonstrating excellence here

- Support you in submitting your work to flagship conferences, and connect you with funding opportunities to travel to present the work

Your TODOs

Course Application!

Both Due **1 hour** after class ends.

197: Section Application

Linked from cs197.stanford.edu or on Canvas.

Rank your section preferences

Decisions announced Wed

If you do not get your first choice, you can sign up for the waitlist.

197C: Course application

Linked from cs197c.stanford.edu or on Canvas.

Register your mentor!

Due next Tues, Apr 11 at 10am

Assignment 1: Read a paper!

197:

- Individual assignment
- Based on your assigned section:
 - Read a paper
 - Starter task
 - Project ranking form

197C:

- Ask your mentor for a paper to read (due this Thurs at 5pm)
- Schedule weekly check-in meetings with Sean/Lauren
- Read a paper

Questions?

Computer Science Research

Slide content shareable under a Creative Commons Attribution-NonCommercial 4.0 International License.