CS 8803-MDS

# Human-in-the-loop Data Analytics

Lecture 2

08/24/22

# Logistics

Office hour:

Kexin: Friday 12PM-1:30PM, Klaus 3410

Kaushik: Thursday 11AM-12PM, Klaus 3319

Piazza for online discussions:

https://piazza.com/gatech/fall2022/cs8803mds/home

Update on waitlist

# Today’s class

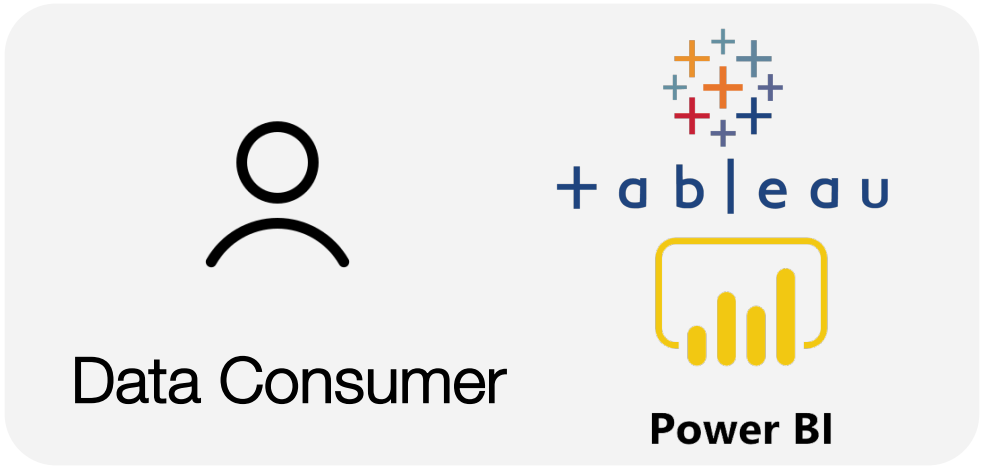
Overview of course topics

What is research?

How to develop a new idea?

How to read a paper?

# Human roles in data analytics



* 3-4 papers each topic

# Part I: Data Scientists

INTERACTIVE SQL

INTERACTIVE VISUALIZATION

DATA SCIENCE NOTEBOOKS

# Part I: Data Scientists

## Interactive SQL

[BlinkDB: Queries with Bounded Errors and Bounded Response Times on Very Large Data]

[AQP++: Connecting Approximate Query Processing With Aggregate Precomputation for Interactive Analytics]

[Experiences with Approximating Queries in Microsoft's Production Big-Data Clusters]

# Part I: Data Scientists

## Interactive Visualization

Database Benchmarking for Supporting Real-Time Interactive Querying of Large Data

Hillview: A trillion-cell spreadsheet for big data

M4: A Visualization-Oriented Time Series Data Aggregation

# Part I: Data Scientists

## Data Science Tools

[Benchmarking Spreadsheet Systems]

[Finding Related Tables in Data Lakes for Interactive Data Science]

[Auto-Suggest: Learning-to-Recommend Data Preparation Steps Using Data Science Notebooks]

[Towards Effective Foraging by Data Scientists to Find Past Analysis Choices]

# Part II: Data Consumer

EXPLANATION

RECOMMENDATION

INTERFACES

# Part II: Data Consumer

## Explanation

MacroBase: Prioritizing Attention in Fast Data

Slice Finder: Automated Data Slicing for Model Validation

ExplainIt! – A Declarative Root-cause Analysis Engine for Time Series Data

# Part II: Data Consumer

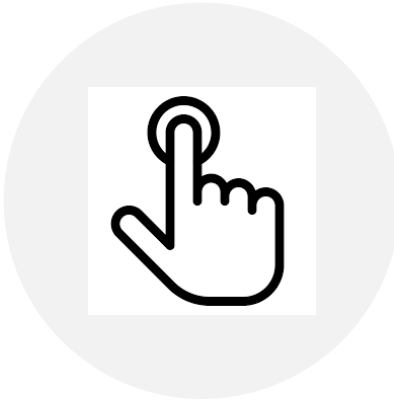
## Recommendation

SeeDB: efficient data-driven visualization recommendations to support visual analytics

Lux: Always-on Visualization Recommendations for Exploratory Dataframe Workflows

Controlling false discoveries during interactive data exploration

# Part II: Data Consumer



Interfaces

Vega-lite: A grammar of interactive graphics

Expressive Time Series Querying with Hand-Drawn Scale-Free Sketches

Falx: Synthesis-Powered Visualization Authoring

# Schedule

## Introduction

| Date | Topic | Content | Presentor |
| --- | --- | --- | --- |
| 08/22 | Introduction | Course Introduction and Logistics | Kexin Rong |
| 08/24 | Introduction | Research Skills | Kexin Rong |

## Part II - Data Consumer

| Date | Topic | Content | Presentor |
| --- | --- | --- | --- |
| 10/12 | Overview | Part II topics overview | Kexin Rong |
| 10/17 | No Class (Fall Break) | | |
| 10/19 | Explanation | MacroBase: Prioritizing Attention in Fast Data | |

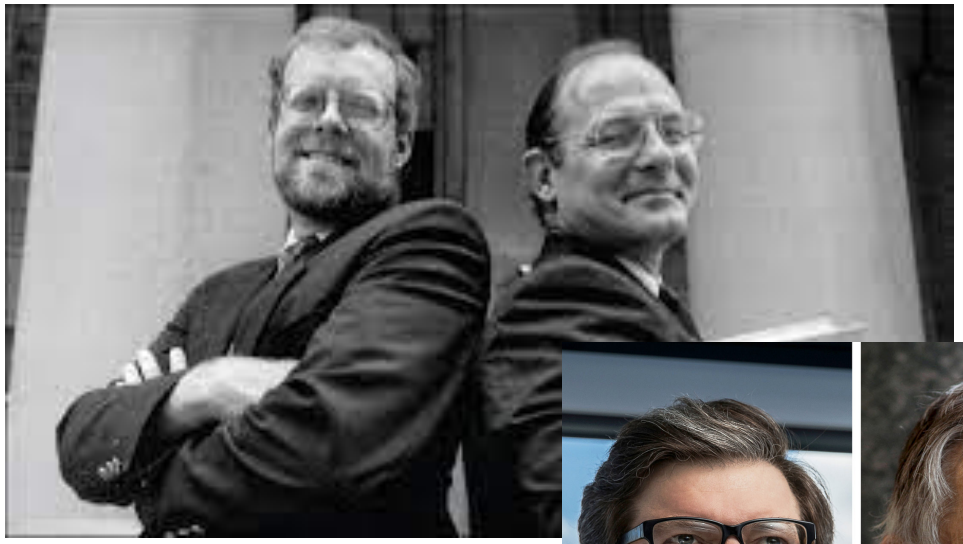
# Computer Science Research

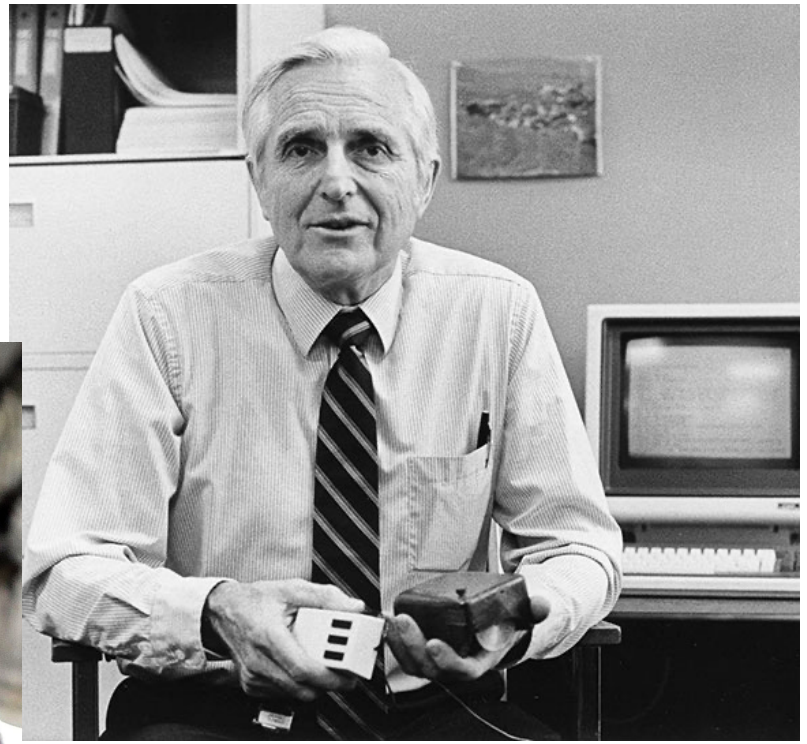
What is the goal of research?

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.

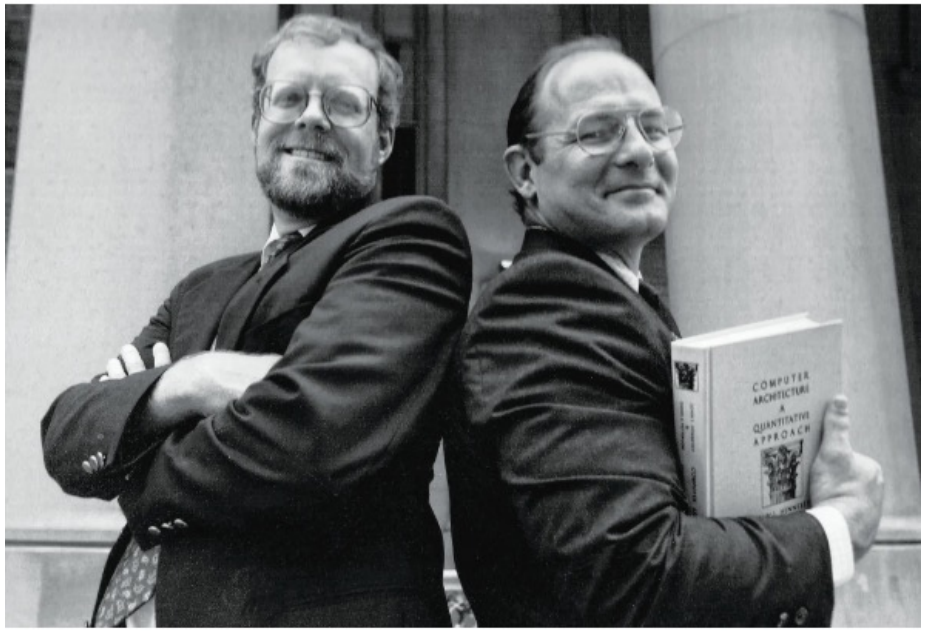
Everyone thought that *increasingly powerful processors needed increasingly complicated instruction* sets to take advantage of them.

Adapted from Stanford CS197

The New York Times

## *Computer Chip Visionaries Win Turing Award*

Give this article




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

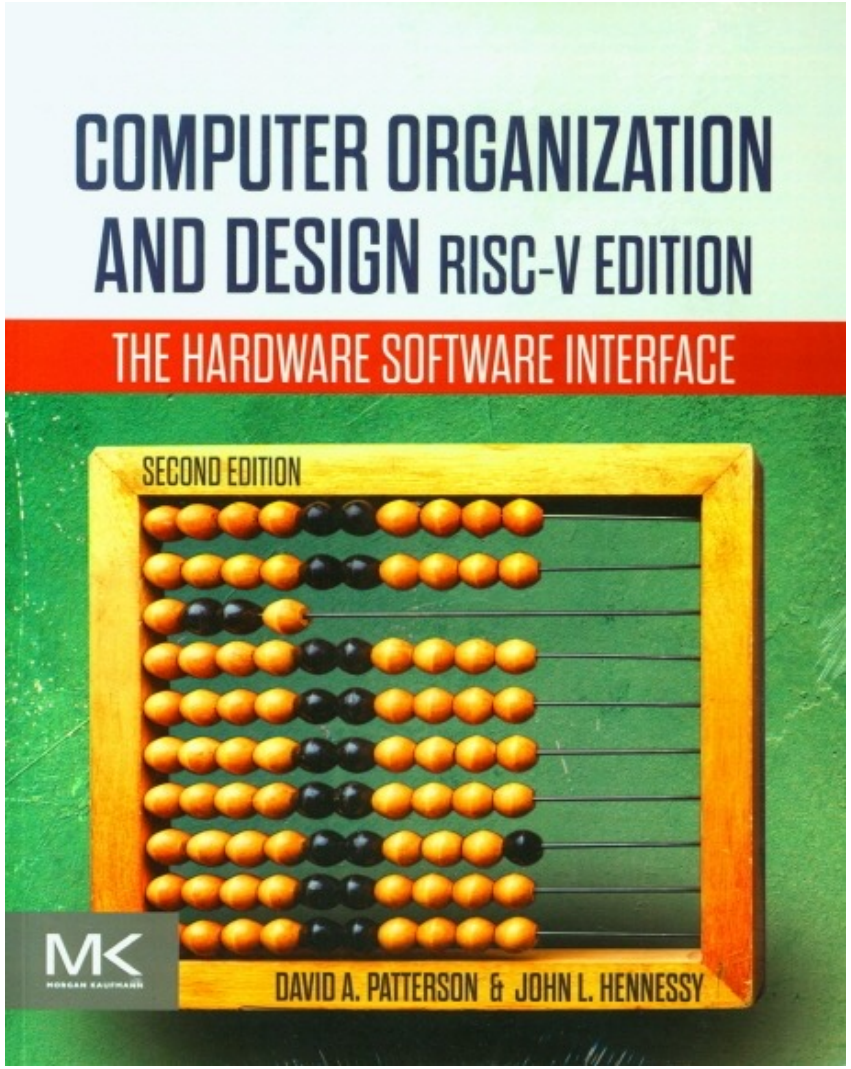
March 21, 2018

SAN FRANCISCO — In 1980, Dave Patterson, a computer science professor, looked at the future of the world's digital machines and saw their limits.

# 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). Today, more than 99 percent of all new chips use the RISC architecture they developed.



# 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 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,

# 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.

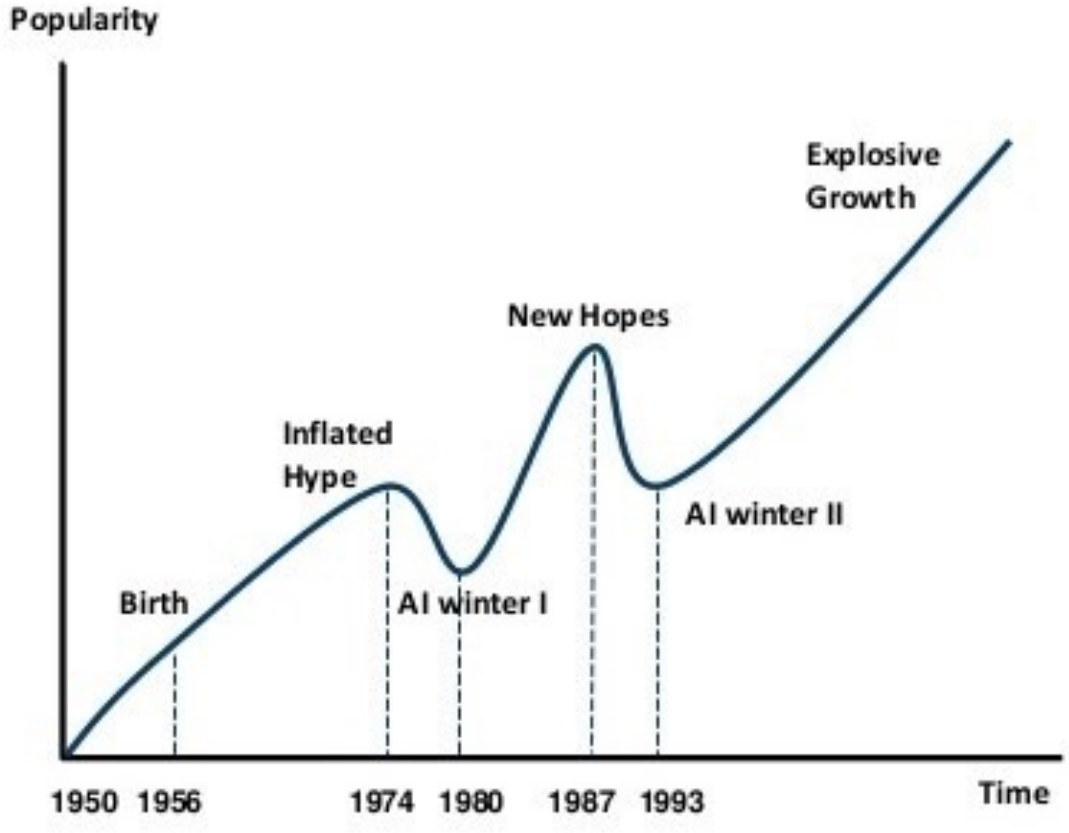
This led to Xerox Star. Steve Jobs saw it, was wow'ed, and infused the ideas into the Mac.

# Engelbart: interactive computing

# 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.

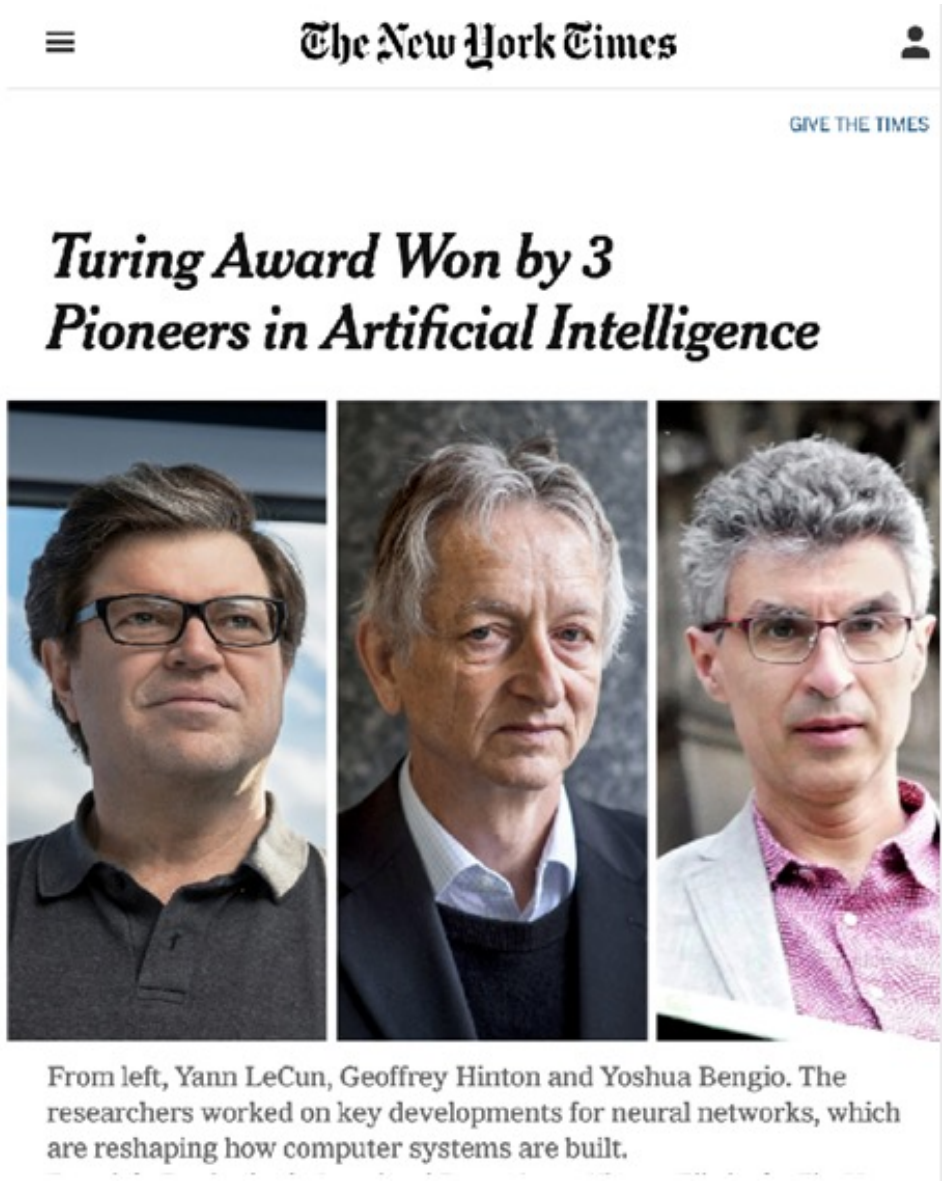
Instead, machine learning was based on other models, for example the support vector machine and graphical models. Neural networks did not perform well.



# 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.



# Not all research wins Turing Awards. But…

It all follows the 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.

# What is research?

Research introduces a fundamental new idea into the world.

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.

# How to develop a novel idea?

Novel ideas rarely come out of a vacuum

They're much more often pivoted off of today's work:

A realization that an idea has been applied in domains like X and needs to be rethought in domains like ~X

A recognition that others have tried this technique in users of context A, or data of up to size N, but ~A or >>N breaks the technique.

Some constraint that exists but shouldn't, or visa versa

# The "bit flip" method: invert an assumption

bit flip: an inversion of an assumption that the world has about how the world is supposed to work.

Recipe for a bit flip:

1) Define the bit: articulate an assumption, often left implicit in prior work
2) Introduce the flip: argue for an alternative to that assumption / "No, let's do it this way instead"

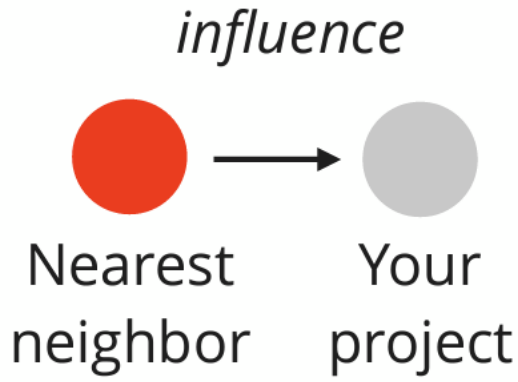
| Bit | Flip | Project |
| --- | --- | --- |
| We need complicated instruction sets to accommodate powerful computer processors. | Simple instruction sets are better since they let you compare performance, optimize, and prevent errors. | RISC architecture |
| Computing was just for numerical calculations: slow, done in batches, and for teams. | Computing should be interactive, individual, and support thought. | Mother of all demos |
| Neural networks exist, but don't perform very well and aren't accurate. | We need more data and different algorithms for this to work. | Deep learning |

| Bit | Flip | Project |
| --- | --- | --- |
| A minimum graph cut algorithms should always return correct answers. | A randomized, probabilistic algorithm will be much faster, and we can still prove a limited probability of an error. | Karger's algorithm |
| Activity tracking requires custom hardware. | Activity tracking requires just a standard cell phone. |  |
| NLP machine learning models should read sentences word by word | Models should consume the entire sentence at once | BERT |

# Single paper bit slip

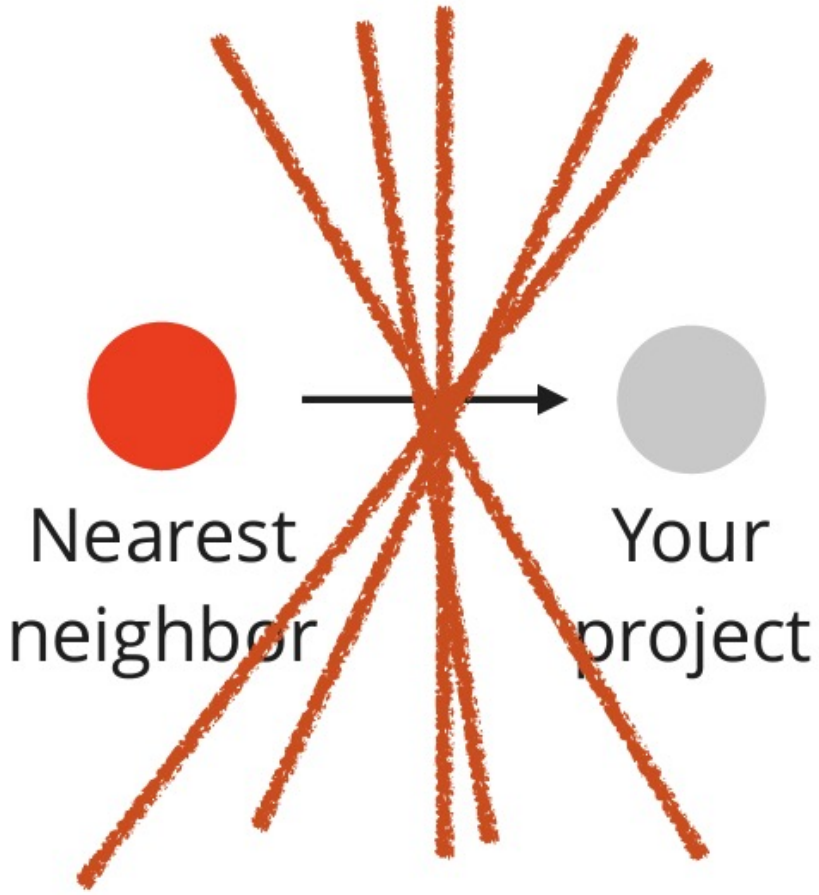
Find a paper that is adjacent to your idea. Think of this as your nearest neighbor paper.

Your project will be some sort of delta off of that paper. What assumption or limitation did it have, that you're erasing?




Adapted from Stanford CS197

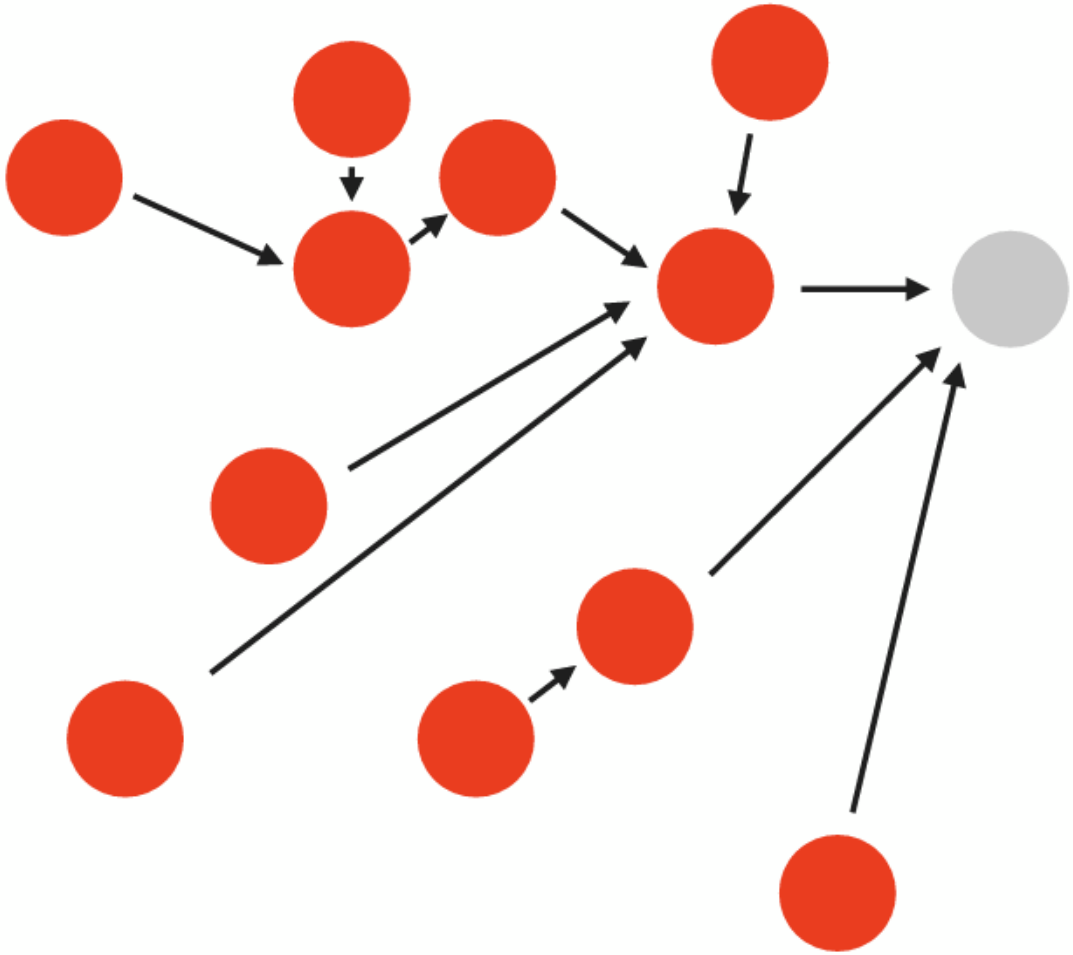
# Single paper bit slip




Each separating line is a possible bit flip.

Which one should you go for?

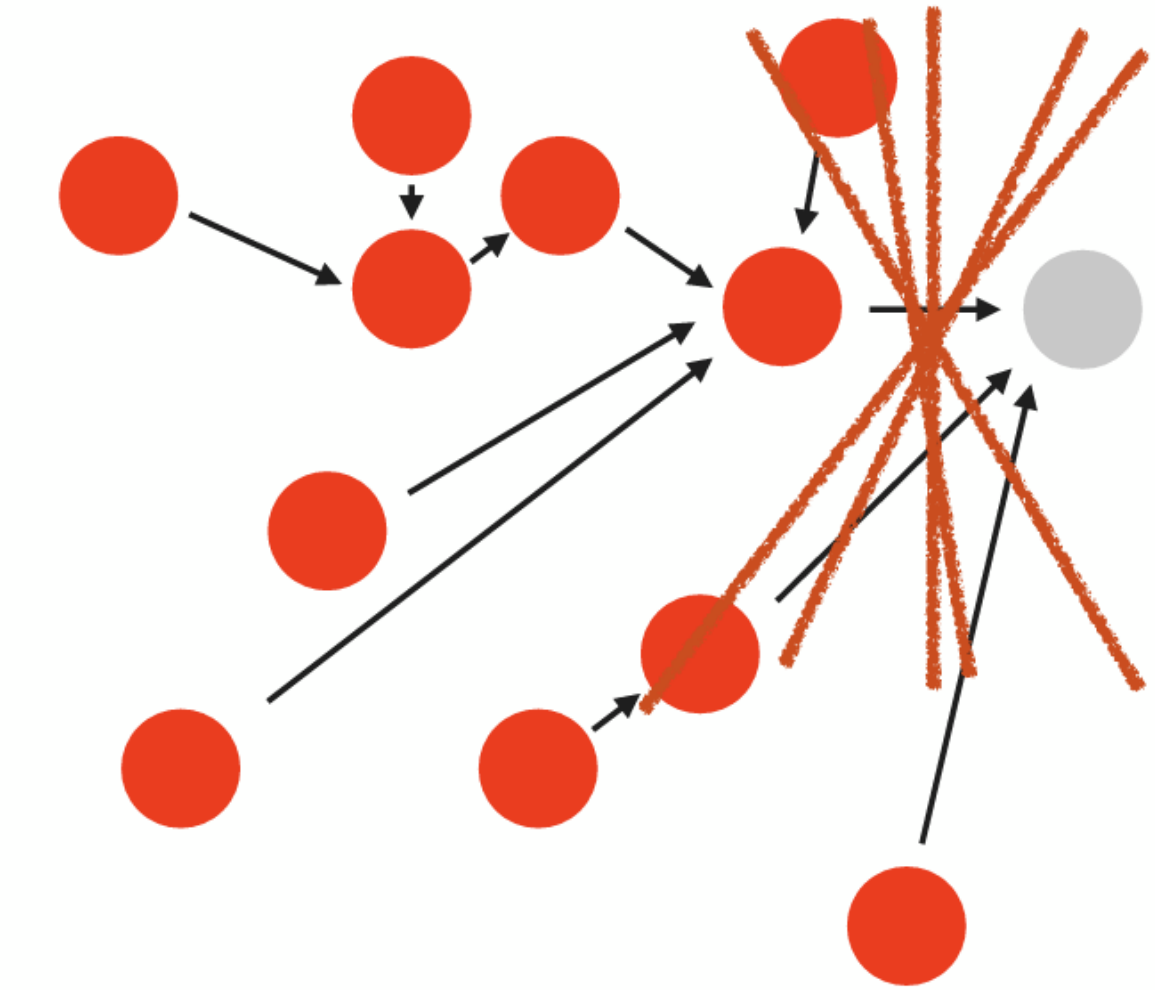
# Literature bit flip



The broader an understanding you have of the literature and the design axes underneath it, the more effectively you can pick the right bit flip.

Adapted from Stanford CS197

# Literature bit flip



The broader an understanding you have of the literature and the design axes underneath it, the more effectively you can pick the right bit flip.

Adapted from Stanford CS197

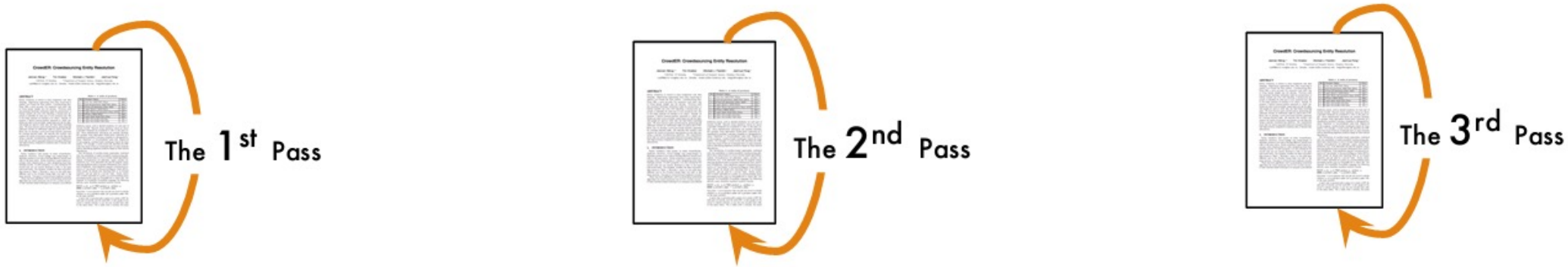
# How to read a paper

The "three-pass" approach [1]

first pass: a quick scan

second pass: with greater care, but ignore the details

third pass: re-implementing the paper




[1] S. Keshav. How to read a paper? http://blizzard.cs.uwaterloo.ca/keshav/home/Papers/data/07/paper-reading.pdf

# The first pass: a quick scan

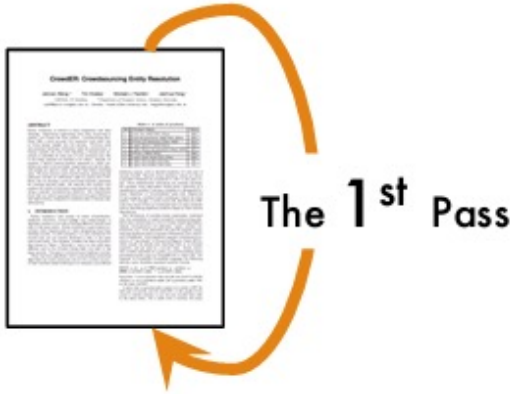
Goal: get bird's-eye view of the paper (5~10 min)

What to read:

- Title, abstract, introduction and conclusion
- Section and sub-section headings
- Main figures
- Scan of bibliography

You should be able to answer:

- What type of paper is this?
- What are the main contributions?



# Let’s try it!

1. Category: What type of paper is this? A measurement paper? An analysis of an existing system? A description of a research prototype?
2. Context: Which other papers is it related to? Which theoretical bases were used to analyze the problem?
3. Correctness: Do the assumptions appear to be valid?
4. Contributions: What are the paper’s main contributions?
5. Clarity: Is the paper well written?

# The second pass: grasp the content

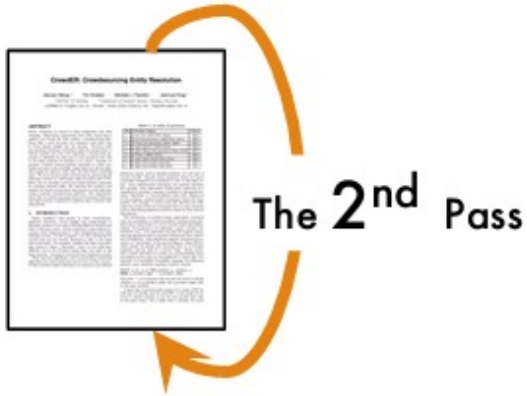
Goal: get a good understanding of the ”meat” of the paper

How to read:

- Look carefully at figures, diagrams and examples
- Take notes of questions, unread references etc.
- Ignore proofs, appendix, extensions etc.

You should be able to:

- Summarize main thrusts of the paper, with supporting evidence, to someone else



# The third pass: all about the details

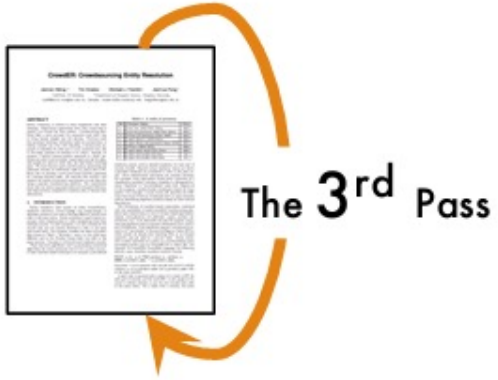
Goal: think about what you would have done if you were to re-implement such an idea

How to read:

- Challenge every assumption
- Compare your version with the actual paper
  - Often leads to questions like: why not do it this way?

You should be able to:

- Identify hidden assumptions/potential design flaws
- Get ideas for future work



# How to do a literature review

1. Pick your favorite academic search engine (e.g., Google scholar) and start with keywords
2. Find 3-5 recent and highly cited papers
    - From reputable venues and by reputable institution/author
    - If you find a survey paper, start from the survey paper
3. Do the first pass to identify key papers and researchers that these works cite
4. Track down these papers/researchers
5. Iterate as needed

# Tools

**Backward influence**: influential citations in the papers that you've read

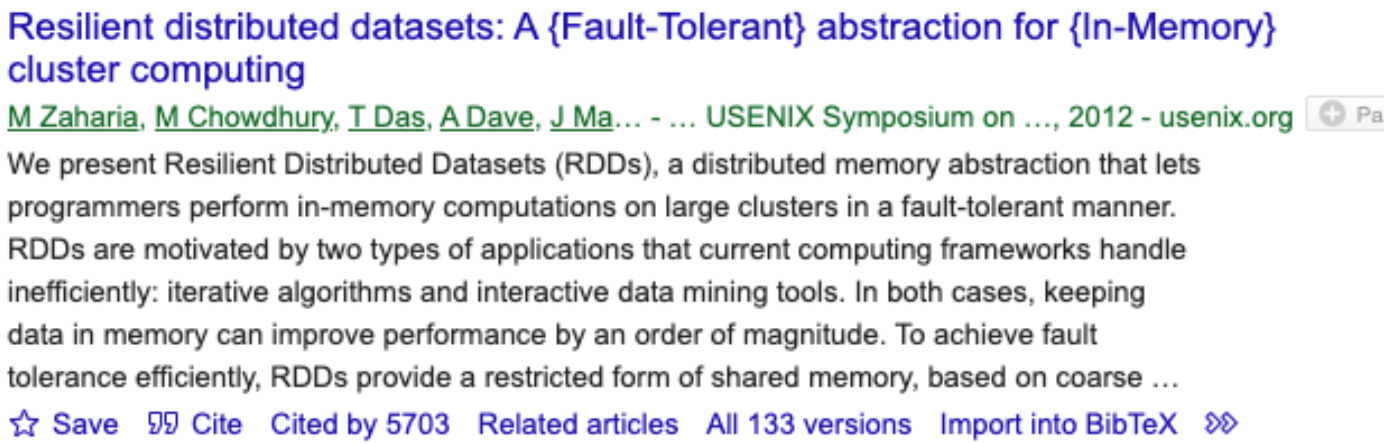
How: reading

**Forward influence**: papers citing the ones that you've read

How : Google Scholar's "Cited By"

**Relatedness**: contemporaneous but not citing

How: Google Scholar's "Related articles"



# Top conferences and journals

Databases and Data Management:

VLDB, SIGMOD, ICDE

Data Mining:

KDD, WWW

HCI:

CHI, UIST, InfoVis

Machine Learning:

NeurIPS (formerly NIPS), ICML, ICLR, AAAI

CVPR (vision), ACL (NLP), EMNLP (NLP)

# Next class

Intro to approximate query processing

Intro to user studies and usability evaluations

**Your task:**

Sign up for presentation: https://bit.ly/3CnBOSa

Post on Piazza to look for teammates

First paper review due next Tuesday (08/30) midnight