big data

data systems

M. LEARNING

DATA SCIENCE

BLOCKCHAIN

IMAGES/VIDEO

ANALYTICS
~6 decades of research
IBM, Microsoft, Oracle, Teradata, etc.
and a gazillion start-ups today

declarative interface
ask “what” you want

↓   ↑

data* system

the system decides
“how” to best store
and access data
~6 decades of research
IBM, Microsoft, Oracle, Teradata, etc., and a gazillion start-ups today

declarative interface
ask “what” you want

↓

data* system

↑

the system decides “how” to best store and access data

why is this good
learning outcome
Fundamentals of storage
learning outcome
Fundamentals of storage

data structures, SQL, NoSQL, Neural Networks, Statistics, Images, Blockchain
learning outcome
Fundamentals of storage

data structures, SQL, NoSQL, Neural Networks, Statistics, Images, Blockchain
same set of principles across all fields (performance: design & implementation)
learning outcome
Fundamentals of storage

data structures, SQL, NoSQL, Neural Networks, Statistics, Images, Blockchain
same set of principles across all fields (performance: design & implementation)

from algorithms to systems
software engineering   data-driven startup   research
HOW TO STORE DATA
ALGORITHMS

data structure decisions define the algorithms that access data
ALGORITHMS

unordered

[7, 4, 2, 6, 1, 3, 9, 10, 5, 8]
ALGORITHMS

unordered [7,4,2,6,1,3,9,10,5,8]

INDEX

DATA
DATA STRUCTURES
DEFINE PERFORMANCE

[Graph showing the trend of Compute and Data Movement speed in 2023]

speed

2023

DATA

STRUCTURES

DEFINE

PERFORMANCE
register = this room

memory = nearby city

disk = Pluto

Jim Gray, Turing Award 1998
no perfect structure
no perfect structure
no perfect structure
no perfect structure

- Array
- Hash-Table
- Linked-List
- Sorted Array
- Skip-List
- Trie
- B-tree
Parameters/Context that affect design

1. Workload
2. Hardware
3. Desired Performance/Cloud Cost
Parameters/Context that affect design

1. Workload
2. Hardware
3. Desired Performance/Cloud Cost

Adaptivity —> Automated Systems Design
Registers on chip cache

On board cache

Memory

Disk

CPU

Cache miss: looking for something which is not in the cache

~1ns

~10ns

~100ns

Memory miss: looking for something which is not in memory

Memory wall

Faster

Cheaper

Speed

Time

CPU

Mem
registers
on chip cache
on board cache
memory
disk
CPU

Faster

cheaper

~1ns

~10ns

~100ns

CPU

Speed

Cache Miss: looking for something which is not in the cache

Memory Miss: looking for something which is not in memory

DONT MISS!
need to only read x... but have to read all of page 1

data value x

page1  page2  page3  ...

data move

<table>
<thead>
<tr>
<th>CPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>registers</td>
</tr>
<tr>
<td>on chip cache</td>
</tr>
<tr>
<td>on board cache</td>
</tr>
<tr>
<td>memory</td>
</tr>
<tr>
<td>disk</td>
</tr>
</tbody>
</table>
query \ x<5

(size=120 bytes)

memory level N

memory level N-1

page size: 5x8 bytes
query $x < 5$

memory level N

memory level N-1

(scan)

(size=120 bytes)

5 10 6 4 12

page size: 5x8 bytes

2 8 9 7 6

7 11 3 9 6

...
query $x < 5$

Scan memory level $N$

(size=120 bytes)

memory level $N-1$

pagesize: 5x8 bytes
query $x < 5$

scan

(memory level N)

5 10 6 4 12

(size=120 bytes)

memory level N-1

5 10 6 4 12

2 8 9 7 6

7 11 3 9 6

...
query $x < 5$

memory level N

5 10 6 4 12

scan

memory level N-1

2 8 9 7 6

4

(size=120 bytes)

page size: 5x8 bytes

...
query $x < 5$

memory level N

memory level N-1

page size: 5x8 bytes
**query**: $x < 5$

- **memory level N**: 5 10 6 4 12
- **memory level N-1**: 2 8 9 7 6

Page size: 5x8 bytes

(scan) 80 bytes
query $x < 5$

(memory level $N$

(size=120 bytes)

(memory level $N-1$

/page size: 5x8 bytes

80 bytes
query $x < 5$

(memory level N)

scan

(size=120 bytes)

memory level N-1

page size: 5x8 bytes
query \( x < 5 \)

(scan)

[size=120 bytes]

memory level N

memory level N-1

page size: 5x8 bytes
query $x < 5$

(scan)

(size=120 bytes)
memory level N

memory level N-1

page size: 5x8 bytes
an oracle gives us the positions

query \( x < 5 \)

(size=120 bytes)

memory level N

memory level N-1

page size: 5x8 bytes
an oracle gives us the positions

query \( x < 5 \)

memory level N

memory level N-1

page size: 5x8 bytes
an oracle gives us the positions

query $x < 5$

(memory level N)

(page size: 5x8 bytes)
an oracle gives us the positions

query \( x < 5 \)

memory level N

memory level N-1

page size: 5x8 bytes
an oracle gives us the positions

query: $x < 5$

memory level $N$

(memory level $N$)

(page size: 5x8 bytes)
an oracle gives us the positions

query \( x < 5 \)

 remembering level \( N \)

\[
\begin{array}{cccccc}
5 & 10 & 6 & 4 & 12 \\
\end{array}
\]

\[
\begin{array}{cccccc}
2 & 8 & 9 & 7 & 6 \\
\end{array}
\]

\[
\begin{array}{cccccc}
7 & 11 & 3 & 9 & 6 \\
\end{array}
\]

source: \( 40 \) bytes

page size: \( 5 \times 8 \) bytes
an oracle gives us the positions

query \( x<5 \)

(size=120 bytes)

memory level N

memory level N-1

page size: 5x8 bytes
an oracle gives us the positions

\[ \text{query } x < 5 \]

(size=120 bytes)
memory level N

\[ \begin{align*}
5 & \quad 10 & \quad 6 & \quad 4 & \quad 12 \\
2 & \quad 8 & \quad 9 & \quad 7 & \quad 6
\end{align*} \]

memory level N-1

\[ \begin{align*}
4 & \quad 2 & \quad 7 & \quad 11 & \quad 3 & \quad 9 & \quad 6 \\
2 & \quad 8 & \quad 9 & \quad 7 & \quad 6 & \quad 7 & \quad 11 & \quad 3 & \quad 9 & \quad 6
\end{align*} \]

page size: 5x8 bytes

80 bytes
an oracle gives us the positions

query $x < 5$

memory level $N$

(size = 120 bytes)

oracle

7 11 3 9 6 2 8 9 7 6 4 2

memory level $N-1$

5 10 6 4 12 2 8 9 7 6 7 11 3 9 6 ...

page size: 5x8 bytes

80 bytes
an oracle gives us the positions

query $x < 5$

memory level N

oracle

2 8 9 7 6

4 2 3

(size=120 bytes)

memory level N-1

5 10 6 4 12

2 8 9 7 6

7 11 3 9 6

page size: 5x8 bytes

80 bytes
an oracle gives us the positions

query $x < 5$

<table>
<thead>
<tr>
<th>oracle</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 11 3 9 6</td>
</tr>
<tr>
<td>2 8 9 7 6</td>
</tr>
<tr>
<td>4 2 3</td>
</tr>
</tbody>
</table>

(size=120 bytes)

memory level N

memory level N-1

| 5 10 6 4 12 |
| 2 8 9 7 6   |
| 7 11 3 9 6  |
| ...         |

page size: 5x8 bytes
when does it make sense to have an oracle
how can we minimize the cost

e.g., query \( x < 5 \)
algorithm/system design = not just computation
How do I make my **data system** run x times as fast? (sql,nosql,bigdata, ...)

DALab
@ Harvard SEAS
How do I make my data system run x times as fast? (sql, nosql, bigdata, ...)

How do I minimize my bill in the cloud?
How do I make my data system run x times as fast? (sql, nosql, bigdata, ...)

How do I minimize my bill in the cloud?

How can I do 10x Blockchain transactions?
How do I make my **data system** run x times as fast? (sql,nosql,bigdata, ...)

How do I minimize my **bill** in the **cloud**?

How can I do 10x **Blockchain transactions**?

How to accelerate **statistics** computation for data science/ML?
How do I make my **data system** run x times as fast? (sql,nosql,bigdata, ...)

How do I minimize my **bill** in the **cloud**?

How can I do 10x **Blockchain transactions**?

How to accelerate **statistics** computation for data science/ML?

How do I train my **neural network** x times faster?
NEW APPLICATIONS
NEW APPLICATIONS

existing systems need to change too
NEW APPLICATIONS

existing systems need to change too
NEW APPLICATIONS

existing systems need to change too

WORKLOAD  HARDWARE  ADAPT

IMPROVE WITHIN A BUDGET

REASON

WHAT WILL BREAK MY SYSTEM?
more data

continuous need for new storage solutions

new applications

new h/w
basic CS265 logistics
first 4 weeks:
Basic background
Introduction to research problems
Adaptivity vision (self-designing)
Research thinking
as of week 5: discussions/
presentations
as of week 5: discussions/presentations

interaction: in and out of class
DailyOH/labs, Sat/Sun remote OH
There is no such thing as a wrong question/answer!!!!

as of week 5: discussions/presentations

interaction: in and out of class
DailyOH/labs, Sat/Sun remote OH
Recent Research Papers

Each student:
2 reviews per week/1 presentation

review and slides should focus on

- what is the problem
- why is it important
- why is it hard
- why existing solutions do not work
- what is the core intuition for the solution
- solution step by step
- does the paper prove its claims
- exact setup of analysis/experiments
- are there any gaps in the logic/proof
- possible next steps

* follow a few citations to gain more background
Recent Research Papers

- review and slides should focus on
  - what is the problem
  - why is it important
  - why is it hard
  - why existing solutions do not work
  - what is the core intuition for the solution
  - solution step by step
  - does the paper prove its claims
  - exact setup of analysis/experiments
  - are there any gaps in the logic/proof
  - possible next steps

- * follow a few citations to gain more background

Each student:
2 reviews per week/1 presentation

learn to judge constructively

learn to present

learn to prepare slides
<table>
<thead>
<tr>
<th>Semester Project: Due in the end of semester + A midway check in (early March, 10%)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Systems Project</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>Research Project</strong></td>
</tr>
</tbody>
</table>
semester project: due in the end of semester + a midway check in (early March, 10%)

**systems project**

**individual project**

**NoSQL**, in c/c++

**research project**
semester project: due in the end of semester + a midway check in (early March, 10%)

systems project

individual project
NoSQL, in c/c++

research project

groups of max three
Adaptivity/Performance
Across all subject areas
semster project: due in the end of semester + a midway check in (early March, 10%)

systems project

individual project

**NoSQL**, in c/c++

research project

groups of max three

Adaptivity/Performance

Across all subject areas

Adaptive Denormalization
Evolving Trees
Splaying LSM-Trees
Adaptive NoSQL
Adaptive Filters
Distributed Deep Learning
all announcements & discussions
as of week 2
link on class website - check out usage guidelines
all announcements & discussions
as of week 2
link on class website - check out usage guidelines

classes are recorded
(links on canvas)
all announcements & discussions
as of week 2
link on class website - check out usage guidelines

- Project: 40%
- Midway Check-in: 10%
- Discussion: 20%
- Presentation: 15%
- Reviews: 15%

classes are recorded
(links on canvas)
all announcements & discussions
as of week 2
link on class website - check out usage guidelines

Project: 40%
Midway Check-in: 10%
Discussion: 20%
Presentation: 15%
Reviews: 15%

classes are recorded
(links on canvas)

NO LAPTOP/PHONE POLICY
class is based on participation!
Teaching Fellows

Subarna Chatterjee
Teaching Fellow
(Room: SEAS 4.435)

Utku Sirin
Teaching Fellow
(Room: SEAS 4.435)

Hao Jiang
Teaching Fellow
(Room: SEAS 4.435)

Sanket Purandare
Teaching Fellow
(Room: SEAS 4.435)
Prerequisites

knowledge of algorithms, data structures, hardware, systems
Prerequisites

knowledge of algorithms, data structures, hardware, systems

Research track:
open to CS165 students
after discussion also CS161 and systems PhDs

Systems track allows taking the class without all prerequisites (but at least CS61)
Get familiar with the very basics of traditional database architectures:

Get familiar with very basics of modern database architectures:

Get familiar with the very basics of modern large scale systems:

Check out: syllabus, preparation readings, project 0, systems project, online sections

http://daslab.seas.harvard.edu/classes/cs265/