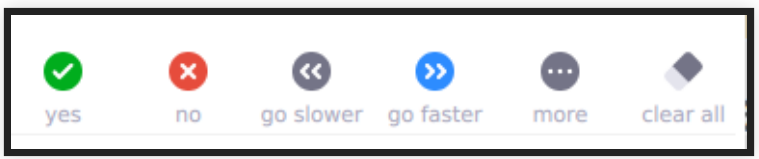
MACHINE LEARNING IN PRODUCTION / AI ENGINEERING

MOTIVATION, SYLLABUS, AND INTRODUCTIONS

Spring 2022, Eunsuk Kang & Christian Kaestner

WE ARE BACK ONLINE :(

If you can hear me, open the participant panel in Zoom and check "yes"



THIS IS NOT NORMAL. WE UNDERSTAND.

Speaker notes

Picture AP Photo/Noah Berger, https://www.abc27.com/news/thats-2020-photographers-california-wildfire-image-a-sign-of-the-times/

THIS IS NOT NORMAL. WE UNDERSTAND.

Expect:

- Internet and bandwidth issues
- Timezone issues?
- Distractions -- parents, siblings, pets
- Feeling isolated, feeling overwhelmed
- Additional sources of stress
- Hard time dealing with -gestures widely- *everything*...

Talk to us about accommodations of any kind

SIMULATING IN-CLASS EXPERIENCE

Discussions and interactions are important. We'll have regular in-class discussions and exercises

- Use chat, "raise hand" feature, or just speak
- If possible, keep camera on, muted by default
- Set preferred name in Zoom
- Synchronous "live" attendance only

- Suggestion: Have chat and participant list open, maybe separate window for gallery view for faces, second monitor highly recommended
- **Contact us for accommodations!**

CATASTROPHIC SUCCESS

PERSONAL CONNECTION

This is hard. We know.

- Talk inside and outside of class
- We are here always 10 min before class and stay after class if you have questions, want to chat
- We encourage collaboration in all assignments, even "individual" assignments and reading quizzes
- We encourage social activities in teams

LEARNING GOALS

- Understand how ML components are parts of larger systems
- Illustrate the challenges in engineering an ML-enabled system beyond accuracy
- Explain the role of specifications and their lack in machine learning and the relationship to deductive and inductive reasoning
- Summarize the respective goals and challenges of software engineers vs data scientists
- Explain the concept and relevance of "T-shaped people"

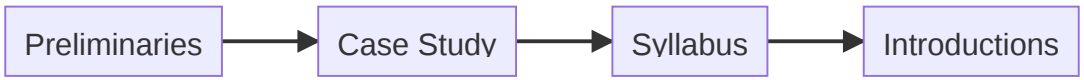
DISCLAIMERS

This class captures a rapidly evolving field.

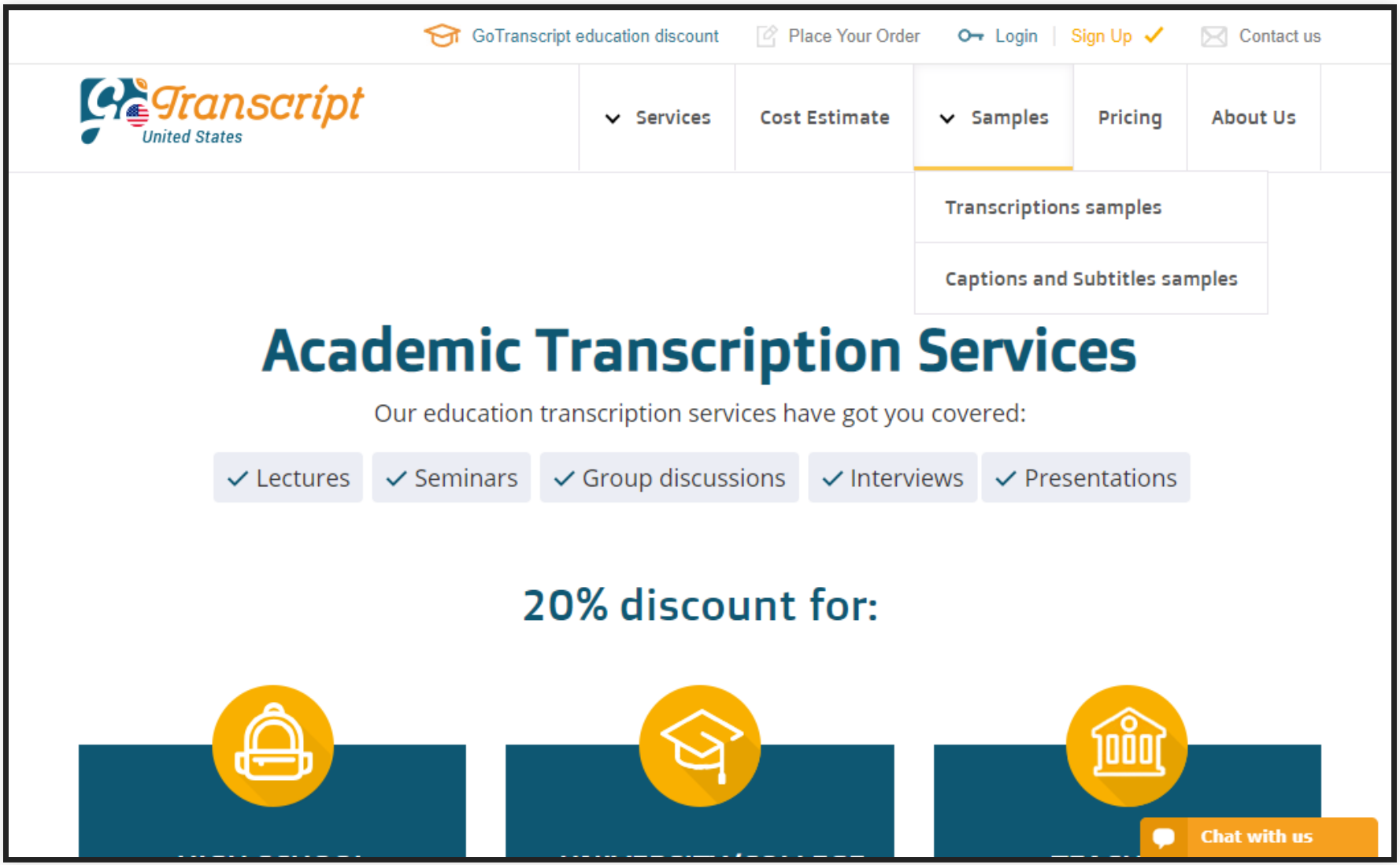
We are scaling from 30 to 150 students. Expect some friction.

We are software engineers.

AGENDA



CASE STUDY: THE TRANSCRIPTION SERVICE STARTUP


GoTranscript education discount
Place Your Order
Login
Sign Up
Contact us
Transcript
United States
Services
Cost Estimate
Samples
Pricing
About Us
Transcriptions samples
Captions and Subtitles samples
Academic Transcription Services
Our education transcription services have got you covered:
✓ Lectures
✓ Seminars
✓ Group discussions
✓ Interviews
✓ Presentations
20% discount for:
Chat with us

TRANSCRIPTION SERVICES

- Take audio or video files and produce text.
    - Used by academics to analyze interview text
    - Podcast show notes
    - Subtitles for videos
- State of the art: Manual transcription, often mechanical turk (1.5 $/min)

THE STARTUP IDEA

PhD research on domain-specific speech recognition, that can detect technical jargon

DNN trained on public PBS interviews + transfer learning on smaller manually annotated domain-specific corpus

Research has shown amazing accuracy for talks in medicine, poverty and inequality research, and talks at Ruby programming conferences; published at top conferences

Idea: Let's commercialize the software and sell to academics and conference organizers

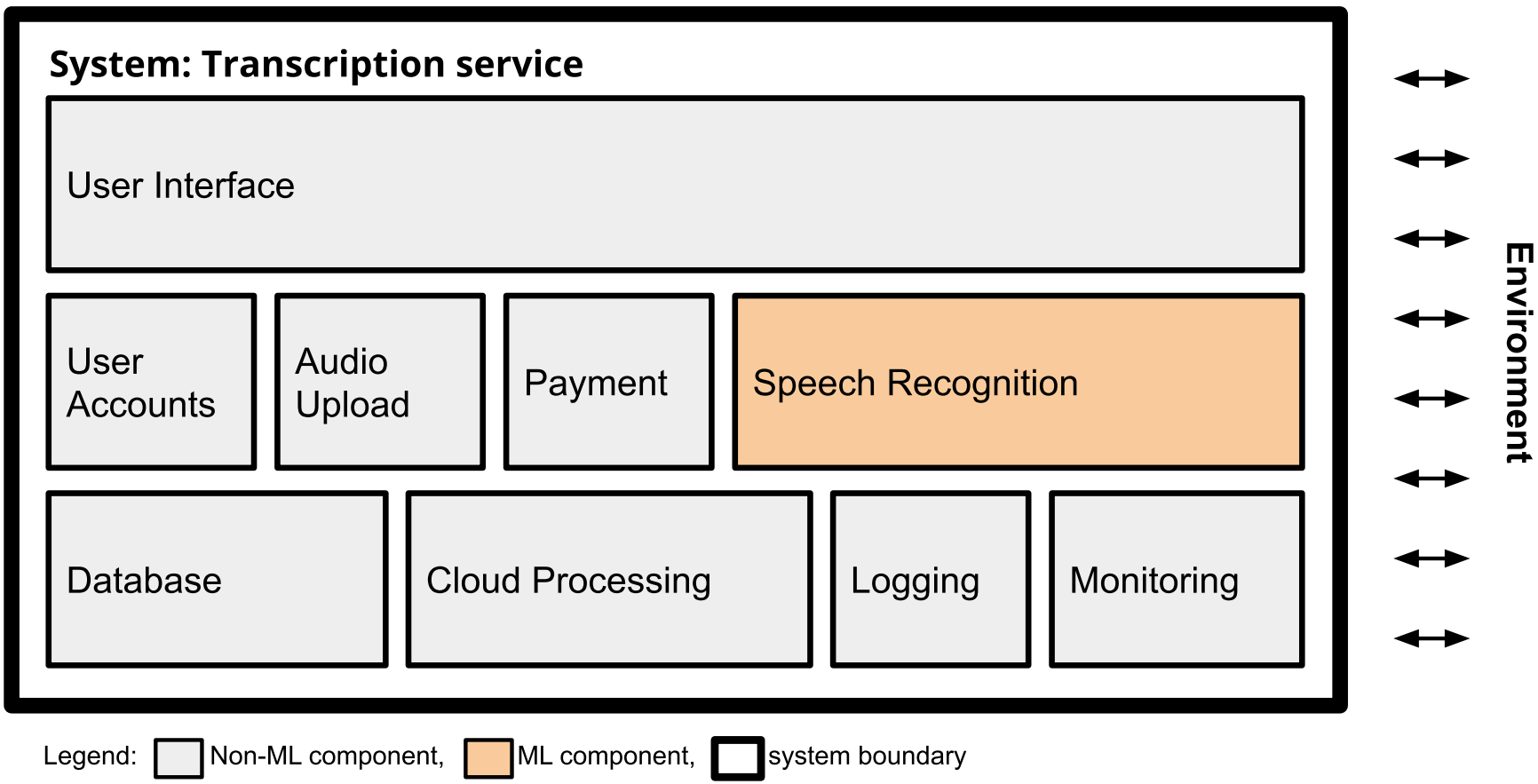
SHORT BREAKOUT

Likely challenges in building commercial product?

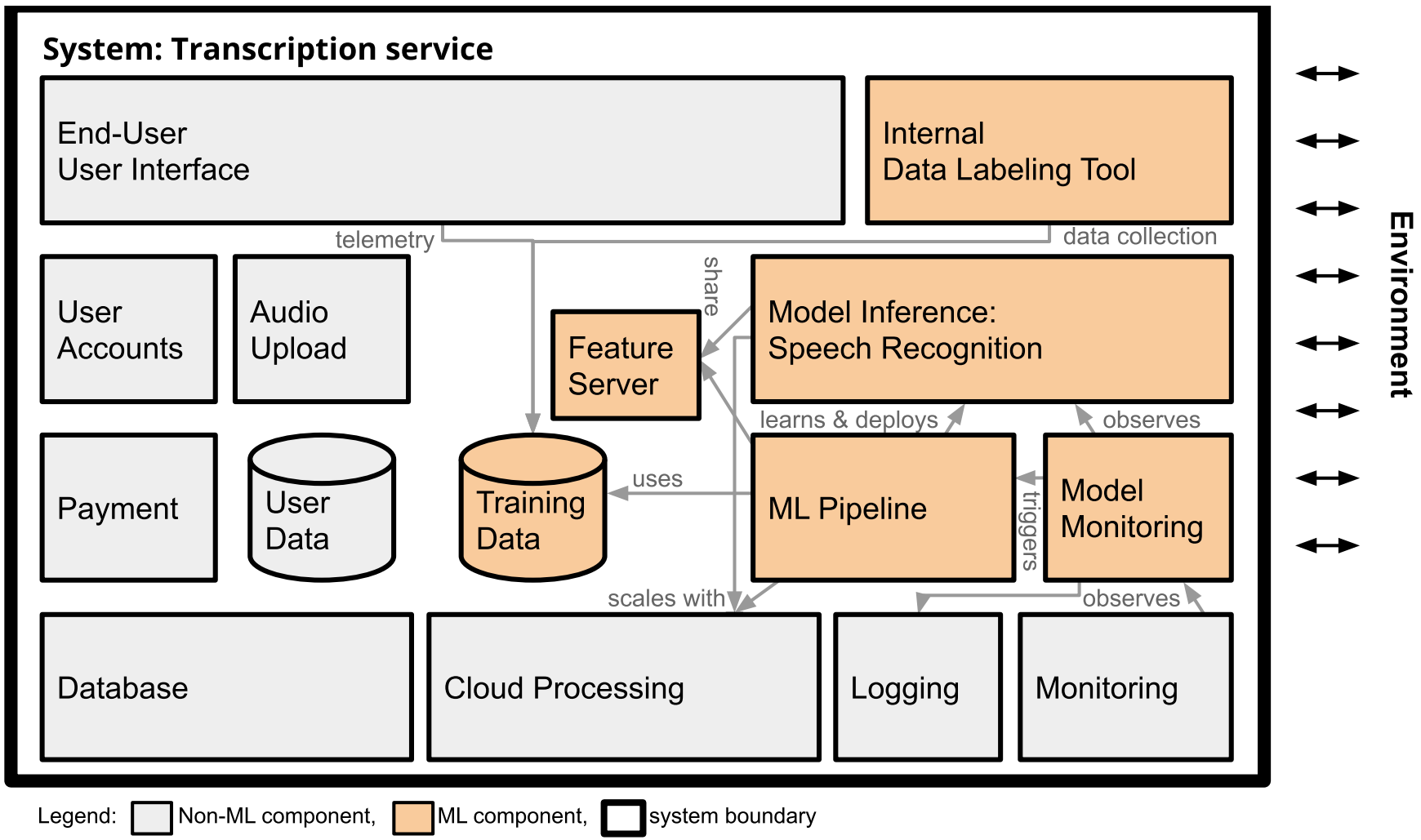
- Think about challenges that the team will likely focus when turning their research into *a product*:
    - One machine-learning challenge
    - One engineering challenge in building the product
    - One challenge from operating and updating the product
    - One team or management challenge
    - One business challenge
    - One safety or ethics challenge
- Fill out one form per team and meet back here in 8 minutes to share suggestions

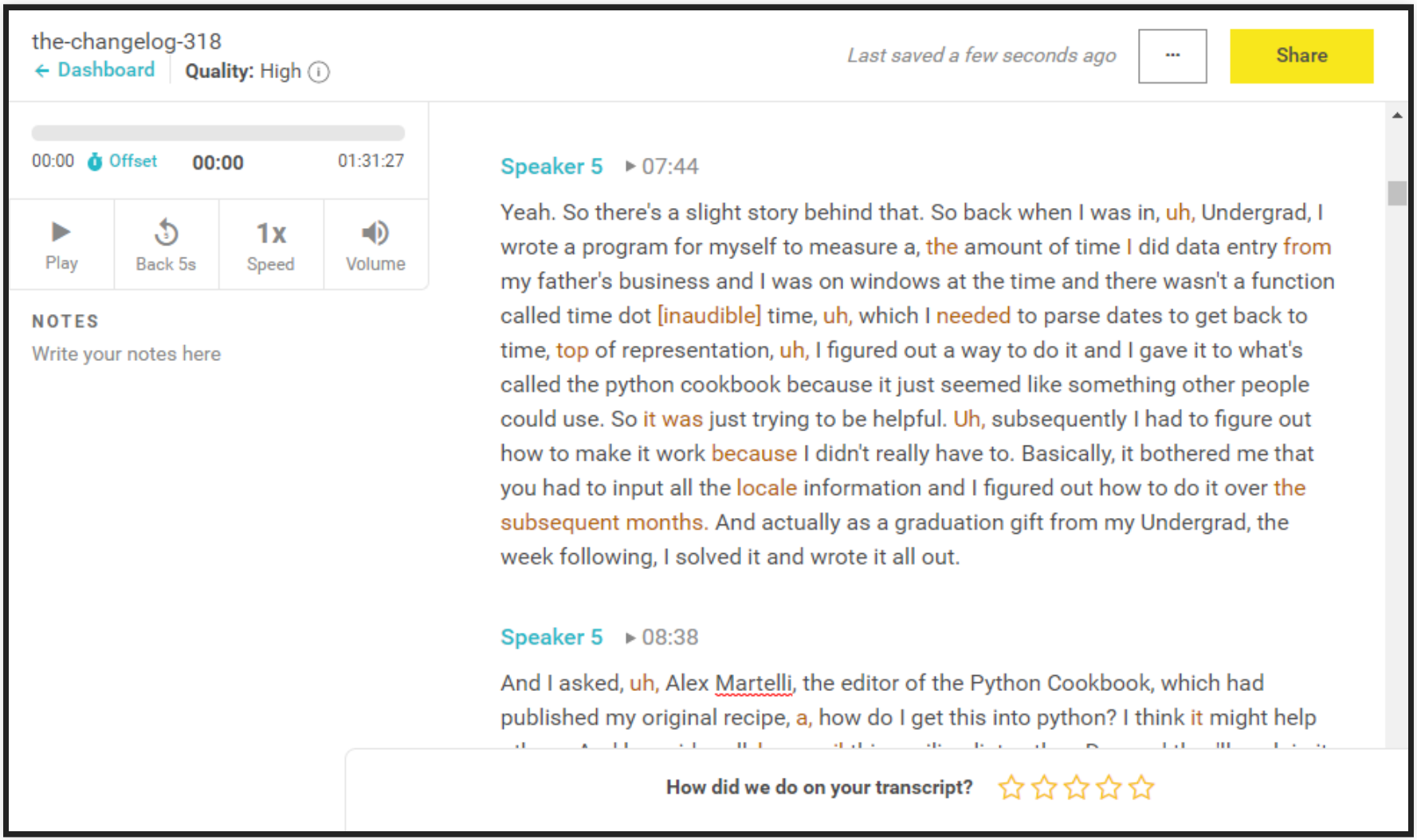
WHAT QUALITIES ARE IMPORTANT FOR A GOOD COMMERCIAL TRANSCRIPTION PRODUCT?

ML IN A PRODUCTION SYSTEM



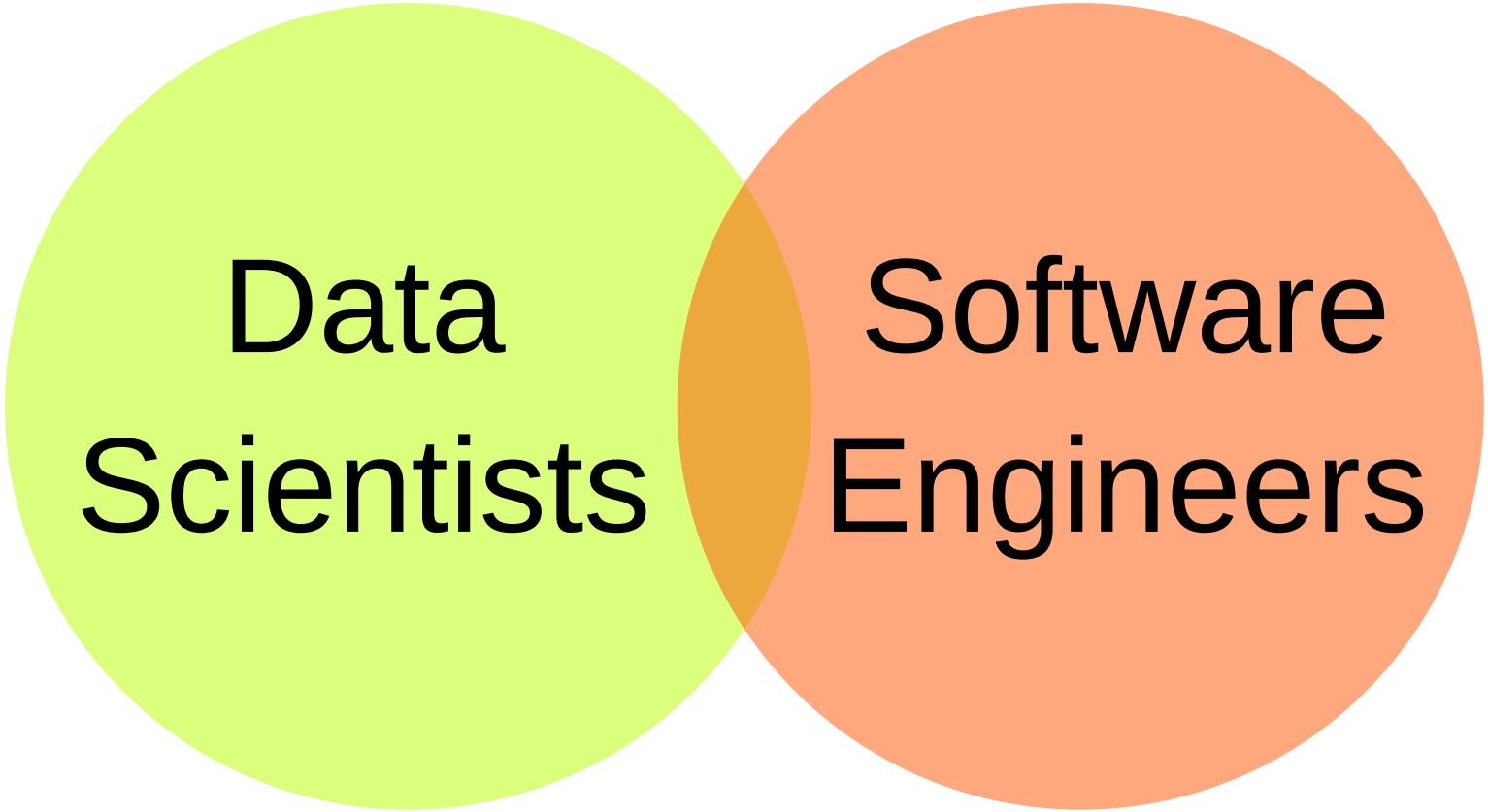
ML IN A PRODUCTION SYSTEM




the-changelog-318
← Dashboard
Quality: High
Last saved a few seconds ago
...
Share
00:00
Offset
00:00
01:31:27
Play
Back 5s
1x
Speed
Volume
NOTES
Write your notes here
Speaker 5 ▸ 07:44
Yeah. So there's a slight story behind that. So back when I was in, uh, Undergrad, I wrote a program for myself to measure a, the amount of time I did data entry from my father's business and I was on windows at the time and there wasn't a function called time dot [inaudible] time, uh, which I needed to parse dates to get back to time, top of representation, uh, I figured out a way to do it and I gave it to what's called the python cookbook because it just seemed like something other people could use. So it was just trying to be helpful. Uh, subsequently I had to figure out how to make it work because I didn't really have to. Basically, it bothered me that you had to input all the locale information and I figured out how to do it over the subsequent months. And actually as a graduation gift from my Undergrad, the week following, I solved it and wrote it all out.
Speaker 5 ▸ 08:38
And I asked, uh, Alex Martelli, the editor of the Python Cookbook, which had published my original recipe, a, how do I get this into python? I think it might help
How did we do on your transcript?

Speaker notes

Highlights challenging fragments. Can see what users fix inplace to correct. Star rating for feedback.



and Data engineers + Domain specialists + Operators + Business team + Project managers + Designers, UI Experts + Safety, security specialists + Lawyers + Social scientists + ...

DATA SCIENTIST

- Often fixed dataset for training and evaluation (e.g., PBS interviews)
- Focused on accuracy
- Prototyping, often Jupyter notebooks or similar
- Expert in modeling techniques and feature engineering
- Model size, updateability, implementation stability typically does not matter

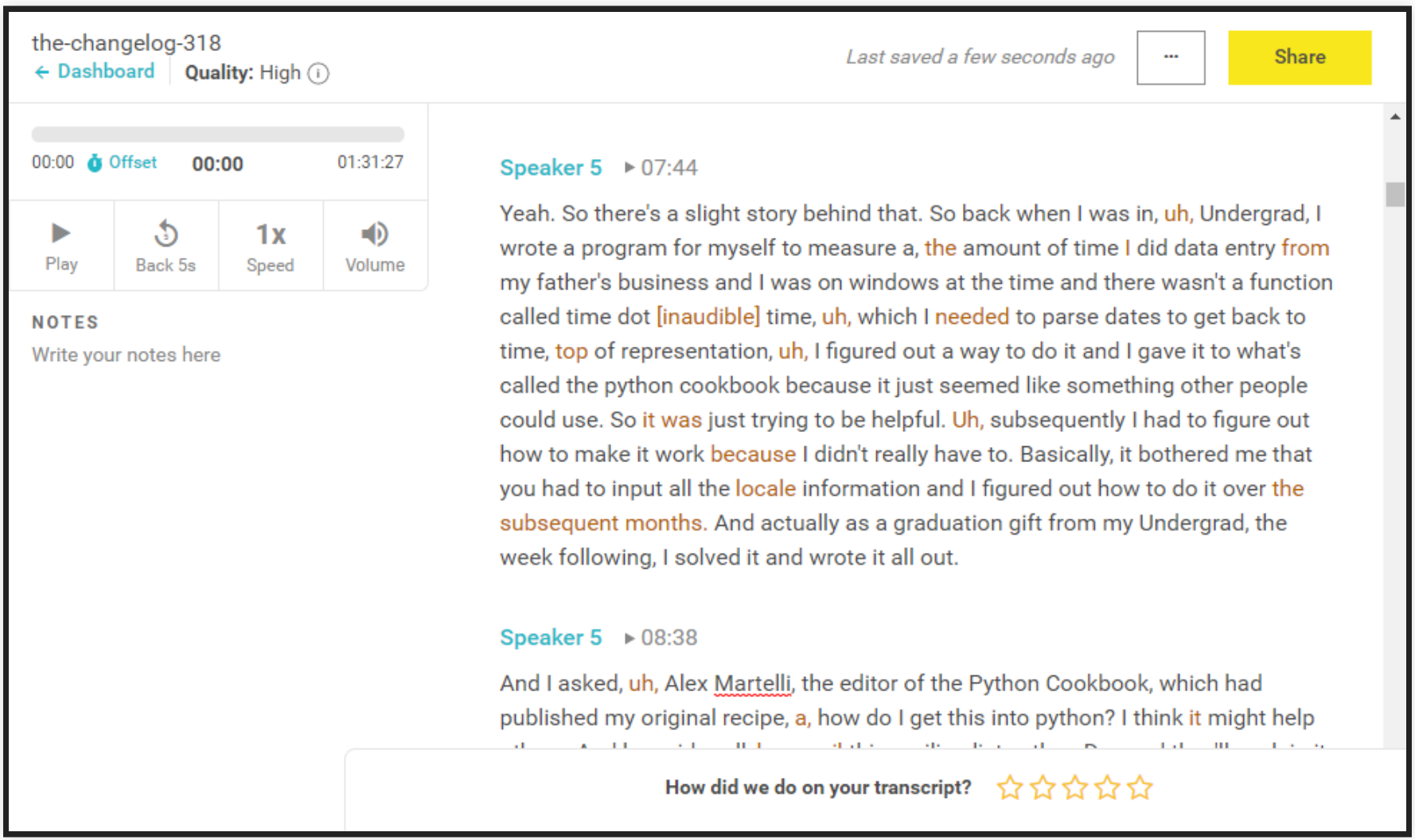
SOFTWARE ENGINEER

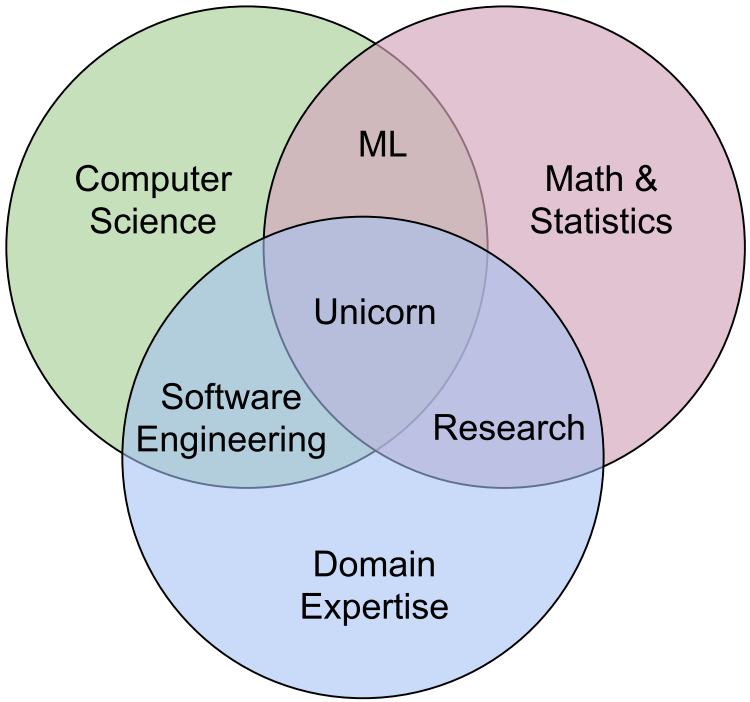
- Builds a product
- Concerned about cost, performance, stability, release time
- Identify quality through customer satisfaction
- Must scale solution, handle large amounts of data
- Detect and handle mistakes, preferably automatically
- Maintain, evolve, and extend the product over long periods
- Consider requirements for security, safety, fairness

LIKELY COLLABORATION CHALLENGES?

- Everybody, **type one or two likely collaboration challenges in the chat but *do not send them yet*. Vote "yes" when done.**

WHAT MIGHT SOFTWARE ENGINEERS AND DATA SCIENTISTS FOCUS ON?





By Steven Geringer, via Ryan Orban. Bridging the Gap Between Data Science & Engineer: Building High-Performance Teams. 2016

T-SHAPED PEOPLE

Broad-range generalist + Deep expertise

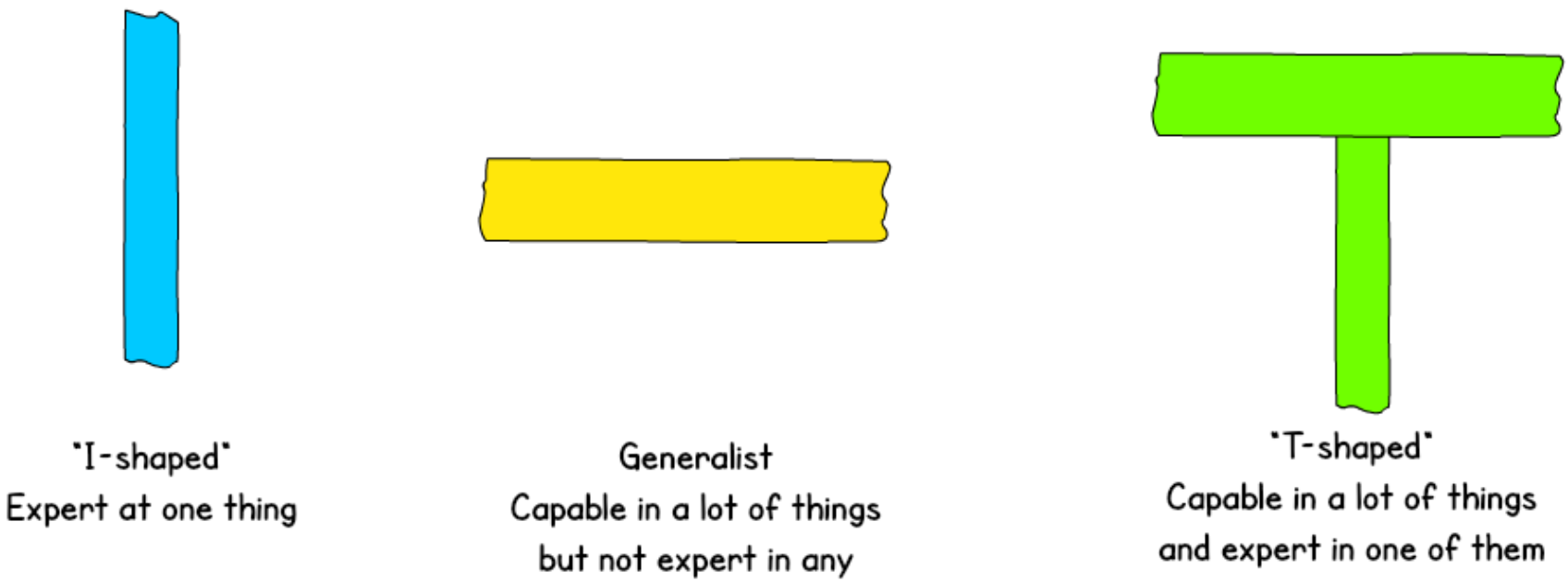


Figure: Jason Yip. Why T-shaped people?. 2018

T-SHAPED PEOPLE

Broad-range generalist + Deep expertise

Example:

- Basic skills of software engineering, business, distributed computing, and communication
- Deep skills in deep neural networks (technique) and medical systems (domain)

EXAMPLES FOR DISCUSSION

- What does correctness or accuracy really mean? What accuracy do customers care about?
- How can we see how well we are doing in practice? How much feedback are customers going to give us before they leave?
- Can we estimate how good our transcriptions are? How are we doing for different customers or different topics?
- How to present results to the customers (including confidence)?
- When customers complain about poor transcriptions, how to prioritize and what to do?

- What are unacceptable mistakes and how can they be avoided? Is there a safety risk?
- Can we cope with an influx of customers?
- Will transcribing the same audio twice produce the same result? Does it matter?
- How can we debug and fix problems? How quickly?

EXAMPLES FOR DISCUSSION 2

- With more customers, transcriptions are taking longer and longer -- what can we do?
- Transcriptions sometimes crash. What to do?
- How do we achieve high availability?
- How can we see that everything is going fine and page somebody if it is not?
- We improve our entity detection model but somehow system behavior degrades... Why?
- Tensorflow update; does our infrastructure still work?
- Once somewhat successful, how to handle large amounts of data per day?
- Buy more machines or move to the cloud?

- Models are continuously improved. When to deploy? Can we roll back?
- Can we offer live transcription as an app? As a web service?
- Can we get better the longer a person talks? Should we then go back and reanalyze the beginning? Will this benefit the next upload as well?

EXAMPLES FOR DISCUSSION 3

- How many domains can be supported? Do we have the server capacity?
- How specific should domains be? Medical vs "International Conference on Allergy & Immunology"?
- How to make it easy to support new domains?

- Can we handle accents?
- Better recognition of male than female speakers?

- Can and should we learn from customer data?
- How can we debug problems on audio files we are not allowed to see?
- Any chance we might private leak customer data?
- Can competitors or bad actors attack our system?

SYLLABUS AND CLASS STRUCTURE

11-695/17-445/17-645/17-745, Spring 2022, 12 units

Monday/Wednesdays 1:25-2:45pm

Recitation Fridays 10:10 / 11:15am

INSTRUCTORS

Christian Kaestner, Eunsuk Kang

Luke Dramko, Nadia Nahar, Sreenidhi Sundaram, Tasheena Narraidoo, Xuchen Zhang

< brief introductions >

COMMUNICATION

- Email us or ping us on Slack (invite link on Canvas)
- Class announcements made through Canvas
- Weekly office hours (see Canvas for schedule)
    - Online for the first two weeks
- Post questions on Slack
    - Please use #general and post publicly if possible; your classmates will benefit from your Q&A!
- All course materials (lectures, assignments, etc.,) available on GitHub. Pull requests encouraged!

CLASS WITH SOFTWARE ENGINEERING FLAVOR

- Focused on engineering judgment
- Arguments, tradeoffs, and justification, rather than single correct answer
- "it depends..."
- Practical engagement, building systems, testing, automation
- Strong teamwork component
- Not focused on formal guarantees or machine learning fundamentals (modeling, statistics)
- Both text-based and code-based homework assignments

PREREQUISITES

Some machine-learning experience required

- Basic understanding of data science process, incl data cleaning, feature engineering, learning
- High level understand of machine-learning approaches
    - supervised learning
    - regression, decision trees, neural networks
    - accuracy, recall, precision, ROC curve
- Ideally, some experience with notebooks, Sklearn or other frameworks

No software-engineering knowledge required

- Basic programming and command-line skills will be needed
- Teamwork experience in product team is useful but not required
- No required exposure to requirements, software testing, software design, continuous integration, containers, process management, etc
    - If you are familiar with these, there will be some redundancy -- sorry!

ACTIVE LECTURE

- Case study driven
- Discussions highly encouraged
- Contribute your own experience!
- Regular active in-class exercises
- In-class presentations
- Discussions over definitions

Fundamentals of Engineering AI-Enabled Systems

Holistic system view: AI and non-AI components, pipelines, stakeholders, environment interactions, feedback loops

Requirements:
System and model goals
User requirements
Environment assumptions
Quality beyond accuracy
Measurement
Risk analysis
Planning for mistakes

Architecture + design:
Modeling tradeoffs
Deployment architecture
Data science pipelines
Telemetry, monitoring
Anticipating evolution
Big data processing
Human-AI design

Quality assurance:
Model testing
Data quality
QA automation
Testing in production
Infrastructure quality
Debugging

Operations:
Continuous deployment
Contin. experimentation
Configuration mgmt.
Monitoring
Versioning
Big data
DevOps, MLOps

Teams and process: Data science vs software eng. workflows, interdisciplinary teams, collaboration points, technical debt

Responsible AI Engineering

Provenance, versioning, reproducibility

Safety

Security and privacy

Fairness

Interpretability and explainability

Transparency and trust

Ethics, governance, regulation, compliance, organizational culture

TEXTBOOK

Building Intelligent Systems: A Guide to Machine Learning Engineering

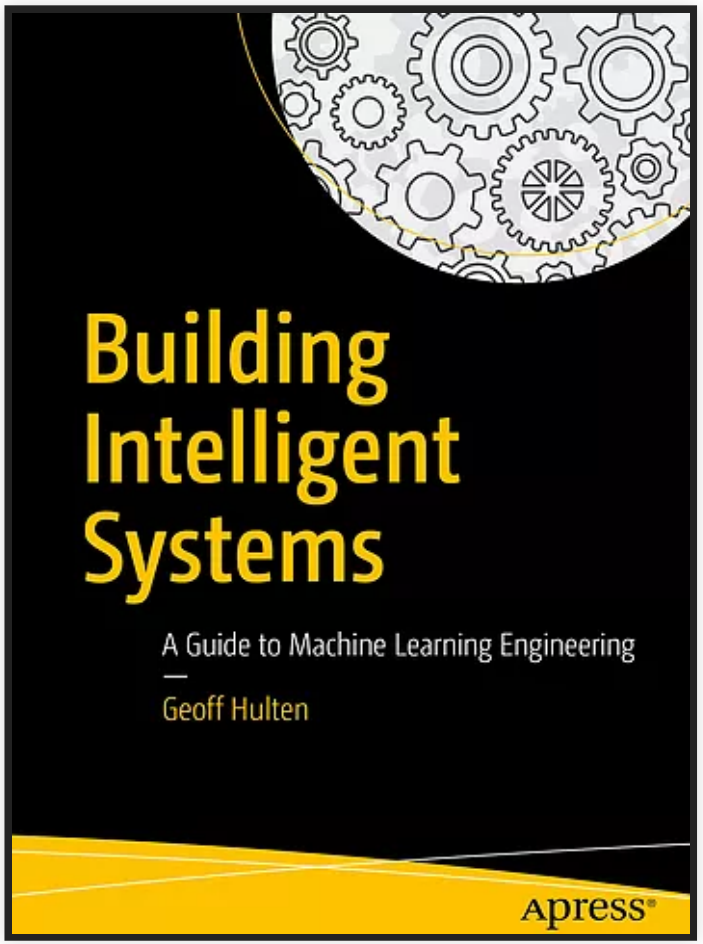
by Geoff Hulten

https://www.buildingintelligentsystems.com/

Most chapters assigned at some point in the semester

Supplemented with research articles, blog posts, videos, podcasts, ...

Electronic version in the library



WE ARE WRITTING A BOOK

"Machine Learning in Production: From Models to Products"

Mostly similar coverage to lecture.

Not required, use as supplementary reading.

Not all chapters finished yet.

Feedback appreciated.

Published online

READINGS AND QUIZZES

- Reading assignments for most lectures
  - Preparing in-class discussions
  - Background material, case descriptions, possibly also podcast, video, wikipedia
  - Complement with own research
- Short essay questions on readings, due before start of lecture (Canvas quiz)
- Planned for: about 30-45 min for reading, 15 min for discussing and answering quiz

ASSIGNMENTS

- All assignments available on GitHub
- Series of 4 small to medium-sized individual assignments
    - Engage with practical challenges
    - Analyze risks, fairness
    - Reason about tradeoffs and justify your decisions
    - Mostly written reports, a little modeling, limited coding
    - May be done with a partner (more on this later)
- Large team project with 4 milestones (mostly in second half)
    - Build and deploy a prediction (movie recommendation) service
    - Testing in production, monitoring
    - Final presentation
- Usually due Wednesday night; see schedule

17-745 PHD RESEARCH PROJECT

- Research project instead of individual assignments I3 and I4
- Design your own research project and write a report
    - A case study, empiricial study, literature survey, etc.,
- See the project description and talk to us

RECITATIONS

Typically hands on exercises, use tools, analyze cases

Designed to introduce tools and discuss material relevant for assignments

First recitation on **this Friday, Jan 21**! Remote work and collaboration with Git

GRADING

- 40% individual assignment
- 30% group project with final presentation
- 10% midterm
- 10% participation
- 10% reading quizzes
- No final exam (final presentations will take place instead)

- expected grade cutoffs: 81-90% B, 91-100% A

GRADING PHILOSOPHY

- Specification grading, based in adult learning theory
- Giving you choices in what to work on or how to prioritize your work
- We are making every effort to be clear about expectations (specifications), will clarify if you have questions
- Assignments broken down into expectations with point values, each graded **pass/fail**
- You should be able to tell what grade you will get for an assignment when you submit it, depending on what work you chose to do
- Opportunities to resubmit work until last day of class

[Example]

PARTICIPATION

- Participation is important
    - Participation in in-class discussions
    - Active participation in recitations
    - Alternative arrangements if you cannot attend classes live
- Participation != Attendance
- Grading:
    - 100%: Participates at least once in most lectures by (1) asking or responding to questions or (2) contributing to breakout discussions
    - 100%: Participates in 25% of lectures and actively contributes to discussions in most recitations
    - 90%: Participates at least once in over half of the lectures
    - 70%: Participates at least once in 25% of the lectures
    - 40%: Participates at least once in at least 3 lectures or recitations.
    - 0%: No participation in the entire semester.

FLEXIBILITY AND ACCOMMODATIONS

(details in syllabus)

- 7 tokens per student:
    - Submit individual assignment 1 day late for 1 token (after running out of tokens 15% penalty per late day)
    - Redo individual assignment for 3 token
    - Resubmit or submit reading quiz late for 1 token
    - Remaining tokens count toward participation

- 7 tokens per team:
    - Submit milestone 1 day late for 1 token (no late submissions accepted when out of tokens)
    - Redo milestone for 3 token
- Exceptions and accommodations on request, email us.

GROUP PROJECT

- Instructor-assigned teams
- Teams stay together for project throughout semester, starting next week
- Please fill out survey after class on **Monday, Jan 24**
- Some advice in lectures; we'll help with debugging team issues
- Peer grading on all milestones (based on citizenship on team)

ADDITIONAL GROUPWORK OPTIONS

- Encouraging interactions
- Can complete all individual assignments and quizzes as pairs
- Can't work with the same partner again on a different assignment/quiz

- Bonus points for social interaction in project teams
    - See "Social Activities Bonus" on the project description

ACADEMIC HONESTY

See web page

In a nutshell: do not copy, do not lie, do not share or publicly release your solutions

In group work, be honest about contributions of team members, do not cover for others

If you feel overwhelmed or stressed, please come and talk to us (see syllabus for other support opportunities)

WHAT MAKES SOFTWARE WITH ML CHALLENGING?

ML MODELS MAKE MISTAKES

NeuralTalk2: A flock of birds flying in the air

Microsoft Azure: A group of giraffe standing next to a tree

Image: Fred Dunn, https://www.flickr.com/photos/gratapictures - CC-BY-NC

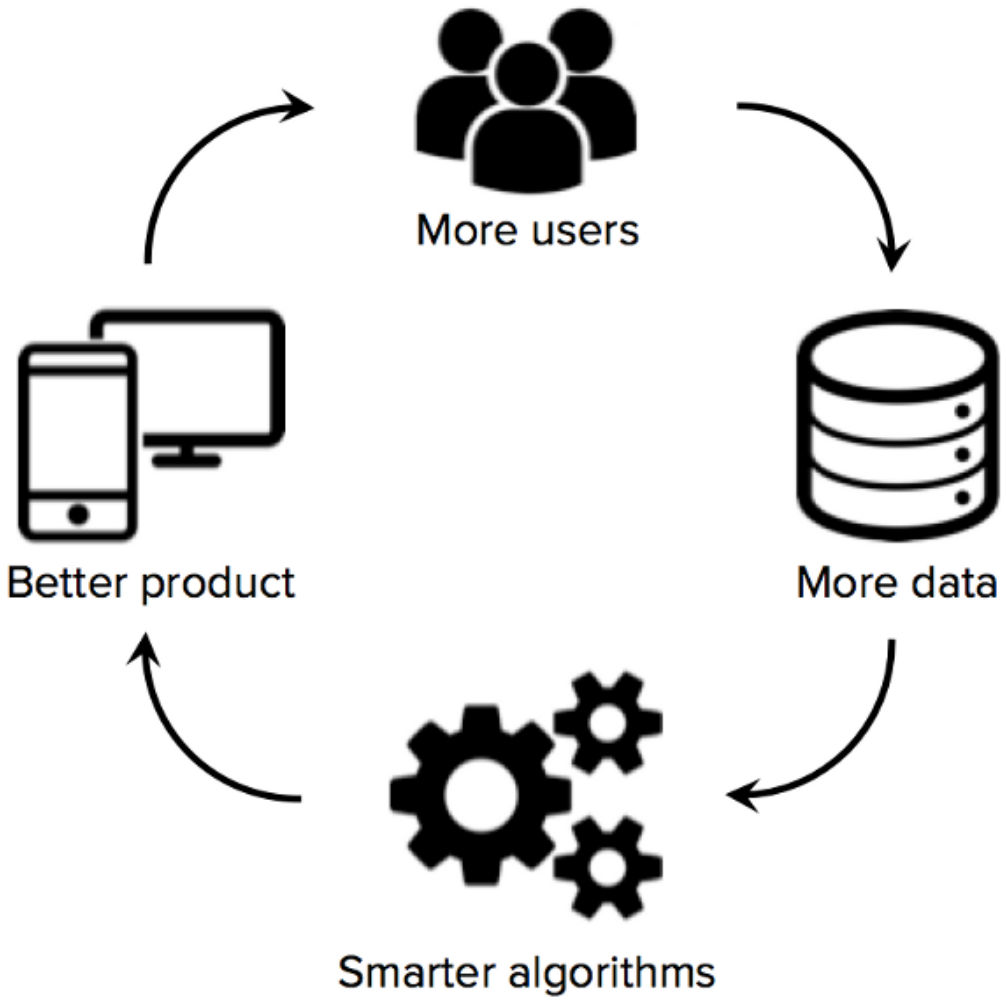
Speaker notes

Source: https://www.aiweirdness.com/do-neural-nets-dream-of-electric-18-03-02/

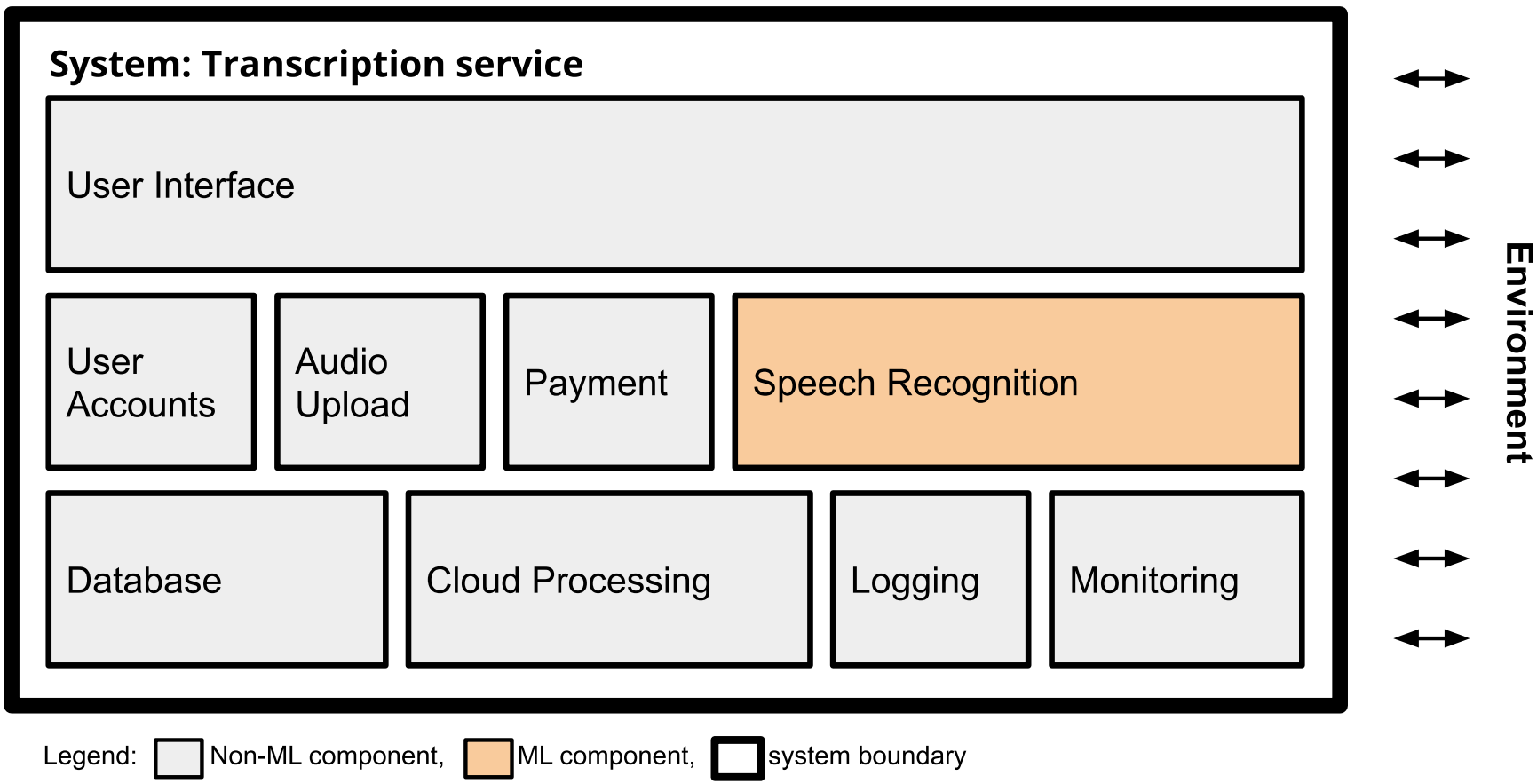
LACK OF SPECIFICATIONS

```
/**
  Return the text spoken within the audio file
  ????
*/
String transcribe(File audioFile);
```

DATA FOCUSED AND SCALABLE



INTERACTION WITH THE ENVIRONMENT

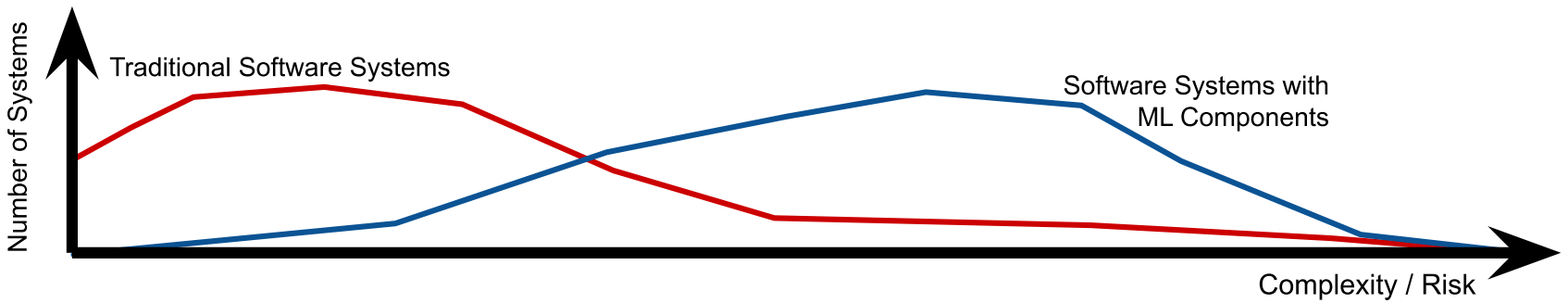


IT'S NOT ALL NEW

- Safe software with unreliable components
- Cyberphysical systems
- Non-ML big data systems, cloud systems
- "Good enough" and "fit for purpose" not "correct"

- We routinely build such systems
- ML intensifies our challenges

COMPLEXITY


Number of Systems
Traditional Software Systems
Software Systems with
ML Components
Complexity / Risk

INTRODUCTIONS

By the end of today, enter into Slack channel #intro:

- Your (preferred) name
- In 1~2 sentences, your data science background and goals (e.g., coursework, internships, work experience)
- In 1~2 sentences, your software engineering background, if any, and goals (e.g., coursework, internships, work experience)
- One topic you are particularly interested in learning during this course?
- A hobby or a favorite activity outside school

SUMMARY

- Machine learning components are part of larger systems
- *Data scientists* and *software engineers* have different goals and focuses
    - Building systems requires both
    - Various qualities are relevant, beyond just accuracy
- Machine learning brings new challenges and intensifies old ones