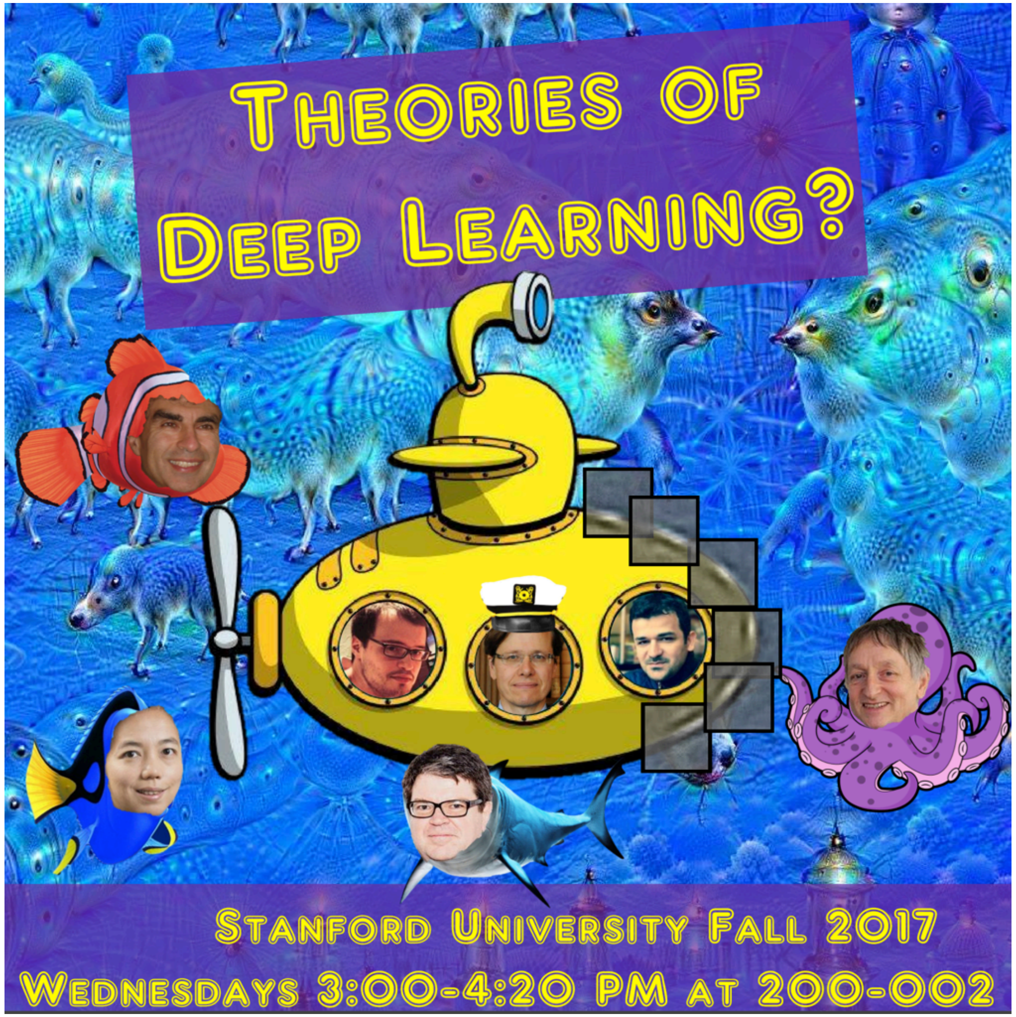
Research Paradigms in the AI age

Speaker: Qingyun Sun
Math PhD @ Stanford

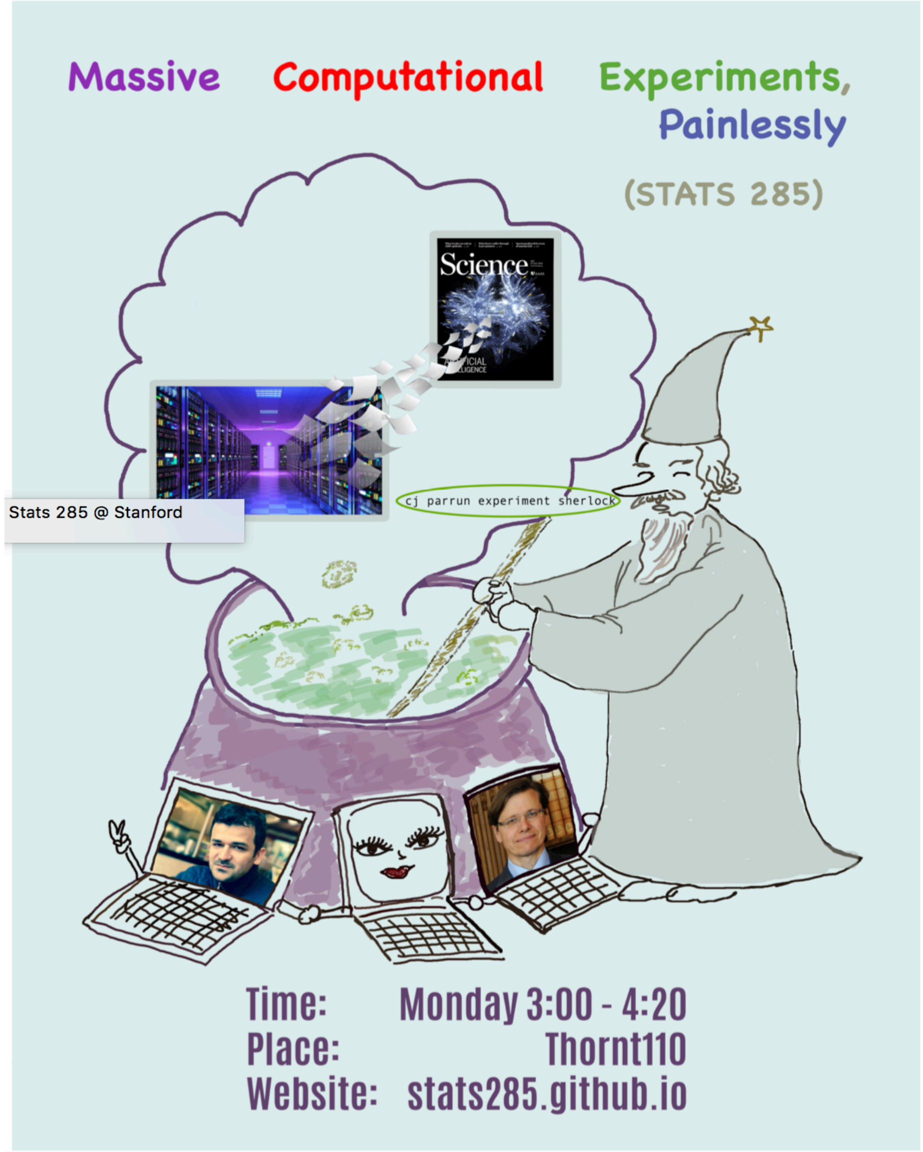
Acknowledgement: Stats 385 @ Stanford

https://stats385.github.io/


Theories of Deep Learning?
Stanford University Fall 2017
Wednesdays 3:00-4:20 PM at 200-002

Acknowledgement: Stats 285 @ Stanford

https://stats285.github.io/


Massive Computational Experiments, Painlessly
(STATS 285)
Science
cj parrun experiment sherlock
Stats 285 @ Stanford
Time: Monday 3:00 - 4:20
Place: Thornt110
Website: stats285.github.io

Outline

- The deep learning research paradigm
- Crisis of AI research
- Massive computation on the cloud painlessly
- Moxel: Model serving and sharing

The “classical ML” pipeline:

- Researcher looks at dataset
- Applies his favorite ML algorithms
- Maybe do some math to adjust the algorithm
- Compare the results and iterate.

The “deep learning” pipeline:

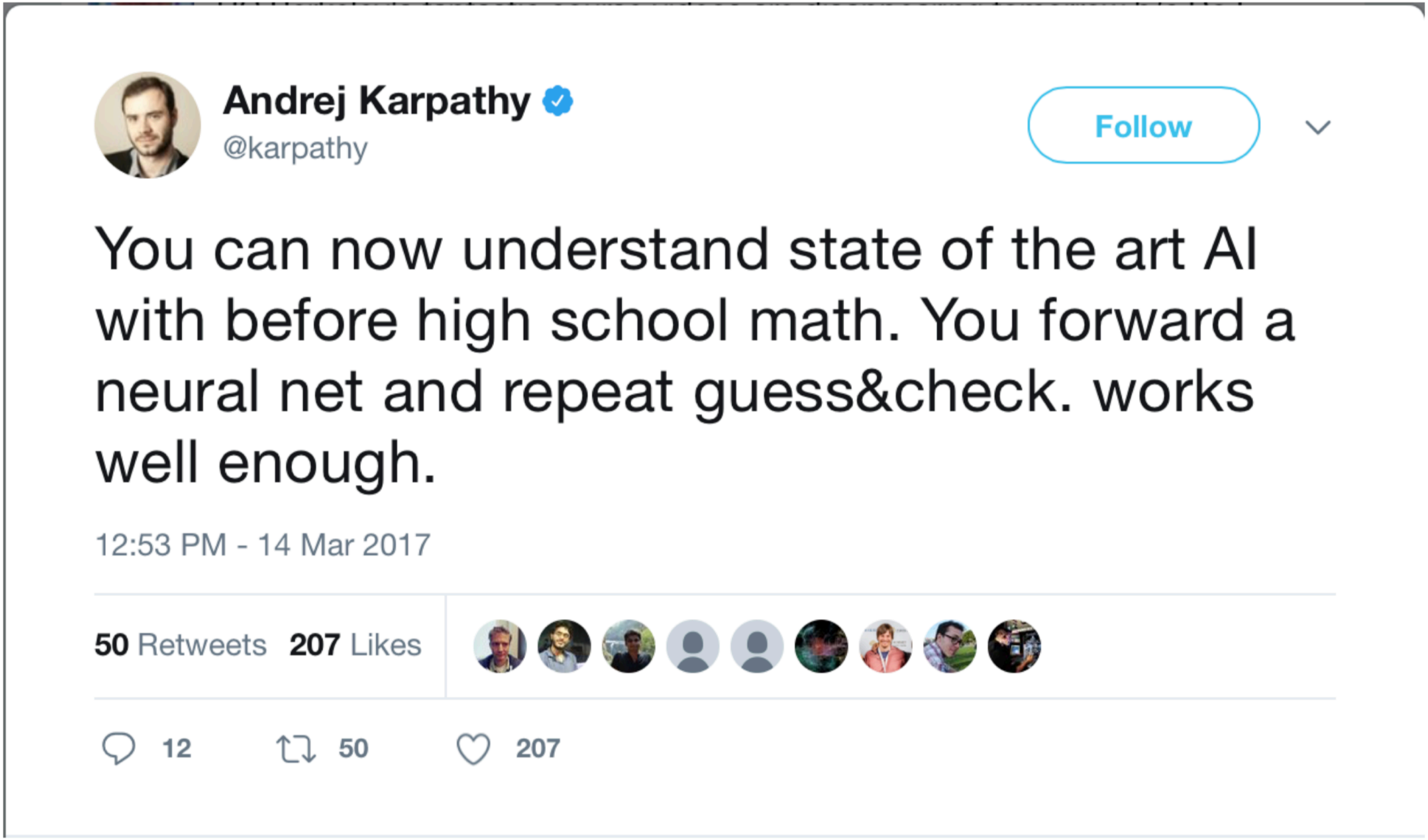
- Researchers work on a large dataset competition (say, ImageNet)
- Start with your favorite Network in Tensorflow
- Make small tweaks to the network
- Training the network using variants of SGD
- On your local GPU, school cluster or AWS cloud
- Evaluate your trained model for generalization
- Serve your model in production

Big picture: Common task framework

1. Researchers set up local copies of Challenge
   - Data – Training, Test carved out of public dataset
   - Scoring – same as challenge scoring rule
2. Researcher's job: *'tuning models'*
   - Think up a family of model variations – *'tweak's*
   - Run a full *'experiment'* – suite of tweaks – *'grid'*
   - Score each tweak
   - Submit best-scoring result to central authority
3. Successful researchers perpetually motivated by *Game-ification*: tweaking, scoring, winning.
4. Researchers who tweak more often, win more often!.
5. If easier to implement tweaks and faster to evaluate them, more likely to win!.

1. *Researchers who tweak more often, win more often!*
2. *If easier to implement tweaks and faster to evaluate them, more likely to win!*
3. Successful Research Environment
    - Easy to tweak models
    - Easy to score tweaks
    - Fast to score tweaks
4. Successful researchers perpetually motivated by *Game-ification*: tweaking, scoring, winning.
5. Easier to stay motivated when easier and more comfortable to play the game.
    - Elegant expression of tweaks
    - Rapid turn-around for scoring

Crisis of AI research:
The barrier of conducting AI research is growing lower!



Andrej Karpathy
@karpathy

You can now understand state of the art AI with before high school math. You forward a neural net and repeat guess&check. works well enough.

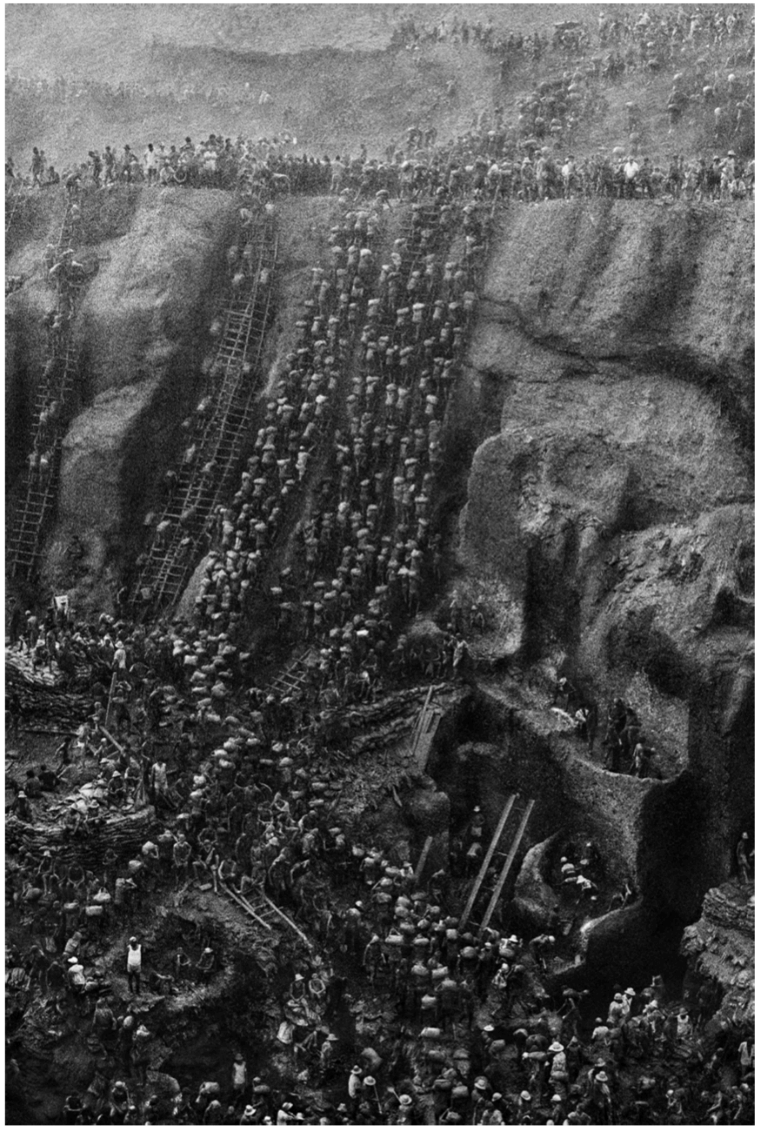
12:53 PM - 14 Mar 2017

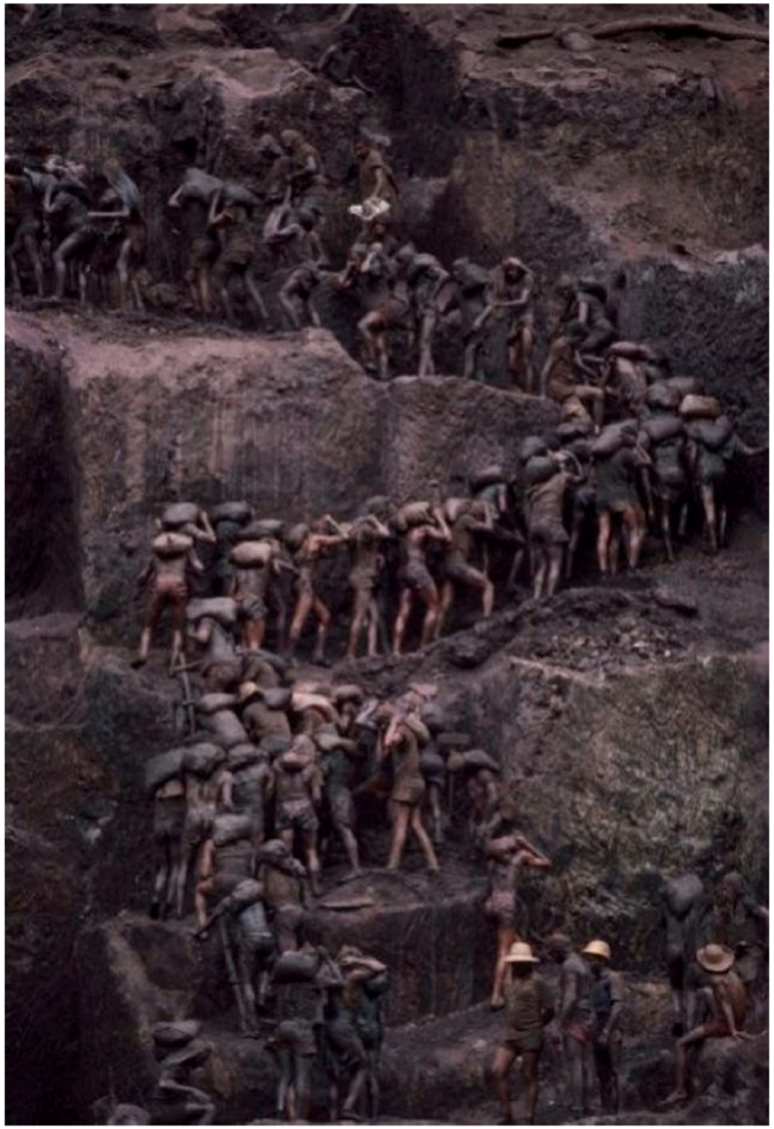
50 Retweets **207** Likes

12 50 207

SGD+ GSD:

stochastic gradient descent
+ graduate student descent





Crisis again:

A big part of AI research work could be automated by meta-learning.

Most time spent in graduate student descent!

Fight with clusters to run more jobs and wait.

Academic research in crisis!

We are at a university!

1. Q: *Where's the intellectual activity in tuning?*
2. Q: *I didn't come here to do hard manual labor!*
3. Q: *I didn't come here to compete as mindless drones!*

What we **imagine**:

Computers as Slavery!

Traditionally, ‘using computers’ involves interactively running programs (Excel, Point-and-click)

Claerbout’s Dictum: “... dependence on an interactive program can be a form of slavery”

http://sepwww.stanford.edu/sep/jon/reproducible.html

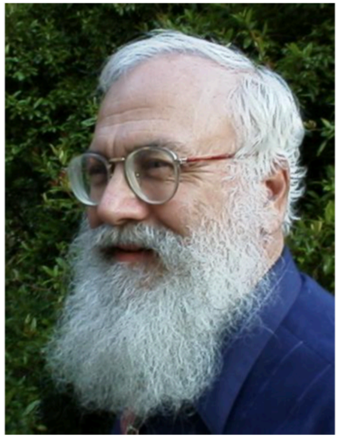


Photo: Jon Claerbout

Cartoon: http://fritsAhlefeldt.com

Response to the crisis:

1. Stop fighting to run more jobs by hand.
2. Push button to start computation on the cloud painlessly.
3. Spend time on higher level thinking.
4. Improve your frameworks and processes.

The real action is all in frameworks

1. Dream up, test, and publish better ...
   - Types of models
   - Types of tweaks
   - Properties for evaluation
2. Implement better *frameworks* ...
   - More elegant expression of models, tweaks
   - Distributed Learning across clusters
   - Smoother collection and analysis of results

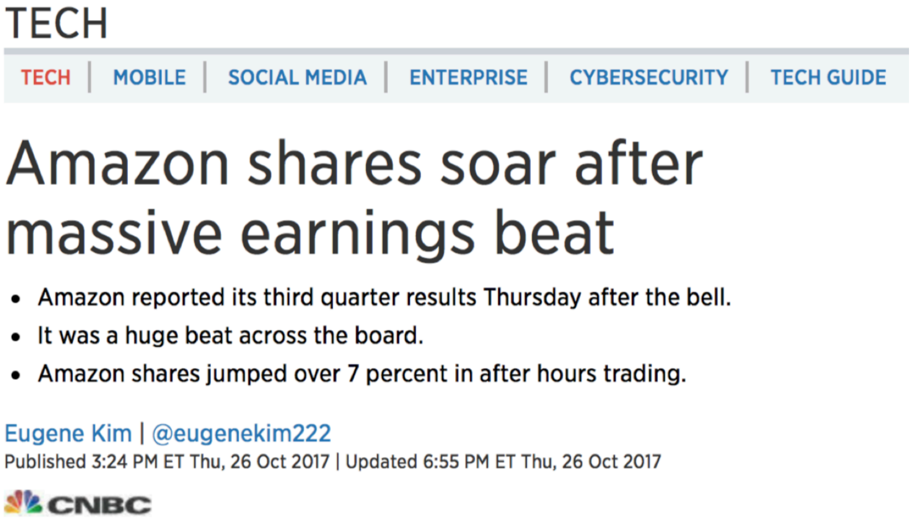
Framework evolution

- Traditional issues
    - Experiments implicitly defined by executing unorganized code
    - Hard to understand what the baseline is, what variations are
    - Code dependencies unclear
    - Ordeal to get all the jobs to run, maybe gave up early
    - Tedious to harvest all the data, maybe missing some data
    - Confusing manual compilation and reporting
- Modern Frameworks
    - Systematic structure to coding
    - Base experiment clearly defined
    - Tweaks clearly defined
    - Code dependencies explicit
    - Grid of Jobs run systematically
    - Automatic transparent access of (cluster, AWS,...)
    - Data Harvested automatically to central data repository
    - Data analyzed automatically using defined tools

The fundamental change that drives the AI evolution?

Science
ARTIFICIAL INTELLIGENCE

AWS is eating the world!



TECH

TECH | MOBILE | SOCIAL MEDIA | ENTERPRISE | CYBERSECURITY | TECH GUIDE

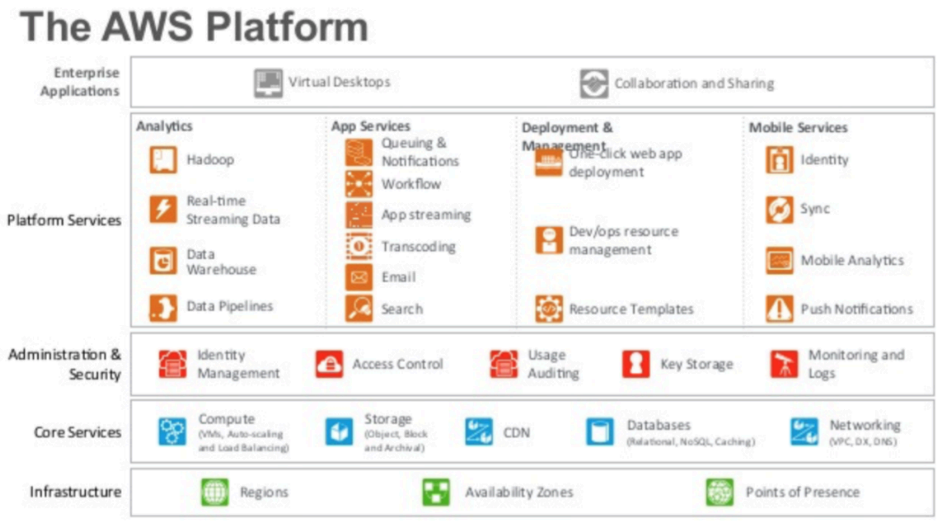
Amazon shares soar after massive earnings beat

- Amazon reported its third quarter results Thursday after the bell.
- It was a huge beat across the board.
- Amazon shares jumped over 7 percent in after hours trading.

Eugene Kim | @eugenekim222
Published 3:24 PM ET Thu, 26 Oct 2017 | Updated 6:55 PM ET Thu, 26 Oct 2017

CNBC

AWS services become ubiquitous



The AWS Platform

Enterprise Applications: Virtual Desktops · Collaboration and Sharing

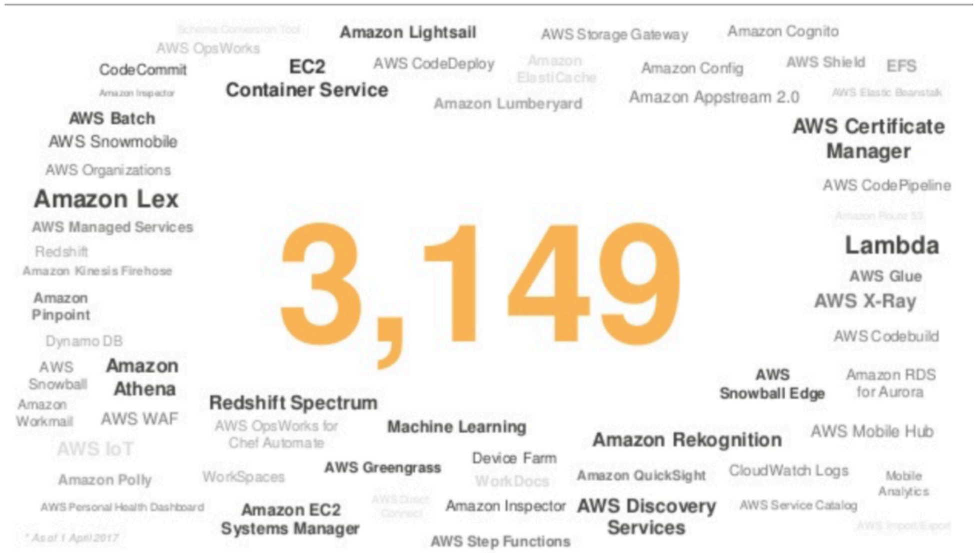
Platform Services:

- Analytics: Hadoop · Real-time Streaming Data · Data Warehouse · Data Pipelines
- App Services: Queuing & Notifications · Workflow · App streaming · Transcoding · Email · Search
- Deployment & Management: One-click web app deployment · Dev/ops resource management · Resource Templates
- Mobile Services: Identity · Sync · Mobile Analytics · Push Notifications

Administration & Security: Identity Management · Access Control · Usage Auditing · Key Storage · Monitoring and Logs

Core Services: Compute (VMs, Auto-scaling and Load Balancing) · Storage (Object, Block and Archival) · CDN · Databases (Relational, NoSQL, Caching) · Networking (VPC, DX, DNS)

Infrastructure: Regions · Availability Zones · Points of Presence



Cloud Paradigm:

- Billions of smart devices each drive queries to cloud servers
- Millions of business relying on cloud for all needs

Symbiosis of cloud and economy is *lasting* and *disruptive*.

Cloud provides *any user* **same-day** delivery:

- Tens to hundreds of thousands of hours of CPU
- Pennies per CPU hour
- ≈ 50 cents per GPU hour

Any user can consume *1 Million CPU hours* over a few days for a few $10K's.