Announcements

Project 6 is out:

- Due next Monday. Opportunity to build something from scratch. Will probably take some experimentation on your part.
- I’ve delegated autograder development to Allen. Thanks Allen!
- See great Piazza thread here: https://piazza.com/class/is2qi3hwdjn3i8?cid=1505

Remember Midterm 2? It is 97% graded. Meeting tonight to finish.

Final exam will be comrehensive.

- Information on review sessions coming soon.

CS188: Fall 2016

Lecture 26: Conclusion

- 188 in a Huge Nutshell.
- The Future of AI.
- AMA.
- After 188.
- Survey.

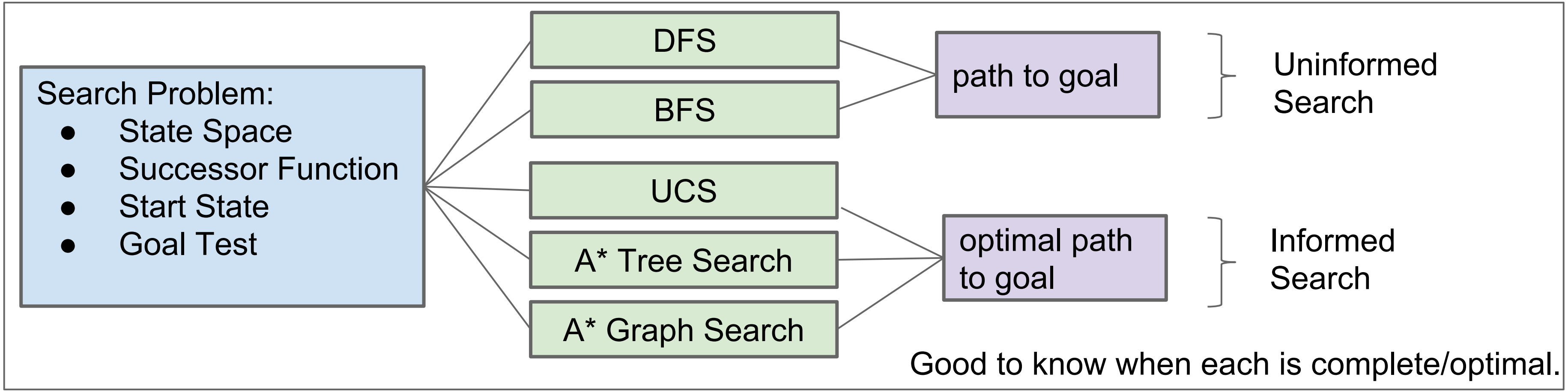
188

Throughout 188, we learned to maximize our expected utility.

Three main parts to course:

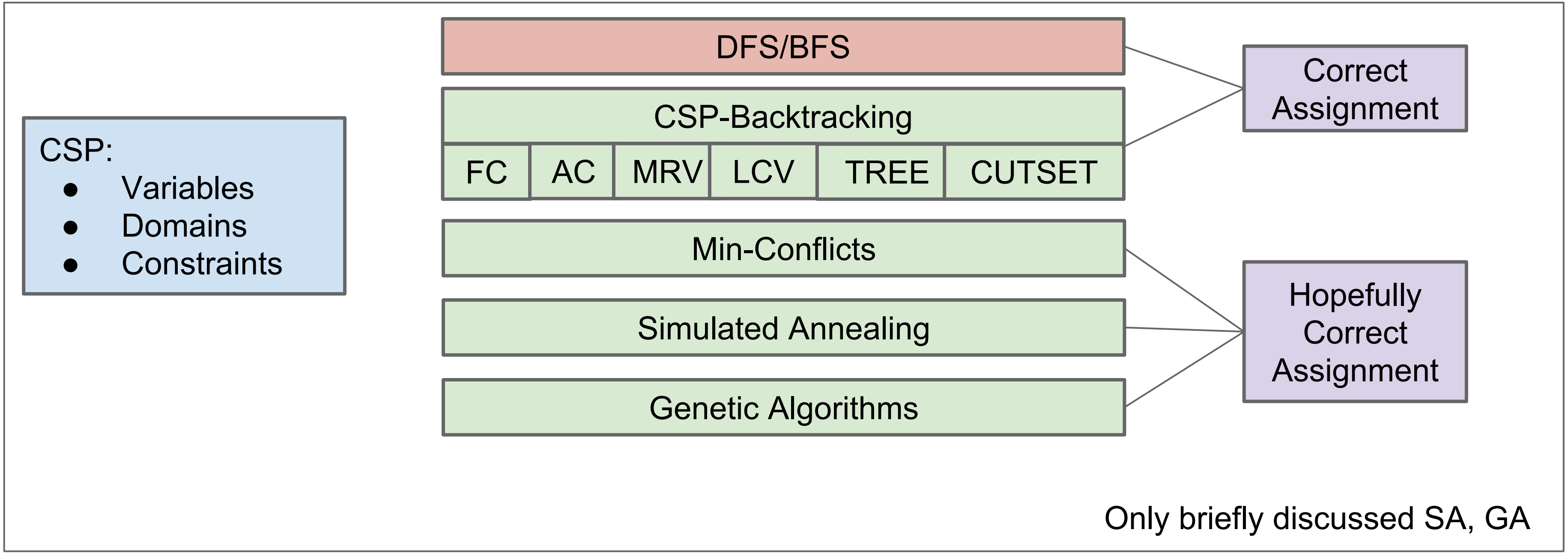
- Part I: Search (culminating in A* for state space search, Min-Conflicts and CSP-Backtracking for CSPs, Policy/Value Iteration for MDPs, Q-Learning for RL)
- Part II: Probabilistic Inference (culminating in Variable Elimination and Gibbs Sampling for Inference, and ideas of VPI/Decision Networks for solving real problems).
- Part III: Machine Learning (culminating in Neural Networks for classification and Backpropagation+Gradient Descent for training).

Part I: Search -- State Space Search



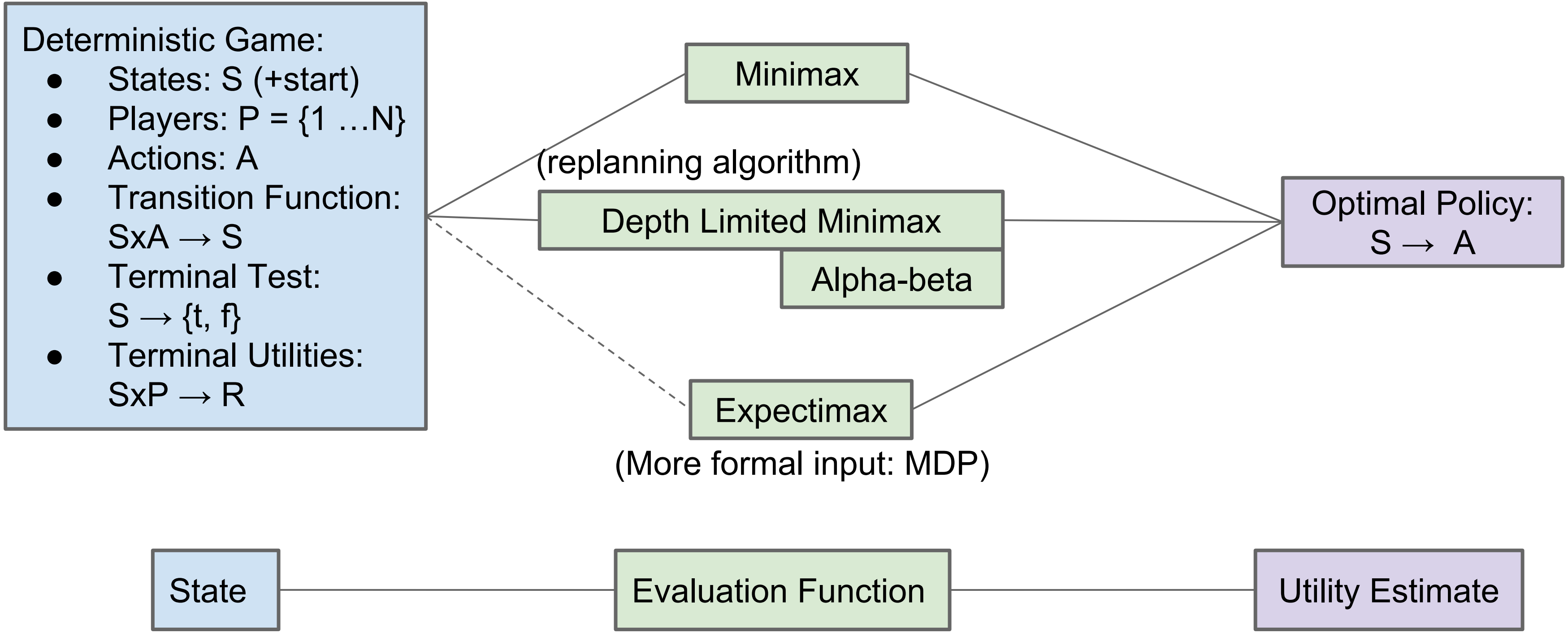
Lecture 1/2/3

Part I: Search -- Constraint Satisfaction Problems



Lecture 4/5

Part I: Search -- Deterministic (Adversarial) Games



Note #1: Replanning algorithms give a policy only for current state! Can thrash.
Note #2: If N > 2, players optimize values from a tuple.

Part I: Search -- Formal Definition of Utility

Given any preferences satisfying these axioms, there is a real-valued function U such that:

$$U(A) \geq U(B) \;\Leftrightarrow\; A \succeq B$$

$$U(\underbrace{[p_1, S_1;\ \ldots\ ;\ p_n, S_n]}_{\text{Lottery}}) = \textstyle\sum_i p_i U(S_i)$$

The Axioms of Rationality

Orderability

$$(A \succ B) \vee (B \succ A) \vee (A \sim B)$$

Transitivity

$$(A \succ B) \wedge (B \succ C) \Rightarrow (A \succ C)$$

Continuity

$$A \succ B \succ C \Rightarrow \exists p\ [p, A;\ 1-p, C] \sim B$$

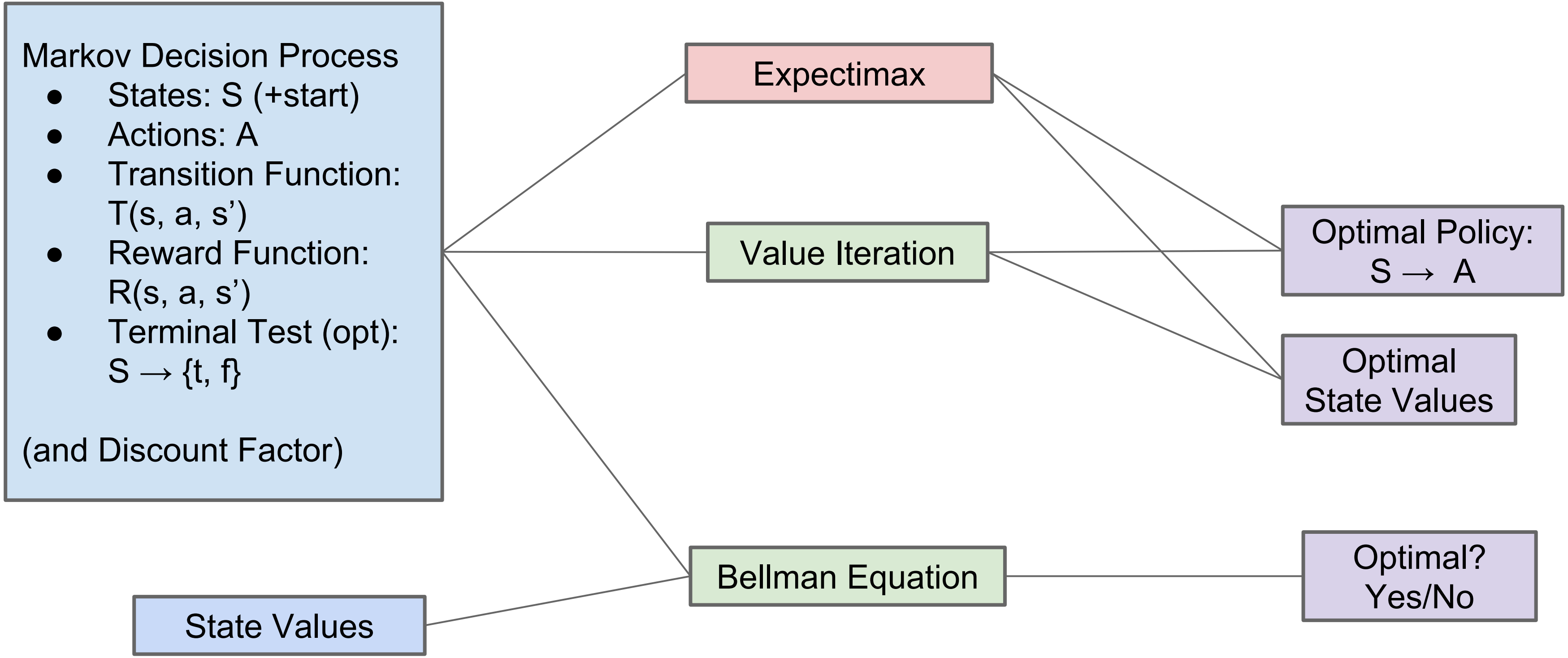
Substitutability

$$A \sim B \Rightarrow [p, A;\ 1-p, C] \sim [p, B; 1-p, C]$$

Monotonicity

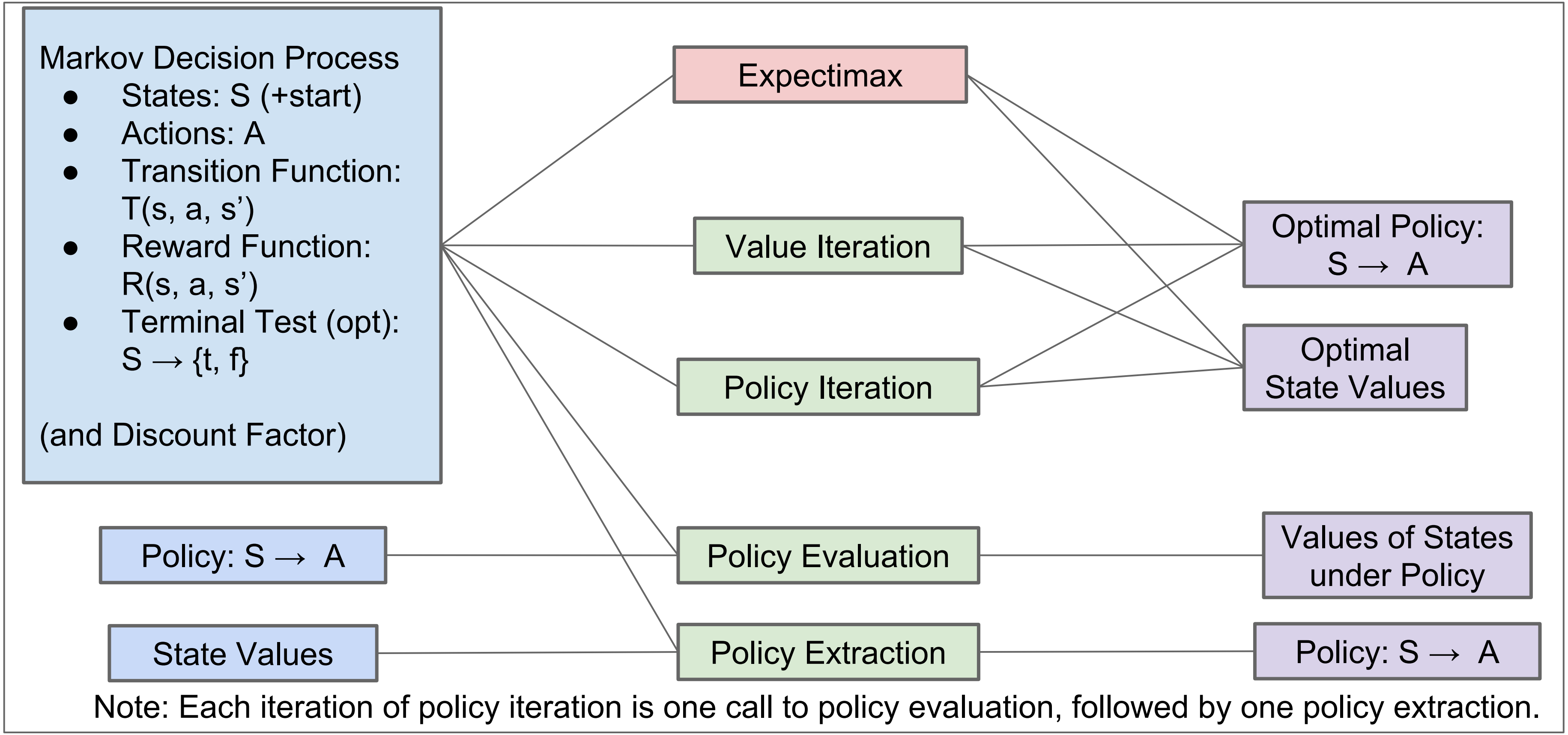
$$A \succ B \Rightarrow$$
$$(p \geq q \Leftrightarrow [p, A;\ 1-p, B] \succeq [q, A;\ 1-q, B])$$

Part I: Search -- Markov Decision Processes

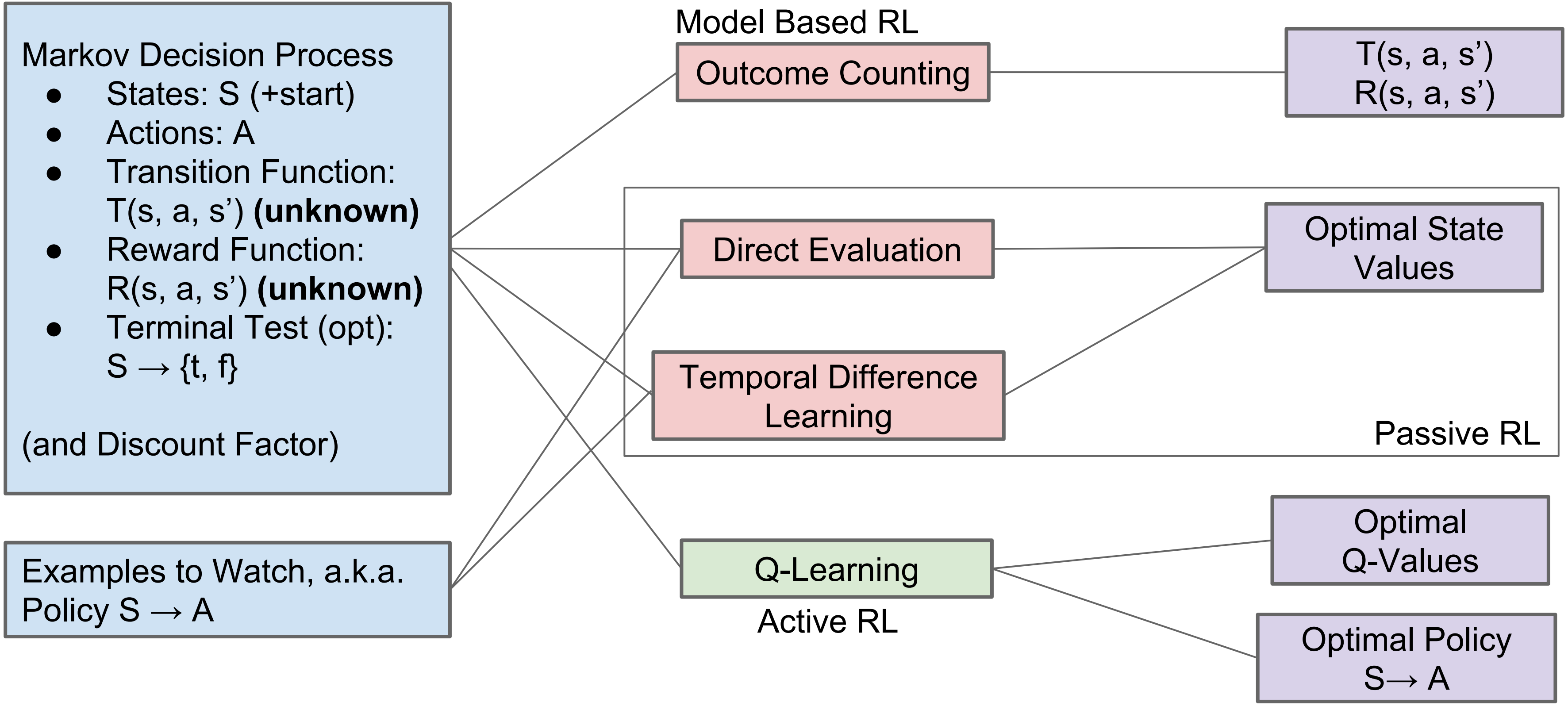


Note: Value Iteration is repeated application of Bellman Equation to find optimal state values.

Part I: Search -- Markov Decision Processes

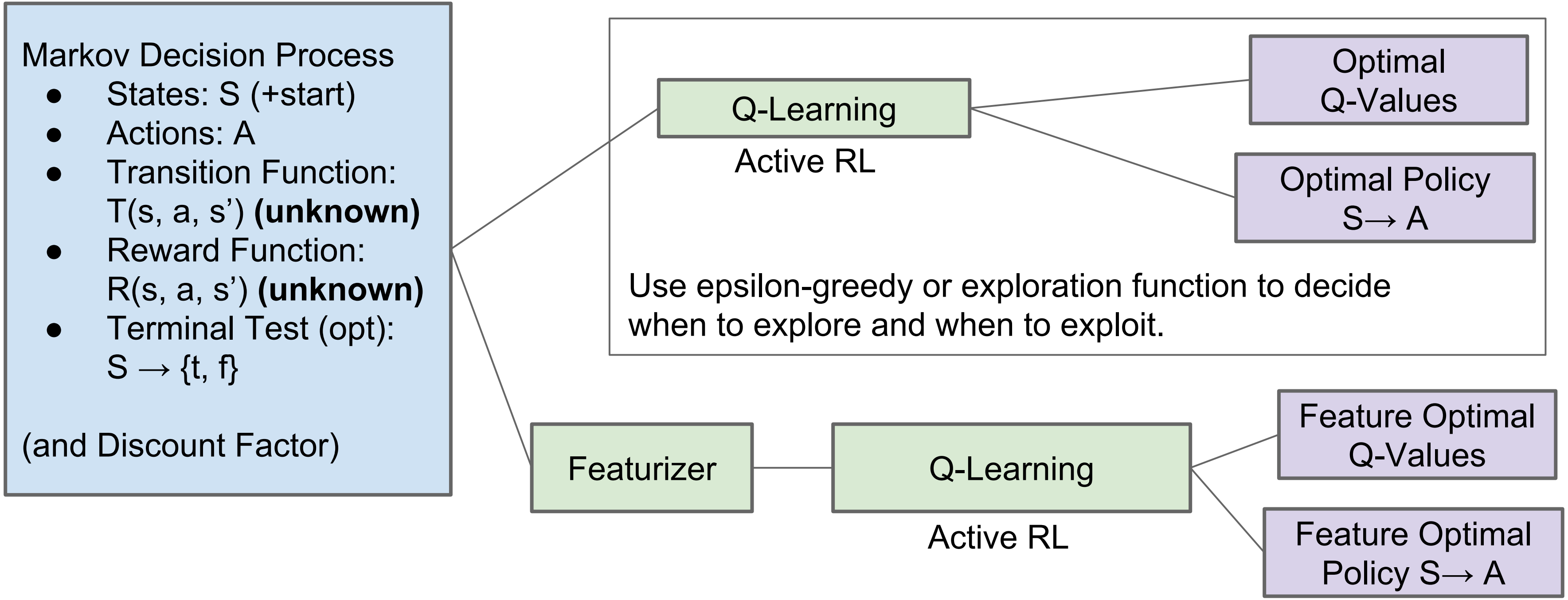


Part I: Search -- Reinforcement Learning



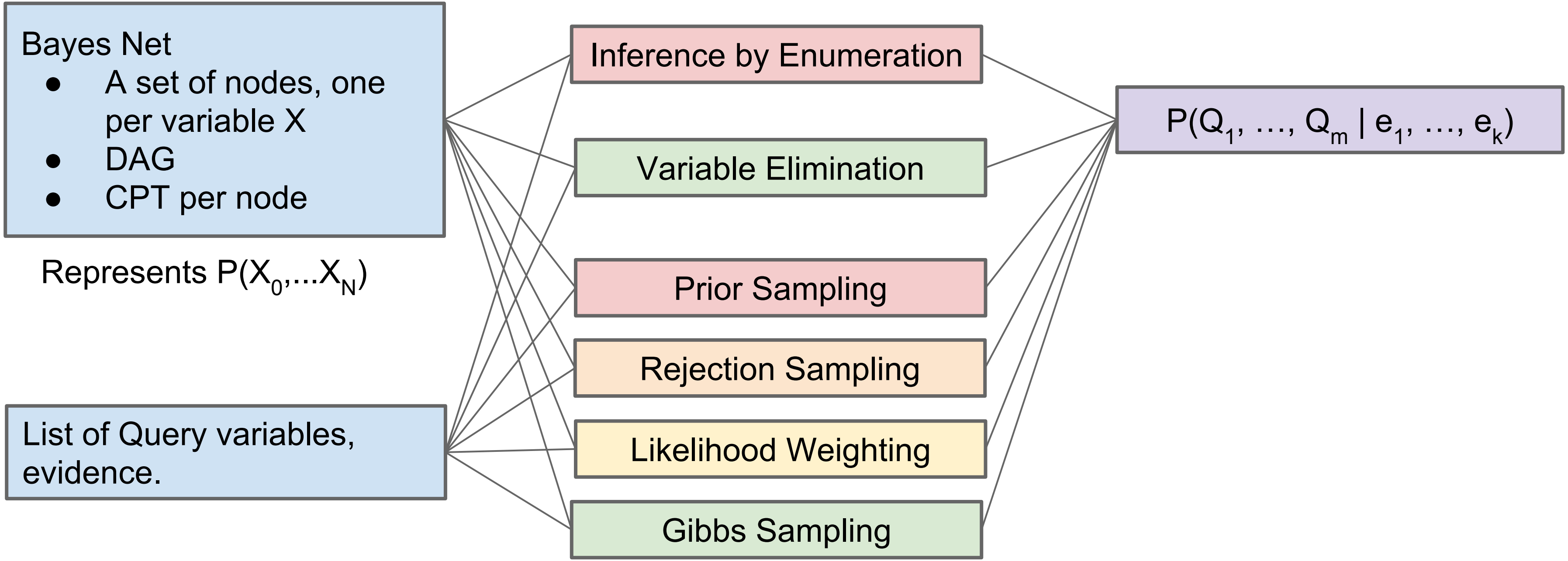
Note: TDL is actually good at what it does, but optimal state values are of questionable use.

Part I: Search -- Reinforcement Learning



Note #1: Important to adjust learning rate over time for Q-Learning. Policy
Note #2: TDL is actually good at what it does, but optimal state values are of questionable usesearch?

Part II: Inference -- Bayes Nets



Note #1a: Inference in Bayes Nets is NP-Hard for general case.
Note #1b: Polytrees (no undirected cycles) always allow efficient variable orderings.
Note #2: Can use samples directly to approximate a distribution even without a Bayes Net.
Note #3: Did not fully discuss in class how to do efficient resampling in one variable for Gibbs.

Part II: Inference -- Bayes Nets

Decision Networks
- Bayes Net
- Action Nodes
- Utility Node

Let $Z_1, \ldots, Z_k$ be parents of Utility Node.

$P(Z_1, \ldots, Z_m \mid e_1, \ldots, e_k)$

Computed using techniques from last slide.

Choice of Action A

Table Lookup

$EU(A \mid e_1, \ldots, e_k)$

$$MEU(e) = \max_a \sum_e P(s|e)U(s,a)$$

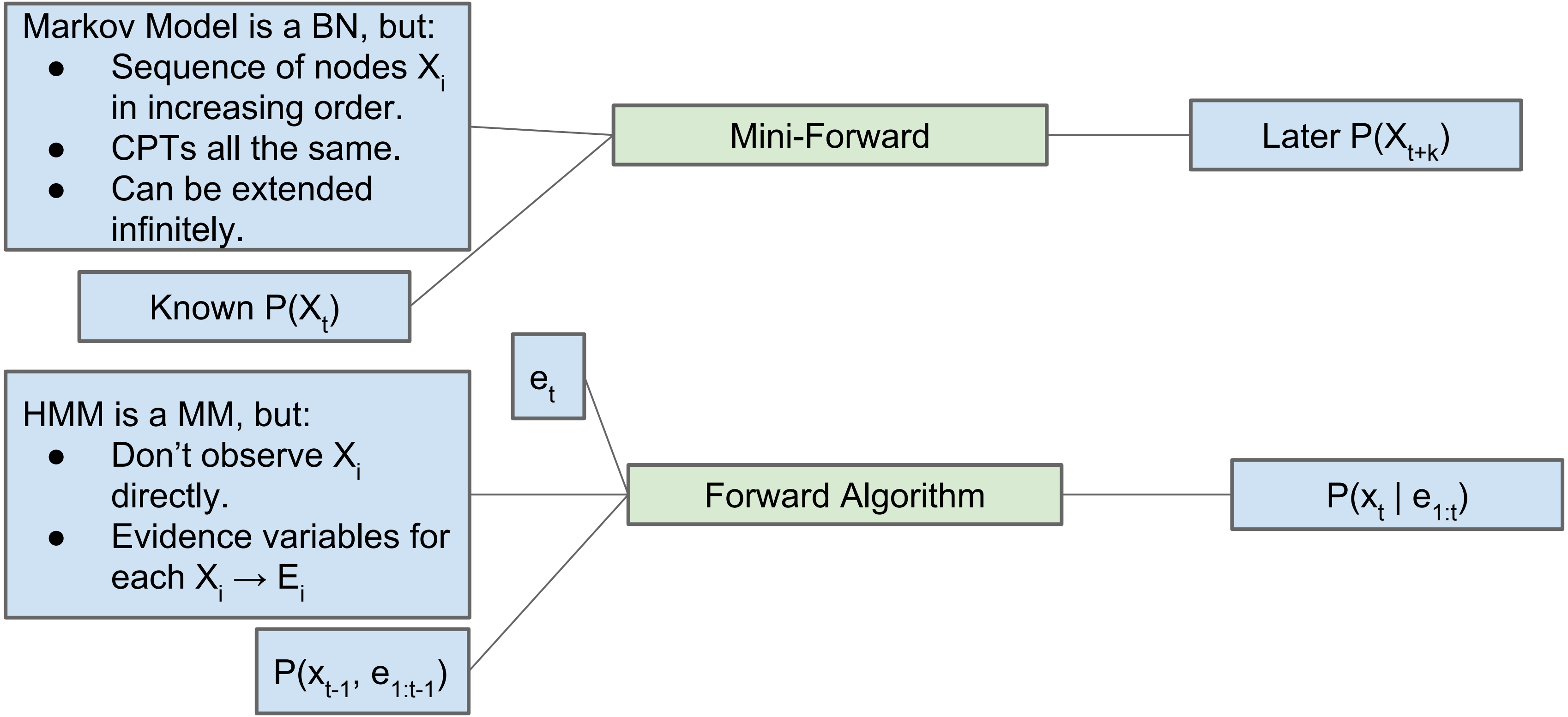
$$MEU(e,e') = \max_a \sum_e P(s|e,e')U(s,a)$$

$$MEU(e,E') = \sum_{e'} P(e'|e)MEU(e,e')$$

$$VPI(E'|e) = MEU(e,E') - MEU(e)$$

Note: Gives rise to POMDPs.

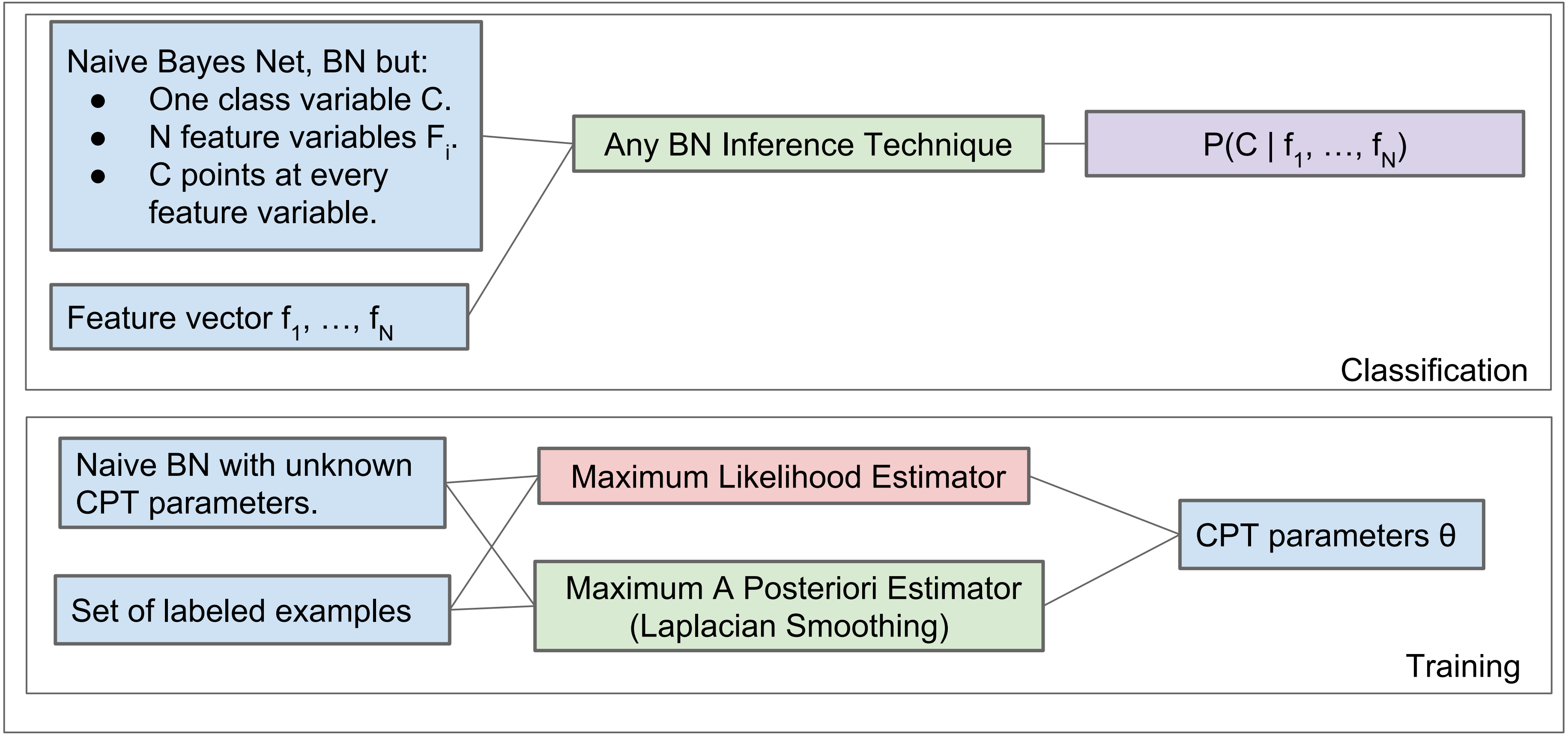
Part II: Inference -- HMMs



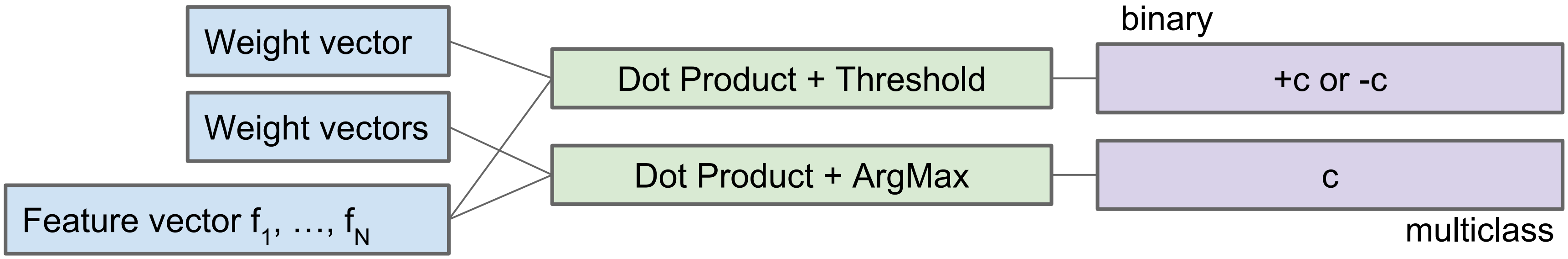
Note #1: A MM has a stationary distribution $P(X_\infty)$

Part II: Inference -- Particle Filtering [coming later]

Part III: Machine Learning -- Naive Bayes

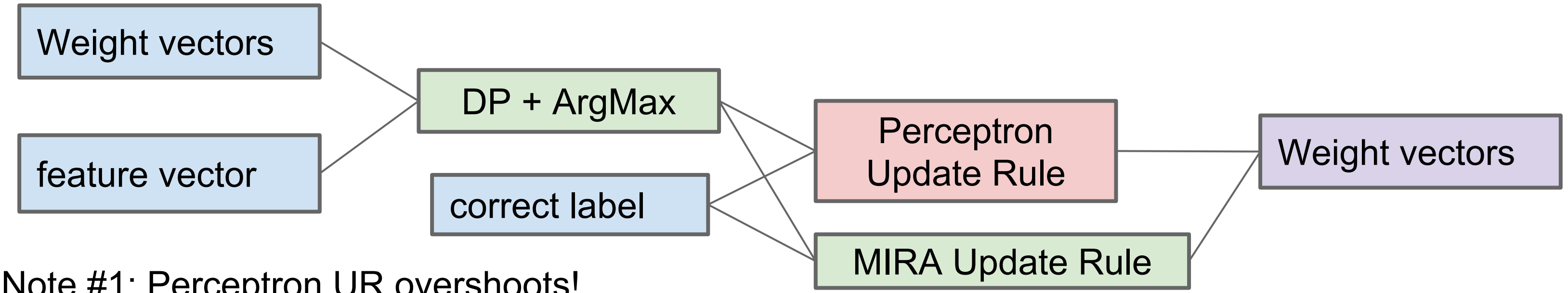


Part III: Machine Learning -- Perceptron



Note: Binary classifier draws separating hyperplane.

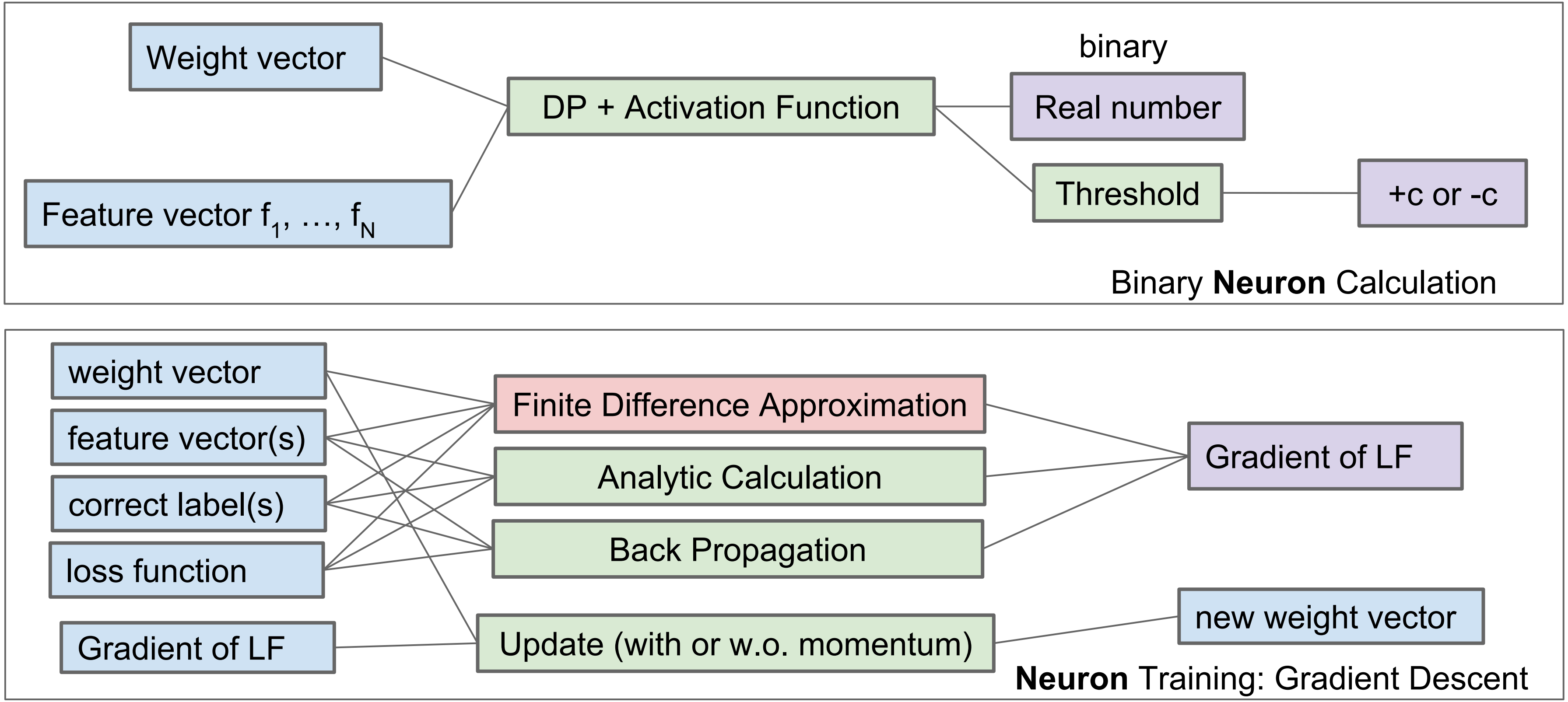
Perceptron Classification



Note #1: Perceptron UR overshoots!
Note #2: Similar classifier called SVM is very important. Covered in 189. Similar to MIRA, but optimizes over all data at once instead of single data point.

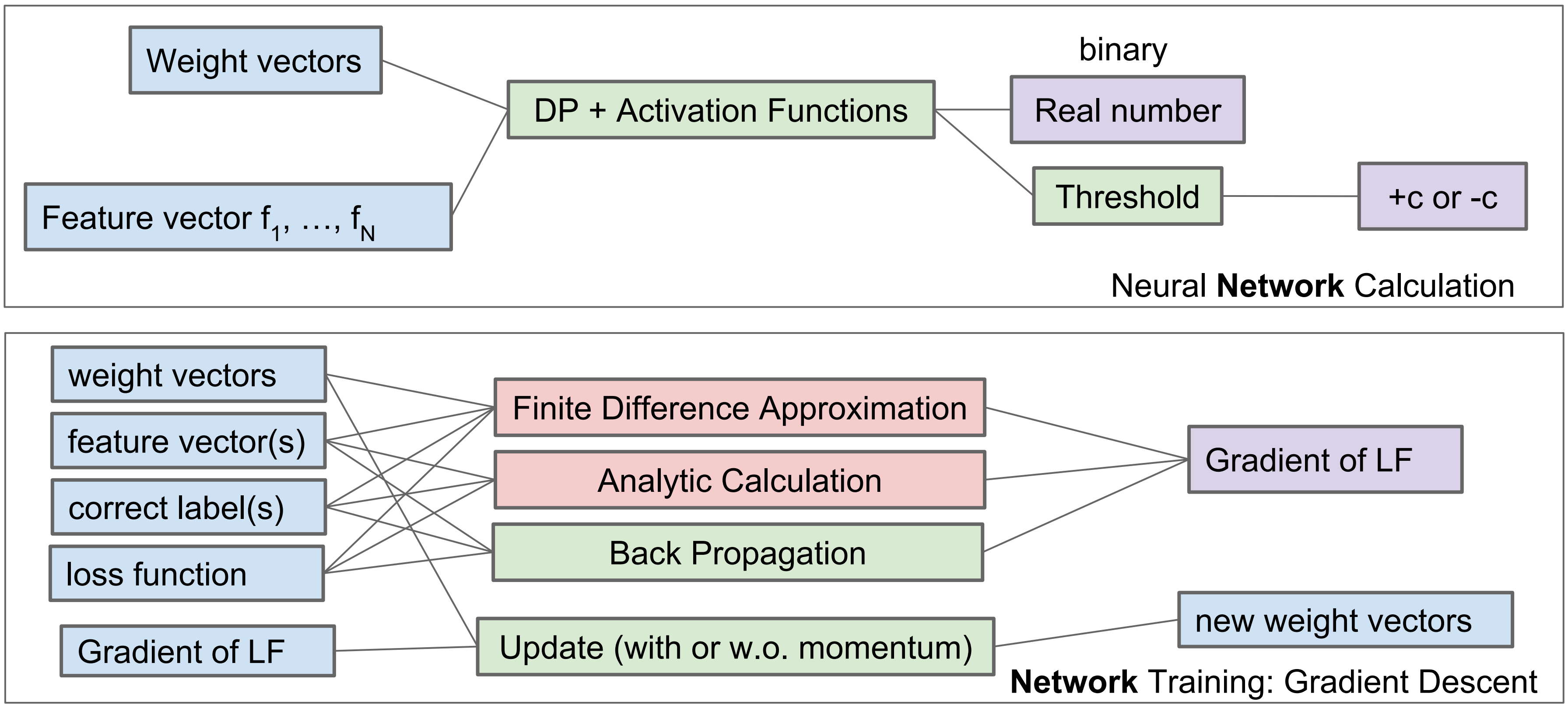
Training

Part III: Machine Learning -- Deep Learning / Multilayer Perceptron



Note: 1 sample is Stochastic Gradient Descent, k samples is mini-batching

Part III: Machine Learning -- Deep Learning / Multilayer Perceptron



Note: 1 sample is Stochastic Gradient Descent, k samples is mini-batching

Lecture 21/22

The Future of AI

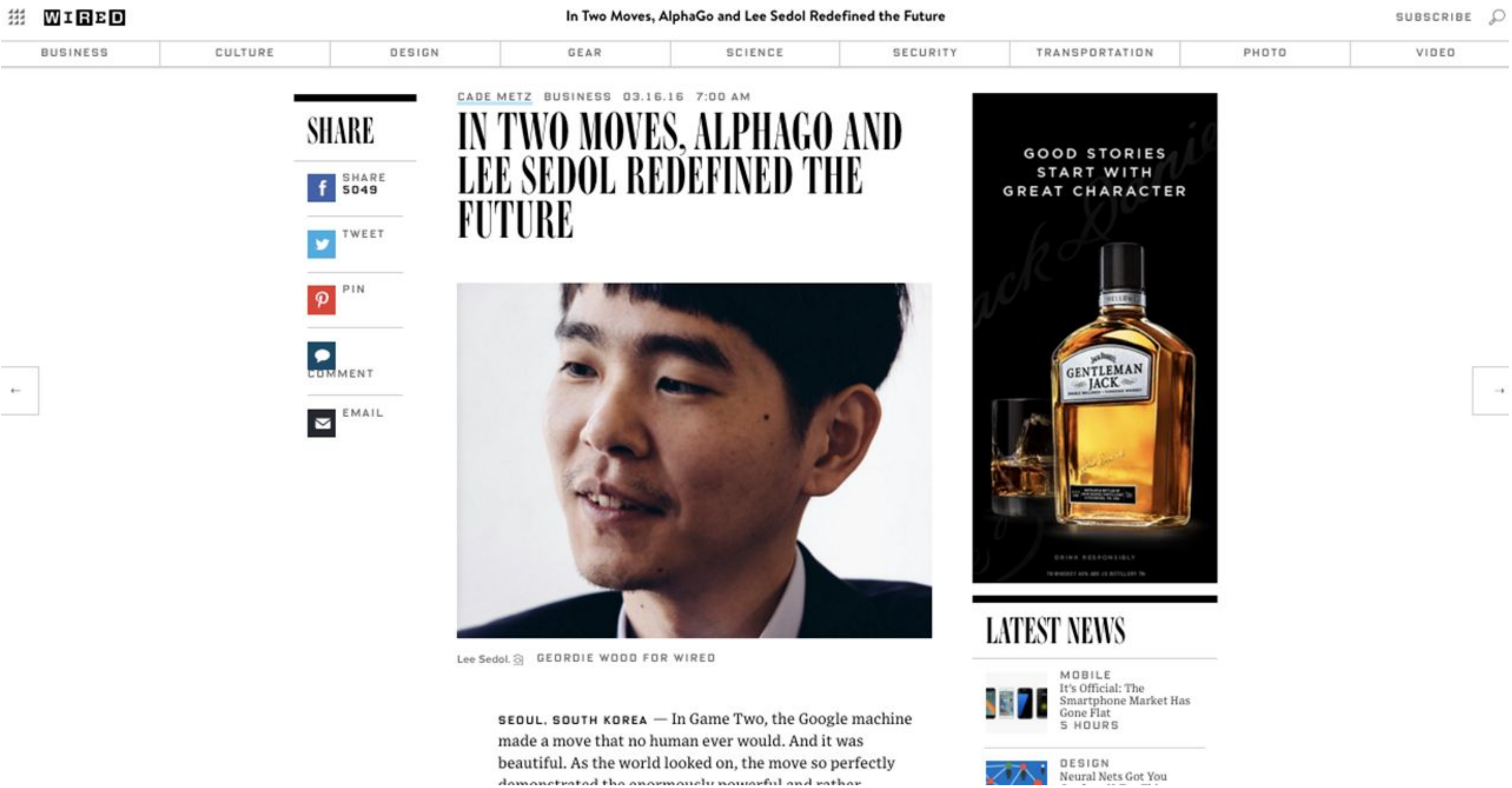
Google's DeepMind AI Can Lip-Read TV Shows Better Than a Pro

New Scientist (11/21/16) Hal Hodson

Researchers at Google's DeepMind and the University of Oxford are applying deep-learning techniques to a massive dataset of BBC TV programs to create a lip-reading system that can perform better than professional lip readers. The artificial intelligence (AI) system was trained using 5,000 hours from six TV programs that aired between January 2010 and December 2015.

The AI's lip-reading performance was then tested on TV programs broadcast between March and September 2016, accurately deciphering 46.8 percent of all words without any errors. In comparison, a professional lip reader deciphered just 12.4 percent of words correctly in a dataset of 200 clips. Many of the AI's errors were small, such as missing an "s" at the end of the word.

News



WIRED — In Two Moves, AlphaGo and Lee Sedol Redefined the Future — SUBSCRIBE

BUSINESS | CULTURE | DESIGN | GEAR | SCIENCE | SECURITY | TRANSPORTATION | PHOTO | VIDEO

SHARE
SHARE 5049
TWEET
PIN
COMMENT
EMAIL

CADE METZ BUSINESS 03.16.16 7:00 AM

IN TWO MOVES, ALPHAGO AND LEE SEDOL REDEFINED THE FUTURE

Lee Sedol. GEORDIE WOOD FOR WIRED

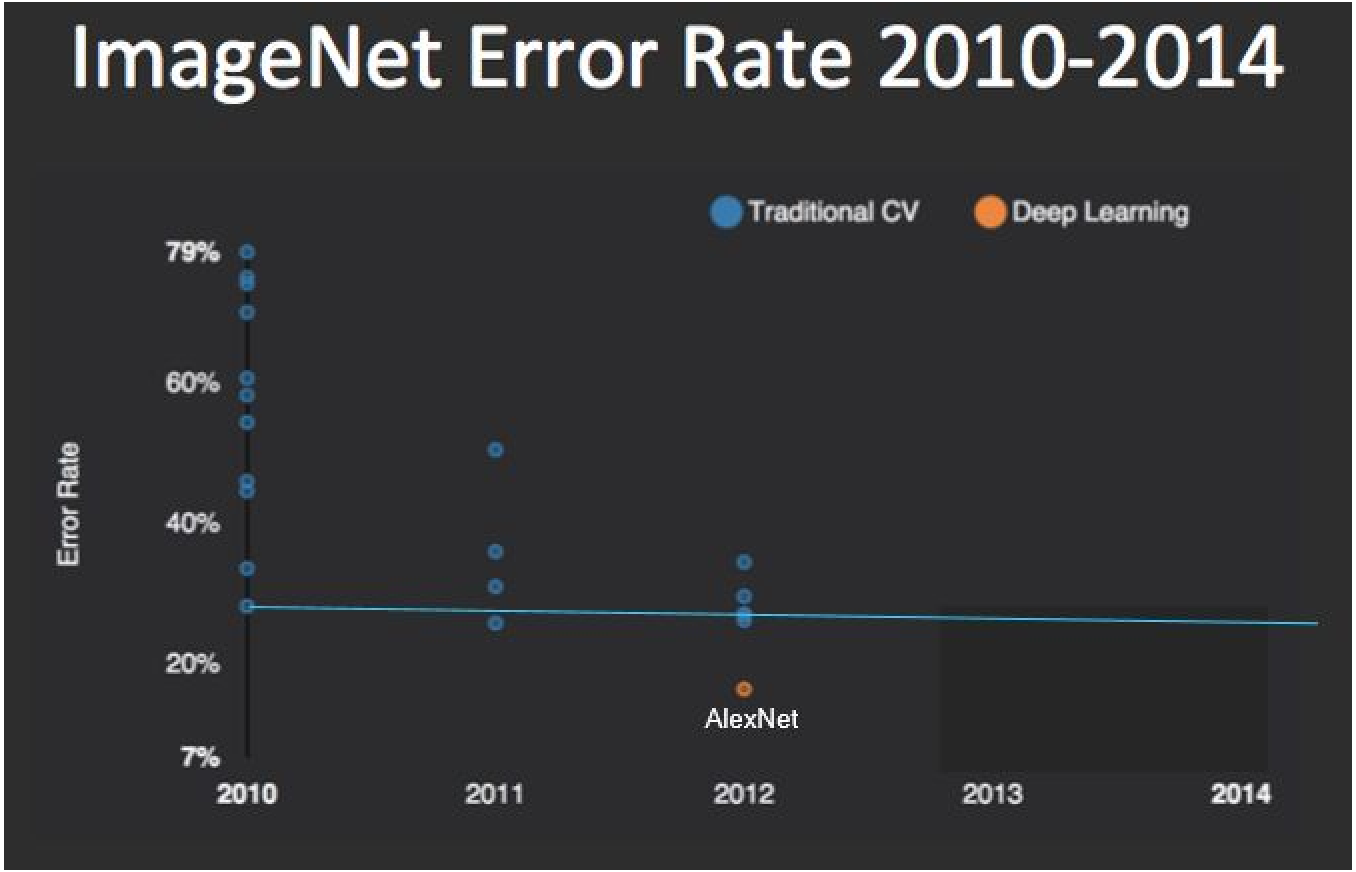
SEOUL, SOUTH KOREA — In Game Two, the Google machine made a move that no human ever would. And it was beautiful. As the world looked on, the move so perfectly

LATEST NEWS

MOBILE
It's Official: The Smartphone Market Has Gone Flat
5 HOURS

DESIGN
Neural Nets Got You

Old News (2012)



History of Deep Learning

Some key ideas:

- 1958: Perceptrons (Rosenblatt)
- 1986: Multilayer Perceptrons and Backpropagation (Rumelhart)
- 1989: Convolutional Networks (LeCun)
- 1993: Sparse Coding (Olshausen)
- 2000s: Sparse, Probabilsitic and Energy Models (Hinton, Bengio, LeCun, Ng)

Is deep learning 3, 30, or 60 years old?

(based on history by K. Cho)

What Changed by 2012

Data, e.g. ImageNet:

- 1.2 million original training examples.
- * 2048 (shifts)
- * 90 (color shifts)
- $\rightarrow$ ~200 billion training samples.
- Equivalent to 6.84 years of video at 1000 frames/second

Compute power (2012):

- Two NVIDIA GTX 580 GPUs.
- 5-6 days of training time to find parameters for an architecture.
    - "All of our experiments suggest that our results can be improved simply by waiting for faster GPUs and bigger datasets to become available."

What Changed by 2012

Nonlinearity:

- Went from Sigmoid activation function to ReLU activation function.
    - Roughly 10 times faster optimization! [5 days vs. 50 days]

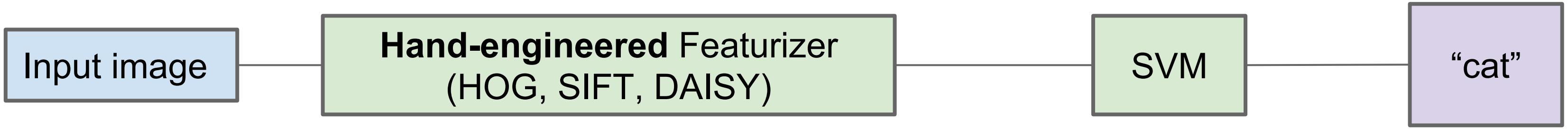
Regularization:

- Drop out (zeroing out features randomly during training)
- (Training data augmentation: Didn't discuss. Old idea, but now more compute power)

General Optimization Know-How (e.g. EE127)

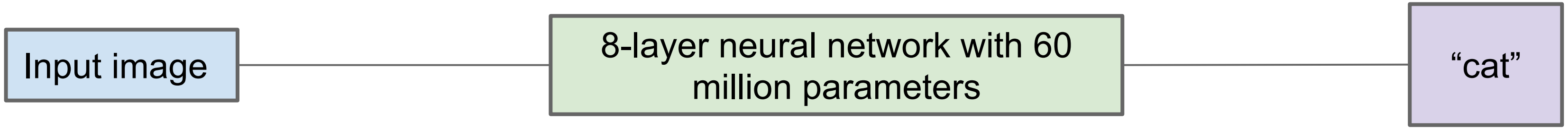
Computer Vision History

State of the art until 2012:



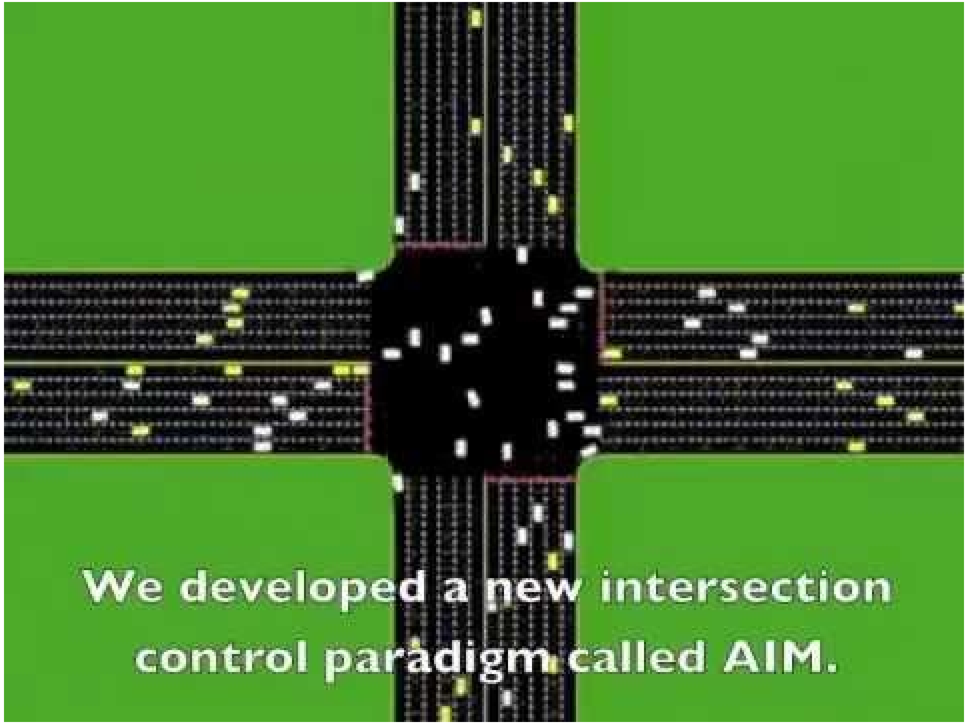
AlexNet (2012): Krizhevsky, Sutskever, Hinton

- 60 million learned parameters (now up to billions)
- 1.2 million training images



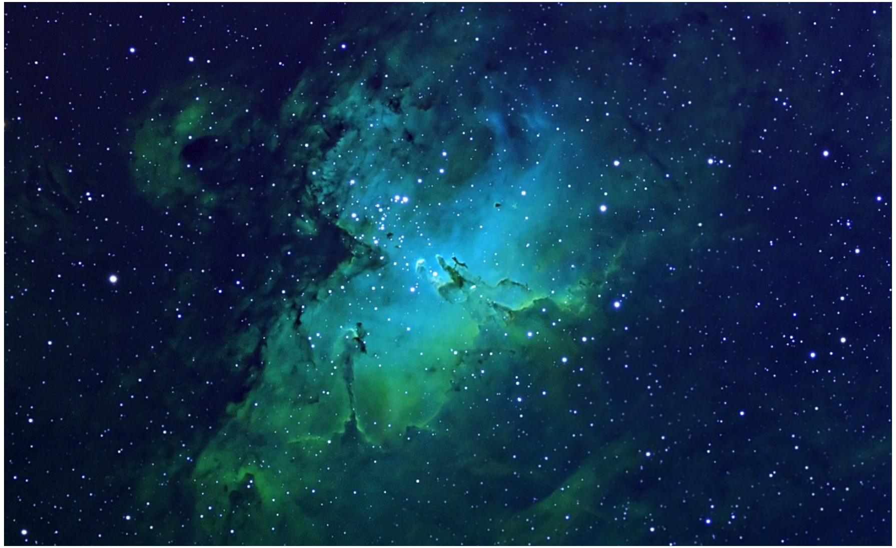
AI Will Fundamentally Change Society

Tesla Self Driving Demo: https://www.facebook.com/tesla/videos/10154712005882801/?pnref=story



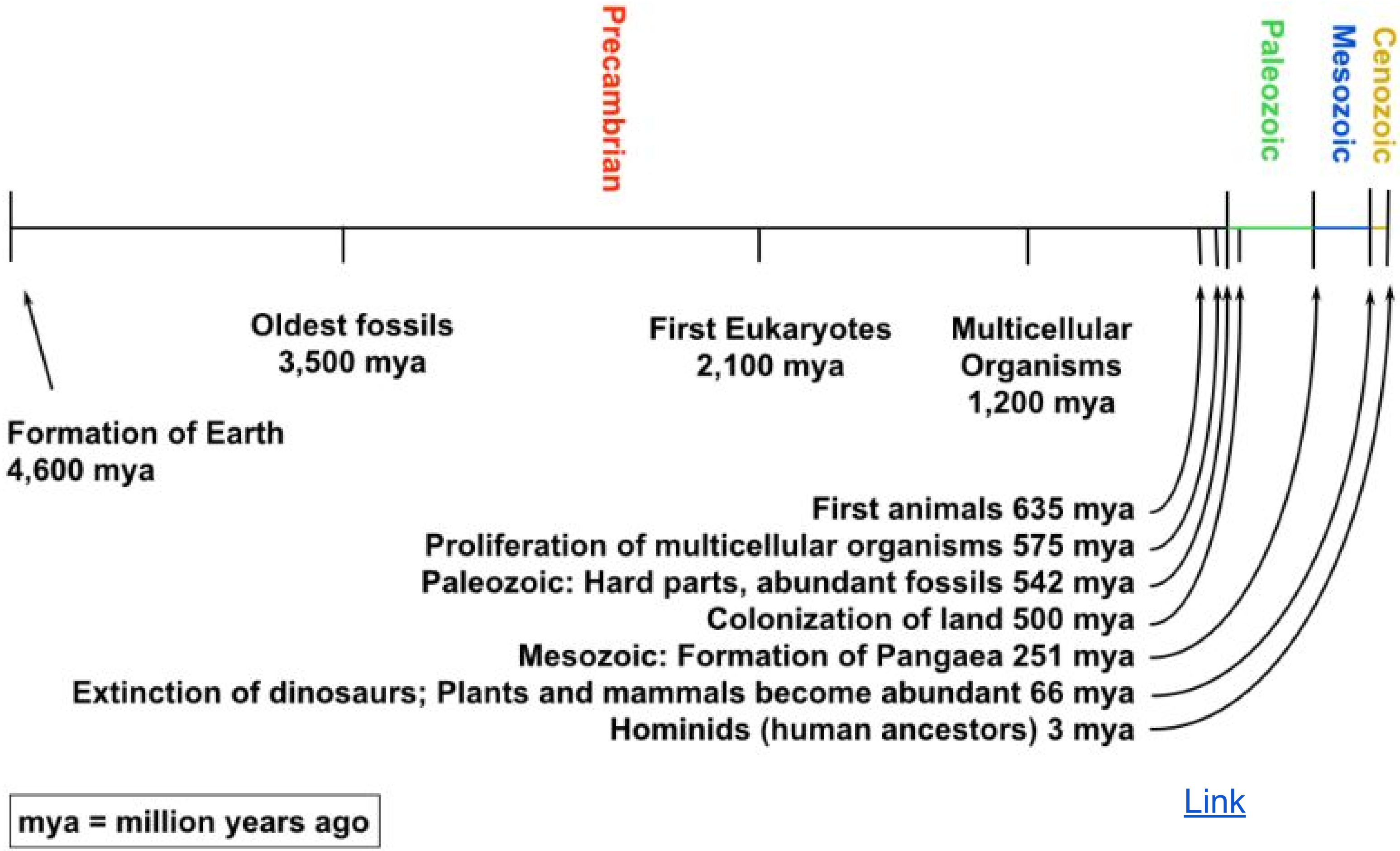
The Long View

Leaving a bunch of hydrogen lying around in space seems to result in rather complex outcomes.



Link

History of Biological Intelligence



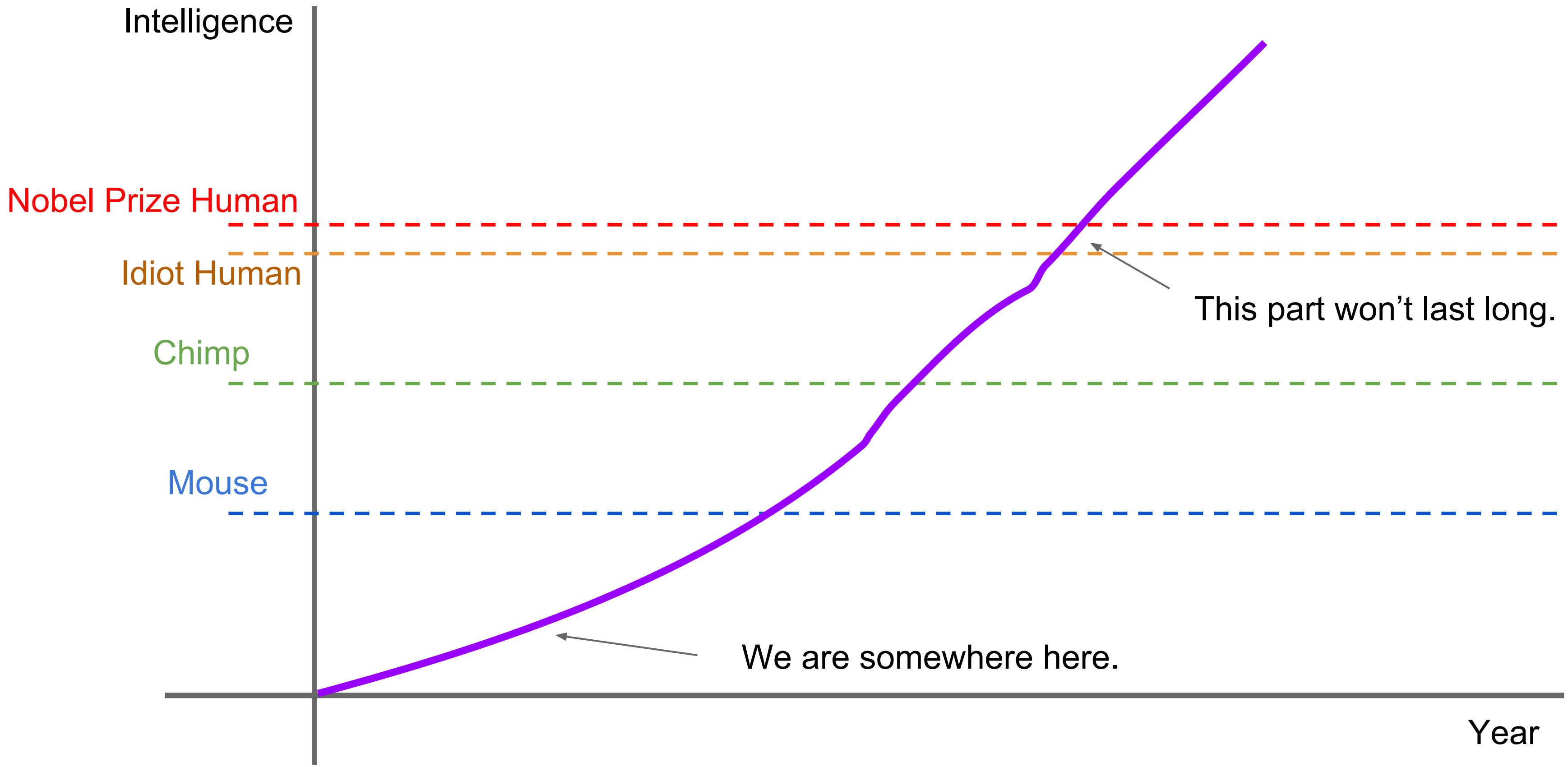
Link

Human Equivalent AI?

The ultimate achievement of AI would be human-equivalent intelligence.

- Assuming the world is purely physical (physical reductionism), cognition is achievable in ~50 watts and 3 pounds of hardware.
- Assuming that evolution is correct, possible to arrive at this design through a very crude and slow training process.
- Conceivable that in decades, centuries, or millennia that our human-directed-evolution-of-machines will arrive at human equivalent intelligence.
    - "...we might very well see relatively slow and incremental progress that doesn't really raise any alarm bells until we are just one step away from something that is radically superintelligent." - Nick Bostrom
- Then what?

Future History of AI (Naive)



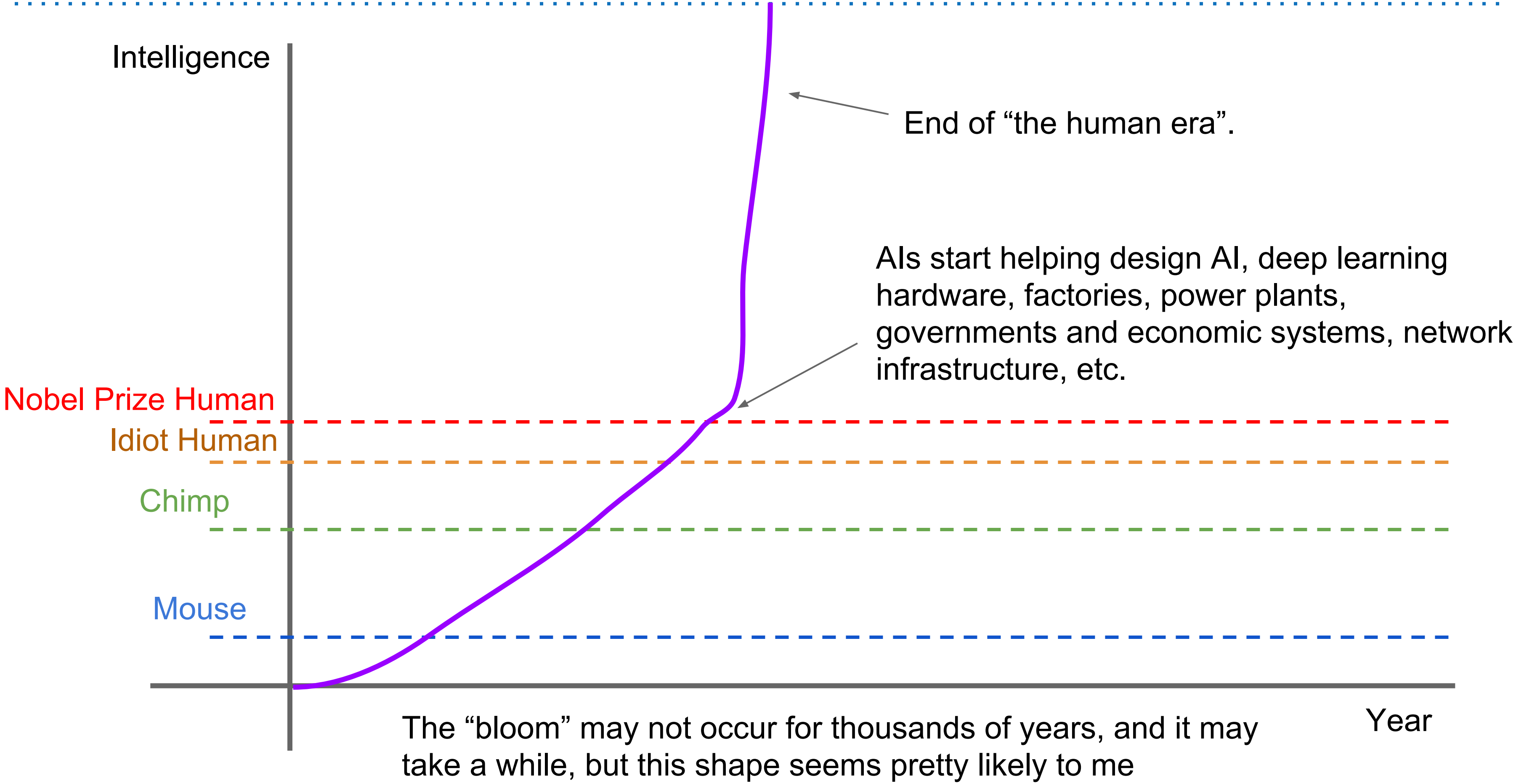
Human Equivalent Artificial Intelligence

Human equivalent intelligence could eliminate the need for any further invention by mankind.

- Want a new employee? Spin up a process.

Even more interesting: “Let an ultraintelligent machine be defined as a machine that can far surpass all the intellectual activities of any man however clever. Since the design of machines is one of these intellectual activities, an ultraintelligent machine could design even better machines; there would then unquestionably be an 'intelligence explosion,' and the intelligence of man would be left far behind.” - IJ Good (1965)

Future History of AI [My Best Guess]



Questions About Anything?

After 188

Major Things That Were Different This Time (From Spring 16)

- Week 1 through 8 notes (special thanks to **Nikhil** and Rosie).
  - Sorry we didn't produce them all semester, but I hope that you found them useful!
- Today's lecture slide summary of entire course.
- Question-goat questions.
- Substantial reorganization of many lectures (esp. Deep Learning).
  - In-lecture math review for ML topics (too slow??)
- Dedicated project-expert staff from previous semester (Anwar and Won).
- Two midterms instead of 1.
- Token based client side autograder (Thanks Allen!)
- New Project 6.
  - Sorry it's rough, but the old project was :(
  - Let me know what you thought of this flavor of "from scratch" project vs. the existing "fill-in-the-function" projects.

Things For Next Time

Possibilities:

- Cut CSPs to one day.
  - Go back in time and delete the “count the number of times this algorithm backtracks” style problem.
- Put D-Separation back in?
- More structure on learning the theory?
  - One or two fewer projects?
  - A few written homeworks?
- Make real-world usefulness of techniques more apparent throughout course.

Let me know what else you’ve missed.

Where to go next?

Congratulations, you've seen the basics of modern AI

- … and done some amazing work putting it to use!

What next?

- Machine learning: **cs189**, stat154
- Intro to Data Science: CS8 or new course (SP17?)
- Probability: ee126, stat134
- Optimization: **ee127**
- Cognitive modeling: cog sci 131
- Machine learning theory: cs281a/b
- Vision: cs280
- Robotics: cs287
- Algorithmic Human Robot Interaction: CS294-115
- NLP: cs288

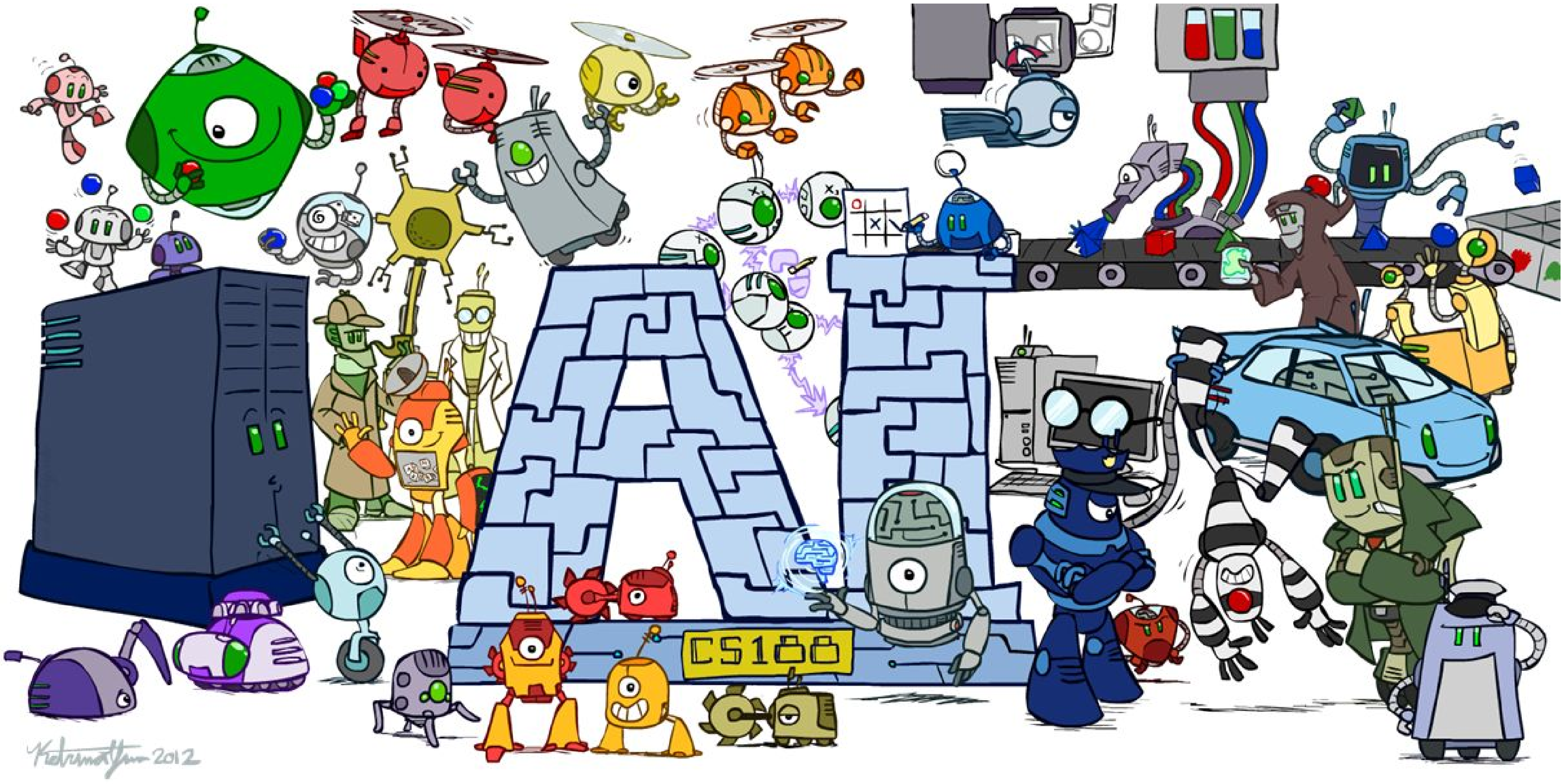
Getting Involved in AI/ML

For research opportunities:

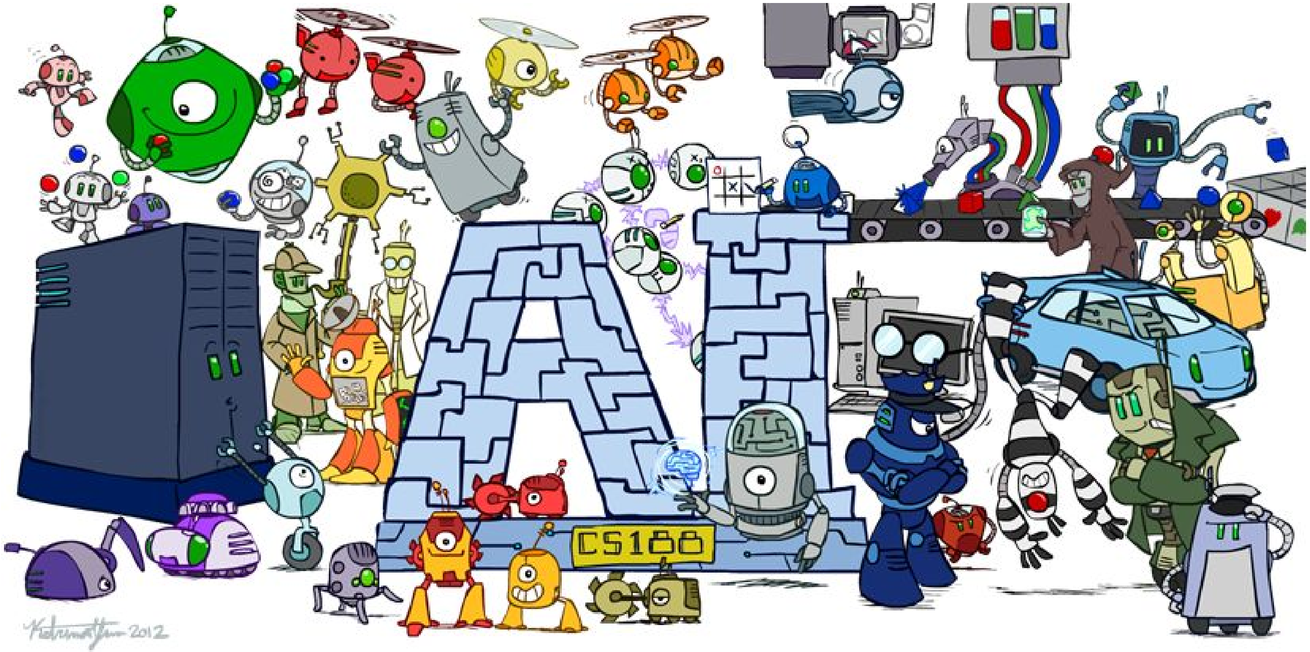
- See: https://eecs.berkeley.edu/research
- Talk to your TAs about dropping in on research group meetings.
- Volunteer to help out in lab (by talking to TAs, postdocs or professors)

One interesting opportunity:

- Professor Laurent El Ghaoui and one of my old grad school buddies Andrew Godbehere are working on a set of AI tools for understanding congressional transcripts.
    - Email me if you want to help out! Requires no fancy knowledge.


AI
CS188

Special thanks to the Ketrina Yim!



Ketrina Yim
CS 188 Artist

Special thanks to the Staff!

Becca Roelofs	**Steven Bi**	**Allen Guo**	**Rosie Jia**
Michael Laskey	**Sherdil Niyaz**	**Aldo Pacchiano**	Nick Rose
Nikhil Sharma	Caryn Tran	Anwar Baroudi	Won Park
Ted Xiao			

(Link)

And to Adam Janin as well!

- First time teaching in a long while, and quite a hero for operating in this strange 500+ student environment while also having another full time job!

Things to Ponder Filling Out Course Survey

New Aspects of Course: What did you like / not like?

- In-lecture question-goat, reworking of lectures to have a strong story (but sometimes slower than previous semesters)
- Two midterms instead of 1.
- New project 6: Build and train perceptron and neuron from scratch.
- Week 1 - 8 lecture notes.
- Token-based client side autograder.

Coming up? What do you think your successors will want?

- Written homeworks or other ways to reinforce understanding of theory?
- Increase obviousness of connections to real world?
- Cut some CSPs, add back D-separation officially?
- Changes you'd like to projects?