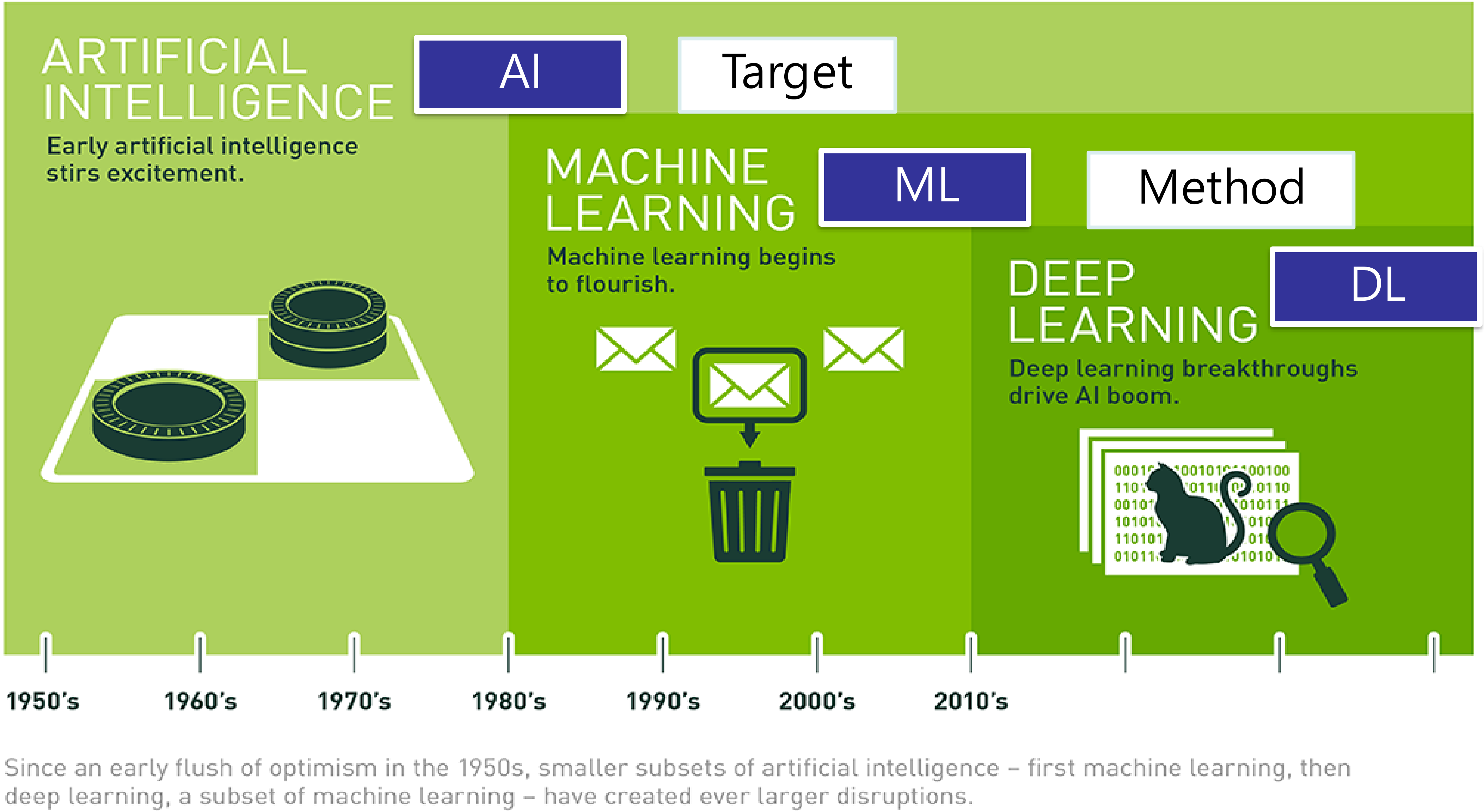
ECE 661 COMP ENG ML & DEEP NEURAL NETS

1. INTRODUCTION

HAI LI, SPRING 2025

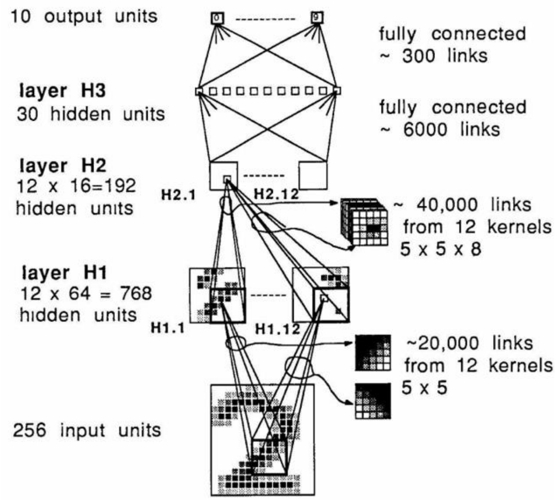
AI ↔ ML ↔ DL



Source of image: https://blogs.nvidia.com.tw/2016/07/whats-difference-artificial-intelligence-machine-learning-deep-learning-ai/

Historical Overview

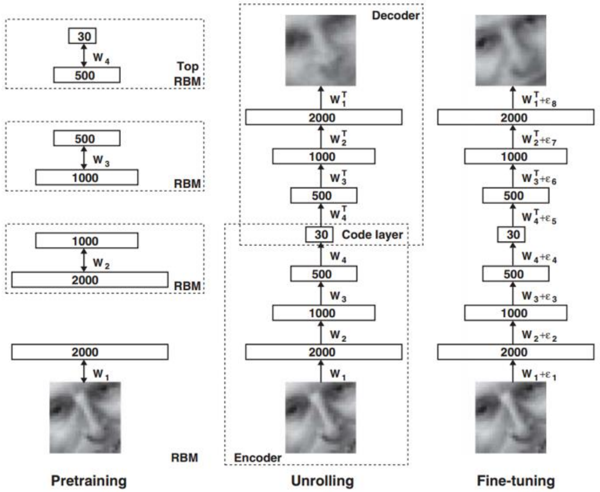
Convolutional Network
(1980s)



Dark period
(1990s)

- Serious problem:
  Vanishing gradient
- No benefits observed
  by adding more layers
- No high-performance
  computing devices

Renaissance
(2006 ~ Present)



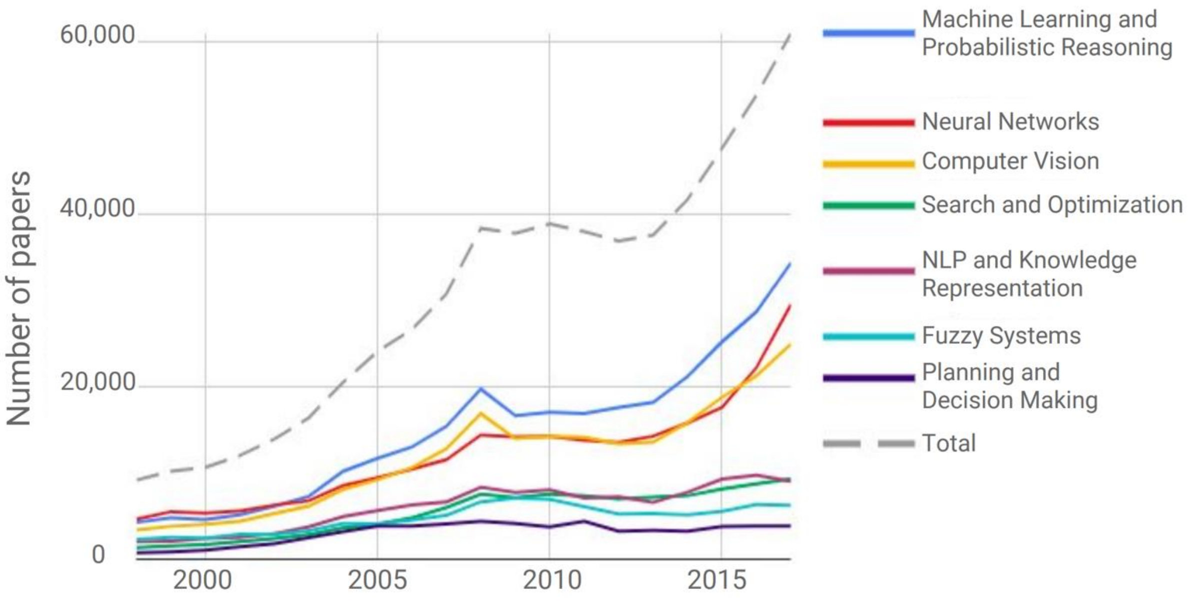
Y. Lecun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard and L. D. Jackel. Backpropagation Applied to Handwritten Zip Code Recognition.1989.

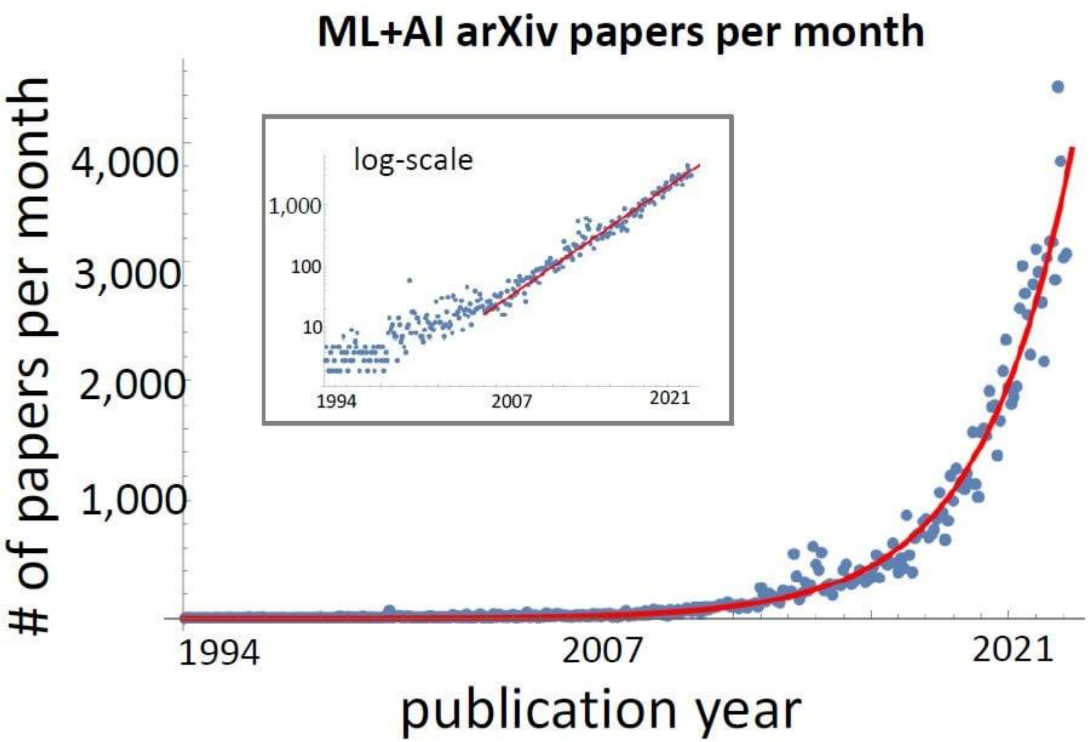
J. Schmidhuber. Deep Learning in Neural Networks: An Overview. arxiv, 2014.

G. E. Hinton and R. R. Salakhutdinov. Reducing the Dimensionality of Data with Neural Networks. Science, 2006.

AI Papers

The number of AI papers on Scopus





Source of image: http://cdn.aiindex.org/2018/AI%20Index%202018%20Annual%20Report.pdf (left)
https://www.reddit.com/r/singularity/comments/xwdzr5/the_number_of_ai_papers_on_arxiv_per_month_grows/ (right)

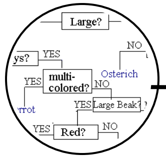
Historical Milestones

Timeline

1943 First Neural Network Model

1941 First Electronic Computer

1970 Expert Systems

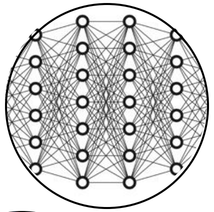


1950 Turing Test

1997 Deep Blue Defeats Kasparov

1990 AI in the Military

2006 The Concept of Deep Learning



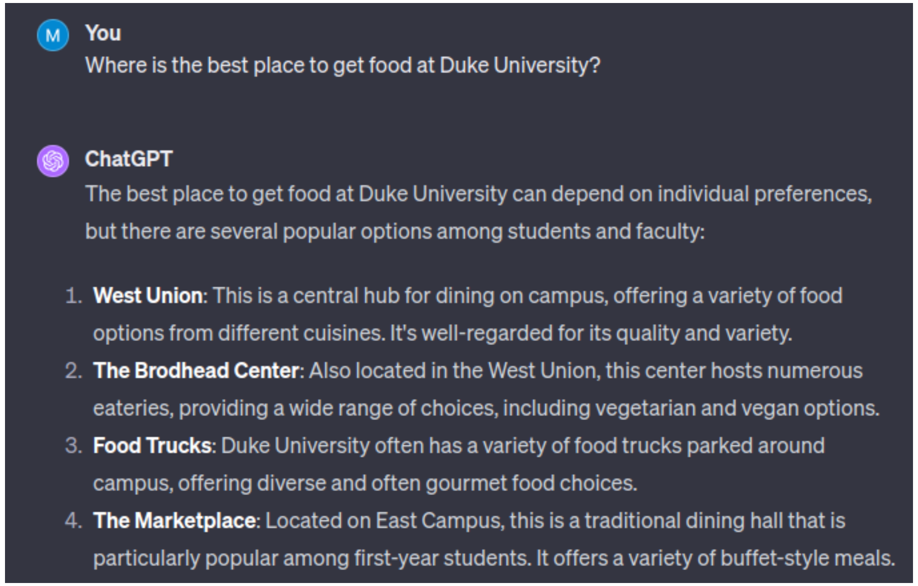
2012 Google X learns to Find Cat Videos

2016 Alpha Go Defeated Lee Sedol

~~Now 180K $/year for AI Engineer~~

Recent Development: ChatGPT and LLMs

- The recent creation of ChatGPT has drawn massive attention to AI.
- Large Language Models (LLMs) have been a developing research area for years, but ChatGPT was the first highly competent chatbot.
- Useful for simple automation tasks, coding, teaching, etc., leading to a renaissance in LLM-based products and research.
- Prone to hallucination and not human-level at reasoning/planning, but very knowledgeable across a wide range of topics.



Source: https://upload.wikimedia.org/wikipedia/commons/thumb/0/04/ChatGPT_logo.svg/1200px-ChatGPT_logo.svg.png

Outline

- Course Introduction
  - Refer to the course syllabus for details
- Machine Learning & Deep Neural Networks
  - Applications
  - Categories
  - Important Metrics

Course Objectives

- For MS/MEng students and undergraduate students who want to learn computer engineering methods commonly performed in developing and using machine learning and deep neural network models.
- For PhD students who want to learn and practice a wide variety of ML topics that are beyond any single focus area. The breadth of knowledge covered may spur new ideas to be used in your own research.

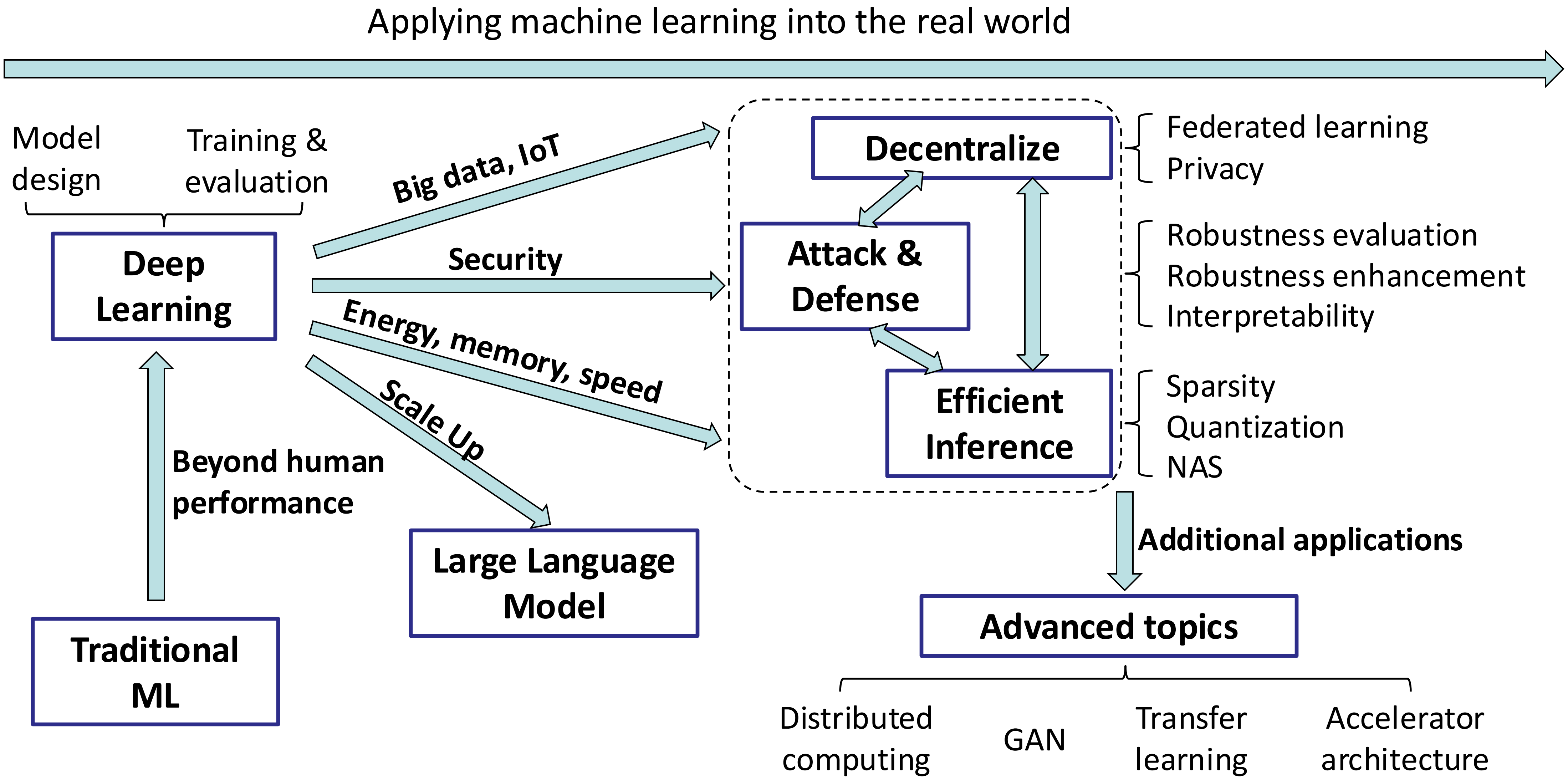
- ***<u>Practice</u>*** will be the focus of this course, while ***<u>theoretical understanding</u>*** is essentially important.

Course Objectives

This course is designed to improve your ability to:

1. **Comprehend** the mechanisms, applications, and limitations of techniques commonly used in training and inference of machine learning and deep neural networks algorithms;
2. **Formulate** hypotheses and conduct experiments employing these techniques;
3. **Analyze** experimental results obtained by these techniques and your own practices and **derive** the conclusions that are supported or not supported by your data;
4. **Synthesize** and **communicate** the experimental results and data through oral narrative, graphs, figure legends, and result narratives;
5. **Utilize** proper engineering techniques for novel machine learning algorithms and deep neural network models;
6. **Propose** new engineering approaches and techniques to further enhance machine learning and deep neural network training and inference execution.

Course Roadmap



Course Overview

- DNN fundamentals
  - Forward/backward propagation, training, convolutional neural network (CNN), network architecture, recurrent neural network (RNN), language models, …
- DNN acceleration
  - Compact neural architecture, model compression, pruning, quantization, sparsification, …
- Machine learning security
  - Adversarial attack, robust learning method, …
- Advanced topics
  - Large language model (LLM), Distributed computing, neural architecture search (NAS), transfer and reinforcement learning, generative adversarial network (GAN), decentralization and privacy, DNN accelerator, …

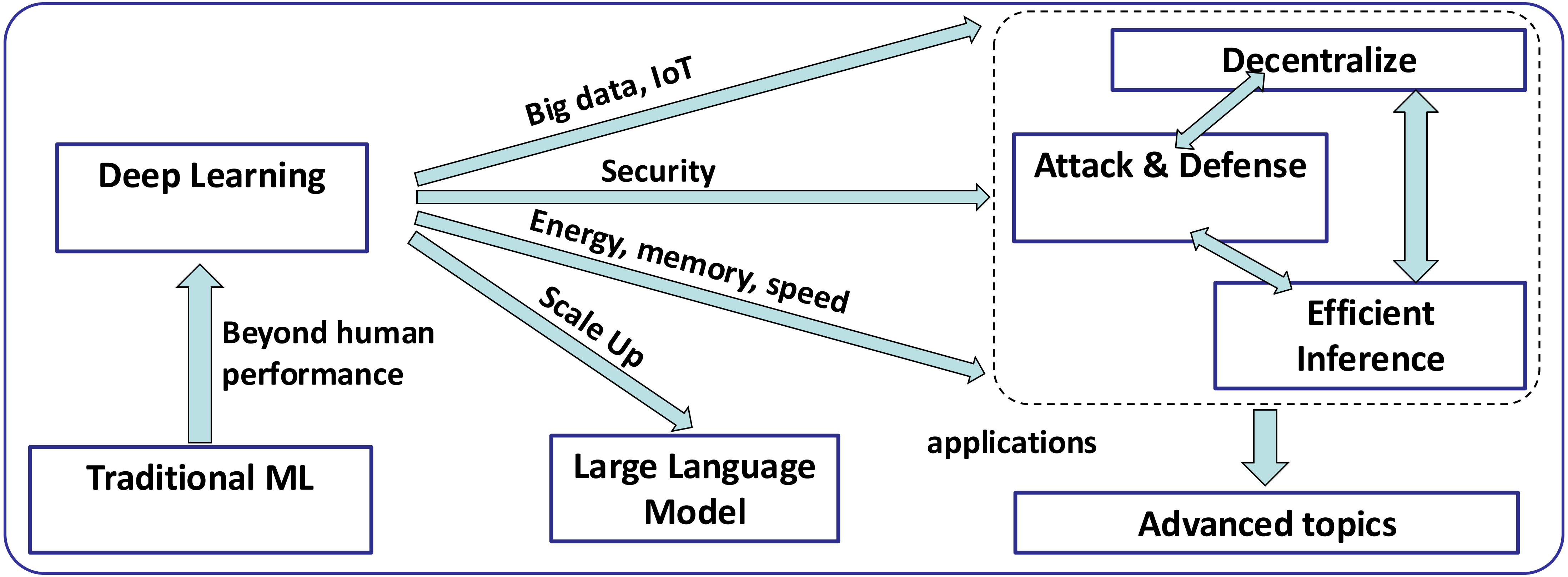
Spring 2025 Tentative Schedule *(Subjective to change)*

Week	Date	Lecture	Content	Assign	Due	Assignment
1	1/8/25	Course Introduction	Lec01			
2	1/13/25	Perceptron and back propagation	Lec02	**HW 1**		Gradient computation, simple CNN
	1/15/25	Image feature and 2D convolution	Lec03			
3	1/20/25	**Martin Luther King Jr. Day holiday; No Class**				
	1/22/25	Convolutional Neural Network (CNN)	Lec04		**Class Drop/Add ends**	
4	1/27/25	CNN Training - Basic	Lec05			
	1/29/25	CNN Training - Basic & Advanced	Lec06			
5	2/3/25	CNN Architectures	Lec07	**HW 2**	**HW 1**	Advanced DNN
	2/5/25	Compact Neural Architecture Design	Lec08			
6	2/10/25	RNN and Language Models	Lec09			
	2/12/25	Attention Model & Transformers	Lec10			
7	2/17/25	Large Language Model Inference	Lec11	**HW 3**	**HW 2**	RNN & LLM
	2/19/25	Large Language Model Training	Lec12			
8	2/24/25	Deep Learning Hardware Systems / Project Introduction	Lec13	**Project idea**		
	2/26/25	Deep Compression	Lec14			
9	3/3/25	Sparse Regularization	Lec15		**HW3**	
	3/5/25	Sparse Optimization	Lec16			
10	3/10/25	**Spring recess; No Class**				
	3/12/25	**Spring recess; No Class**			**Project Proposal**	
11	3/17/25	Fixed-point Quantization	Lec17	**HW4**		Pruning and Fixed-point Quantization
	3/19/25	LLM Optimizations	Lec18			
12	3/24/25	LLM Optimizations				
	3/26/25	Machine Learning Security	Lec19	**HW 5**		Adversarial attack and adversarial training
13	3/31/25	Adversarial Attack - Attacks	Lec20		**HW 4**	
	4/2/25	Adversarial Attack - Defenses	Lec21		**Project Mid-point Check-in**	
14	4/7/25	Federated Learning	Lec22			
	4/9/25	Transfer learning / Generative Models	Lec23		**HW 5**	
15	4/14/25	AutoML / Neural Architecture Search	Lec24			
	4/16/25	**Lab Q&A**				
16	4/21/25	**Lab Q&A**				
	4/23/25	**Project Poster Session** **(TBD, tentatively 4/23 1-4pm)**			**Poster & Report**	

Related Topics and Courses

NLP: **ECE 684**
Deep learning: **ECE 685**

Cloud computing: **ECE 563**; Smart sensor: **ECE 590-04**
Security: **ECE 663**; Image processing & denoise: **ECE 588**
Information theory: **ECE 587**; Compressed sensing: **ECE 741**



Math basics: **ECE 581, 586**
Machine learning: **ECE 681, 682, 687, 689**

Implementation: **ECE 550, 551, 650**
Architecture design: **ECE 552, 654**
System optimization: **ECE 558, 563, 565**
Hardware: **ECE 538, 539, 559, 590 (ML accel.)**

“Learn by Doing”

- Five (5) assignments (conceptual questions + labs in PyTorch)
  - 1: Building and understanding CNN modules
  - 2: CIFAR-10 training
  - 3: RNN model & Large Language Model
  - 4: Advanced sparse optimization and quantization techniques
  - 5: Adversarial attack and adversarial training

- One (1) project

- Multiple in-class quizzes

Prerequisites

- We expect that students to have basic object-oriented programming experience (e.g., C++, Python) and be familiar with linear algebra and computer hardware fundamentals prior to taking this course, such as
  - For graduate students: ECE 551D
  - For undergraduate students: CS201
- If you are familiar with a topic that we are covering …
  - You may learn something new
  - I may present it slightly differently than you are used to
  - You may be able to help other students learn it
- If you do not have these pre-requisites and are unfamiliar with these topics
  - We will **NOT** be slowing down to cover them
  - Please come talk to me or a TA sooner rather than later!

Logistics

ECE 661 COMP ENG ML & DEEP NEURAL NETS		
Faculty:	Dr. Hai "Helen" Li	Hai.li@duke.edu
Lectures:	Mondays/Wednesdays 10:05 AM – 11:20 AM FITZPATRICK SCHICIANO B 1466 In person only. No recording	
Office Hours:	By Appointment (please send email to Dr. Li)	
Teaching Assistants:	Junyao Zhang Ben Morris Easop Lee James Kiessling Mark Horton Zhixu Du	jz420@duke.edu ben.morris@duke.edu easop.lee@duke.edu james.kiessling@duke.edu mark.horton@duke.edu zhixu.du@duke.edu
Office Hours:	TBD	

TAs are NOT under obligation to bail you out at 3am or debug your code.
Your best bet is to get help in a timely and reasonable manner!
Office hour starts from Friday of the second week (January 17).

Getting Info

- **Canvas:**
  - Syllabus, schedule, slides, assignments, rules/policies, prof/TA info, office hour info, gradebook
  - Links to useful resources and Gradescope

- **Slack workspace:** questions/answers
  - Use you Duke email to sign up at the following link (expired in 30 days, join ASAP) https://join.slack.com/t/ece661-25sp/shared_invite/zt-2xkdb7i8g-VJk6goye~M1Df1FZ_jXGcQ **Post all your questions here**
  - Questions must be “public” unless good reason otherwise
  - No code in public posts!

- **Gradescope**
  - Homework submission, grading and regrading requests

Getting Answers to Questions

- What do you do if you have a question?
  - Check Canvas (Announcements)
  - Check Slack
    - If you have questions about homework, use Slack – then everyone can see the answer(s) posted there by me, a TA, or your fellow classmate
  - Contact TA directly if need additional background materials for prerequisite knowledge
  - Contact professor directly if issue that is specific to you and that can’t be posted publicly (e.g., regrade)

Textbook & Software

- There are no designated textbooks for this course yet.
  - We are writing it (with significant delay)
- The related reading materials (e.g., papers, webpages, etc.) will be distributed through Canvas before the classes.
- We use PyTorch (https://pytorch.org/) in this course

PyTorch

Grading

Assignment	%
Assignments (5)	55%
Project	25%
Quiz	20%

- Completion of all assignments is required in order to earn a passing grade of D- or better in this course.
- Course grades are determined using an absolute, but adjustable scale (i.e., there is no curve). A final course average (rounded to the nearest 0.1 point) of at least 93.3 = A, 90.0 = A-, 86.7 = B+, 83.3 = B, 80.0 = B-, etc.
- Note: the professor reserves the rights to scale the grades.
- Expect 6 **in-class** quizzes, dropping the lowest (or the missing) one.

Homework Submission

- Homework assignments and lab reports will be submitted as **PDF files and code files** through the Assignments tool in Gradescope. The details will be given in assignments.
- Late policy
  - < 24 hours late: deduct 10% credits
  - < 48 hours late: deduct 25% credits
  - No credit for late work after 48 hours
- Strict cutoff time will be enforced based on submission timestamp.
- Consider a small margin in case of system/internet issue.
- Homework bonus credit is added to each corresponding homework assignment, up to a maximum score of 100.

Grade Appeals

- All re-grading requests must be submitted in **Gradescope**
  - A brief written description of the error
- I will respond to your regrade request and make arrangement to return your work to you.
- As a matter of policy, when you request re-grading, you agree that the grading of the entire assignment may be re-evaluated.
- All re-grading requests must be submitted no later than 1 week after the assignment was returned to you.

Academic Misconduct

- Academic Misconduct
  - Refer to **Duke Community Standard**
  - Homework/lab is individual – you do your own work
  - Common examples of cheating:
    - Running out of time and using someone else's output
    - Borrowing code from someone who took course before
    - Using solutions found on the Web
    - Having a friend help you to debug your program
- We will not tolerate any academic misconduct!
  - We use software for detecting cheating
- "But I didn't know that was cheating" is not a valid excuse

Academic Integrity: General

Some general guidelines

- If you don’t know if something is OK, please ask.
- If you think “I don’t want to ask, you will probably say no” that is a good sign its NOT acceptable.
- If you do something wrong, and regret it, please come forward—I recognize the value and learning benefit of admitting your mistakes. (Note: this does NOT mean there will be no consequences if you come forward).
- If you are aware of someone else’s misconduct, you should report it to me or another appropriate authority.

Course Problems

- Struggling in course
  - Come to see me/TAs: We are here to help

- Other problems:
  - Feel free to talk to the instructor, who generally understands and will try to work with you
  - Some problems may extend well beyond my course
    - Academic Advisor
    - DGS Team

Our Responsibilities

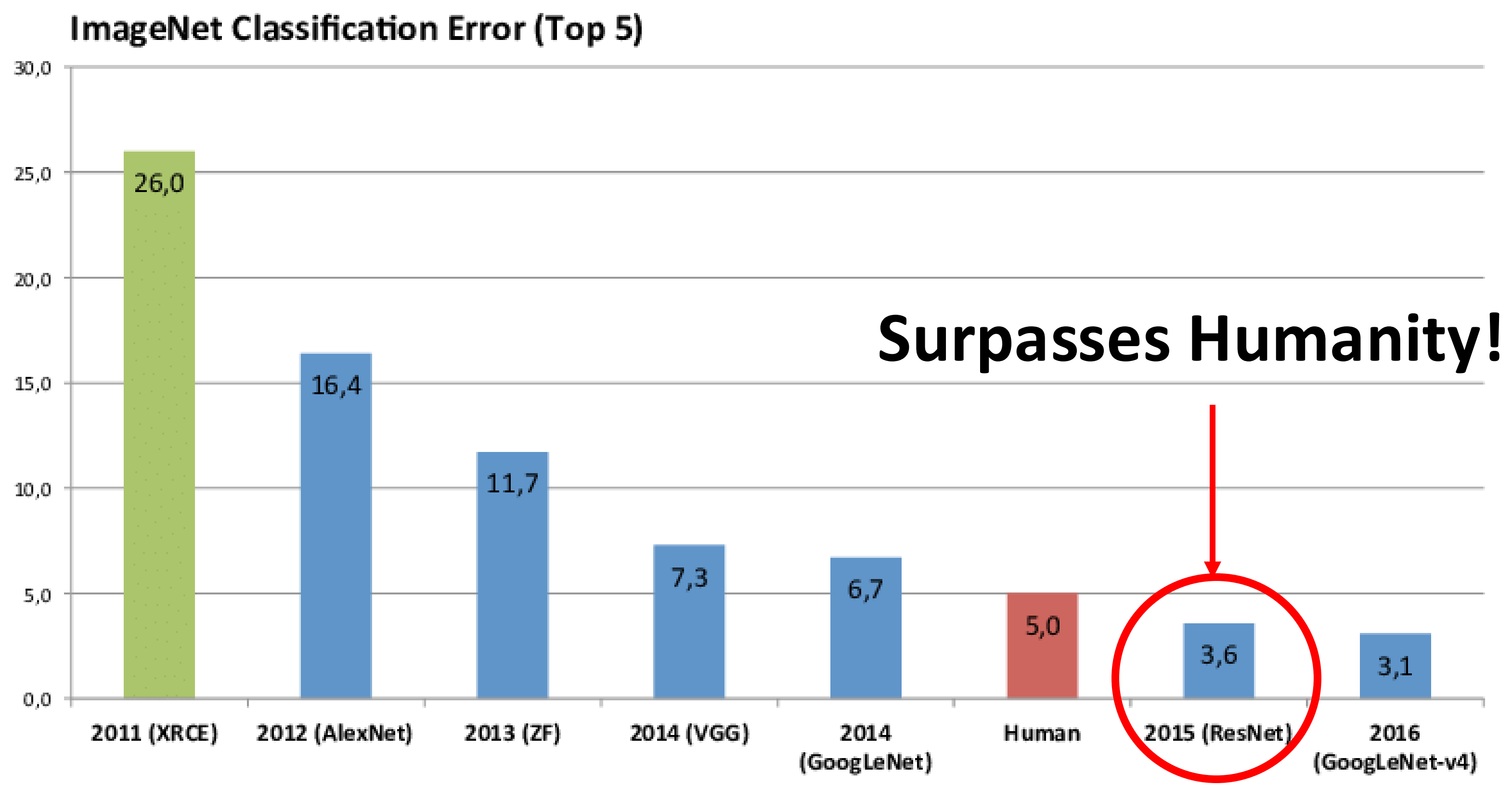
- The instructor and TAs will…
  - Provide lectures/office hours at the stated times
  - Set clear policies on grading
  - Provide timely feedback on assignments
  - Be available out of class to provide reasonable assistance
  - Respond to comments or complaints about the instruction provided

- Students are expected to…
  - Receive lectures/recitations at the stated times
  - Turn in assignments on time
  - Seek out of class assistance in a timely manner if needed
  - Provide frank comments about the instruction or grading as soon as possible if there are issues
  - Assist each other within the bounds of academic integrity

Outline

- Course Introduction
  - Refer to the course syllabus for details
- Machine Learning & Deep Neural Networks
  - Applications
  - Categories
  - Important Metrics

Applications: Images



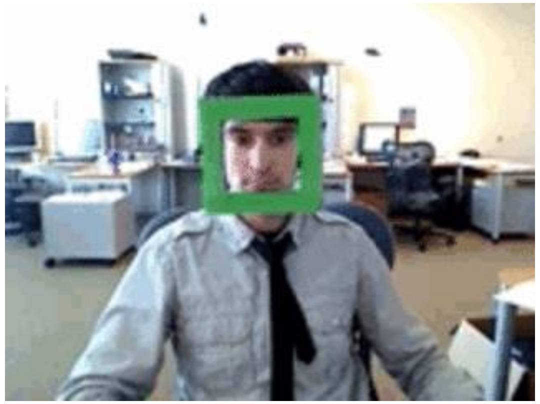
Can you tell what kind of turtle this is?

A. Dermochelys coriacea
B. Caretta caretta
C. Lepidochelys kempii
D. Lepidochelys olivacea

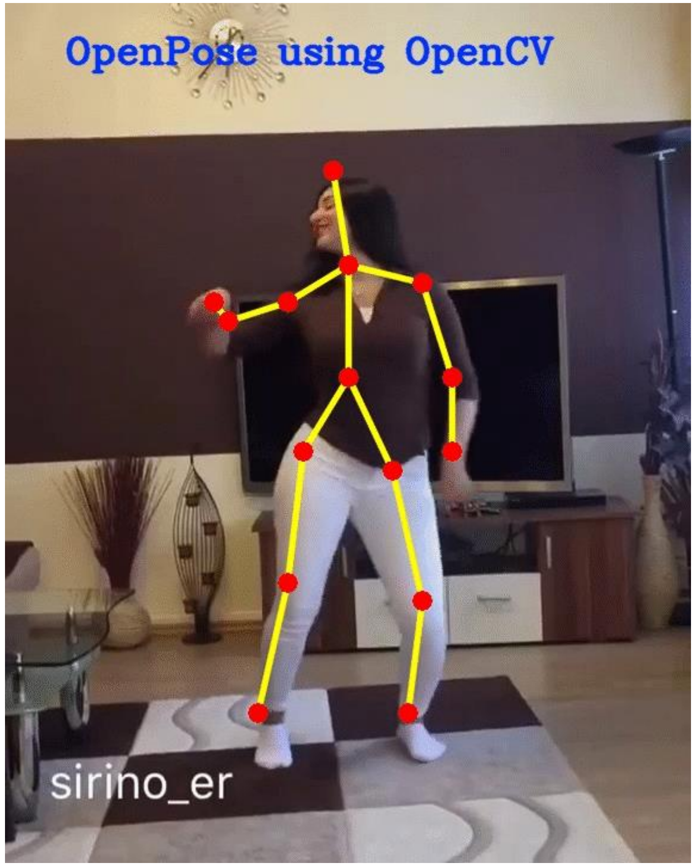
From Hung-yi Lee's introduction v8. Source of image: https://www.researchgate.net/figure/Winner-results-of-the-ImageNet-large-scale-visual-recognition-challenge-LSVRC-of-the_fig7_324476862

Applications: Videos

Object Detection



~~The Perfect Real Time Face Tracking~~

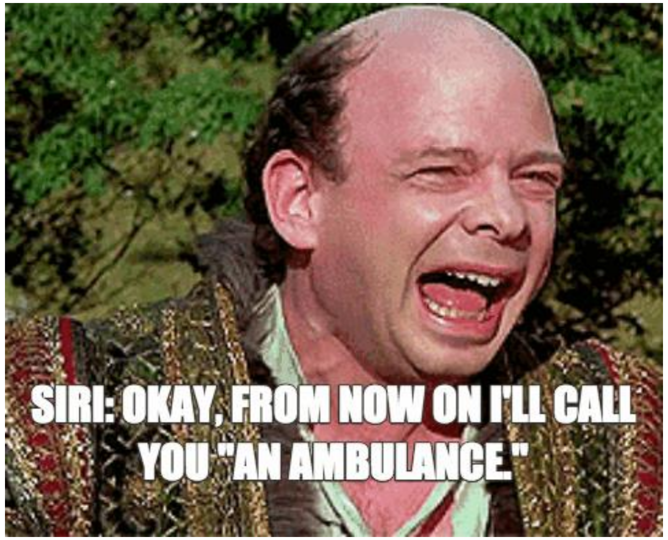


Human Pose Estimation

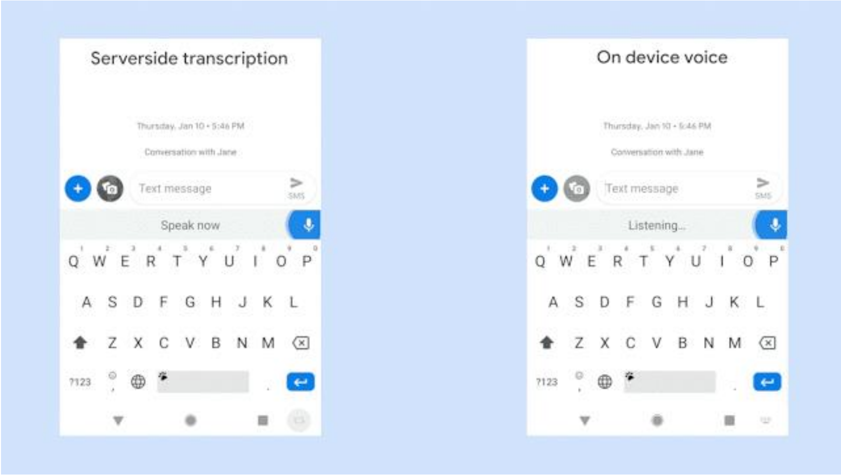
Source of Gifs: https://towardsdatascience.com/real-time-and-video-processing-object-detection-using-tensorflow-opencv-and-docker-2be1694726e5; https://www.learnopencv.com/object-tracking-using-opencv-cpp-python/; https://www.learnopencv.com/deep-learning-based-human-pose-estimation-using-opencv-cpp-python/

Applications: Speech

Cortana



"Siri, Call me an ambulance"

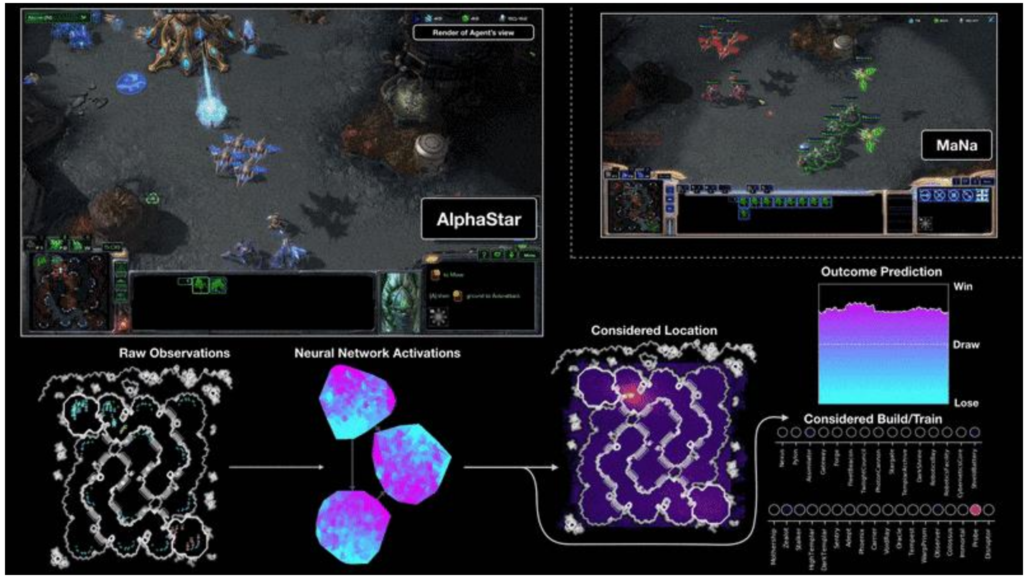


Speech To Text

"Remember when people typed with two fingers? My voice is faster."

Source of Gifs: http://www.impactlab.net/2016/10/26/microsofts-speech-recognition-is-now-as-accurate-as-a-humans/; https://www.nextbigfuture.com/2019/04/google-has-80-mb-speech-recognizer-that-can-work-offline-on-your-smartphone.html ; https://media.giphy.com/media/LYlgUG96n20h2/giphy.gif

Applications: Game; Strategy

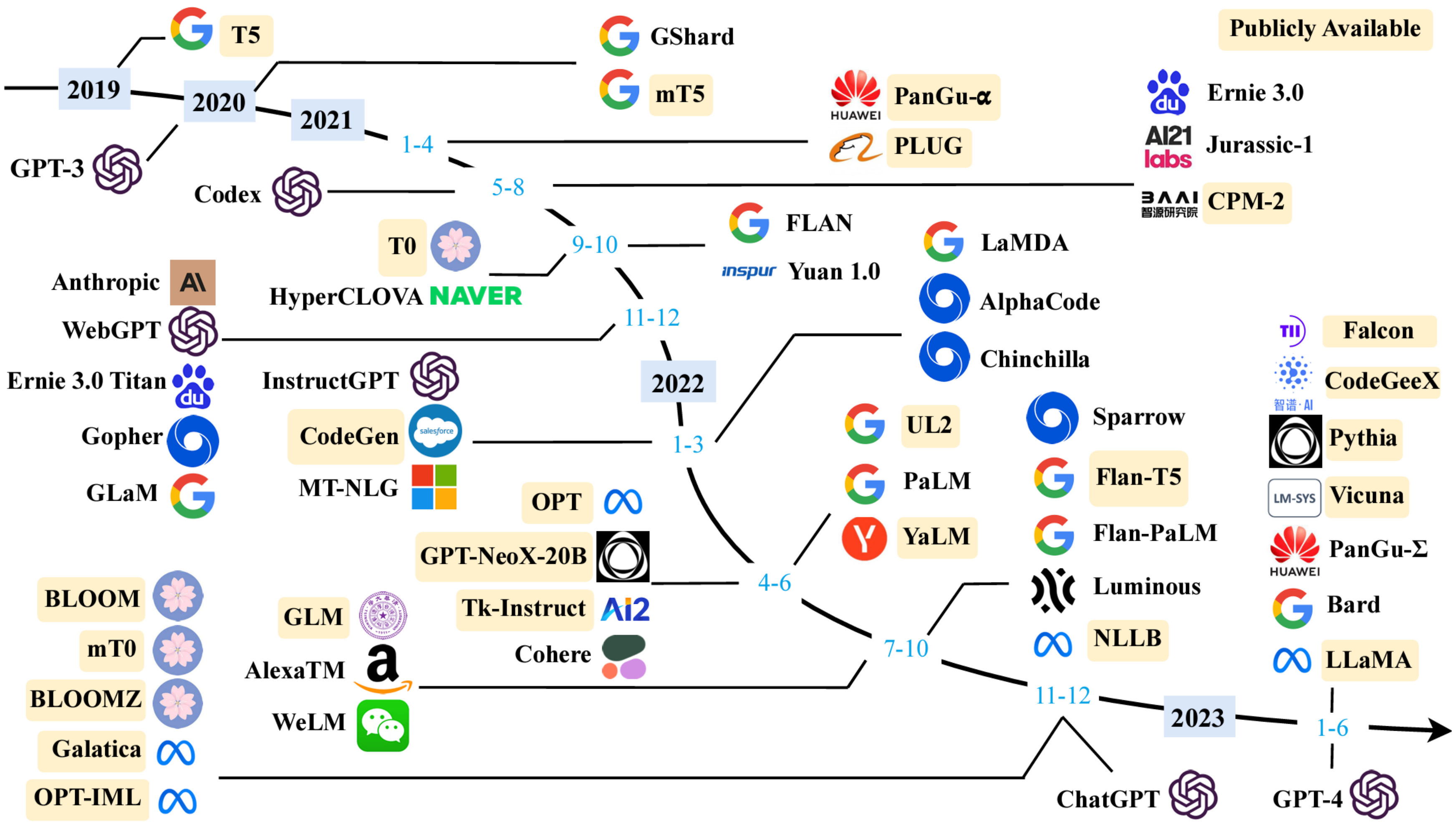


AlphaStar： StarCraft II

Alpha Go

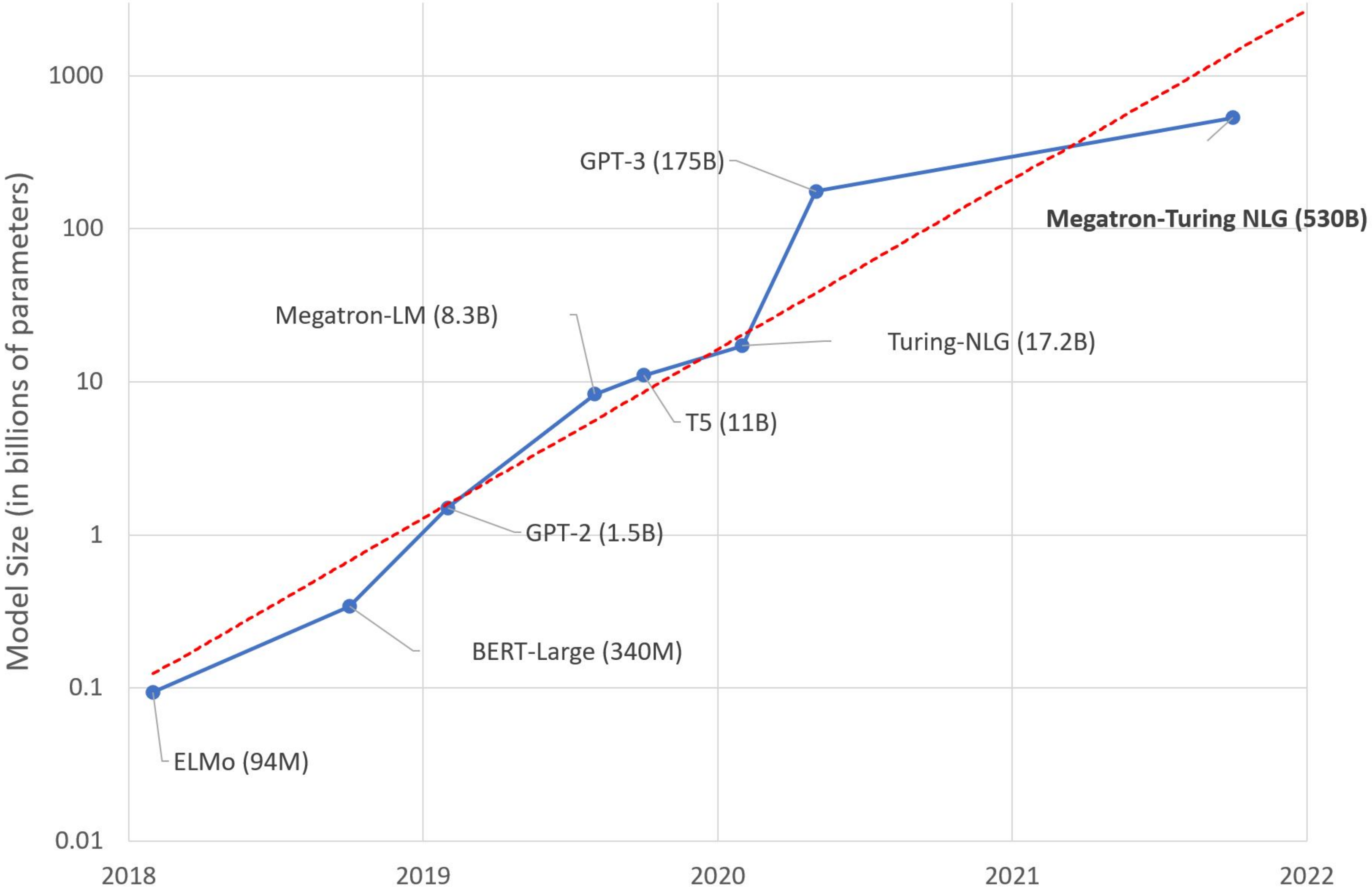
Source of Gifs: https://www.techspot.com/news/78431-human-player-finally-beat-deepmind-alphastar-ai-starcraft.html; https://imgur.com/gallery/aPw2D; Matthew Inkawhich's RT meeting Slides

Large Language Models (LLM)



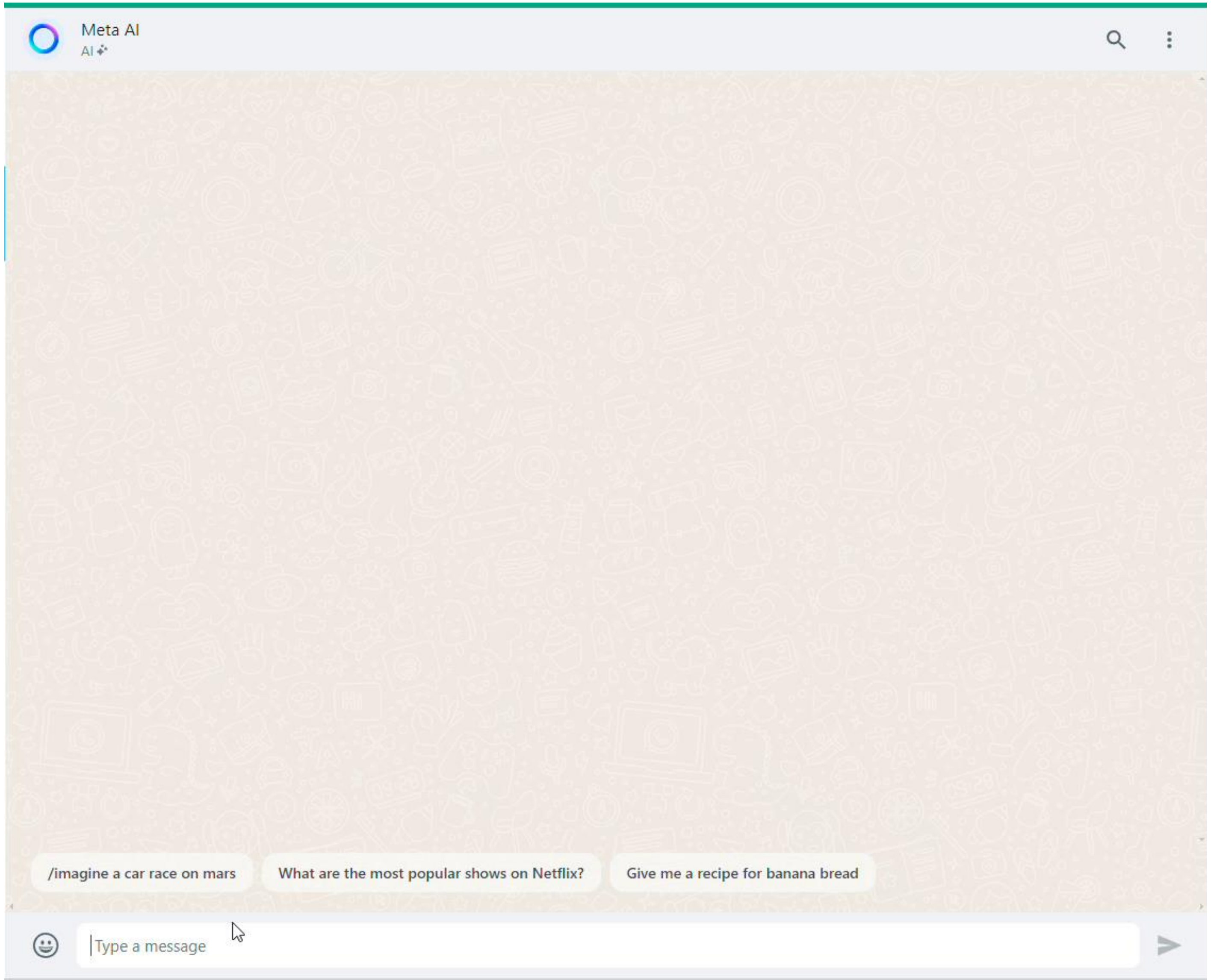
A Survey of Large Language Models, [Zhao et al., 2023]

Large Language Models : A New Moore's Law ?

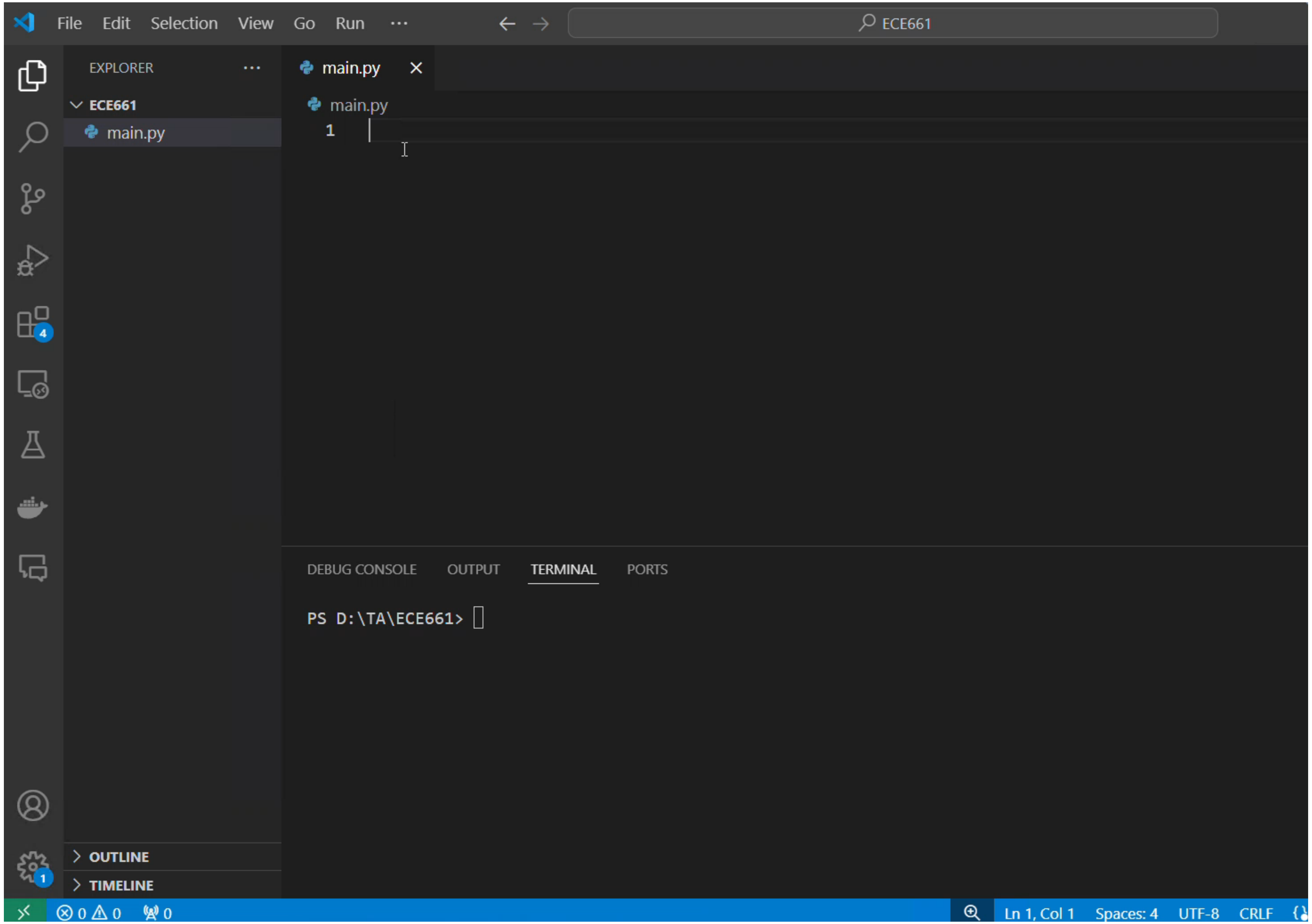


https://github.com/huggingface/blog/blob/main/large-language-models.md

Applications : Chatbot and Text Generation



Applications : Code Generation

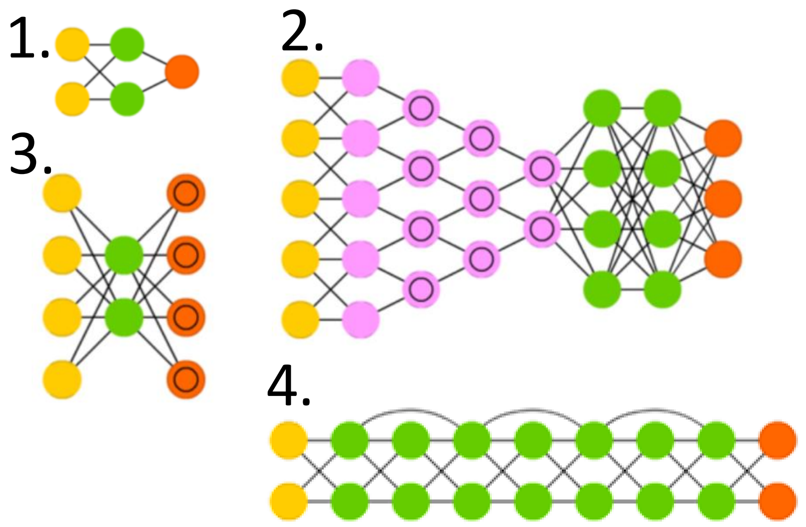


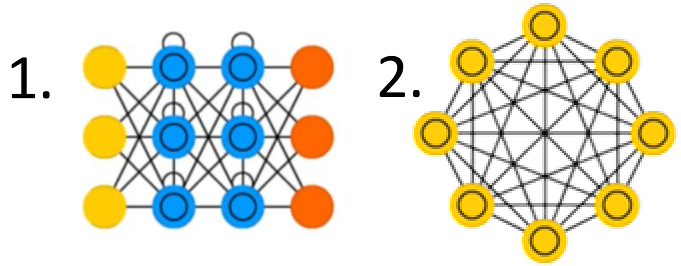
Outline

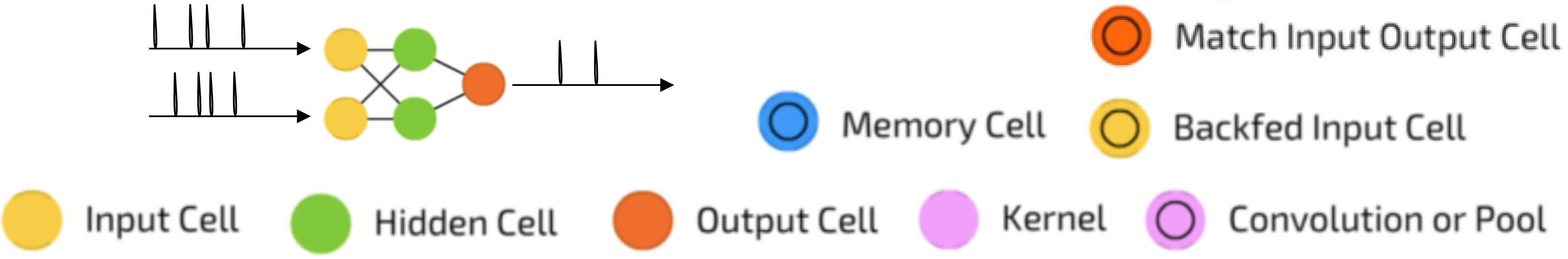
- Course Introduction
- Machine Learning & Deep Neural Networks
  - Applications
  - Categories
  - Important Metrics

Structures

- Feedforward neural network:
    1. Multilayer perceptron
    2. Convolutional neural network
    3. Autoencoder
    4. Deep residual network
- Recurrent neural network:
    1. Long short-term memory
    2. Hopfield
    3. …
- Spiking neural network







Source of image: https://towardsdatascience.com/the-mostly-complete-chart-of-neural-networks-explained-3fb6f2367464

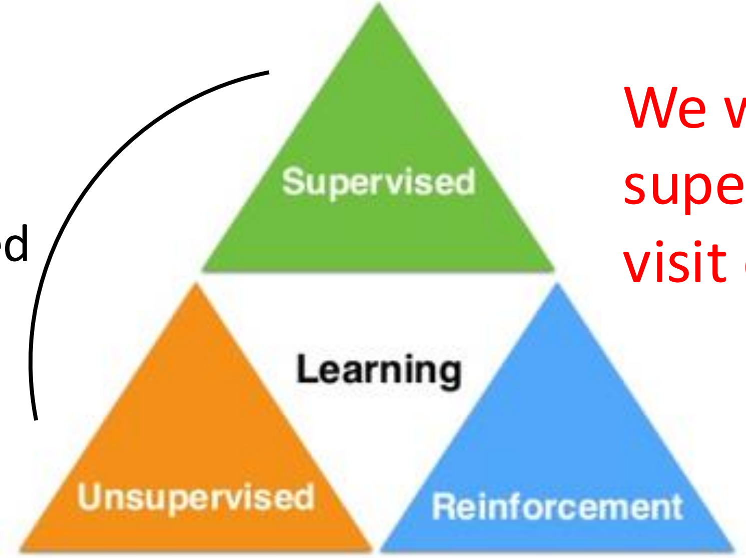
Learning Types

- Supervised
- Semi-supervised
- Unsupervised
- Reinforcement

Labeled data
Direct feedback
Predict outcome/future
Apps: Classification

Semi-supervised

Weakly-supervised



We will start with supervised learning and visit other topics later

No labels
No feedback
Find hidden representations
Apps: Reconstruction

Decision process
Reward system
Learn series of actions
Apps: Decision-making

Outline

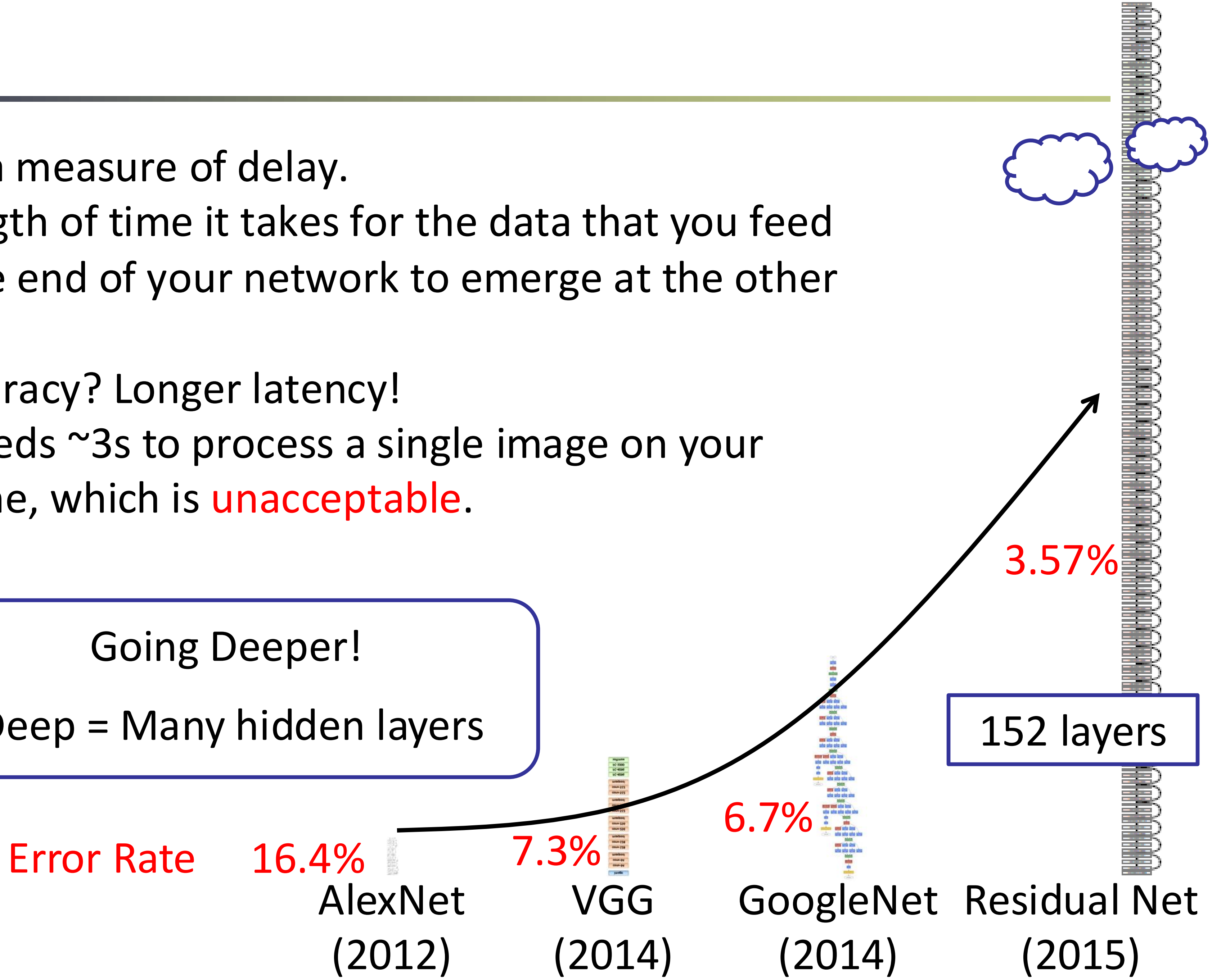
- Course introduction
- Machine Learning & Deep Neural Networks
  - Applications
  - Categories
  - Important Metrics (**LASER**)
    - **L**atency
    - **A**ccuracy
    - **S**ize of Model
    - **E**nergy Efficiency
    - **R**obustness

Latency

- Latency is a measure of delay.
  - The length of time it takes for the data that you feed into one end of your network to emerge at the other end.
- Better accuracy? Longer latency!
- VGG-16 needs ~3s to process a single image on your smart phone, which is unacceptable.

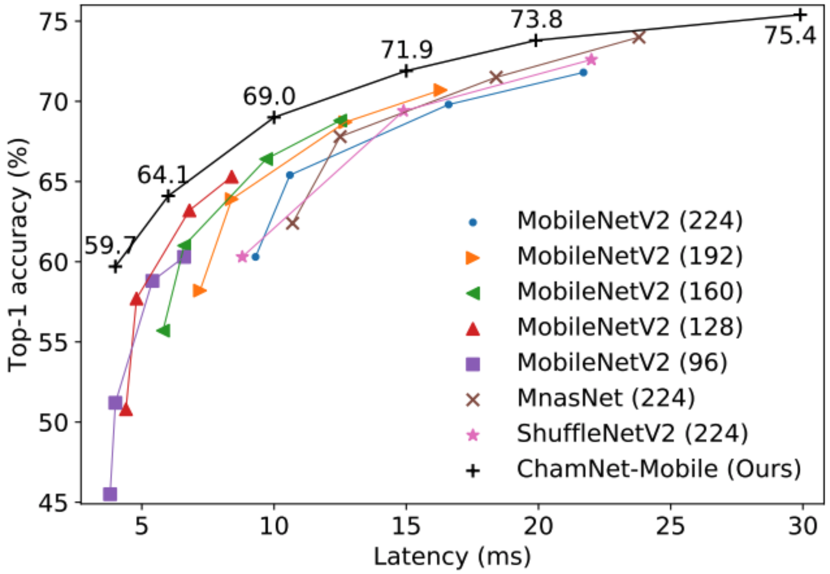
Going Deeper!

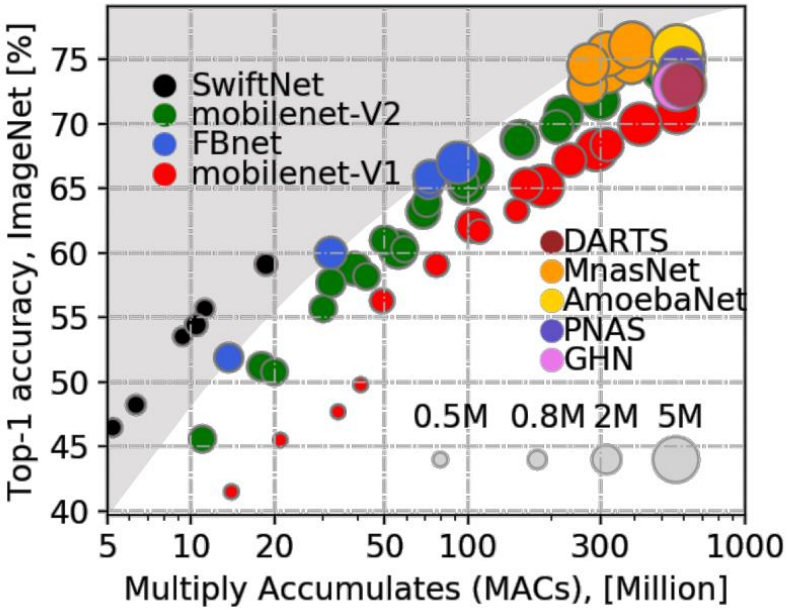
Deep = Many hidden layers



Accuracy

- Accuracy is a metric for classification problem
- We call it: "Top-*K* Accuracy"
- Higher accuracy is good, but we need to pay for it
  - Everything is a trade-off.





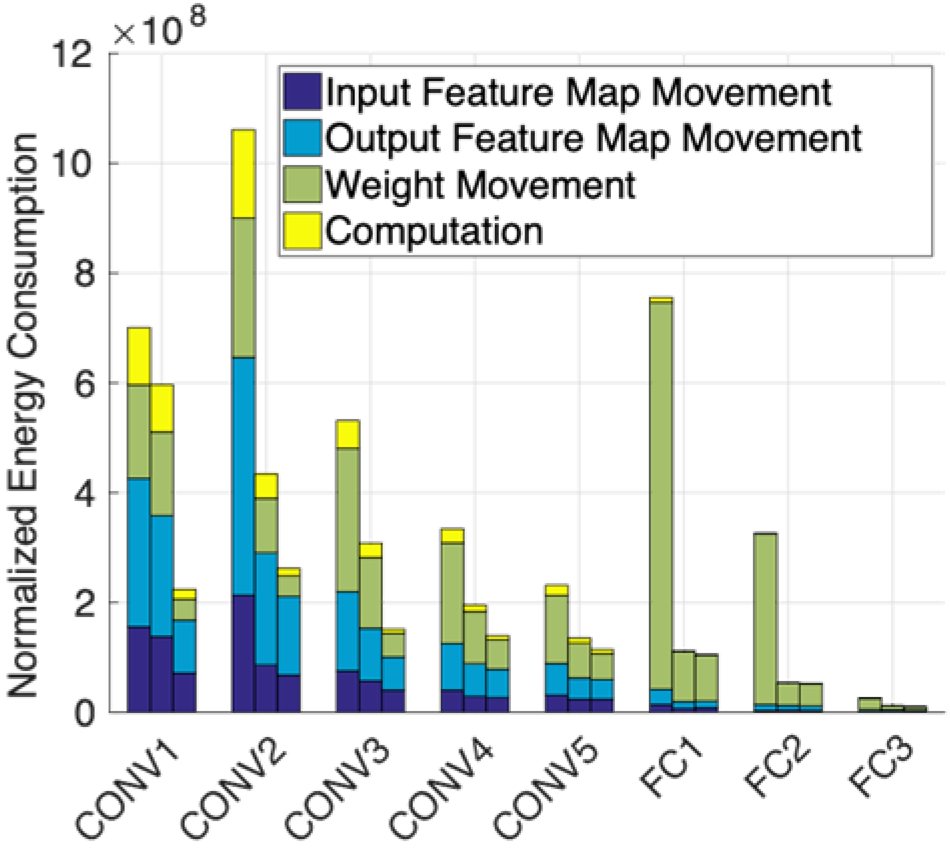
Source:
1. Dai, Xiaoliang, et al. "Chamnet: Towards efficient network design through platform-aware model adaptation." (2019)
2. Cheng, Hsin-Pai et al. "SwiftNet: Using Graph Propagation as Meta-knowledge to Search Highly Representative Neural Architectures" (2019)

Size of Model

- # FLOP: Number of floating point operations.
- # MAC：Number of multiply-and-accumulate operations
  – Usually, 1 floating-point multiply-and-accumulate is considered equivalent to 2 FLOPs.
- # Parameters
- Area [mm^2]

Energy Efficiency

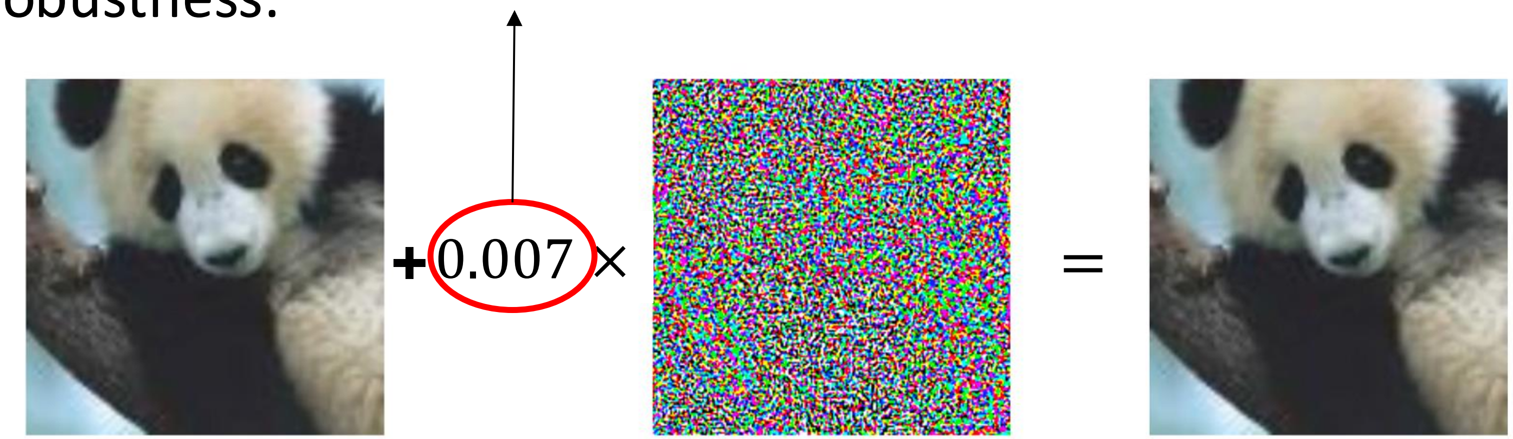
- Power consumption [mW]
- Energy is mainly used for
  - Calculation
  - Data movement
- Energy is a different thing:
  - A lower number of MACs does not necessarily lead to lower energy consumption.
  - Convolutional layers consume more energy than fully-connected layers.
  - Deeper CNNs with fewer weights do not necessarily consume less energy than shallower CNNs with more weights.



Source of image: http://eyeriss.mit.edu/

Robustness

- This parameter, is used to evaluate a neural network's robustness.



Panda noise gibbon

57.7% confidence 99.3% confidence

- Usually, a high accuracy model is not robust.
- Compared to the size of a neural network, the structure has more impact towards robustness.

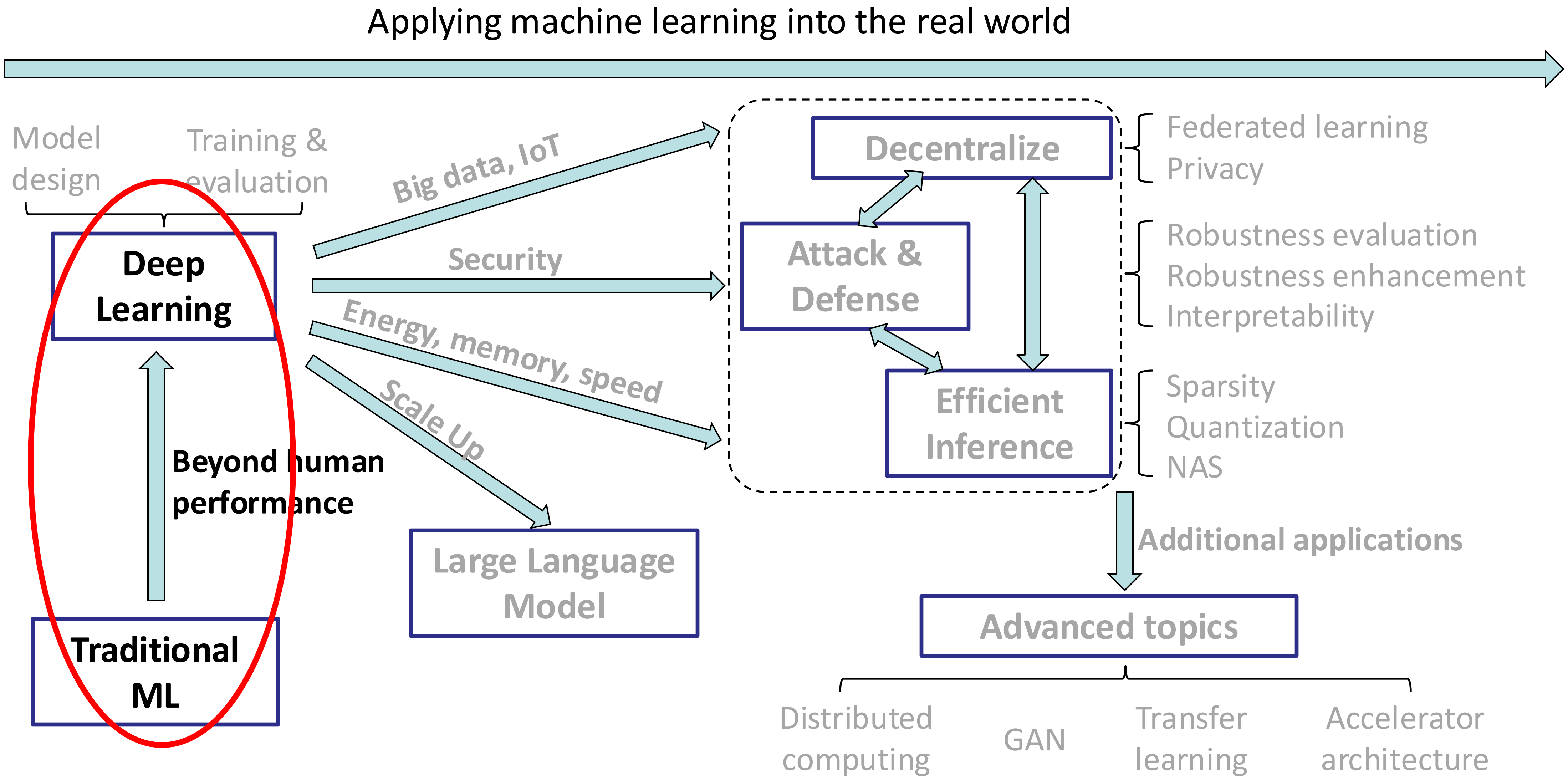
Everything is a trade-off

Source of image: Explaining and harnessing adversarial examples

Future

Smart **Low latency** **Privacy** **Mobility** **Energy efficient**

Next Lecture



Reading Material

- Deep Learning (2016), Ian Goodfellow and Yoshua Bengio and Aaron Courville http://www.deeplearningbook.org/
  - Chapter “Introduction”