DataSci 420

lesson 9: deep learning (part 1)

Seth Mottaghinejad

today's agenda

- AI vs ML vs DL
- what is **deep learning**?
- the advent of deep learning
- deep learning vs **traditional ML**
- deep learning **applications**
- tensors and deep learning **frameworks**

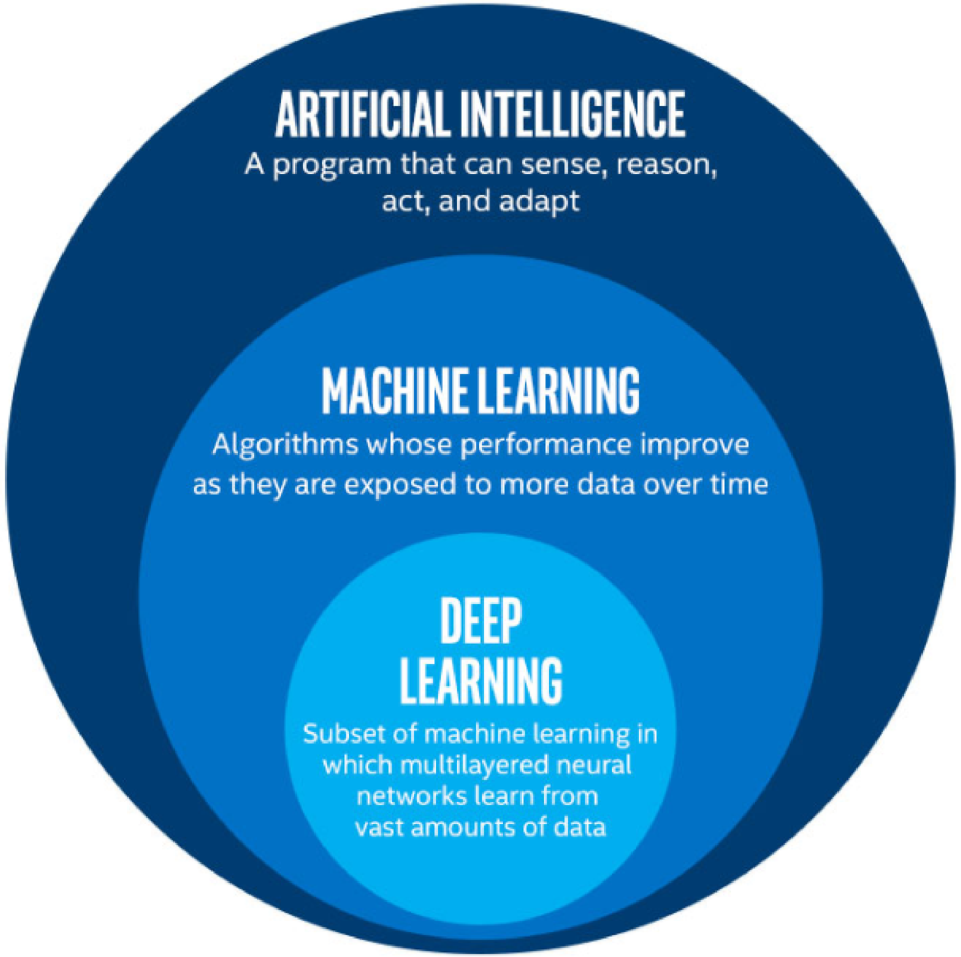
ARTIFICIAL INTELLIGENCE
A program that can sense, reason, act, and adapt
MACHINE LEARNING
Algorithms whose performance improve as they are exposed to more data over time
DEEP LEARNING
Subset of machine learning in which multilayered neural networks learn from vast amounts of data

image source: towardsdatascience.com

AI vs ML vs DL for making guacamole

- you can hire experts to help you write a program that spits out a guac recipe based on available ingredients and quantities
- you can use ML to look at a data set of ingredients and their quantities and how the resulting guac was rated
- you can do **feature engineering** prior to ML and engineer features that make sense (to some extent based on expert know-how): e.g. ratios of particular ingredients
- you can use DL which does ML with **feature engineering built-in** (no experts needed)

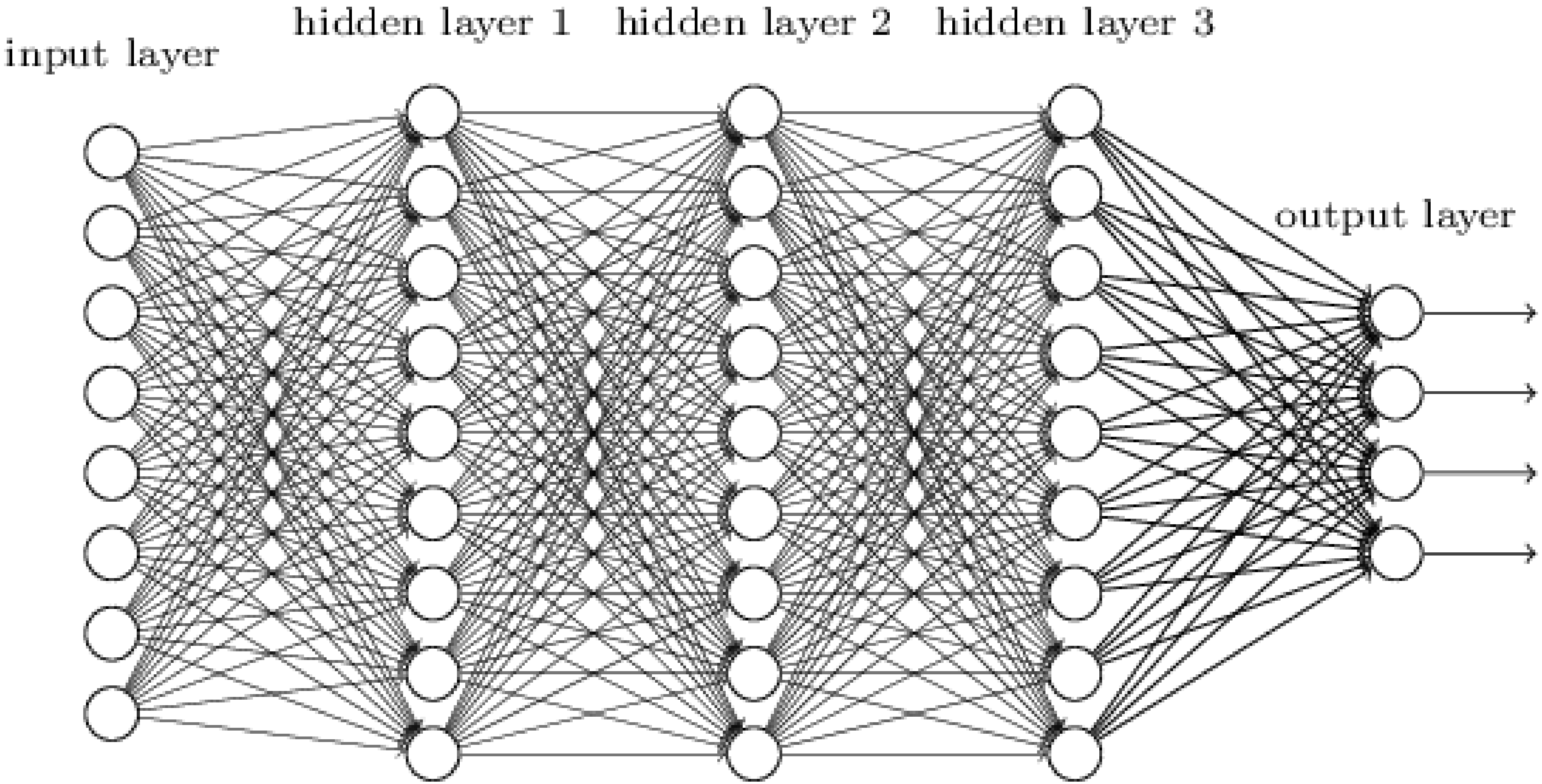

image source: neuralnetworksanddeeplearning.com

ML / ML with FE / ML with automated FE

salt	· · ·	avoc	onions	rating
2 tbs	· · ·	1 lbs	3 oz	2/5

salt	· · ·	avoc	onions	salt/pepp	· · ·	avoc/onions	rating
2 tbs	· · ·	1 lbs	3 oz	0.50	· · ·	1.245	2/5

salt	· · ·	avoc	onions	HL1N1	· · · · · · · · · ·	H3N10	rating
2 tbs	· · ·	1 lbs	3 oz	0.582	· · · · · · · · · ·	1.253	2/5

what makes learning deep?

- **neural networks** are a type of machine learning algorithms
- deep learning refers to using neural networks with **many many hidden layers** to solve problems where
- **feature engineering** is built into DL and hidden layers make it possible to engineer **increasingly abstract features**
- most ML concepts still apply to DL algos, but DL also has many concepts that are unique to it
- DL methods can be used in supervised, unsupervised, and reinforcement learning

advent of deep learning

- **deep networks** are neural networks with **lots of hidden layers**
- not a new idea, but it only recently became **computationally feasible**: better hardware and advances in optimization
    - **GPUs** (graphical processing units) are made to do **array computations** efficiently, big deal in rendering graphics but made its use to deep learning
    - **TPUs** (tensor processing units) are the next gen hardware made specifically for deep learning
- DL models also need a **lot of data**, which we now have

deep learning vs traditional ML

- traditional ML has a more limited number of parameters
  - for most applications less than 1000
  - for some applications with wide data sets 100K to 1M
- deep learning model can have **millions** of parameters
- more **parameters** → more **complex model** → prone to **over-fitting** and hence
  - **more data** needed to compensate and
  - **more fine-tuning** of hyper-parameters needed

deep learning applications

- **auto-encoders** / **encoder-decoder** (unsupervised learning)
  - dimensionality reduction, denoise images, remove watermarks
- **convolutional networks** for image segmentation
  - image localization and classification, description
- **recurrent networks** and **attention networks** for NLP
  - word / sentence completion, translation, entity extraction
- **deep reinforcement learning**
  - playing games, self-driving cars, kill all humans!

deep learning frameworks

- high-level and low-level library for training NNs
  - no need to build network from scratch
  - **auto-differentiation** saves us from figuring out the math
  - focus on the **architecture** (number of hidden layers, hidden units, activation functions, connections)
- CPU vs GPU vs TPU (or FPGA) - switch the context without having to change code

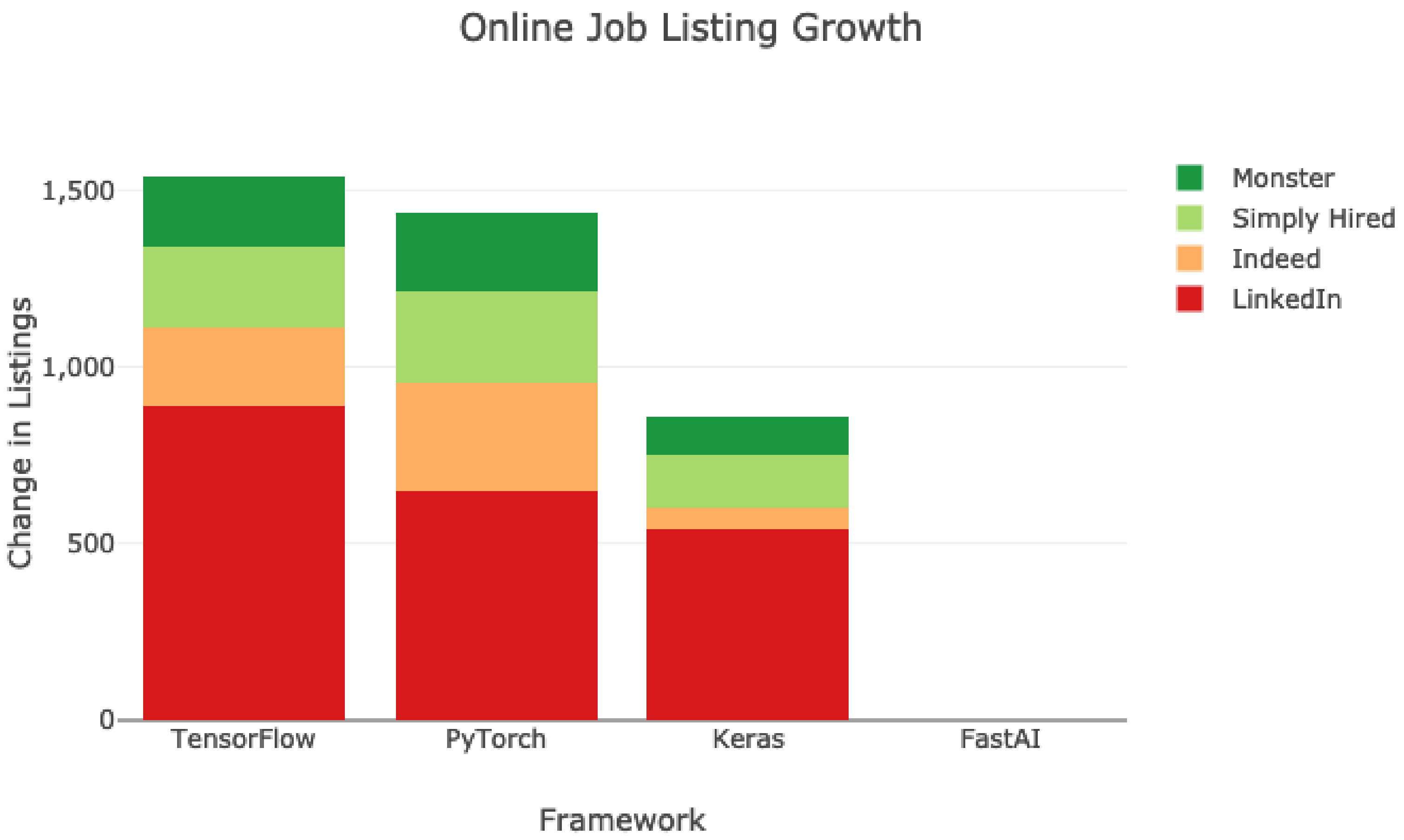

image source: kdnuggets.com

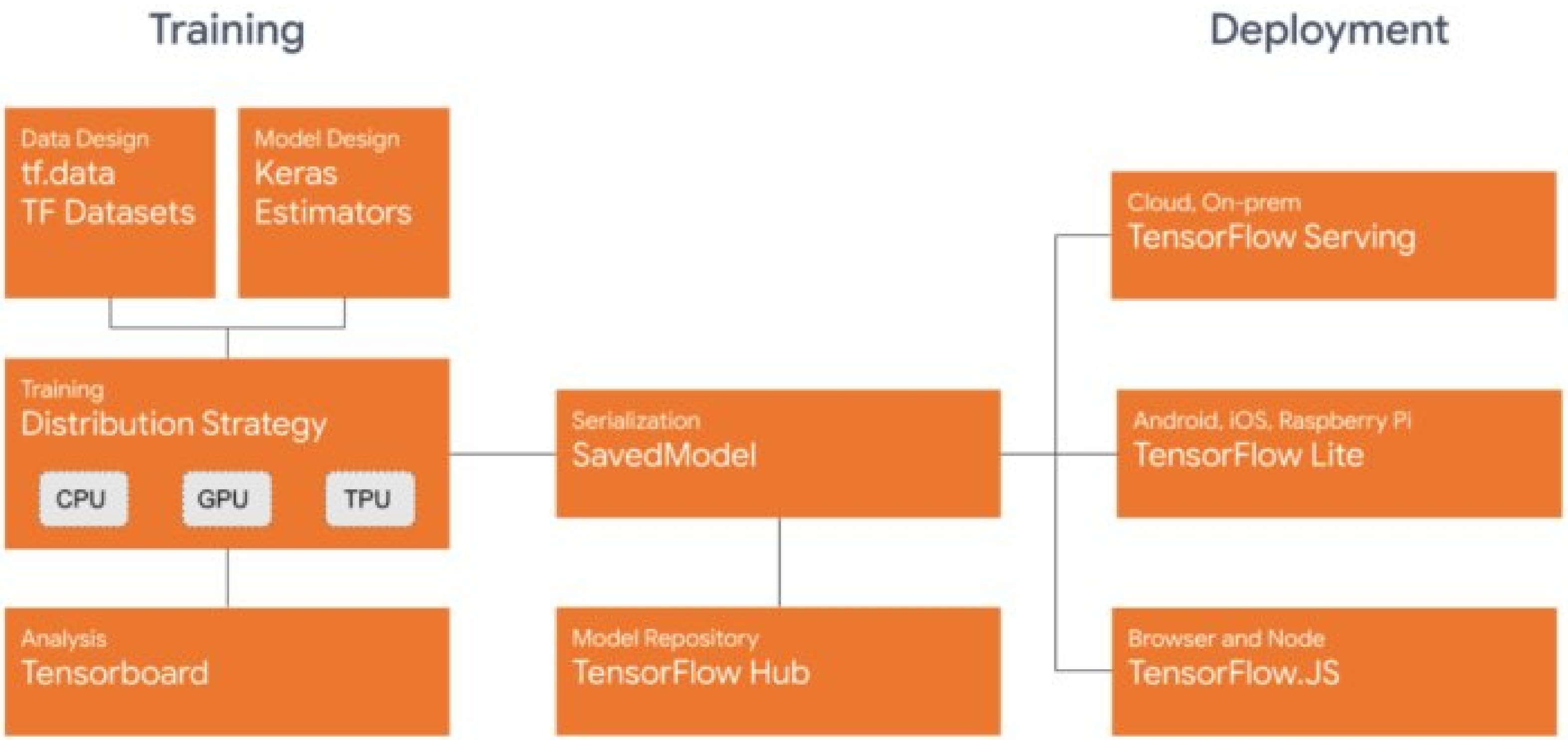

image source: kdnuggets.com

the end