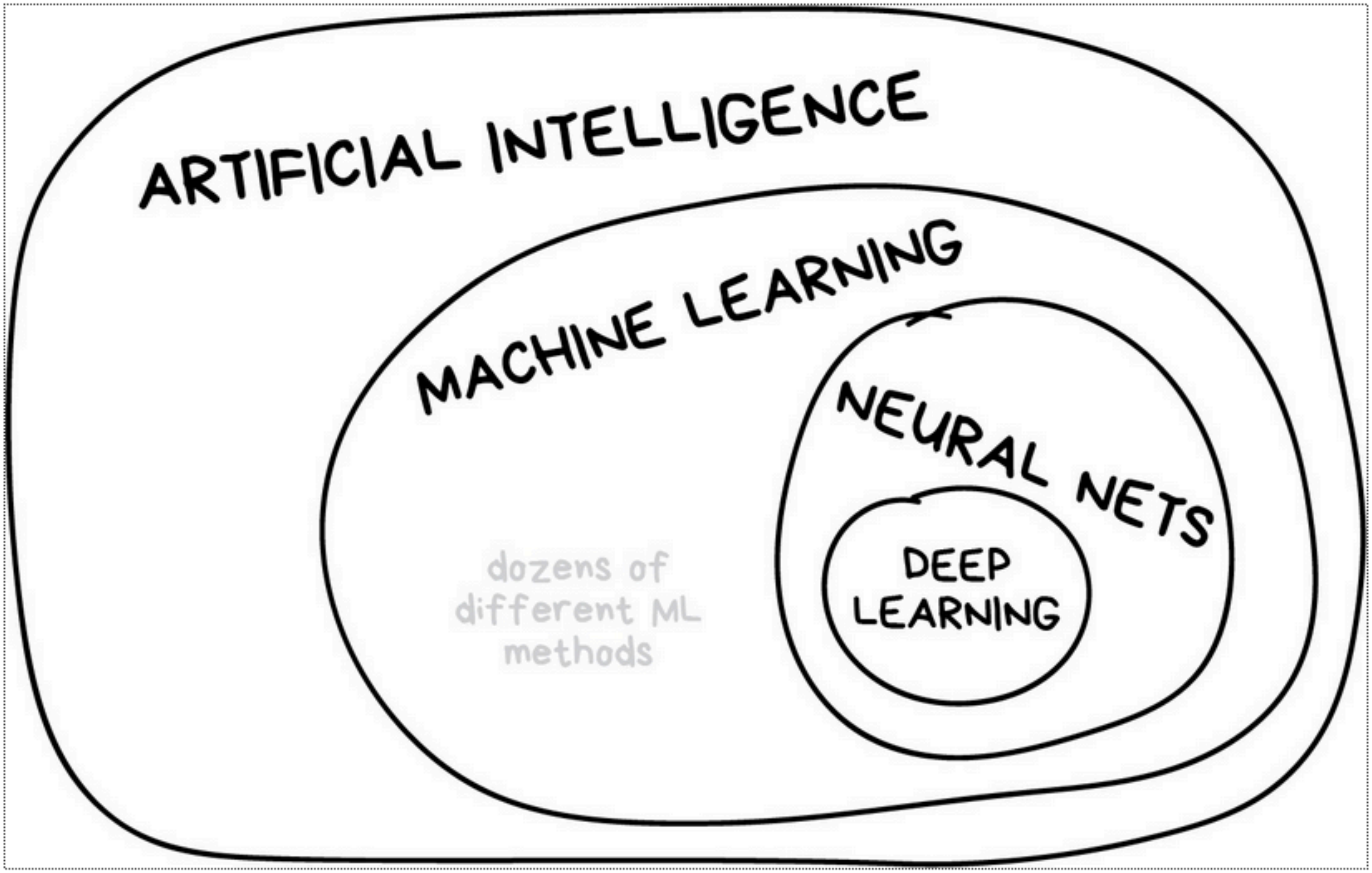
Lecture 23: Artificial intelligence, machine learning, natural language processing, ...

- buzzwords, hype, real accomplishments, wishful thinking
  - big data, deep learning, neural networks, ...
- brief history
- examples
  - games (chess, Go)
  - classification (spam detection)
  - prediction (future prices)
  - recommendation systems (Netflix, Amazon, Goodreads, ...)
  - natural language processing (sentiment analysis, translation, generation)
  - large language models
- issues and concerns
  - accuracy
  - fairness, bias, accountability, explainability
  - appropriate uses
- Beware: on this topic, I am even less of an expert than normal.

Revisionist history of AI (non-expert perspective)

- 1950s, 1960s: naive optimism about artificial intelligence
  - checkers, chess, machine translation, theorem proving, speech recognition, image recognition, vision, ...
  - almost everything proved to be much harder than was thought
- 1980s, 1990s: expert or rule-based systems
  - domain experts createrules, computers apply them to make decisions
  - it's too hard to collect the rules, and there are too many exceptions
  - doesn't scale to large datasets or new problem domains
- 2010s: machine learning, big data, deep learning, ...
  - provide a "training set" with lots of examples correctly characterized
  - define "features" that might be relevant, or let the program find them itself
  - write a program that "learns" from its successes and failures on the training data (basically by figuring out how to combine feature values)
- 2020s: large language models
  - ChatGPT-3, GPT-4, DALL-E2, ...
  - near-human performance on many text understanding and generation tasks

The big picture (vas3k.com/blog/machine_learning)



Examples of ML applications (a small subset)

- games
  - checkers, chess, Go
- classification
  - spam detection, digit recognition, optical character recognition, authorship, ...
  - image recognition, face recognition, ...
- prediction
  - house prices, stock prices, credit scoring, resume screening, ...
  - tumor probabilities, intensive care outcomes, ...
- recommendation systems
  - e.g., Netflix, Amazon, Goodreads, ...
- natural language processing (NLP)
  - language translation
  - text to speech; speech to text
  - sentiment analysis
  - text generation (ChatGPT et al)
  - image generation (Dall-E2, Stable Diffusion, etc)

Types of learning algorithms

- supervised learning (labeled data)
  - teach the computer how to do something with training examples
  - then let it use its new-found knowledge to do it on new examples

- unsupervised learning (unlabeled data)
  - let the computer learn how to do something without training data
  - use this to find structure and patterns in data

- reinforcement learning
  - some kind of "real world" system to interact with
  - feedback on success or failure guides / teaches future behavior

- recommender systems
  - look for similarities in likes and dislikes / behaviors / ...
  - use that to predict future likes / behaviors

Classification example: spam detection

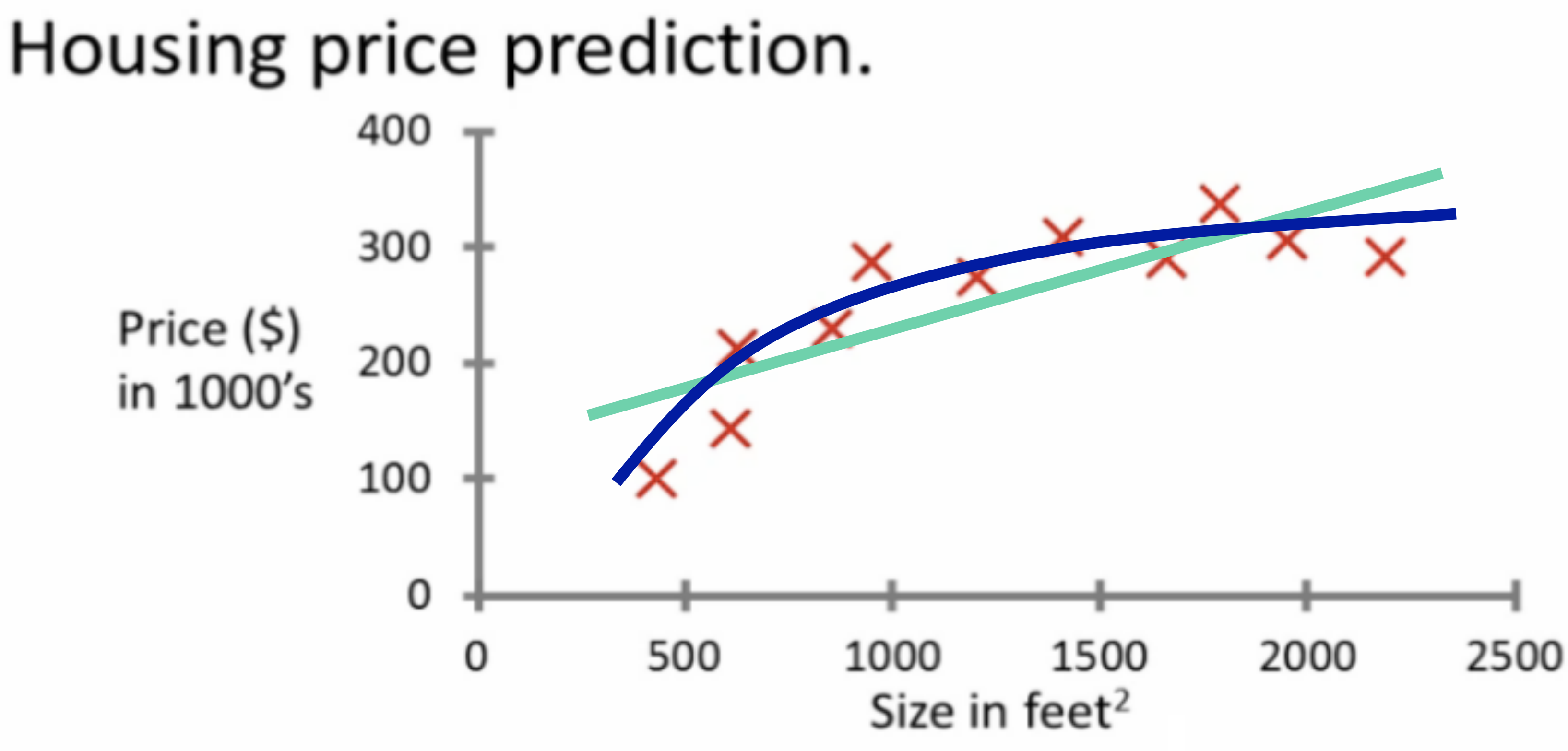
- rule-based machine learning: choose a set of features like
  - odd spelling, weird characters, language and grammar, origin, length, ...
- provide a training set of messages marked as "spam" or "not spam"

- ML algorithm figures out parameter settings that let it do the best job of separating spam from not spam in the training set
- then apply that to real data

- potential problems:
  - training set isn't good enough or big enough
  - creating it is probably done manually
  - "over-fitting": does a great job on training set but little else
  - spammers keep adapting so we always need new training material

Prediction example: house prices

- only one feature here: square footage
- straight line? ("linear regression")
- some kind of curve?



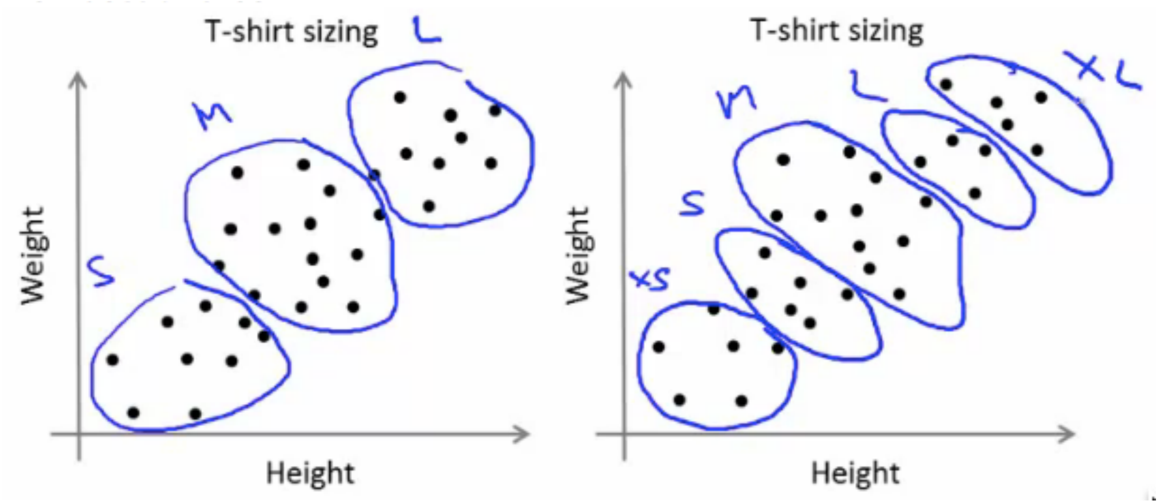
Clustering: learning from unlabeled data

- contrast with supervised learning
  - supervised learning:

    given a set of labels, fit a hypothesis to it
  - unsupervised learning:

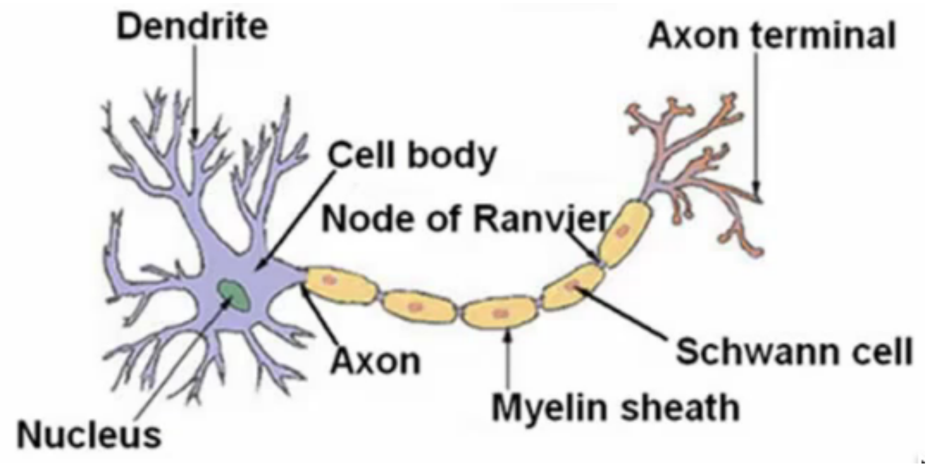
    try and determine structure in the data

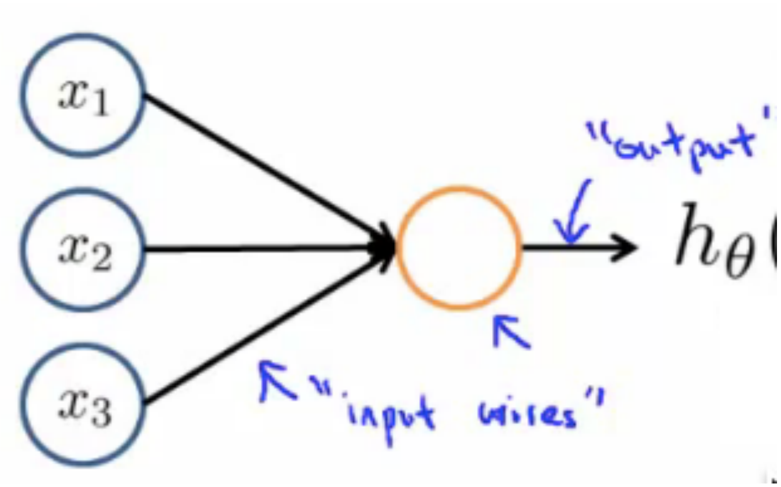
    clustering algorithm groups data together based on data features
- clustering is good for
  - market segmentation – group customers into different market segments
  - social network analysis – identify friend groups
  - topic analysis
  - authorship

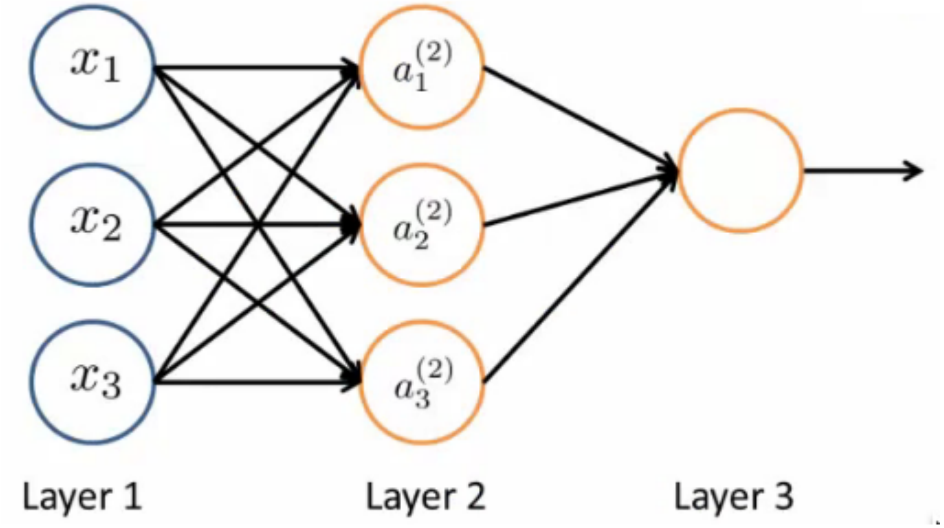


Neural networks, deep learning

- simulate human brain structure
  with artificial neurons
  in simple connection patterns







Large Language Models (LLM)

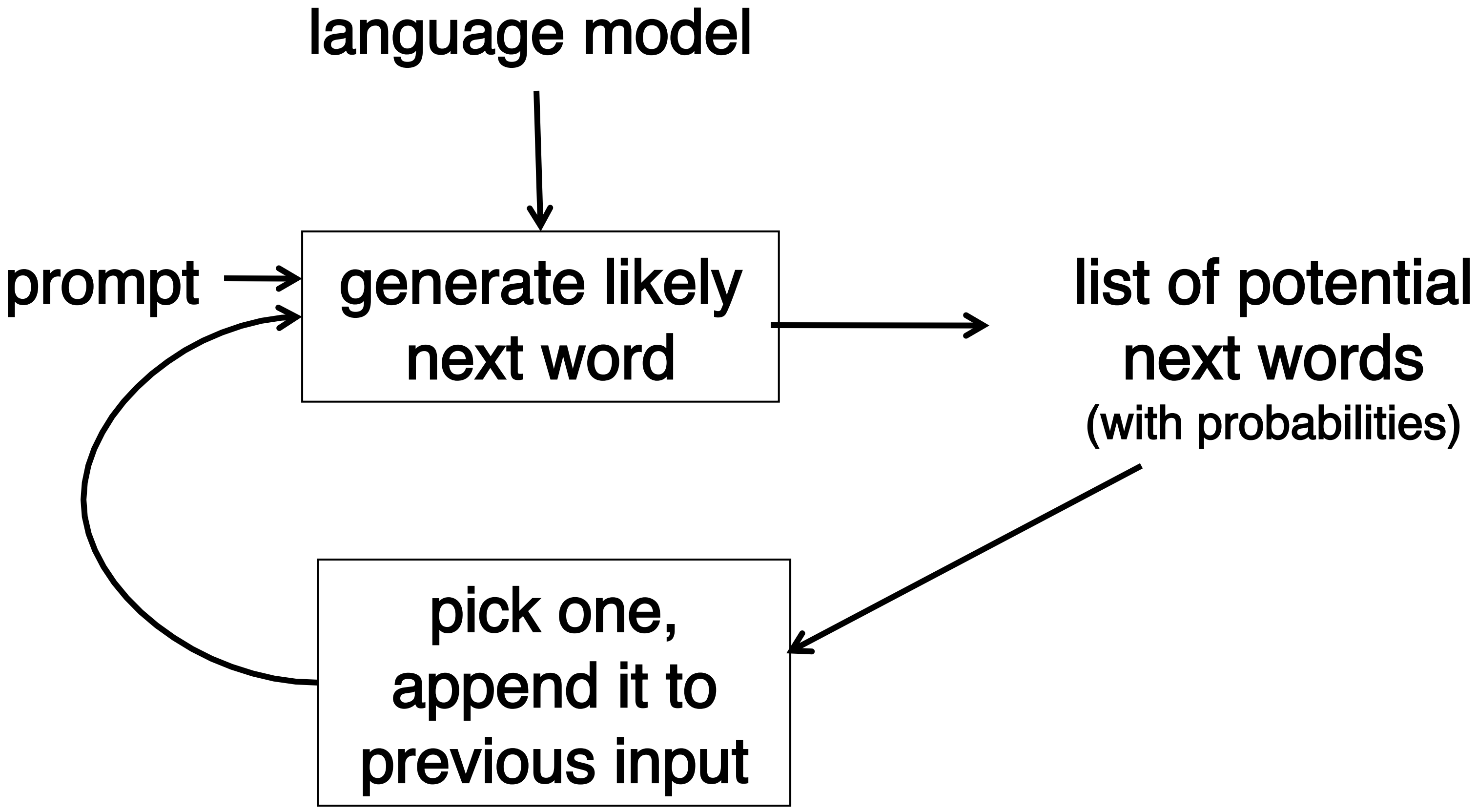
- **language models based on very large text corpus**
  - use deep learning to learn how language is used
  - use that to generate text that seems human-written
  - and give the (strong) impression of understanding

- **models are proprietary (mostly)**
  - e.g., GPT-3, -4 licensed by Microsoft from OpenAI
  - in part because they cost a *lot* to create, plus competitive value

- **GPT = generative pre-trained transformer**
  - transformer is a particular architecture for training

- **ChatGPT is based on GPT-3 (chat.openai.com)**
  - tuned for conversational style
  - can remember previous parts of a conversation
  - very new: became available Nov 30, 2022
  - has already revolutionized the field and public perception of AI

How LLMs work (layman's view)



language model

prompt

generate likely next word

list of potential next words (with probabilities)

pick one, append it to previous input

ML / AI issues (very incomplete list)

- algorithmic fairness
  - results can't be better than training data
  - if that has implicit or explicit biases, results are biased
  - can we detect and eliminate bias?
- accountability and explainability
  - what is the algorithm really doing?
  - can its results be explained
- appropriate uses? (lots of inappropriate uses!)
  - prison sentencing
  - drone strikes
  - weapon systems
  - resume evaluation
  - medical decisions
  - ...
- to learn more:

  https://fairmlbook.org