Introduction to Deep Learning

Sargur N. Srihari
srihari@cedar.buffalo.edu

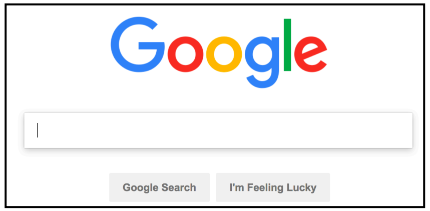
This is part of lecture slides on Deep Learning:
http://www.cedar.buffalo.edu/~srihari/CSE676

Topics

- Artificial Intelligence Paradigms
  - Knowledge-based, Simple ML, Deep Learning
- Origins of Deep Learning
- Representation Learning
- Comparison of approaches to AI
- Software 1.0 vs Software 2.0

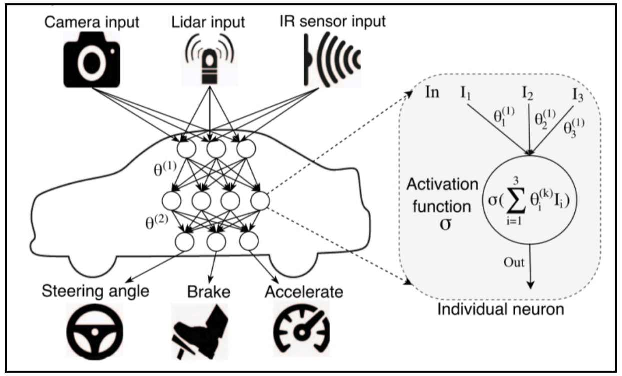
Today AI is ubiquitous

- Automate routine labor
  - Search



- Understand speech
  - SIRI, Alexa

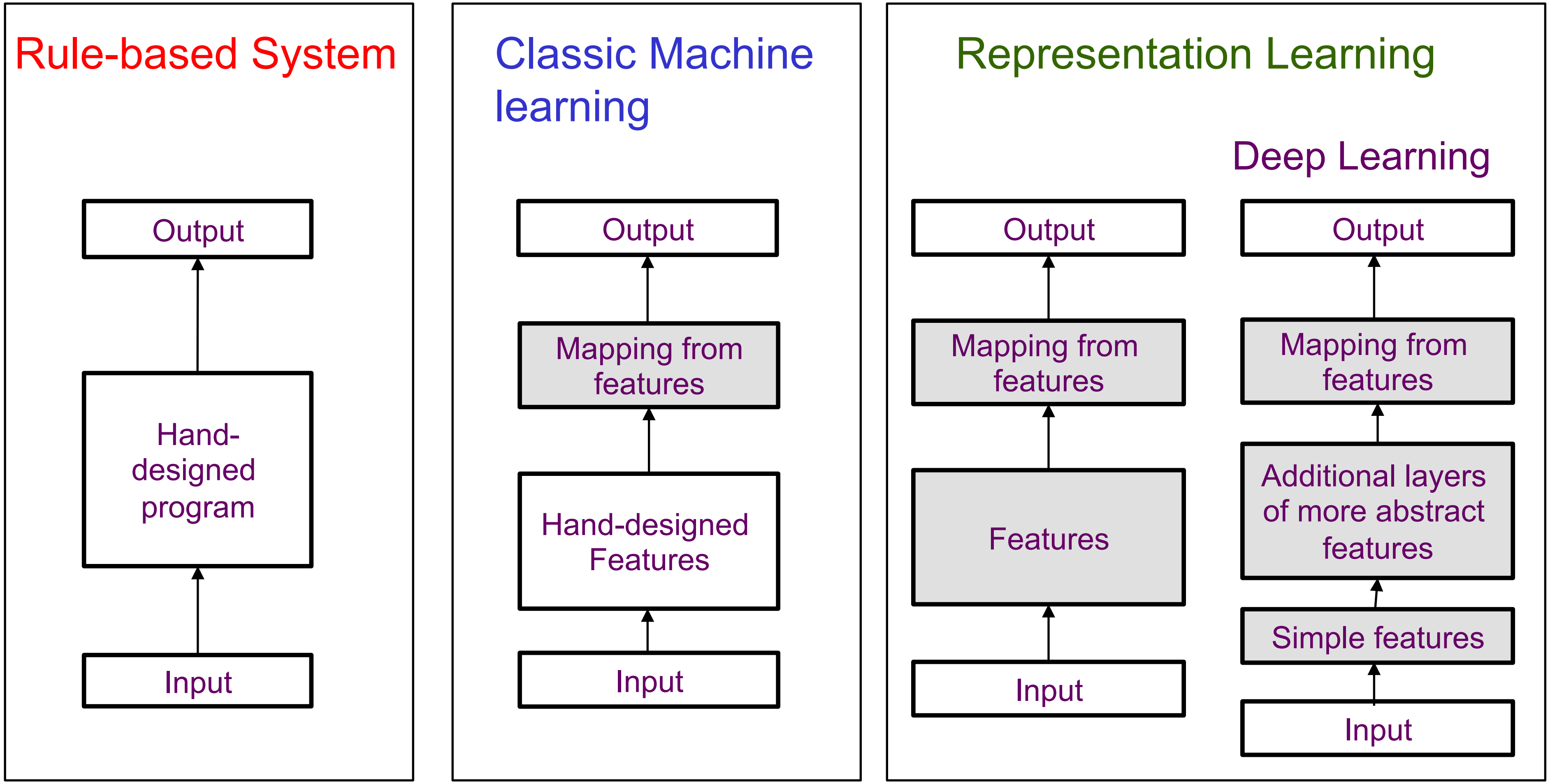
- Autonomous Vehicles



Challenge of AI

- Early successes of AI:
  - Solved problems intellectually difficult for humans
    - problems described by small set of rules
      - Sterile formal environment
      - Little knowledge about the world
- True challenge of AI
  - Solve tasks easy for people but hard to describe formally
    - solved intuitively, that feel automatic
      - e,g recognize spoken words, or faces in images
- Today's AI:
  - It is about solving these more intuitive problems

Summary of AI Models



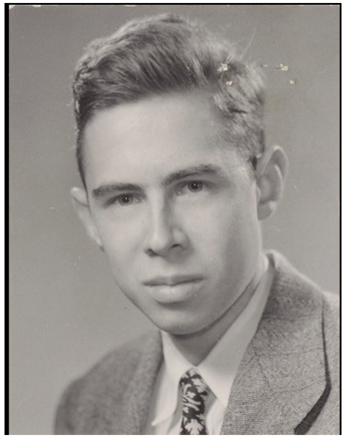
Shaded boxes indicate components that can learn from data

AI Paradigm Shift

- Physics paradigm shift
  - Newtonian Physics
    - Cannot explain black-body radiation
  - Quantum Mechanics
- AI paradigm shift
  - Knowledge-based systems
    - Cannot perform simple recognition tasks
  - Simple machine learning methods
    - Cannot perform complex recognition tasks
  - Deep Learning methods

Srihari

Beginnings of Deep Learning



Perceptron

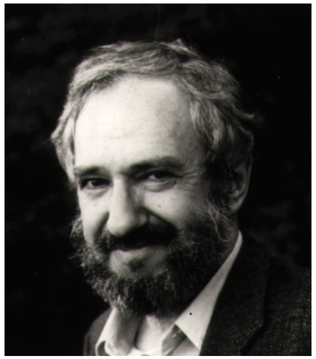
Invented at Calspan Buffalo, NY

Rosenblatt, Frank,

The Perceptron--a perceiving and recognizing automaton.

Report 85-460-1, 1957

Cornell Aeronautical Laboratory



Minsky and Papert

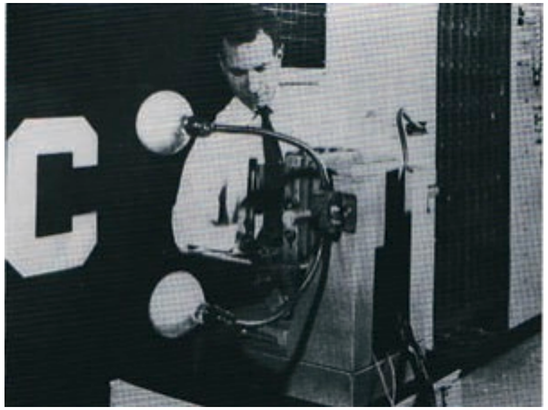
dedicated book to him

Minsky M. L. and Papert S. A. 1969

Perceptrons, MIT Press

Srihari

Perceptron Hardware (Analog)



Learning to discriminate shapes of characters

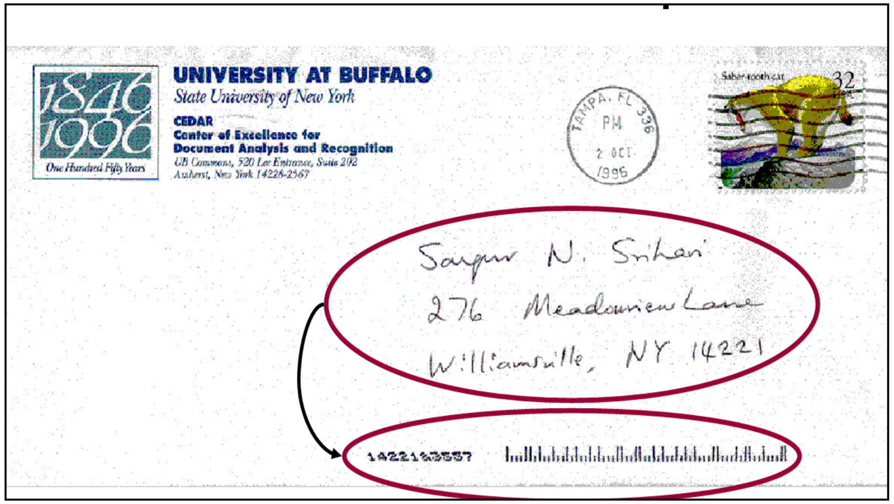
20x20 cell
Image of character

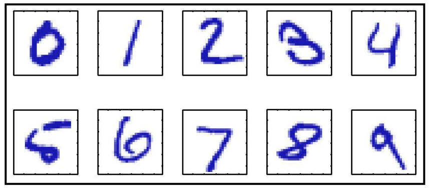
Patch-board to allow different configurations of input features ϕ

Racks of Adaptive Weights
Implemented as potentiometers

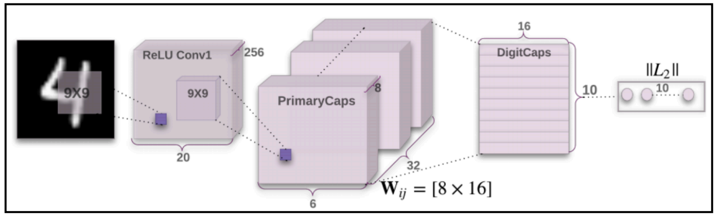
Known as Mark 1 Perceptron. It is now in the Smithsonian.

UB, AI and Fruit-fly





- Many handcrafted rules and exceptions
- Better learn from training set
- Handwriting rec cannot be done without ML!



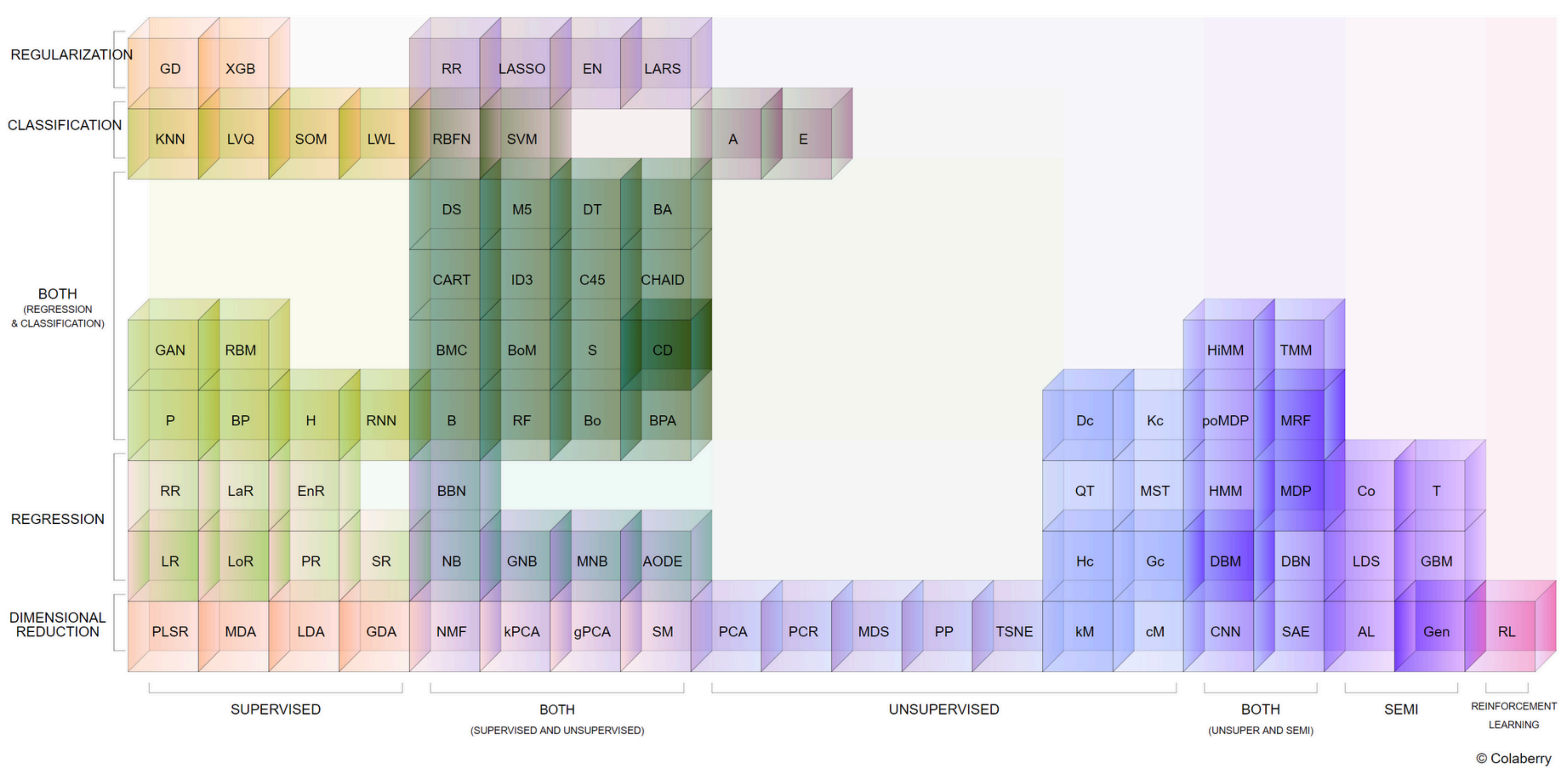
Handwriting is the fruit fly of AI

NYU, Toronto, Montreal, Baidu

ML Problem Types

1. Based on Type of Data
   1. Supervised, Unsupervised, Semi-supervised
   2. Reinforcement Learning
2. Based on Type of Output
   - Regression, Classification
3. Based on Type of Model
   - Generative, Discriminative

Periodic Table of ML algorithms



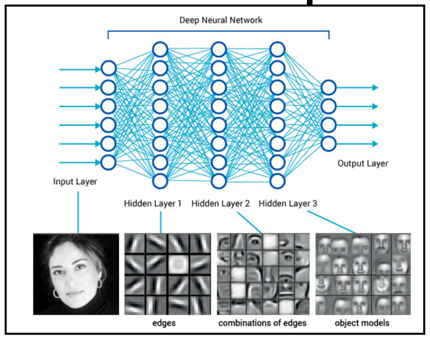
We will look at examples first proceeding along the horizontal axis and then along the vertical

Everyday life needs knowledge

- A person's everyday life requires immense amount of knowledge of the world
  - Much of this knowledge is intuitive and subjective
    - Difficult to articulate in a formal way
- Computers need to capture same knowledge to behave intelligently
- Key challenge of AI is how to get this informal knowledge into a computer

Solution to Intuitive Problems

1. Allow computers to learn from experience
   - By gathering knowledge from experience
     - Avoids need for human operators to specify knowledge that the computer needs
2. Understand the world as hierarchy of concepts
   - Thereby learn complicated concepts by building them out of simpler ones
     - A graph of how these concepts are built on top of each other is deep, with many layers

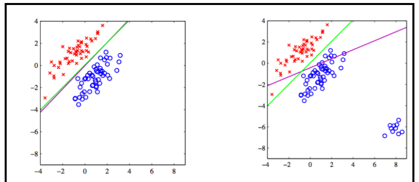


The Knowledge-based Approach

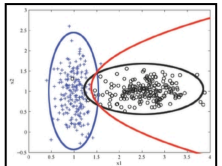
- Hard-code knowledge in a formal language
  - Computer can reason about statements in these languages using inference rules
  - Ex: Cyc is an inference engine and database of statements in CycL
- Unwieldy process
  - Staff of human supervisors
  - People struggle formalize rules with enough complexity to describe the world

The ML approach

- Difficulties of hard-coded approach suggests:
  - AI systems need ability to acquire knowledge
    - By extracting patterns from raw data
- This approach is known as machine learning
  - It allowed tackling problems requiring knowledge of real world and make decisions that seem subjective
  - Simple ML algorithms are:
    - Logistic Regression
      - Caeserian delivery using uterine scar feature
    - Naiive Bayes
      - Decide whether email is spam



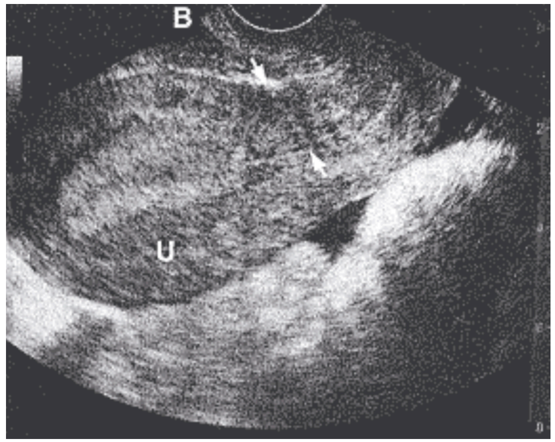
Magenta: Least Squares
Green: Logistic Regression (more robust)



2D binary classification with Naive Bayes. A density contour is drawn for the Gaussian model of each class and the decision boundary is shown in red

Simple ML depends on representation

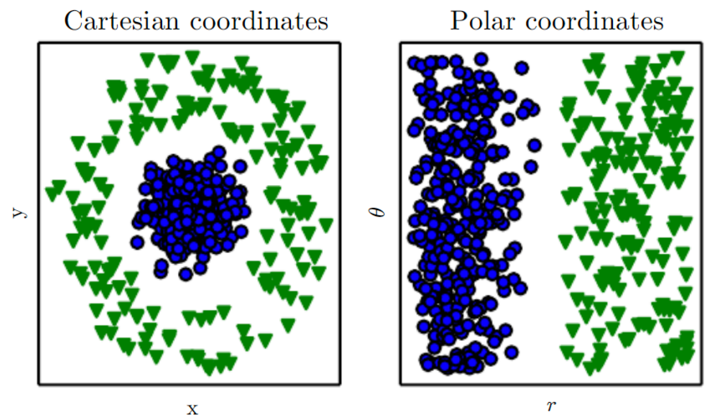
- Simple ML algorithms depend heavily on *representation* of given data
- Ex: Logistic regression to recommend delivery does not examine patient directly
  - Instead doctor provides information such presence of uterine scar (called a feature)
  - If Logistic regression was provided with MRI scan, it could not make useful predictions



Individual pixels have negligible correlation with complications in delivery

Dependence on Representation

- This dependence appears in computer science and daily life
  - In CS: search is exponentially faster if data is data is structured and indexed intelligently 
  - People: arithmetic on Arabic numerals 1,2,…9,10, easier than using Roman numerals I, II,…IX, X,
  - Choice of representation has enormous effect on ML algorithm performance
    - Ex: straight line separation
      - Impossible in Cartesian
      - Simple straight line in polar



Designing right set of features

- Useful feature for speaker ID: size of vocal tract
  - Provides strong clue for: man, woman or child
- Difficult to know what set of features are good for detecting a car in photographs
  - Presence of wheel as a feature
    - Has a simple geometric shape but difficult to describe in terms of pixel values
      - Shadows, glare from metal parts, fender or object in front obscures

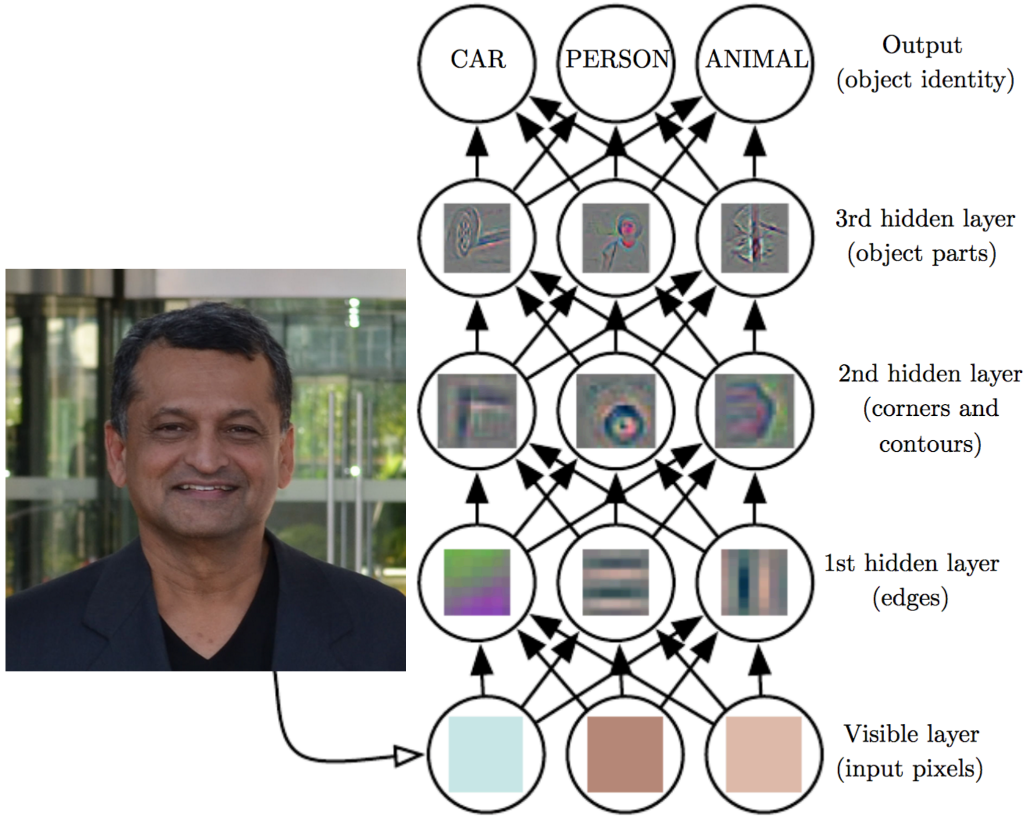


Representation Learning

- Solution: use ML to not only learn mapping from representation to output but representation itself
  - Better results than hand-coded representations
- Allows AI systems to rapidly adapt to new tasks
  - Designing features can take great human effort
    - Can take decades for a community of researchers

Feature Learning for Classification

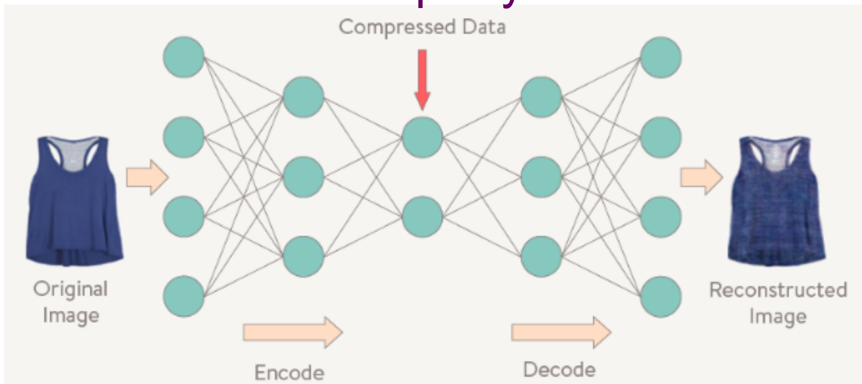
- Function to map pixels to object identity is complicated
- Series of hidden layers extract increasingly abstract features
- Final decision made by a simple classifier



Unsupervised Representation Learning

- Autoencoder:
  - Quintessential example of representation learning
  - Encoder:
    - Converts input into a representation with nice properties
  - Decoder:
    - Converts the representation back to input

Deep Style



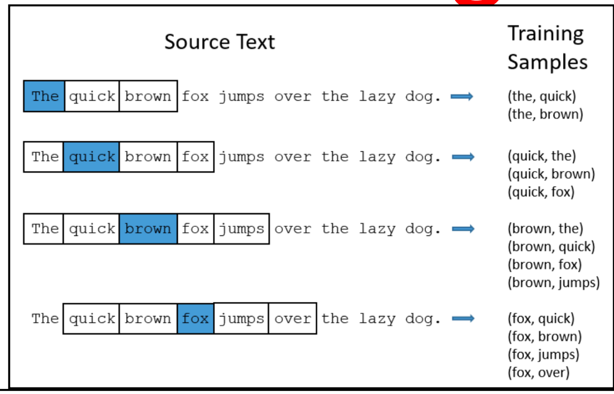
New designs from representation

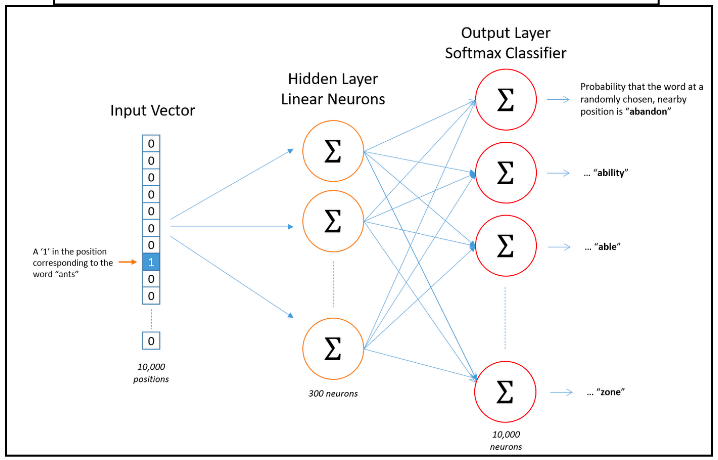
Characteristics of Deep Learning

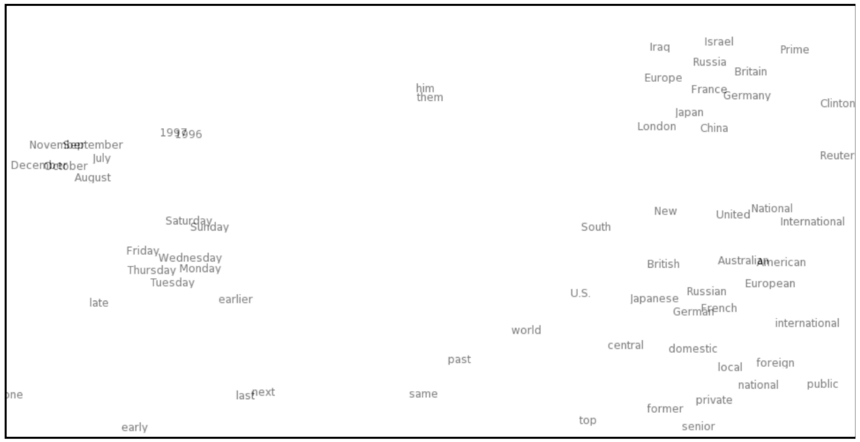
1. A type of Machine Learning that improves with experience and data
2. Only viable approach to building AI systems in real-world environments
3. Obtains its power by a nested hierarchy of concepts
   - each concept defined by relationship to simpler concepts
     - More abstract representations computed in terms of less abstract ones

Natural Language Processing

- Training Data
- Word-to-vec
  - One-hot vector mapped to vector of 300
- Word embedding
  - Similar words are close together

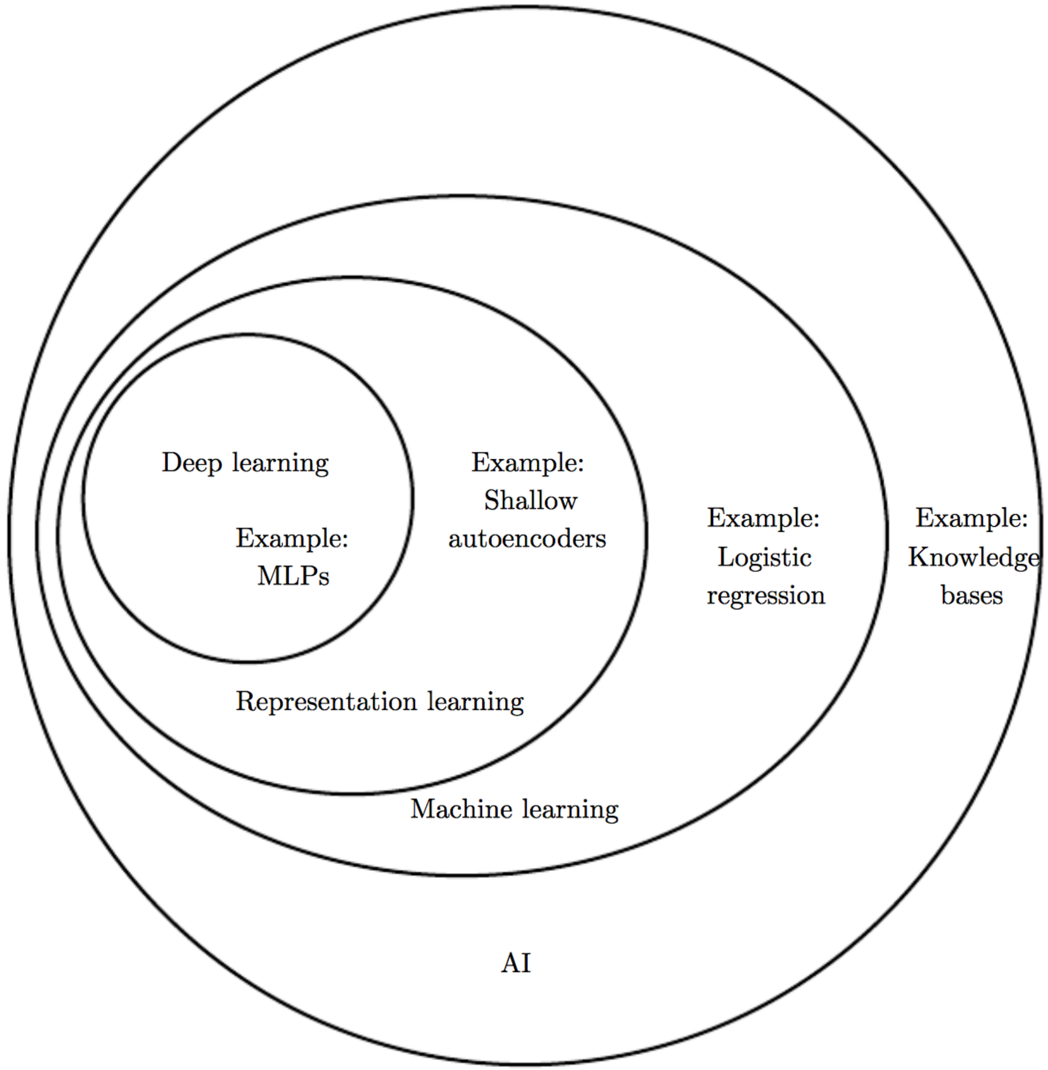




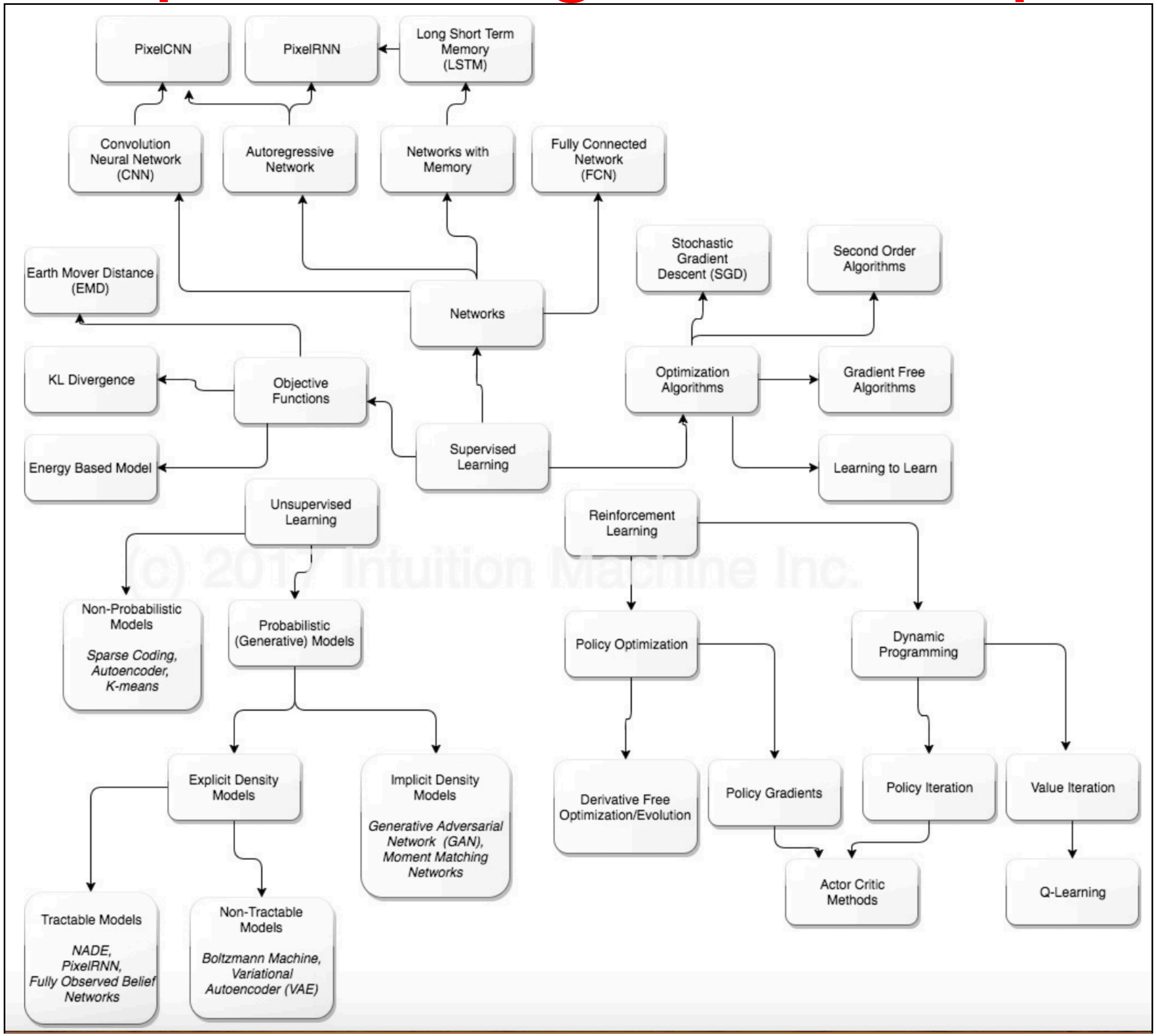


Approaches to AI: Venn Diagram

- AI methods
- Deep Learning is a type of representation learning
- An example AI technology is included in each



Deep Learning Road Map



Study of Deep Learning