SDM274: AI and Machine Learning

Zhiyun Lin

南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

设计智造学院
School of
System Design and
Intelligent Manufacturing

Lecture: Introduction

- Administration details
- Introduction to AI and Machine Intelligence

The Team

Instructor:

- ✓ Zhiyun Lin (林志赟)

Email:

- ✓ linzy@sustech.edu.cn

Office:

- ✓ 701 South Tower, College of Engineering

TA:

- ✓ 吴金泽 12250015@mail.sustech.edu.cn
- ✓ 马晨昊 12332632@mail.sustech.edu.cn
- ✓ 朱骏玮 12431380@mail.sustech.edu.cn

Course Information

- Lecture time and place:
  - ✓ Monday 10:20AM-12:10PM, 1st Lecture hall 506 (weekly)
  - ✓ Wednesday 8:00AM-9:50PM, 1st Lecture hall 506 (biweekly, odd weeks)
  - ✓ Both start from the week of September 9
- Instruction language:
  - ✓ English/Chinese
- Lecture Website:
  - ✓ https://bb.sustech.edu.cn/

群聊：2024秋-SDM274-人工智能与机器学习

Class Group WeChat

该二维码7天内(9月13日前)有效，重新进入将更新

Pre-requisites

Pre-requisites

- ✓ MA102B, MA107A

Do I have the appropriate background?

- ✓ Linear algebra: vector/matrix manipulations, properties
- ✓ Calculus: derivatives and partial derivatives
- ✓ Probability: common distributions; Bayes Rule
- ✓ Statistics: mean/median/mode; maximum likelihood
- ✓ Programming language: Python

Textbook(s)

- Textbook:
  - ✓ C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006

- Other recommending book:
  - ✓ A. Downey, Think Python (version 2.0), Green Tea Press, 2012

E-book are available in SUSTech Blackboard

Provisional Calendar (48 credit hours): Part I

- 01. Introduction to artificial intelligence and machine learning
- 02. Basics of the Python programming language
- 03. Regression and linear regression
- 04. Classification and logistic regression
- 05. Multi-layer perceptron neural networks

Supervised learning; Parametric models;

- 06. Nearest neighbors
- 07. Decision trees

Supervised learning; Nonparametric models;

Provisional Calendar (48 credit hours): Part II

- 08. Multi-class classification
- 09. Clustering
- 10. Principal components analysis
- 11. Support vector machine (SVM)
- 12. Ensemble methods

Unsupervised learning; Parametric models;

Supervised learning; Parametric models;

Assessment

Attendance and Quiz 20%:

- ✓ Attendance
- ✓ Class performance
- ✓ Homework problems
- ✓ In-class quizzes

Projects 40%:

- ✓ Two projects (20% for each): code + project report (paper-like)
- ✓ The topic will be assigned to you (one before midterm and the other before final)

Final Exam 40%:

- ✓ Paper and pencil
- ✓ Cover all learned from lectures

More on Assignments and Projects

- Cheating is not permitted.
  - ✓ Violation of this policy is grounds for a semester grade of F, in accordance with university regulations.

- Each student is responsible for his/her own assignments and projects.
  - ✓ Discussion of assignments and projects should be limited to clarification of the handout itself, and some possible ideas.
  - ✓ Copying pseudocode, codes or simulation results as a hand-in solution is not allowed.

- The schedule of assignments and projects will be announced in SUSTech Blackboard.
  - ✓ Projects handed in later than the due time will receive 0 mark for that piece of work.
  - ✓ Assignments handed in later than the due time will be counted.

Learning Objectives

Upon completion of this course, you will ...

- understand fundamental knowledge and concepts about machine learning and artificial intelligence
- familiarize with broad classes of machine learning algorithms and related fundamental theory (supervised and unsupervised)
- apply and adapt learned machine learning techniques to solve engineering problems.

Ways of Study

- In class
  - ✓ Concentrate on the lecture
  - ✓ Understand at least 70%, otherwise you may stop your lecture instructor
  - ✓ Raise your questions to help your understanding

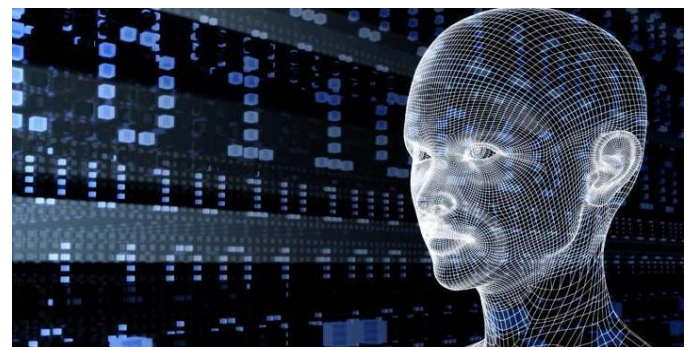
- After class
  - ✓ Do reading!
  - ✓ Do coding!
  - ✓ Practice, Practice, and Practice, ...

- Administration details
- Introduction to AI and Machine Intelligence

什么是人工智能？

AI，即人工智能（Artificial Intelligence），是一门研究如何让机器模拟人类智能的学科。

涉及到构建可以感知、推理、学习和决策的智能系统，以解决复杂问题和实现人类类似的任务。



智慧的源头？

人类是最具智慧的物种，但智慧到底是潜藏于

人类的大脑之中？还是寄居于身体之内？

人工智能的方向

图灵在他1948年写给英国科学院的《智能机器》论文中，曾经把研究智能的方向分为：

- “非具身智能”（Disembodied Intelligence）
- “具身智能”（Embodied Intelligence）

非具身智能

- 关于“非具身智能”的研究，演进成我们今天基于计算的认知科学。
- 以非具身智能的角度来看，所谓智能即是符号操作，起始于大脑的输入，终止于大脑的输出，智能和认知只关注这个符号操作过程。

非具身智能

◘ 符号主义：基于逻辑推理的智能模拟方法

符号主义是一种基于逻辑推理的智能模拟方法，又称为逻辑主义、心理学派或计算机学派，其原理主要为物理符号系统假设和有限合理性原理。

◘ 联结主义：受脑科学的启发

联结主义学派把人的智能归结为人脑的高层活动，强调智能的产生是由大量简单的单元通过复杂的相互联结和并行运行的结果。

具身智能

- “具身智能”的观点则认为智能、认知都是与具体的身体、环境密切相关的，它们之间存在内在的和本质的关联。

- 智能和认知两者必须以一个在环境中的具体的身体结构和身体活动为基础。智能是基于身体和涉及身体的，智能始终是具体身体的智能，而不能仅仅存在于脑海之中。

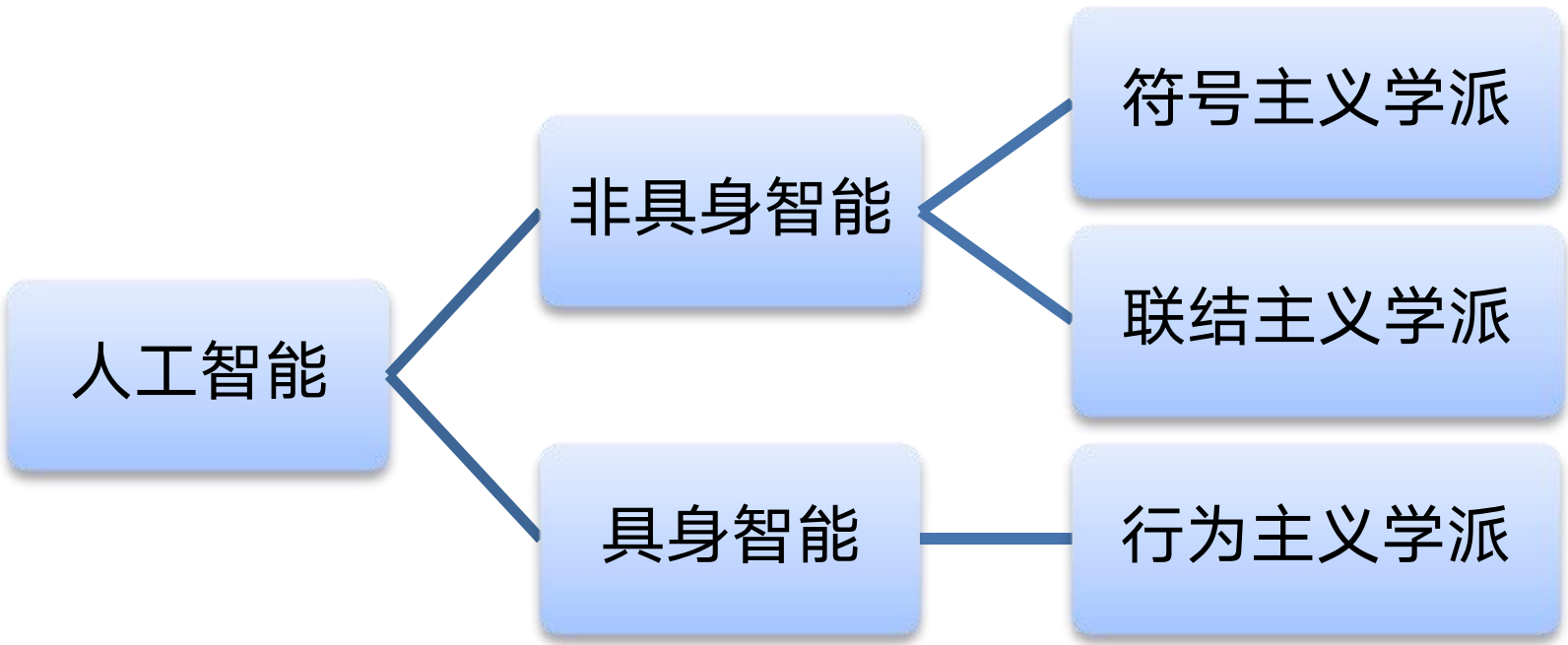
具身智能

- 行为主义：基于“感知——行动”

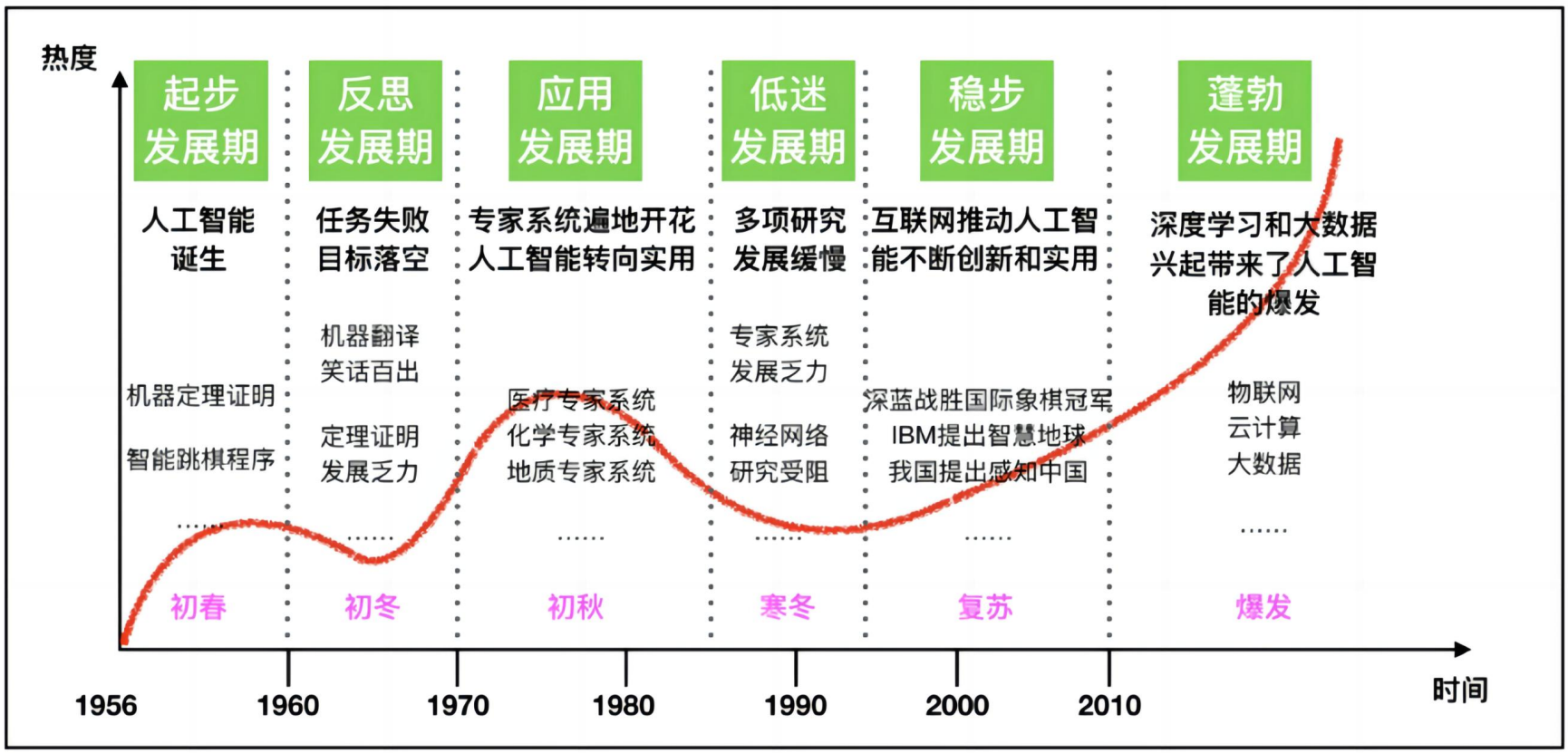
行为主义，是一种基于“感知——行动”的行为智能模拟方法。

- 行为主义学派认为人工智能源于控制论。早期的研究工作重点是模拟人在控制过程中的智能行为和作用，并进行“控制论动物”的研制。到20世纪60~70年代，播下智能控制和智能机器人的种子，并在20世纪80年代诞生了智能控制和智能机器人系统。

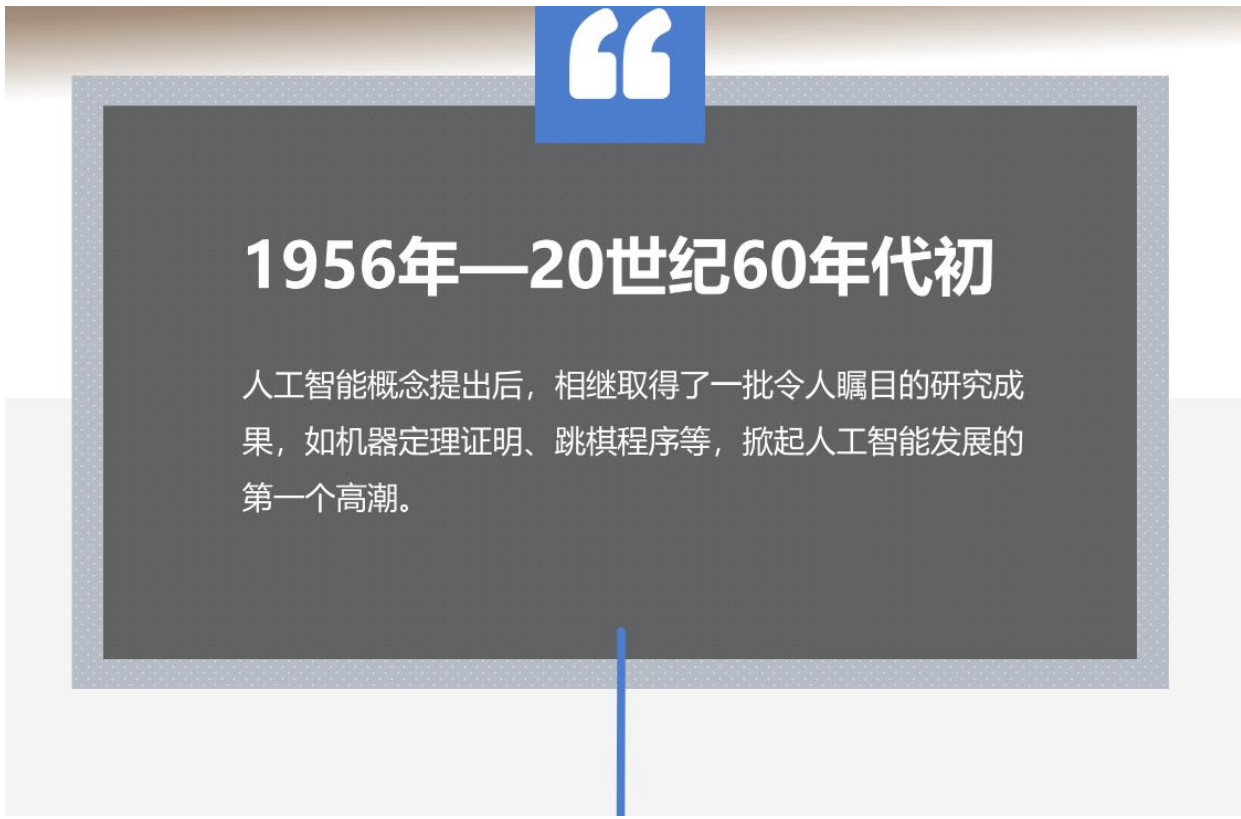
人工智能两大方向、三大学派



人工智能发展史


热度
起步发展期
反思发展期
应用发展期
低迷发展期
稳步发展期
蓬勃发展期
人工智能诞生
任务失败 目标落空
专家系统遍地开花 人工智能转向实用
多项研究发展缓慢
互联网推动人工智能不断创新和实用
深度学习和大数据兴起带来了人工智能的爆发
机器定理证明
智能跳棋程序
.....
机器翻译
笑话百出
定理证明
发展乏力
.....
医疗专家系统
化学专家系统
地质专家系统
.....
专家系统
发展乏力
神经网络
研究受阻
.....
深蓝战胜国际象棋冠军
IBM提出智慧地球
我国提出感知中国
.....
物联网
云计算
大数据
.....
初春
初冬
初秋
寒冬
复苏
爆发
1956
1960
1970
1980
1990
2000
2010
时间

阶段一：起步发展期


1956年—20世纪60年代初
人工智能概念提出后，相继取得了一批令人瞩目的研究成果，如机器定理证明、跳棋程序等，掀起人工智能发展的第一个高潮。

阶段二：反思发展期

反思发展期：
20世纪60年代—70年代初

人工智能发展初期的突破性进展大大提升了人们对人工智能的期望，人们开始尝试更具挑战性的任务，并提出了一些不切实际的研发目标。然而，接二连三的失败和预期目标的落空（例如，无法用机器证明两个连续函数之和还是连续函数、机器翻译闹出笑话等），使人工智能的发展走入低谷。

阶段三：应用发展期

应用发展期：
20世纪70年代初—80年代中

20世纪70年代出现的专家系统模拟人类专家的知识和经验解决特定领域的问题，实现了人工智能从理论研究走向实际应用、从一般推理策略探讨转向运用专门知识的重大突破。专家系统在医疗、化学、地质等领域取得成功，推动人工智能走入应用发展的新高潮。

阶段四：低迷发展期

低迷发展期：
20世纪80年代中—90年代中

随着人工智能的应用规模不断扩大，专家系统存在的应用领域狭窄、缺乏常识性知识、知识获取困难、推理方法单一、缺乏分布式功能、难以与现有数据库兼容等问题逐渐暴露出来

阶段五：稳步发展期

稳步发展期

20世纪90年代中—2010年

由于网络技术特别是互联网技术的发展，加速了人工智能的创新研究，促使人工智能技术进一步走向实用化。1997年国际商业机器公司（简称IBM）深蓝超级计算机战胜了国际象棋世界冠军卡斯帕罗夫，2008年IBM提出“智慧地球”的概念。以上都是这一时期的标志性事件

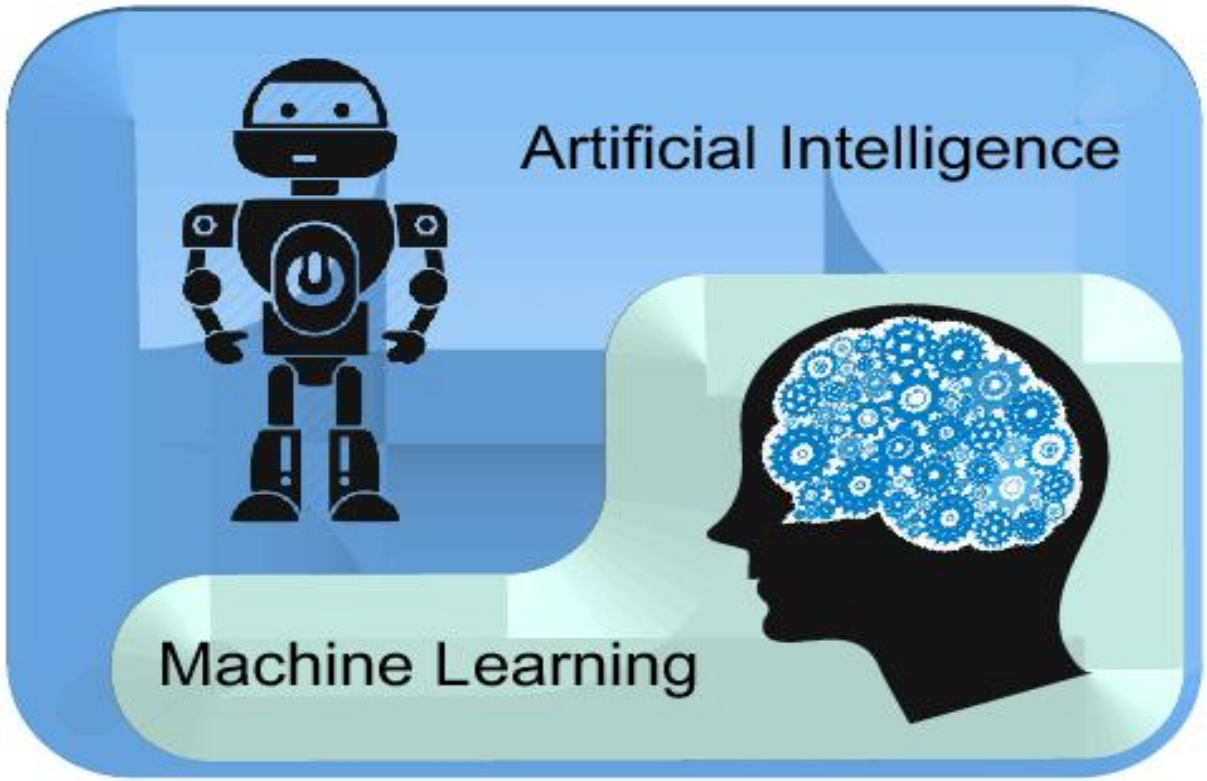
阶段六：蓬勃发展期

蓬勃发展期：

2011年至今

随着大数据、云计算、互联网、物联网等信息技术的发展，泛在感知数据和图形处理器等计算平台推动以深度神经网络为代表的人工智能技术飞速发展，大幅跨越了科学与应用之间的“技术鸿沟”，诸如图像分类、语音识别、知识问答、人机对弈、无人驾驶等人工智能技术实现了从“不能用、不好用”到“可以用”的技术突破，迎来爆发式增长的新高潮

Machine Learning ≠ Artificial Intelligence



- AI is a bigger concept to create intelligent machines that can simulate human thinking capability and behavior
- Machine learning is an application or subset of AI that allows machines to learn from data without being programmed explicitly

What is Machine Learning?

- How can we solve a specific problem?
- ✓ As computer scientists, we write a program that encodes a set of rules that are useful to solve the problem

What is Machine Learning?

- Machine learn to classify a cat picture
- ✓ Offer a large number of cat pictures with labels and train a model
- ✓ Use the trained model to predict where there is a cat or not for any given picture

What is Machine Learning?

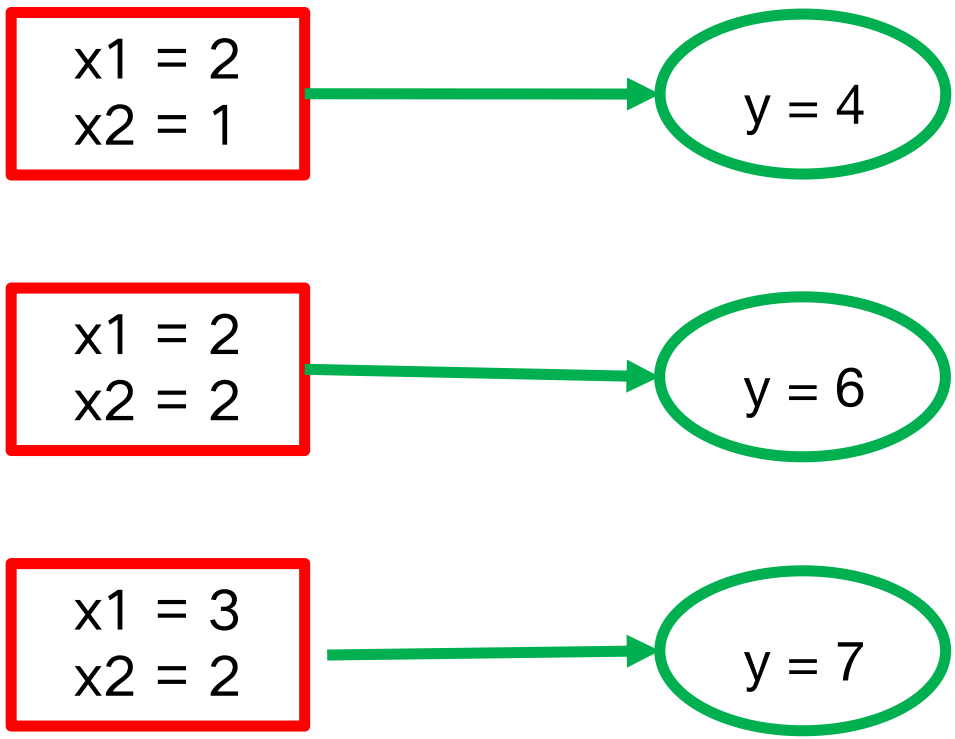
- Learning systems are not directly programmed to solve a problem, instead, they develop own program (rules) based on:

  - ✓ Examples of how they should behave
  - ✓ From trial-and-error experience trying to solve the problem

- Different than standard CS:

  - ✓ Want to implement unknown function, only have access e.g., to sample input-output pairs (training examples)
  - ✓ Learning simply means incorporating information from the training examples into the system

What is Machine Learning?



$y = f(x1, x2)$

$= a * x1 + b * x2$

a? b?

机器学习通过数据样本学习映射 f，或者说学习参数 a 和 b

Why use learning?

- It is very hard to write programs that solve problems like recognizing a handwritten digit

✓ What distinguishes a 2 from a 7?

Why use learning?

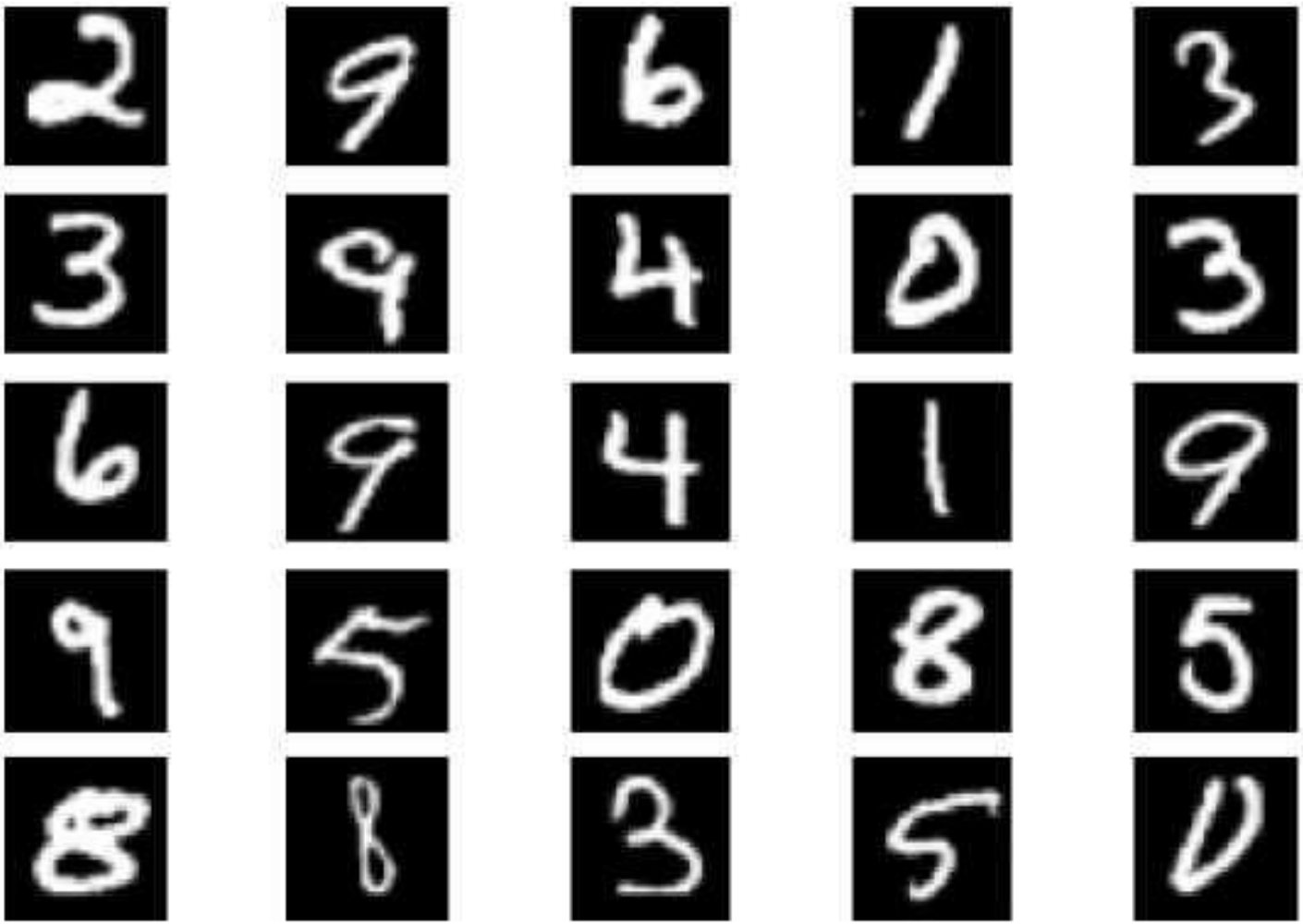
- It is very hard to write programs that solve problems like recognizing a handwritten digit
  - ✓ What distinguishes a 2 from a 7?

- How does our brain do it?
  - ✓ Instead of writing a program by hand, we collect examples that specify the correct output for a given input
  - ✓ A machine learning algorithm then takes these examples and produces a program that does the job
  - ✓ The program produced by the learning algorithm may look very different from a typical hand-written program. It may contain millions of numbers.
  - ✓ If we do it right, the program works for new cases as well as the ones we trained it on.

Learning algorithms are useful in many tasks

1. Classification: Determine which discrete category the example is

Examples of Classification



What digit is this?

Examples of Classification

Is this a dog?

Examples of Classification

Am I going to pass the exam?

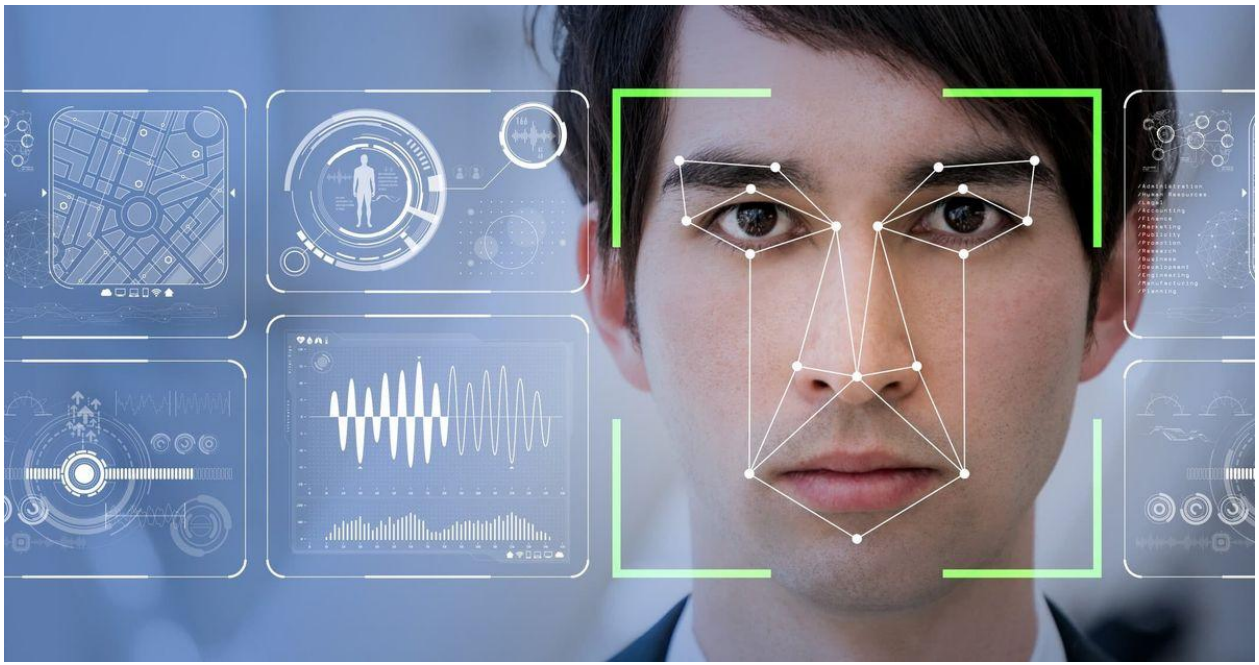
Learning algorithms are useful in many tasks

1. Classification: Determine which discrete category the example is
2. Recognizing patterns: Speech recognition, facial identity, etc

Examples of Recognizing Patterns

Siri supports a wide range of user commands, including performing phone actions, checking basic information, scheduling events and reminders. Its speech recognition engine uses advanced machine learning technologies to function.

Examples of Recognizing Patterns



Facial recognition system: Matching a human face from a digital image or a video frame against a database of faces, typically employed to authenticate users through ID verification services, works by pinpointing and measuring facial features from a given image.

Examples of Recognizing Patterns

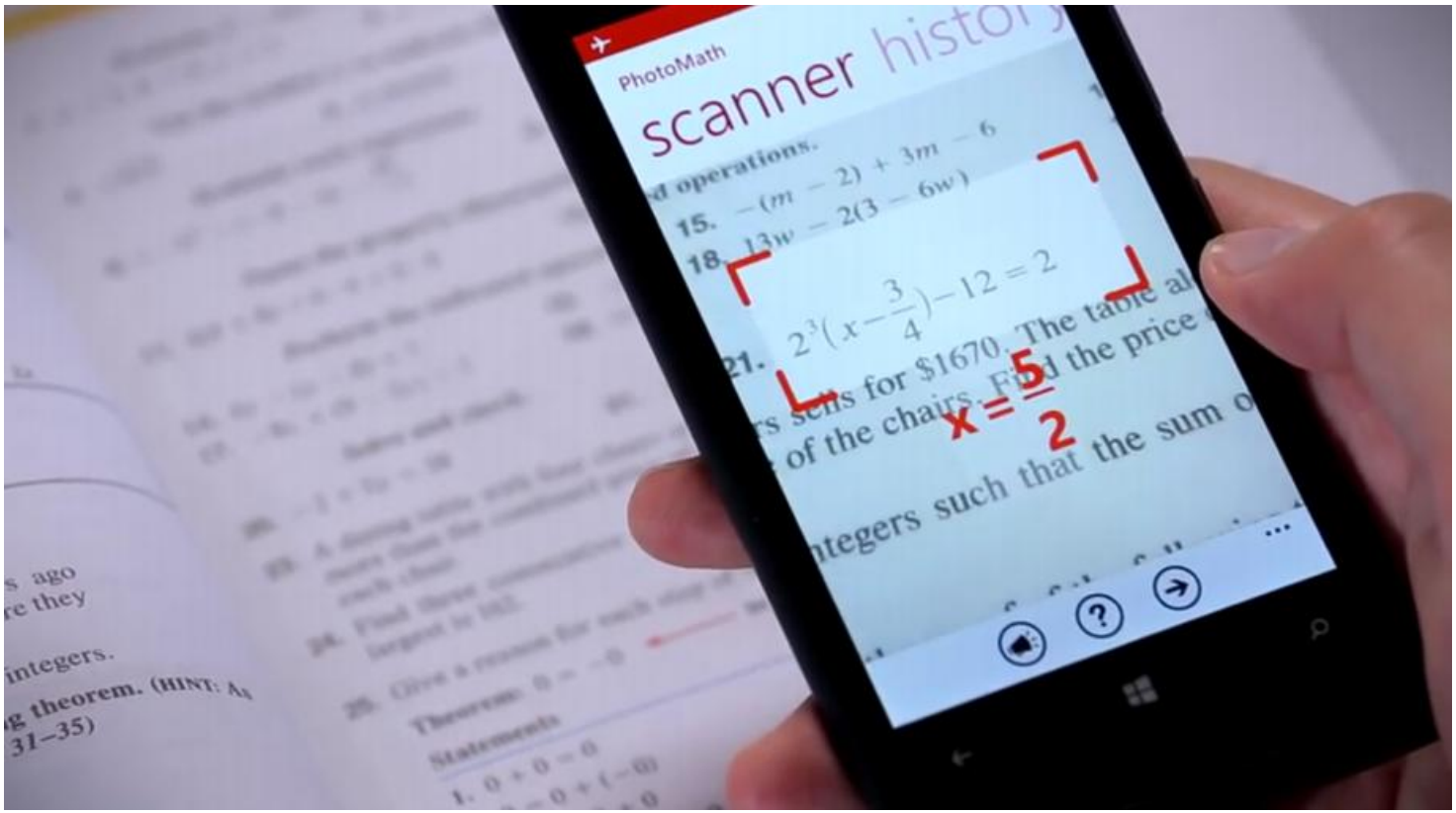


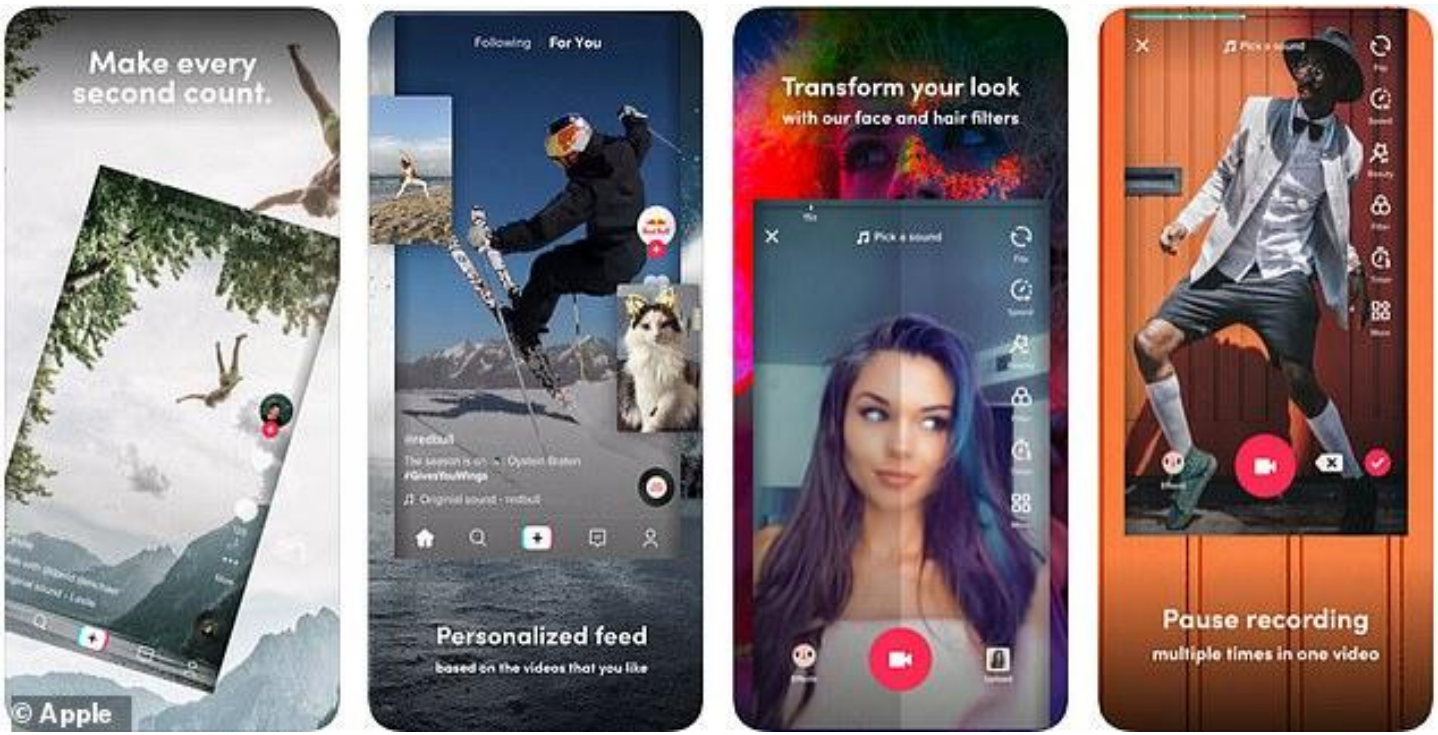
Figure : Photomath: https://photomath.net/

Photomath is a mobile computer algebra system with an augmented optical character recognition system designed for use with a smartphone's camera to scan and recognize mathematical equations.

Learning algorithms are useful in other tasks

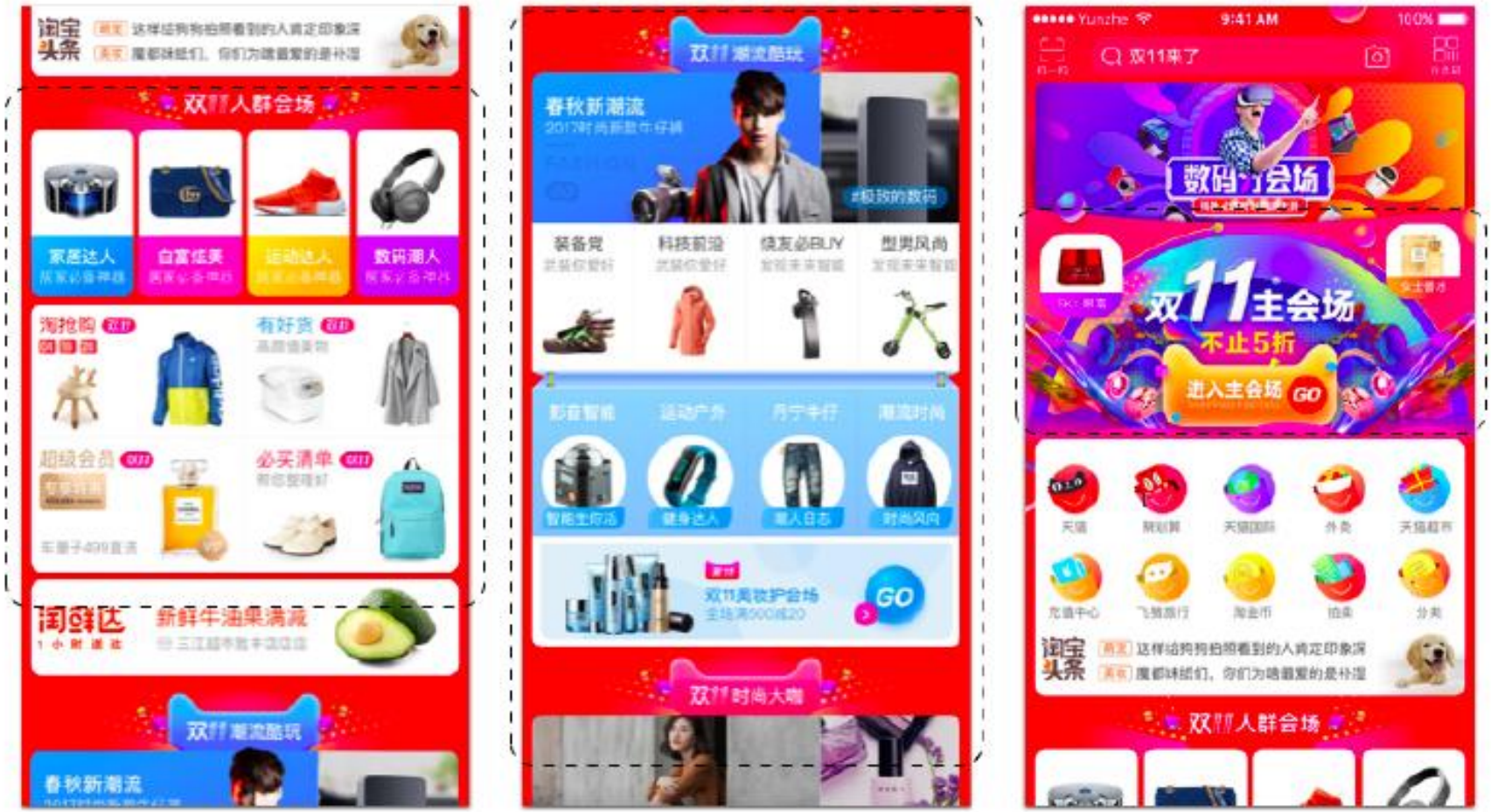
1. Classification: Determine which discrete category the example is
2. Recognizing patterns: Speech Recognition, facial identity, etc
3. Recommender systems: Noisy data, commercial pay-off (e.g., TikTok, Taobao, Amazon, Netflix).

Examples of Recommendation Systems



How TikTok recommends videos #ForYou

Examples of Recommendation Systems

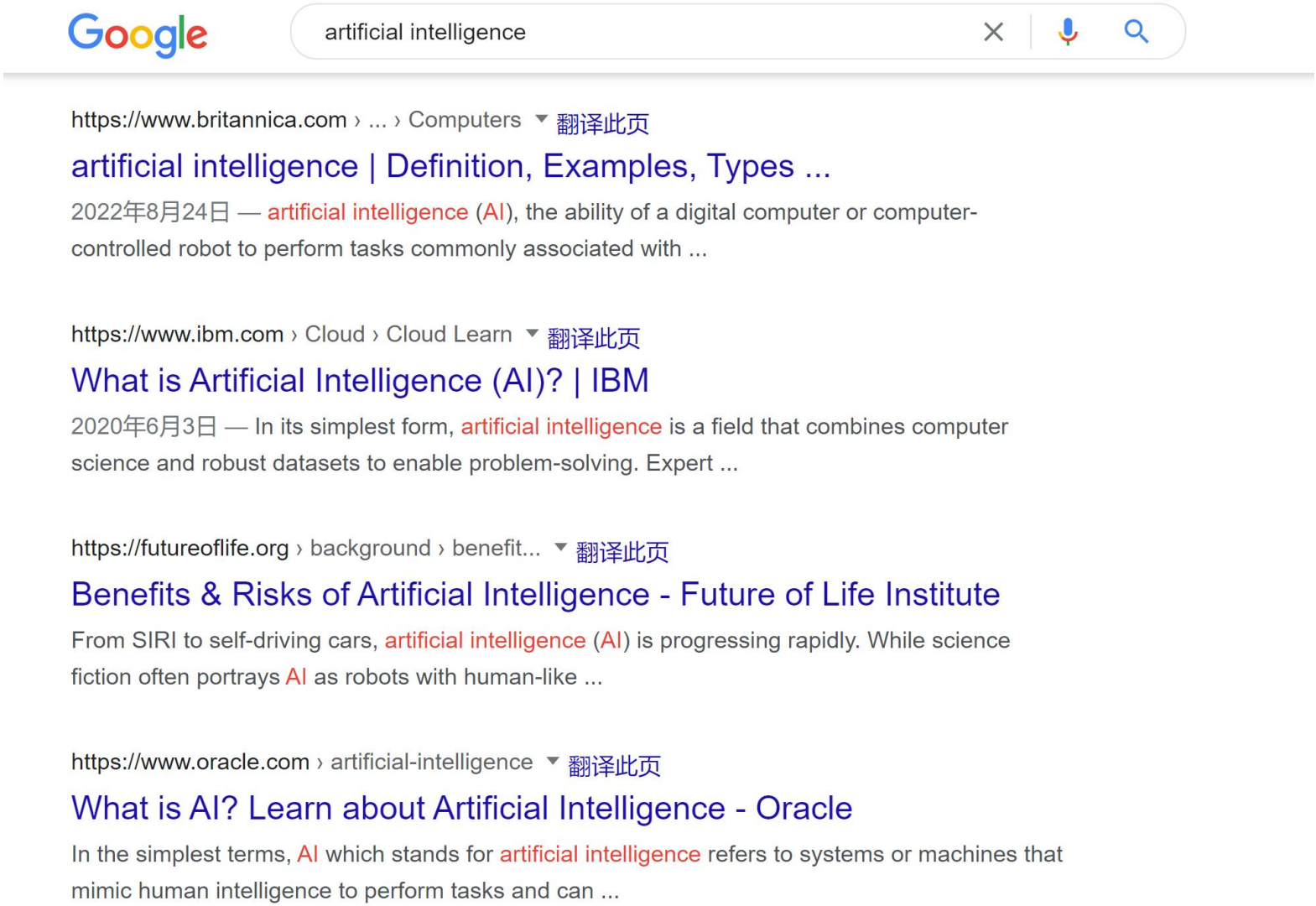


Aims at providing users with attention-grabbing items based on their preferences in Taobao

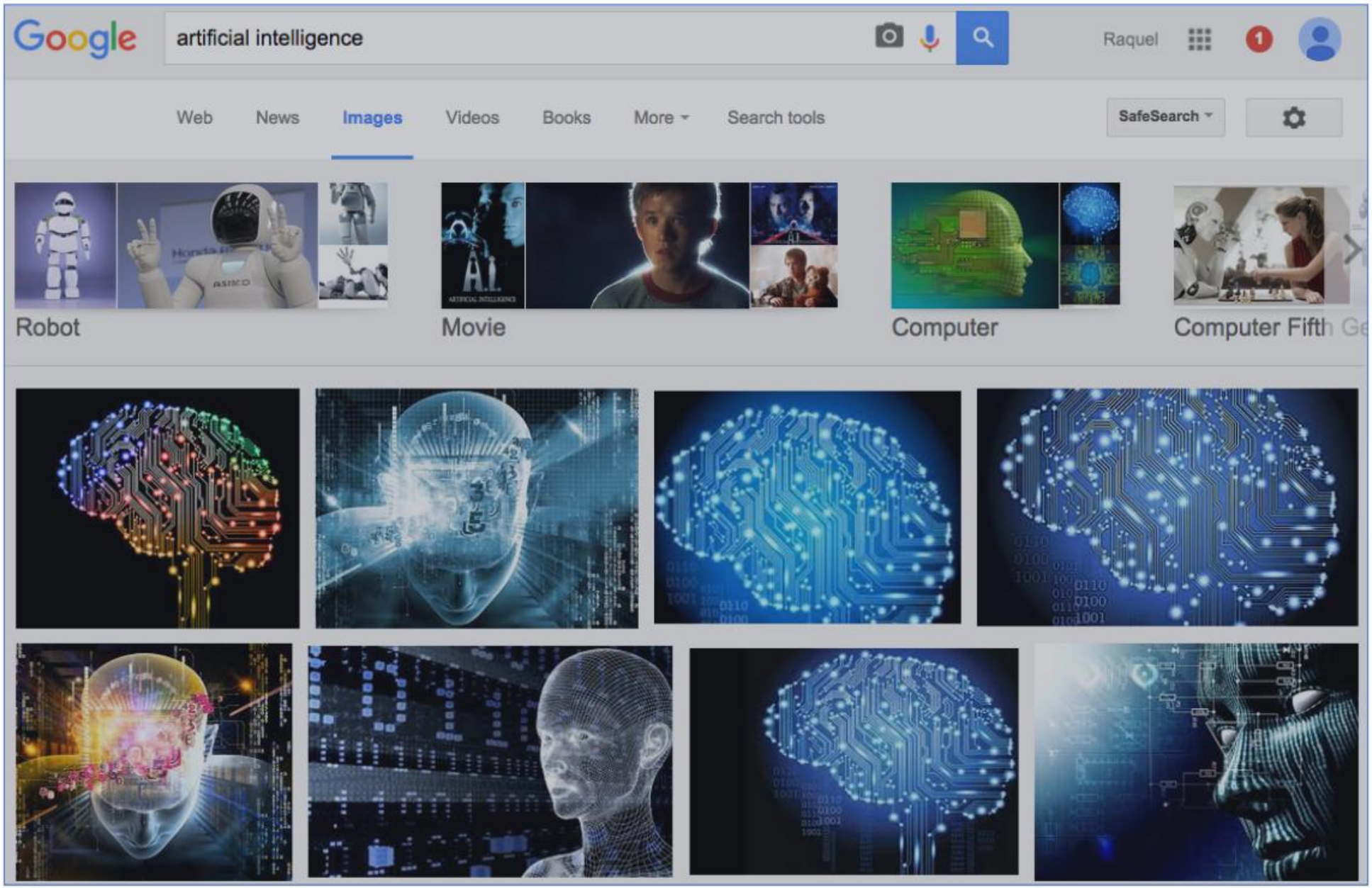
Learning algorithms are useful in other tasks

1. Classification: Determine which discrete category the example is
2. Recognizing patterns: Speech Recognition, facial identity, etc
3. Recommender systems: Noisy data, commercial pay-off (e.g., TikTok, Taobao).
4. Information retrieval: Find documents or images with similar content

Examples of Information Retrieval



Examples of Information Retrieval

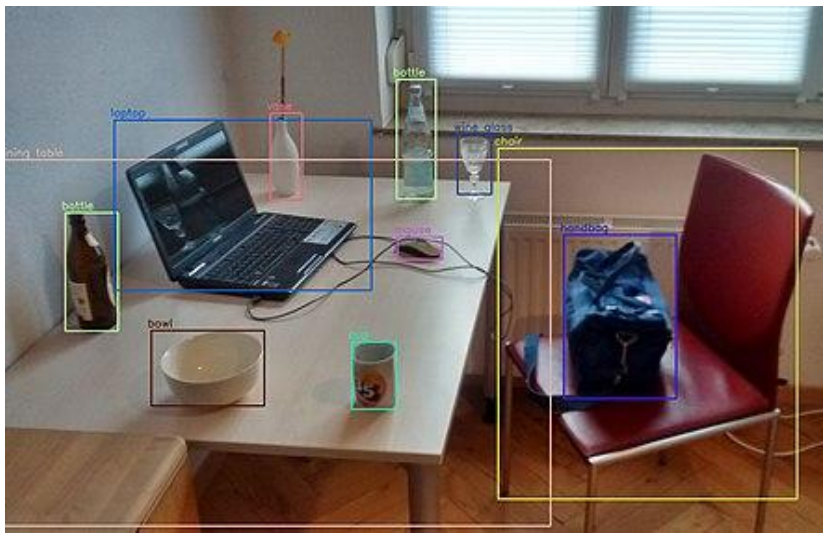


Learning algorithms are useful in other tasks

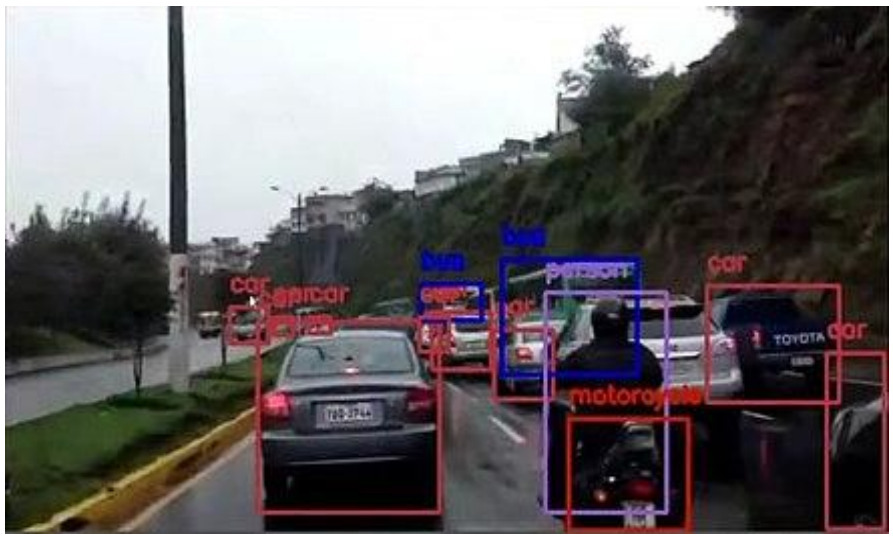
1. Classification: Determine which discrete category the example is
2. Recognizing patterns: Speech Recognition, facial identity, etc
3. Recommender systems: Noisy data, commercial pay-off (e.g., TikTok, Taobao).
4. Information retrieval: Find documents or images with similar content
5. Computer vision: detection, segmentation, depth estimation, optical flow, etc

Computer Vision

Object detection: deals with detecting instances of semantic objects of a certain class (such as humans, buildings, or cars) in digital images and videos.



Objects detected with OpenCV's Deep Neural Network module (dnn) by using a YOLOv3 model capable to detect objects of 80 common classes.



Detection of objects on a road

Computer Vision

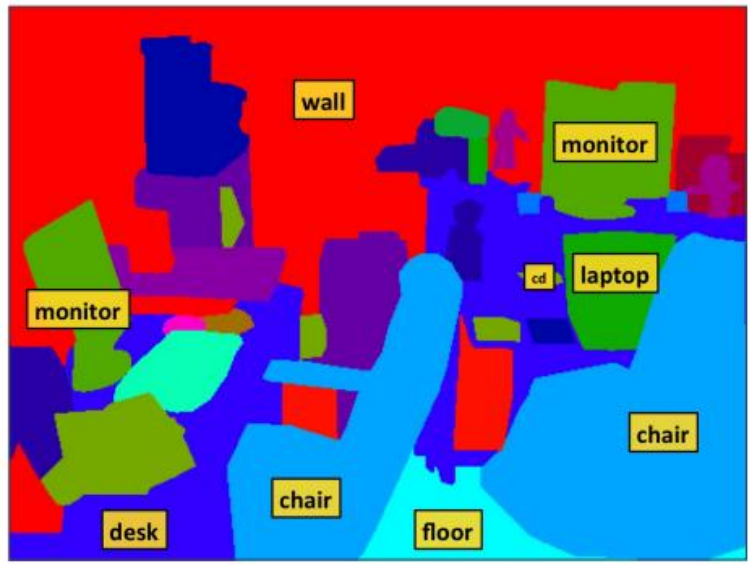
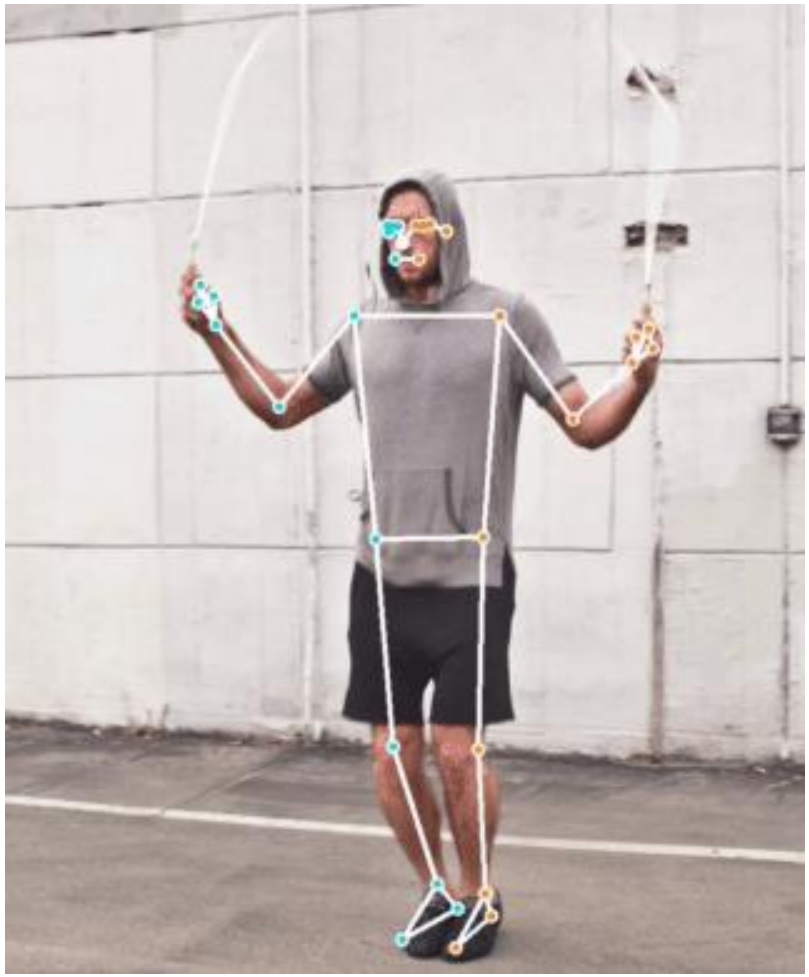


Image segmentation: partition a digital image into multiple image segments (sets of pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze.

Computer Vision



By using RGB-D camera, estimate a human pose

[Gatys, Ecker, Bethge. A Neural Algorithm of Artistic Style. Arxiv'15.]

Convert a picture with artistic styles

Learning algorithms are useful in other tasks

1. Classification: Determine which discrete category the example is
2. Recognizing patterns: Speech Recognition, facial identity, etc
3. Recommender systems: Noisy data, commercial pay-off (e.g., TikTok, Taobao).
4. Information retrieval: Find documents or images with similar content
5. Computer vision: detection, segmentation, depth estimation, optical flow, etc
6. Robotics: perception, planning, etc

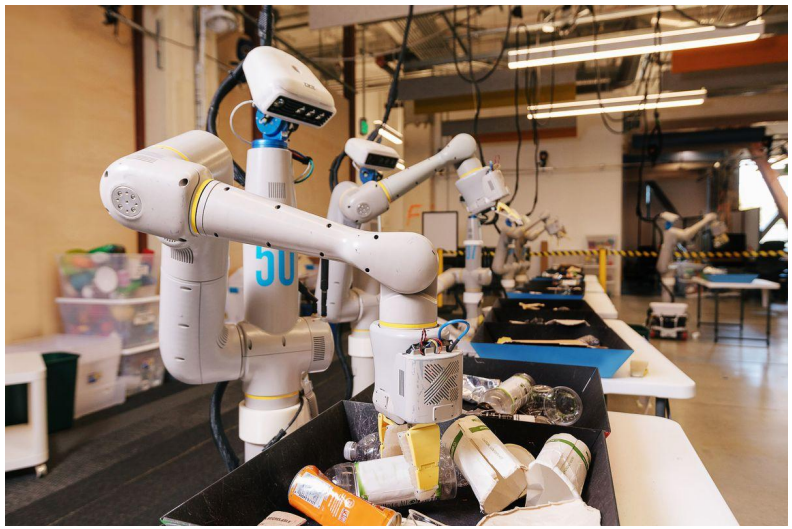
Autonomous Driving

A self-driving car (also known as an autonomous car or driverless car) is a vehicle that uses a different number of sensors, radars, cameras, and artificial intelligence to travel to destinations without needing a human driver. In many aspects, it uses artificial intelligence techniques to make cars self-driving.

Flying Robots

A deep learning neural network helps drones sift through crowds to find and identify persons of interest. It can even inspect large industrial equipment, like telephone towers, and generate a real-time damage report.

Everyday Robots



- Today, robots are expensive and perform highly specialized tasks. But what if a robot could be affordable and taught by just about anyone?

- The everyday robot project creates a robot that can be taught by anyone requires tackling some of the hardest challenges in the field of robotics.They could help people with whatever they needed, doing tasks we haven't even dreamed up yet.

Learning algorithms are useful in other tasks

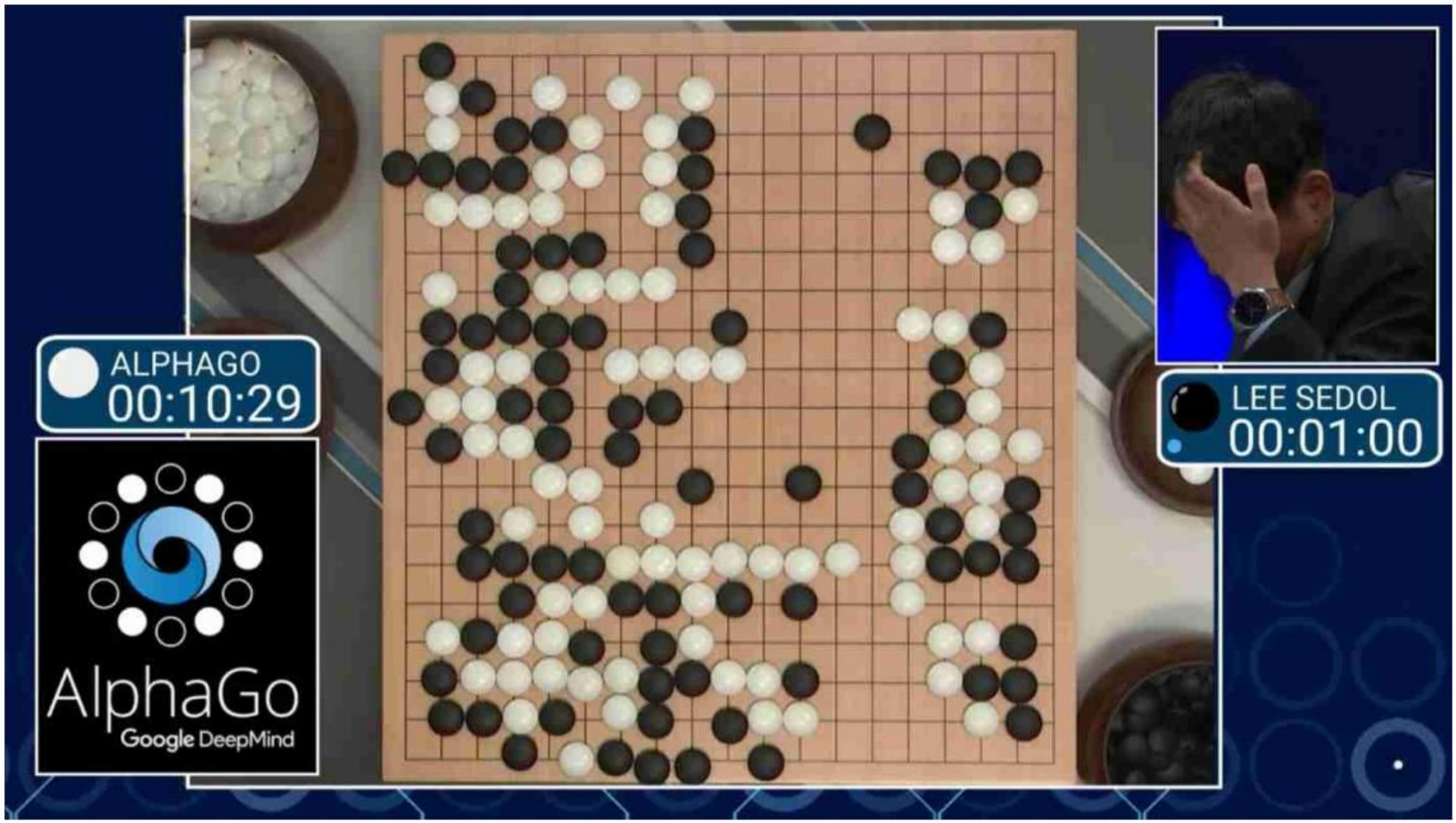
1. Classification: Determine which discrete category the example is
2. Recognizing patterns: Speech Recognition, facial identity, etc
3. Recommender Systems: Noisy data, commercial pay-off (e.g., Amazon, Netflix).
4. Information retrieval: Find documents or images with similar content
5. Computer vision: detection, segmentation, depth estimation, optical flow, etc
6. Robotics: perception, planning, etc
7. Learning to play games

Playing Games: Atari

- Researchers at Google's DeepMind have produced a reinforcement learning (RL) system called Agent57 that has scored above the human benchmark on all 57 Atari 2600 games

- Agent57 uses machine learning called deep reinforcement, which allows it to learn from mistakes and keep improving over time.

Playing Games: AlphaGo



- AlphaGo is the first AI program to defeat a professional human Go player, the first to defeat a Go world champion, and is arguably the strongest Go player in history.

Learning algorithms are useful in other tasks

1. Classification: Determine which discrete category the example is
2. Recognizing patterns: Speech Recognition, facial identity, etc
3. Recommender Systems: Noisy data, commercial pay-off (e.g., Amazon, Netflix).
4. Information retrieval: Find documents or images with similar content
5. Computer vision: detection, segmentation, depth estimation, optical flow, etc
6. Robotics: perception, planning, etc
7. Learning to play games
8. Recognizing anomalies: Unusual sequences of credit card transactions, panic situation at an airport

Learning algorithms are useful in other tasks

1. Classification: Determine which discrete category the example is
2. Recognizing patterns: Speech Recognition, facial identity, etc
3. Recommender Systems: Noisy data, commercial pay-off (e.g., Amazon, Netflix).
4. Information retrieval: Find documents or images with similar content
5. Computer vision: detection, segmentation, depth estimation, optical flow, etc
6. Robotics: perception, planning, etc
7. Learning to play games
8. Recognizing anomalies: Unusual sequences of credit card transactions, panic situation at an airport
9. Spam filtering, fraud detection: The enemy adapts so we must adapt too

Learning algorithms are useful in other tasks

1. Classification: Determine which discrete category the example is
2. Recognizing patterns: Speech Recognition, facial identity, etc
3. Recommender Systems: Noisy data, commercial pay-off (e.g., Amazon, Netflix).
4. Information retrieval: Find documents or images with similar content
5. Computer vision: detection, segmentation, depth estimation, optical flow, etc
6. Robotics: perception, planning, etc
7. Learning to play games
8. Recognizing anomalies: Unusual sequences of credit card transactions, panic situation at an airport
9. Spam filtering, fraud detection: The enemy adapts so we must adapt too
10. Many more!

Questions

1. How do we understand the relationship between aritificial intelligence and machine learning? ( )

(a) Artificial intelligenc is just machine learning
(b) Artificial intelligence is a subset of machine learning
(c) Machine learning is a subset of artificial intelligence
(d) Machine learning has nothing related to artificial intelligence

2. How many AI winters were there prior to 2020? ( )

(a) 0
(b) 1
(c) 2
(d) 3

Questions

3. Which of the following describes meachine learning? ( )

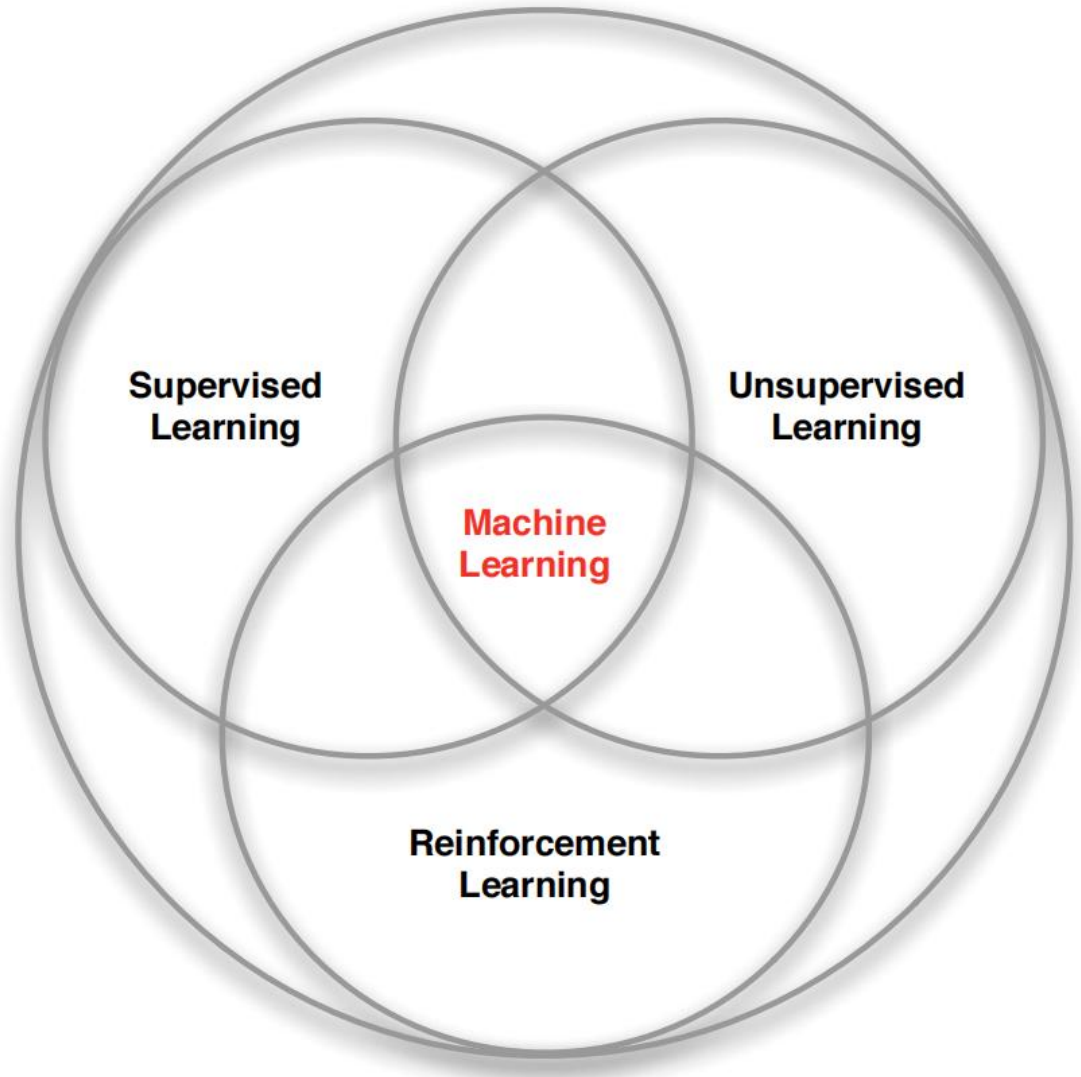
(a) Write a program that encodes a set of specific rules that are useful to solve a problem.

(b) Let the computer develop its own program (rules) based on data.

(c) Offer a large number of samples with labels and train a model; then use the trained model to predict something new.

(d) From trial-and-error experience try to solve a problem.

Branches of Machine Learning



Types of learning tasks

- Supervised learning:

✓ Require a training set of example inputs and their corresponding desired outputs, which the algorithm uses to learn a model of the mapping from inputs to outputs.

✓ After the model is learned, the algorithm can generate outputs for new inputs.

  - Classification: 1-of-N output (speech recognition, object recognition, medical diagnosis)
  - Regression: real-valued output (predicting market prices, customer rating)

Types of learning tasks

- Unsupervised learning:

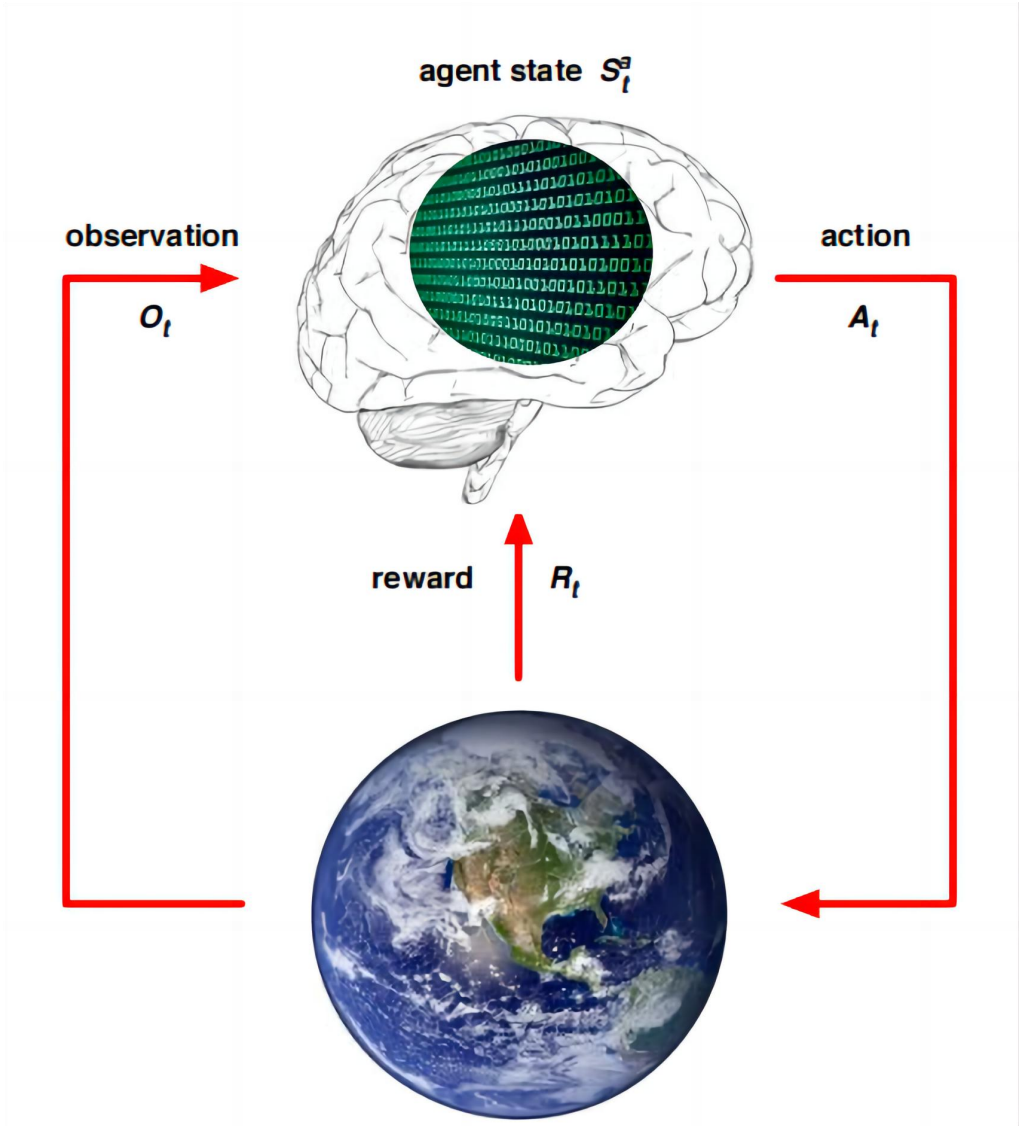
✓ Do not require a training set, but instead, learn a model of the input data by detecting patterns in it.

✓ Unsupervised learning algorithms can be used to discover structure in data or to cluster data into groups.

✓ Examples: Form clusters; Extract features

Types of learning tasks

- Reinforcement learning:



- Agent, environment
- State, action, reward
- Policy, value
- Exploitation, exploration

Types of learning tasks

Reinforcement learning:

✓ What makes reinforcement learning different from other machine learning paradigms?

- There is no supervisor, only a *reward* signal
- Feedback is delayed, not instantaneous
- Time really matters (sequential, non i.i.d data)
- Agent's actions affect the subsequent data it receives

✓ Reinforcement learning is based on the reward hypothesis

Definition (Reward Hypothesis)

All goals can be described by the maximisation of expected cumulative reward

Questions

1. Which of the following is a branch of machine learning? **( )**

(a) Supervised learning
(b) Unsupervised learning
(c) Reinforcement learning
(d) None of the above

2. Which of the following belongs to the classification problem? **( )**

(a) Speech recognition
(b) Object recognition
(c) Medical diagnosis
(d) Marketing price prediction

Install Python

- Install Anaconda
  - ✓ Anaconda is a distribution of the Python and R programming languages for scientific computing (data science, machine learning applications, large-scale data processing, predictive analytics, etc.), that aims to simplify package management and deployment.
  - ✓ Download the right version from https://www.anaconda.com/

- After installing Anaconda, you will notice the following in the start-up menu (under “Anaconda3” folder)
  - ✓ Anaconda Powershell Prompt
  - ✓ Jupyter Notebook (Jupyter is a very popular application used for data analysis. It's an IPython notebook ("interactive python"). You can run each block of code separately.)
  - ✓ Spyder (Spyder is an Integrated Development Environment (IDE) for Python like Atom, Visual Studio, etc.)

Install Python (cont'd)

- Open "Anaconda Powershell Prompt"

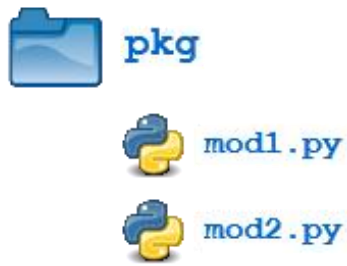
```
Anaconda Powershell Prompt (anaconda3)
(base) PS C:\Users\Daniel> python --version
Python 3.8.3
(base) PS C:\Users\Daniel> conda env list
# conda environments:
#
base                  *  D:\Users\Daniel\anaconda3
py3.10                   D:\Users\Daniel\anaconda3\envs\py3.10
py3.8_Jorldy             D:\Users\Daniel\anaconda3\envs\py3.8_Jorldy
py3.9                    D:\Users\Daniel\anaconda3\envs\py3.9

(base) PS C:\Users\Daniel> conda activate py3.8_Jorldy
(py3.8_Jorldy) PS C:\Users\Daniel> conda deactivate py3.8_Jorldy
(base) PS C:\Users\Daniel>
```

- ✓ Manage Python environment with conda

```
conda create -n your_env_name python=x.x
```

Install Python (cont'd)

- Python Packages:
  - ✓ Suppose you have developed a very large application that includes many modules. Packages allow for a hierarchical structuring of the module namespace using dot notation.
  - ✓ Creating a package is quite straightforward, since it makes use of the operating system's inherent hierarchical file structure.



- Install Python Packages (e.g., NumPy: the core library for scientific computing in Python)
  - ✓ conda install xxx
  - ✓ pip install xxx

	pip	Conda
Installs Python	No	Yes, as package
3rd-party shared libraries	Inside the wheel	Yes, as package
Executables and tools	No	Yes, as package
Python source code	Yes, as package	Yes, as package

Edit and Run Python Code

- Spyder (an IDE for Python)
  - ✓ Python file: xx.py

```
test1 > helloWorld.py
    1   print("Hello World!")
```

- VS Code (suggest! Visual Studio Code is a code editor redefined and optimized for building and debugging modern web and cloud applications.)
  - ✓ https://code.visualstudio.com/

- How to run Python code
  - ✓ pyhton xxx.py

```
(py3.8_Jorldy) D:\Academic_WORK\academic22-25\work2022\coding\MyPython\test1>python helloWorld.py
Hello World!
```

- Jupyter Notebook (Interactive python)
  - ✓ IPython file: xx.ipynb
  - ✓ Convert xx.ipynb to xx.py: Download as Python (.py)
  - ✓ run xx.py into an IPython notebook cell: %run xx.py

Questions

1. Which of the following statement is true? **( )**

(a) Conda can install python

(b) Pip can install python

(c) Conda can manage pthon package by using conda install xxx

(d) Pip can manage pthon package by using pip install xxx

Homework Assignment

- To have the Python environment and packages ready on your computer

✓ Install Anaconda (Windows/Linux OS)
✓ Create a virtual environment and install Python (suggest Python version 3.7, 3.8, or 3.9)
✓ Install required packages: numpy, scipy, scikit-learn, matplotlib, PyTorch
✓ Take a screen shot for the following
  - conda env list
  - activate your environment and python --version
  - show that all above packages are installed by using the command pip show xxx (xxx refers to the package name above)

- Install VS code and get familiar with the use of VS code as well as Jupyter Notebook

✓ Take a screen shot for the following
  - Edit a simple sample xx.py file using VS code
  - Run the python script in VS code terminal and show the output
  - Edit a simple sample xx.ipynb file using Jupyter Notebook
  - Run the cell in the iPython file and show the output