

# 程序 de 序

# 进

Evolution

# 化

# 学习计算机编程语言的时间路线

The timeline of learning programming

人工智能，大数据、机器学习

01

计算机的基本语言  
The computer language

02

C语言及其使用  
C language and application

03

C#语言及其使用  
C# language and application

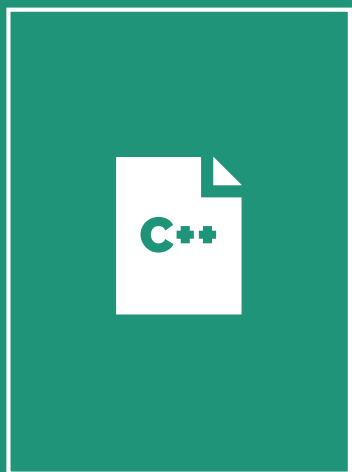
04

python语言及其应用  
Python language and application



# 计算机语言

The language of computer



高级语言

High-level language

人类易于阅读的语言，  
如C语言、C#、Java、  
Python等

编译器

compiler



低级语言

Machine code

计算机能够执行的最底  
层语言，包括汇编语言、  
机器语言等

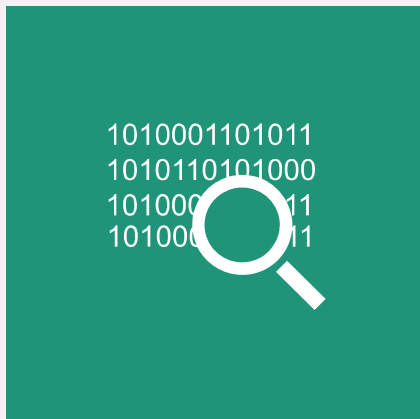
## 2020年5月编程语言Top5

排名	编程语言	流行度
1	C	17.19%
2	Java	16.10%
3	Python	8.36%
4	C++	5.95%
5	C#	4.73%

数据来源：[www.tiobe.com](http://www.tiobe.com)

# 经典计算机与量子计算机

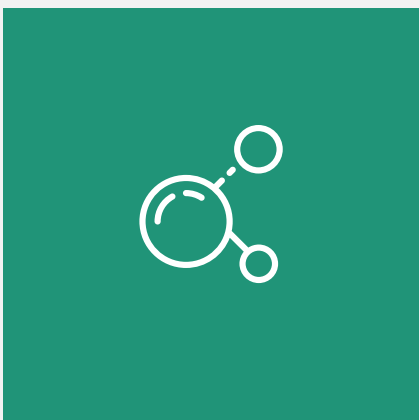
Classical and quantum



经典状态的二进制

Classical State

经典计算机使用经典物理状态（高低电势）来操作信息



量子叠加状态

Quantum State

量子计算机使用量子叠加的物理状态（如自旋的状态）来操作信息

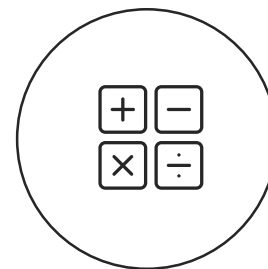
QINGLU青鹿

Quantum spin state

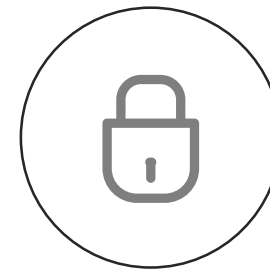
量子自旋叠加态

Superposition of two spin states

$$|\nearrow\rangle = a |\uparrow\rangle + b |\downarrow\rangle$$



可以同时操作2个信息

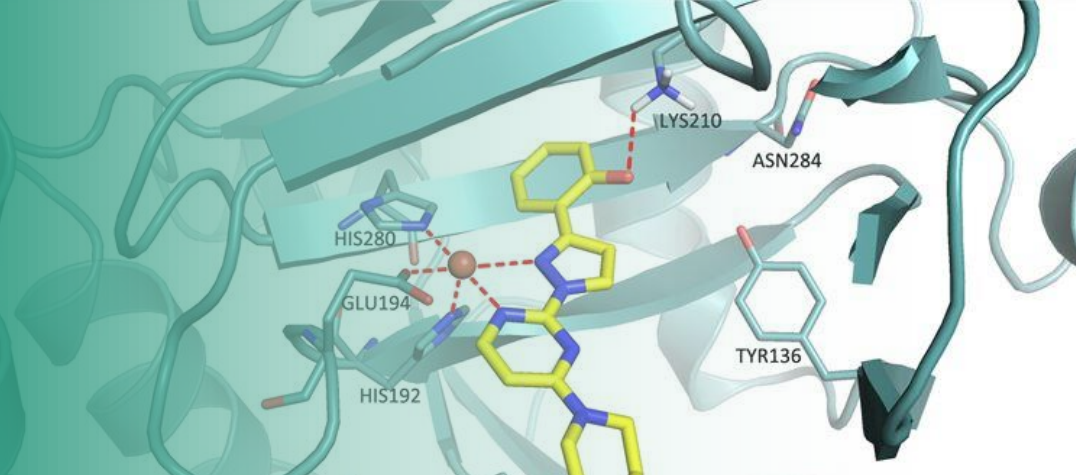


保密性强

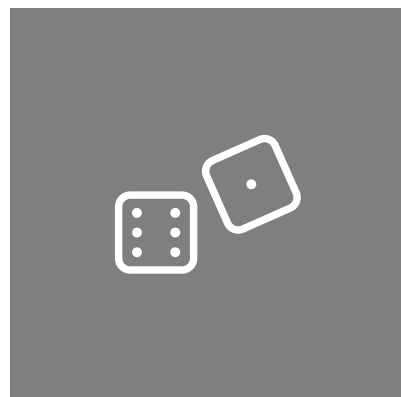


# 第1次

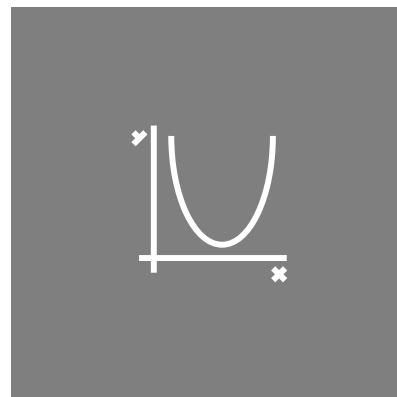
The First Time for programming  
编程解决实际问题



RNA 1级序列



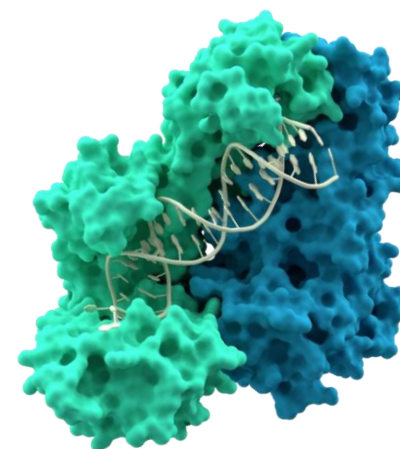
Monte Carlo method



Statistical Potential



Retraining Mechanism

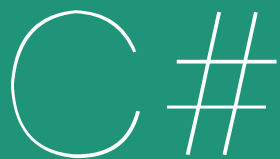


RNA 3维空间结构

使用C语言与BashShell脚本语言构建

# 学习面向对象的编程

Study OOP language

The C# logo is displayed in white text on a dark green square background.

C Sharp Language

由微软公司开发，基于.net框架，面向对象的  
高级编程语言

# 面向对象的编程

Object Oriented Programming



# 利用C#爬虫

Using C# language for web crawling



HtmlAgilityPack

利用Xpath爬取



GeckoFx

使用浏览器加载JS渲染后的  
网页爬取数据

## 工作应用

Using in Jobs

爬取国家资源平台资源信息





# PPT 插件

## 青鹿资源平台插件

结合GeckoFx与Ribbon技术将青鹿资源平台嵌入到PPT中

QINGLU 青鹿

精品  
资源



公共  
资源

在PowerPoint中直接调用

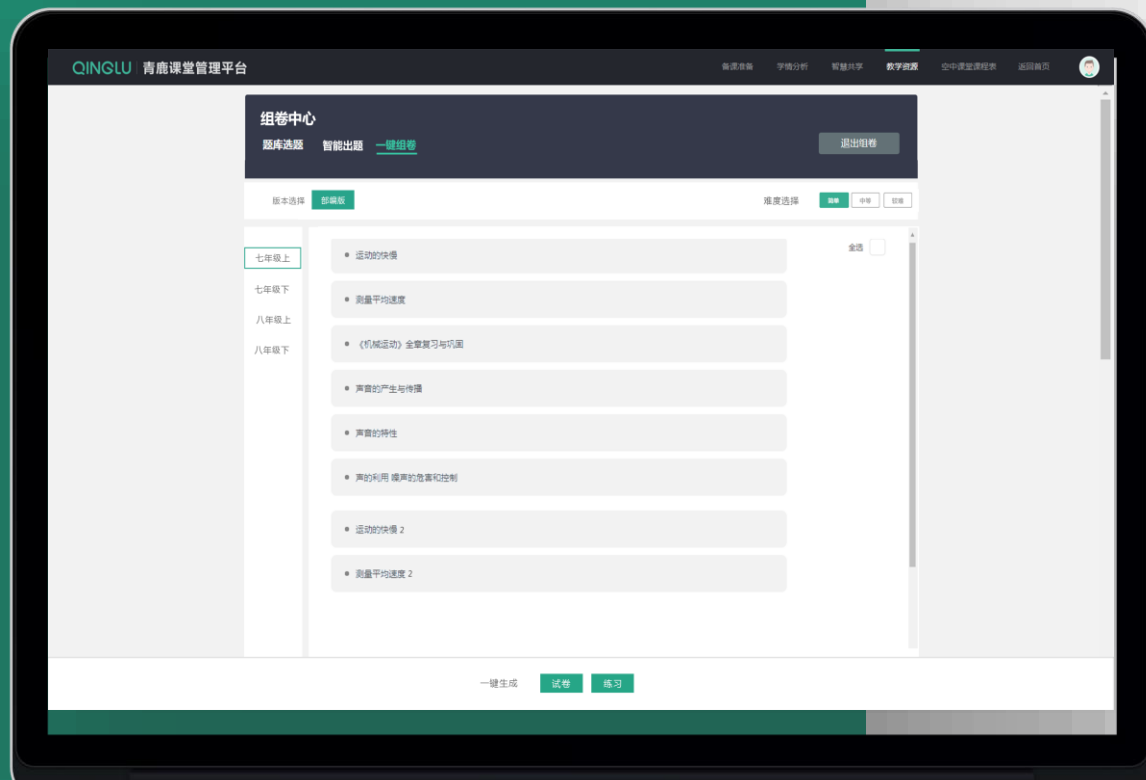




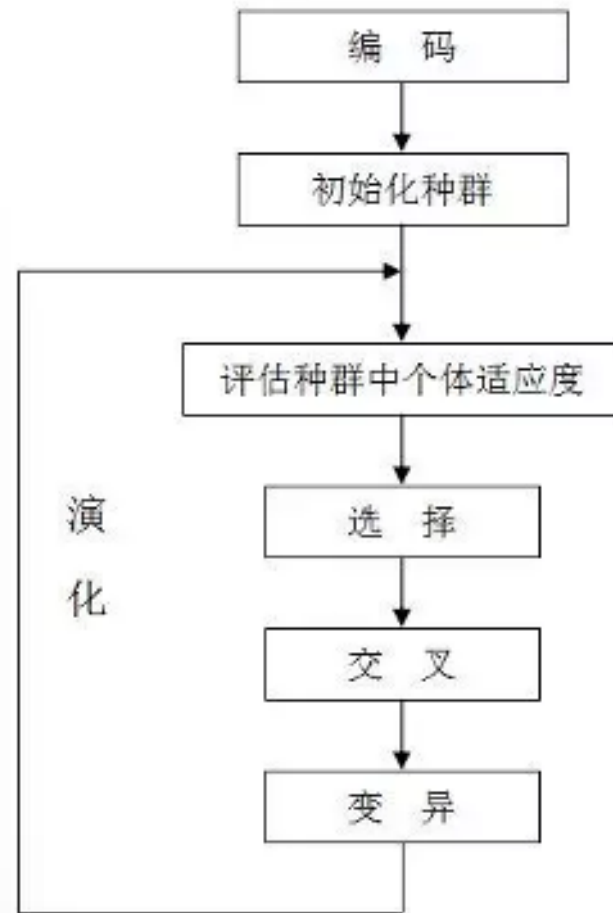
# 基于遗传算法的一键组卷DEMO

Generating the Test based on GA algorism

## 2 步组卷



数学问题：在一定的约束条件下，搜索最优解问题

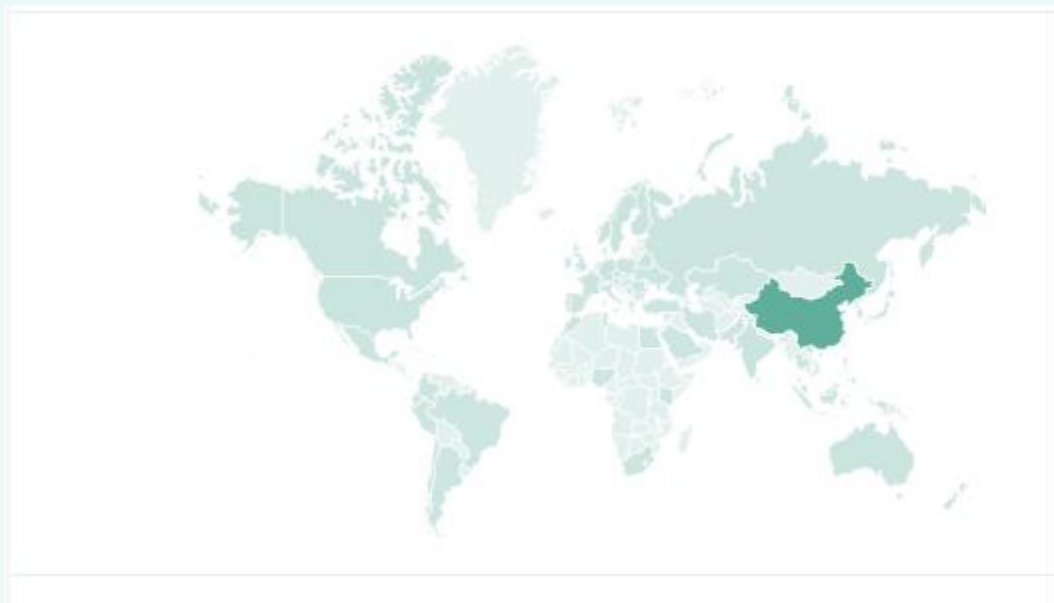




# Python语言

Using Python language

python关键词搜索热度（Google趋势）



Python语言在中国非常流行

为什么Python语言这几年这么火？

具有数量巨大的第三方库开源生态圈，约24万\*

实现人工智能框架的最主流语言，如TensorFlow等

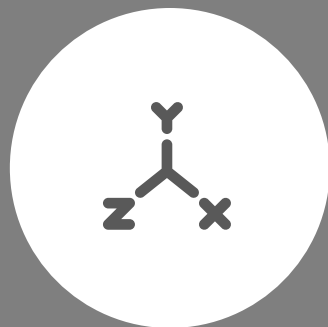
# 利用FuzzyWuzzy库模糊匹配

Using Python library FuzzyWuzzy



分词

Generating the words



相似度计算

Similarity

```
from fuzzywuzzy import fuzz
from fuzzywuzzy import process
```

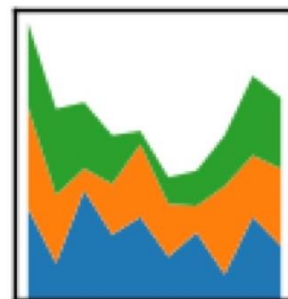
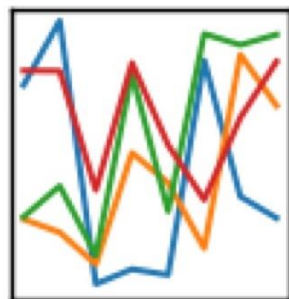
```
file1=open("章节匹配.csv",'r',encoding='utf-8')
file2=open("青鹿章节.csv",'r',encoding='utf-8')
ls1=file1.readlines()
ls2=file2.readlines()
```

```
for line1 in ls1:
    if "无匹配章节" in line1 and "小学" in line1 and "数学"
        ls11 = line1.strip().replace(","," ").split() #当
        str1=ls11[ls11.index("小学",1)-1].strip() #提取学库
        str1_=str(ls11[:ls11.index("小学",1)]).strip('[]')
        ls22A=[]
        ls22B=[]
        ls22C=[]
        for line2 in ls2: #遍历青鹿章节信息
            ls22 = line2.strip().strip(",").split(',')
            if ls22[0] == ls11[0] and ls22[1] == ls11[1]
                if len(ls22) > 5: # 青鹿章节信息组成
                    ls22A.append(ls22[-1])
                    ls22B.append(ls22[-2])
                    ls22C.append(str(ls22).strip())
```



pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



题目难度系数分布



题目数量分布



Machine Learning

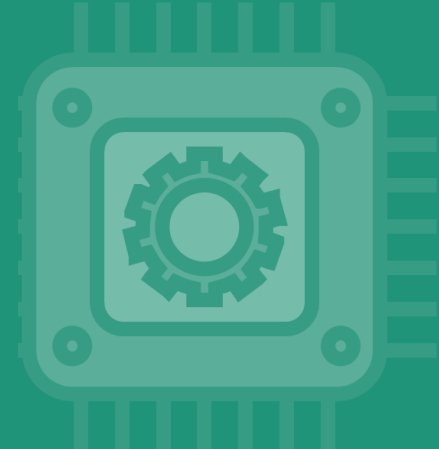
# 机器学习

## 什么是机器学习

What is machine learning?

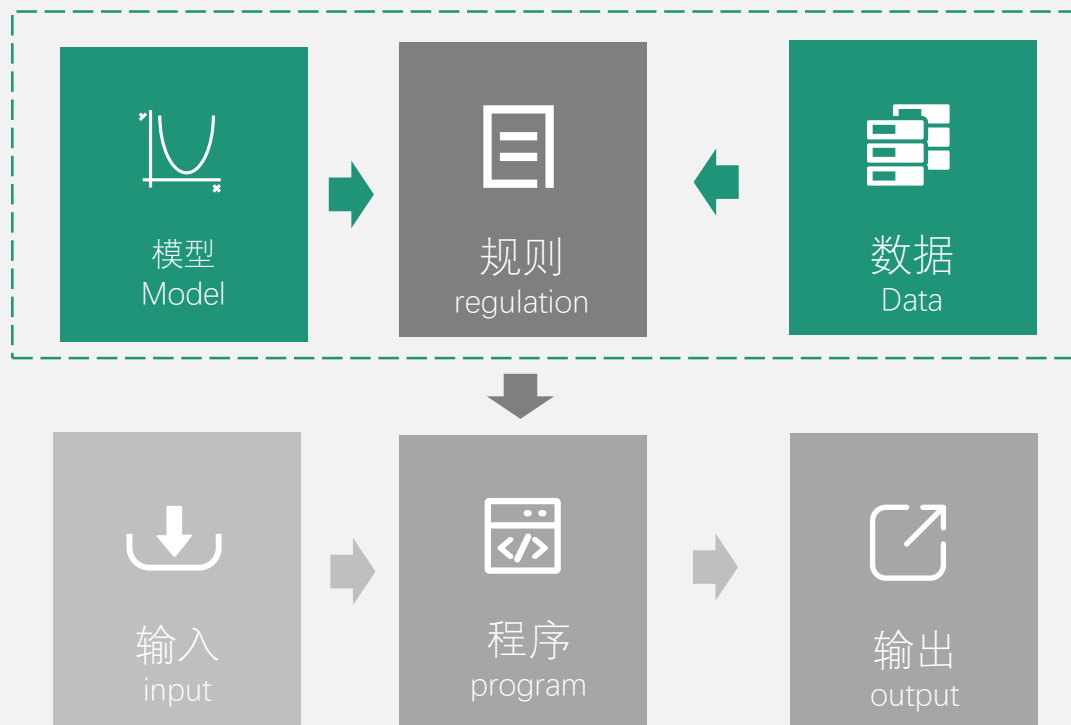
"Machine Learning is the study of computer algorithms that improve automatically through experience.

——Wikipedia



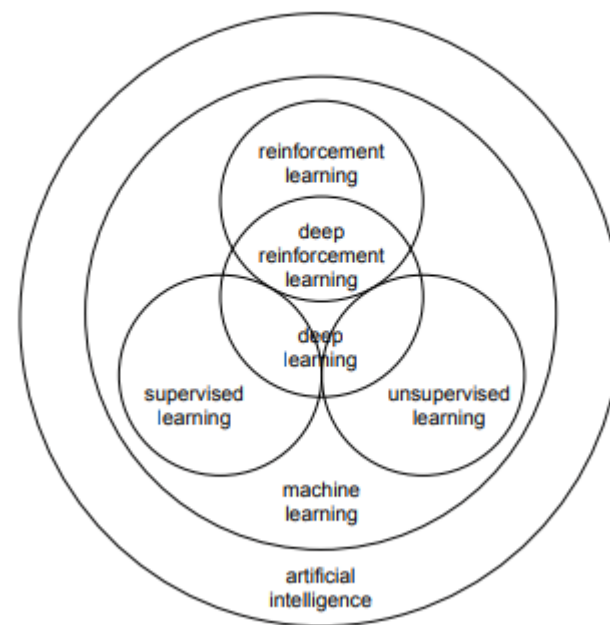
## 机器学习方法与传统程序

Machine learning and the normal programming



## 人工智能、机器学习、深度学习、大数据

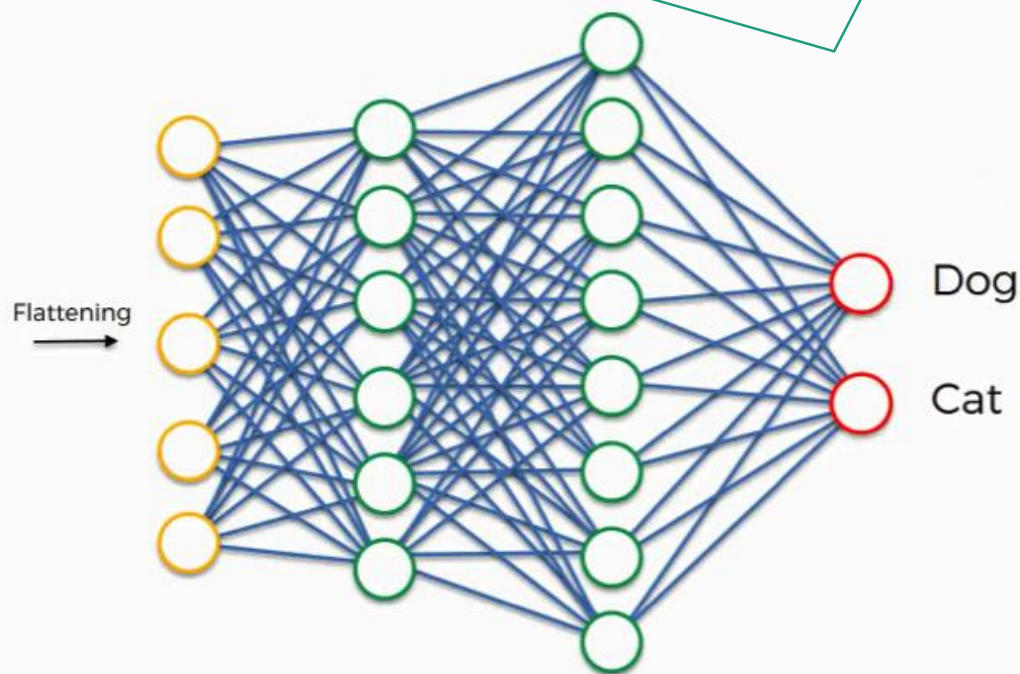
Artificial Intelligence, Machine learning, Deep Learning, Bigdata



# 机器学习中的深度学习

Deep Learning

“利用神经网络模型进行学习过程的机器学习为深度学习”



Full connected neural network  
全连接神经网络



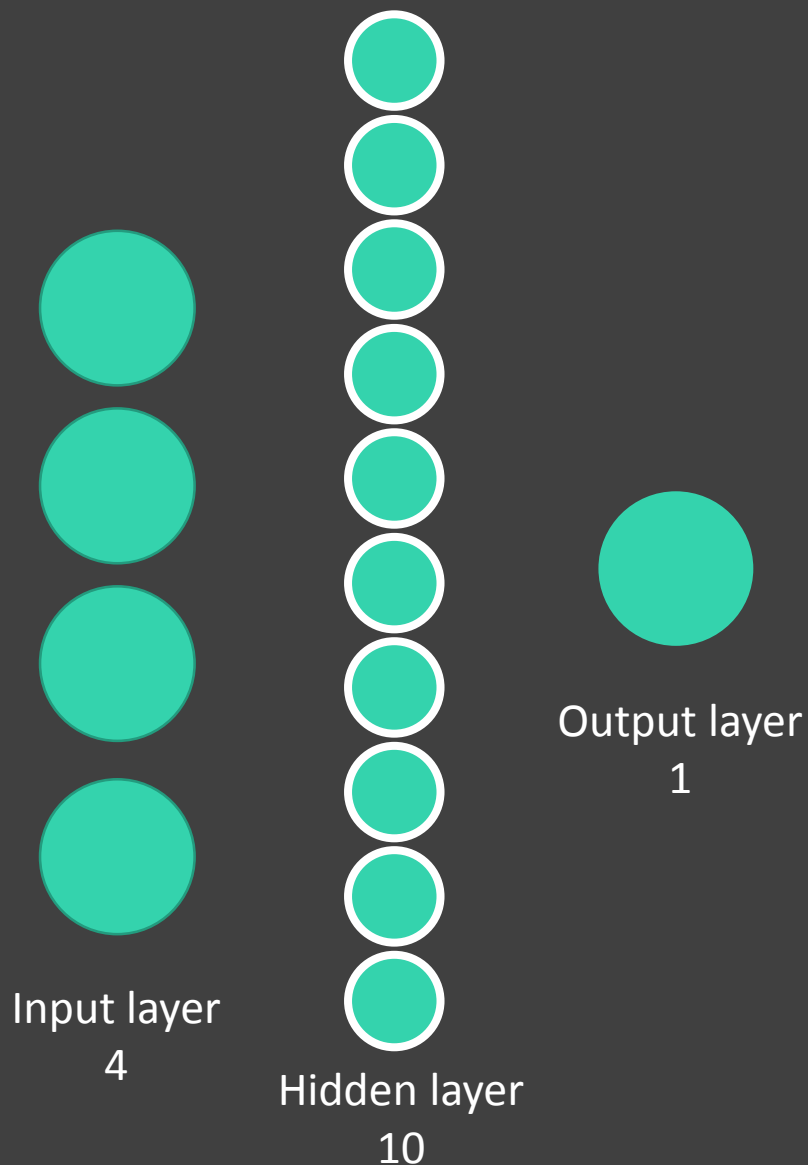
预测小强去不去看电影

Predict the results

如花	小倩	小明	小强
0	0	1	0
1	1	1	1
1	0	1	1
0	1	1	0
1	1	0	?

# 利用Deep Learning预测小强是否去看电影

Using Deep Learning to predict the results



```
from numpy import random,
dot, exp, array
```

#正向推导：根据输入和权重，算出结果

```
def fp(input):
    l1 = 1/(1+exp(-dot(input,
w0)))
    l2 = 1/(1+exp(-dot(l1, w1)))
    return l1, l2
```

#反向传播：用计算结果和实际结果的误差，反向推算权重的调整量

```
def bp(l1, l2, y):
    error = y - l2
    slope = l2 * (1-l2)
    l1_delta = error * slope

    l0_slope = l1 * (1-l1)
    l0_error = l1_delta.dot(w1.T)
    l0_delta = l0_slope * l0_error

    return l0_delta, l1_delta
```

#准备数据: X是输入参数, y是正确结果

```
X = array([[0,0,1],[0,1,1],[1,0,1],[1,1,1]])
y = array([[0,1,1,0]]).T
```

#设置随机的权重

```
random.seed(1)
w0 = random.random((3,10)) * 2 - 1
w1 = random.random((10,1)) * 2 - 1
```

for it in range(10000): #迭代循环

```
l0 = X
l1, l2 = fp(l0) #正向传播计算
```

```
l0_delta, l1_delta = bp(l1, l2, y) #反向传播计算
```

```
w1 = w1 + dot(l1.T, l1_delta) #更新权重
w0 = w0 + dot(l0.T, l0_delta)
```

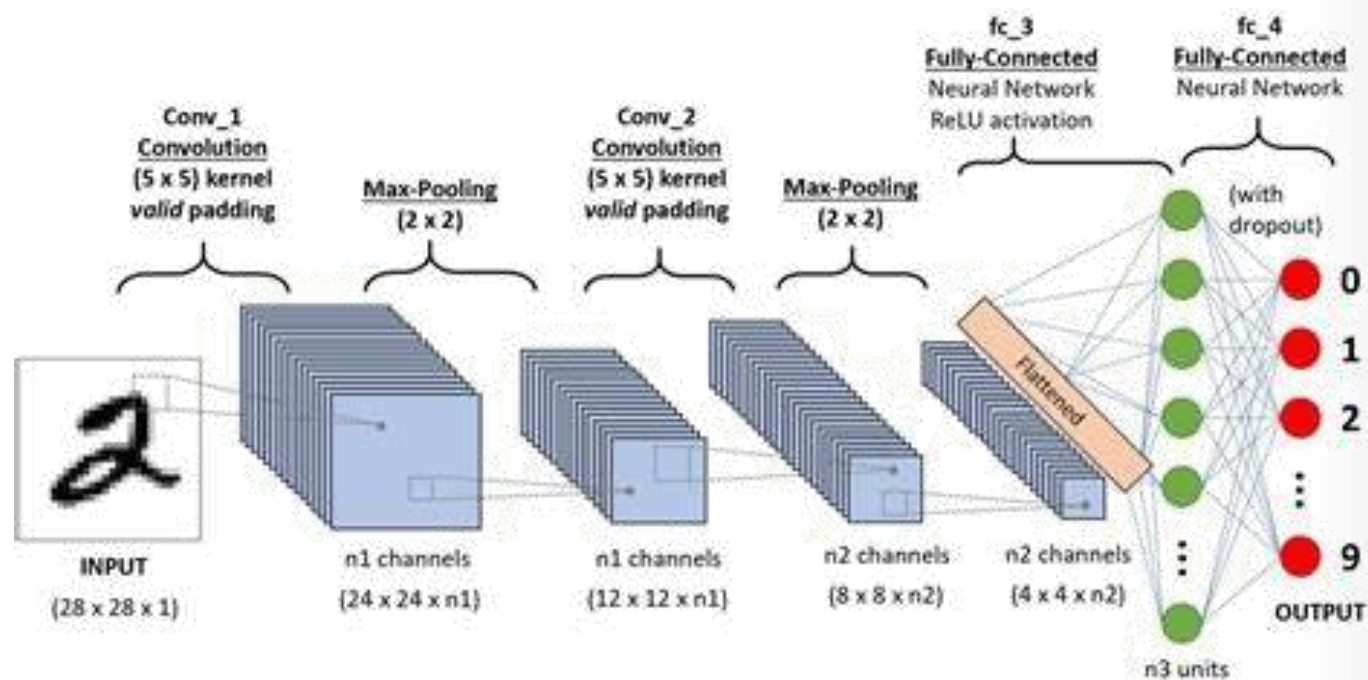
```
Print(fp([[1,0,1]])[1]) #输出结果
```

结果：0.9914



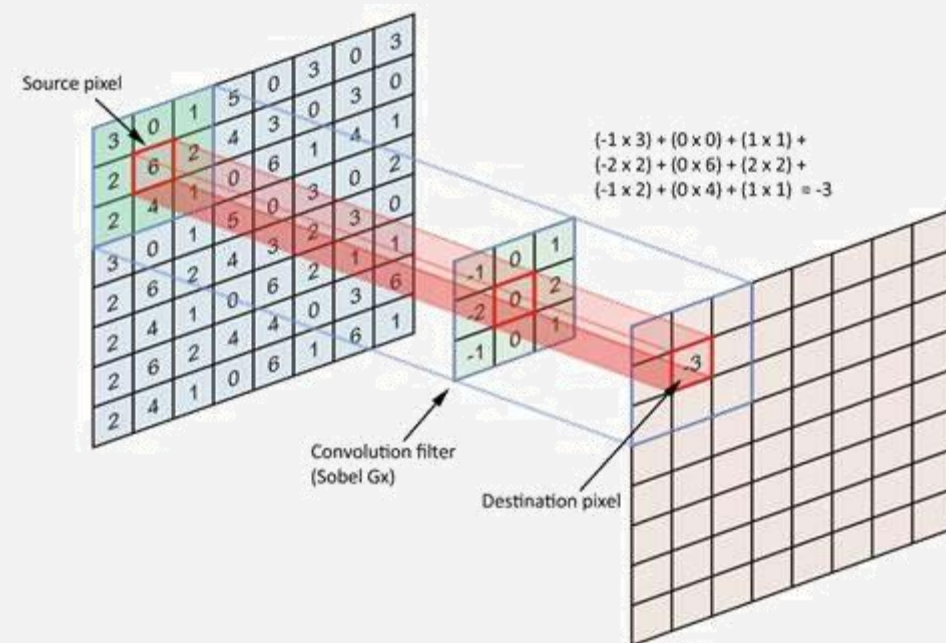
# 卷积神经网络

Convolution Neural Network



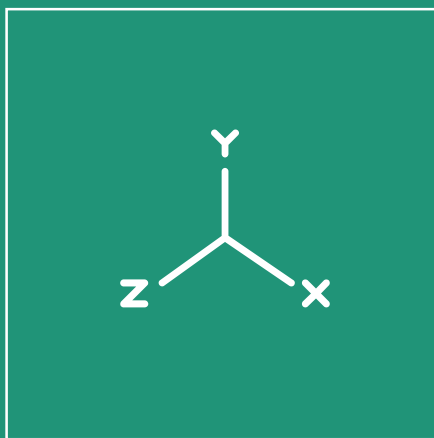
## 卷积核

Convolution Kernel



# 深度学习 资源类产品的应用

Deeping Learning in K12education



利用各种卷积神经网络,如TextCNN等

Using various CNN



智能标注



OCR识别



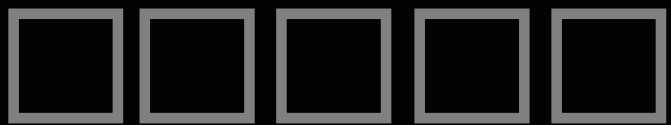
智能推送



自适应学习



知识图谱



# 算法 de 进化

