



尔真 识理 真释 理尔

5.2 探秘人工智能

广州协和学校 信息技术科

2025年4月





尔 识 真 理
真 理 释 尔

从函数到神经网络（10分钟视频）

飞天闪客

飞天闪客 bilibili



尔 识 真 理
真 理 释 尔

人工智能

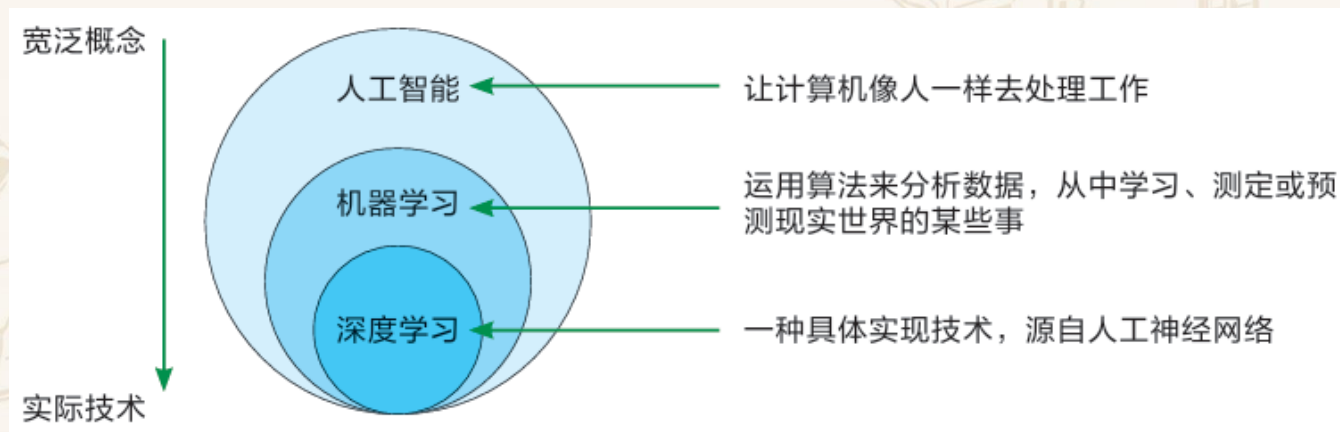
- 人工智能是利用数字计算机或者数字计算机控制的机器**模拟、延伸和扩展人的智能**，感知环境，获取知识并使用知识获得最佳结果的理论、方法、技术及应用系统。
- 弱人工智能
 - 弱人工智能一般指**实现特定功能的专用智能设备**，不能真正实现推理和解决问题。
 - 例如：自动驾驶、车牌识别、刷脸支付
- 强人工智能
 - 强人工智能是指**真正能思考、有知觉、有自我意识**的人类级别的智能机器。
 - 例如：类人智能机器人（目前暂未实现）



尔 识 真 理
真 理 释 尔

机器学习、深度学习

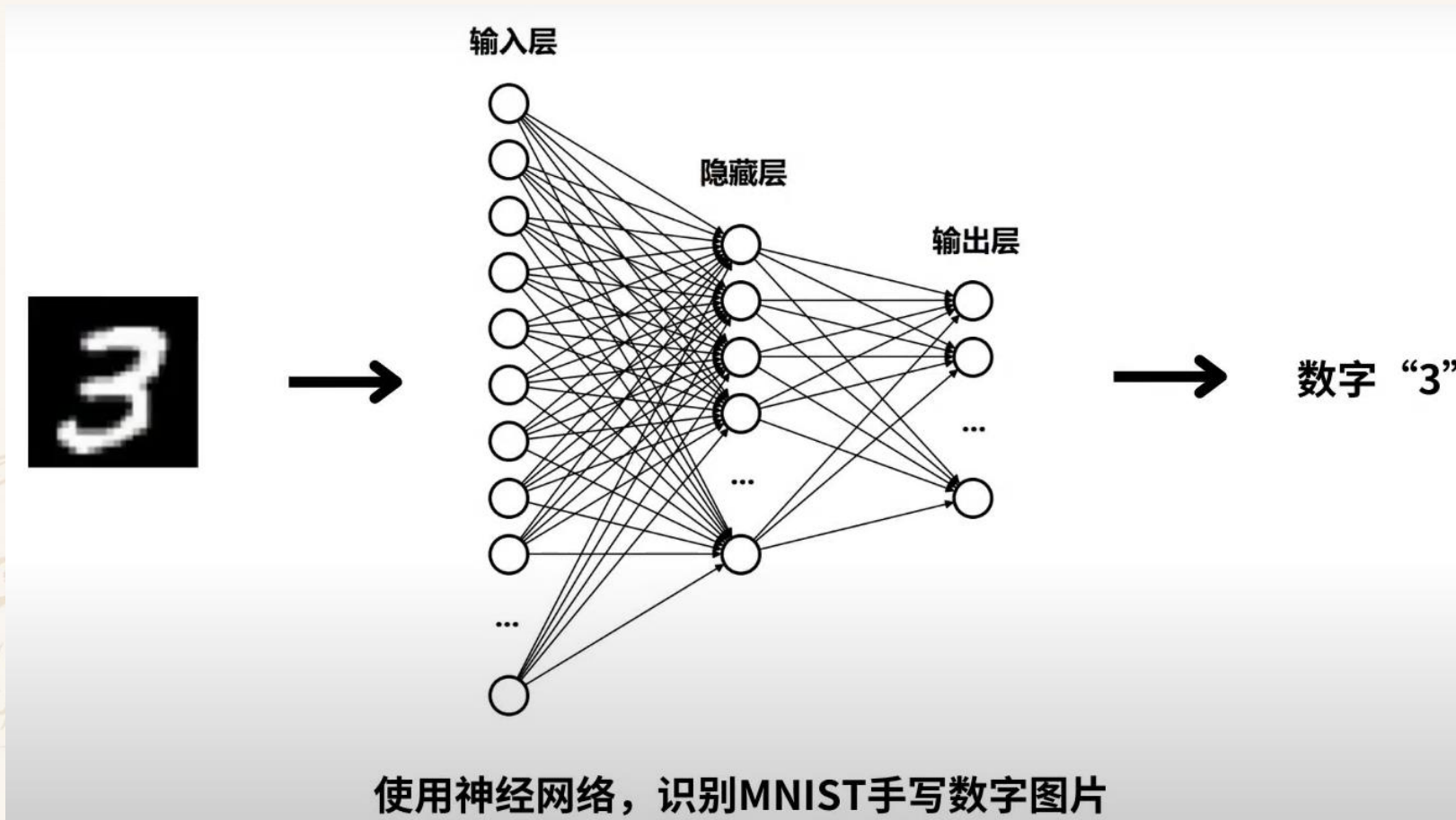
- 机器学习：使计算机能**模拟或实现人类的学习行为**，获取新的知识或技能，重新组织已有的知识结构，从而不断完善自身的性能。
- 深度学习：源自人工神经网络，通过**模拟人脑神经元处理数据的方式**构建多层网络结构（输入层、隐藏层、输出层）。





尔 识 真 理
真 理 释 尔

手写数字识别





尔 识 真 理
真 理 释 尔

手写数字识别

- 其实就是做分类



待分类的图像中，只包括0到9

手写数字识别



图像分类



label = 0

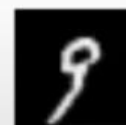


label = 1



label = 2

...



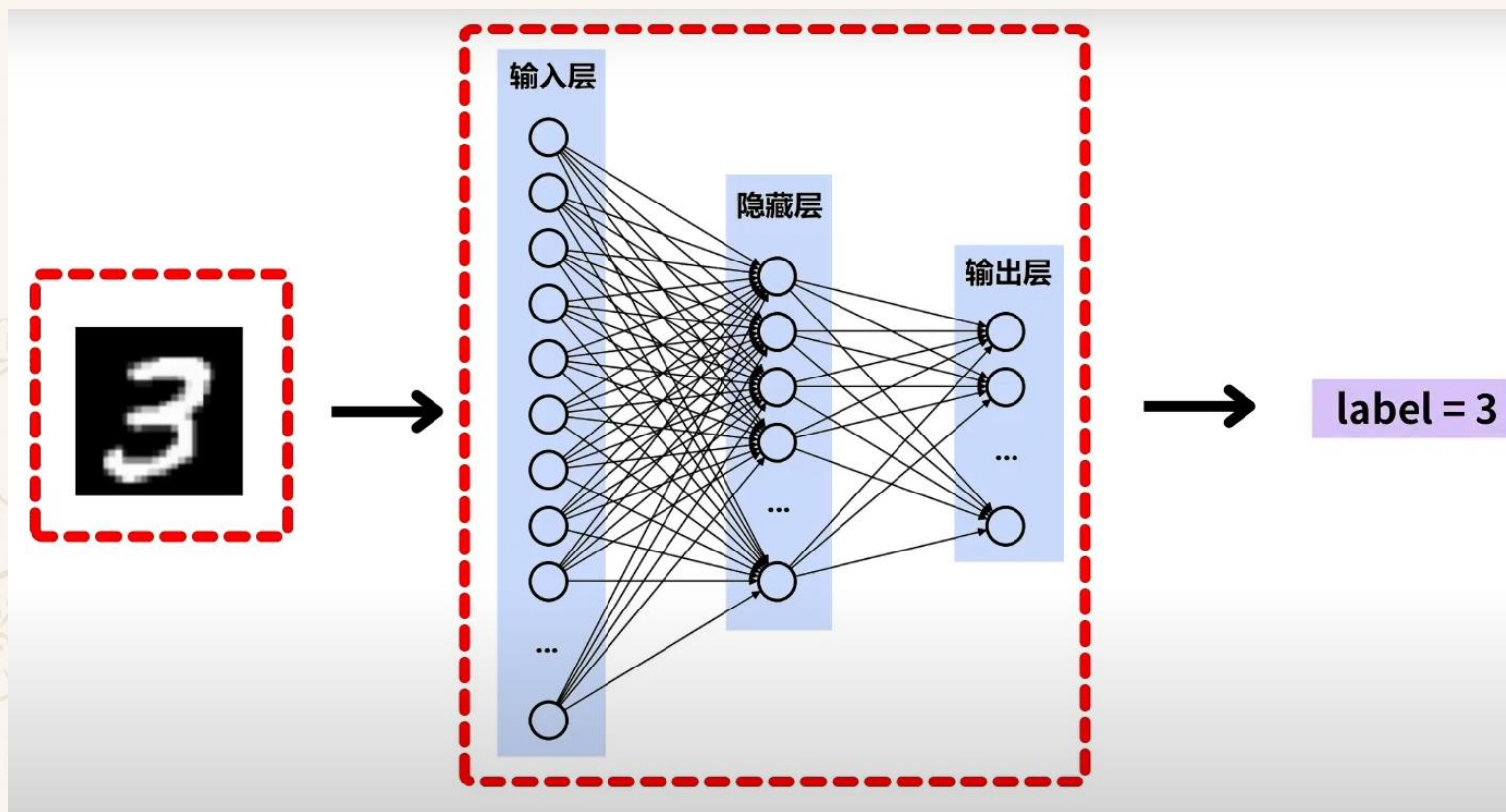
label = 9

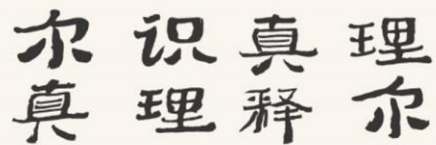


尔 识 真 理
真 理 释 尔

手写数字识别

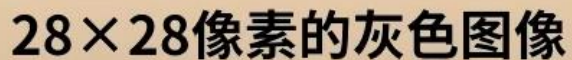
- 例如数字3最后的分类结果就是label=3



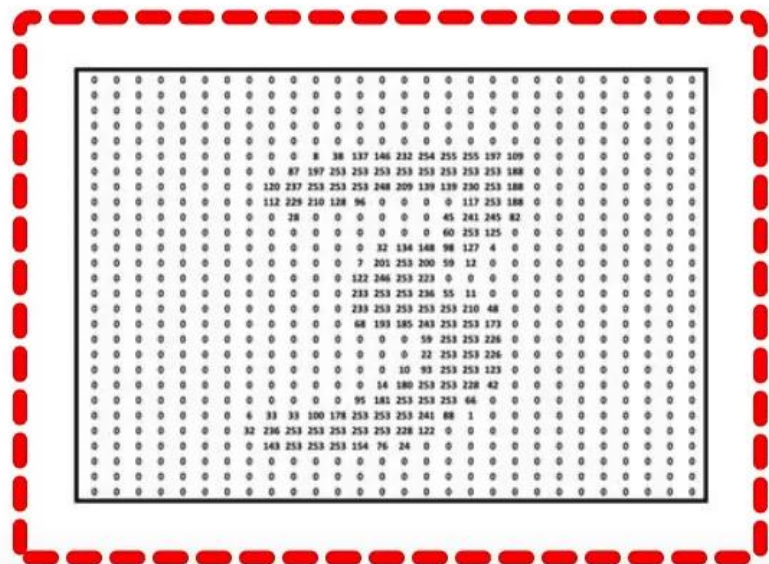


手写数字识别

- 把图片转换为矩阵，矩阵数值为该像素点对应的灰度值。



保存方式



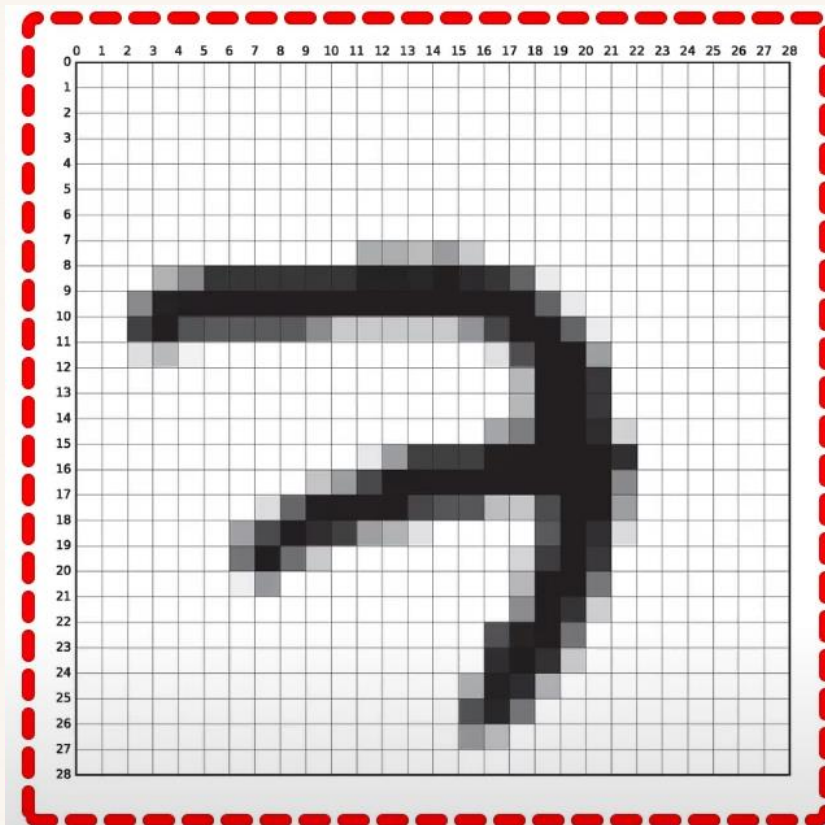
$28 \times 28 = 784$ 个数据点



尔 识 真 理
真 理 释 尔

手写数字识别

- 转换为列向量



$28 \times 28 = 784$

展平



$$\begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_{783} \end{bmatrix}$$

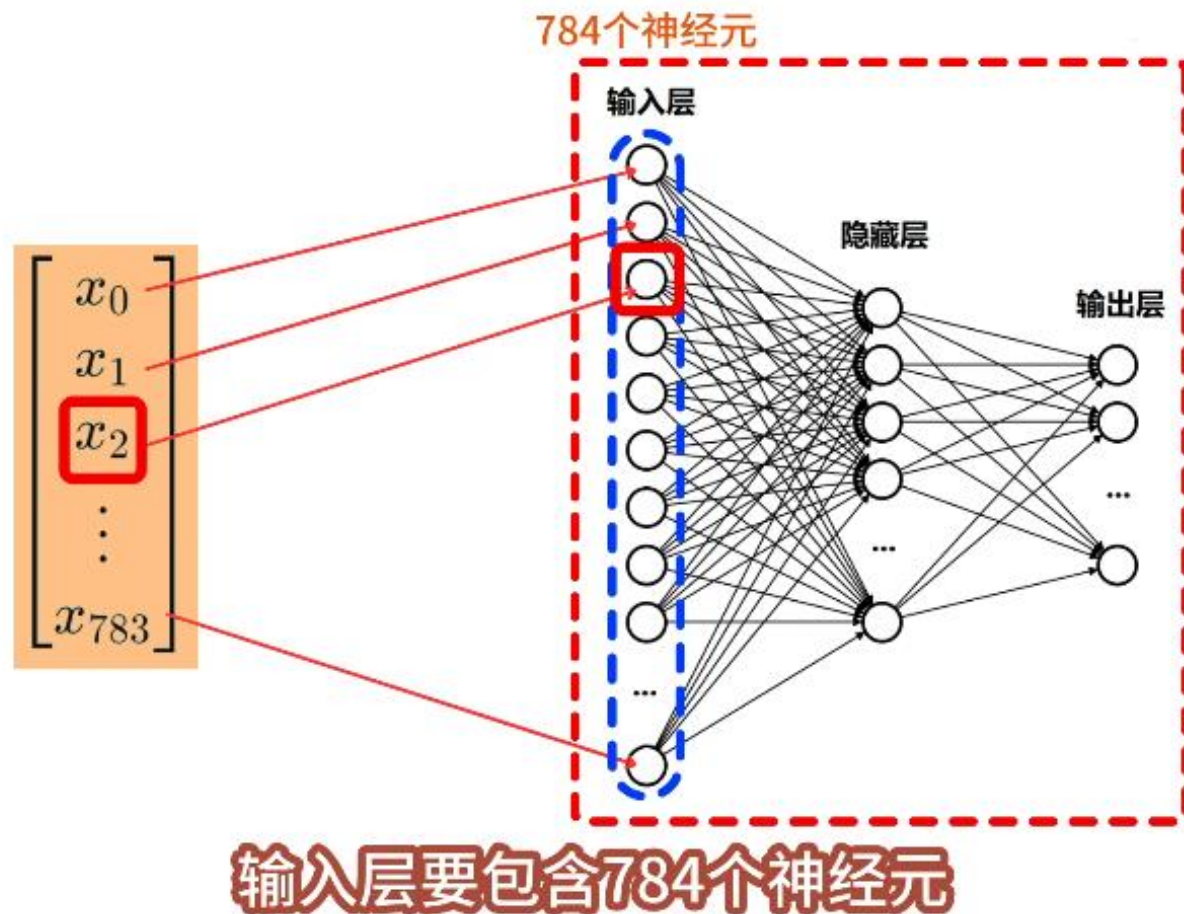
784×1



尔 识 真 理
真 理 释 尔

手写数字识别

- 将列向量输入到输入层

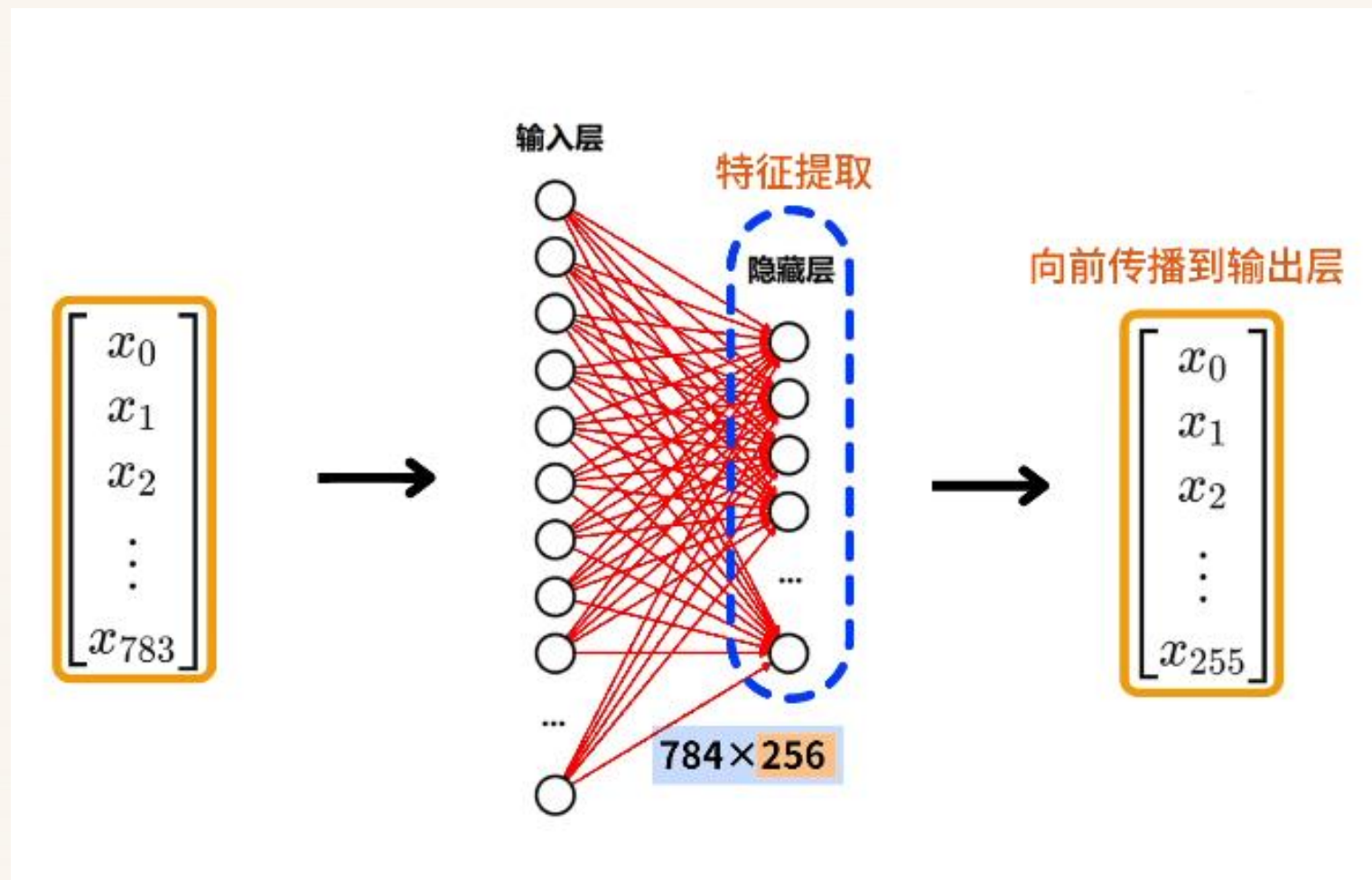




尔识真理
真理释尔

手写数字识别

- 隐藏层用于提取特征

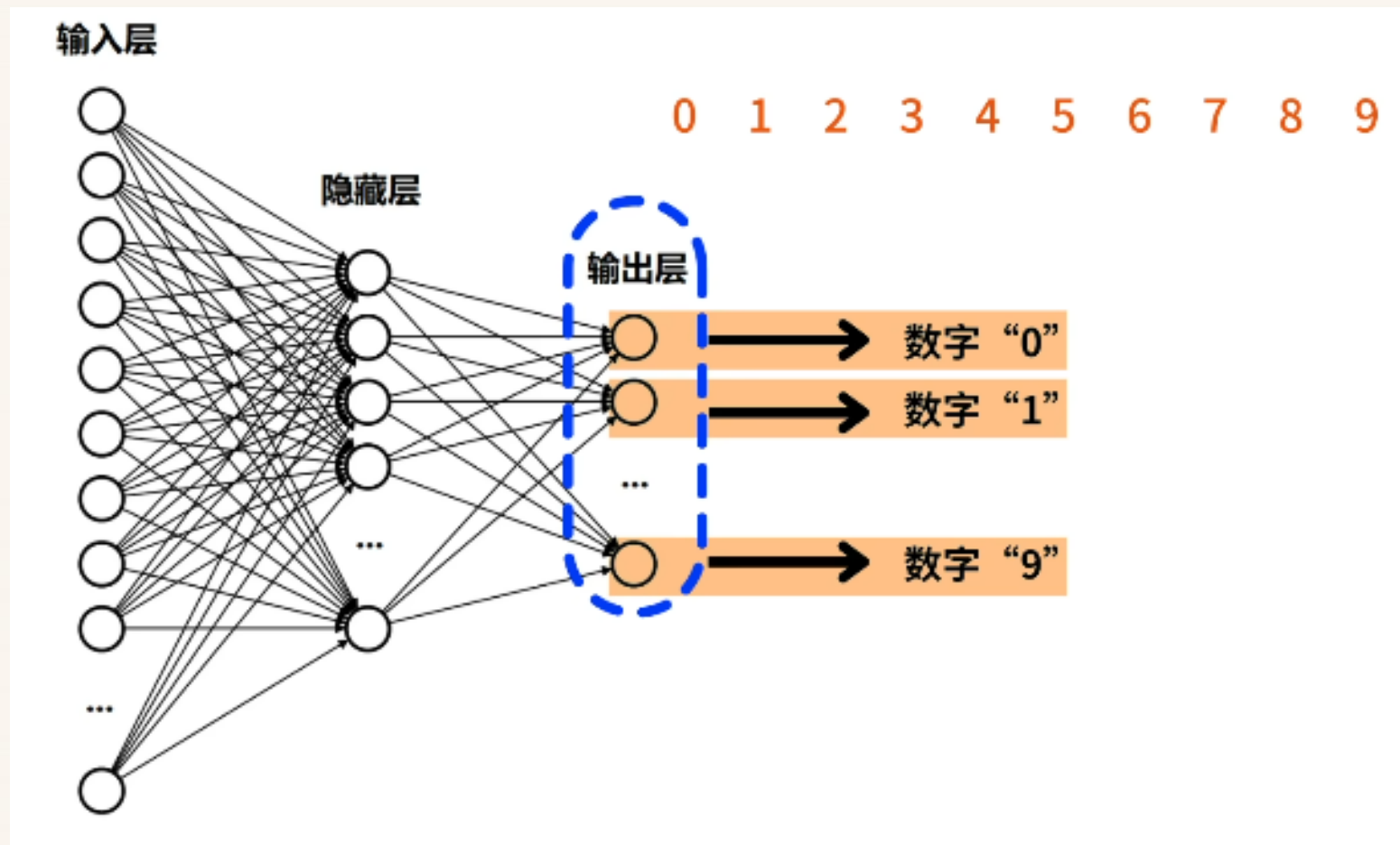




尔 识 真 理
真 理 释 尔

手写数字识别

- 输出层用于预测最终的数字

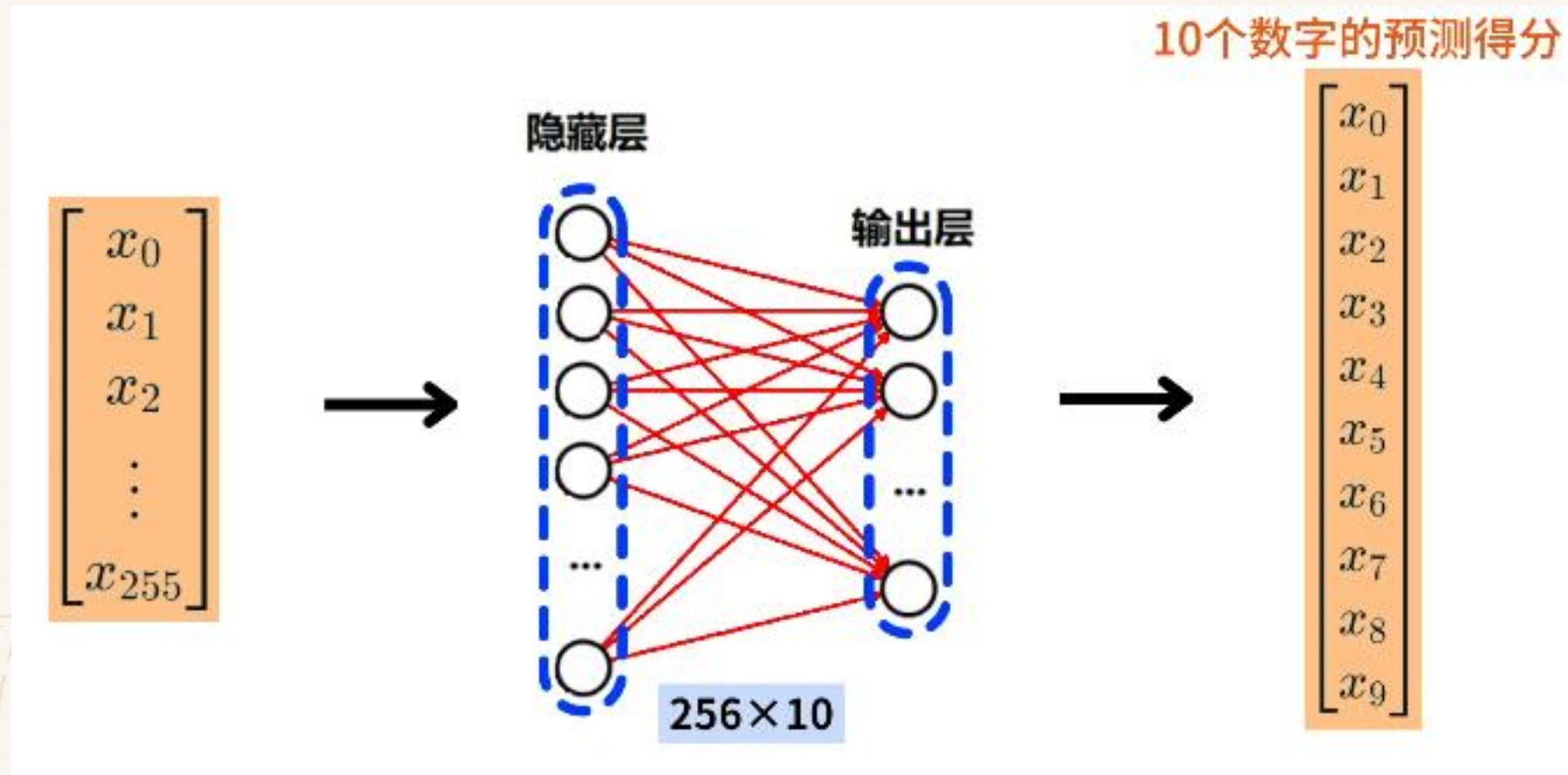




尔 识 真 理
真 理 释 尔

手写数字识别

- 最终得到10个数字的预测得分





尔 识 真 理
真 理 释 尔

手写数字识别

$\begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \end{bmatrix}$

其中哪个数最大，那么它对应的下标就是识别结果：
如果 x_3 最大，那么3就是识别结果

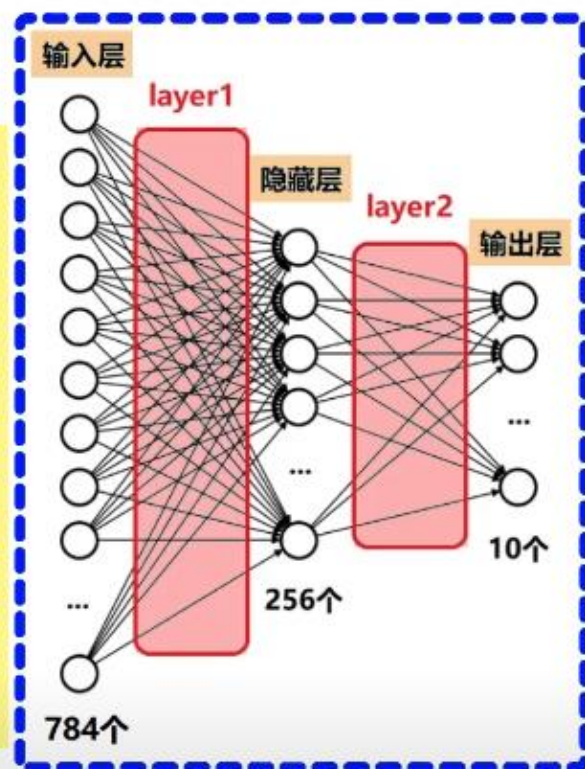


尔识真理
真理释尔

手写数字识别

• 神经网络对应的Python代码

```
1 import torch
2 from torch import nn
3 # 神经网络的实现代码
4 # 实现任何的深度学习模型，最重要的两步骤
5 class Network(nn.Module):
6     def __init__(self):
7         super().__init__()
8         # 线性层1，输入层和隐藏层之间的线性层
9         self.layer1 = nn.Linear(784, 256)
10        # 线性层2，隐藏层和输出层之间的线性层
11        self.layer2 = nn.Linear(256, 10)
12
13        # 在forward函数中，定义该神经网络的推理计算过程
14        # 函数输入张量x，x的尺寸是n×28×28，其中n表示了n张图片
15        def forward(self, x):
16            x = x.view(-1, 28 * 28) # 将向量x展平，尺寸变为n×784
17            x = self.layer1(x) # 将x输入至layer1
18            x = torch.relu(x) # 使用relu激活
19            x = self.layer2(x) # 输入至layer2计算
20            return x # 返回结果
21
22
```





尔 识 真 理
真 理 释 尔

手写数字识别

• 推理过程对应的Python代码

```
22 if __name__ == '__main__':  
23     model = Network() # 定义模型  
24     print(model) # 打印model  
25     print("")  
26  
27     # 定义一个5×28×28的张量x, 表示了5个28×28的图像  
28     x = torch.zeros([5, 28, 28])  
29     output = model(x) # 将x输入至model计算  
30     # 打印output的尺寸  
31     print(f"output.shape: {output.shape}")  
32
```

程序输出:

Network(
35

(layer1): Linear(in_features=784, out_features=256, bias=True)
36

(layer2): Linear(in_features=256, out_features=10, bias=True)
37

)
38

output.shape: torch.Size([5, 10])
39
40

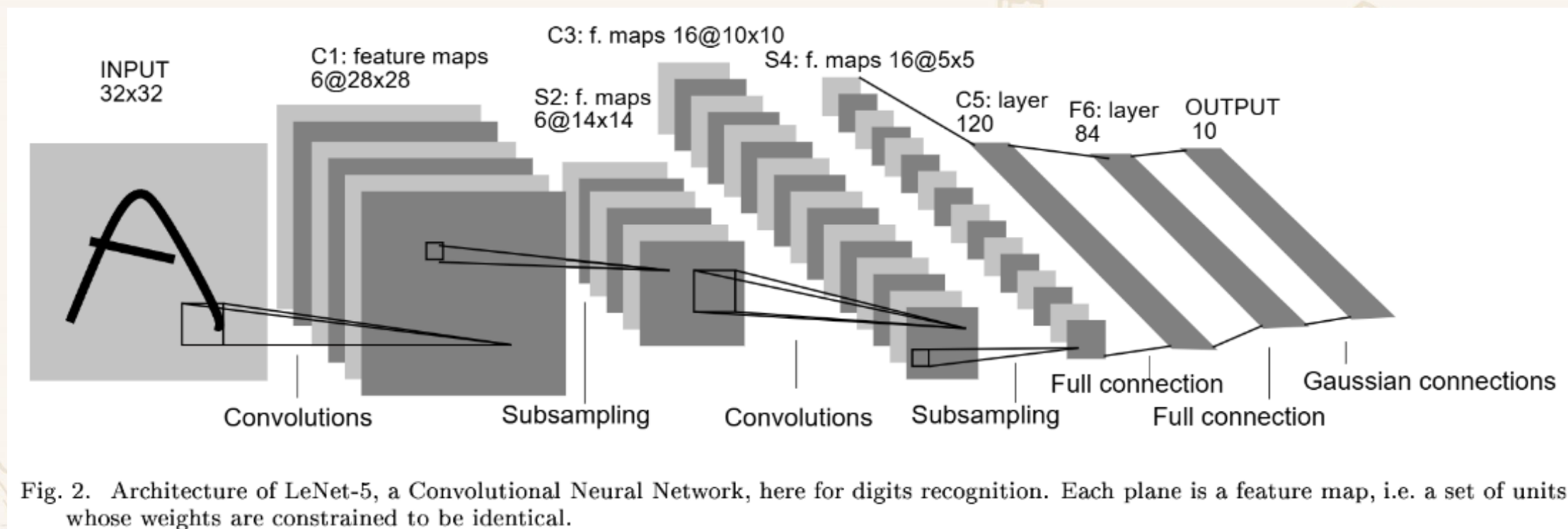
对应了5个图片的推理结果



尔 识 真 理
真 理 释 尔

手写数字识别

- LeNet-5完整网络框架

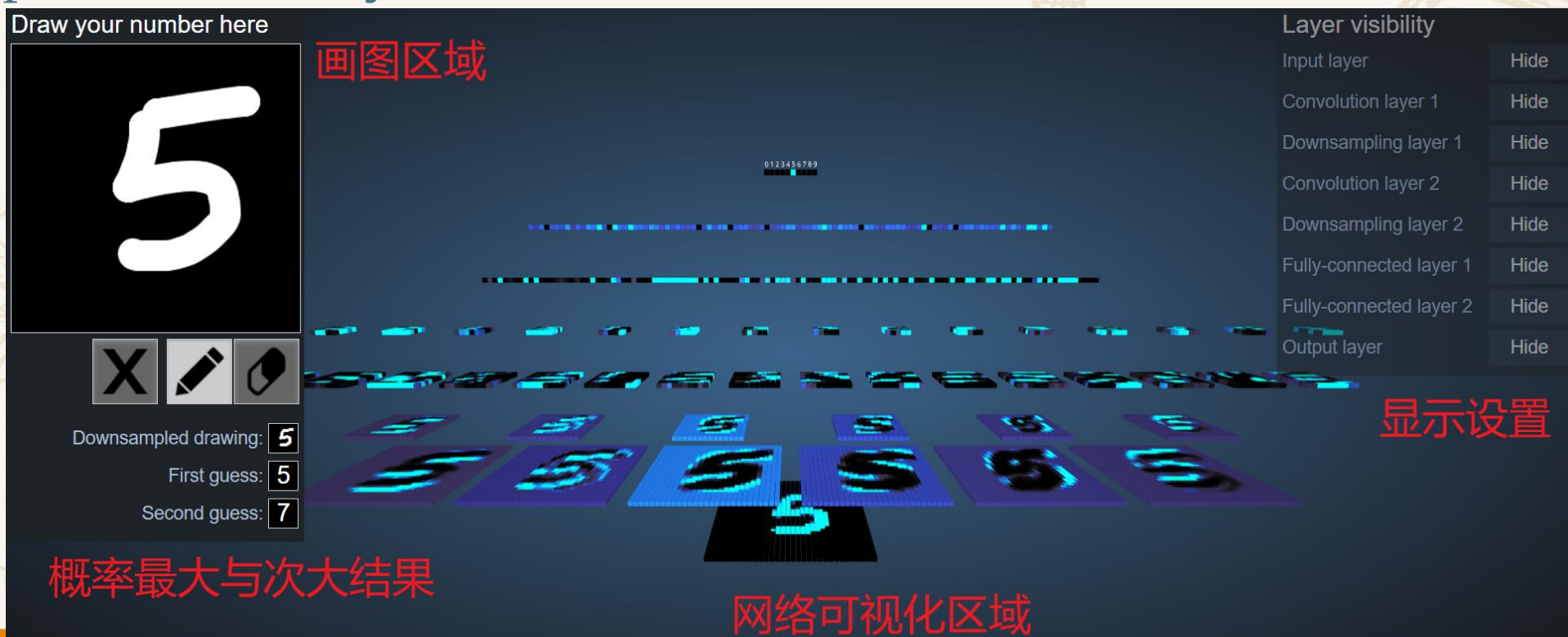




尔识真理
真理释尔

手写数字识别

- LeNet-5手写数字识别可视化网址
- https://adamharley.com/nn_vis/cnn/3d.html

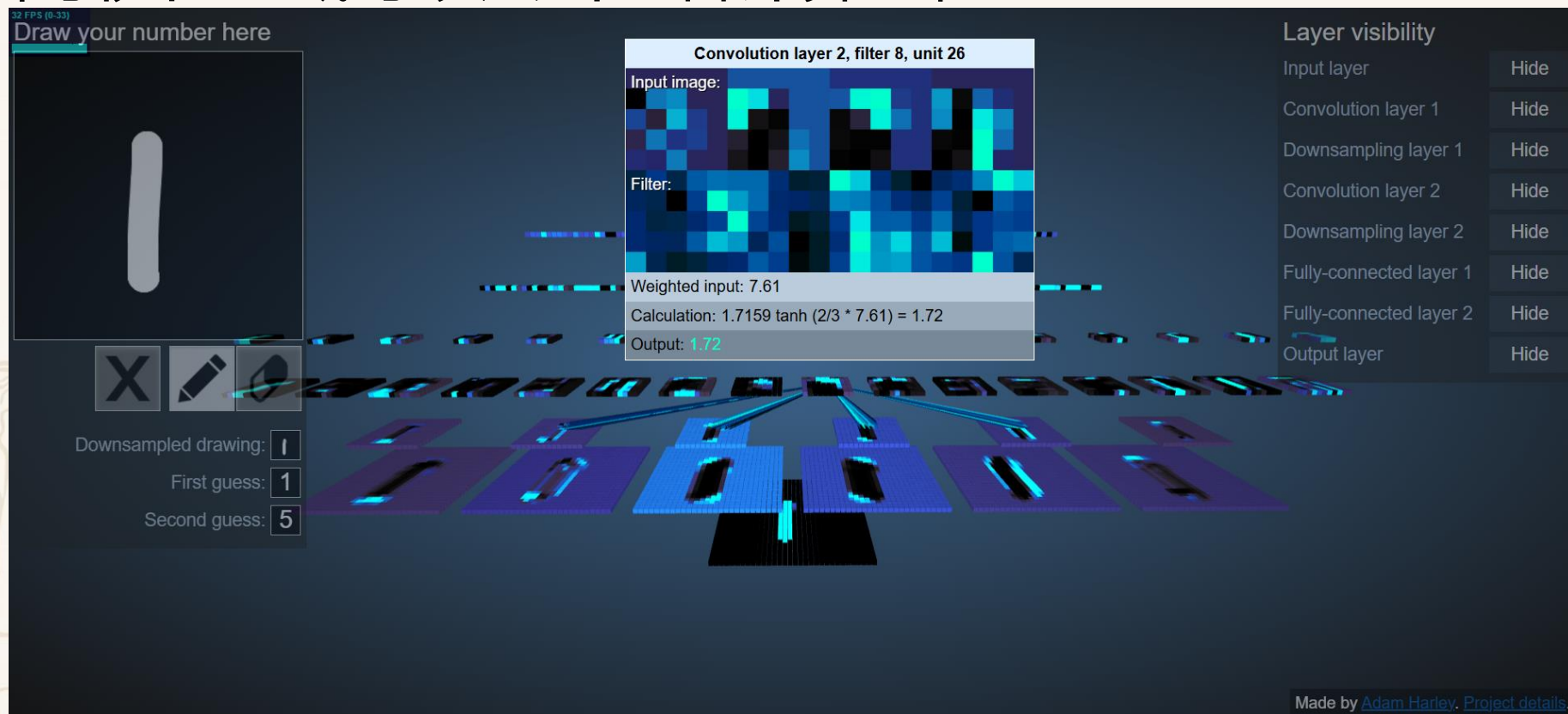




尔 识 真 理
真 理 释 尔

手写数字识别

- 网络可视化区域可以点击查看计算过程





尔识真理
真理释尔

手写数字识别

- 出错案例

Draw your number here



0123456789

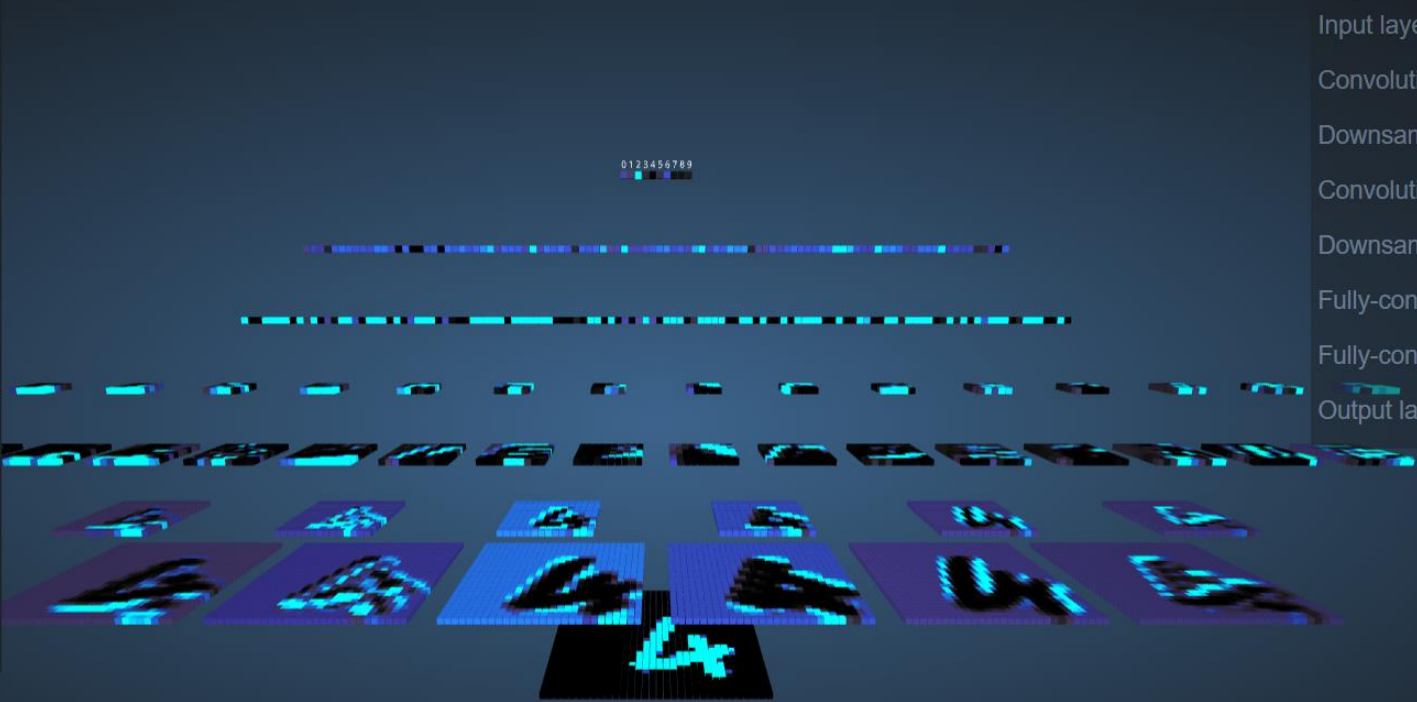
Layer visibility

Input layer	Hide
Convolution layer 1	Hide
Downsampling layer 1	Hide
Convolution layer 2	Hide
Downsampling layer 2	Hide
Fully-connected layer 1	Hide
Fully-connected layer 2	Hide
Output layer	Hide

Downsampled drawing: 4

First guess: 2

Second guess: 6





尔 识 真 理
真 理 释 尔

任务单

- 打开协和OJ，完成作业题《探秘人工智能》，10道选择题（5分钟）
- 打开LeNet-5手写数字识别可视化网址（10分钟）

https://adamharley.com/nn_vis/cnn/3d.html

- 观察LeNet-5的7层结构，思考：
 - 哪几层是卷积层？滤波器尺寸分别是多少？
 - 池化层Downsampling layer使用什么运算？输出尺寸如何变化？
 - 自己手写数字，记录成功识别和失败案例，分析错误原因



尔 识 真 理
真 理 释 尔

拓展：如何计算神经网络的参数（11分钟视频）

飞天闪客