**学习目标**

* 理解赛题背景和赛题数据
* 完成赛题报名和数据下载，理解赛题的解题思路

**了解赛题**

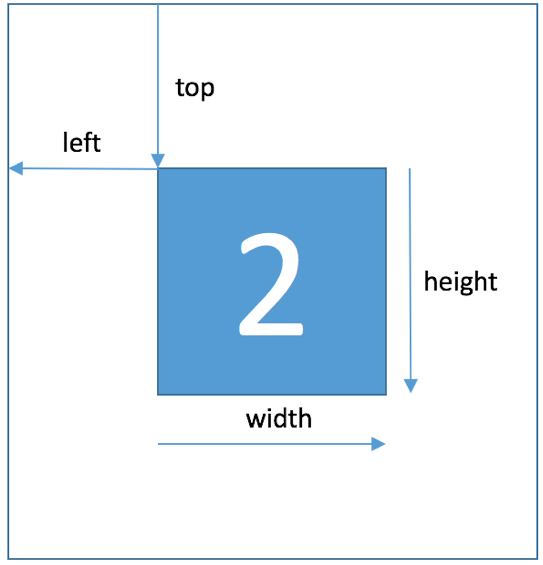
* 赛题数据

赛题以街道字符为为赛题数据，数据集报名后可见并可下载，该数据来自收集的SVHN街道字符，并进行了匿名采样处理。训练集数据包括3w张照片，验证集数据包含1w张照片，每张照片包括颜色图像和对应的编码类别和具体位置；为了保证比赛的公平性，测试集A包括4w张照片，测试集B包含4w张照片。

* 数据标签
* 对于训练数据每张图片将给出对于的编码标签，和具体的字符框的位置（训练集、验证集都给出字符位置），可用于模型训练：



字符坐标具体如下所示：

在比赛数据（训练集和验证集）中，同一张图片中可能包括一个或者多个字符，因此在比赛数据的JSON标注中，会有两个字符的边框信息：



* 评测指标

选手提交结果与实际图片的编码进行对比，以编码整体识别准确率为评价指标。任何一个字符错误都为错误，最终评测指标结果越大越好，具体计算公式如下：  
Score=编码识别正确的数量/测试集图片数量

* 数据读取

JSON中标签的读取方式：

import json

train\_json = json.load(open('../input/train.json'))

# 数据标注处理

def parse\_json(d):

arr = np.array([

d['top'], d['height'], d['left'], d['width'], d['label']

])

arr = arr.astype(int)

return arr

img = cv2.imread('../input/train/000000.png')

arr = parse\_json(train\_json['000000.png'])

plt.figure(figsize=(10, 10))

plt.subplot(1, arr.shape[1]+1, 1)

plt.imshow(img)

plt.xticks([]); plt.yticks([])

for idx in range(arr.shape[1]):

plt.subplot(1, arr.shape[1]+1, idx+2)

plt.imshow(img[arr[0, idx]:arr[0, idx]+arr[1, idx],arr[2, idx]:arr[2, idx]+arr[3, idx]])

plt.title(arr[4, idx])

plt.xticks([]); plt.yticks([])

* 解题思路

一、简单思路：定长字符识别

二、专业字符识别思路：不定长字符识别

三、专业分类思路：检测再识别