

用XEdu做一个趣味甲骨文学习系统

陆雅楠 上海人工智能实验室
谢作如 浙江省温州科技高级中学

摘要: 作者基于甲骨文和AI技术设计并制作了一个互动性强且具趣味性的甲骨文学习系统,从技术分析和实现两个角度呈现了这一项目的开发过程,并介绍了从数据采集、模型训练到模型部署的全过程,给出了将项目转化为课程的设计方案。

关键词: XEdu; 甲骨文; 图像分类; 手写字符识别

中图分类号: G434 **文献标识码:** A **论文编号:** 1674-2117 (2023) 21-0081-03

近年来,与甲骨文相关的小视频曾一度火爆,但这类视频大多是将甲骨文做成动画或表情包,创意比较单一。因此,笔者决定另辟蹊径,做个凸显互动性的学习小游戏,让学生在网页上以“涂鸦”的形式书写甲骨文,系统则给出评价。这样,不仅能给学生提供更加直观、生动的学习体验,也能使其在感受甲骨文独特魅力的同时,体验AI的图像分类技术的神奇。

● 趣味甲骨文学习系统的开发技术分析

从技术角度看,系统的核心是实现甲骨文识别。按照深度学习的一般流程,首先是收集很多甲骨文的图片,形成数据集,然后训练一个图像分类的AI模型。在XEdu工具的图像分类模块的帮助下,训练AI模型用几行代码就能完成,已经

不再是难题。相对而言,难度较高的是数据集的制作,因为在互联网上暂时没有找到能够直接使用的甲骨文字符数据,只能自己以手绘的形式来制作这个数据集。考虑到工作量,笔者仅仅选择了“人”和“大”这两个文字。

采集数据需确保数据的质量和多样性,笔者参照深度学习领域著名的手写数字数据集MNIST,进行了手绘甲骨文字符数据集的采集与整理(如图1)。手绘主要使用电脑自带的画图软件进行字符手绘并保

存为图片,特意使用了多种类型的、不同粗细的画笔工具绘制。每个类别的图片达到500张后,便开始整理数据集为ImageNet格式,即XEdu的图像分类模型支持的数据集格式,合理划分训练集、验证集和测试集。

对于这个支持“涂鸦”识别的学习系统来说,设计一个交互网页反而难度最高。这涉及Web界面的搭建,需要具备前端开发的能力。可喜的是,已经有团队将常见的Web交互功能封装为一个很好

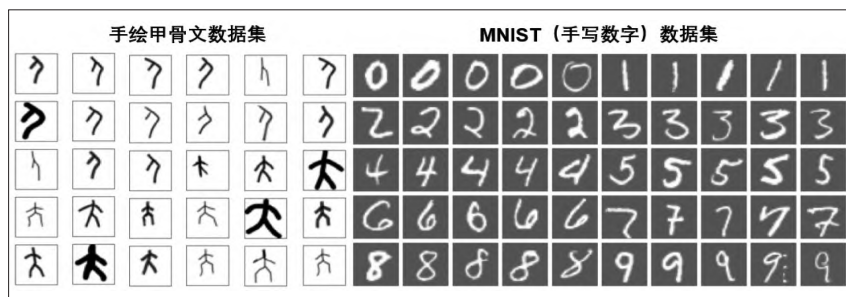


图1 数据集和MNIST对照图

用的Python库,也就是Gradio。借助Gradio只需定义输入和输出接口即可快速构建简单的交互页面,并轻松部署模型,且只需在一行代码中加个参数,即设置“source”为“canvas”,Gradio便可实现“涂鸦”功能,非常方便。

● 趣味甲骨文学习系统的实现

在图像分类模型方面,笔者选择了LeNet网络。因为对于白底黑色字符的甲骨文字符数据集来说,LeNet是非常适合的,速度快且效果好。如果不想写代码,那就用XEdu内置的EasyTrain工具。图2是EasyTrain训练的图示,“loss”表示模型在训练集上的损失值,用以衡量模型预测结果与真实值之间的差异。而“accuracy”代表每轮训练结束后,模型在验证集上的预测准确率。

趣味甲骨文学习系统的交互设计部分,是借助Gradio来实现的。Gradio的核心是Interface类,参考代码如图3所示,通过关联的处理函数“predict”,以及定义“inputs”输入组件类型、“ouputs”输出组件类型,运行代码便可启动一个直观的、用户友好的模型交互界面(如图4)。

● 系统测试与完善

当完成了简易的学习系统后,笔者开始从学生的学习角度思考如何完善这一系统。例如,加入学习游戏的逻辑设置,先自己输入甲骨文

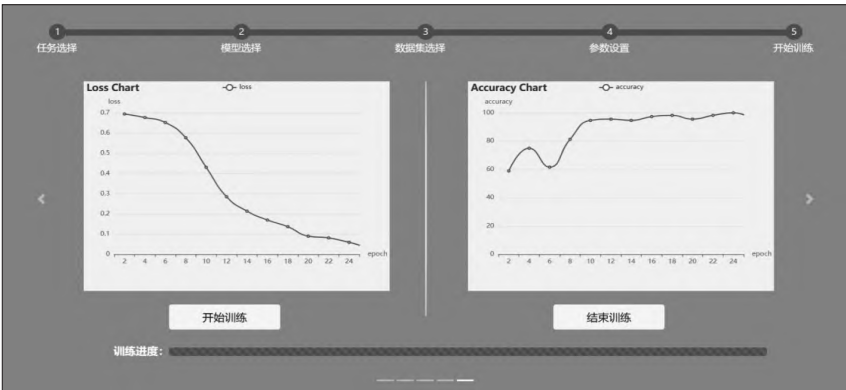


图2 EasyTrain训练界面

```
Python
import gradio as gr
from MMEDu import MMClassification as cls
model = cls(backbone = 'LeNet')
checkpoint='best_accuracy_top-1_epoch_14.pth'

def predict(input_img):
    result = model.inference(image=input_img, show=False,
                             checkpoint=checkpoint)
    result = model.print_result(result, input_img,result)
    return result

demo = gr.Interface(fn=predict, inputs=gr.Image(shape=(128,
128),source="canvas"), outputs=["image","text"])
demo.launch(share=True)
```

图3 参考代码



图4 Gradio的运行界面

绘制目标文字,再进行绘图,提交后传入AI模型进行判断,不仅要判断类别是否准确,还要根据置信度给出评分。同时,还可以给学习系统的界面加入使用说明、计时积分等功能。经过优化后,这个学习系统能够给出比较有趣的提示,交互更加友好,如下页图5所示。

考虑到部署这一模型的计算

机系统可能没有安装XEdu,笔者将模型转换为通用ONNX格式,再借助XEduHub来推理,这样不仅代码更加精简,而且推理速度更快,放在行空板之类的开源硬件上都能运行。

AI模型转换的代码如下页图6所示,转换后部署模型的代码如下页图7所示。完整的项目相关文件可

在以下网址找到: <https://www.openinnolab.org.cn/pjlab/project?id=64faba48929a840fc49ced33&sc=62f34141bf4f550f3e926e0e#public>。

● 从项目设计到课程开发

在成功搭建了这个趣味甲骨文学学习系统后,笔者才意识到这一做法为学生提供了一个有趣的学习途径。因为在制作数据集时,找了好多资料,也学到了很多关于中国古代文字的知识,而这一过程完全可以让来参与,也就是让学生来制作数据集。学生不仅可以自己感兴趣的文字,通过研究和手绘制做出不同的数据集来,还可以通过测试,找出数据集的“瑕疵”,研究更多的甲骨文写法,让数据集更加丰富。

因此,笔者准备以这一项目为基础,开发一个完整的项目式学习课程,引导学生亲身参与从数据收集、模型训练到模型部署的完整过



图5 甲骨文学学习小游戏运行效果

```
Python
from MMEDu import MMClassification as cls
model = cls(backbone='LeNet')
checkpoint = 'best_accuracy_top-1_epoch_11.pth'
out_file='Oracle.onnx'
```

图6 参考代码

```
Python
import gradio as gr
from XEdu.hub import Workflow as wf
my_cls = wf(task='MMEDu', checkpoint='Oracle.onnx')
def predict(input_img):
    result, image = my_cls.inference(data=input_img, img_type='cv2')
    re = my_cls.format_output(lang="zh")
    return image, re
demo = gr.Interface(fn=predict, inputs=gr.Image(shape=(128, 128), source="canvas"), outputs=["image", "text"])
demo.launch(share=True)
```

图7 参考代码

程。项目计划6课时,配合课程设计可将此项目分成大任务和子任务,大任务是完成一个趣味甲骨文学学习小游戏,小任务则围绕项目创作的流程进行拆分,其中包括主题选

择、创意构思、数据准备、模型训练、游戏开发、测试优化等。

参考文献:

[1]汤欣雯.中国传统文化主题的AI艺术创作与效果研究——以人工智能综艺节目《机智过人》为例[J].视听, 2022(11):40—43.

[2]张楠.AI促进传统文化的创造性转化[J].软件和集成电路, 2023(09):17.

[3]谢作如.借助Gradio制作AI体验活动的教学课件[J].中国信息技术教育, 2023(13):85—87. 