**基于深度学习技术的甲骨文自动化识别**

吕肇辰

# 摘 要

摘要正文300字左右，五号宋体，首行缩进2个汉字符，行距18磅

**关键词：**关键词3~5个，五号宋体，逗号分开，最后一个关键词后面无标点符号

目 录

[摘 要 1](#_Toc79852424)

[1 介绍 3](#_Toc79852425)

[1.1. 背景和意义 3](#_Toc79852426)

[1.2. 相关工作 3](#_Toc79852429)

[1.1.1. 基于图像处理的方法 3](#_Toc79852430)

[1.1.2. 基于深度学习的方法 4](#_Toc79852431)

[1.3. 创新性及实施步骤 4](#_Toc79852433)

[2 数据描述 4](#_Toc79852434)

[2.1. 数据集描述 4](#_Toc79852436)

[2.1.1. IsOBS[9] 4](#_Toc79852437)

[2.1.2. HWOBC[12] 5](#_Toc79852438)

[2.1.3. OBC306[12] 6](#_Toc79852439)

[2.1.4. OBIDataset[] 7](#_Toc79852440)

[2.2. 局限性和挑战 7](#_Toc79852441)

[2.2.1. 长尾效应[10] 7](#_Toc79852442)

[2.2.2. 编号不统一 7](#_Toc79852443)

[2.2.3. 噪音和残缺 8](#_Toc79852444)

[2.2.4. 异体字[15] 8](#_Toc79852445)

[3 数据扩增 8](#_Toc79852446)

[3.1. 数据集融合 8](#_Toc79852448)

[3.1.1. 重要性和挑战 8](#_Toc79852449)

[3.1.2. 思路 9](#_Toc79852450)

[3.1.3. 实施步骤 10](#_Toc79852451)

[3.2. 数据扩增 12](#_Toc79852452)

[4 模型搭建 13](#_Toc79852456)

[4.1. 本地环境 13](#_Toc79852462)

[4.2. 选择AlexNet的原因 13](#_Toc79852463)

[4.2.1. AlexNet与ResNet50的比较 13](#_Toc79852464)

[4.3. 神经网络结构的改进 14](#_Toc79852465)

[4.3.1. 跳跃连接 (Skip Connection) 14](#_Toc79852466)

[4.3.2. 批标准化 (Batch Normalization) 15](#_Toc79852467)

[4.4. 利用Optuna进行超参数优化 15](#_Toc79852468)

[4.4.1. 超参数优化的方法 15](#_Toc79852469)

[4.4.2. 超参数优化的实施步骤 16](#_Toc79852470)

[4.4.3. 超参数优化实验结果 18](#_Toc79852471)

[4.5. 甲骨文定位 19](#_Toc79852472)

[5 结果分析 19](#_Toc79852473)

[6 总结 20](#_Toc79852474)

[参考文献 20](#_Toc79852475)

[谢辞 23](#_Toc79852476)

# 1 介绍

* 1. 背景和意义

早在新石器时代，中原地区就已经形成了文字的雏形。成熟汉字约形成于夏商之际，而甲骨文是我国目前已知最早的、成熟的文字系统，是汉字的童年时期。虽然甲骨文中还残留一些早期文字不规范、不严密、不准确的地方，但已经属于可以严格记录语言的真正意义上的文字。

甲骨文最早于 1899 年由晚清官员王懿荣在河南省安阳市出土的甲骨上发现。早期甲骨文主要来源于河南省安阳市西北殷都区小屯村，是中国商朝晚期都城遗址“殷墟”的所在地。殷墟甲骨出土数量众多，祭祀占卜的记录内容多样，有祭祀先王、神邸、天地，也有占卜天气天象、农事、疾病、祸福等[1]。殷墟甲骨给我们留下了研究商文化的重要资料，其所载殷 商时期祭祀占卜记录更是对当时整个社会产生了深远影响[1]。因此，甲骨文对于中国历史有着非常重要的意义。

甲骨文对于汉语文字的演变的研究也起着不可忽略的作用。甲骨文总字头数已经达到4500多个[2]，已经可以自由记录汉语。甲骨文中常用字的结构已经基本定型，这些字经过几千年演变，虽然笔势笔意屡有变迁，但其基本结构没有变，脉络清晰，层次分明。

甲骨文的发现是中国历史研究的一件大事，同时鉴于甲骨文在世界文明史上的地位，也是世界历史研究中的重要事件。甲骨文是迄今为止中国发现的年代最早的成熟文字系统，是汉字的源头和中华优秀传统文化的根脉，值得倍加珍视、更好传承发展。2019年，国家主席习近平致信祝贺甲骨文发现和研究120周年指出：“殷墟甲骨文的重大发现在中华文明乃至人类文明发展史上具有划时代的意义。”[3] 研究甲骨文，对追寻中华民族先民顽强生存发展的历史轨迹有深远的意义；对研究中国文字的产生、发展，进而研究中华文化的产生与发展有着极其重要的历史价值。

虽然已发掘的甲骨数量已达到10万片左右[4]，但甲骨文破译工作十分困难，目前已破译出的甲骨文字仅有2000多个[2]。传统的甲骨文字检测、识别方法需要甲骨学专家手动查阅大量资料并结合自身经验才可以完成，十分耗时耗力，效率低下而成本极高，这是甲骨学亟待解决的重要问题。因此，甲骨文破译前景仍然十分广阔。

在最近机器学习等技术迅猛发展的情况下，使用计算机实现甲骨文字检测、识别自动化成为甲骨文破译极具潜力的一种解决方案。

在这样的条件下，我将利用深度学习技术实现甲骨文的高精度自动识别。

1. 2. 相关工作
      1. 基于图像处理的方法

顾绍通等[5]分析了甲骨文字形的拓扑结构，实现了对甲骨文字形的拓扑描述。在此基础上，利用拓扑配准的方法，通过计算基准拓扑与待配准拓扑之间的欧氏距离，实现基于拓扑结构的甲骨文字形的配准，从而识别甲骨文字形。

鲁绪正等[6]建立了一个基于Capsule 网络和迁移学习的模型 OracleNet，通过甲骨文构件数据集对其进行训练，可以同时对甲骨文字形中多个构件同时进行识别，对甲骨文构件的预测精确度达到了 60% 以上，其中 Top5 精确度达到了 71.56%。

* + 1. 基于深度学习的方法

Yi-Kang Zhang等[7]在卷积神经网络(CNN，Convolutional Neural Network)的基础上采用最近邻居(Nearest Neighbor, NN)的规则实施文字识别，实现驳回未知类别(unseen categories)并接受已知类别，准确度达到92.43%。

Shanxiong Chen等[8]使用多区域卷积神经网络(Multiregional Convolutional Neural Network)识别甲片上的盾状纹路(shield grain)和齿状纹路(tooth grain)以便将整片甲骨的图片分割为多个部分，再通过卷积神经网络识别并合并每个区域的图像特征，最终在AlexNet上达到了84.3%的精确度。

Xu Han等[9]创造了名为IsOBS的甲骨文字数据库，并对存入的文字进行编码和整理，并对比原形网络(Prototypical Network)、关系网络(Relation Network)和孪生神经网络(Siamese Network)，选择最适合甲骨文文字识别的原形网络在该数据集上进行试验，在oracle600数据集上达到了90%左右的精确度。

王浩彬[10]构建的甲骨文检测数据集 OBCD (Oracle Bone Character Detection) 为深度学习奇数应用于甲骨字符检测工作提供了基准数据库，并使用基于区域的全卷积网络 (R-FCN, Region-based Fully Convolutional Network)算法，结合特征金字塔网络( FPN, Feature Pyramid Network)这种网络结构的多尺度设计，搭建了基础的甲骨字符检测算法框架，最终F-measure达到83.79%。在甲骨文字识别领域，他创建了OBC306数据集，并提出一种基于循环式生成对抗模型(CycleGAN, Cycle Generative Adversarial Network)的数据扩充算法，解决长尾效应(在2.2.1中将会提到)，并在OBC306上取得了86.54%的平均识别精度。

* 2. 创新性及实施步骤

首先，我选用IsOBS, HWOBC, OBCD等甲骨文数据集进行融合和训练，并通过数据扩增(Data Augmentation)技术进一步增加机器学习的训练样本数量。然后，我利用基于深度学习的物体检测(Object Detection)技术，实现文字分类。接下来，我会基于经典的卷积神经网络模型，例如AlexNet，ResNet等，提出适合甲骨文内容识别的分类算法，并对模型进行调优，以提高识别精度。最后，我还将尝试利用上下文的语境对所识别的文字进行校准。

本文的创新点包括：利用深度学习技术建立了一种快速，精准的融合不兼容的数据集以便训练的方法，使用卷积神经网络对经过融合和数据扩增后的甲骨文字进行识别和分类，利用超参优化的方法提高模型精度，并利用对象分割(Instance Segmentation)对带有噪音、残缺的甲片上的甲骨文字进行定位。

# 数据描述

1. 1. 数据集描述
      1. IsOBS[9]

IsOBS取自分为三个数据集，如下图所示。图片来源于来自中国社会科学院整理的甲骨文和标准化字体[11]，以及甲骨文及其对应的现代汉字变体(字形)集。每个数据集被3:1至4:1的比例分离出测试数据。总计有属于1621个类的20420张图片，大小为110\*200，为纯白的背景上的清晰字形。共111 MB训练数据及26.2 MB测试数据。

|  |  |  |  |
| --- | --- | --- | --- |
| 数据集名称 | 类别个数 | 样本数量 | 描述 |
| oracle300 | 353 | 11586 | 每个类别有多于20个样例 |
| oracle600 | 617 | 15638 | 每个类别有多于12个样例 |
| oracle1600 | 1621 | 20420 | 每个类别有多于2个样例 |

图 1： IsOBS数据集下属的三个小数据集的特征描述[9]

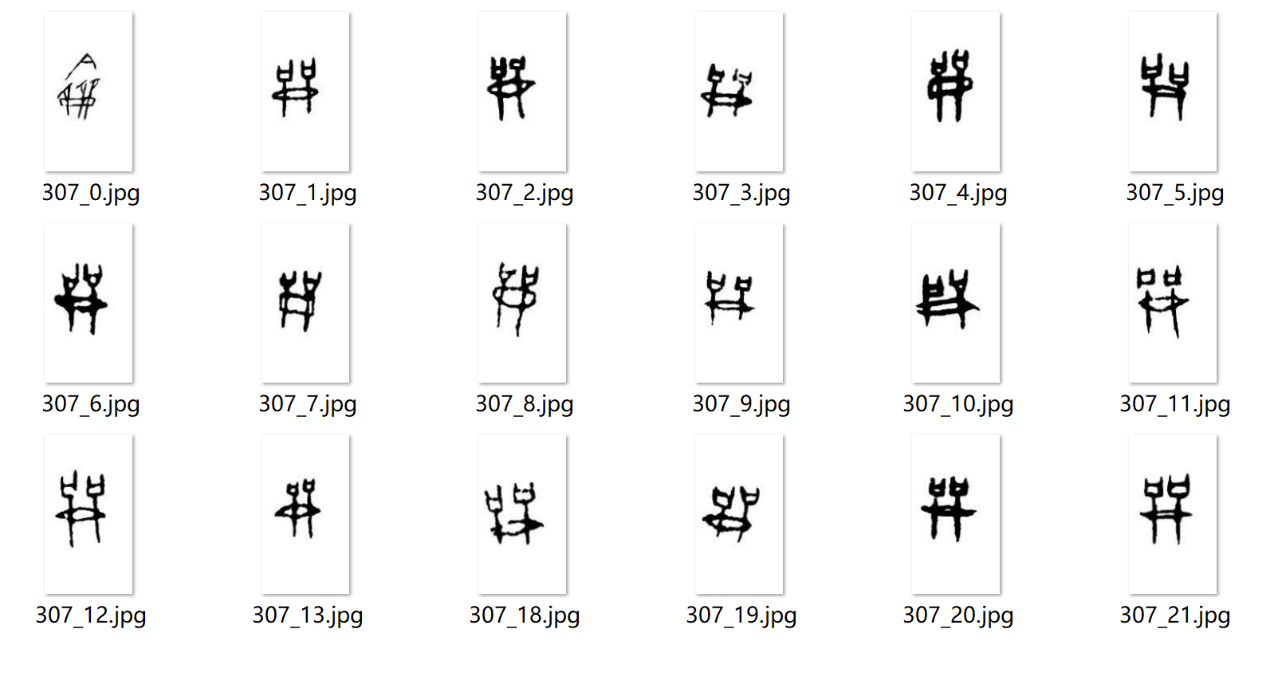
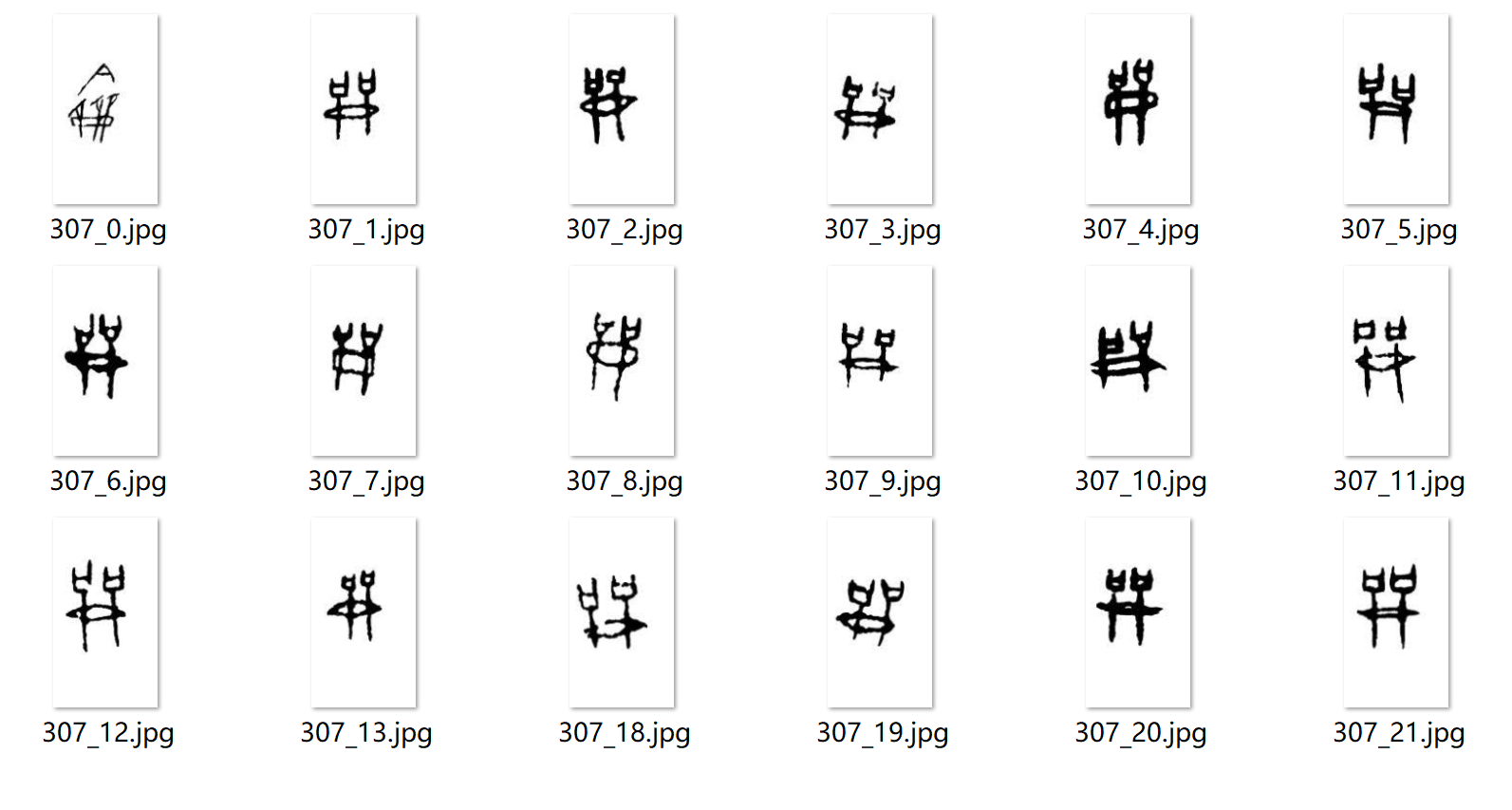


图 2：oracle1600数据集示例



* + 1. HWOBC[12]

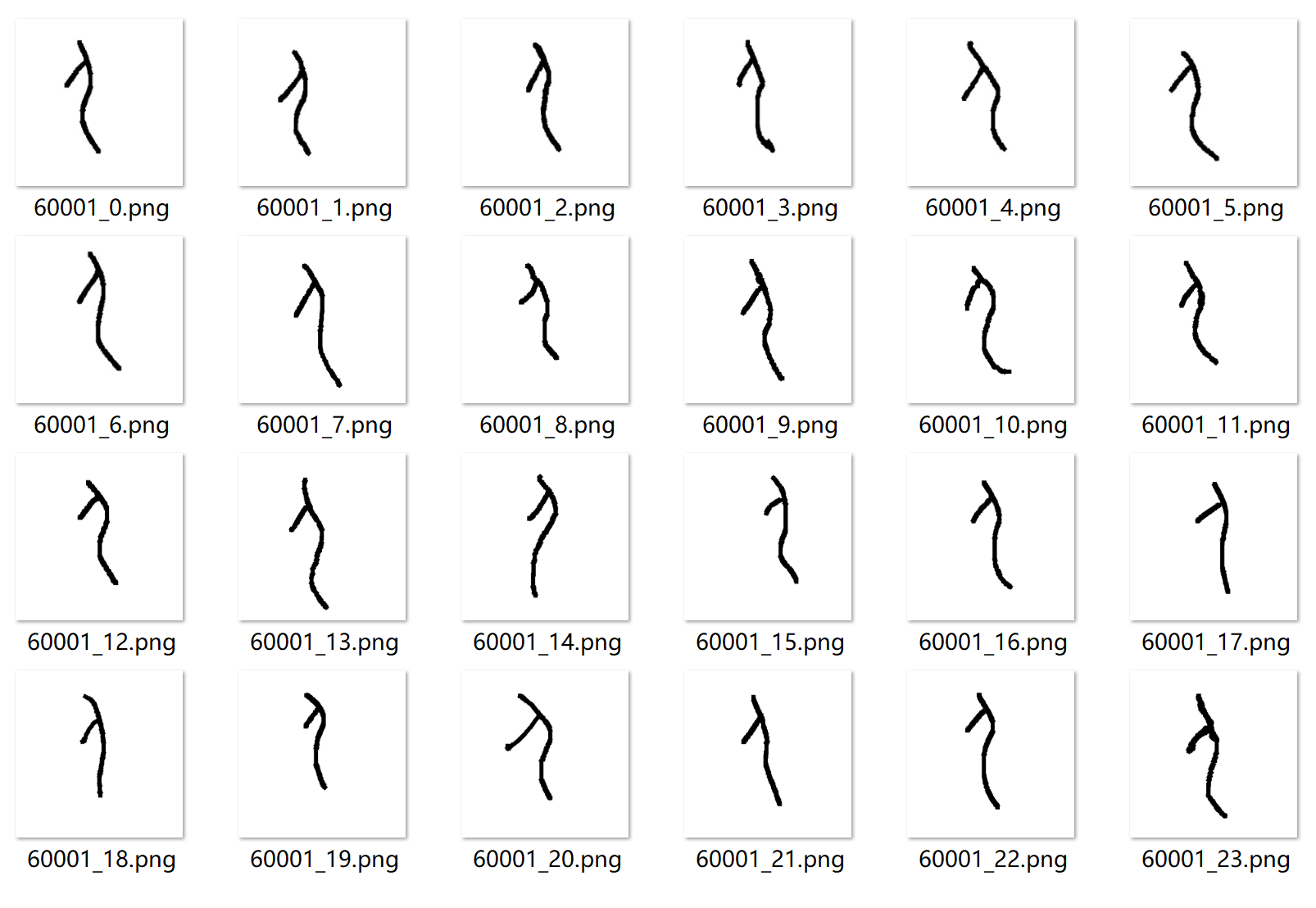
HWOBC由22位来自不同专业（文字、书法、考古、历史、计算机等）的甲骨学研究者，通过手写甲骨字搜集软件比照甲骨字标准字形书写并整理而成，共83245张图片，分为3881类。每张图片大小为400\*400，为纯白的背景上的清晰字形，总共1.44 GB。

图 3：HWOBC数据集示例

* + 1. OBC306[12]

OBC306由广泛受学术界认可的甲骨文出版著录作为数据集的素材来源，将著录的所有页面扫描为数字图像，借助“甲骨文字典”工具，从沈建华整理的《甲骨文字形表》出发，检索出表中甲骨文字符在拓片上出现的所有位置，并手动裁剪出来作为每个字符的样本图像，汇集构建数据集。因为图片为电子扫描的拓片，每张图片的大小不定，并含有大量噪音。共有属于306个类的309551张图片，共5.94 GB。

图 4：OBC306数据集示例

* + 1. OBIDataset[]

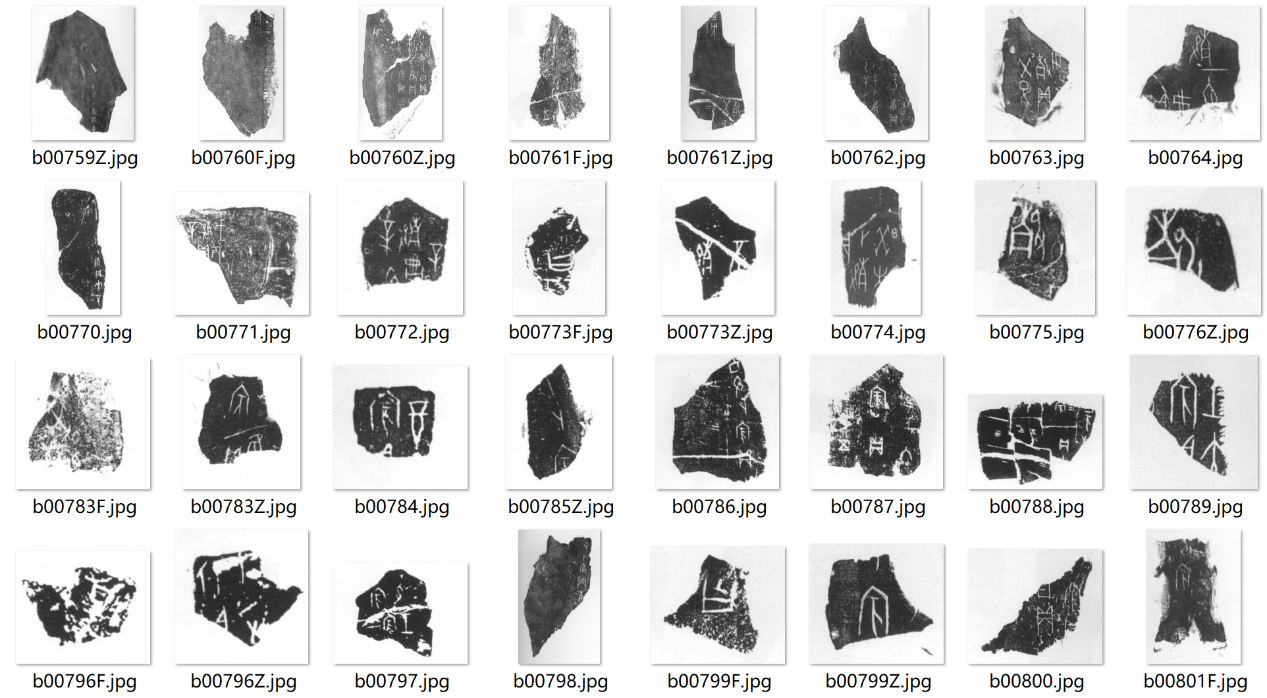
OBIDataset来源于《甲骨文字合集》,《甲骨文字合集补编》与《怀特氏等收藏甲骨文集》，通过高分辨率扫描仪转为电子书后手工截选并标注。每张图片为整张甲骨或甲骨残片的拓片，或是几张较小的甲骨的拼接图片，大小从3KB到1500KB不等，含有大量噪音和残缺。

图 5：OBIDataset数据集示例

标注分为训练和测试两种，都以.json文件保存，格式如下：

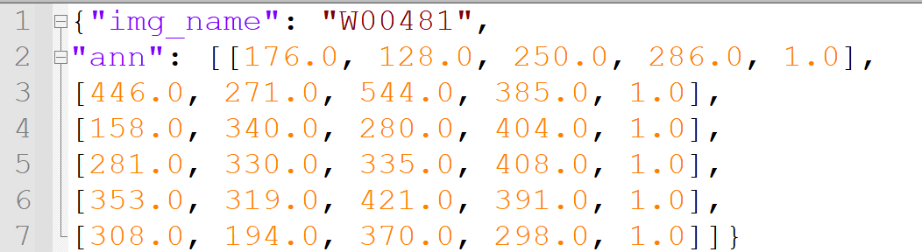


图 6：OBIDataset使用的标注格式；”img\_name”代表图片名，”ann”包括所有的标注框，其中前四位分别是左上角的x,y坐标和右下角的x,y坐标，第五位为检验

由于标注是人工进行的，有一部分图片的标注框位置出现了偏差，整体平移了一段距离。

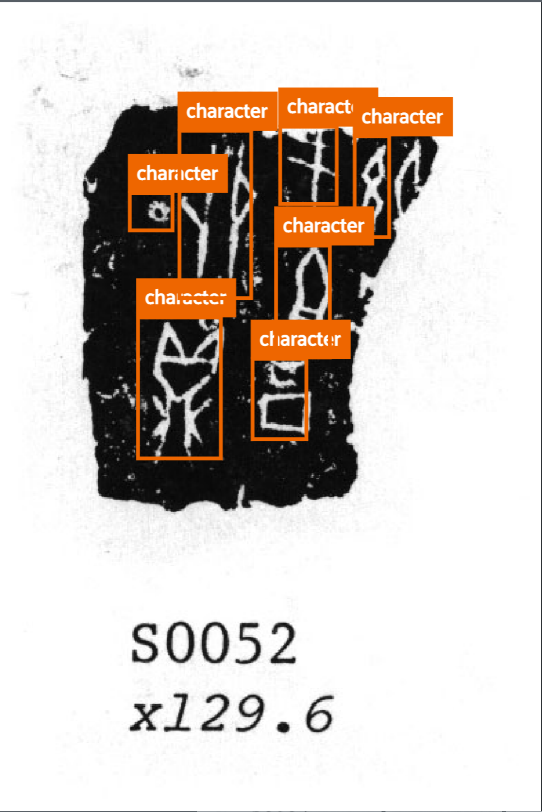


图 7：左图为正常注释，右图为出现偏差错误的注释,每个标注都向下偏移了一段距离；图片使用Fifty-One[12]实现可视化

* 1. 局限性和挑战
     1. 长尾效应[10]

长尾效应是甲骨文数据集的一种常见问题。由于甲骨文出土量较少，且甲骨文中有很多异体字、生僻字等原因，虽然一个数据集的分类数可达几千个字头，但很多分类下的数据量可能极少，甚至只有1至2张图片。如果为每个分类的数据量从高到低画成一张条形图，可以发现数据分布极不平衡，形成一条“长尾”，长尾效应因此得名。

长尾效应会严重影响神经网络模型识别生僻字和异体字的精确度，降低数据集质量，而且在短期内不可能从数据来源方面解决问题。目前有研究者使用生成式对抗网络的技术(Generative Adversarial Network, GAN)有效解决了长尾效应[10]。

* + 1. 编号不统一

目前对于基于深度学习的甲骨文识别项目中，没有一个统一的甲骨文字分类、编号标准，不同的研究者会根据甲片不同的特征(如甲片出土地、记载文献、文字内容、记录顺序等)使用不同的编号方式。这导致单个研究项目使用的数据集无法兼容其他数据集，影响了研究成果的普适性。

由于短期内不可能有适用所有甲骨文字的数据集分类标准，甲骨文编号不统一的问题很难有效地解决。我使用的数据集融合方法(在3.1中提到)在一定程度上提供了一个可行的解决方案。

* + 1. 噪音和残缺

甲骨文的历史有一千年之久，这意味着甲片在发掘出来的时候已经严重氧化、破损，再加上挖掘时的磨损、在刻录甲骨文时火烧出的裂痕、龟壳自带的生长痕等，导致甲片的扫描图片有严重的噪音和残缺。有些甲骨文字中出现破损，导致笔画粘连、字形模糊，难以和裂痕区分开来，甚至专家都很难人工辨认；甲骨文在地底时和沙土等的摩擦造成整块甲骨上会有斑点状噪音；火烧出的裂痕和龟壳的生长痕由于时间久远或发掘人员疏忽，可能会整块断裂，造成甲片不完整，这些都会严重影响神经网络模型识别甲骨文的精度和甲骨文文意的破译。

目前存在很多减缓噪音影响的滤波方式[13][14]，但在甲骨文数据上的效果并不理想[]。通常情况下，增加数据量是一个有效的针对噪音的解决方案，但发掘出的甲骨文数量不多，所以无法依靠这种方法。

* + 1. 异体字[15]

甲骨文的书写系统极为特殊，虽然本身为现代汉语的前身，但汉字的书写系统经过了很长时期的演化，甲骨文字和现代文字之间的形态、意义差距很大，增加了破译难度。甲骨文中经常会出现异体字、通假字等特殊用法，甚至单个字的不同样本之间也存在着差距。这会进一步减少单个甲骨文字的样本数量，并加剧长尾效应，严重影响模型精度。对于异体字的问题，除了增大数据量、使用数据扩增等以外，还需要一个更为有效的解决方案。

# 数据扩增

1. 1. 数据集融合

本节的主要目的在于高效、精准地将两个不兼容的数据集进行合并，达到数据扩增的目的。在本节中，将较大的HWOBC数据集设定为基准数据集，将较小的train-1600数据集设定为待融合数据集。

* + 1. 重要性和挑战

当前，关于机器学习在识别甲骨文方面的研究中大部分[7][8][9][10][15]主要只采用一个数据集，数据集来源一般由出土的甲骨文原始文献通过电子扫描[9]，或甲骨文专家手动誊写[8]得到。虽然这样的数据集质量较为可靠，但存在着明显的局限。单一数据集数据量较少，很难直接训练出一个精度较高的同时也比较通用的文字识别模型。由于当前并没有一个统一的甲骨文字分类标准[15]，不同的研究所用的数据集差异很大，无法兼容，导致研究者不能简单地直接将多个数据集合并以增加训练数据量，在接受新数据方面造成困难，也导致单个研究中的数据对于现存可用的数据量来说利用率低下。因此，利用数据集融合对于调高模型精度非常有帮助。

若使用人工识别合并数据集，虽然精准度较高，但由于图片数量巨大，人工识别大量耗时耗力，不切实际。

对于待融合的两个数据集，各自的分类标准和编码方式不同，无法直接看出属于同一个类别的图片，也不能简单的将两个子目录复制在一个目录下。

基准数据集和待融合数据集不是简单的包含关系，待融合数据集中可能有基准数据集没有的类型，如何准确判断带融合数据集中的每一个类型是否是新类型需要解决。

本文提出的利用机器学习的数据集融合方法可以快速、高效的合并兼容多个数据集以供训练，大幅度提高单个模型的兼容性和实用性。

* + 1. 思路

为了解决上述的问题，我提出了一种基于神经网络“过度学习”的数据融合方案。首先，我用数据量较大的一个数据集作为基准数据集来训练一个AlexNet模型，使得这一模型能够判断甲骨文图片在较大数据集中所属类别。

由于AlexNet已经被证实拥有较高的识别精确度[8]，我将AlexNet应用到待识别数据集上来识别该数据集中每个图片对应于基准数据集中的类别。这一过程中的难点在于：待融合数据集中可能会出现基准数据集中不存在的类别，我基于AlexNet输出层预测概率提出了一种有效判断准则，来判断当前图片对于基准数据集来说是否属于一个新类别。这样能减少识别时间，减轻可能存在的误差带来的影响。

在数据融合之后，我还通过人工随机抽样验证了数据融合的效果，发现数据融合达到了理想的表现。

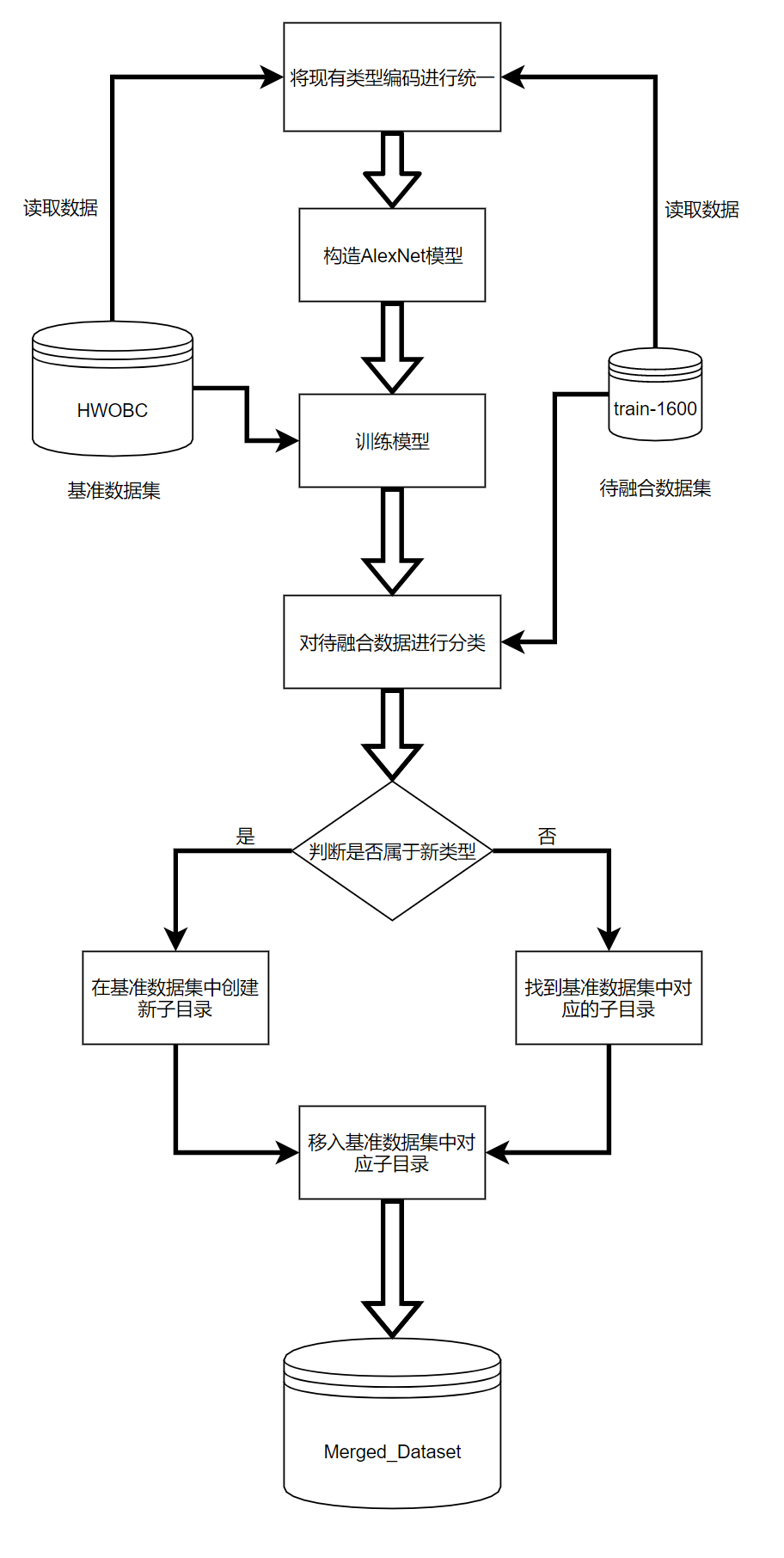
* + 1. 实施步骤

图 8：实施数据集融合的基本流程

本文使用两个分类格式不同、包含甲骨文字个数不同的数据集HWOBC和train-1600进行融合。由于两个数据集各有一套甲骨文字编码方法[9][12]，无法直接找到两个数据集中可能同时存在的甲骨文字。将两个数据集的每个class的标签改为数字编码(例如0000,0001等)。

我构造了一个AlexNet模型，在较大(3881个class)的HWOBC数据集上训练至略微过度拟合(25-30个epochs)。比起选用train-1600 (1621个class)数据集作为基准数据集，使用更大、class数更多的数据集可以提高模型精准度、减少需要识别出的新字的数量，节省时间。

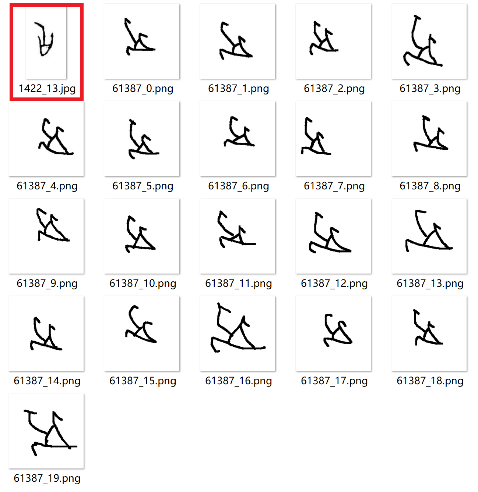
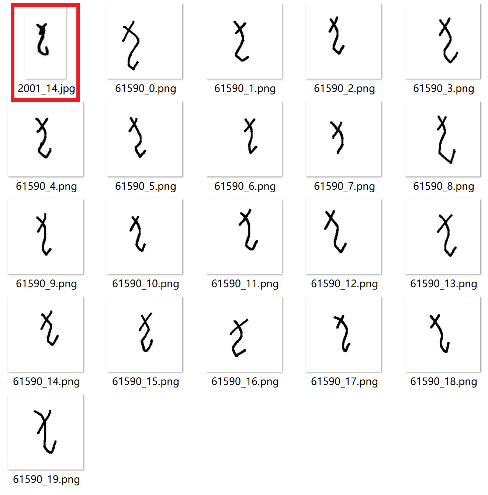
我再使用训练好的AlexNet模型对train-1600数据集进行分类，对train-1600中每张图片生成一个3881列的NumPy数组，每一列分别代表该图片与HWOBC数据集中3881个class的相似度。

对于train-1600中的每个class，选出该class全部数组中的最大值，此值的数组下标代表该class“最有可能”为HWOBC中的class的编号。如果这个值超出了一个人工选定的阈值时，判定该class与HWOBC中的class属于同一类，并将两组数据合并在新数据集的一个class下。否则，判定该class为HWOBC中未出现的class，并单独归入新数据集的一个类下。

通过多次试验，以人工抽样检查相似度的方法，我确定阈值为0.72。此时新数据集共有4540个class，大致符合目前出土甲骨文字数量[]，并略大于此数。当新数据集的class数多于实际字数时，代表同样的字可能被分类至多class中，这样的问题比新数据集的class数少于实际字数时不同的字分类入同样的class的情况更容易解决，因此选用略大于实际字数的误差。命名新数据集为Merged\_Dataset。在融合完成后，我进行随机的人工抽查以保证数据集融合的效果。

Merged\_Dataset大小为247 MB，包含在4540个子目录下的88963张图片。

图 9：分类后的数据集，红框标注的来源于train-1600数据集，其余来自HWOBC数据集



* 1. 数据扩增

本节主要利用已有的原始数据，在保持不严重过度拟合的基础上增加训练样本，以达到增加数据量、丰富数据多样性、提高模型的泛化能力的目的。

重要性

在目前出土的甲骨中，单个甲骨文字在不同的甲片、骨片上的形态差别较大，在文字朝向、大小、笔画完整度、噪音等方面均存在差异[]。而在目前整理扫描出的可供机器学习训练使用的甲骨文数据量极少，且在短时期内无法改变现状。这会导致训练出的模型精准度较差。若采用某种措施使得可供机器学习的数据量大幅度增加，该问题会得到减轻。有调查[16]证明，对于某些小数据集使用数据扩增可以使模型精确度增加多于20%。

实施步骤

选用在3.1中已融合的数据集(Merged\_Dataset)，将其中的每张图片进行以下操作的随机组合：

·水平平移0%-20%

·竖直平移0%-20%

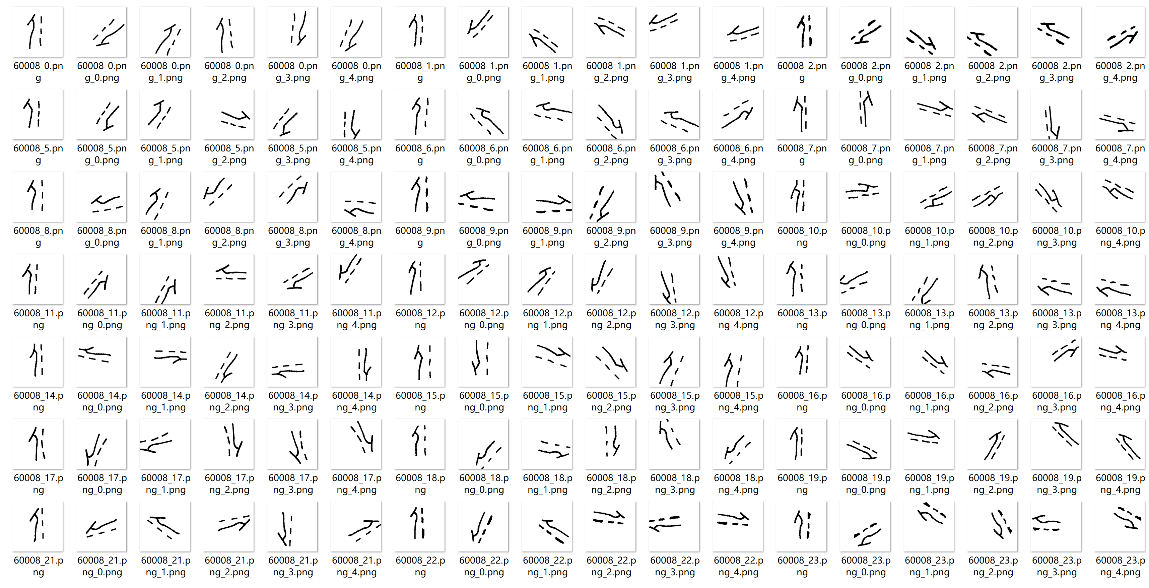
·水平翻折

·竖直翻折

·逆时针旋转0-180°

为了防止大幅度增加训练时间，将每张图片进行的操作次数选为5次，再生成新数据集。新数据集(Augmented\_Dataset)的大小为Merged\_Dataset的6倍，但子目录数不变。

图 10：同一个子目录经过数据扩增的前后对比；每一张图片都被扩增了5次，大幅度增加数据量









# 模型搭建

5. 1. 本地环境

甲骨文识别

操作系统：Windows 10 64位

处理器：英特尔 Core i9-10900K @ 3.70GHz

主板：戴尔 0KV3RP ( LPC Controller - 0684 )

内存：128 GB ( 镁光 DDR4 3200MHz )

硬盘：英特尔 SSDPEKNW010T9 (固态硬盘)

显卡：Nvidia GeForce RTX 2080 Ti

甲骨文定位

操作系统：Windows 10 专业版 64位

处理器：英特尔 Xeon(至强) Gold 5218R @ 2.10GHz

主板：戴尔 060K5C（C620 series chipset - A1C1）

显卡：NVIDIA GeForce RTX 3090

内存：128 GB (DDR4 3200MHz )

主硬盘：英特尔 SSDPEKNW010T8

* 1. 神经网络结构的改进
     1. 跳跃连接 (Skip Connection)

过度拟合是机器学习中的常见问题。在数据集较少，模型深度较大，在数据集上训练时间过长时，模型会记忆数据集中的噪音和无关信息，导致模型适应性随着时间反而下降，无法可靠预测新数据。在本研究中，可供训练的甲骨文数据集数量较少，很有可能出现训练时间太少，模型精准度不够，训练时间太多，又会过度拟合的问题。

针对这样的问题，可以在模型中跳过一些已有的连接，使这些节点的输出被丢弃，造成“遗忘”的效果，有效的减少过度拟合发生的可能性。

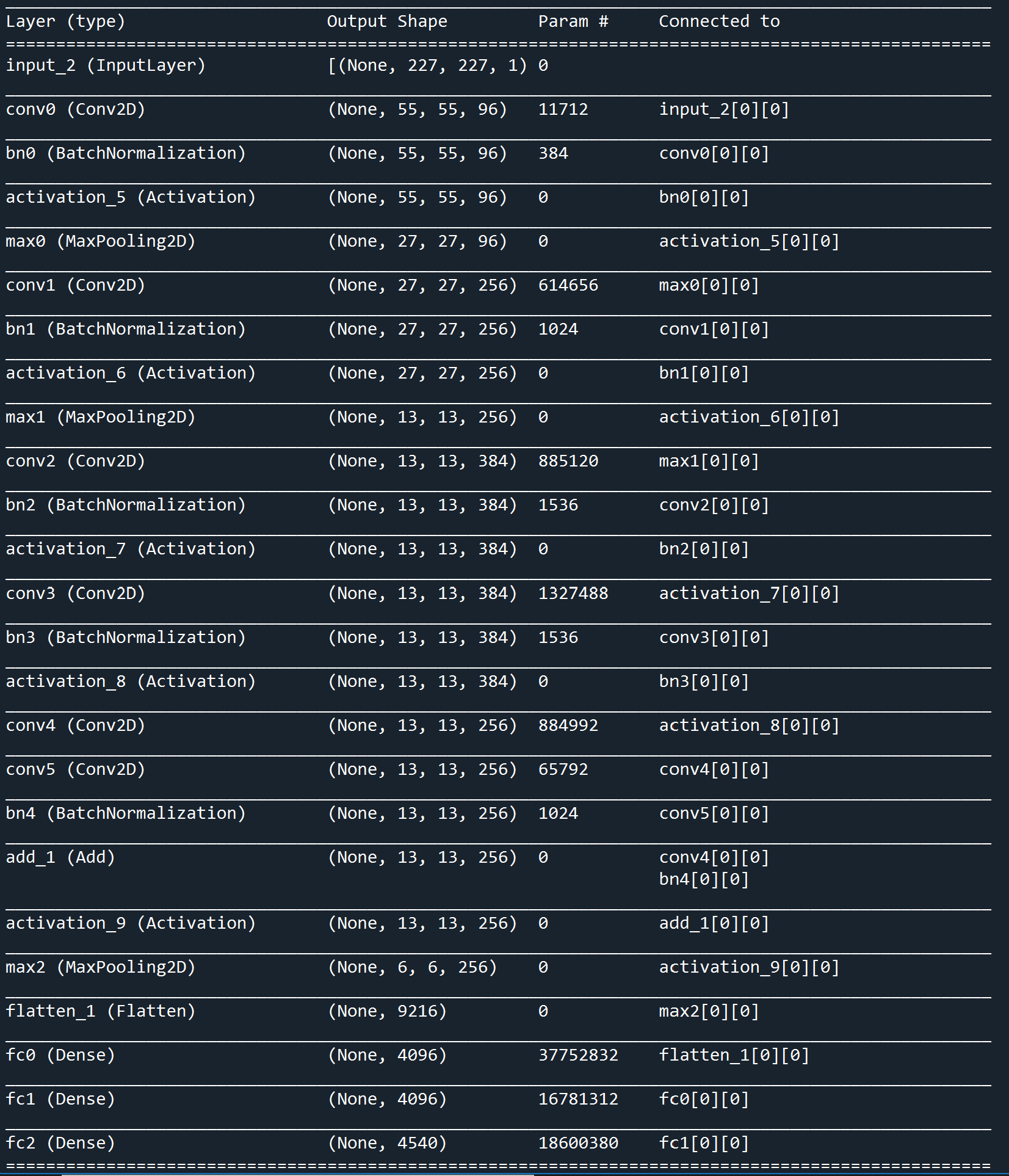
在我使用的模型中，我在传统的AlexNet结构中的第五层卷积层之后又加入了一个残缺模块(Residual Block)，直接连接了输入层和第六层卷积层，也具有断开连接的效果。改造过的AlexNet结构如下图。

图 11：增加了残缺模块的AlexNet模型

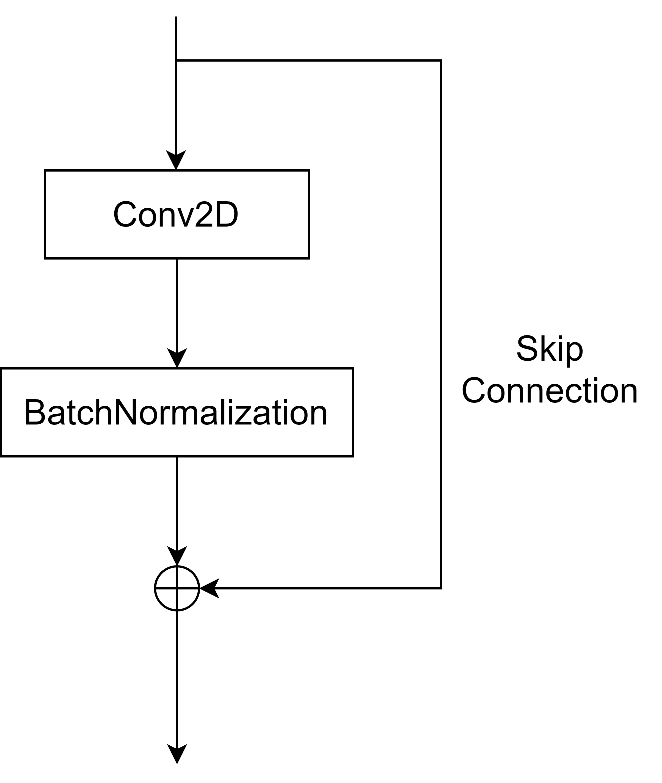


图 12：残缺模块

* + 1. 批标准化 (Batch Normalization)
  1. 利用Optuna进行超参数优化
     1. 超参数优化的方法

如2.2.3和2.2.4节所描述，作为一种古老的书写体系，甲骨文的图像特征较为特殊。这导致了现有的机器学习模型中很少有适合甲骨文字的预训练模型(Pre-trained Models)。对于一个未经训练的模型，如果模型的超参数(Hyperparameter)设置不当，会严重影响模型精度和效率。因此，针对识别甲骨文的神经网络模型需要进行超参数优化(Hyperparameter Optimization)。

机器学习模型中存在两种类型的参数：模型参数和超参数。模型参数在机器学习模型训练过程中不断自动更新，以达到优化模型的目的，而超参数无法直接从数据学习获得，必须在训练机器学习模型之前进行设置，因为它们定义了模型架构、训练目标、训练方式、数据预处理的参数等[17]。

超参数优化的方法包括手动调参、网格化搜索(Grid Search)、纯随机搜索(Random Search)、贝叶斯优化(Bayesian Optimization)等。

* 手动调参适合模型搭建初期对不同参数影响进行初步评估，但若使用该方法对多个超参数进行调试，会需要大量人工操作，效率低下。
* 网格化搜索指在参数空间内遍历所有参数组合，并返回最优参数组合。该方法的好处在于几乎肯定能够找到最优组合，但过程极其耗时[18]。
* 纯随机搜索指在参数空间内随机抽取一定数量的参数组合，并返回其中的最优参数组合。虽然该方法不一定能找到最优组合，但依然可以达到较高的精度，同时大量节约时间[19]。
* 贝叶斯优化利用贝叶斯理论对超参数进行探索，以达到快速找到最优参数组合的目的[18][20]。贝叶斯优化也是一种随机搜索，但理论上能经过较少的次数就可以找到较好的参数组合。

由于我的神经网络模型训练时间较长，超参数组合数量很多，手动调参和网格化搜索耗时过长，在本实验中不具有可行性。即使我在训练时使用GPU，单次训练时间依然长达2-3小时，我决定采用贝叶斯优化算法中的TPE(Tree-structured Parzen Estimator)算法。

在具体实施过程中，我使用了Optuna算法包实现对模型的超参数优化。Optuna[21]是一个基于Python 3的超参数优化算法包，提供不同的超参数优化算法进行自动搜索，支持PyTorch、TensorFlow、Keras等多种机器学习框架。本节中，我采用Optuna进行超参数优化试验。相比于scikit-optimize[22]等超参数优化算法包，Optuna并不主要依附于scikit-learn平台，而是支持我使用的以TensorFlow为后端，Keras为前端的机器学习平台。Optuna相比同样支持Keras的Hyperopt[23]和Ray-tune[24]等算法包而言操作更为简便，体量更小，不会大幅度影响单GPU训练的速度，同时支持便捷的存档功能，将搜索结果存入本地数据库文件，方便下次继续搜索时导入。

* + 1. 超参数优化的实施步骤

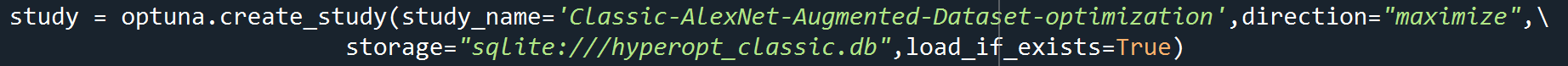
 在开始超参数优化之前，我先使用Optuna创建了一个学习过程(optuna.study.Study)，代码如下图所示。

图 13：定义的学习过程

其中，study\_name变量定义了该次超参数优化的实验名称。direction为选择的优化方向，因为我选择的优化标准为测试精度，所以方向选为最大化。storage变量定义了超参数优化过程中进度存储的路径和方式，我选择在本地创建一个.db数据库来保存进度。将load\_if\_exists变量设置为True后，程序会在遇到异常状况或人为中断时自动存储进度，以使得下次无需重新开始，而可以直接将上次的搜索进度导入以继续上次搜索。由于一次超参数搜索试验可能长为40-60小时，这样节省了大量时间。

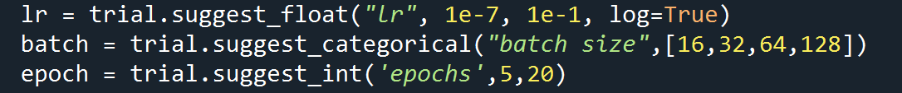
在本次实验中，我同时采用了传统的AlexNet、增加了残差模块(Residual Block)的AlexNet 和ResNet50三个神经网络模型，并选取对于神经网络训练精度影响最大[]的学习率(Learning Rate, LR)、批大小(Batch Size)和时期数量(Epochs)作为参与调优的超参数。三个参数的调优范围如下图所示。

图 14：选择用于优化的三个超参数，其中trial是一个用于生成超参数具体值的类的对象

|  |  |  |
| --- | --- | --- |
| 超参数 | 范围类型 | 范围 |
| 学习率 (Learning rate) | 采用对数域(Log domain)的浮点数 | 1e-7 ~ 1e-1 |
| 批大小 (Batch size) | 离散 | {16, 32, 64, 128} |
| 时期数量 (Epochs) | 整数 | 5 ~ 20 |

图 15：超参数的调优范围

我创建了一个名为objective的函数，其中读入了需要调优的超参数及范围，并定义了需要训练的神经网络模型结构，读取验证数据集并进行训练和验证。在一次训练结束之后，返回本次训练精度数据，最后由主程序挑选最高值。

在训练过程中，我先对2-3个参数组合进行优化以便测试程序有效性，在确定无误之后再逐渐增加选择的参数组合个数，直至最大值20个。在训练完成之后，程序读取每次训练的参数组合和最终训练精度，并输出精度最高的参数组合，实现超参数调优。最后，我使用最优参数组合在一个合并了训练和验证数据集的综合数据集上进行正式训练，并在测试数据集上评定该参数组合的性能。

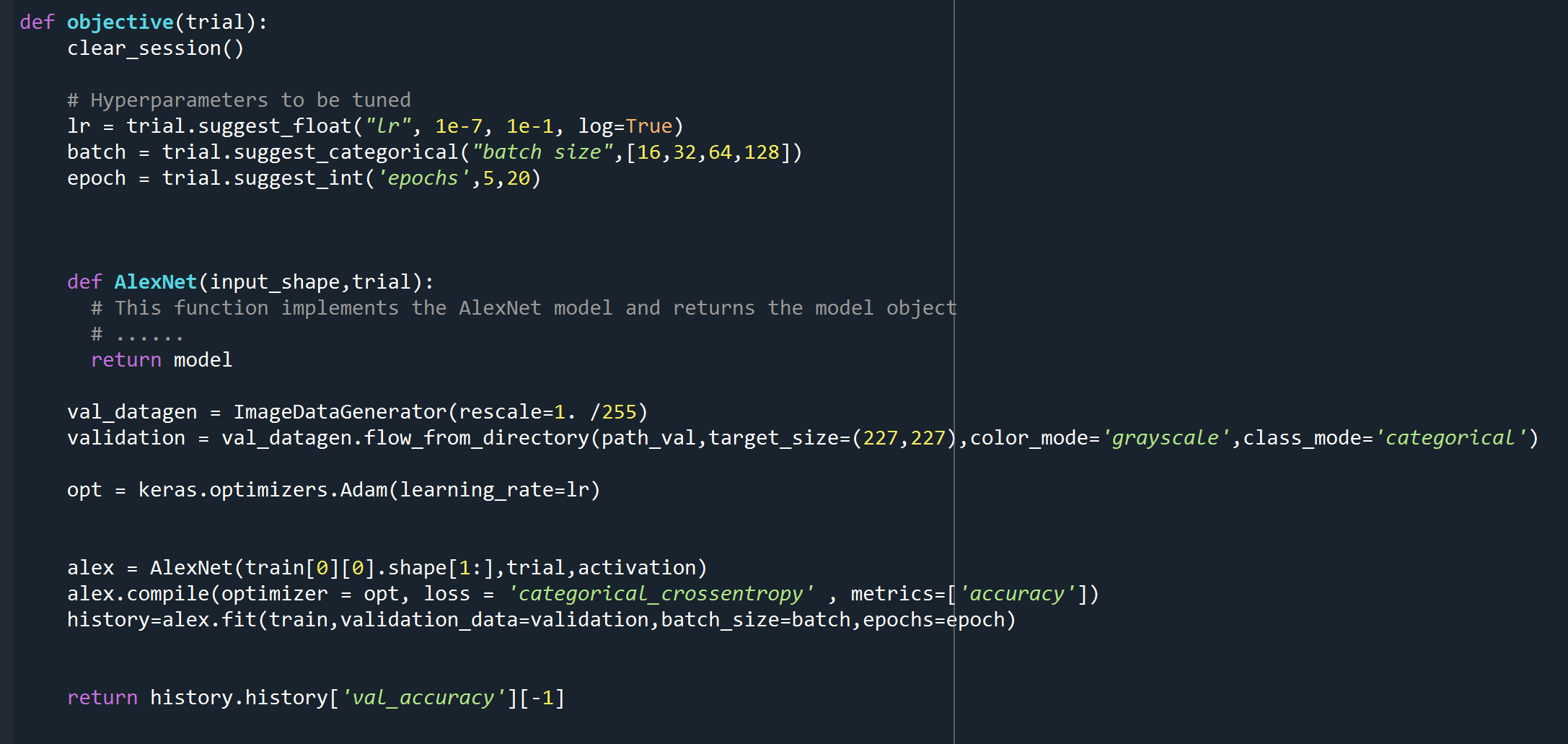


图 16：objective函数，读入了超参数、网络结构和验证数据集，对模型进行训练，并输出本次验证的精确度

图 17：使用optimize函数进行优化，n\_trials变量决定了选择的参数组合个数，timeout决定超时时长

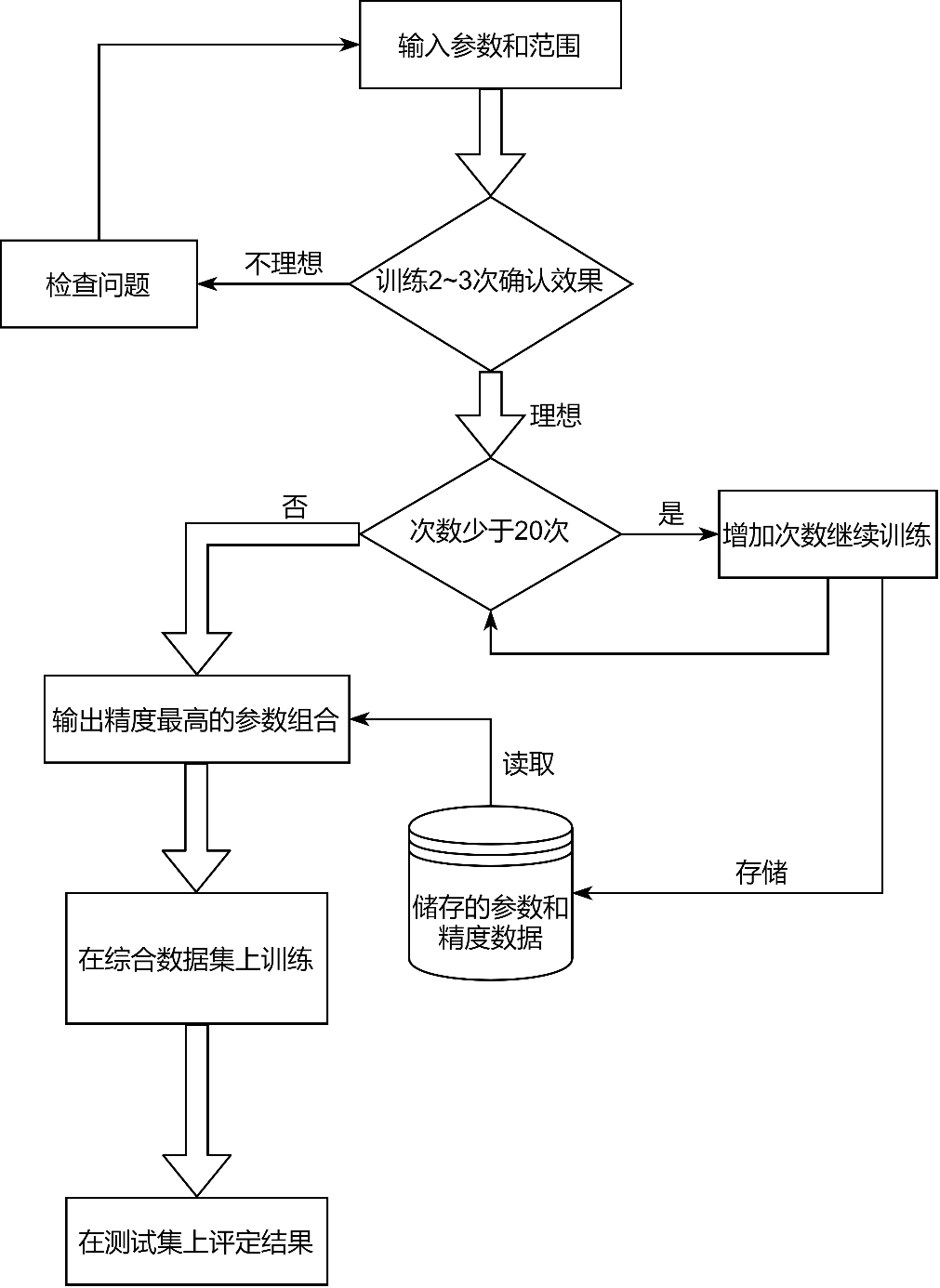


图 18：超参数优化的步骤；“次数”指尝试的参数组合个数

* + 1. 超参数优化实验结果

在三个神经网络中，我从上图的参数空间中各进行了20次采样，以精确率作为评判标准由Optuna选出最优参数组合，数据如下表。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 神经网络 | 最优学习率 | 最优批大小 | 最优时期数量 | 最高精确率(测试集) | 最高精确率(验证集) |
| 传统AlexNet | 1.647e-5 | 128 | 20 | 91.53% | 88.42% |
| 带有残缺模块的AlexNet | 4.304e-5 | 32 | 20 | 91.86% | 89.51% |
| ResNet50 | 1.112e-4 | 128 | 15 |  | 0.3211% |

图 19：三种神经网络的最优参数组合，所有数据均取四位有效数字

由以上数据可以初步看出，两种AlexNet的结构都达到了预期的效果，其中增加了残缺模块的AlexNet训练效果略微好于传统AlexNet，而ResNet50的效果明显差于预期，这可能是因为ResNet50的深度过大，导致严重的过度拟合。

另外，两种AlexNet的最优时期数量都在该参数范围的最大值，说明该参数的范围可能需要进一步扩大以达到更好的效果。

* 1. 甲骨文定位

除了将已知的一个甲骨文字分类到相对的字头外，深度学习技术也可以用于在一块甲骨的图片上定位所有甲骨文字。这需要使用物体检测(Object Detection)技术实现。

我选择了目前较为先进且在数据量较少时表现良好的YOLOv5m[26]网络，在OBIDataset数据集上进行训练，并在Wandb (Weights and Biases) [27]平台上记录训练及验证效果。

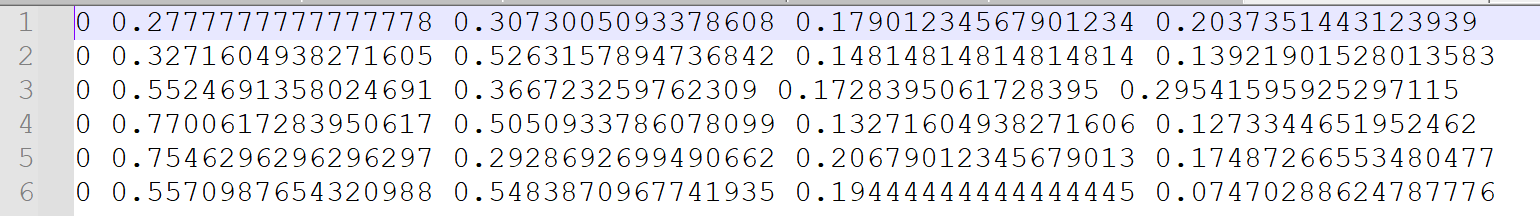
首先，我将OBIDataset原本的.json格式注释转为YOLOv5所要求的格式,并存储在相应的路径中。YOLOv5的格式如下图所示。

图 20：YOLOv5的标注格式；每一行为一个标注框，第一位整数为物体类型（此处都为“甲骨文字”），后面四位分别代表标注框中心的横坐标、纵坐标以及标注框的宽和高，都已相对整张图片尺寸的比值保存

我从数据集中随机提取了25%的数据作为测试集，并创建了一个.yaml文件，用于给模型识别数据集的内容和位置。该文件的内容如下图所示。

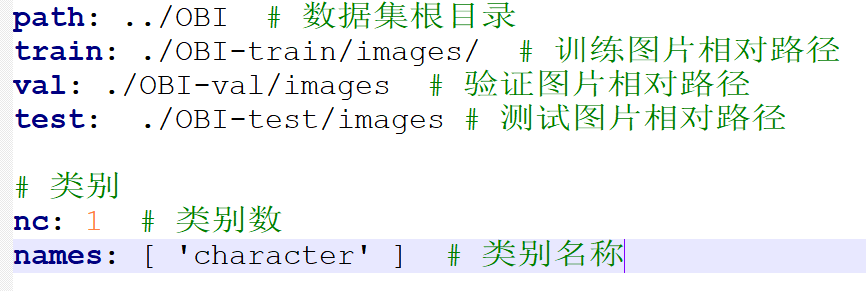


图 21：obi.yaml文件，用于描述数据集

# 结果分析

对比实验：

1. 经典AlexNet，数据融合，超参优化

lr=1.647048583923323e-05

batch size=128

epoch=20

accuracy=0.8842096328735352

1. 经典AlexNet, 无数据融合，超参优化
2. 经典AlexNet，数据融合，无超参优化
3. 经典AlexNet，无数据融合，无超参优化
4. AlexNet with residual block，数据融合，超参优化

lr=4.303709151641001e-05

batch size=32

epoch=20

accuracy=0.8950909376144409

1. AlexNet with residual block, 无数据融合，超参优化

lr=4.866306999596021e-06

batch size=128

epoch=16

accuracy=0.884843111038208

1. AlexNet with residual block，数据融合，无超参优化

lr=1e-3

batch size=32

epoch=15

accuracy= 0.005232992582023144

1. AlexNet with residual block，无数据融合，无超参优化

lr=1e-3

batch size=32

epoch=15

accuracy=0.0002615518751554191

1. AlexNet with residual block，数据扩增，无超参优化
2. AlexNet with residual block，数据扩增，超参优化

lr=2.5722257853319e-05

batch size=32

epoch=20

accuracy=0.918594241142273

1. ResNet50，数据融合，超参优化
2. \*ResNet50, 无数据融合，超参优化
3. ResNet50，数据融合，无超参优化
4. ResNet50，无数据融合，无超参优化

# 6 总结

# 参考文献

1. 王潇.从殷墟甲骨文看商代祭祀占卜活动[J].文化产业,2021(08):79-80.
2. Cheung C. The Chinese History That Is Written in Bone [EB/OL]. https://www.sapiens.org/archaeology/chinese-oracle-bones-history/. 2021.5.21.
3. 马昌，曹昆. “习近平致甲骨文发现和研究120周年的贺信.”人民网. 人民日报, 2019. 2021.5.21.
4. 阴崔雪. 基于甲骨文的殷商学校教育研究[D].华东师范大学,2019.
5. 顾绍通.基于拓扑配准的甲骨文字形识别方法[J].计算机与数字工程,2016,44(10):2001-2006.
6. 鲁绪正, 蔡恒进, 林莉. 基于 Capsule 网络的甲骨文构件识别方法 [J]. 智能系统学报, 2020, 15(2): 243–254.
7. Y. Zhang, H. Zhang, Y. Liu, Q. Yang, C. Liu. Oracle Character Recognition by Nearest Neighbor Classification with Deep Metric Learning.[J] 2019 International Conference on Document Analysis and Recognition (ICDAR), 2019.
8. Chen Shanxiong, Mo Bofeng, Han Xu, Gao Weizhe, Liu Xuxing. A Classification Method of Oracle Materials Based on Local Convolutional Neural Network Framework.[J]. IEEE computer graphics and applications, 2020.
9. Han X, Bai Y, K Qiu, et al. IsOBS: An Information System for Oracle Bone Script[C]. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, 2020.
10. OBC306数据集 [EB/OL]. <http://jgw.aynu.edu.cn/ajaxpage/home2.0/DataOBC/detail.html?sysid=20>. 2021.8.28.
11. 王浩彬. 基于深度学习的甲骨文检测与识别研究[D].华南理工大学,2019.
12. 郭沫若, 胡厚宣. 甲骨文合集[J]. 北京: 中华书局, 1982, 21.
13. Voxel51[EB/OL]. <https://voxel51.com/>. 2021.8.14.
14. Bang Li, Li Bang, Dai Qianwen, Gao Feng, Zhu Weiye, Li Qiang, Liu Yongge. HWOBC-A Handwriting Oracle Bone Character Recognition Database[J]. Journal of Physics: Conference Series,2020,1651(1).
15. 李传朋，秦品乐，张晋京．基于深度卷积神经网络的图像去噪研究. [J]. 计算机工程. 2017, 43(3): 253-260.
16. 徐健. 基于深度卷积神经网络的图像椒盐噪声去除算法研究[D].合肥工业大学,2019.
17. 刘国英.基于深度学习的甲骨文字检测与识别[J].殷都学刊,2020,41(03):54-59.
18. Minaee Shervin, Boykov Yuri Y, Porikli Fatih, Plaza Antonio J, Kehtarnavaz Nasser, Terzopoulos Demetri. Image Segmentation Using Deep Learning: A Survey.[J]. IEEE transactions on pattern analysis and machine intelligence,2021,PP.
19. 浮盼盼,司琪,王鑫赛.机器学习算法的超参数优化：理论与实践[J].电脑编程技巧与维护,2020(12):116-117+146.
20. 张爱军,杨泽斌.自动化机器学习中的超参调优方法[J].中国科学:数学,2020,50(05):695-710.
21. Bergstra J , Bengio Y . Random Search for Hyper-Parameter Optimization[J]. Journal of Machine Learning Research, 2012, 13(1):281-305.
22. Mockus J , Tiesis V , Zilinskas A . The application of Bayesian methods for seeking the extremum[M]. 1978.
23. Takuya A., Shotaro S., Toshihiko Y., Takeru O., Masanori K. Optuna: A Next-generation Hyperparameter Optimization Framework[C]. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '19), 2019.
24. scikit-optimize: Sequential model-based optimization in Python [EB/OL]. <https://scikit-optimize.github.io/stable/>. 2021.6.19.
25. Hyperopt: Distributed Hyperparameter Optimization [EB/OL].<https://github.com/hyperopt/hyperopt>. 2021.6.19.
26. Liaw, R., Liang, E., Nishihara, R., Moritz, P., Gonzalez, J. E., Stoica, I. Tune: A research platform for distributed model selection and training[M]. 2018.
27. Jocher, G., et al. Ultralytics/YOLOv5: V5. 0—YOLOv5-P6 1280 Models, AWS, Supervise.ly and YouTube Integrations. 2021.
28. L. Biewald, “Experiment Tracking with Weights and Biases,” Weights & Biases. [EB/OL]. http://wandb.com/. 2021.8.28.

# 谢辞

谢辞需要包含以下内容：

（1）论文的选题来源、研究背景；

（2）每一个队员在论文撰写中承担的工作以及贡献；

（3）指导老师与学生的关系，在论文写作过程中所起的作用，及指导是否有偿；

（4）他人协助完成的研究成果。

如有必要，请附上团队成员和指导老师的简历。