

医学图像处理中数据类别不均衡问题的研究与实现

摘 要

语义分割是医学图像处理的重要任务之一，它可以自动地将医学图像中目标区域的像素分割出来，从而加速医生的读片、分析速度，减轻医生的负担。在医学图像的语义分割中，数据类别不均衡是非常常见的问题，给神经网络对于图像的学习带来了很大的挑战。为了解决该问题，本文从损失函数入手，对于多种针对数据不平衡的损失函数在两个细胞核分割的数据集上进行实验，同时采用了多种语义分割网络，得到了最合适的算法组合。同时对于类别权重函数进行了探究，提出了新的类别权重函数；还对于在线挖掘困难样本的损失函数进行了困难样本数量的探究，找到了最适合的困难样本数。通过对于上述损失函数存在的问题的分析，又结合了二分类模型对于背景分类的高精度，提出了一种复用背景的两阶段方法，在两个数据集上均达到了更好的效果。

本文所做的工作如下：

- （1）对于针对数据不平衡的损失函数及其组合进行了深入的探究，在实验中得到了较好的结果。
- （2）对于 Focal Loss 的类别权重函数进行了研究，并提出了一种新的适用于数据极不平衡的类别权重函数。
- （3）对于在线挖掘困难样本损失函数的困难样本数进行了实验，找到了最适合于数据不平衡的数据集的困难样本数。
- （4）提出了一种复用背景的两阶段方法，在两个数据集上均取得了精度的提升。

关键词：医学影像，语义分割，数据不平衡，损失函数

Research and Implementation of Methods for Imbalanced Data in Medical Images

ABSTRACT

Semantic segmentation is one of the important tasks of medical image processing. It can automatically segment the target objects of medical image, so as to speed up the reading and analysis speed of doctors and reduce the burden of doctors. In the semantic segmentation of medical images, the imbalance of data categories is a very common problem, which brings great challenges to the learning of neural networks. In order to solve this problem, this paper explores the loss function, carrying out experiments on two nuclear segmentation datasets for a variety of loss functions aiming at imbalanced data, and uses a variety of semantic segmentation networks to get the most appropriate algorithm combination. At the same time, the class weight function is explored and a novel class weight function is proposed; This paper also explores the number of difficult samples for online bootstrapping loss function, and finds the most suitable number of difficult samples. Through the analysis of the problems of the above loss function, and combined with the high accuracy of the binary-classification model for background classification, a novel two-stage method of reusing background is proposed, which achieves better results on both datasets.

The work done in this paper is as follows:

- (1) Loss functions and their combinations for imbalanced data are deeply explored, and good results are obtained in the experiment.
- (2) Different class weight functions of focal loss are studied, and a novel class weight function for extremely imbalanced data is proposed.
- (3) Experiments are carried out on the number of difficult samples of online bootstrapping loss function, and the number of difficult samples most suitable for the unbalanced dataset is found.
- (4) A novel two-stage method of reusing background is proposed, and the accuracy is improved on two datasets.

Key words: Medical image, Semantic segmentation, Imbalanced data, Loss function

目 录

1	引 言	1
1.1	研究背景与意义	1
1.2	研究现状	2
1.3	本文所作的工作及创新点	2
2	相关理论介绍	4
2.1	语义分割的神经网络模型	4
2.1.1	U-Net 家族网络	4
2.1.2	DeepLab 家族网络	5
2.2	数据不平衡问题的解决方法	6
2.2.1	数据增强	6
2.2.2	损失函数	7
3	复用背景的两阶段方法	10
3.1	概述	10
3.2	针对不平衡数据的损失函数存在的问题	10
3.3	二分类语义分割	11
3.4	复用图像背景	12
3.5	复用背景的两阶段方法在医学影像上的应用	13
3.5.1	训练多分类语义分割网络	13
3.5.2	训练二分类语义分割网络	15
3.5.3	复用背景生成预测结果	15
4	实验及结果分析	17
4.1	概述	17
4.2	数据集以及评价标准	17
4.3	模型及损失函数筛选	18
4.3.1	模型与损失函数的初步筛选	18
4.3.1	损失函数组合筛选	21
4.4	各类别权重函数的实验结果与分析	23
4.5	困难样本数量的实验结果与分析	25
4.6	复用背景的两阶段方法的实验结果与分析	27
4.6.1	多分类模型训练	27
4.6.2	二分类模型训练	28
4.6.3	复用背景	29
5	演示程序设计与实现	31
5.1	概述	31
5.2	总体设计	31
5.3	详细设计	32
5.3.1	顶层模块	32
5.3.2	参数解析模块	32
5.3.3	多分类模块	33
5.3.4	二分类模块	33
5.4	输入输出设计	33
6	结论和展望	35
6.1	结 论	35

6.2 展望	35
参考文献	36
谢 辞	38

装

订

线

1 引言

1.1 研究背景与意义

医学影像学在医学诊断领域是一门新兴的学科，它为医生对于患者做出诊断提供了有效且科学的依据。因此，医学影像的使用在目前的临床诊断上是非常广泛的。但是医学影像仅仅只能提供人体各个部位的病理图像，这些图像还需要医生进一步的分析才能得出相应的诊断。而分析这些图像是较为费时的，因此大量的图像分析会给医生带来很大的负担。为了能够减轻医生的负担，于是便引入了深度学习的方法来实现医学影像的自动分割。如今，医学影像分割已经应用到了诸多方面，如心脏图像的分割、癌细胞的分割、脑部和肺部 CT 的分割等。医学影像分割技术很大程度上提高了医生读片的效率，能够帮助医生为患者做出快速诊断。

自 2014 年以来，图像分割技术主要是基于深度卷积神经网络。经过多年的发展，出现了越来越多的算法模型，不断地提高了模型的精度。但是大多数模型都是聚焦于自然场景的图像，而不是医学影像。这导致那些在自然场景下表现较好的模型，在医学影像上的表现不佳。同时各个论文中提出的诸多优化手段也都是针对自然场景的，少有针对医学影像独特特点的优化算法。因此若能够应用针对自然场景的优化算法到医学影像中，同时再根据医学影像的特点进行微调，就让语义分割技术能够在医学影像上达到更高的精度。

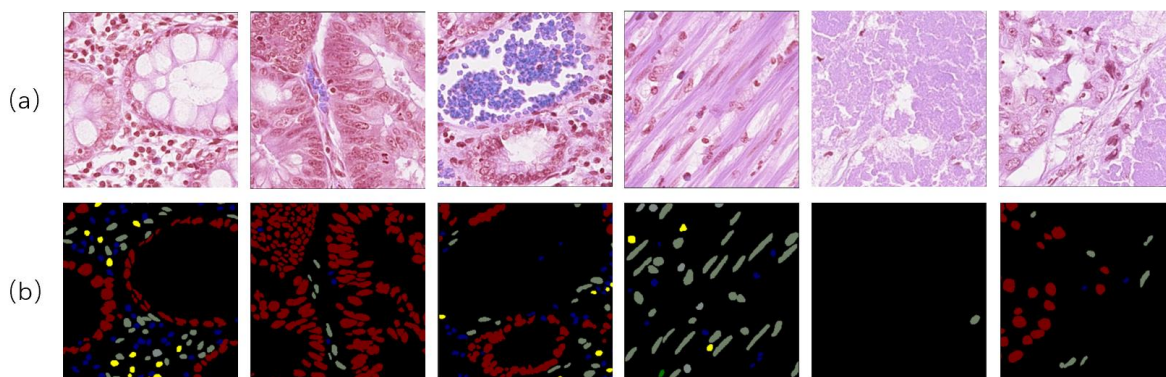


图 1.1 医学影像数据集 CoNIC^[1-2]，（a）医学影像（b）实际像素分类图

数据不平衡问题普遍存在于图像分割中，而一般较为少见于图像分类等计算机视觉问题。这是因为一般图像分类数据集，如 ImageNet^[3]、CIFAR-10^[4]等，在制作数据集时已经将数据不平衡问题考虑在内。但是在语义分割任务中，由于目标物体的大小不同，所包含的像素就不同，自然地就会出现各个类别的像素数不同，而这一问题在医学图像中尤为严重。在医学影像的组织切片中，各种类别的细胞的比例是差距很大的，如在癌细胞的图像中，癌细胞的数量一般而言都是远少于正常的细胞。有时候不同细胞之间的数量比例会达到 1:1000 以上。同时医学影像数据集的构建是非常费时费力的，也无法构建出像 ImageNet 那样庞大的数据集，更增加了数据不平衡问题带来的影响。而且对于医学影像而言，其不感兴趣部分的占比是非常大的，往往会达到 80% 以上。如图 1.1 所示，在人体组织切片的图像中，细胞本身占据的面积较小，或者是数据集所感兴

趣的细胞的面积占比较小，多数面积为其他的不感兴趣的部分。因此不仅有细胞类别之间的不平衡问题，而且有背景与前景之间的不平衡问题。这些不平衡问题都增大了医学影像的语义分割任务的难度。

因此缓解或解决医学影像的数据不平衡问题是医学影像分割任务中非常重要的一部分，如果能够找到合适的方法，那么将会让医学影像的分割精度有很大的提升。

1.2 研究现状

医学影像的分割任务有着非常多的数据集，主要通过各类挑战赛公开，如 LiTS 数据集，这是一个肝脏肿瘤分割的数据集；还有前列腺 MR 图像分割数据集等。这些数据集多为 CT 影像的数据集，这类数据集一般是灰度图像。还有一类数据集，如 CoNIC 数据集，这是一个结肠细胞核分割的数据集，还有 MoNuSAC^[5]，这是一个多器官的细胞核分割数据集。这类数据集的图像是通过苏木素伊红染色过的组织学图像，属于较为难分割的图像类别。由于数据集数量较多，且多以公开赛的形式，因此各类相关的模型算法都是在分散的数据集上进行实验。因此各类算法或模型的应用都需要通过具体的数据集进行实验，以找到最适合的算法。

医学影像的分割任务主要通过深度卷积神经网络来解决。深度卷积神经网络对于图像信息的高层语义的提取是非常强大的，能够充分地提取出图像中像素与像素之间的关系。在语义分割中，最早的将深度学习引入的是 2014 年的 FCN^[6]，而第一个针对医学影像语义分割的神经网络是发表于 2015 年的 U-Net^[7]网络，而后 U-Net 网络也被作为基础的网络结构一直沿用至今，也出现了一系列基于 U-Net 的神经网络，如 U-Net++^[8]、Attention U-Net^[9]、KiU-Net^[10]等，都在不断地提高着神经网络的分割能力。

在语义分割中的数据不平衡问题主要通过损失函数来解决。如以 Dice 指标作为目标函数的 Dice Loss^[11]，还有同样基于区域的损失函数 Tversky Loss^[12]。基于区域的损失函数不是逐像素的计算损失，因此不会受到数据不平衡的影响，因此可以用来作为存在数据不平衡的数据集的损失函数。也有其他基于分布的损失函数可以用于解决数据不平衡问题，如 Focal Loss^[13]，通过给予每个类别不同的权重以及对于简单样本的惩罚来解决数据不平衡问题，还有如 Online Bootstrapping Loss^[14]，通过在线挖掘困难样本，使用困难样本进行反向传播，让网络着重于学习那些困难样本，以此来解决数据不平衡问题。但是这些损失函数也存在问题，基于区域的损失函数，无法平衡各类别样本，尤其是在存在极度数据不平衡问题时；而基于分布的损失函数由于强调较少类别的权重，会导致网络对于较多类别的像素的学习不够充分，导致误分类。

1.3 本文所作的工作及创新点

本文使用了多种语义分割模型与多种损失函数组合对于医学影像的数据不平衡问题进行了探究，同时提出了一种两阶段的方法来缓解数据不平衡带来的问题。主要做了以下工作：

（1）对于 5 种语义分割模型与 5 种损失函数及其组合在两个数据集 CoNIC 与 MoNuSAC 上进行了实验，对于两个数据集，分别得到了最为适合的算法组合。

（2）对于 Focal Loss 的类别权重函数进行探究，共设定了 5 种不同的类别权重函数，包含了 3 种新提出的函数，对于它们分别进行了实验，并得到实验结果。

（3）对于在线挖掘困难样本的损失函数进行了困难样本量的实验，对于两个数据集分别设定了多种困难样本量，并得到实验结果，找到了最合适的困难样本量。

（4）提出了新的复用背景的两阶段方法，对于该方法在两个数据集上进行了实验，并得到了相关的实验结果。

本文的创新点：

（1）提出了新的类别权重函数，该类别权重函数比平衡权重函数更为平缓，在保证模型对于较少类别的像素的学习程度上，提高了模型对于较多类别的像素的学习程度。

（2）提出了复用背景的两阶段方法，利用多分类模型对于前景类别的区分以及二分类模型对于背景预测的高精度，将两者结合从而得到更加精确的预测。

装
订
线

2 相关理论介绍

2.1 语义分割的神经网络模型

图像的语义分割利用深度学习的开山之作就是在 2014 年的 FCN，它采用了端到端的全卷积神经网络，实现了逐像素的分类。SegNet^[15]使用了 2 个 VGG16 作为主干网络，并采用最大池化层的索引来实验上采样。PSPNet^[16]提出了金字塔池化模块，金字塔池化是将输入特征通过不同的最大池化层，生成不同大小的特征图，再通过 1×1 的卷积层，然后将每个卷积层都拼接到原先的特征图中。DeepLab v1^[17]采用了空洞卷积扩大感受野，同时使用条件随机场来细化边界。而后又有了很多的改进版本^[18-20]，都取得了很好的效果。上述诸多模型都是针对现实场景的神经网络，并不是针对于医学影像。由 Olaf Ronneberger 等人提出了 U-Net，是针对医学影像独有的挑战而提出的算法，它采用了对称的编码器-解码器结构。而之后，有了非常多的基于 U-Net 框架的一系列网络的涌现，如 U-Net++、Attention U-Net、MultiResU-Net^[21]、KiU-Net 等。

最近 Transformer^[22]席卷计算机视觉领域，而 Segmenter^[23]是基于 Transformer 的一个语义分割模型，他将图像分割成多个块，并将其映射为一个线性序列输入网络。同时，也有文章提出了 Transformer 与 U-Net^[24]的结合，它使用了 transformer 作为 U-Net 的中间件，来完成长距离的建模。但是 Transformer 对于数据量有着更高的需求，若是数据量不够，则很难达到较好的效果。

对于医学影像的语义分割任务，最常用的模型就是 U-Net 家族的模型，DeepLab 家族也在图像分割领域表现优秀。因此本次实验采用了 U-Net 家族的部分网络以及 DeepLab 家族中 Deeplabv3+。这是由于在医学影像分割中，从实践结果来看，U-Net 家族模型的表现是最优秀的，因此选取了多个 U-Net 家族的网络。而在 DeepLab 家族中，只选取了目前在较为成熟的 Deeplabv3+ 网络。下面来介绍一下本次实验所用到的各个网络的大致原理。

2.1.1 U-Net 家族网络

(1) U-Net 网络是经典的编码器-解码器的结构，网络结构如图 2.1(a)所示。在编码器部分，它将输入的图片进行 4 次下采样。而解码器就是将下采样得到的深层语义特征进行 4 次上采样，从而恢复到原来的图片大小。同时在上采样过程中，每次上采样都会将下采样过程中具有相同大小特征通过跳跃连接拼接起来。这样就能够在深层语义与浅层语义相结合，从而让模型具有更好的感受野。

(2) U-Net++也称为 Nested U-Net，顾名思义，这是一个密集型的 U-Net 网络，网络结构如图 2.1(b)所示。它的网络基本结构与 U-Net 一致，也是进行了 4 次下采样以及 4 次上采样。但是不同之处就在于上采样过程中的特征拼接。从图 2.1 可以看出，U-Net++在每次下采样之后，都会又进行一次上采样，然后与下采样之前的特征进行拼接。以此类推，形成一个网状的结构。通过这样的网状结构，在解码器的上采样过程中，用于拼接的浅层语义特征就具有了更强的适应性。通过网状的结构，可以缩小编码器和解码器的特征图之间的差距，使得网络能够更加有效地捕捉到图片中细粒度的前景部分。

(3) U-Net3+^[25]是基于 U-Net++的改进版本，网络结构如图 2.1(c)所示。它不但保留了 U-

Net++的全尺度跳跃连接，还引入了全尺度的深度监督，同时还减少了网络的参数数量，增加了训练效率。从图 2.1 可以看出，在 U-Net3+中，解码器的每一层都与编码器层数小于该解码器层的特征进行直接的跳跃连接。而不是像 U-Net++那样，编码器层的每一层特征都在经过了多次的跳跃连接之后才与解码器层拼接。这样保留了全尺度的跳跃连接，也减少了参数数量。同时 U-Net3+还对于解码器的每一层都使用真实分类图进行监督。

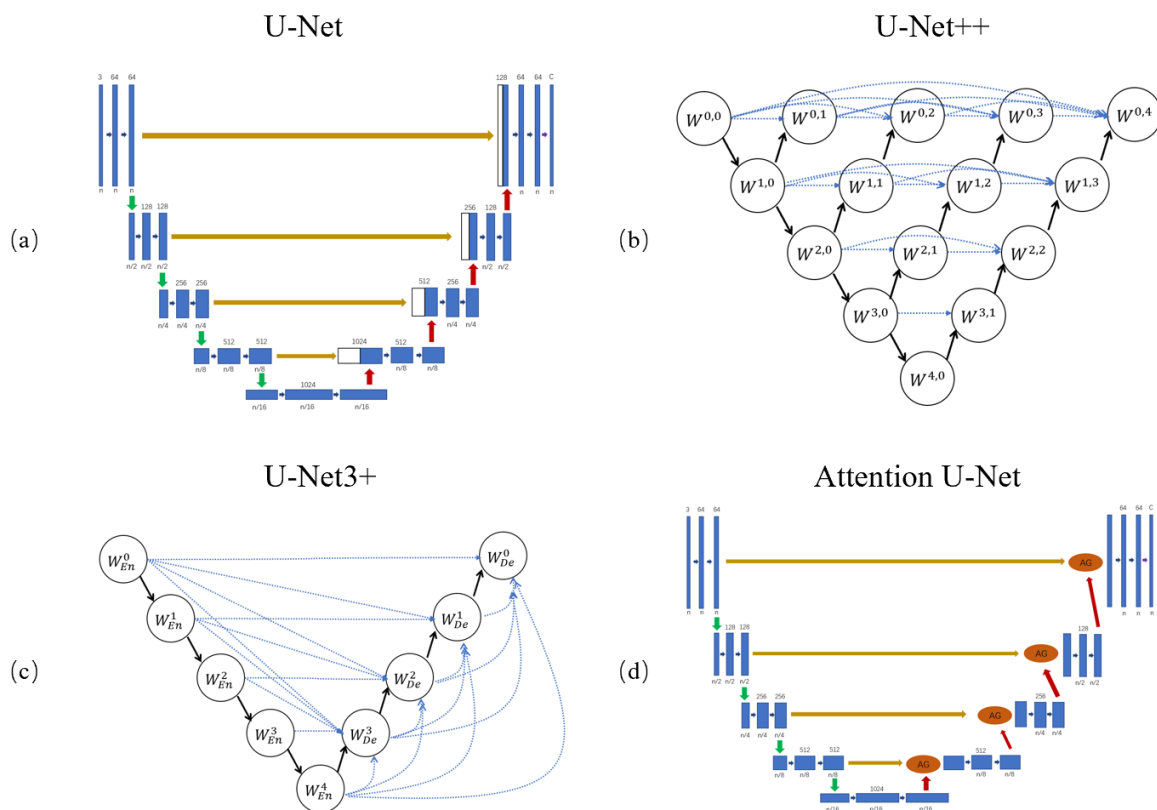


图 2.1 U-Net、U-Net++、U-Net3+和 Attention U-Net 的模型结构

(4) Attention U-Net 是在 U-Net 的基础上加上了注意力机制。网络结构如图 2.1(d)所示，Attention U-Net 将注意力机制添加在了解码器的每一层的跳跃连接处。解码器的每一层将上一层输出以及跳跃连接而来的编码器层做加法，从而实现注意力门控中的门控系数，最后将该门控系数乘上跳跃连接的编码器层，并将结果拼接到相应的解码器层上。这样的机制使得网络能够自动地抑制图像中的无关部分，从而聚焦于目标部分。

2.1.2 DeepLab 家族网络

在 DeepLab 家族中，只选取了 DeepLabv3+进行实验，如图 2.2 所示，它也是一种编码器-解码器的结构，但是与 U-Net 家族的对称式的结构不同。它采用了带有优化版的 Xception 模块的 ResNet-101 作为主干网络，然后将输出送入带有空洞卷积的金字塔池化模块中，再将输出与主干网络的低层次语义特征拼接，最后通过解码器的上采样得到预测图。

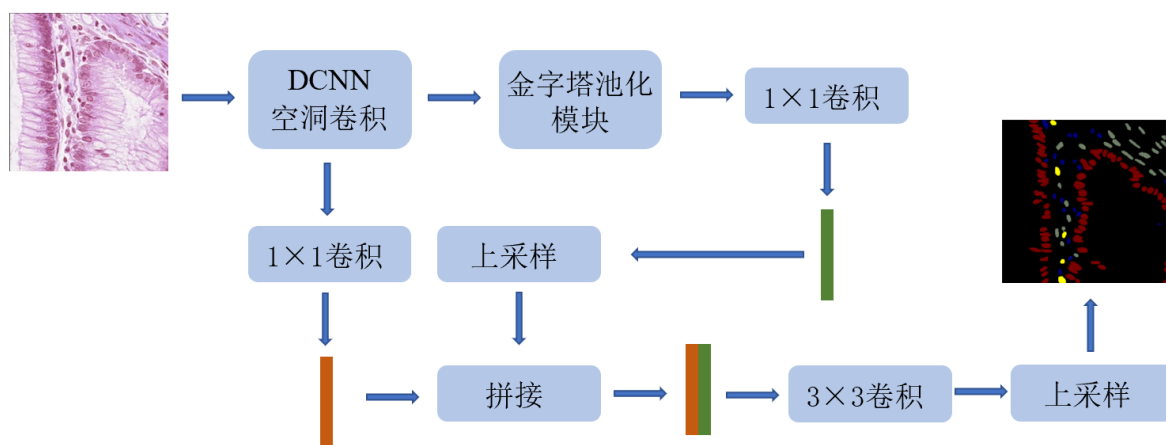


图 2.2 DeepLab v3+的模型结构

2.2 数据不平衡问题的解决方法

2.2.1 数据增强

数据增强能够在不实质性的增加数据的情况下，让有限的的数据产生等价于更多数据的价值，即在原始数据上进行变换操作，使得其能够扩充数据集的大小。常用的数据增强方法可以分为两类，其一是单样本数据增强，其二是多样本数据增强。通过数据增强，就可以增加数据集的大小，增多较少类别的样本数，使数据集回归平衡；或者是成比例地扩大数据集，这样即使某些类别的数据占比较少，但是它们的绝对数量较多，也可以让模型较好地学习它们的特征。

单样本数据增强一般是指在通过对于每个样本本身的几何或颜色的变换以达到增加数据的效果。几何变换指对于图像进行几何变换，包括了翻转、旋转、裁剪、形变、缩放、翻转、拼接等。在对于原始图像进行几何变换的同时也要对于像素分类的标签也进行相同的几何变换。而颜色的变换则是通过对于原始图像添加随机噪声或颜色扰动。对于医学影像而言，几何变换的效果比颜色更加有效，因为医学影像的颜色往往都是染色而产生的，或者就是如 CT 图像一样的灰度图像，因此颜色变换的意义不大。

多样本数据增强是指利用多个样本来产生新的样本。对于不平衡的数据集，可以向较少的数据类别添加数据，通过复制等方式，称为过采样；或者从较多的数据类别中删除部分数据，称为欠采样。这两种方法都会带来较为负面的影响，前者会让模型在训练时容易过拟合，而后者会导致部分重要数据的丢失。为了解决上述欠采样的问题，又提出了带有信息的欠采样。如 Easy Ensemble^[26]，它是一种无监督学习的算法，它对于训练集欠采样多次并训练，从而得到多个分类器，然后将多个分类器组合到一起得到最终的结果；上述方法都是从原数据集中进行下采样减小较多类别的数量或通过复制增加较少类别的数量。而 SMOTE^[27]算法则是人工地创造一些数据，它在较少数量的类别，基于特征空间相似度，在两个数据点连线中心采样，从而人工地创造一些数据。但是这样的方法容易出现过泛化和高方差的问题，因为不能保证在两个相同类别的数据点的连线上的数据点也属于该类别。为了克服 SMOTE 的缺点，Borderline-SMOTE^[28]被提出，它与

SMOTE 会对于所有的较少类别的样本都产生新样本不同，它只对于那些靠近边界的较少类别的样本创造新数据。还有基于聚类的采样算法 CBO^[29]，它利用 K-Means 技术在每个类别内部进行聚类，从而分成子类，再在子类中进行上采样，从而让多数类别和少数类别数量相等。使得生产的数据能够与原始数据尽量保持在同一分布上。

数据增强能够有效地缓解数据不平衡问题，数据不平衡问题往往是因为某些类别的样本的数量太少，网络难以学习到那些数量较少的类别的特征，导致最终模型对于这些数量较少的类别的预测精度较差。而当较少类别的样本得到增加，即使整个数据集还是存在数据不平衡的现象，但是网络有了更多的样本，就能够学习到各类别的特征。

2.2.2 损失函数

损失函数，又称目标函数，是神经网络训练的目标。调整损失函数会直接影响到模型参数的梯度变化趋势，而不同的变化趋势能够让模型倾向于学习不同的特征。对于语义分割问题，往往把它看作是一种逐像素的分类分为，既然是分类问题，那么就可以对于每一个像素都使用经典的交叉熵函数来作为损失函数，然后再将所有的损失值求平均得到最终的损失值，再进行反向传播。这样的方法在数据平衡的语义分割任务中是没有问题的，但是在数据不平衡的任务中却存在问题。在数据极不平衡的数据集中，有些类别的像素占比高达 80%，而有些类别只占 0.1%。在这样极端不平衡的情况下，若是再简单地使用交叉熵函数并求平均的方式来计算损失，会导致在最终的损失值中有些类别的贡献高达 80%，而有些类别只贡献了 0.1%。因而在反向传播的过程中，较少类别的特征对于梯度的变化几乎没有贡献，也就是模型几乎无法学习到较少类别的特征。但是若是能够改变损失函数，使得不同类别之间不会对于损失值的贡献差距过大，就让模型能够较好地学习到每一种类别的特征。因此本次毕业设计探究了四种针对数据不平衡的损失函数，分为两种。其一是基于分布的损失函数，即是逐像素计算损失的损失函数，可视为交叉熵的变体。其二是基于区域的损失函数，这是针对语义分割问题的损失函数，它将预测区域与目标区域的差距作为损失函数，这样计算方法与数据的分布无关，因此数据不平衡问题对其影响较小。

（1）Focal Loss

Focal Loss 是一种基于分布的损失函数，它的提出是用于目标检测的损失函数，而语义分割也可视为一种分类问题，因此 Focal Loss 在语义分割上也十分有效。Focal Loss 的公式如下：

$$FL(p, \hat{p}) = -(\alpha(1 - \hat{p})^\gamma \log(\hat{p}) + (1 - \alpha)\hat{p}^\gamma(1 - p) \log(1 - \hat{p})) \quad (2.1)$$

其中 p 表示实际值， \hat{p} 表示网络的预测值， α 为每个类别的权重， $\gamma > 0$ 称为可调节因子。超参数 α 可以用于控制每个类别对于最终损失的贡献值，可以依据样本在数据集中的比例来设定。正式因为这个超参数，使得 Focal Loss 对于不平衡的数据有着较好的学习能力。而超参数 γ 是用于抑制简单样本，而放大困难样本的指数系数。这让简单样本对于损失值的贡献降低，而提高困难样本的贡献。使得网络能够聚焦于困难样本，而在不平衡的数据集中，往往样本量较少的类别就是困难样本，因此 Focal Loss 能够进一步的解决数据不平衡问题。

公式（2.1）是二分类的 Focal Loss，而本次实验的数据集均为多分类问题，因此改为多分类的 Focal Loss：

$$FL(p_i, \hat{p}) = -\alpha_i(1 - \hat{p})^\gamma \log(\hat{p}) \quad (2.2)$$

其中 p_i 表示实际值为第 i 类， α_i 则表示第 i 类对应的损失权重，其余参数与公式（2.1）相同。在实际的实验中，使用公式（2.2）作为 Focal Loss。

（2）Online Bootstrapping Loss

Online Bootstrapping Loss 是一种在线困难样本挖掘的损失函数，也是一种基于分布的损失函数。公式如下：

$$L = - \sum_{i=0}^{C-1} y_i \log(p_i), p_i \in P \quad (2.3)$$

其中计算损失的部分使用的是交叉熵函数， C 表示总的分类类别， y_i 表示该像素是否属于类别 i ， P 则表示本次损失计算时，网络预测值与实际值差距最大的像素点的集合，该集合的元素个数为 n 。而该距离的计算如下：

$$distance = \frac{1}{C} \sum_{i=0}^{C-1} \sqrt{(p_i - y_i)^2} \quad (2.4)$$

为图片的每一个像素点计算上述距离，从而得到最困难的 n 个像素点。于是在每次计算损失函数时，只对于属于集合 P 的 n 个像素点计算损失函数，并进行反向传播。这样使得网络每次都只聚焦于最困难的像素点。而在不平衡的数据中，往往数据量较少的类别就会成为困难的样本，因此该方法能够一定程度上的解决数据不平衡问题。

（3）Dice 损失函数

Dice 损失函数来自于语义分割中的评价指标之一 Dice 系数，是一种基于区域的损失函数。Dice 系数的定义如下：

$$dice = \frac{2TP}{FP + FN + 2TP} \quad (2.5)$$

其中 TP、FP、FN 分别为混淆矩阵中的真阳性、假阳性和假阴性。因此 Dice 系数可以衡量模型预测区域和真实区域的重叠程度。它的取值范围为[0,1]，越接近 1 则表示预测区域与真实区域的重叠程度越高，即模型的效果越好。而在计算 Dice 系数时，已经将模型的预测值通过设定的阈值转为了标签，这样的转换使得其无法直接作为损失函数。因为通过阈值的直接转换使得其不可微，不能进行反向传播。

为了能够使得 Dice 系数能够直接应用于训练中，去除掉阈值对于预测值的转换，直接将预测值与标签进行计算，公式如下：

$$dice = \frac{2|X \cap Y|}{|X| + |Y|} \quad (2.6)$$

$$|X \cap Y| \approx X * Y \quad (2.7)$$

$$|X| \approx \text{sum}(X), |Y| \approx \text{sum}(Y) \quad (2.8)$$

其中 X 表示预测值， Y 表示实际值， $*$ 符号表示矩阵与矩阵之间按元素相乘， $\text{sum}(\cdot)$ 表示矩阵的元素之和。在训练模型时，使用式（2.6）表示 Dice 系数，由此得到 Dice 损失函数为：

$$Dice Loss = 1 - dice \quad (2.9)$$

使用式（2.9）作为损失函数即可让训练正常进行。而在多分类任务中，则对于每一个类别（包

括背景类）都计算 Dice Loss，并对其求平均得到最终的 Dice Loss。同时该式对于不平衡的数据是不敏感的，它只关注于区域，因此它能作为解决数据不平衡问题的损失函数。

（4）Tversky 损失函数

Tversky 损失函数是基于 Tversky 系数的一种损失函数，也是一种基于区域的损失函数。Tversky 系数的定义与 Dice 系数类似，也是利用了混淆矩阵，公式为：

$$tversky = \frac{TP}{\alpha FP + \beta FN + TP} \quad (2.10)$$

其中 α 和 β 为超参数，用于控制假阳性和假阴性的平衡，TP、FP、FN 值的计算类似于 Dice 系数。因此 Tversky 损失函数的公式如下：

$$Tversky Loss = 1 - tversky \quad (2.11)$$

在不平衡的医学影像数据集中，不平衡的数据往往会导致严重的高精度和低召回率，同时，在医学影像的实际应用中，假阴性比假阳性更难容忍。而 Tversky 损失函数能够通过调整超参数 α 和 β 的值，使得网络着重于假阴性样本的学习，让模型的预测结果能够更好地平衡精度和召回率。

装
订
线

3 复用背景的两阶段方法

3.1 概述

在医学影像数据集的语义分割任务中，往往都是采用单阶段的普通方法，即选取神经网络模型、损失函数、优化器进行训练，然后将网络模型的预测结果作为预测的像素分类图与实际的像素分类图进行比较，计算各类评价指标。而如今许多优化方法都是针对数据集，从神经网络模型的结构、损失函数等角度切入，较少考虑多阶段的方法来优化。而本次毕业设计通过观察针对不平衡数据的损失函数存在的问题提出了一种复用医学图像背景的两阶段方法，通过减少由于损失函数带来的背景预测错误，来提升整个算法的预测精度。

3.2 针对不平衡数据的损失函数存在的问题

针对不平衡数据的损失函数的问题主要在于基于分布的损失函数。在本次实验中，基于分布的损失函数又分为两类，其中一种是预先设定好每个类别权重的损失函数，如 Focal Loss。另一种是在线挖掘的损失函数如 Online Bootstrapping Loss。下面对于这两种损失函数进行讨论。

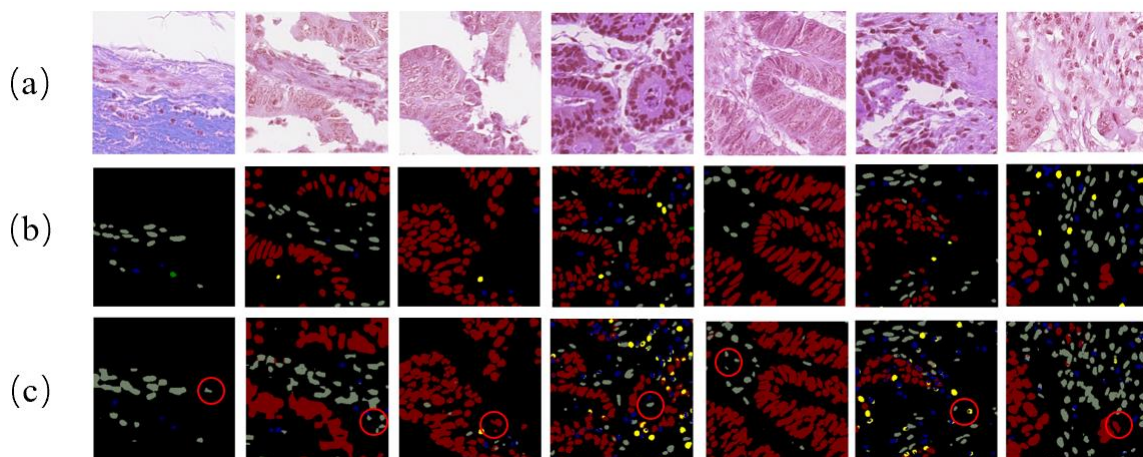


图 3.1 使用 Focal Loss 时，出现的问题。（a）原始图像（b）真实像素分类图（c）预测像素分类图

Focal Loss 之所以能够在一定程度上解决数据不平衡问题，是因为人为地预先为每一个类别设定了权重。一般的权重就是按照每一类所占的比例来分配的，类别所占比例越大，则权重越低。这样才能保证模型对于每一类的学习是平衡的，每一类别对于损失值的贡献程度是大致相当的。但是在最终评价分割效果时，不单单是看那些数量较少的类别，而是整体。而大部分的像素又是由占比较大的类别组成，其中，往往背景是占比最大的类别，尤其是对于细胞图像分割的数据集而言。同时背景的分割也是细胞分割最重要的任务之一。由于预先权重的设定，让网络都聚焦于了非背景类，即前景的各类别细胞上，这就容易导致如图 3.1 中的情况。许多非细胞区域也被误识别为了细胞。

而 Online Bootstrapping Loss 能够解决数据不平衡问题的原因是，对于类别占比较少或极少的

像素，由于数量的原因他们对于损失值的贡献是较少的，所以网络几乎不会去学习他们，因而导致了他们容易成为困难样本。因此网络就会较多地学习占比较少的类别，而较少地学习占比较多的类别。但是问题也随之而来。为了困难样本的代表性和有效性，每次选取的困难样本个数是有限的。这就会导致每次训练都只会关注到较少部分的像素，而其中大部分都还是属于占比较少的类别。如先前提到的，背景往往是占比最大的类，因此背景类的像素对于损失值的贡献就会降低。如表 3.1 展示了在 Online Bootstrapping Loss 中，各个类别的占比。因此大部分的背景像素在训练过程中都不会被选为困难样本，因此网络就无法学习到它们的特征，导致许多背景像素被误分类为了其他的前景类别。

表 3.1 Online Bootstrapping Loss 的各类别困难样本比例

	中性粒细胞	上皮细胞	淋巴细胞	血浆	嗜酸性粒细胞	结缔组织	其他（背景）
训练占比	9.915	16.67	20.18	13.22	4.803	17.55	17.66
实际占比	0.126	10.308	1.855	0.551	0.103	3.081	83.975

因此，当我们在使用这些基于分布的损失函数时，由于其减弱了占比最大的类，即背景在损失值中的权重，同时又增大了其他前景，即各个细胞类。因而导致网络对于背景的学习不够充分，就会让许多原本属于背景的像素，也被预测为了前景。又由于背景的占比最大类，最终导致了如 Dice、mIoU 等评价指标的下降。

3.3 二分类语义分割

在细胞分割的任务中，多分类语义分割是较为困难的任务，因为一般而言，前景细胞类的像素数量较少，存在数据不平衡问题，而在分为多个类别之后，数据不平衡问题愈加严重。此时，网络不仅要区分前景和背景，还需要区分前景细胞类中不同类别的细胞，这就导致了多分类模型无论是前景还是背景上的分类精度都较低。如上所描述的损失函数的问题，得到的多分类分割模型在背景类的预测上容易出现错误。为了能够缓解这两个问题，本次实验又采用了一种二分类的语义分割模型。

首先要对于数据集进行改变，生成新的二分类语义分割的标签。原本的多分类语义分割的标签分为 $\{l_0, l_1, \dots, l_{C-1}\}$ ，其中 C 表示总类别数， l_0 表示背景类。将该标签转换为二分类标签，如式（3.1）所示，将除背景类之外的类别均归为前景类。

$$L = \begin{cases} l_{bg}, l = l_0 \\ l_{fg}, l \neq l_0 \end{cases} \quad (3.1)$$

将上述生成的二分类标签作为语义分割网络训练时的标签。这样的标签分类大大降低了数据集的分割难度。这样网络在学习图像的特征的过程中，不会因为多分类而导致梯度的偏移，能够专注于分割前景与背景，从而让前景和背景，尤其是背景的分割能够有更高的精度。同时将标签只设定为背景和前景会让我们在使用如 Focal Loss 等损失函数时，设定的类别权重与多分类产生很大的不同。如表 3.2 与表 3.3 分别表示了多分类的类别权重和二分类的类别权重，类别权重之

和均为 1。

表 3.2 CoNIC 数据集多分类类别权重

细胞类别	类别权重
中性粒细胞	0.39031
上皮细胞	0.00473
淋巴细胞	0.02632
血浆	0.08854
嗜酸性粒细胞	0.47268
结缔组织	0.01584
其他（背景）	0.00058

表 3.3 CoNIC 数据集二分类类别权重

细胞类别	类别权重
前景	0.83977
背景	0.16023

由此可以看出，在多分类的类别权重中的背景类被大大降低，降低至 0.00058；而二分类中的类别权重仅被降低到 0.16023。由此可见，在二分类的语义分割中，背景类会被强调，因此在背景类的预测上也能够获得更高的精度和召回率。

3.4 复用图像背景

通过分析损失函数存在的问题以及二分类语义分割在背景类上的精度和召回率的提升，本次毕业设计提出了一种复用图像背景的方法来解决由基于分布的损失函数的权重设定带来的背景的错误分类问题。

用带有针对数据不平衡的损失函数训练多分类以及二分类的语义分割模型。其中多分类模型的预测结果能够较为有效地对于前景像素的各个类别的细胞进行分类，但是对于背景像素的分类略有欠缺。而二分类模型的预测结果能够精确地找出背景像素，但是无法对于前景像素进行分类。因此将两者的预测结果结合，让最终的预测结果既可以有效地预测前景，也可以精确地预测背景。结合方法如下式所示：

$$C_{normal}(i) = M_{normal}(I(i)) \quad (3.2)$$

$$C_{binary}(i) = M_{binary}(I(i)) \quad (3.3)$$

$$C_{combine}(i) = \begin{cases} C_{binary}(i), & C_{binary}(i) = C_{bg} \\ C_{normal}(i), & otherwise \end{cases} \quad (3.4)$$

其中， $I(i)$ 表示在图像 I 中位置为 i 的像素， C_{normal} 表示多分类模型的预测结果， C_{binary} 表示二分类模型的预测结果，而 $C_{combine}$ 则是最终的预测结果。式（3.4）表示了，若某个位置的类别

在二分类模型中被预测为了背景，则该位置的分类即为背景类，反之则使用多分类模型对于该位置的预测结果。

3.5 复用背景的两阶段方法在医学影像上的应用

整个复用背景的两阶段方法分为三个部分，分别是训练多分类模型、二分类模型以及结合两者预测结果的算法。下面来一一介绍这三部分的流程，整个流程如图 3.2 所示。

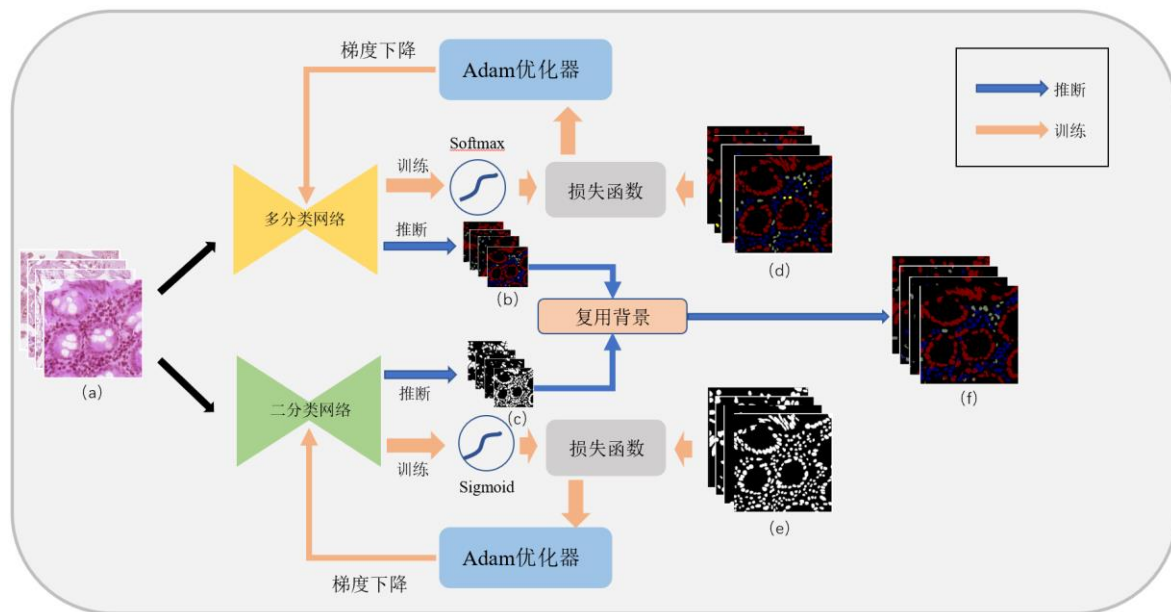


图 3.2 复用背景的两阶段算法流程，（a）原始图像（b）多分类网络的预测结果图
（c）二分类网络的预测结果图（d）多分类标签（e）二分类标签（f）复用背景算法的预测结果

3.5.1 训练多分类语义分割网络

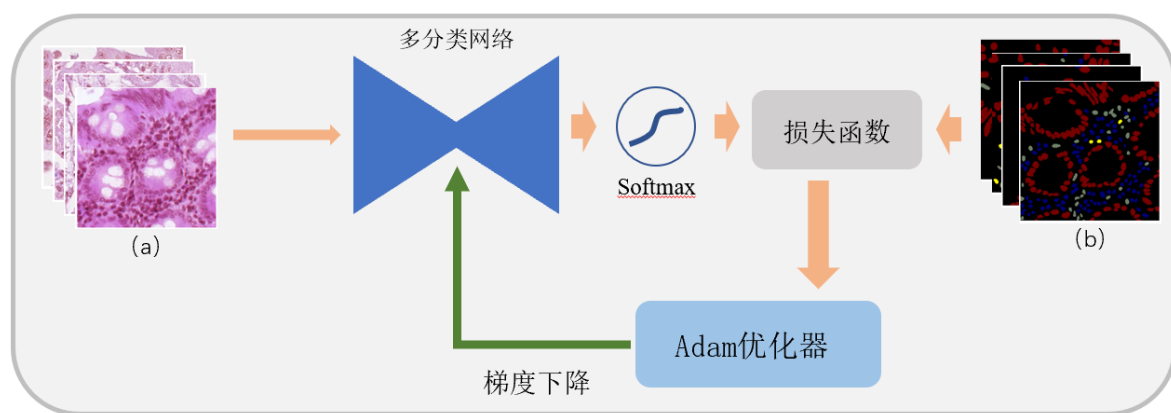


图 3.3 训练多分类网络算法流程（a）原始图像（b）多分类标签

训练多分类语义分割网络包含了四个要素，分别是数据集、神经网络模型、损失函数以及优

化器。整个流程如图 3.3 所示。其中神经网络模型与损失函数不定，共设置了 5 种神经网络模型以及 5 种损失函数及其组合进行实验。通过实验找出表现最好的模型与损失函数的组合。因此在本部分只称其为神经网络模型与损失函数。

首先从数据集中取出图像，图像的数量由批的数量决定，记为 X 。然后将图像传入神经网络进行前向传播。通过神经网络内的一层层的卷积神经网络和激活函数，最终得到神经网络的输出，记为 Z 。然后将该输出经过 *softmax* 函数，如式（3.5）所示，其中 Z_i 表示第 i 个通道的输出。

$$\text{Softmax}(Z_i) = \frac{e^{Z_i}}{\sum_{c=0}^{C-1} e^{Z_c}}, i \in 0, 1, \dots, C-1 \quad (3.5)$$

由于通过神经网络的输出是不保证在 $[0,1]$ 的范围内的，因此为了能够让每个通道的输出可以表示概率，于是使用 *softmax* 将每个通道的输出映射到 $[0,1]$ ，且各个通道之和为 1，这样就将神经网络的第 i 个通道的输出转换为了神经网络对于输入图像是第 i 类的预测概率。在推断时，概率最大的通道的序号，就是神经网络预测该图像类别。

然后将上述 *softmax* 函数输出结果记为 Y ，然后使用上述结果进行损失值的计算。在计算损失值之前首先要对于图像的分割标签图进行处理，由于在数据集中的图形分割标签是记录了每个像素点的类别序号，因此在计算损失时需要先转换为独热码（one-hot）的形式，即将每个标签转换为长度为 C 的向量，并将第 \hat{Y} 个值设为 1，其余均为 0， \hat{Y} 则表示该标签的数值。将标签转换成独热码之后，就可以与 *softmax* 函数的输出结果 Y 进行损失计算，如式（3.6）所示。

$$L = \sum_i \frac{1}{P} \sum_p \text{loss function}^i(Y_p, \hat{Y}_p) \quad (3.6)$$

在式（3.6）中， loss function^i 表示在训练时使用的第 i 种损失函数， \hat{Y}_p 则表示像素 p 的预测值。在基于分布的损失函数中，如交叉熵函数、Focal Loss，则是逐点的计算损失；而在基于区域的损失函数中，如 Dice Loss 和 Tversky Loss，则是逐通道地计算损失。无论是何种损失函数，最终将计算的损失值求平均。若用了多个损失函数则再以多个损失函数之和作为最终的损失值。

在得到损失值之和，使用该损失值进行反向传播。在反向传播时，计算出每个参数的梯度，并使用优化器来更新每个参数。本次实验所使用的优化器是 Adam^[30]优化器，该优化器是一种改良版的梯度下降优化器，结合了 AdaGrad 和 RMSProp 的优点，该优化器会对于模型的每一个参数进行梯度下降的操作。该优化器的计算式如式（3.7）~（3.11）所示。

$$m_t \leftarrow \beta_1 * m_{t-1} + (1 - \beta_1) * g_t \quad (3.7)$$

$$v_t \leftarrow \beta_2 * v_{t-1} + (1 - \beta_2) * g_t^2 \quad (3.8)$$

$$\hat{m}_t \leftarrow \frac{m_t}{1 - \beta_1^t} \quad (3.9)$$

$$\hat{v}_t \leftarrow \frac{v_t}{1 - \beta_2^t} \quad (3.10)$$

$$\theta_t \leftarrow \theta_{t-1} - \alpha * \frac{\hat{m}_t}{(\sqrt{\hat{v}_t} + \epsilon)} \quad (3.11)$$

其中 t 表示模型参数的迭代次数， g_t 表示本次反向传播计算的梯度， m_t 和 v_t 是计算过程中的变量，而 β_1 和 β_2 分别为 m_t 和 v_t 的指数衰减率，是两个超参数，一般可以设 $\beta_1 = 0.9, \beta_2 = 0.999$ ，

α 则表示学习率，也是一个超参数。从式（3.7）和（3.8）可以看出，通过这两个超参数，可以控制当前的 m_t 和 v_t 与之前的 m_{t-1} 和 v_{t-1} 和本次计算的梯度 g_t 之间的关系。这样就拥有了记忆的功能，在每次梯度下降时，不仅是本次计算的梯度会有影响，先前的梯度也会存在一定的影响。从 β_1 和 β_2 的取值也可以看出，每次计算梯度时，更注重先前的取值，而本次计算的梯度只占了很小的比例。而式（3.9）和（3.10）则分别是修正了 m_t 和 v_t 的值，否则通过式（3.7）和（3.8）的计算，会让 m_t 和 v_t 的值越来越小。最终将修正后的 \hat{m}_t 与 \hat{v}_t 应用式（3.11）进行计算最终的梯度，其中用 $\hat{m}_t/(\sqrt{\hat{v}_t} + \epsilon)$ 来代替原始梯度下降中的参数梯度。

通过以上的策略对于模型的参数进行更新，然后再次读取数据集中的下一批数据继续训练，知道模型收敛为止。

3.5.2 训练二分类语义分割网络

二分类语义分割的训练流程与多分类语义分割的训练流程基本相同，如图 3.4 所示。两者都采用相同的神经网络模型、相同的损失函数以及相同的优化器。唯一不同的地方在于，神经网络的输出 Z 不经过 *softmax* 函数。因为是二分类，所以对于图像的每一个像素，神经网络只会输出一个值，因此将其经过 *Sigmoid* 函数，该函数的表达式如式（3.12）所示。

$$\text{Sigmoid}(x) = \frac{1}{1 + e^{-x}} \quad (3.12)$$

通过 *Sigmoid* 函数，就可以将神经网络的输出映射到 $[0,1]$ 之间，从而可以表示该像素是否为前景类别的概率。

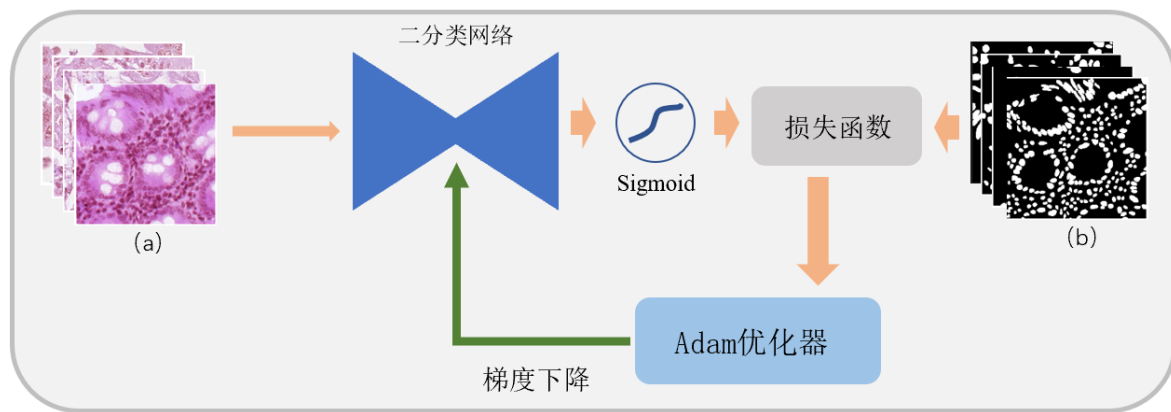


图 3.4 训练多分类网络算法流程，（a）原始图像（b）二分类标签

3.5.3 复用背景生成预测结果

通过上述的训练，得到了两个训练好的模型，分别是多分类模型和二分类模型。使用这两个模型对于数据集进行预测，从而实现复用背景算法，整个流程如图 3.5 所示。

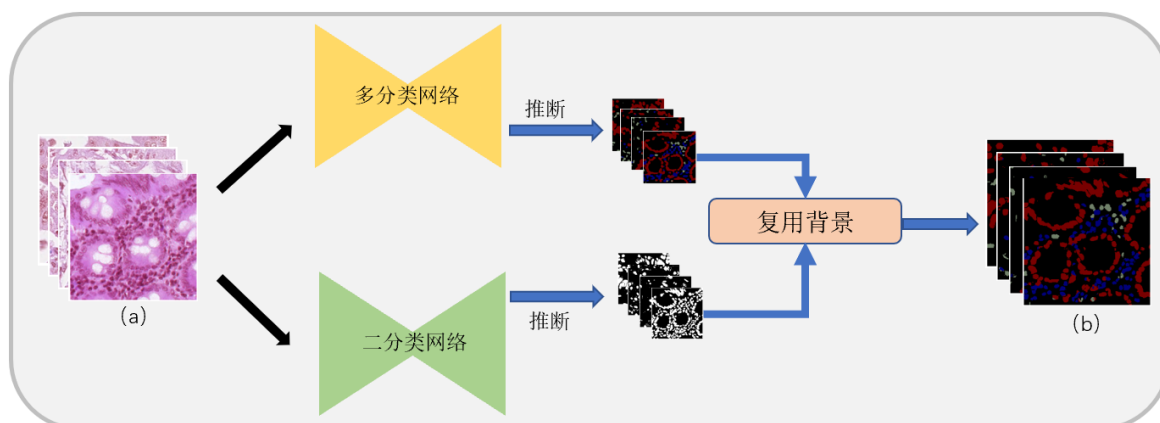


图 3.5 复用图像背景的两阶段算法流程，（a）原始图像（b）复用背景算法的预测结果

首先将数据集中的图像输入到多分类模型中，得到神经网络的输出 Z_{multi} ，然后将该输出送入 $argmax$ 函数。该函数可以找出在每个像素点所对应的 C 个通道中，数值最大的通道序号，也就是多分类模型对于每个像素点所预测的类别。因此就得到了输入图像的多分类预测图。然后再将同一张图像输入到二分类模型中，得到神经网络的输出 Z_{binary} ，对于每个像素，神经网络的输出就只有一个数值，因此设定阈值 $\tau = 0.5$ 。若 $Z_{binary} > \tau$ ，则该像素为前景类，反之，则为背景类。因此又得到了输入图像的二分类预测图。

然后应用在 3.4 节中所描述的复用背景算法，将多分类预测图和二分类预测图相结合，得到最终的预测结果。

4 实验及结果分析

4.1 概述

在这一章节，将详细介绍本次毕业设计的实验流程、实验结果及其分析。本章共分为四个部分，首先是数据集和评价标准的介绍，然后是针对多个模型和损失函数的组合进行筛选。还着重对于各个类别的权重的设定进行了分析。最后是针对提出的复用背景的两阶段方法的实验结果和分析。本次的实验使用了 PyTorch 框架，在 Tesla V100 显卡上进行训练。

4.2 数据集以及评价标准

本次实验采用了两个数据集，分别为 CoNIC 数据集和 MoNuSAC 数据集，均来自于医学影像的机器学习比赛 Grand Challenge 的公开数据集。其中 CoNIC 为结肠细胞核的语义分割数据集，包含了 4981 张图片及其实际的像素分类图。该数据集共有 6 类不同的细胞或组织，再加上背景类，共 7 类。MoNuSAC 数据集则是包括了四种器官的医学影像，包含了 209 张大小不等的图片，共有 4 类不同的细胞，再加上背景类，共有 5 个类别。表 4.1 和表 4.2 分别展示了两个数据集的各个类别的分布情况。在训练时，将 CoNIC 数据集中的图像统一缩放为 256×256 的大小，将 MoNuSAC 数据集统一缩放为 512×512 的大小。还按照 1:9 的比例划分了测试集以及训练集。

表 4.1 CoNIC 数据集各类别分布表

细胞类别	占比 (%)
中性粒细胞	0.126
上皮细胞	10.308
淋巴细胞	1.855
血浆	0.551
嗜酸性粒细胞	0.103
结缔组织	3.081
其他（背景）	83.975

表 4.2 MoNuSAC 数据集各类别分布表

细胞类别	占比 (%)
中性粒细胞	0.363
上皮细胞	10.905
淋巴细胞	4.422
巨噬细胞	1.737
其他（背景）	82.574

为了能够从多个角度分析各个算法组合的优劣，本次的实验采用了多种评价标准，包括 Dice、

mIoU、mAP 以及召回率。为了能够分析算法处理数据不平衡问题的效果，不仅对于总体设定了评价标准，还对于每一类都计算了 Dice、IoU、精度以及召回率。

4.3 模型及损失函数筛选

4.3.1 模型与损失函数的初步筛选

为了筛选出最好的模型以便进行进一步的实验，在这一部分，采用了 5 种神经网络：U-Net，U-Net++，U-Net3+，Attention U-Net 以及 DeepLabv3+。为了保证实验的完成性和有效性，将上述 5 种网络分别与 5 种损失函数相结合。这 5 种损失函数为：交叉熵损失函数，Focal Loss，Dice Loss，Online Bootstrapping Loss 以及 Tversky Loss。对于每一种组合都使用了统一的 Adam 优化器。表 4.3 为模型与损失函数的组合分别在 CoNIC 和 MoNuSAC 数据集上实验结果，由于对于每个数据集的实验都有 25 个组合，为了简化，只列出了最具代表性的 10 种组合。

（1）CoNIC 数据集实验结果

在表 4.3 中，AttU-Net 表示 Attention U-Net。从表 4.3 可以看出，在各方面都最优秀是 DeepLabv3+与 Dice 损失函数的组合。在同样使用 Dice 作为损失函数时，DeepLabv3+与 Attention U-Net 的表现远强于其他模型。而在同样使用 Focal Loss 时，其他模型并没有表现出较大的优势。因此整体来看，选取 DeepLabv3+与 Attention U-Net 作为 CoNIC 数据集的模型。

表 4.3 5 种模型与 2 种损失函数组合在 CoNIC 数据集上的实验结果

组合	Dice	mIoU	mAP	平均召回率
DeepLab / Dice	0.6167	0.2640	0.4043	0.3032
AttU-Net / Dice	0.6045	0.2683	0.3525	0.3521
U-Net3+ / Dice	0.4978	0.1705	0.2049	0.2722
U-Net++ / Dice	0.5248	0.1955	0.2424	0.2746
U-Net / Dice	0.4102	0.2231	0.2996	0.3899
DeepLab / Focal	0.2823	0.1314	0.2944	0.2484
AttU-Net / Focal	0.2957	0.1644	0.2866	0.3018
U-Net3+ / Focal	0.3363	0.1916	0.3487	0.2959
U-Net++ / Focal	0.3962	0.2279	0.3695	0.3395
U-Net / Focal	0.3360	0.1924	0.3400	0.3020

在表 4.4 中，CE 表示交叉熵损失函数，OB 表示了 Online Bootstrapping 损失函数。从表 4.4 中可以看出，使用 Dice 损失函数和 Tversky 损失函数时，Dice 指标和 mIoU 指标都很高。这是因为 Dice 损失函数和 Tversky 损失函数都是基于图像的重叠程度的损失函数，而 Dice 指标和 mIoU 指标也是与图像的重叠程度相关的指标，因此非常契合，也让他们得到比较高的评分。但是 Dice Loss 与 Tversky Loss 是较为相似的损失函数，且得到的实验结构也基本相似，因此后续

实验只保留 Dice Loss。而 Online Bootstrapping 损失函数的表现在各个方面都比较差，这可能是因为在设定困难样本数量时，设定的较少，对于 256×256 的图像，只选择了 512 个像素的损失进行反向传播，导致网络学习到的信息太少，从而导致了效果较差。在之后的部分会对于该损失函数进行细致的研究。而表中最为反常的部分要属交叉熵函数和 Focal Loss 了。Focal Loss 是在交叉熵函数上针对数据不平衡问题的改进，因此 Focal Loss 的效果应该是优于交叉熵的，但是在 Dice 指标和 mIoU 指标上均是远远低于交叉熵。

表 4.4 2 种模型与 5 种损失函数组合在 CoNIC 数据集上的实验结果

组合	Dice	mIoU	mAP	平均召回率
DeepLab / CE	0.5034	0.1678	0.1945	0.2050
DeepLab / Focal	0.2823	0.1314	0.2944	0.2484
DeepLab / Dice	0.6167	0.2640	0.4043	0.3032
DeepLab / Tversky	0.5857	0.2408	0.4074	0.2790
DeepLab / OB	0.4023	0.1076	0.1405	0.1257
AttU-Net / CE	0.5459	0.2218	0.2651	0.2969
AttU-Net / Focal	0.2957	0.1644	0.2866	0.3018
AttU-Net / Dice	0.6045	0.2683	0.3525	0.3521
AttU-Net / Tversky	0.5905	0.2444	0.3279	0.3584
AttU-Net / OB	0.2849	0.0644	0.1540	0.1561

表 4.5 Focal Loss 与交叉熵在 CoNIC 数据集的每个类别的 mIoU 值

组合	中性粒细胞	上皮细胞	淋巴细胞	血浆	嗜酸性粒细胞	结缔组织	背景
DeepLab / CE	0	0.3592	0	0	0	0	0.8682
DeepLab / Focal	0.0131	0.2905	0.1047	0.0268	0.0057	0.0460	0.4319
AttU-Net / CE	0	0.2699	0.316	0.0463	0	0	0.8860
AttU-Net / Focal	0.0071	0.2024	0.1506	0.0609	0.0207	0.1186	0.6167

从表 4.5 中可以找到原因。对于类别占比很少的类，使用交叉熵最为损失函数，会让模型对于这些类完全失去分辨能力，转而会让模型倾向于将像素都预测为那些占比较大的类，尤其是背景，因此背景的 IoU 极高，从而拉高了整体的 mIoU 指标。而 Focal Loss 由于给予了占比较小的类较大的权重，因此让模型能够具有分别这些类的能力，但是也就降低了对于占比较大的类的识别精度，从让整体的 mIoU 值小于交叉熵的结果。但是从实用性而言，利用 Focal Loss 所训练得到的模型比交叉熵更好。从图 4.1、图 4.2 中可以更加直观的感受模型与损失函数的在 CoNIC 数据集上的表现。

通过上述对于损失函数的分析，后续的实验中使用 Focal Loss、Dice Loss 以及 Online Bootstrapping Loss 及其组合进行实验。

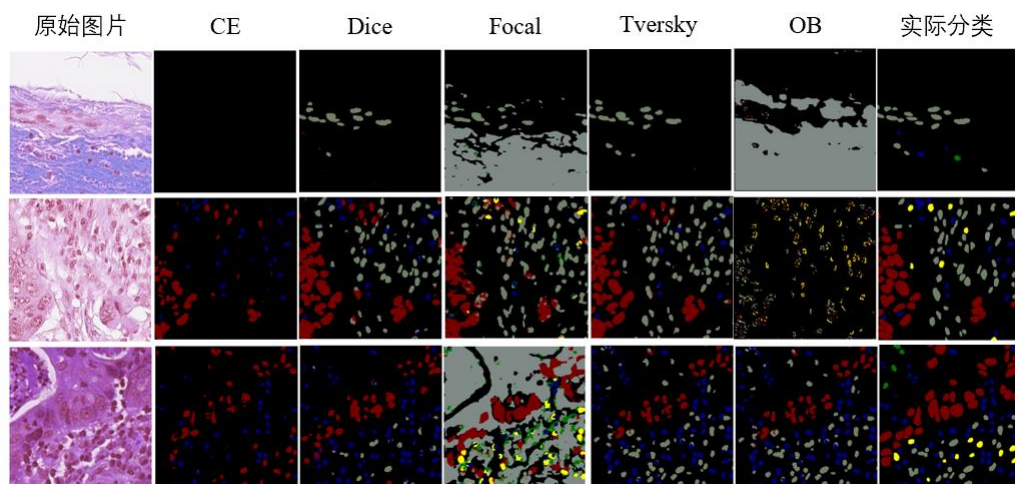


图 4.1 CoNIC 数据集，使用 Attention U-Net 与各个损失函数结合的实验结果图

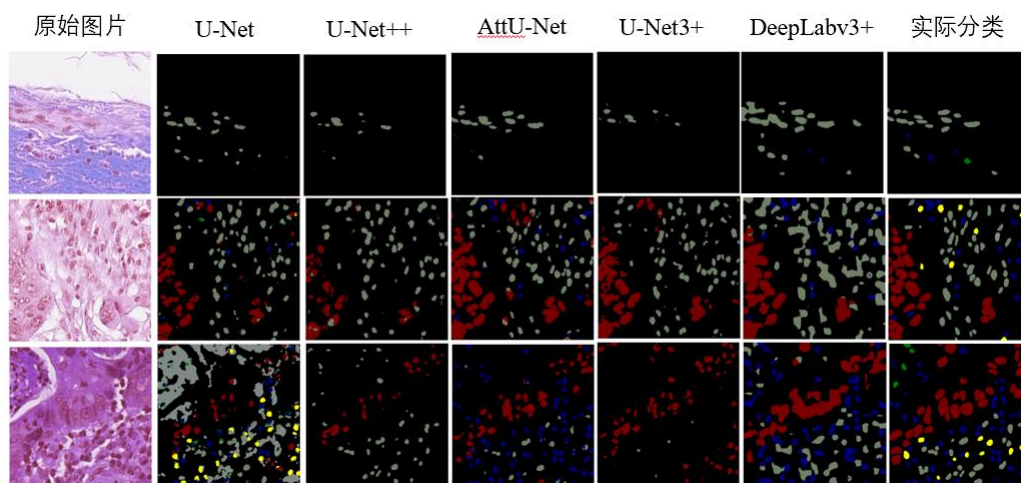


图 4.2 CoNIC 数据集，使用各个神经网络与 Dice Loss 结合的实验结果图

（2）MoNuSAC 数据集实验结果

从表 4.6 可以看出，对于 MoNuSAC，表现较好的模型是 Attention U-Net 与 U-Net。在这一点上与 CoNIC 数据集有所不同。

表 4.6 5 种模型与 2 种损失函数组合在 MoNuSAC 数据集上的实验结果

组合	Dice	mIoU	mAP	平均召回率
DeepLab / Dice	0.5510	0.2328	0.2838	0.3120
AttU-Net / Dice	0.4340	0.2408	0.3495	0.3535
U-Net3+ / Dice	0.3383	0.1987	0.2840	0.2773
U-Net++ / Dice	0.2786	0.2138	0.2650	0.3231
U-Net / Dice	0.3077	0.2390	0.3134	0.3769
DeepLab / Focal	0.2808	0.0370	0.1465	0.0393

续表 4.6

组合	Dice	mIoU	mAP	平均召回率
AttU-Net / Focal	0.3254	0.1324	0.2061	0.2387
U-Net3+ / Focal	0.2951	0.1208	0.2121	0.2505
U-Net++ / Focal	0.3173	0.1515	0.2287	0.2652
U-Net / Focal	0.2475	0.1019	0.1933	0.2303

从表 4.7 上可以看出，在损失函数上的表现与 CoNIC 基本一致，Dice Loss 依旧是最优秀的损失函数。而交叉熵会比 Focal Loss 表现更好的原因，与 CoNIC 也是相同的。因此选取 Dice Loss 与 Focal Loss 两个损失函数进行研究。

表 4.7 5 种模型与 2 种损失函数组合在 MoNuSAC 数据集上的实验结果

组合	Dice	mIoU	mAP	平均召回率
AttU-Net / CE	0.5600	0.2188	0.2746	0.2730
AttU-Net / Focal	0.3254	0.1324	0.2061	0.2387
AttU-Net / Dice	0.4340	0.2408	0.3495	0.3535
AttU-Net / Tversky	0.3065	0.2004	0.2829	0.2877
AttU-Net / OB	0.2110	0.1729	0.2867	0.2408
U-Net / CE	0.5466	0.2237	0.2944	0.2728
U-Net / Focal	0.2475	0.1019	0.1933	0.2303
U-Net / Dice	0.3077	0.2390	0.3134	0.3769
U-Net / Tversky	0.3619	0.2296	0.3343	0.2881
U-Net / OB	0.1969	0.1433	0.1769	0.1732

4.3.1 损失函数组合筛选

通过先前的实验，已经为两个数据集都筛选出了最为适合的两个模型。接下来，利用这两个模型来对于损失函数的组合进行筛选。除了以上已经做过的单独的损失函数的实验，接下来对损失函数的组合进行实验，将之前选择的三种损失函数：Dice Loss、Focal Loss 以及 Online Bootstrapping Loss 两两结合，再三者结合，形成共四种组合。在模型上使用先前筛选出的模型，在两个数据集上分别进行实验，由此得到较为适合的损失函数组合。

(1) CoNIC 数据集

表 4.8 损失函数组合在 CoNIC 数据集上的实验结果

组合	Dice	mIoU	mAP	平均召回率
DeepLab / Dice	0.6167	0.4002	0.6241	0.5765
AttU-Net / Dice	0.6045	0.4000	0.5447	0.6622
DeepLab / Focal	0.2823	0.1466	0.4364	0.2938
AttU-Net / Focal	0.2957	0.1988	0.5447	0.3579
DeepLab / Dice Focal	0.6148	0.3396	0.5908	0.6029↑
AttU-Net / Dice Focal	0.6115	0.4034↑	0.5505	0.6263
DeepLab / Dice OB	0.4591	0.2328	0.3603	0.5599
AttU-Net / Dice OB	0.4178	0.2222	0.2896	0.5252
DeepLab / Focal OB	0.4122	0.1929	0.2556	0.5431↑
AttU-Net / Focal OB	0.2154	0.1254	0.2575	0.2088
DeepLab / Dice Focal OB	0.4503	0.2221	0.3522	0.5433
AttU-Net / Dice Focal OB	0.4258	0.2327	0.3509	0.3967

表 4.8 中的向上箭头表示，在多个损失函数结合之后，评价指标超过了单一的损失函数。从表 4.8 中可以看出，Dice 损失函数依旧是最为适合的函数，其与 Focal Loss 的结合也展现出了非常优秀的水平。而无论何种组合，Online Bootstrapping Loss 的都会让整体的表现下降许多，这可能是在选取困难样本数量上的超参数设置的问题，后续对于该超参数的设置进行实验分析。

表 4.9 损失函数组合在 CoNIC 数据集的每个类别的 mIoU 值

组合	中性粒细胞	上皮细胞	淋巴细胞	血浆	嗜酸性粒细胞	结缔组织	背景
DeepLab / Dice	0	0.4429	0.2676	0	0	0.2858	0.8522
AttU-Net / Dice	0	0.3975	0.3731	0.	0	0.2325	0.8750
DeepLab / Focal	0.0131	0.2905	0.1047	0.0268	0.0057	0.0460	0.4319
AttU-Net / Focal	0.0071	0.2024	0.1506	0.0609	0.0207	0.1186	0.6167
DeepLab / Dice Focal	0	0.4290	0.2657	0	0	0.2671	0.8558
AttU-Net / Dice Focal	0	0.3318	0.3793↑	0.0997↑	0	0.2483	0.8800↑

从表 4.9 可以看出，当 Dice Loss 与 Focal Loss 结合之后，许多类别的指标都有所上升。虽然从整体的效果来看，单独使用 Dice Loss 与 Dice 和 Focal 两者的结合差距不大，但是通过每个类别的 mIoU 可以看出，两者结合的效果更好，尤其是在使用 Attention U-Net 作为模型时。因此对于 CoNIC 数据集，最适合的损失函数就是 Dice 损失函数以及 Dice 和 Focal 的组合损失函数。

（2）MoNuSAC 数据集

表 4.10 损失函数组合在 MoNuSAC 数据集上的实验结果

组合	Dice	mIoU	mAP	平均召回率
U-Net / Dice	0.3077	0.2390	0.3134	0.3769
AttU-Net / Dice	0.4340	0.2408	0.3495	0.3535
U-Net / Focal	0.2475	0.1019	0.1933	0.2303
AttU-Net / Focal	0.3254	0.1324	0.2061	0.2387
U-Net / Dice Focal	0.2991	0.2237	0.2967	0.2728
AttU-Net / Dice Focal	0.2909	0.2250	0.2785	0.3725

从表 4.10 可以看出，最为适合的是 Dice Loss，比 Dice Loss 与 Focal Loss 的结合更加优秀。而 Focal Loss 在该数据集上的表现远不如 Dice Loss，在 Focal Loss 的权重实验部分，会对于原因进行分析。

4.4 各类别权重函数的实验结果与分析

从上述的实验可以看出，Focal Loss 的表现并不尽如人意，因此希望通过使用不同的权重函数来对于 Focal Loss 进行调整。在先前的实验中，使用的权重函数为平衡权重函数，公式如下所示：

$$W_i = \frac{\sum M_i}{N * M_i}$$

其中 W_i 表示第 i 类的权重， M_i 表示第 i 类的样本数量， N 表示像素类别数量。然后对以上式子得到的 W 做归一化处理。通过上式得到的结果是平衡的权重，这也导致了背景类以及另外几个占比较大的类的权重被极大的压缩，使得网络对于这些类别不够敏感。而在进行评价时，是要对于全部的像素进行统一的评价，这就会让这些类别的像素的不足影响到整体的效果。因此考虑使用多种权重函数进行实验，以找出最适合的权重函数。

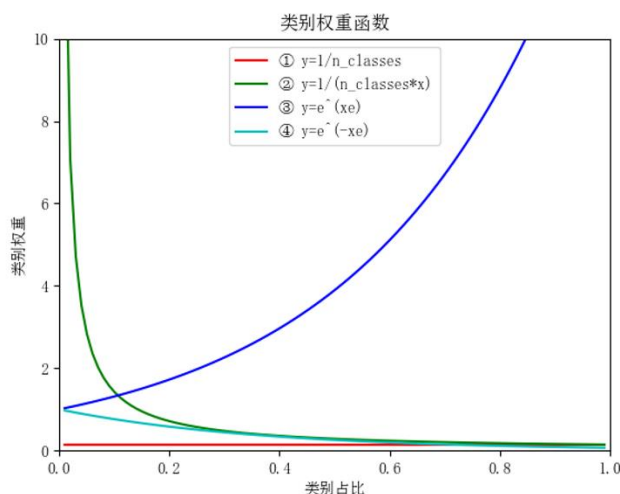


图 4.3 四种类别权重函数曲线

图 4.3 中描述了 4 种不同的权重函数。其中①号权重函数就是普通的平均权重函数，即每一类别的权重均相等。而②号权重函数就是先前的实验中一直使用的权重函数。可以看出该权重函数是反比例函数，在接近于 0 的地方权重非常大，而接近于 1 的地方非常小。虽说这样的设定能够让整体的权重达到平衡，但是也会让网络无法充分地学习那些占比较大，权重较小的类别。因此在本次实验中又再加入了两种指数形式的权重函数。指数函数在自变量处于 0-1 之间的的梯度较缓，能够较大程度上缓解反比例函数所带来的问题。所设定的两种指数函数分别是③号以及④号，其中一个为单调递增函数，一个是单调递减函数。其中单调递增的③号压缩了原本类别比例的差距，但是保持了类别占比的大小关系。而单调递减的④号函数，则是遵从了平衡权重函数的思想，但是又缓解了权重差距过大的问题。

对于④号函数，又再次进行了变换，不是直接使用④号函数，而是在其基础上衍生出另外两种函数，分别称为⑤号函数和⑥号函数。这两种函数都是针对背景类的权重进行了微调。其中⑤号函数的生成方法如下：首先将除背景类之外的类别的占比进行归一化，即计算出每一类在前景中的占比，记为 $\{p_{fg1}, p_{fg2}, \dots, p_{fg(n-1)}\}$ ，其中 n 表示总类别数。再将该占比通过④号函数的变换得到前景类的权重 $\{w_{fg1}, w_{fg2}, \dots, w_{fg(n-1)}\}$ 。然后将原始的背景占比也通过④号函数变换，只取其背景类权重 w_{bg} 。将该背景类权重与先前计算得到的前景权重拼接得到最终的类别权重： $\{w_{bg}, w_{fg1}, w_{fg2}, \dots, w_{fg(n-1)}\}$ 。如表 4.11 中记录的权重可以看出，由于④号权重函数在占比小于 0.1 时，所得到的权重基本相同，差距不大，都在 0.16 左右，因此在前景的分类上，无法解决数据不平衡问题，只是降低了背景的权重。而通过 5 号权重函数的变换，可以看到在前景类中，各个类别的差距有了明显的变化，能够解决数据不平衡问题。而⑥号权重函数的计算方法与⑤号类似，也是先计算出每一类在前景中的占比 $\{p_{fg1}, p_{fg2}, \dots, p_{fg(n-1)}\}$ ，然后通过④号函数变换为权重 $\{w_{fg1}, w_{fg2}, \dots, w_{fg(n-1)}\}$ ，然后将背景类的权重设为 0，从而得到最后的权重 $\{0, w_{fg1}, w_{fg2}, \dots, w_{fg(n-1)}\}$ 。这样的方法与得到的结果与⑤号权重函数在前景类上基本相同，只是不再让背景参与到反向传播中。

表 4.11 ④号权重函数及其衍生计算所得的 CoNIC 类别权重

细胞类别	④号权重函数	⑤号权重函数	⑥号权重函数	占比 (%)
中性粒细胞	0.1746	0.2189	0.2240	0.126
上皮细胞	0.1324	0.0389	0.0398	10.308
淋巴细胞	0.1666	0.1633	0.1671	1.855
血浆	0.1726	0.2037	0.2084	0.551
嗜酸性粒细胞	0.1747	0.2198	0.2249	0.103
结缔组织	0.1611	0.1326	0.1357	3.081
其他（背景）	0.0179	0.0228	0	83.975

在最终的实验上，采用了①②③⑤⑥，这 5 种权重函数在两个数据集上进行实验，使用统一的 Attention U-Net 网络。在实验时，会将上述各类别权重进行归一化，使得权重之和为 1。

表 4.12 各权重函数在 CoNIC 与 MoNuSAC 数据集上的实验结果

CoNIC				MoNuSAC		
数据集	CoNIC			MoNuSAC		
函数编号	mIoU	mAP	平均召回率	mIoU	mAP	平均召回率
①	0.2105	0.2534	0.2919	0.1903	0.2404	0.2441
②	0.1758	0.2959	0.3088	0.1308	0.1959	0.2350
③	0.1200	0.1429	0.1200	0.1643	0.2000	0.1643
⑤	0.2768	0.3611	0.3911	0.1917	0.2613	0.2608
⑥	0.0482	0.2561	0.0590	0.0358	0.1033	0.0622

从表 4.12 中的数据可以看出，在 CoNIC 和 MoNuSAC 两个数据集中，⑤号权重函数在三个指标上均达到了最好的效果，由此可见⑤号权重函数是最为有效的权重函数。从图 4.4 中可以看到每一种权重函数在 MoNuSAC 数据集上的实验结果。

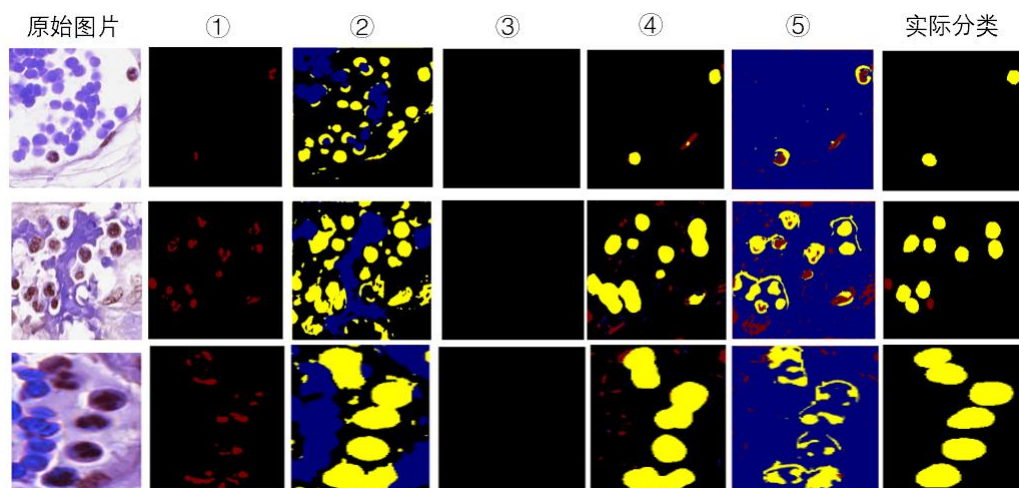


图 4.4 5 种权重函数在 MoNuSAC 数据集上的实验结果

4.5 困难样本数量的实验结果与分析

Online Bootstrapping Loss（下简称为 OB Loss）作为一种在线挖掘的损失函数，又是基于交叉熵函数，应该要比交叉熵函数的表现更加优秀。但是在先前的实验中，OB Loss 的表现非常糟糕，几乎让网络失去了学习能力，因此本部分对于 OB Loss 的困难样本数量进行调整实验，并分析原因。

在先前的实验中，困难样本的数量都是定在 512 个，但是对于 256×256 或 512×512 的图片而言，512 个像素所包含的有效信息太少了，因此分别再设定了 1024、2048、4096、8192、16384、32768、65536 等多种困难样本，让网络能够学习到更多的信息。

从表 4.13 中的结果可以看出，对于 CoNIC 数据集，当困难样本量设定为 4096 时，效果最

好；而对于 MoNuSAC 数据集，困难样本量设定为 1024 时效果最好。

表 4.13 各困难样本数量在 CoNIC 与 MoNuSAC 数据集上的实验结果

数据集		CoNIC			MoNuSAC		
困难样本数	mIoU	mAP	平均召回率	mIoU	mAP	平均召回率	
512	0.1075	0.1607	0.1928	0.0907	0.1107	0.2387	
1024	0.1358	0.1843	0.1916	0.1851	0.2758	0.2654	
2048	0.1269	0.1591	0.1492	0.1573	0.1918	0.1681	
4096	0.1739	0.2327	0.2680	-	-	-	
8192	0.1573	0.1972	0.2319	0.1202	0.1506	0.1644	
16384	0.1199	0.1427	0.1324	0.1382	0.1707	0.1676	
32768	-	-	-	0.1148	0.1544	0.1988	
65536	-	-	-	0.1650	0.2003	0.1844	

表 4.14 OB Loss 在 CoNIC 数据集上的困难样本比例

类别\困难样本量（%）	512	1024	2048	4096	8192	16384
中性粒细胞	9.915	6.051	3.497	1.930	0.983	0.492
上皮细胞	16.67	14.41	18.87	25.99	35.08	31.38
淋巴细胞	20.18	19.57	26.71	10.92	7.389	5.890
血浆	13.22	13.43	14.34	7.690	4.431	2.215
嗜酸性粒细胞	4.803	3.642	1.608	1.068	0.576	0.408
结缔组织	17.55	20.90	12.78	22.94	20.64	11.48
其他（背景）	17.66	22.00	22.19	29.46	30.90	48.14

为了能够更加深入的理解 OB Loss，还对于 OB Loss 在计算损失时，各个类别作为困难样本的比例进行了研究，如表 4.14 所示。可以看出 OB Loss 在选取困难样本时，也基本上能够选取较多的占比较小的类，因此具有一定的解决数据不平衡问题的能力。而随着困难样本量的增大，背景类的占比也随之增大，而前景类的占比不断减小。尤其是如中性粒细胞和嗜酸性粒细胞这两种的占比更是成倍的减少。而从表 4.14 可以看出，随着总的困难样本量的增大，虽然这些类别的像素被选为困难样本的数量也在增加，但是只有少量的增大，甚至到后面以及不增加了。其实这是因为这已经是该类别像素的总数了，即已经将所有该类别的像素都作为了困难样本供模型学习。可以很明显的看出，当总的困难样本量设定为 4096 时，基本上这些类别的像素已经几乎都被选为了困难样本。因此此时 OB Loss 的表现效果最好。而当再增加总的困难样本量时，仅仅只是增加了背景的困难样本数，反而让网络着重于背景类的学习，因此最终的效果越来越差。

表 4. 14 三种 CoNIC 数据集的细胞的被选为困难样本的像素数量

类别\困难样本量	512	1024	2048	4096	8192	16384
中性粒细胞	227574	277775	321107	354354	361171	361171
血浆	303458	616194	1316680	1412097	1627385	1627385
嗜酸性粒细胞	110254	167178	147614	196056	211449	299932

4.6 复用背景的两阶段方法的实验结果与分析

通过先前的实验，对于两个数据集都已经得到了最为合适的算法组合，因此在本部分将利用先前的算法组合在复用背景的两阶段方法上进行实验，并分析结果。

表 4. 15 CoNIC 数据集的多分类模型训练的算法组合

组合编号	神经网络	Focal	Dice	OB
1	AttU-Net	-	√	-
2	AttU-Net	-	√	√
3	AttU-Net	√	√	-
4	AttU-Net	√	√	√
5	DeepLabv3+	-	√	-
6	DeepLabv3+	-	√	√
7	DeepLabv3+	√	√	
8	DeepLabv3+	√	√	√

表 4. 16 算法组合在 CoNIC 数据集上的实验结果

组合编号	Dice	mIoU	mAP	平均召回率
1	0.5938	0.2663	0.3546	0.3659
2	0.4872	0.1619	0.2222	0.2250
3	0.6008	0.2656	0.3511	0.3710
4	0.5022	0.1802	0.2168	0.3335
5	0.6234	0.2686	0.3977	0.3114
6	0.4563	0.1373	0.1636	0.2250
7	0.6227	0.2703	0.4010	0.3093
8	0.4534	0.1342	0.1578	0.2202

4.6.1 多分类模型训练

（1）CoNIC 数据集

在多分类模型训练中，依照先前的多方实验，对于数据集 CoNIC 选择了共 8 种算法组合，如表 4.15 所示。其中 Focal Loss 的类别权重使用的是在 4.4 部分中所介绍的⑤号权重函数；而 OB Loss 的困难样本数量设定为表现最好的 4096。实验的结果如表 4.16 所示，可以看出，表现最好的算法组合是 DeepLabv3+、Focal Loss 与 Dice Loss 的组合。

表 4.17 MoNuSAC 数据集的多分类模型训练的算法组合

组合编号	神经网络	Focal	Dice	OB
1	AttU-Net	-	√	-
2	AttU-Net	-	√	√
3	U-Net	-	√	-
4	U-Net	-	√	√

表 4.18 算法组合在 MoNuSAC 数据集上的实验结果

组合编号	Dice	mIoU	mAP	平均召回率
1	0.4340	0.2408	0.3495	0.3535
2	0.4773	0.1774	0.2238	0.2132
3	0.3077	0.2390	0.3134	0.3769
4	0.4602	0.1930	0.2619	0.3035

（2）MoNuSAC 数据集

而对于数据集 MoNuSAC，选择了 4 种算法组合，如表 4.17 所示。OB Loss 所使用的困难样本数为 4096。从表 4.18 中的实验结果可以看出，Dice Loss 是最适合 MoNuSAC 的损失函数。而 Attention U-Net 与 Dice Loss 的组合是最适合的。

4.6.2 二分类模型训练

医学影像的二分类问题是比较多分类问题简单许多的问题，模型只需要学习背景和前景两种，而图像中的细胞与背景有着较为明显的差别，因此模型能够较为容易地达到较好的效果。因此对于二分类模型，直接采用在多分类模型中表现最好的算法组合来训练。表 4.19 记录了两个数据集的二分类模型的评价指标。

表 4.19 两个数据集的二分类模型实验结果

数据集	Dice	mIoU	mAP	平均召回率	背景精度	背景召回率
CoNIC	0.6681	0.5206	0.8543	0.7756	0.8319	0.9849
MoNuSAC	0.6247	0.4721	0.7788	0.7959	0.8366	0.9552

从表 4.19 可以看出，两个数据集的二分类模型均达到了较好的效果，尤其是在背景的分类

上。模型对于背景类别的分类精度较高，而召回率更是达到了 98.5%与 95.5%。而两个数据集的 F1 值分别达到了 0.9 与 0.89。因此该二分类模型的背景是可信度很高的，复用该模型的背景可以提高算法的背景精度。

4.6.3 复用背景

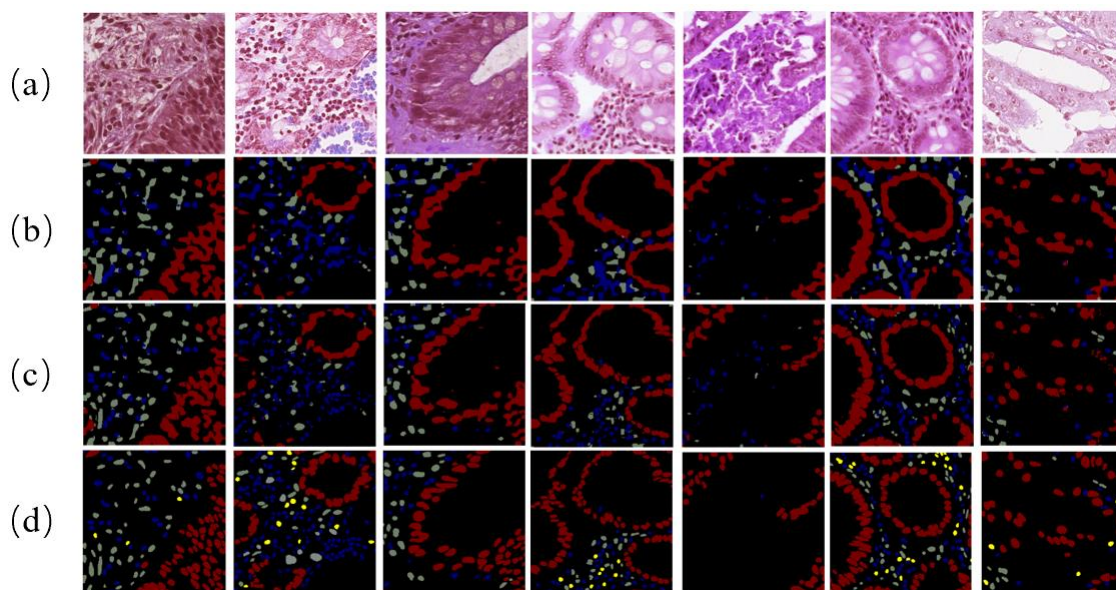


图 4.5 复用背景算法在 CoNIC 数据集上的实验结果，（a）原始图像（b）普通算法（c）复用背景算法（d）真实分类图像

利用多分类模型与二分类模型的预测结果，按照先前已经描述过的算法，生成最终的结果，如图 4.5 所示。

表 4.20 CoNIC 数据集的复用背景算法的实验结果对比

算法	Dice	mIoU	mAP	平均召回率
普通	0.6227	0.2703	0.4010	0.3093
复用背景	0.6509↑	0.300	0.3915	0.3611

表 4.21 MoNuSAC 数据集的复用背景算法的实验结果对比

算法	Dice	mIoU	mAP	平均召回率
普通	0.4340	0.2408	0.3495	0.3535
复用背景	0.4655	0.2581	0.3716	0.3789

从表 4.20 与 4.21 可以看出，当使用了复用背景的算法之后，几乎每种指标都比原先的表现效果最好的算法组合还提高了平均 3 个百分点左右。从图 4.5 中也可以看出，复用背景的算法能

够让原本由于数据不平衡算法而导致的部分背景类像素被预测为前景的问题得到缓解，通过二分类的较为准确的预测结果，去除掉那些被误分类为前景的像素。

装
订
线

5 演示程序设计与实现

5.1 概述

对于上述的各部分算法，设计并实现了一个演示程序。该程序包括了以下功能：使用多种神经网络模型以及多种损失函数的组合对于多分类和二分类的医学影像语义分割模型进行训练。可以通过参数的设置，选择不同的网络模型和不同的损失函数，同时对于 Focal Loss 和 Online Bootstrapping Loss 都可以自定义地选择不同的类别权重或不同的困难样本数。还可以自定义数据集的路径。该程序还可以通过参数的设定对于训练好的模型进行预测，同时会将预测的结果图保存到本地。除了训练模型和预测之外，本程序还实现了由本次毕业设计提出的复用背景的两阶段算法。

5.2 总体设计

该程序的主要功能是根据输入的设定对于多分类及二分类的语义分割模型进行训练、预测。同时保存训练和预测的结果。再通过复用背景的算法得到最终的预测结果。因此将该程序分为以下模块：顶层模块、参数解析模块、多分类模块、二分类模块、复用背景模块。模块之间的关系如图 5.1 所示。

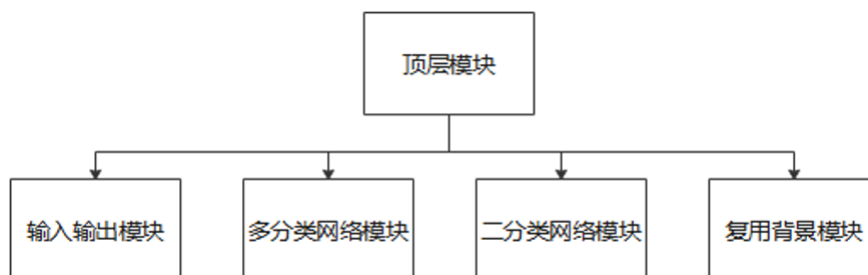


图 5.1 演示程序总体设计图

（1）顶层模块

总体功能：通过输入输出模块所获取到的设定参数，控制整个程序的运行

描述：通过输入输出模块获取到的模型、数据集、损失函数、训练设定等相关参数，控制多分类模块、二分类模块中训练、预测、保存的执行，同时也控制复用背景模块的执行。

（2）参数解析模块

总体功能：分析输入的程序参数，并得到相应的结果

描述：对于程序输入的主函数参数进行分析，得到每个参数的分析结果。并将该分析结果传递给顶层模块。

（3）多分类模块

总体功能：通过顶层模块传递的参数，对于所设定的算法组合在设定的数据集上进行训练、预测以及保存结果。

描述：通过顶层模块控制，设定数据集、模型、损失函数、损失函数的特定参数、预测、训

练及预测数据的保存文件夹等信息，根据上述信息进行多分类模型的训练，使用该模型进行预测，并保存训练时的数据和预测的结果。

（4）二分类模块

二分类模块的功能和描述与多分类模块类似，不再赘述，区别在于训练的是二分类模型。

5.3 详细设计

5.3.1 顶层模块

（1）创建文件夹功能

每次运行该程序，需要创建文件夹用于保存训练、预测时的数据。需要为多分类和二分类的训练数据和预测结果创建文件夹。还需要为复用背景算法创建结果图的文件夹。

（2）生成自定义选项功能

本程序是一个高度自定义的程序，因此该功能实现了从程序参数到实际设定之间的转换，包括多分类和二分类的模型、数据集、损失函数及其参数、优化器、学习率、批的大小、训练次数等。

（3）调用训练、预测功能

该模块通过自定义的选项来控制二分类、多分类以及复用背景的调用。可以只调用其中的某一块进行单独的训练或预测。

5.3.2 参数解析模块

该模块的功能为解析输入到主函数中的参数，具体的参数如表 5.1 所示。

表 5.1 演示程序的主函数参数设定

编号	参数	含义
1	--multi-class	是否训练多分类网络
2	--binary-class	训练二分类网络
3	--predict	使用模型进行预测
4	--epochs	训练次数
5	--batch-size	批的大小
6	--learning-rate	学习率
7	--device	使用的 GPU 或 CPU
8	--m_net	多分类神经网络
9	--b_net	二分类神经网络
10	--save	保存模型参数
11	--dataset	数据集路径
12	--validation	测试集在数据集中的比例

续表 5.1

编号	参数	含义
13	--train-set	训练集路径
14	--test-set	测试集路径
15	--m_focal_loss	多分类网络 Focal Loss
16	--m_dice_loss	多分类网络 Dice Loss
17	--m_tversky_loss	多分类网络 Tversky Loss
18	--m_ob_loss	多分类网络 Online Bootstrapping Loss
19	--b_focal_loss	二分类网络 Focal Loss
20	--b_dice_loss	二分类网络 Dice Loss
21	--b_tversky_loss	二分类网络 Tversky Loss
22	--b_ob_loss	二分类网络 Online Bootstrapping Loss
23	--optimizer	优化器
24	--focal_weight	Focal Loss 的类别权重函数编号
25	--hard_num	Online Bootstrapping Loss 的困难样本数
26	--combine	运用复用背景的算法

5.3.3 多分类模块

（1）训练功能

根据设定的训练次数、数据集、批的大小、神经网络模型、损失函数、优化器、学习率进行训练。

（2）评价功能

在对于整个数据集训练完一次之后，对于网络模型进行评价，使用指标 Dice、mIoU、mAP 以及平均召回率进行评价。

（3）预测功能

当训练完毕之后，使用模型进行预测，同时将预测的结果保存在指定的文件夹中。

（4）保存功能

如果在参数中设定了需要保存，则在每次训练之后将模型的评价结果以及模型参数保存到指定的文件夹中。

5.3.4 二分类模块

二分类模块的功能与多分类模型基本相同，不再赘述。

5.4 输入输出设计

输入: `python demo.py --multi-class 1 --binary-class 1 --combine 1 --predict 1 --m_net attunet --b_net unet++ --m_dice_loss 1 --b_dice_loss 1 --dataset CoNIC`

以上输入表示在 CoNIC 数据集上进行训练,对于多分类网络使用 Attention U-Net 与 Dice Loss 的组合,对于二分类网络使用 U-Net++与 Dice Loss 的组合。在训练之后会保存二分类与多分类网络的模型参数以及模型的预测结果。同时最后应用复用背景的背景算法,并保存结果。所有的结果将会保存到以程序的运行时间为命名的文件夹中。

```

Dataset Info:
  Path: MoNuSAC
  Validation Ratio: 0.1
Training Info:
  Device: cuda:0
  Epoch: 10
  Batch Size: 1
  Learning Rate: 0.001
  Multi:
    Loss: Dice
    Net: attunet
  Binary:
    Loss: Dice
    Net: unet++
  Optimizer: Adam
Predict: Yes
Save: Yes
  Save Path: results/2022-05-14 14:24:26
Start training multi-class model
mIoU: 0.2319: 100%|██████████| 10/10 [08:52<00:00, 53.28s/it, loss (batch)=0.86]
Predicting: 100%|██████████| 20/20 [00:03<00:00, 6.11it/s]
Finish predicting
Save predictions in results/2022-05-14 14:24:26/multi_class
Finish training multi-class model
Start training binary-class model
70%|██████████| 7/10 [10:26<04:21, 87.33s/it, batch loss=0.414]
    
```

图 5.2 演示程序的输出提示

输出如图 5.2 所示,其中包括了①输入参数的说明,②训练时的进度、损失值、评估值和剩余所需时间,③当前训练的阶段。

当训练完毕之后,会将模型的参数保存,还会将多分类模型、二分类模型、复用背景的背景算法的三种预测结果保存到指定文件夹。保存结果的目录结构如图 5.3 所示。

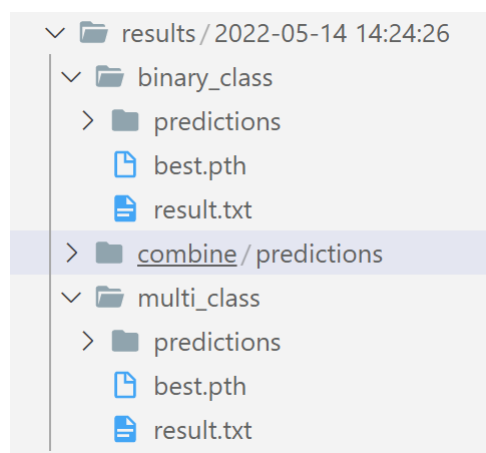


图 5.3 演示程序保存训练模型参数及预测结果的文件目录结构

6 结论和展望

6.1 结论

（1）通过多种损失函数组合在两个数据集上的实验，可以看出对于该类数据极不平衡的细胞核分割数据集，基于区域的损失函数的表现要优于基于分布的损失函数。即使对于 Focal Loss 的类别权重进行调整实验，也没能优于 Dice Loss 的表现。这可能是因为基于分布的损失函数是将语义分割任务简单地视为了每一个像素的分类问题，而没有考虑到像基于区域的损失函数那样区域性的计算损失，因此效果较差。

（2）通过多种 Focal Loss 的类别权重函数在两个数据集上的实验，可以看出，对于极不平衡的数据集，采用平衡权重函数会使模型对于较多类别的特征学习不够充分。因此采用新的较为平缓的权重函数，来缓解这个问题，使得各个类别之间能够更好地平衡。新的权重函数能够在多个指标上比平衡权重函数有所提升。对于 CoNIC 数据集，能够提升 8%；对于 MoNuSAC 数据集，能够提升 2%。

（3）通过对于在线困难样本数量的探究，发现困难样本主要集中于较少类别的像素，这一趋势与 Focal Loss 的权重函数相近。同时当选取的困难样本数量能够使得在训练时，正好每次训练都能够覆盖到大部分的较少类别样本时，模型的表现效果最好。

（4）复用背景的两阶段算法能够缓解由于针对数据不平衡的损失函数所带来的对于较多类别（背景）的误预测问题。结合二分类模型对于背景预测的高精度和多分类模型对于前景类别的区分，获得较好的分割效果。能够在多分类模型的各个指标的基础上再提升 3%。

6.2 展望

（1）本次只对于损失函数进行了探究和实验，而并没有探究解决数据不平衡问题的另一类方法——数据增强。因此如果采用合适的数据增强的方法，应该能够让最终的性能有更多的提升。尤其是可以考虑 GAN 相关的数据增强方法，包含较少类别像素的图像，来缓解数据不平衡问题。

（2）因为主要是对于损失函数的探究，因此本次实验只选取了经典的语义分割模型，对于最新的模型如 KiU-Net、TransU-Net 并没有进行实验。如果能够结合各个最新的模型对于图像特征的提取模块，使得模型本身对于数据不平衡能够敏感，也能够提升模型的性能。

参考文献

- [1] Graham S, Jahanifar M, Azam A, et al. Lizard: A Large-Scale Dataset for Colonic Nuclear Instance Segmentation and Classification[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 684-693.
- [2] Graham S, Jahanifar M, Vu Q D, et al. CoNIC: Colon Nuclei Identification and Counting Challenge 2022[J]. arXiv preprint arXiv:2111.14485, 2021.
- [3] Deng J, Dong W, Socher R, et al. Imagenet: A large-scale hierarchical image database[C]//2009 IEEE conference on computer vision and pattern recognition. Ieee, 2009: 248-255.
- [4] Krizhevsky A, Hinton G. Learning multiple layers of features from tiny images[J]. 2009.
- [5] Verma R, Kumar N, Patil A, et al. MoNuSAC2020: A multi-organ nuclei segmentation and classification challenge[J]. IEEE Transactions on Medical Imaging, 2021, 40(12): 3413-3423.
- [6] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 3431-3440.
- [7] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation[C]//International Conference on Medical image computing and computer-assisted intervention. Springer, Cham, 2015: 234-241.
- [8] Zhou Z, Rahman Siddiquee M M, Tajbakhsh N, et al. Unet++: A nested u-net architecture for medical image segmentation[M]//Deep learning in medical image analysis and multimodal learning for clinical decision support. Springer, Cham, 2018: 3-11.
- [9] Oktay O, Schlemper J, Folgoc L L, et al. Attention u-net: Learning where to look for the pancreas[J]. arXiv preprint arXiv:1804.03999, 2018.
- [10] Valanarasu J M J, Sindagi V A, Hacihaliloglu I, et al. Kiu-net: Towards accurate segmentation of biomedical images using over-complete representations[C]//International conference on medical image computing and computer-assisted intervention. Springer, Cham, 2020: 363-373.
- [11] Milletari F, Navab N, Ahmadi S A. V-net: Fully convolutional neural networks for volumetric medical image segmentation[C]//2016 fourth international conference on 3D vision (3DV). IEEE, 2016: 565-571.
- [12] Salehi S S M, Erdogmus D, Gholipour A. Tversky loss function for image segmentation using 3D fully convolutional deep networks[C]//International workshop on machine learning in medical imaging. Springer, Cham, 2017: 379-387.
- [13] Lin T Y, Goyal P, Girshick R, et al. Focal loss for dense object detection[C]//Proceedings of the IEEE international conference on computer vision. 2017: 2980-2988.
- [14] Wu Z, Shen C, Hengel A. Bridging category-level and instance-level semantic image segmentation[J]. arXiv preprint arXiv:1605.06885, 2016.
- [15] Badrinarayanan V, Kendall A, Cipolla R. Segnet: A deep convolutional encoder-decoder architecture

for image segmentation[J]. IEEE transactions on pattern analysis and machine intelligence, 2017, 39(12): 2481-2495.

[16] Zhao H, Shi J, Qi X, et al. Pyramid scene parsing network[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 2881-2890.

[17] Chen L C, Papandreou G, Kokkinos I, et al. Semantic image segmentation with deep convolutional nets and fully connected crfs[J]. arXiv preprint arXiv:1412.7062, 2014.

[18] Chen L C, Papandreou G, Kokkinos I, et al. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs[J]. IEEE transactions on pattern analysis and machine intelligence, 2017, 40(4): 834-848.

[19] Chen L C, Papandreou G, Schroff F, et al. Rethinking atrous convolution for semantic image segmentation[J]. arXiv preprint arXiv:1706.05587, 2017.

[20] Chen L C, Zhu Y, Papandreou G, et al. Encoder-decoder with atrous separable convolution for semantic image segmentation[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 801-818.

[21] Ibtehaz N, Rahman M S. MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation[J]. Neural Networks, 2020, 121: 74-87.

[22] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.

[23] Strudel R, Garcia R, Laptev I, et al. Segformer: Transformer for semantic segmentation[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 7262-7272.

[24] Chen J, Lu Y, Yu Q, et al. Transunet: Transformers make strong encoders for medical image segmentation[J]. arXiv preprint arXiv:2102.04306, 2021.

[25] Huang H, Lin L, Tong R, et al. Unet 3+: A full-scale connected unet for medical image segmentation[C]//ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2020: 1055-1059.

[26] Liu X Y, Wu J, Zhou Z H. Exploratory undersampling for class-imbalance learning[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 2008, 39(2): 539-550.

[27] Chawla N V, Bowyer K W, Hall L O, et al. SMOTE: synthetic minority over-sampling technique[J]. Journal of artificial intelligence research, 2002, 16: 321-357.

[28] Han H, Wang W Y, Mao B H. Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning[C]//International conference on intelligent computing. Springer, Berlin, Heidelberg, 2005: 878-887.

[29] Jo T, Japkowicz N. Class imbalances versus small disjuncts[J]. ACM Sigkdd Explorations Newsletter, 2004, 6(1): 40-49.

[30] Kingma D P, Ba J. Adam: A method for stochastic optimization[J]. arXiv preprint arXiv:1412.6980, 2014.

谢 辞

感谢苗夺谦老师、赵才荣老师和王楚天学长对我毕业设计的帮助。同时也感谢同济大学以及各位老师对我的栽培。

装
订
线