提升
多语言历史文本的 OCR 性能

Hylke Westerdijk　Ben Blankenborg　Khondoker Ittehadul Islam
University of Groningen
{ h.p.westerdijk, b.b.g.blankenborg, k.i.islam } @student.rug.nl

Abstract

本文展示了我们在光学字符识别（OCR）和文档布局分析三个任务中的方法和发现，这些任务使用了先进的深度学习技术。首先，对于死海古卷的希伯来文历史碎片，我们通过广泛的数据增强扩展了我们的数据集，并采用 Kraken 和 TrOCR 模型来提高字符识别。在我们对 16 至 18 世纪会议决议任务的分析中，我们使用了一个卷积递归神经网络（CRNN），该网络结合了用于语义分割的 DeepLabV3+ 和双向 LSTM，并结合基于置信度的伪标签来优化我们的模型。最后，对于现代英文手写识别任务，我们应用了一个带 ResNet34 编码器的 CRNN，并使用连接时序分类（CTC）损失函数进行训练，以有效捕捉序列依赖性。这份报告提供了宝贵的见解，并提出了未来研究的潜在方向。

1　引言

识别手写文本中的字符任务面临重大挑战。尽管字母的基本形状保持一致，但每个人独特的书写风格会引入可变性。此外，书写表面的状况可能会随着时间的推移而恶化，缺乏上下文线索可能会导致解释上的模糊。分析大量手写文件并将其转换为数字格式的过程需要人工转录员付出相当多的时间和精力。

自动化这个转录过程具有明显的优势。因为字母识别的手动特征工程是一个复杂的工作，字母的形状受到许多因素的影响，包括作者的风格、使用的笔类型、表面材质和书写方向。这一复杂性自然引导我们考虑使用深度学习，深度学习采用层次模型，能够自主从数据中学习特征和任务。此外，深度学习模型通常展示出优于基于手工特征模型的性能 [LBH15] 。然而，深度学习的一个显著限制是它对数据量的高度依赖以有效地学习特征；数据不足可能导致过拟合，使模型仅在训练集上表现良好 [LBH15] 。这个问题在涉及历史手写稿的光学字符识别（OCR）任务中尤为突出。由于几个世纪前的手稿往往稀少而且书写体多样，获取足够的训练数据的挑战变得更加显著。

在这份报告中，我们通过应用深度学习方法解决与光学字符识别（OCR）相关的三个具体挑战。我们首先在本节中概述任务描述。在第 2 节中，我们回顾了在类似任务中被证明有效的现有方法。第 3 节提供了我们研究中使用的数据集的详细概述。在第 4 节中，我们描述了实施的方法论，第 5 节概述了整个研究过程中采用的实验设置。第 6 节展示了我们的研究结果，而第 7 节和第 8 节分别提供了我们的讨论和结论。

1.1　任务一

此任务的目标是对死海古卷数据集中的希伯来文片段进行光学字符识别（OCR），该数据集由公元前三世纪至公元一世纪的手写片段组成 [MT99] 。为了有效地完成此任务，我们启动了一个数据增强过程，以扩展我们的训练数据集。随后，我们使用专为历史手稿进行 OCR 设计的 Kraken [Kie19] 和 TrOCR [LLC+23] 模型来训练我们的模型。

在任务 2 中，我们将关注点从光学字符识别转向文档布局分析。文档布局分析的主要目标是通过识别各种文本类型，例如段落和标题，并对它们进行准确分类，从而揭示文档的基本结构。对于这一分析，我们使用了 Overijssel 省的数据集，该数据集包含荷兰语的会议决议，涵盖九种不同的文本类型。我们获得了该数据集的 90 个标记图像，这些图像作为我们研究的基础。

我们的方法采用了语义分割方法，其中图像中的每个像素都被分类，而不是在分类之前划定文本区域边界的对象检测方法。首先，我们对图像进行了预处理，以获得更好的分辨率质量。随后，我们利用有限数量的标记图像训练了 DeepLabV3+ 模型 [CZP+18] 。因此，我们应用训练好的模型对数据集

中剩余的图像进行伪标记。那些获得高置信度伪标签的图像随后被用于增强我们模型的训练。在测试集上评估我们的模型时，我们获得了 0.7030 的损失和 0.6506 的平均交并比，表明该分类任务的性能优良。

1.2 任务 3

任务 3 专注于在 IAM 数据集中对手写体进行精确识别，该数据集包含多行英文手写文本。对于这个任务，我们采用了集成 ResNet34 骨干网络的卷积递归神经网络（CRNN），这有助于从视觉数据中提取稳健的特征。在特征提取之后，我们利用多层 BiLSTM 架构来有效捕获手写中存在的序列依赖关系。该模型的训练使用连接时序分类（CTC）损失函数以优化性能。

2 文献综述

2.1 任务 1

OCR 领域提供了多种方法，本文这一部分特别强调基于深度学习的方法。我们首先研究了基于 transformer 的技术，例如基于 transformer 的光学字符识别（TrOCR）[LLC+23] 。TrOCR 作为手写识别的领先方法而脱颖而出，其性能优于现有的印刷和手写文本的最新技术方法。此模型利用了带有自注意力机制的 Transformer [VSP+17] 架构，特别是通过连接时序分类（CTC）[Gra12] 损失函数进行训练。重要的是，TrOCR 不依赖于会引入偏差的卷积神经网络（CNN）骨干网。该架构包括一个编码器，利用预训练的视觉 transformer 模型来生成图像块的表示，而解码器则采用预训练的大型语言模型来解码从这些图像表示中得到的文本内容。在数据有限的情况下，采用预训练模型至关重要，因为 transformer 架构通常需要大量数据——通常在十万数据点左右——才能充分学习而无先前偏见 [IDBN+21] 。在这种背景下，TrOCR 采用了 DeIT [TCJ22] 和 Beit [BDPW21] 作为预训练的视觉 transformer，RoBERTa [LOG+19a] 和 MiniLM [WWD+20] 作为其语言模型，这些模型都是在英文数据上训练的。这种对英文语言模型的依赖可能会对涉及不同语言和文字的识别任务构成挑战。

另一个显著的模型，使用视觉变换器的手写文本识别（HTR-VT）[LCTS25] ，通过结合卷积神经网络进行特征提取，而不是单纯依赖于标准的补丁嵌入，解决了视觉变换器数据密集的特性。该模型仅使用变换器架构的编码器组件。此外，它引入了跨度掩码技术，这是一种有效的正则化方法，通过掩盖特征图中连接的特征来实现。值得注意的是，HTR-VT 显示出具竞争力的结果，甚至在大约 20,000 个数据点的小型数据集上，其性能与基于卷积神经网络的模型相当或有时超过这些模型。

最后，我们研究了 Kraken [Kie19] ，它采用了一种独特的方法，通过结合卷积神经网络和双向 LSTM，使用 CTC 损失函数进行训练，同时完全省略了 transformer 架构。该模型专为历史文献中的字符识别而设计，因为这些文献通常比现代文本更为退化。值得注意的是，Kraken 在各种字体上表现出了令人印象深刻的性能，在包括西里尔字母到拉丁字母的各种字体上，平均字符准确率超过 95 %。特别是在我们应用于死海古卷数据集时，它达到了 96.9 的平均字符准确率。

2.2 任务 2

文档布局分析模型通常可以分为两种主要方法：语义分割方法和对象定位方法。在本小节中，我们将探讨这两类方法的代表性例子。

一个普遍的物体定位选择是 Mask R-CNN 模型。该方法基于 Faster R-CNN 框架 [RHGS16] ，通过添加一个额外的分支来为拟议的兴趣区域生成分割掩码。为了确保分割掩码中的像素位置与兴趣区域一致，Mask R-CNN 采用 RoIAlign 而不是传统的 RoIPool。这分割分支与 Faster R-CNN 的本征边界框回归和分类过程并行运行，有效地将分割与分类解耦。对 Faster R-CNN 的修改包括将卷积层产生的特征图输入到区域提案网络中，随后产生的区域提案会与特征图组合，生成包含叠加区域提案的整合特征图。应用于报纸版面进行文档分析时，Mask R-CNN 实现了平均精确度 81.6，同时伴随着 0.13 的掩码损失 [AA19] 。

与物体定位相反，语义分割是根据对应的类别对图像中的每个像素进行分类，而不仅仅是生成一个边界框。DeepLabV3+ [CZP+18] 是对先前 DeepLabV3 [CPSA17a] 模型的增强版本，引入了一个解码器，旨在优化分割结果，形成一个由编码器和解码器组成的端到端卷积神经网络（CNN）。这一架构在多个基准数据集上产生了最先进的结果。DeepLabV3+ 编码器常用的主干网络是 ResNet50 [HZRS16] ，这是一种由 50 层组成的强大 CNN 架构，并在 ImageNet 上经过预训练。

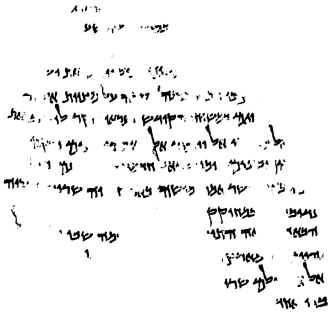

Figure 1: 来自数据集中包含从死海古卷中提取文本的二值化图像。

2.3 任务 3

针对包含超过 10,000 个数据点的现代英文手写数据集，光学字符识别（OCR）可以通过多种方式进行。TrOCR [LLC+23] 在 IAM 数据集上展示了出色的表现，利用了基于变压器的架构。另一种有前途的方法涉及针对段落或行的端到端手写识别而设计的垂直注意力模块 [CCP22] 。这个垂直注意力模块整合了混合注意力机制——即内容基础和位置基础——以应对从零开始训练变压器时少于 100,000 个数据点的挑战。通过在训练过程中专门应用数据增强，这种方法在每个周期通过对原始数据进行轻微修改（如透视变换或弹性扭曲）来生成“新”数据点。这个创新的模块带来了最先进的性能，在 RIMES [GCG+24] 数据集上实现了 1.91 % 的字符错误率。不过，值得注意的是，该模型需要较长的收敛期，通常大约在一千个周期左右。

超越标准文本识别，场景文本识别专注于解释户外环境中捕获的文本，在那里文本出现在各种背景和其他物体之间。IterVM [CW22] 通过将迭代视觉模型与迭代语言模型结合以获得最佳性能，展现了一个实现最先进成果的模型。然而，由于手写风格变化所增加的复杂性，这种模型在传统 OCR 任务中的效能仍然不确定。

除了使用转换器方法之外，利用循环神经网络（RNNs）和卷积神经网络（CNNs）的混合方法也可以有效。例如，双向 LSTM 相较于其单向对应物 [GS05] 具有捕捉更多序列信息的优势，使其在理解文本行中的依赖关系方面特别擅长。然而，双向 LSTM 无法直接处理图像数据，因此，它与卷积网络配对以提取相关特征，这些特征随后被输入到 RNN 中。在 [KDN+25] 中，其中一个卷积循环神经网络（CRNN）在文本识别任务中实现了 92 % 的整体准确率。值得注意的是，CRNNs 利用现有框架，相比于基于转换器的模型，实现了更快的收敛速度和更少的数据需求。

3 数据集

3.1 任务 1

对于任务 1，我们获得了一批来自死海古卷的希伯来文页面。需要注意的是，希伯来文的阅读顺序是从右到左，这与许多基于记忆的模型通常接受的从左到右的结构形成对比。卷轴可以追溯到公元前三世纪，由于材料的显著腐蚀，导致文本损坏，从而使字符识别任务变得复杂。我们总共获得了二十幅图像，每幅图像提供了三种不同的格式：RGB，灰度，和二值化。作为说明，图 1 展示了一个二值化图像的例子。此外，我们还收到了各种对应于希伯来字母表 22 个字母及其变体的 27 个字符的示例图像。

3.2 任务 2

对于任务 2，我们获得了一组由 90 张图像及其标签组成的数据集，这些标签来自一系列名为“斯塔滕范奥弗耶塞尔，骑士团和城市以及后续学院”的历史文本。这些文档包含了 1578 年至 1795 年间的政府会议决议。每张图像都有两个不同的标签：一个 JSON 文件，它详细描述了边界框坐标并识别文本区域；每张图像的一个语义掩码文件，它为不同类别分配不同的像素值。这些图像尺寸不一；有些描绘单页，而另一些展示双页。在语义掩码中，不同的文本类型被分类，每个类别都有唯一的标识符：背景标记为 0，标题赋值为 1，段落被分配 2，申请（摘要）取值为 3，决策标记为 4，旁注（笔记）标

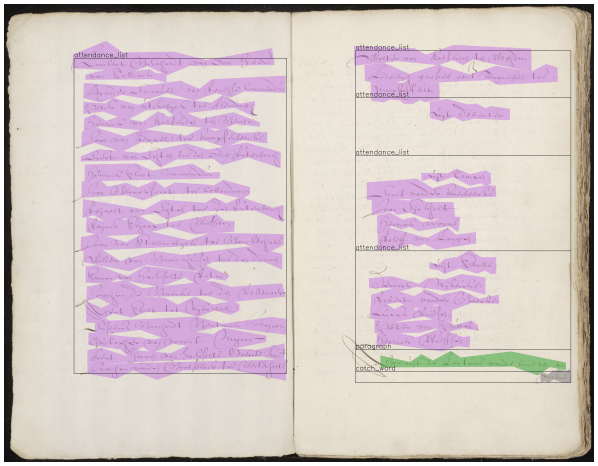

Figure 2: 一个包含分割掩膜和边界框注释的分辨率页面。

记为 5，出席名单分类为 6，关键词记为 7，日期表示为 8。此外，还存在一个标记为 11 的“未定义”类别，它仅出现一次且没有明确解释。如图 2 所示，展示了一个页面及其分割掩码和边界框覆盖的例子。值得注意的是，这些标签的分布是不平衡的，一些类别，如段落，发生的频率明显高于其他类别，比如日期。

3.3 任务 3

对于任务 3，我们获得了 IAM 数据集的一个子集，该数据集包括了 657 位不同作者的手写行。我们的特定子集包括从完整数据集中的 15,539 行中提取的 7,458 行。这些行在大小、长度和内容上有所不同，示例如图 3 所示。原始文本来自于兰开斯特-奥斯陆/卑尔根的英国英语语料库 [SLG78] 。此外，我们还提供了一个包含每一行的真实数据的.txt 文件，以确保我们训练和评估过程中的准确参考。

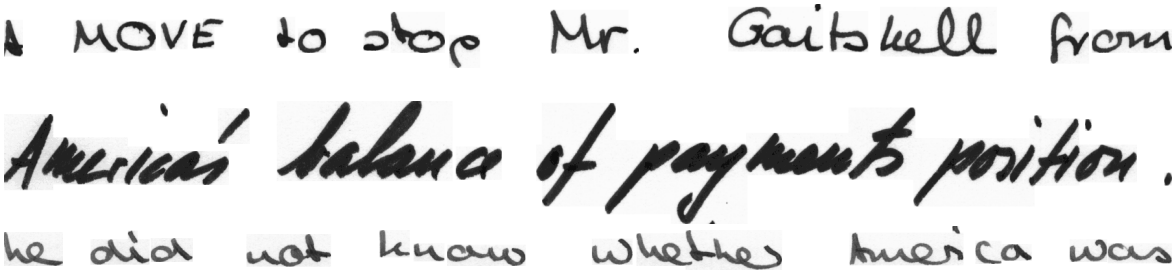

Figure 3: IAM 数据集中的行样本，展示了整体任务的独特挑战。

4 方法

4.1 任务 1

4.1.1 数据增强

为了增加用于训练的图文对的数量，我们研究了多个在线的机器打字来源。我们开始考虑使用 Habakkuk 字体的《旧约》文本块的在线版本。结果，我们确定了 (1) he.wiki [1] 和 (2) thewaytoyahuweh [2] 是唯一符合我们标准的来源。我们从每个文本块中过滤掉标点符号。最终，我们生成了 40 个包含图文训练对文本版本的文本文件。

为了生成图像合成，我们以 *number_of_lines* 乘以 *token_size* 的文本文件画布为基础。随后，我们从右到左插入相应的字符图像。在每个画布上，我们设置不同的帕林噪声 [3] 参数以复制测试集的背

[1] https://he.wikisource.org/wiki/
[2] https://downloads.thewaytoyahuweh.com/
[3] https://rtouti.github.io/graphics/perlin-noise-algorithm

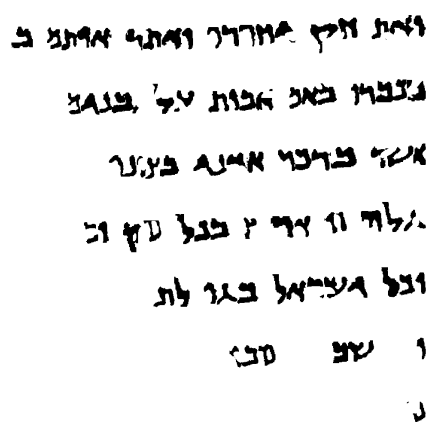

Figure 4: 进行图像组合后增强图像的一个示例，处理对象为收集的文本块。

景和图像纹理。在这个过程中，我们通过实验为布局参数、字母间距、空格宽度、波幅和波频赋值。随后，我们在这些图像上应用我们的线分割技术 [4] ，生成用于模型的增强训练样本。针对任务中提供的字符的每个不同版本，我们重复这些步骤。最终，我们生成了 9,438 个用于模型训练的线分割图像文本对。图中提供了一个图像文本对的示例，见图 4 。

4.1.2 模型

TrOCR 我们利用了 TrOCR $_{Large}$ [LLC+23] ，通过 Transformer 架构，这比现有的文本识别模型表现更好。该架构允许模型在编码器级别理解图像，并在解码器中以词级生成文本。更具体地说，编码器使用预训练的 ViT 模型 [DBK+20] 初始化，而解码器使用 RoBERTa [LOG+19b] 初始化。该模型在英语数据集上进行了训练，如 IIT-HWS [KJ16] 和 IAM 手写数据库。

我们还尝试使用 Kraken [5] ，因为它的架构与我们的流程相一致。具体来说，该模型在字符识别之前执行行分割。此外，它已被训练用于历史手写文本，并且能够处理图像中的从右到左的语言，包括希伯来语。

4.2 任务 2

4.2.1 数据处理

为了克服分辨率限制，我们决定使用漫水填充算法将双页文档拆分为单页文档。在每张彩色图像上，我们在坐标 (image_height - (image_height/3), (image_width/2)) 处启动算法，然后向左移动，直到检测到 RGB 颜色值中的 R 值低于 200。这表明进入了一个较暗的区域，通常为边框或边缘，我们将其标记为页面之间的分隔点。在单页面文档的情况下，我们的算法识别外边框并将其保留为单页。

我们将我们的方法应用于训练集中的所有彩色图像，将识别出的分割 x 坐标存储在以文件名为键的字典中。然后使用这些 x 坐标将相应的分割掩膜和二值图像中的页面分开，为所有相关文件添加后缀 _l 和 _r，以便进行正确的映射。

为了准确对齐坐标的边界框，我们遇到了三种情况：(1) 盒子的 x 轴两端都在左侧，(2) x 轴的两端都在右侧，或者 (3) 我们识别的 x 坐标在框之间。在情况 (3) 中，我们评估了盒子的一侧是否匹配情况 (1) 或 (2)，并将另一侧的 x 轴设置为我们识别的分页分隔 x 坐标。

图 5 展示了彩色图像中情况 (3) 的一个示例，我们分别将边缘和日期标签的 x 轴设置为我们识别的 x 坐标值，以确保没有遗漏框。然后，我们在未标记的彩色图像上重复此算法，以生成未标记集的预处理二值图像和分割掩码图像。最终，我们实现了所有图像的半像素维度缩减。

4.3 模型

在审阅任务组织者提供的语义掩码图像时，我们观察到每个掩码通过不同的颜色与各种标签区分开来。因此，我们决定通过颜色编码作为多类别预测任务的真实标签来进行语义分割和类别标签预测任务。

[4] 由于伦理限制，我们不能披露其实现
[5] https://kraken.re/main/index.html

我们的目标是选择一个预训练的语义分割模型，该模型能够有效地在带有硬标签的图像上收敛，具体来说就是区分前景（文本）和背景。因此，`segmentation\_models\_pytorch.DeepLabV3Plus` 模型 [6] [CPSA17b] 能够有效地满足我们的需求。最初，我们使用此模型进行语义分割，随后我们将其用于多类别预测。

4.4 自监督学习

为了克服有限的训练数据，我们通过自监督学习利用了 1,255 张未标记的图像 [7] 。对于每个单独的任务，我们使用整个未标记数据集中表现最佳的模型进行伪标签生成。然后，我们将模型表现出高平均置信度的图像整合到我们的训练集中，并从未标记的集合中移除它们。这个过程反复进行，直到在预定的若干个周期内不再观察到验证性能的提升。在语义分割任务中，这个过程进行了一次，而在多类别预测任务中，这个过程进行了两次。

4.5 任务 3

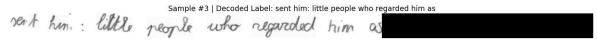

Figure 5: 我们的图像预处理示例，首先进行填充，然后将尾随的空白区域涂黑。

为了确保模型能够有效理解图像，我们首先调整图像的高度，同时保留纵横比。然后，我们填充每个图像以匹配批处理中的最大宽度，并仔细记录填充的起始索引，通过在填充区域上绘制黑线来消除尾随空白，如图 5 所示。

我们采用卷积递归神经网络（CRNN）[KDN+25] ，该网络由三个主要组件构成：卷积层、递归层 [Mem97] 和转录层。CRNN 底部的卷积层从每个输入图像中提取特征序列，而递归网络使用这些特征序列对每个帧进行预测。最后的转录层将每帧预测转换为标签序列。根据作者的建议，我们使用单一损失函数进行联合训练。

5 实验设置

5.1 任务 1

我们在 Nvidia A100 和 V100 GPU 上实施了我们的实验。我们选择了第一个 33 个文本文件（即，80 %）中合成的所有图像用于训练，其余用于验证集以避免泄漏。我们报告了两个测试样本（即 25-Fg001 和 124-Fg004 ）上的性能，这些样本由任务协调员提供给我们。我们评估模型的性能是使用两个测试样本上的 Levenshtein 距离比例的累加平均值。我们对 TrOCR 和 Kraken 模型进行端到端的所有层微调。对于 TrOCR 模型，我们设置了解码器模型的特殊标记，以便于它与编码器的同步学习。对于 Kraken 模型的训练，我们将 `batch\_size` 设置为 8 ，将 `min\_delta` 设置为 0.001 。此外，对于两个模型，我们设置了最大迭代次数为 10 的提前停止。

5.2 任务 2

我们在 Nvidia V100 GPUs 上进行实验。我们进行 80-20 的训练-测试拆分。然后，在 80 个% 训练集上，我们进一步进行 85-15 的训练-验证拆分。在所有情况下，我们进行随机拆分，并将 random_state 设为 42。在无监督和自监督学习环境中，我们保持相同的验证集和测试集。

由于我们利用用于分割掩膜的不同颜色作为标签的真实值，因此我们使用 `Image.Resampling.NEAREST` [8] 来防止类标签的插值。由于背景也由自己的颜色组成，我们在训练期间增加了另一个类（总共 9 类）。通过这种方式，我们也将所有未定义的标签设置为背景标签。在损失传播和评估期间，我们没有将背景的预测视为一个类标签。在自监督学习期间，我们通过将平均值和标准差设置为 [9] 建议的 `DeepLabV3Plus`的值来规范化未标记的图像，以改进迁移学习。由于一些未标记的文档是空的，我们

[6] https://smp.readthedocs.io/en/v0.1.3/_modules/segmentation_models_pytorch/deeplabv3/model.html

[7] 在本文的其余部分中，当我们利用未标记的数据集进行训练时，我们将其称为自监督学习，而当我们仅利用任务组织者提供给我们的 90 张图像进行训练时，我们将其称为无监督学习。

[8] https://pillow.readthedocs.io/en/stable/reference/Image.html

[9] https://docs.pytorch.org/vision/stable/models.html

仅对非空页面计算和传播损失。我们在测试集上报告我们的性能，并分别使用文本区域的 IOU [10] 的均值来评估语义分割和多类预测任务的 8 个类。

我们采用 ResNet50 作为我们的编码器，并用 ImageNet 初始化它的权重。输入通道数设置为 3。为了缓解多类别预测任务中过度代表类的过拟合，我们使用带权重的类计算，其中 ϵ 被设置为 $1e^{-9}$ 以防止除以零。我们使用的公式是：

$$\frac{total\_pixels}{8 \times (foreground\_class\_pixel\_counts + \epsilon)}$$

其中 *total_pixels* 代表所有类别像素的总和，排除 *background_class_pixel_counts* 。在整个实验过程中，我们将所有随机种子设为 42。对于有监督和无监督训练，批量大小固定为 2。我们分别使用交叉熵损失（CrossEntropyLoss）和 Adam 优化器来进行损失计算和优化。由于时间复杂度的考虑，我们仅实验了学习率为 0.0001。在所有情况下，如果模型在验证集上的平均 IOU 在连续五个周期内没有提高，我们就停止训练。在自监督训练阶段，如果模型的平均置信度达到 0.70 或以上，我们就将一个未标记的图像纳入训练集。

为了最大化特征提取能力，我们在扩展馈送到 RNN 的特征图宽度之前，用空权重初始化 `ResNet34`架构。由于 IAM 图像是灰度的，我们修改第一个卷积层以接受灰度图像而非 RGB。利用 `adaptive\_avg\_pool2d` [11] ，我们从 512 维输出的 `ResNet34`提取特征。然后将该特征向量输入到双向长短期记忆网络（Bi-LSTM），其丢弃率为 0.20 ，随后是 `log\_softmax`输出。对于训练，我们使用带有均值归约的 `CTCLoss`和 `AdamW`优化器，并将学习率设置为 0.001 。

我们在一个测试集上评估我们的模型，该测试集由数据集中随机选择的 100 个样本组成，同时将剩余数据按 80-20 分割以生成训练和验证集。我们在 5 个 epoch 后实施提前停止，并使用字符级别的 [12] 和单词级别的 [13] 错误率评估性能。最终，我们的模型共训练了 24 个 epoch。

6 结果

6.1 任务 1

我们在表格 1 中展示了在我们的数据集上微调的两个模型的结果。值得注意的是，Kraken 的表现比 TrOCR 模型好，比例为 0.178 。我们进一步展示了 Kraken 在单个测试项上的表现，并在表格 2 中报告它们。请注意，微调模型发现第二个测试项 124-Fg004 比第一个测试项 25-Fg001 更具挑战性。此外，我们比较了在我们增强数据集上预训练和微调的 TrOCR 和 Kraken 的结果。表格 3 显示，微调后 TrOCR 模型的 Laventish 距离比率比 Kraken 显著更高。具体来说，微调后的 TrOCR 模型的改进率大约是微调后 Kraken 模型的五倍。

Model	Laventish Distance Ratio
TrOCR	0.339
Kraken	0.447

Table 1: 在两个测试集上评估的每个模型的 Laventish 距离比率，使用增强数据进行了微调。

Test Sample Name	Laventish Distance Ratio
25-Fg001	0.467
124-Fg004	0.426

Table 2: 微调后的 Kraken 模型的 Laventish 距离比在每个测试项目中的评估。

6.2 任务 2

我们在表格 4 中展示了 `DeepLabV3Plus`模型在无监督和自监督设置下的语义分割和多类别预测任务的表现结果。在这两种任务中，该模型通过自监督学习方法实现了性能提升。更具体地说，我们注意到

[10] https://smp.readthedocs.io/en/latest/metrics.html
[11] https://docs.pytorch.org/docs/stable/generated/torch.nn.functional.adaptive_avg_pool2d.html
[12] https://lightning.ai/docs/torchmetrics/stable/text/char_error_rate.html
[13] https://lightning.ai/docs/torchmetrics/stable/text/word_error_rate.html

Model	Training Method	Laventish Ratio
TrOCR	Pretrained	0.171
	Finetuned	0.339 (+0.168)
Kraken	Pretrained	0.418
	Finetuned	0.447 (+0.029)

Table 3: 对预训练和微调后的 TrOCR 模型与 Kraken 模型的性能对比。当基于我们扩展的训练数据进行微调时，TrOCR 模型相比其预训练版本表现出比 Kraken 模型更大的性能提升。

Task	Learning Method	MIOU
Semantic Segmentation	Un-Supervised Learning	0.8437
	Self-Supervised Learning	0.8439
Multi-class Prediction	Un-Supervised Learning	0.5892
	Self-Supervised Learning	0.6506

Table 4: DeepLabV3Plus 模型在语义分割任务和多类别预测任务上的表现分别通过自监督学习和无监督学习方法来进行评估。

在语义分割任务中性能提升了一个非常小的 0.002 ，而在多类别预测任务中则显著提升了 ≈ 0.7。

6.3 任务 3

Evaluation Metric	Performance
Loss	0.926
CER	0.199
WER	0.528

Table 5: CRNN 在我们的测试集上的损失、字符错误率（CER）和词错误率（WER）。

我们在表 5 中报告了表现最好的模型在测试集上的损失、字符错误率 (CER) 和单词错误率 (WER)。结果表明，预测的单词中近 50 % 是错误的，同时预测的字符中有五分之一是错误的。

7 讨论

在这个任务中，我们意识到由于时间限制，我们的编码器-解码器架构的潜力没有完全发挥出来。如果能够评估它相对于我们提交的那个相对简单的模型的性能，将会很有趣。此外，我们没有进行充分的超参数调优或交叉验证，这可能会提高模型的性能。更多的时间可能会有助于改进我们的合成死海古卷数据集，该数据集虽然具有功能性，但仍然缺乏实现稳健泛化所需的真实性。

7.1 任务 2

类似于任务 1，我们在超参数调优和交叉验证方面的努力有限。我们在数据增强方面的做法很少，仅限于将页面分成两半，尽管采用更全面的增强技术可以显著提高模型的稳健性。一个显著的错失机会是未能实现原本计划在二分类分割阶段之后的分类头。由于研究不完整和时间限制，我们转而采用多类别分割方法，这使我们无法探索原始模块化设置的潜在优势。此外，我们对预训练模型的实验有限，这也归因于时间限制。考虑到我们最初训练了一个二分类分割模型，随后对其进行微调以实现多类别分割，未来的工作中进行与从零开始开发的多类别模型的比较分析将是有价值的。

对于这个任务，如果能够完成正在开发的基于 transformer 模型的训练，通过与 CRNN 进行直接比较，将提供有价值的见解，特别是关于模型复杂性增加在有限数据集上的表现。我们还忽略了在合成手写数据集上进行预训练的机会，这一策略我曾经研究过，但由于时间限制未能实施。同样，CRNN 流水线缺乏数据增强，这本可以进一步改善泛化能力。与之前的任务一样，我们没有进行超参数调优或交叉验证，这本可以对模型性能提供重要的见解。

为了解决观测到的 50 % 的词预测错误率，在未来的实现中可以考虑几种策略：（1）引入一个后纠正模块以提高输出准确性，（2）将预训练嵌入整合到 LSTM 的嵌入层中，或（3）探索用在子词级别训练的模型替换 LSTM 架构。例如，一个有效的增强措施可以涉及通过一个语言模型来修正预测输出，从而生成语法上准确的结果。此外，利用像 word2vec [**?**] 这样的预训练嵌入到 LSTM 的嵌入层中，可能在很大程度上提高模型在这一任务上的性能。

8 结论

8.1 任务一

任务 1 揭示了历史手稿图像合成的复杂性，例如死海古卷。要实现其独特质地和外观的真实表现，需要仔细的数据准备和图像处理，而在这些方面，我们的合成图像虽然实用，但未能完全成功。尽管如此，我们的模型表现出色，可能是因为在中世纪手写希伯来文本的数据集上进行预训练，这突显了在此背景下迁移学习的有效性。这次经历强调了处理有限和异质数据的困难，尤其是在生成能够准确捕捉真实世界变化的合成样本时。值得注意的是，非 Transformer 模型也表现良好，表明在数据受限的情况下，简单的架构也可以表现得出乎意料地有效。总体而言，这个任务强调了继续优化我们的方法和数据集，以提高未来研究工作中合成图像生成的真实性和性能的必要性。

在任务 2 中，我们的目标是开发一个两阶段的流程，将二元分割模型与后续分类头集成，以有效地标记不同区域。这种模块化设计旨在增强模型架构内的功能明确性。然而，这种结构的实施并没有达到预期的性能，迫使我们回到多类别分割方法，最终提供了更优的结果。

为了解决标记数据有限的挑战，我们采用伪标记技术来增强我们的训练集，从而显著提高了模型的性能。这一经历强调了一个关键的洞察：理论框架虽然有价值，但在实际应用中可能并不总是能够无缝转换，特别是当模型复杂性增加时，更容易引发错误。今后，在优化性能结果的过程中，平衡创新设计和经验验证将是至关重要的。

8.2 任务 3

任务 3 强调了模型复杂性与数据集规模之间的重要平衡。我们尝试在 CNN 和 RNN 之间整合一个 transformer 编码器，但最终阻碍了性能，这可能是由于我们的数据集规模不足。相比之下，我们还探索了基于 transformer 的模型架构，该架构在大量数据增强的支持下展示了有希望的训练指标，但未能在截止日期前完成。结果强调了增强训练数据多样性的重要性，这可能与增加模型复杂性同样有益。此经验强化了这样的观点：即便在数据集受限的情况下，有针对性的数据增强策略也可以大幅提高模型性能。

References

[AA19] Abdullah Almutairi and Meshal Almashan. Instance segmentation of newspaper elements using mask r-cnn. In *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, pages 1371–1375, 2019.

[BDPW21] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021.

[CCP22] Denis Coquenet, Clément Chatelain, and Thierry Paquet. End-to-end handwritten paragraph text recognition using a vertical attention network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1):508–524, 2022.

[CPSA17a] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[CPSA17b] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[CW22] Xiaojie Chu and Yongtao Wang. Itervm: iterative vision modeling module for scene text recognition. In *2022 26th International Conference on Pattern Recognition (ICPR)*, pages 1393–1399. IEEE, 2022.

[CZP+18] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[DBK+20] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[GCG+24] Emmanuèle Grosicki, Matthieu Carré, Edouard Geoffrois, Emmanuel Augustin, Françoise Preteux, and Ronaldo Messina. RIMES, complete, 2024.

[Gra12] Alex Graves. Connectionist temporal classification. In *Supervised sequence labelling with recurrent neural networks*, pages 61–93. Springer, 2012.

[GS05] Alex Graves and Jürgen Schmidhuber. Framewise phoneme classification with bidirectional lstm networks. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, volume 4, pages 2047–2052. IEEE, 2005.

[HZRS16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[IDBN+21] Andrei Ivanov, Nikoli Dryden, Tal Ben-Nun, Shigang Li, and Torsten Hoefler. Data movement is all you need: A case study on optimizing transformers. *Proceedings of Machine Learning and Systems*, 3:711–732, 2021.

[KDN+25] Mahesh Kumar, Ajay Dureja, Rachna Narula, Ravi Arora, et al. Ocr-crnn (wbs): an optical character recognition system based on convolutional recurrent neural network embedded with word beam search decoder for extraction of text. *International Journal of Information Technology*, pages 1–8, 2025.

[Kie19] Benjamin Kiessling. Kraken-a universal text recognizer for the humanities. In *Digital Humanities 2019*, 2019.

[KJ16] Praveen Krishnan and CV Jawahar. Generating synthetic data for text recognition. *arXiv preprint arXiv:1608.04224*, 2016.

[LBH15] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[LCTS25] Yuting Li, Dexiong Chen, Tinglong Tang, and Xi Shen. Htr-vt: Handwritten text recognition with vision transformer. *Pattern Recognition*, 158:110967, 2025.

[LLC+23] Minghao Li, Tengchao Lv, Jingye Chen, Lei Cui, Yijuan Lu, Dinei Florencio, Cha Zhang, Zhoujun Li, and Furu Wei. Trocr: Transformer-based optical character recognition with pre-trained models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 13094–13102, 2023.

[LOG+19a] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[LOG+19b] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[Mem97] Long Short-Term Memory. Sepp hochreiter and jürgen schmidhuber. *Neural Computation*, 9(8):1735, 1997.

[MT99] Florentino García Martínez and Eibert JC Tigchelaar. *The Dead Sea Scrolls*. Wm. B. Eerdmans Publishing Company, 1999.

[RHGS16] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149, 2016.

[SLG78] Johansson Stig, Geoffrey N Leech, and Helen Goodluck. Manual of information to accompany the lancaster-oslo: Bergen corpus of british english, for use with digital computers. *(No Title)*, 1978.

[TCJ22] Hugo Touvron, Matthieu Cord, and Hervé Jégou. Deit iii: Revenge of the vit. In *European conference on computer vision*, pages 516–533. Springer, 2022.

[VSP+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[WWD+20] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 5776–5788. Curran Associates, Inc., 2020.