

毕设课题：基于强化学习的图像到标记识别

LaTeX 数学公式 OCR

方盛俊

课题背景及意义

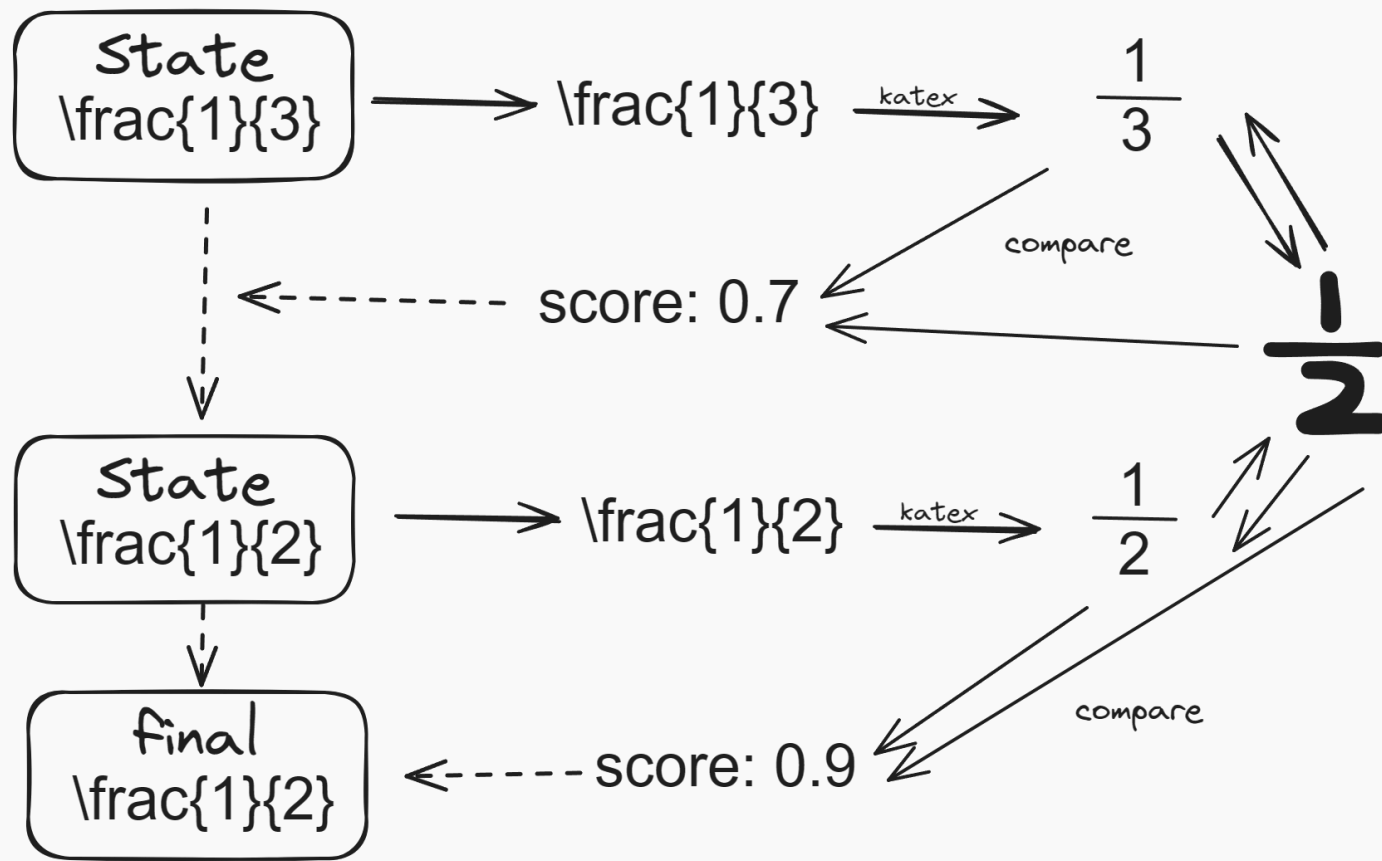
LaTeX 数学公式识别是一种在文档自动化¹、笔记软件等领域都有迫切需求的 OCR 任务，有广泛的应用场景。目前的 LaTeX 公式识别基本都是基于图像识别的有监督学习²，它们只使用了已有的训练集。这种方法忽略了 LaTeX 公式识别任务本身是一种状态完全可观测的、有模型的、可以写出完美的真实环境模拟器的特殊任务的事实。我们能够使用 KaTeX 或 Typst 等工具实时渲染当前状态对应的数学公式图像，引入额外的先验信息，进而进行实时的强化学习训练和决策。

The diagram shows the LaTeX formula $\text{Tr}(\mathbf{M} + \mathbf{M}^-) = 0$ overlaid on a grid. Above the formula, tokens are listed with arrows pointing to their corresponding parts: `\mathr{` points to the opening brace, `T` points to 'Tr', `r` points to the closing brace, `{` points to the opening brace of the argument, `\bf` points to the boldface command, `M` points to the first \mathbf{M} , `+` points to the plus sign, `\bf` points to the second boldface command, `M` points to the second \mathbf{M} , `-` points to the minus sign, `{` points to the opening brace of the argument, `}` points to the closing brace, `=` points to the equals sign, `0` points to the zero, and `}` points to the final closing brace. The grid has a blue shaded area on the left and a purple shaded area in the middle, corresponding to the boldface commands and the first \mathbf{M} .

¹<https://mathpix.com/>

²<https://github.com/lukas-blecher/LaTeX-OCR>

主要技术路线



核心在于借助借助 KaTeX 或 Typst 等工具决策时**实时渲染数学公式**，引入额外的先验信息。

首先需要的网络和 MDP：

1. **图像编码器**：一个 encoder，能够将图像加上 Positional Encoding 并编码为一维特征向量。
2. **策略网络**：一个 decoder，能够根据当前的 LaTeX 序列、当前图像与目标图像，生成一个新的 LaTeX 序列。
3. **State**：「目标图像一维特征向量」+「当前图像一维特征向量」+「当前序列」。
4. **Action**：策略网络 decoder 生成一个新的 LaTeX 序列，同时会调用 Typst 生成新图像。
5. **Reward**：训练时使用当前序列和目标序列的 BLEU score 作为奖励，编译失败则为 0 分。

1. **终止：**终止条件可以是 **最大步数、最大可用时间、评分网络的评分阈值** 或 n 次内评分不再增加。终止后，使用最后状态对应的 BLEU 评分值乘上较大系数作为最终 Reward，并使用策略网络梯度下降更新网络参数。
2. **优点：**由于强化学习是用状态转换进行渐进式地对当前状态进行增强，所以可能还可以支持**渐进式 OCR**，理论上只需要更改当前状态的「目标图像特征向量」即可，这点对手写笔记软件可能有很大帮助。

$$\mathcal{X} \xRightarrow{\text{ocr}} X \xRightarrow{\text{write}} X + y \xRightarrow{\text{ocr}} X + Y \xRightarrow{\text{write}} \frac{X + Y}{Z} \xRightarrow{\text{ocr}} \frac{X + Y}{Z}$$

为什么使用 Typst

1. Typst 是一个类似于 TeX 或 LaTeX 的有着完整功能的排版语言。
2. (TODO) Typst 支持数学公式，虽然和 LaTeX 公式语法不一样，但是可以通过 Rust 和 WASM 实现一个 Converter³ 并嵌入 Typst 里，我正在开发中。
3. Typst 是增量渲染，而且公式渲染速度很快，即使算上 I/O 也能保证在几十毫秒内完成。
4. Typst 可以输出 PDF、SVG 和 PNG 格式，尤其是 PNG 格式可以作为位图输入到神经网络。
 - a. pdfTex 或 MathJax 都只能输出 PDF 或 SVG 格式，矢量图无法直接输入到神经网络。
 - b. 否则就要加一层转换，不仅需要打包像浏览器这样的渲染器，而且速度会慢。
5. Typst 是 Apache 2.0 许可证，而且体积只有 30 MB 左右，很方便打包出来。
6. (TODO) Typst 可以对接 Python，从而避免 I/O 文件读写，能进一步加快速度，也许能降低到几毫秒内完成，这样就远快于一次 seq2seq 推理。
7. (TODO) Typst 是图灵完备的，可以借助 Typst 整点花活，例如换字体、换颜色、甚至是生成完全或部分伪手写的公式，进而增强数据集。

³<https://github.com/OrangeX4/mitex/>

1. 预处理：基于数据集生成一些不完整的 LaTeX 序列数据，进而模拟「运行到一半」的效果。
2. 预训练：
 - a. Token-level: 对编码器-策略网络进行 token-by-token 的训练，每生成一个 token，基于极大似然估计或交叉熵等可微目标函数进行普通的梯度下降训练。
 - b. Sequence-level: 对编码器-策略网络进行 sequence-level 的训练，在生成完整预测序列后，基于 BLEU 等离散的目标函数进行评分，然后借助 **策略梯度定理** 与 **REINFORCE** 获取随机梯度进行梯度下降。
 - a. 这里用到策略梯度定理是为了用随机梯度解决 BLEU score 不可微分、不可求梯度问题。
3. 强化学习训练：
 - a. 在预训练了网络之后，执行 **借助 Typst** 的策略梯度 REINFORCE 的强化学习。
 - b. 每一步都调用 BLEU 进行评分，最终状态的 BLEU 评分较大。

与强化学习的相关性

1. 目前公式识别领域基本都是基于 Image-to-Markup Generation with Coarse-to-Fine Attention 的架构和这篇论文提出的 IM2LATEX-100K 数据集在做，这篇作为 baseline 的论文也用了策略梯度定理，基本是用来处理离散不可微分的目标评分函数的，比如 BLEU，以及这之后的论文也都用了。
 - a. 也即我这里的训练流程中的预训练。
2. 强化学习训练步骤，理论上会有许多好处。
 - a. 从「渐进序贯式决策」很自然地就能联想到「强化学习」。
 - b. 理论上我们的模型可以不用太强，因为可以借助试错改进，类似于 Boosting，进而用小模型加快推理速度，**也能根据不同的复杂度的图像动态地增减推理时间。**
 - c. 采用强化学习而不是单纯的 Boosting 类的方法，就在于强化学习也许可以允许 **试错和绕路**，只要最终状态评分更高即可。
 - d. 保证训练过程和决策过程的一致性，在决策时也能充分利用实时渲染公式提供的额外信息。
3. 具体的效果还是要借助实验评估，但是理论上这种方法引入了外部信息，应该能达到 SOTA。

1. 基于 Attention 的数学公式识别，提出了基于图像识别技术的 LaTeX 数学公式 OCR 方法，作为数学公式识别领域的 baseline 使用 Image-to-Markup Generation with Coarse-to-Fine Attention⁴
2. 使用策略梯度定理强化学习的公式识别： Translating math formula images to LaTeX sequences using deep neural networks with sequence-level training⁵
3. 基于强化学习的自动生成 SQL 语句，其没有简单地模仿人类专家，而是在真实数据库实际执行 SQL 语句进行验证 Seq2SQL: Generating Structured Queries from Natural Language using Reinforcement Learning⁶

⁴<https://arxiv.org/abs/1609.04938v2>

⁵<https://arxiv.org/abs/1908.11415>

⁶<https://arxiv.org/abs/1709.00103>

需要解决的问题

1. 复现之前论文 baseline 的效果，尤其是长序列情形下的效果，作为之后的比对。
2. 搜搜与图像分块识别相关的论文，看看能不能有什么启发。
3. 训练时是否不使用 REINFORCE 而是用 actor-critic，以减小方差。
4. 构思一个不使用强化学习训练步骤，只使用先验知识的方法，作为 baseline 与后续强化学习方法进行比较，评估强化学习在探索方面的效果。例如可以考虑 GAN、Boosting 等。
5. 处理字号、字体和手写体，考虑如何增强对字号大小变化的稳健性；字体和手写体需要想办法生成更多的训练数据，尤其是生成手写数学公式数据的方法。
6. 看看是否能迁移到其他任务，可能迁移到的任务有流程图生成、Logo 生成、SVG 矢量图像生成、前端 UI 代码生成、PDF 文件识别等，这些都是比较典型的识别图像到标记任务。