The Exploration of CNN and ViT in Chinese Mnist Dataset

**Abstract**

1. **Introduction**

ML task:我们想要探究CNN和ViT在汉字识别任务上，各自存在什么优势。

Deep learning has revolutionized image classification, with CNNs historically being the dominant architecture due to their ability to extract hierarchical features efficiently. However, the emergence of Transformers, particularly ViT, has introduced a new paradigm that leverages self-attention mechanisms to model long-range dependencies. This paper examines CNN and ViT performance on a Chinese handwritten digit dataset, which presents a greater structural complexity compared to standard numerical digit datasets. We analyze the effects of image resolution on model performance and resource consumption.

1. **Problem Setting**

数据集为中文手写数字数据，选择其的缘由是相比简单的数字识别，中文数字的特征结构似乎更为复杂。并且中文数字包含简单到较复杂的汉字，像是一，二，三与零，四，五相比就简单很多。

The dataset used in this study consists of Chinese handwritten digits, which possess more complex stroke structures compared to standard Arabic numeral datasets such as MNIST. The choice of this dataset allows us to better evaluate how CNNs and ViTs capture intricate details in images. By experimenting with different image resolutions (64×64 and 128×128), we aim to investigate how each model adapts to changes in input size.

The Chinese MNIST dataset consists of handwritten digits from 0 to 9, each with distinct stroke-based structures. Unlike standard digit recognition, which mainly involves simple shapes, Chinese digits often require models to capture fine-grained patterns and long-range dependencies. The objective of this study is to evaluate how CNN and ViT architectures perform in extracting these patterns while adjusting for different image resolutions (64×64, 128×128, and 224×224).

1. **Methodology**

**CNNs** process images through convolutional layers that detect spatial features, pooling layers that reduce dimensionality, and fully connected layers that perform classification. A standard CNN architecture consists of:

1. **Convolutional layers**: Extract local patterns such as edges and textures.
2. **Pooling layers**: Downsample feature maps to reduce computation.
3. **Fully connected layers**: Convert spatial features into classification outputs.

Unlike CNNs, **ViTs** divide input images into fixed-size patches and treat them as sequences, similar to words in NLP. The main components of ViT include:

1. **Patch Embedding**: Converts image patches into tokenized representations.
2. **Multi-Head Self-Attention (MSA)**: Captures long-range dependencies across the entire image.
3. **Feedforward Layers**: Process extracted global information for classification.

**4. Experiment**

**64x64, 128x128, 224x224**的图像大小

对比训练时间，模型性能(准确率，损失值)，以及图像大小对这些指标的影响程度

* 1. **Dataset and Preprocessing**

Dataset: Chinese MNIST dataset

(展示各类图像一张，给出训练、测试、验证分布表格)

* 1. **Model Training Details**

**CNN Design**: 自定义一个简单的CNN网络，主要由两层卷积，两层池化，两层全连接组成 **(绘制结构图像)**

**ViT** **Design**: 使用transformers库中调用的ViTForImageClassification，我们打算观察最原始的训练过程，并未使用预训练参数。**(给出使用参数 ViTConfig)**

**Optimizer:** Adam with an initial learning rate of 0.0001

**Loss Function:** Cross-entropy loss

**Evaluation Metrics:** Accuracy, loss, training time

**5. Result**

**5.1 Model Performance**

**折线图：展示 两个模型分别绘制，在最好情况下的 Epoch-Acc/ Loss 图像**

**表格图：展示，在各个图像大小下的表现**

**5.2 Computational Efficiency**

**三线图**：展示**~~参数量(待定)，~~训练时间，内存占用，以及增长率**

**5.3 Key Observations(预期)**

1. **CNNs perform better on lower resolutions: CNNs excel in small-scale images due to efficient local feature extraction but struggle as image resolution increases.**
2. **ViTs benefit from larger images: ViTs leverage attention mechanisms, allowing them to extract more meaningful features when image size increases.**
3. **ViTs are more memory-efficient at high resolutions: Unlike CNNs, which require large fully connected layers, ViTs maintain a stable parameter size regardless of image resolution.**

分析结果：

解释：

**Conclusion**

总的来说，本研究简单探讨了CNN和ViT两种底层技术相差较大的机器/ 深度学习技术。我们观察到…

**~~不足：~~**

~~虽然本研究能清晰得出这些结论，但是中文手写数字识别任务并不能完全得出CNN和ViT的优劣。在实验中，我们也发现由于数据集的分割问题，测试集在训练过程中几乎全部优于训练集.~~