# 基于 LSTM 与 Transformer 的中文武侠小说生成方法比较研究

宋雯婧
13501982761@163.com

## 摘要

本文针对中文武侠小说文本生成任务，以金庸短篇小说为研究对象，对比分析了长短期记忆网络(LSTM)与 Transformer 两种深度学习模型在中文小说生成任务上的表现。研究构建了完整的文本预处理流程，分别实现了基于 LSTM 和 Transformer(GPT-2)的文本生成模型，并通过定量指标和人工评估对生成结果进行了系统分析。实验结果表明，在小规模数据集上，LSTM 模型训练效率更高，生成文本的局部连贯性较好；Transformer 模型在捕捉长距离依赖关系和生成全局连贯的故事情节方面表现更优，但对计算资源需求较高。本研究为中文创意写作的自动生成提供了实用的技术参考，并对不同场景下的模型选择提出了建议。

**关键词**：文本生成；LSTM；Transformer；武侠小说；深度学习

## 1. 引言

### 1.1 研究背景

随着自然语言处理技术的发展，自动文本生成已成为人工智能领域的研究热点之一。在文学创作领域，自动生成技术可以辅助作家创作，提供灵感来源，甚至能够生成完整的短篇故事。中文武侠小说作为具有鲜明特色的文学体裁，其语言风格和叙事结构具有一定的规律性，适合作为文本生成的研究对象。

### 1.2 研究意义

本研究选取金庸的短篇武侠小说作为实验数据，具有以下意义：(1)该作品篇幅适中，适合作为小规模文本生成研究的案例；(2)作品具有典型的武侠小说特征，包括人物对话、动作描写和情节推进等多样化的文本元素；(3)比较 LSTM 和 Transformer 两种主流生成模型在中文小说生成任务上的表现[1]，为相关研究提供参考。

### 1.3 研究内容

本文主要研究内容包括：(1)构建中文武侠小说文本预处理流程；(2)实现基于 LSTM 的序列生成模型；(3)实现基于 Transformer 的文本生成模型；(4)设计评估方法对生成结果进行分析[2]；(5)比较两种模型的优缺点及适用场景。

## 2. 实验方法

### 2.1 数据与预处理

本研究采用的数据预处理流程包括：

（1）文本清洗：去除原文中的非中文字符、网站信息等噪声

（2）分词处理：采用字符级分词，适应中文特点

（3）序列生成：构建训练用的输入-输出序列对

（4）向量化表示：使用 Tokenizer 将字符转换为数字索引

**2.2LSTM 模型架构**

实现的 LSTM 生成模型包含以下层次：

（1）嵌入层(Embedding)：将离散的字符索引映射为密集向量

（2）LSTM 层：两层 LSTM 结构捕捉序列依赖关系[3]

（3）全连接层(Dense)：输出每个字符的概率分布

**2.3 Transformer 模型实现**

基于 Hugging Face 的 Transformers 库，使用预训练的中文 GPT-2 模型进行微调：

（1）模型加载

```
tokenizer = GPT2Tokenizer.from_pretrained('uer/gpt2-chinese-cluecorpussmall')
model = GPT2LMHeadModel.from_pretrained('uer/gpt2-chinese-cluecorpussmall')（2）
```

（2）数据格式化

```
class YueNuJianDataset(Dataset):
    def __init__(self, txt_list, tokenizer, max_length):
        self.input_ids = []
        self.attn_masks = []
        for txt in txt_list:
            encodings_dict = tokenizer(txt, truncation=True,
                                       max_length=max_length,
                                       padding="max_length")
            self.input_ids.append(torch.tensor(encodings_dict['input_ids']))
            self.attn_masks.append(torch.tensor(encodings_dict['attention_mask']))
```

（3）训练配置

```
training_args = TrainingArguments(
    output_dir='./results',
    num_train_epochs=10,
    per_device_train_batch_size=2,
    learning_rate=5e-5,
)
```

**2.4 评估方法**

采用以下方法评估生成结果：

（1）困惑度(Perplexity)：衡量模型预测能力

（2）n-gram 多样性：计算生成文本的词汇丰富度

（3）人工评估：从连贯性、创意性和风格一致性三个维度评分

## 3.实验结果与分析

**3.1 训练效率比较**

| 指标 | LSTM 模型 | Transformer 模型 |
|---|---|---|
| 训练时间 | 45 分钟 | 4 小时 |
| 内存占用 | 2GB | 6GB |
| 收敛速度 | 快 | 慢 |

**3.2 生成质量评估**

（1）LSTM 生成示例

"范蠡望着远方的山峦，心中思念着西施。忽然间，一个绿衣少女出现在竹林间，手持竹棒，目光如电。那少女轻声道：'你的剑法可敌得过我的竹棒？'范蠡大惊，只见竹棒一点，已到眼前....."

（2）Transformer 生成示例

"月色如水，阿青站在馆娃宫的屋檐上，竹棒在手中轻轻颤动。'范蠡，你终究还是选择了她。'她的声音带着一丝颤抖。西施从殿内走出，两人目光相接的刹那，阿青看到了那传说中的绝世容颜。竹棒无力垂下，她转身消失在夜色中....."

### 3.3 定量评估结果

| 评估指标 | LSTM 模型 | Transformer 模型 |
|---|---|---|
| 困惑都 | 32.5 | 28.7 |
| 独特 3-gram 比例 | 0.42 | 0.58 |
| 平均句子长度 | 18.3 字 | 24.7 字 |

## 4.讨论

### 4.1 模型性能分析

实验结果表明，在小规模中文武侠小说数据集上：

（1）LSTM 模型的优势在于：

①训练效率高，资源消耗低；

②生成文本的局部连贯性较好；

③对短文本生成任务表现良好。

（2）Transformer 模型的优势在于：

①生成长文本的全局连贯性更佳；

②能更好捕捉人物关系和复杂情节；

③生成文本更具创意性和多样性。

### 4.2 中文处理的特殊挑战

研究发现中文武侠小说生成面临特有挑战：

（1）分词问题：字符级处理虽简单但可能忽略词语语义

（2）成语典故：武侠小说中大量使用四字成语和典故

（3）风格模仿：武侠特有的语言风格(如招式描写)难以准确捕捉

### 4.3 实际应用建议

根据研究结果，针对不同应用场景建议：

（1）快速原型开发：选择 LSTM 模型，迭代速度快

（2）高质量生成需求：选择 Transformer 模型，效果更优

（3）资源受限环境：可考虑小型化 Transformer 模型(DistilGPT)

## 5.结论与展望

### 5.1 研究结论

本研究通过对比实验得出以下结论：

（1）在小规模中文武侠小说数据集上，LSTM 和 Transformer 模型各有优劣；

（2）Transformer 在生成质量上整体占优，但需要更多训练资源；

（3）字符级分词策略适合中文小说生成任务；

（4）两种模型都能有效捕捉武侠小说的部分风格特征。

### 5.2 未来工作

未来研究可从以下方向展开：

（1）混合模型架构：结合 LSTM 和 Transformer 的优势；
（2）多模态生成：结合插图、音乐等元素增强表现力；
（3）交互式创作：开发人机协作的武侠小说创作系统；
（4）大规模评估：建立更全面的中文生成评估体系。

## 6.参考文献

[1] Reiter E, Dale R. Building natural language generation systems[M]. Cambridge university press, 2000.
[2] Brown P F, Desouza P V, Mercer R L, et al. Class-based n-gram models of natural language[J]. Computational linguistics, 1992, 18(4): 467-480.
[3] Graves A. Generating sequences with recurrent neural networks[J]. arXiv preprint arXiv:1308.0850, 2013.