

# P 大树洞生成器

匡宇轩 (2100013089)<sup>1</sup>, 陈红韵 (2100013130)<sup>1</sup>, and 王天源  
(1700017703)<sup>2</sup>

<sup>1</sup> 北京大学信息科学技术学院

<sup>2</sup> 北京大学元培学院

2022 年 6 月

### 摘要

本项目为北京大学信息科学技术学院国家精品课程《人工智能引论》的课程项目。本文为该项目的总结报告。

本项目受到北京大学树洞的启发。树洞作为匿名交流论坛，是一个天然的自然语言处理语料库（在遵守相关规定的情况下）。

本项目基于因果语言建模 (CLM) 的原理开发，使用 Python 和 PyTorch 实现。

我们在树洞文本数据集上训练 LSTM 模型，并根据用户的输入，使用训练好的模型产生回复。我们将模型命名为 HoleAI。

**关键词：**自然语言处理；长短期记忆网络；因果语言建模

# 目录

<b>1 背景介绍</b>	<b>3</b>
<b>2 项目细节</b>	<b>4</b>
2.1 数据集收集与预处理 . . . . .	4
2.2 LSTM 介绍 . . . . .	4
2.3 神经网络构建 . . . . .	4
2.4 神经网络训练 . . . . .	5
<b>3 训练结果分析</b>	<b>6</b>
<b>4 模型生成</b>	<b>6</b>
4.1 生成样例 . . . . .	6
4.2 样例分析 . . . . .	6
<b>5 模型部署与可视化</b>	<b>6</b>
5.1 项目介绍网站 . . . . .	6
5.2 人机交互网站 . . . . .	6
<b>6 项目总结</b>	<b>6</b>
6.1 意义 . . . . .	6
6.2 挑战 . . . . .	6
6.3 局限性 . . . . .	6
6.4 收获 . . . . .	6
<b>7 附录</b>	<b>7</b>
7.1 模型具体代码 . . . . .	7
7.2 github 链接 . . . . .	7
7.3 训练完整日志 . . . . .	7
7.4 其他 . . . . .	7
<b>8 参考文献</b>	<b>7</b>

## 1 背景介绍

P 大树洞作为校内学生匿名交流平台，是一个完美的语料库（在遵守相关规定的情况下），十分适合 NLP 相关任务。

本项目通过收集大量树洞文本数据，用神经网络进行拟合，希望能创造出一个具有 P 大学生气质的 AI。

使用者可以作为“洞主”，模拟发树洞的过程，但是回复的字母君都是 AI 自动生成。

## 2 项目细节

本项目的框架为：

- (1). 数据集收集与预处理
- (2). LSTM 介绍
- (3). 神经网络构建
- (4). 神经网络训练

### 2.1 数据集收集与预处理

鉴于 P 大树洞的相关管理规范，很遗憾我们无法使用爬虫技术获取数据集，因此我们采用了手动收集的方式收集数据。

经过收集，我们获取了一份包含 3000 余条树洞，约一百万字的树洞文本，其中包含了三部分：日常话题、经典话题和神洞（即回复量较多的树洞）。

数据收集完毕后，我们对数据进行了清洗。主要包括：去除洞号、回复数、收藏数等无用信息；去除出现频率过低的字符；去除大量重复的字符。

### 2.2 LSTM 介绍

$$\begin{aligned}f_t &= \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \\i_t &= \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \\\tilde{C}_t &= \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \\C_t &= f_t * C_{t-1} + i_t * \tilde{C}_t \\o_t &= \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \\h_t &= o_t * \tanh(C_t) \\f_t &= \sigma(W_f \cdot [C_{t-1}, h_{t-1}, x_t] + b_f) \\i_t &= \sigma(W_i \cdot [C_{t-1}, h_{t-1}, x_t] + b_i) \\o_t &= \sigma(W_o \cdot [C_t, h_{t-1}, x_t] + b_o) \\C_t &= f_t * C_{t-1} + (1 - f_t) * \tilde{C}_t \\z_t &= \sigma(W_z \cdot [h_{t-1}, x_t]) \\r_t &= \sigma(W_r \cdot [h_{t-1}, x_t]) \\\tilde{h}_t &= \tanh(W \cdot [r_t * h_{t-1}, x_t]) \\h_t &= (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t\end{aligned}$$

### 2.3 神经网络构建

构建如下

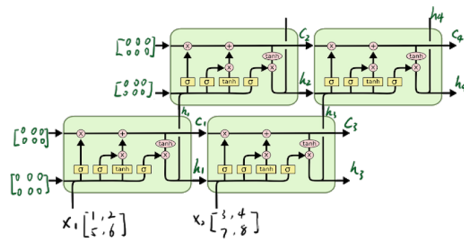


图 1: LSTM

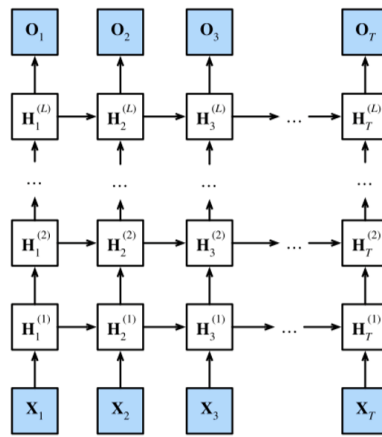


图 2: LSTM

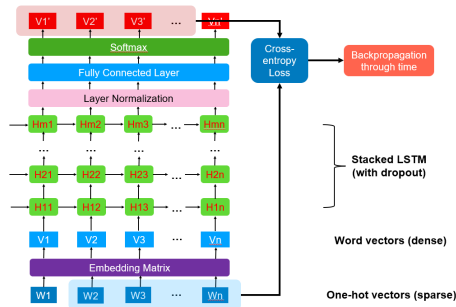


图 3: 神经网络构建

## 2.4 神经网络训练

经过四小时的训练，我们一共进行了 50 次训练迭代。

### 3 训练结果分析

1. dropout rate 不宜过大, num layers 不宜过多 (优化梯度回传)
2. 随机 split 数据集, 可以起到 data augmentation 的作用, 但也因此丢失了部分语义信息
3. 隐层大小至关重要

### 4 模型生成

#### 4.1 生成样例

#### 4.2 样例分析

### 5 模型部署与可视化

#### 5.1 项目介绍网站

#### 5.2 人机交互网站

### 6 项目总结

#### 6.1 意义

这个任务十分新颖有趣, 而且具有一定社会科学方面的研究价值。

#### 6.2 挑战

然而由于数据集的缺乏 (根本没有) 和数据集不够 clean, 给我们带来了很大的挑战。模型大小与数据集大小的匹配则是另一个挑战。

#### 6.3 局限性

#### 6.4 收获

不断学习新的技术, 看 API 文档成为家常便饭。

## 7 附录

### 7.1 模型具体代码

### 7.2 github 链接

### 7.3 训练完整日志

如下：

训练任务		任务描述							
任务名称		描述	模型名称	模型大小	模型大小	模型大小	模型大小	模型大小	模型大小
50_256_3_50		0505	hole_merge	否	• 模型	2022-05-07 22:29:56	2h54m46s	详情	删除
50_128_3_50		0505	hole_merge	否	• 模型	2022-05-07 22:19:29	4h0m13s	详情	删除
30_512_4_50		0505	hole_merge	否	• 模型	2022-05-07 19:42:33	3h37m27s	详情	删除
30_256_4_50		0505	hole_merge	否	• 模型	2022-05-07 18:13:17	3h30m43s	详情	删除

图 4: 训练日志

Name	Size	Input words	Hidden size	Number of layers	Final val loss
HoleAI-small	4.7MB	50	128	3	1.5476
HoleAI-medium	12.6MB	50	256	3	0.4562
HoleAI-large	37.8MB	30	512	3	0.4354
HoleAI-ultra	46.2MB	30	512	4	0.4640

图 5: 训练日志

### 7.4 其他

## 8 参考文献