

演示视频说明：

我们的项目采取的是模式化功能实现在先，如果无法用预设好的语言回答之后再进行基于GPT-2模型的进行NLP文本生成产生答复。

1. all演示：

即项目的最终成品的展示，其中包含了我们所有的实现功能，具体也罗列在本文的最后了。

2. nlp范例：

这是前端搭建好之后，直接连接上模型进行纯文本生成的结果，可以直观地感受文本生成的质量。也附有以下效果图：



```
samples.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
聊天记录2022-05-18 16:22:59.560898:
user:你好
chatbot:你好
user:你是谁啊
chatbot:陌生人
user:好吧
chatbot:嘿嘿
user:你在哪里啊
chatbot:在学校呢
user:现在还在上课吗
chatbot:是呀
user:好巧，我也是
chatbot:哈哈，我们周六还有课呢
user:啊这，有点惨啊
chatbot:你不是还有天么
user:嗯对
chatbot:真好
user:哈哈哈哈哈
chatbot:你也好
user:确实嘿
chatbot:嘿嘿
user:你们 那边天气咋嘛样
chatbot:一天比一天好，万里晴朗
user:真的假的
chatbot:真的呀
user:我们这边也还不错，就是有点热
chatbot:还好啦，我们这边也冷的不行
user:好的吧
chatbot:你们那天不是有暖气吗？
user:嗯对，冬天确实有
chatbot:我们这边没暖气，但还是有点冷
user:要是有了就好了
chatbot:嗯嗯，就是，我们这儿也冷，不能带多厚的衣服，一个个
user:哈哈哈哈哈
chatbot:嗯嗯，就是，太冷了，所以一直没买
user:知道啦
chatbot:嗯嗯
```

实现功能：

1. NLP语句生成

- structure: GPT-2
- training process

50W-Chinese conversation

40 epochs

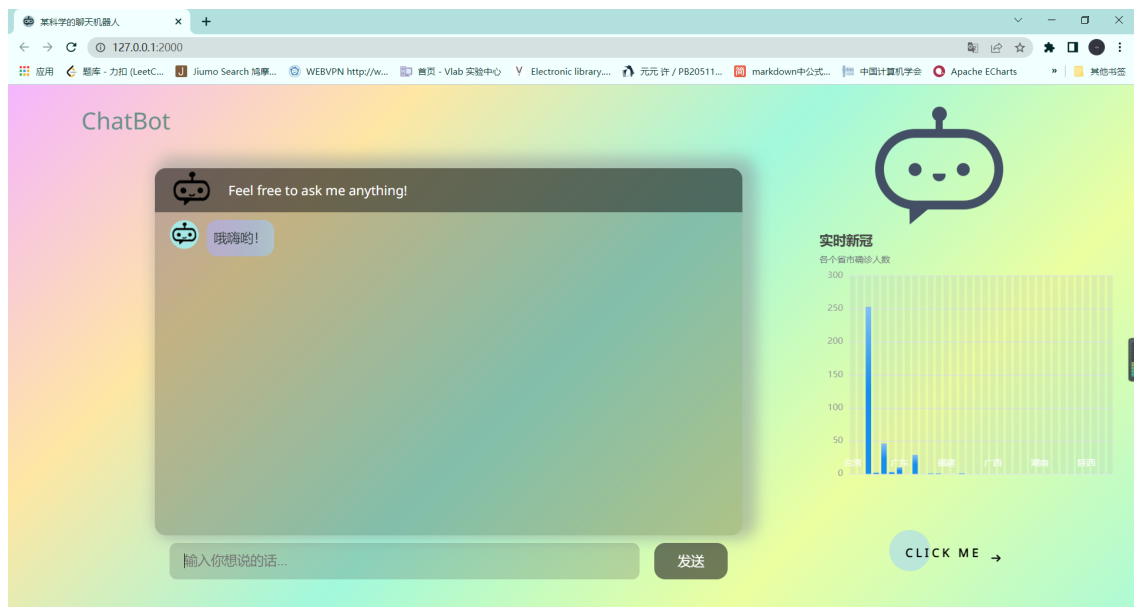
3 days

- srcs: ./train

```
0proj
├── data_parallel.py
├── pre_pro.py
├── response.py
├── train_model.py
├── .vscode
│   └── settings.json
├── config
│   └── config.json
├── data
│   ├── train.log
│   ├── train.pkl
│   └── train.txt
├── list
│   └── list.txt
├── model
│   └── epoch30
│       ├── config.json
│       └── pytorch_model.bin
├── sample
│   └── samples.txt
├── __pycache__
│   ├── dataset.cpython-39.pyc
│   └── pytorchtools.cpython-39.pyc
```

2. 一个平平无奇的最好看的前端

CSS、js、html



3. flask框架

前后端沟通

4. 数据运算

+, -, *, /, 含括号的复杂运算

5. 推荐系统

- 小说
- 电影

分专题、库推荐

6. 当日天气查询

指定城市查询

7. 当日疫情查询

- echarts**图表**：数据来源爬取到的最前的5个城市
- **指定城市查询**

8. 今日热搜

返回热搜最热项 (**点击跳转功能**)