

《大模型应用与实践》实验 报告

姓名：方星皓，王楚谖，王培成

学院：计算机科学与技术

专业：计算机科学与技术

报告日期：2024/9/4



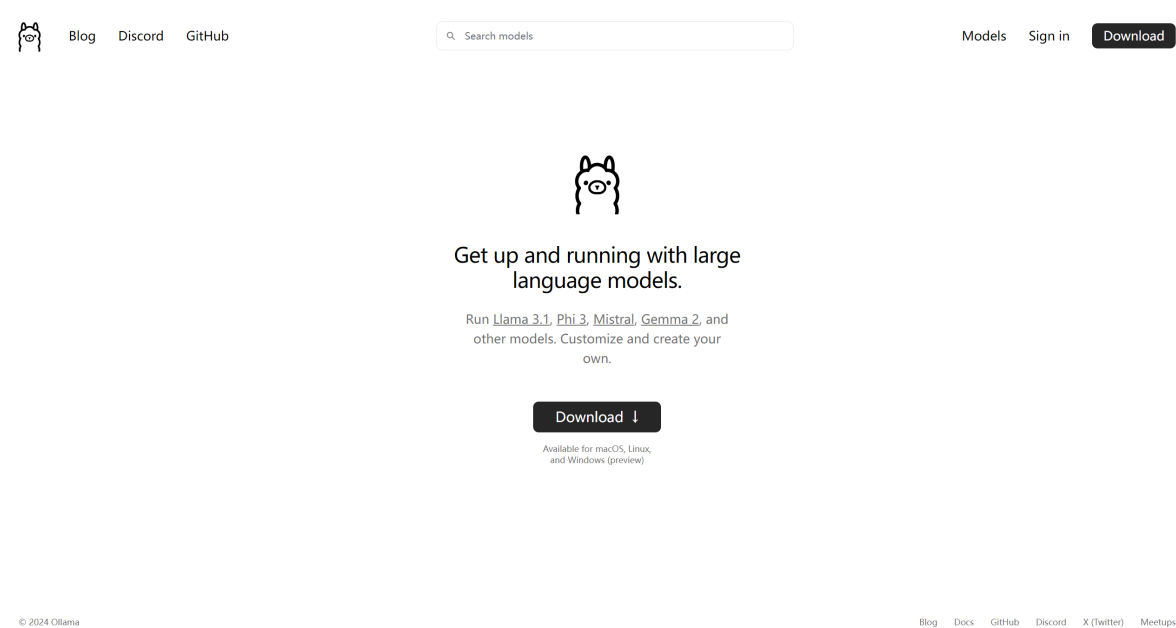
Assignment2 <使用大语言模型进行数据分析和可视化>

<方星皓, 王楚溟, 王培成>

1 实验实现过程

1.明确实验要求，下载下发的文件并建立项目。

2.在本地下载并配置 Ollama 运行环境，进行框架的构建。



```
C:\Windows\System32\Windc x + v

Welcome to Ollama!

Run your first model:

ollama run llama3.1

PS C:\Windows\System32> ollama list
NAME ID SIZE MODIFIED
PS C:\Windows\System32> ollama mistral-latest
Error: unknown command "mistral-latest" for "ollama"
PS C:\Windows\System32> ollama run mistral-latest
pulling manifest
Error: pull model manifest: file does not exist
PS C:\Windows\System32> ollama run mistral:latest
pulling manifest
pulling ff82381e2bea... 10% [progress bar] | 391 MB/4.1 GB 3.7 MB/s 16m33s
```

3.进行需求的具体分析以及相关代码框架的实现。

4.通过调用toolkit，使用 Ollama 提供的功能来构建和训练模型。

5.指定 LLM 接口内容与调用方式。

```
def generate_full_completion(model: str, prompt: str) -> dict[str, str]:
    params = {"model": model, "prompt": prompt, "stream": False}
    response = requests.post(
        "http://localhost:11434/api/generate",
        headers={"Content-Type": "application/json"},
        data=json.dumps(params),
        timeout=1800, # 调整设置为1800, 300限制过低
    )
    return json.loads(response.text)
```

6.通过指定明确的背景与底层信息，进行模型的 tuning。

```
{function_to_json(write_csv_from_text)}
{function_to_json(calculate_population_statistics)}
{function_to_json(visualize_population_distribution)}
{function_to_json(visualize_population_trend)}
You must follow these instructions:
Always select one or more of the above tools based on the user query
If a tool is found, you must respond in the JSON format matching the following schema:
{{
  "tools": [{
    "tool": "<name of the selected tool>",
    "tool_input": <parameters for the selected tool, matching the tool's JSON schema
  }]
}}
If there are multiple tools required, make sure a list of tools are returned in a JSON array.
If there is no tool that match the user request, you must respond empty JSON {}.
```

7.确定检验的问题集，写入 prompts。

```

GPT_MODEL = "mistral:latest"

prompts = [
    "Read from the text file 'permanent_population.txt' and write as csv file: remove the table header and footer, save as a new CSV file",
    "Calculate the mean, maximum, and minimum population values for each region in the given CSV file 'permanent_population_cleaned.csv'",
    "Visualize the population distribution for the last 10 years in the given CSV file 'permanent_population_cleaned_stats.csv' as a pie chart",
    "Visualize the population trend for the last 10 years of '浙江省' in the given CSV file 'permanent_population_cleaned.csv' as a line chart",
]

for prompt in prompts:
    print(f"🔥 {prompt}")
    question = functions_prompt + prompt
    response = generate_full_completion(GPT_MODEL, question)
    try:
        data = json.loads(response.get("response", response))
        # print(data)
        for tool_data in data["tools"]:
            execute_fuc(tool_data)
    except Exception:
        print('No tools found.')
    print(f"Total duration: {int(response.get('total_duration')) / 1e9} seconds")

```


8.进行代码的具体实现与整合，使 LLM 输出的结果最终满足 Assignment 的要求。

9.继续训练，增加模型的泛化能力，并使其支持更加美观的输出格式。


2 实验结果与分析

2.1 Github 框架

我们使用 github 进行了小组作业的共享维护（网址：<https://github.com/miraitowaves/prompt-engineering>）。


prompt-engineering
Public

main
1 Branch
0 Tags
Go to file
Add file

 miraitowaves	修改一些小细节	3d766f8 · 11 hours ago
assets	更新数据处理部分	
src	修改一些小细节	
.gitattributes	Initial commit	
.gitignore	Initialize	
LICENSE	Initial commit	
README.md	更新数据处理部分	

README
MIT license

其中，assets 文件夹用于存储模型的输入输出文件，如初始给出的文本文件以及在运行过程中产生的 csv 文件、用于部署柱状图、饼状图等 html 文件、在过程中根据指令进行修改的 csv 文件等。

2.2 模型输出

2.2.1 csv 生成

prompt:

"Read from the text file 'permanent_population.txt' and write as csv file: remove the table header and footer, save as a new CSV file"

模型输出：

	地区	2023年	2022年	2021年	2020年	2019年	2018年	2017年	2016年	2015年	2014年					
1	地区															
2	北京市	2186	2184	2189	2189	2190	2192	2194	2195	2188	2171					
3	天津市	1364	1363	1373	1387	1385	1383	1410	1443	1439	1429					
4	河北省	7393	7420	7448	7464	7447	7426	7409	7375	7345	7323					
5	山西省	3466	3481	3480	3490	3497	3502	3510	3514	3519	3528					
6	内蒙古自治区	2396	2401	2400	2403	2415	2422	2433	2436	2440	2449					
7	辽宁省	4182	4197	4229	4255	4277	4291	4312	4327	4338	4358					
8	吉林省	2339	2348	2375	2399	2448	2484	2526	2567	2613	2642					
9	黑龙江省	3062	3099	3125	3171	3255	3327	3399	3463	3529	3608					
10	上海市	2487	2475	2489	2488	2481	2475	2466	2467	2458	2467					
11	江苏省	8526	8515	8505	8477	8469	8446	8423	8381	8315	8281					
12	浙江省	6627	6577	6540	6468	6375	6273	6170	6072	5985	5890					
13	安徽省	6121	6127	6113	6105	6092	6076	6057	6033	6011	5997					
14	福建省	4183	4188	4187	4161	4137	4104	4065	4016	3984	3945					
15	江西省	4515	4528	4517	4519	4516	4513	4511	4496	4485	4480					
16	山东省	10123	10163	10170	10165	10106	10077	10033	9973	9866	9808					
17	河南省	9815	9872	9883	9941	9901	9864	9829	9778	9701	9645					
18	湖北省	5838	5844	5830	5745	5927	5917	5904	5885	5850	5816					
19	湖南省	6568	6604	6622	6645	6640	6635	6633	6625	6615	6611					
20	广东省	12706	12657	12684	12624	12489	12348	12141	11908	11678	11489					
21	广西壮族自治区	5027	5047	5037	5019	4982	4947	4907	4857	4811	4770					
22	海南省	1043	1027	1020	1012	995	982	972	957	945	936					
23	重庆市	3191	3213	3212	3209	3188	3163	3144	3110	3070	3043					
24	四川省	8368	8374	8372	8371	8351	8321	8289	8251	8196	8139					
25	贵州省	3865	3856	3852	3858	3848	3822	3803	3758	3708	3677					
26	云南省	4673	4693	4690	4722	4714	4703	4693	4677	4663	4653					
27	西藏自治区	365	364	366	366	361	354	349	340	330	325					
28	陕西省	3952	3956	3954	3955	3944	3931	3904	3874	3846	3827					
29	甘肃省	2465	2492	2490	2501	2509	2515	2522	2520	2523	2531					
30	青海省	594	595	594	593	590	587	586	582	577	576					
31	宁夏回族自治区	729	728	725	721	717	710	705	695	684	678					
32	新疆维吾尔自治区	2598	2587	2589	2590	2559	2520	2480	2428	2385	2325					

2.2.2 对表格的增改

prompt:

"Calculate the mean, maximum, and minimum population values for each region in the given CSV file 'permanent_population_cleaned.csv'"

模型输出：

permanent_population_cleaned_stats.csv - Excel															
文件 开始 OfficePLUS 插入 页面布局 公式 数据 审阅 视图 PDF工具箱 帮助 操作说明搜索															
等线 11 A A 自动换行 常规 表格模板 表格美化 条件格式 套用 单元格样式															
B I U 字体 对齐方式 数字 OfficePLUS 样式															
A1 地区															
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	地区	2023年	2022年	2021年	2020年	2019年	2018年	2017年	2016年	2015年	2014年	均值	最大值	最小值	
2	北京市	2186	2184	2189	2189	2190	2192	2194	2195	2188	2171	2187.8	2195	2171	
3	天津市	1364	1363	1373	1387	1385	1383	1410	1443	1439	1429	1397.6	1443	1363	
4	河北省	7393	7420	7448	7464	7447	7426	7409	7375	7345	7323	7405	7464	7323	
5	山西省	3466	3481	3480	3490	3497	3502	3510	3514	3519	3528	3498.7	3528	3466	
6	内蒙古自治区	2396	2401	2400	2403	2415	2422	2433	2436	2440	2449	2419.5	2449	2396	
7	辽宁省	4182	4197	4229	4255	4277	4291	4312	4327	4338	4358	4276.6	4358	4182	
8	吉林省	2339	2348	2375	2399	2448	2484	2526	2567	2613	2642	2474.1	2642	2339	
9	黑龙江省	3062	3099	3125	3171	3255	3327	3399	3463	3529	3608	3303.8	3608	3062	
10	上海市	2487	2475	2489	2488	2481	2475	2466	2467	2458	2467	2475.3	2489	2458	
11	江苏省	8526	8515	8505	8477	8469	8446	8423	8381	8315	8281	8433.8	8526	8281	
12	浙江省	6627	6577	6540	6468	6375	6273	6170	6072	5985	5890	6297.7	6627	5890	
13	安徽省	6121	6127	6113	6105	6092	6076	6057	6033	6011	5997	6073.2	6127	5997	
14	福建省	4183	4188	4187	4161	4137	4104	4065	4016	3984	3945	4097	4188	3945	
15	江西省	4515	4528	4517	4519	4516	4513	4511	4496	4485	4480	4508	4528	4480	
16	山东省	10123	10163	10170	10165	10106	10077	10033	9973	9866	9808	10048.4	10170	9808	
17	河南省	9815	9872	9883	9941	9901	9864	9829	9778	9701	9645	9822.9	9941	9645	
18	湖北省	5838	5844	5830	5745	5927	5917	5904	5885	5850	5816	5855.6	5927	5745	
19	湖南省	6568	6604	6622	6645	6640	6635	6633	6625	6615	6611	6619.8	6645	6568	
20	广东省	12706	12657	12684	12624	12489	12348	12141	11908	11678	11489	12272.4	12706	11489	
21	广西壮族自治区	5027	5047	5037	5019	4982	4947	4907	4857	4811	4770	4940.4	5047	4770	
22	海南省	1043	1027	1020	1012	995	982	972	957	945	936	988.9	1043	936	
23	重庆市	3191	3213	3212	3209	3188	3163	3144	3110	3070	3043	3154.3	3213	3043	
24	四川省	8368	8374	8372	8371	8351	8321	8289	8251	8196	8139	8303.2	8374	8139	
25	贵州省	3865	3856	3852	3858	3848	3822	3803	3758	3708	3677	3804.7	3865	3677	
26	云南省	4673	4693	4690	4722	4714	4703	4693	4677	4663	4653	4688.1	4722	4653	
27	西藏自治区	365	364	366	366	361	354	349	340	330	325	352	366	325	
28	陕西省	3952	3956	3954	3955	3944	3931	3904	3874	3846	3827	3914.3	3956	3827	
29	甘肃省	2465	2492	2490	2501	2509	2515	2522	2520	2523	2531	2506.8	2531	2465	
30	青海省	594	595	594	593	590	587	586	582	577	576	587.4	595	576	
31	宁夏回族自治区	729	728	725	721	717	710	705	695	684	678	709.2	729	678	
32	新疆维吾尔自治区	2598	2587	2589	2590	2559	2520	2480	2428	2385	2325	2506.1	2598	2325	
33															

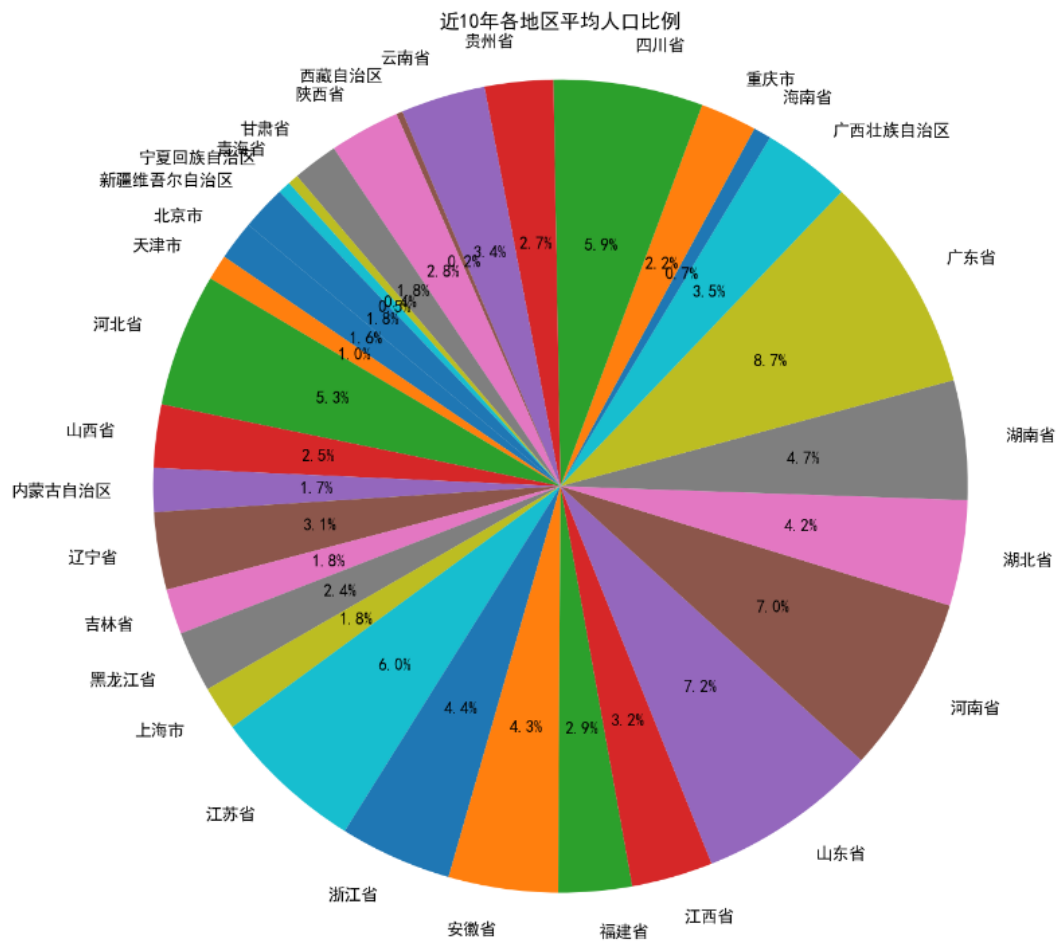
经检验，均值、最大值、最小值的计算完全正确。

2.2.3 对表格纵向的查询以及可视化

prompt:

"visualize the population distribution for the last 10 years in the given csv file 'permanent_population_cleaned_stats.csv' as a pie chart"

模型输出：



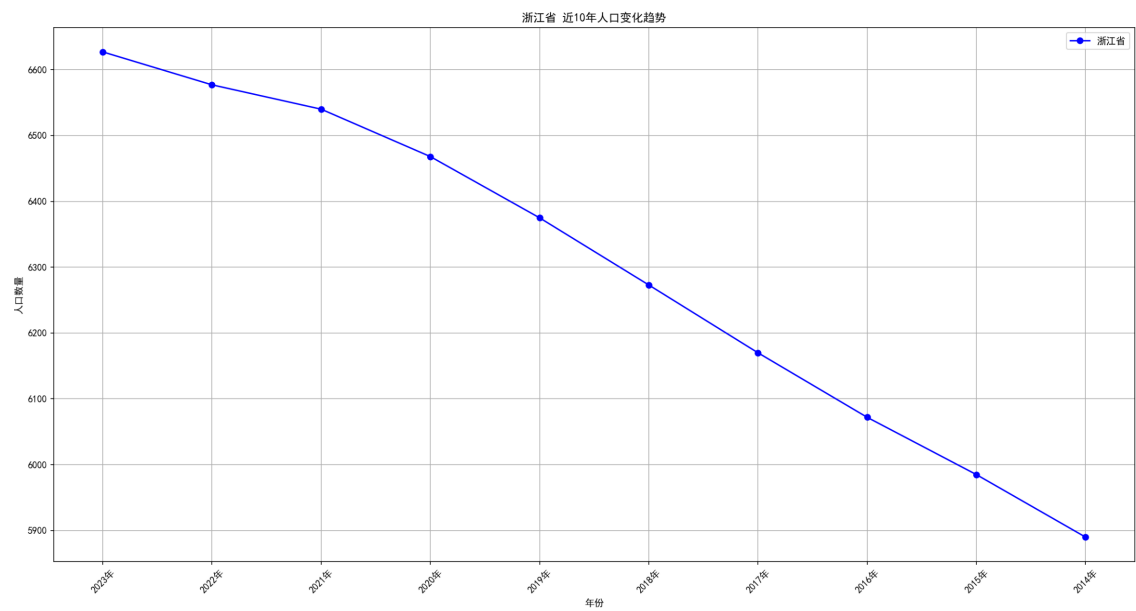
经计算核验，输出的比例正确。

2.2.4 对表格横向的查询以及可视化

prompt:

"Visualize the population trend for the last 10 years of '浙江省' in the given CSV file 'permanent_population_cleaned.csv' as a line chart"

模型输出:

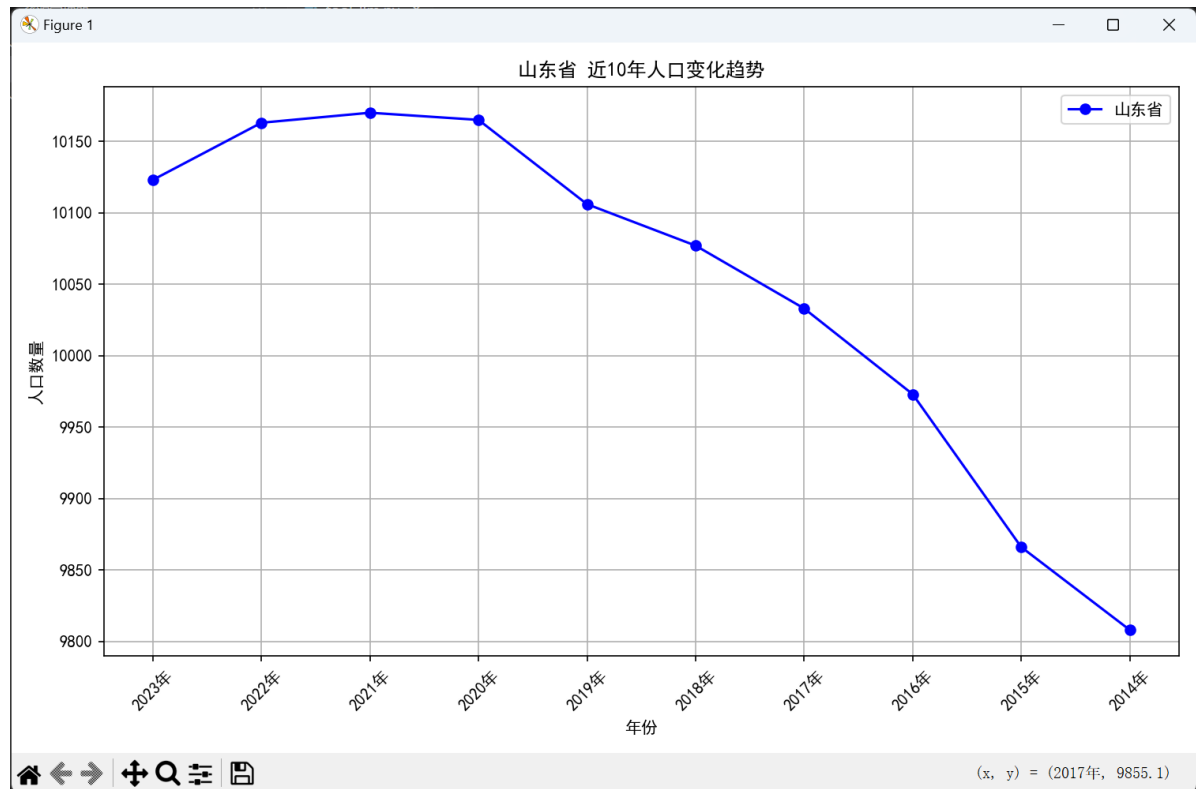


同时，根据对函数具体的实现，模型在这一问题上有比较好的泛化能力。

如，我们将提示词改为：

"visualize the population trend for the last 10 years of '山东省' in the given CSV file 'permanent_population_cleaned.csv' as a line chart"

模型输出就会变为：



2.3 模型目标 prompts 顺序连续执行结果

我们将目标问题进行连续输入，并考察模型对问题的解决能力。

```
prompts = [
    "Read from the text file 'permanent_population.txt' and write as csv file: remove the table header and footer, save as a new CSV file",
    "Calculate the mean, maximum, and minimum population values for each region in the given CSV file 'permanent_population_cleaned.csv'",
    "Visualize the population distribution for the last 10 years in the given CSV file 'permanent_population_cleaned_stats.csv' as a pie chart",
    "Visualize the population trend for the last 10 years of '浙江省' in the given CSV file 'permanent_population_cleaned.csv' as a line chart",
]
```

模型输出：

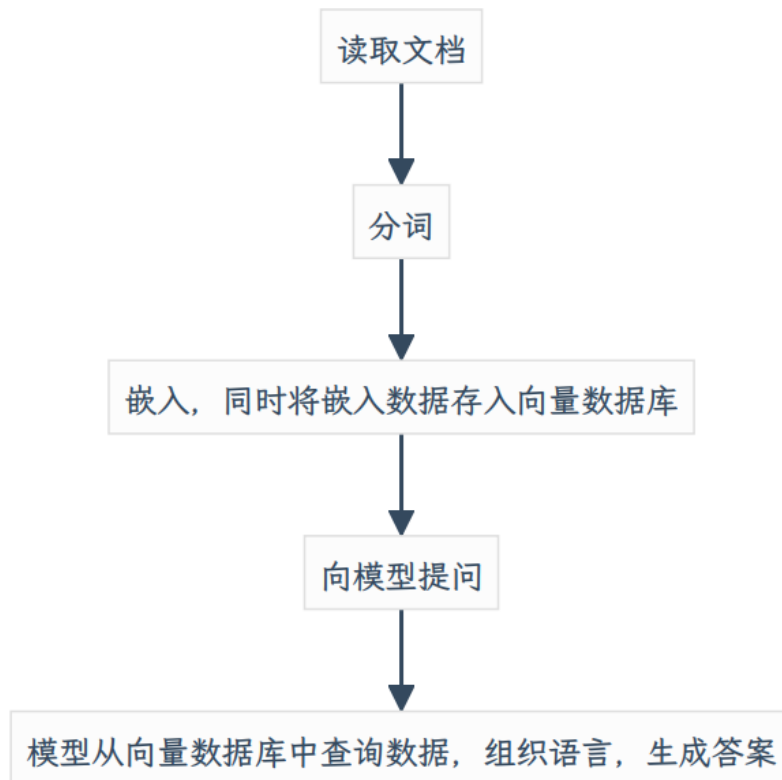
```
PS E:\ZJU\大模型\Assignment2\src> py .\tool_llm.py
🤖Read from the text file 'permanent_population.txt' and write as csv file: remove the table header and footer, save as a new CSV file
Reading the text file E:\ZJU\大模型\Assignment2\src\..\assets\permanent_population.txt...
Cleaned data saved to E:\ZJU\大模型\Assignment2\src\..\assets\permanent_population_cleaned.csv.
Total duration: 1.5110121 seconds
🤖Calculate the mean, maximum, and minimum population values for each region in the given CSV file 'permanent_population_cleaned.csv'
Reading the text file E:\ZJU\大模型\Assignment2\src\..\assets\permanent_population_cleaned.csv...
计算完成，结果已保存至 E:\ZJU\大模型\Assignment2\src\..\assets\permanent_population_cleaned_stats.csv
Total duration: 1.2672583 seconds
🤖Visualize the population distribution for the last 10 years in the given CSV file 'permanent_population_cleaned_stats.csv' as a pie chart
Reading the text file E:\ZJU\大模型\Assignment2\src\..\assets\permanent_population_cleaned_stats.csv...
No tools found.
Total duration: 1.3401359 seconds
🤖Visualize the population trend for the last 10 years of '浙江省' in the given CSV file 'permanent_population_cleaned.csv' as a line chart
Reading the text file E:\ZJU\大模型\Assignment2\src\..\assets\permanent_population_cleaned.csv...
E:\ZJU\大模型\Assignment2\src\..\assets\population_line_chart.html
省份 浙江省 的人口变化趋势图已完成。
Total duration: 1.6096981 seconds
```

经检验，模型对四个问题均按顺序作出了正确的回答。

3 额外的工作

在进行数据分析的Agent的开发的时候，除了上述实验实现的，基于文本分析自动调用工具的Agent外，在一开始的实验目标不明确的时候，我们最开始尝试的其实是一个基于外部知识库的LLM，采用了RAG技术和LangChain工具

3.1 RAG工作流程



3.2 实验流程

除了基本模型外，我们还需要下载一个可以进行文档向量化的模型 `nomic-embed-text`：

```
ollama run nomic-embed-text
```

然后，参考[LangChain的官方文档](#)，可知，我们需要如下三个部分：

- Document Loaders

1. 读取文件

```
documents = TextLoader("../assets/data_for_rag_llm.txt").load()
```

- 分词器 Text Splitter

2. 分词

```
text_splitter = CharacterTextSplitter.from_tiktoken_encoder(chunk_size=7500,
chunk_overlap=100)
doc_splits = text_splitter.split_documents(documents)
```

- 内存型向量数据库 DocArray InMemorySearch

3. 嵌入和存储

```
embeddings = OllamaEmbeddings(model='nomic-embed-text')
vectorstore = DocArrayInMemorySearch.from_documents(doc_splits, embeddings)
retriever = vectorstore.as_retriever()
```

提供的知识库内容如下：

公式：主=6，这是一个恒等式

问题如下：

请问 主 + 1 等于多少？

LLM的回答如下：

根据提供的文档内容，“公式：主=6，这是一个恒等式”，我们可以推断出“主”代表的数值是6。
因此，如果要计算“主+1”的结果，则为：
 $6(\text{主}) + 1 = 7$
所以，主 + 1 等于 7。

3.3 图标输出格式优化

在使用LLM生成饼图和折线图之后，使用python的mpld3库将生成结果保存为html格式文件，并存储在asset文件夹中

- 生成html格式折线图

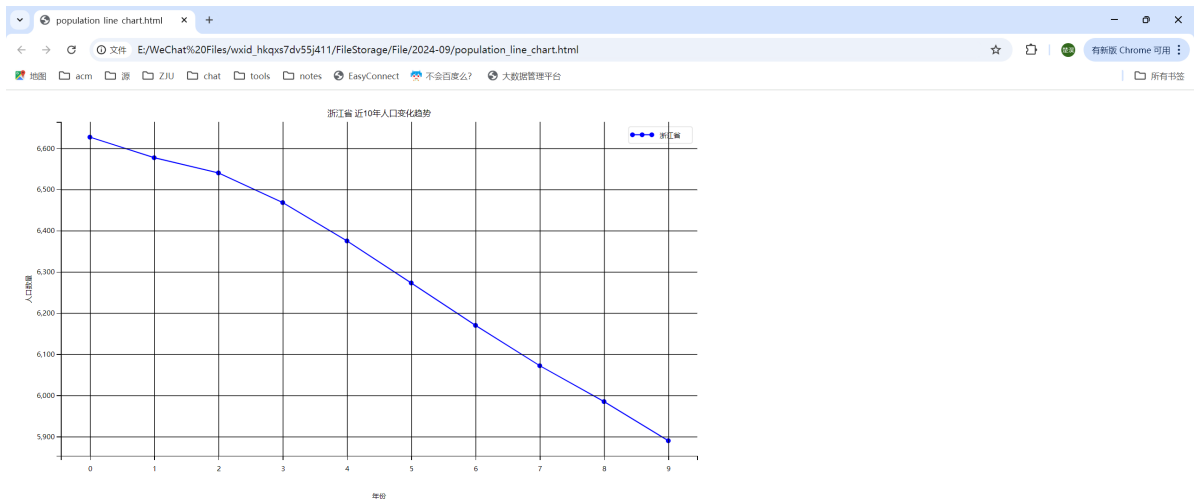
```
html_str = mpld3.fig_to_html(plt.gcf())
output_file_path = os.path.join(current_dir, '..', 'assets',
'population_line_chart.html')
print(output_file_path)
with open(output_file_path, "w", encoding='utf-8') as f:
    f.write(html_str)
```

- 生成html格式饼图

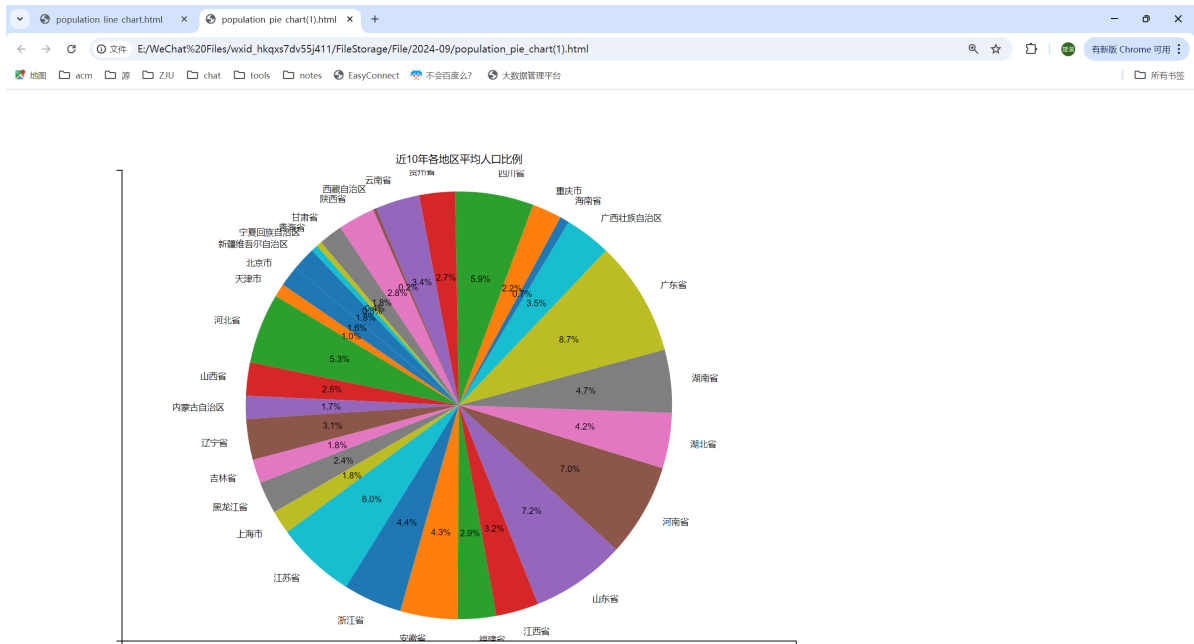
```
html_str = mpld3.fig_to_html(plt.gcf())
output_file_path = os.path.join(current_dir, '..', 'assets',
'population_pie_chart.html')
print(output_file_path)
with open(output_file_path, "w", encoding='utf-8') as f:
    f.write(html_str)
```

html模式下可以对生成的图标进行自由缩放查看，较为美观

- 生成折线图展示



• 生成的饼图展示



4 实验总结

在本次作业中，我们探索了使用大语言模型进行数据分析和可视化的能力。通过设计合适的 prompt，我们成功地利用 LLM 完成了从数据处理到可视化的多个步骤。这一过程展现了 LLM 在自动化数据操作中的强大潜力，特别是在数据清洗、分析和图表生成方面。

首先，我们通过 LLM 生成代码对文本文件进行了清理，去除表头表尾并保存为 CSV 文件。接着，我们使用 LLM 生成的代码对预处理后的 CSV 文件进行了深入分析，计算了各地区近十年的平均人口、最大值和最小值，并将其添加到表格中。随后，我们通过饼状图展示了各地区的平均人口比例，并用折线图展示了特定省份近十年的人口变化趋势。

通过本次实验，我们不仅加深了对大语言模型的理解，也锻炼了 prompt 设计的能力，证明了 LLM 在实际数据分析工作中的高效性和灵活性。未来，我们可以进一步探索如何优化生成代码的逻辑，使其更加适应复杂的分析任务，以及如何利用 LLM 提供更具创新性的可视化方案。这次作业的经验将为我们后续工作提供重要参考。

