

文科生零基础AI编程

——快速提升想象力和实操能力

清华大学
新闻学院 人工智能学院

@新媒沈阳团队

陶炜博士生

2025年2月

目录

01

文科生的
AI编程新思维

02

文生数据

03

文生图表

04

文生网站

05

文生软件

06

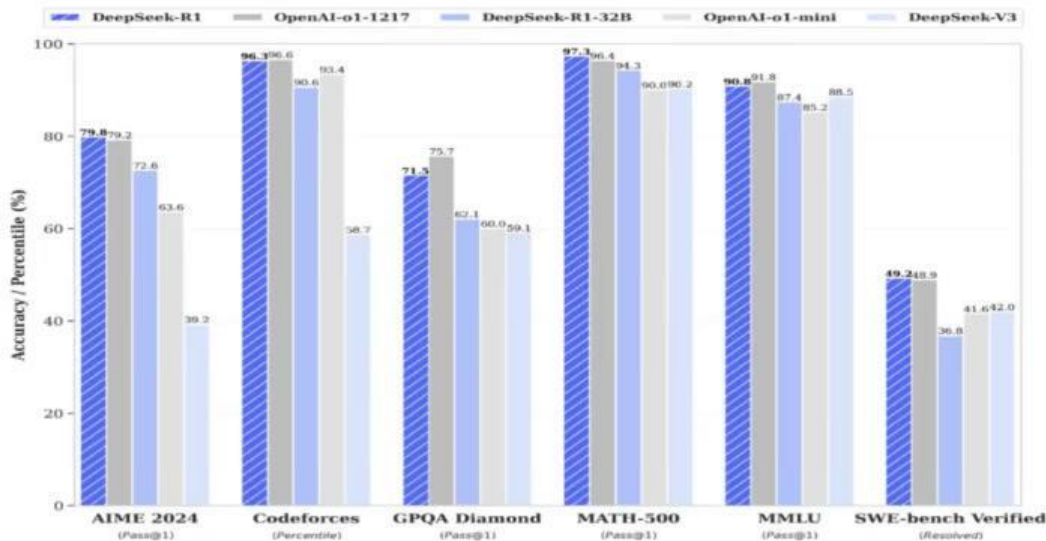
天人智一
问行合一

DeepSeek是什么？

- DeepSeek是一家专注通用人工智能（AGI）的中国科技公司，主攻大模型研发与应用。
- DeepSeek-R1是其开源的推理模型，擅长处理复杂任务且可免费商用。

性能对齐 OpenAI-o1 正式版

DeepSeek-R1 在后训练阶段大规模使用了强化学习技术，在仅有极少标注数据的情况下，极大提升了模型推理能力，在数学、代码、自然语言推理等任务上，性能比肩 OpenAI o1 正式版。



AI

+

国产

+

免费

+

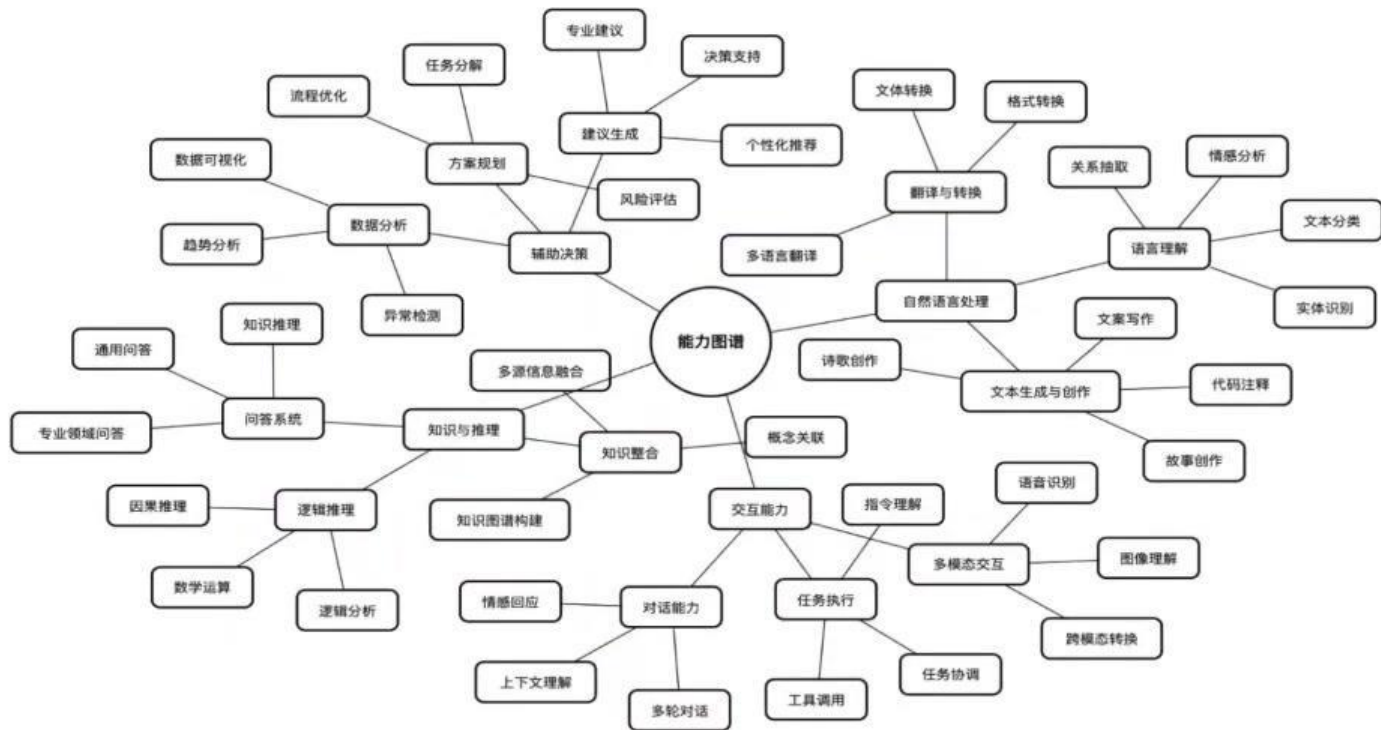
开源

+

强大

Deepseek可以做什么？

直接面向用户或者支持开发者，提供智能对话、文本生成、语义理解、计算推理、代码生成补全等应用场景，支持联网搜索与深度思考模式，同时支持文件上传，能够扫描读取各类文件及图片中的文字内容。



DeepSeek的深度思考过程独树一帜



《香料三重奏》茄椒肠卷配酸奶薄荷酱

灵感地图：巴尔干半岛香料 × 地中海清新感 × 日式天妇罗手法

结构解构：

1. 青椒薄片 - 采用日式天妇罗手法炸至半透明，形成琥珀色脆网
2. 茄泥慕斯 - 融入保加利亚红椒粉与希腊酸奶，制成空气感慕斯
3. 香肠脆粒 - 伊比利亚辣肠低温烘烤后粉碎成黄金脆粒
4. 薄荷冷萃油 - 液态氮急冻薄荷叶萃取的翡翠色精油

- 第1层：青椒脆片释放灯笼椒特有的清甜
- 第2层：烟熏红椒粉在慕斯中爆破出暖意
- 第3层：辣肠碎粒带来咸鲜微辣的颗粒感
- 终章：薄荷冷萃油在舌尖炸开极地般的清凉

分子料理技法：

1. 反卷造型：将传统卷物内外翻转，脆网外露包裹柔嫩内馅
2. 温差游戏：65°C 温热慕斯与 -196°C 冷萃油同盘
3. 质构悖论：看似坚硬的脆片入口即碎，柔软慕斯中暗藏爆破脆粒



已深度思考 (用时 18 秒) ^

好的，用户让我帮忙构思一道新菜，需要用茄子、青椒和香肠，而且要和现有的菜系不一样。首先，我得先分析这三个主要食材的特点。茄子比较吸味，口感软糯；青椒清脆，带点甜味或辣味；香肠通常是咸香，可能有烟熏味。接下来要考虑如何组合它们，创造出与众不同的风味。

用户提到要和现有菜系不同，所以不能走中式的鱼香茄子或者地三鲜，也不能是西式的焗烤之类。可能需要融合不同地区的调味料，或者采用不常见的烹饪方法。比如，可以考虑东南亚的香料，或者南美的风味，或者中东的调料。

然后，思考烹饪方法。茄子通常需要先处理，比如烤、煎或者炸，以去除生涩味。青椒可以生吃或快速烹饪保持脆感。香肠可能需要煎出油分，增加香味。或许可以尝试将茄子做成泥，青椒作为容器，香肠作为填充，这样结构上有新意。

调味方面，可以考虑使用非传统的组合，比如用椰奶和咖喱，或者用柠檬草和鱼露，甚至加入一些果酱来平衡咸味。或者用酸奶为基础的酱汁，加入香料，这样既有异域风情，又能融合食材。

还要考虑口感层次，软糯的茄子泥，脆嫩的青椒，香肠的嚼劲，再加上脆脆的坚果或种子。可能需要一些香草来提升清新感，比如薄荷或香菜。

接下来，需要确保这道菜的步骤不会太复杂，但又有创新点。比如，将茄子烤后挖空，混合香肠和香料，再填回茄子中，配上特制酱汁。或者做成卷的形式，用青椒包裹茄子和香肠的混合物。

另外，名字也很重要，要吸引人，反映菜品的特色。可能需要结合使用的特殊调料或烹饪方法来命名，比如“泰式椰香茄子青椒船”或者“地中海香料茄盒”。

如何使用 DeepSeek?

网页端: <https://chat.deepseek.com>

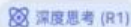
APP: **DeepSeek**



我是 DeepSeek, 很高兴见到你!

我可以帮你写代码、读文件、写作各种创意内容, 请把你的任务交给我吧~

给 DeepSeek 发送消息



深度思考 (R1)



联网搜索



模型规模	适用场景	推荐 GPU 配置	硬件成本
DeepSeek-7B	个人开发者、轻量级任务	1 张 RTX 3090 或 A10	2,000—3,000
DeepSeek-13B	中型团队、中等复杂度任务	1 张 A100 或 2 张 RTX 4090	5,000—8,000
DeepSeek-33B	大型团队、高复杂度任务	2 张 A100 或 4 张 RTX 4090	15,000—25,000
DeepSeek-70B	企业级应用、超大规模任务	4 张 A100 或 8 张 RTX 4090	50,000—100,000

常规绘图

SVG矢量图

- 基础图形
- 图标
- 简单插图
- 流程图
- 组织架构图

Mermaid图表

- 流程图
- 时序图
- 类图
- 状态图
- 实体关系图
- 思维导图

React图表

- 折线图
- 柱状图
- 饼图
- 散点图
- 雷达图
- 组合图表

自然语言理解与分析

语义分析

语义解析

情感分析（评论、反馈）

意图识别（客服对话、用户查询）

实体提取（人名、地点、事件）

文本分类

文本分类

主题标签生成（如新闻分类）

垃圾内容检测



知识推理

知识推理

逻辑问题解答（数学、常识推理）

因果分析（事件关联性）

— | 文科生的AI编程新思维

人机优生 人机快生



通过AI快速生成大量内容，追求效率和数量，适用于需要快速输出或覆盖广泛场景的任务，如批量文案、设计初稿等。



AI生成内容后，进行深度筛选和优化，注重质量而非数量，适用于需要精细打磨的场景，如高质量创作、精准决策等。

破除迷思与核心心法

破除迷思：无需代码语法，描述需求即可

传统编程需精通代码语法，文科生常望而却步。如今，只需清晰描述需求，AI即可转化。例如，想生成柱状图，只需说“用Python画柱状图，对比三国人物出场次数”，AI自动完成代码编写。

01

核心心法：AI是“技术翻译官”和“万能助手”

AI可将文科生的语言需求转化为技术逻辑，跨越技术门槛。比如，想设计问卷统计工具，说“用Excel公式自动统计问卷结果”，AI生成公式并解释逻辑。

02

成果导向：直接生成可运行的代码文件等

描述需求后，AI直接生成可运行代码文件、网页或软件，无需手动编写代码。以制作文化地图为例，说“制作唐代诗人行迹地图”，AI生成包含数据导入、地图绘制等功能的代码文件。

03

文科优势：故事思维与设计能力

结合文科的故事思维，将复杂需求以故事形式呈现，便于AI理解。

在设计软件界面时，文科生可从用户体验角度出发，设计出更符合用户习惯的界面。例如，制作互动小说平台，文科生可设计出吸引读者的剧情走向和交互方式。

04

AI重构编程范式



传统编程困境：语法壁垒 vs 逻辑表达需求

传统编程需掌握复杂语法，文科生常因语法错误无法实现需求。

例如，想实现数据排序功能，因语法错误导致代码无法运行，而AI可直接理解需求生成正确代码。

新范式核心：朴素的语言→抽象逻辑→代码实现

新范式以文学语言为起点，AI解析为抽象逻辑，再转化为代码实现。

比如，说“用Python实现数据分类”，AI先理解分类需求，再生成对应代码。

DeepSeek定位：智能语义解析器与逻辑架构师

DeepSeek可精准解析自然语言需求，构建逻辑架构，生成高效代码。

在开发文化传播模拟器时，DeepSeek根据描述构建模型，生成模拟代码。

个人优势：领域知识+需求能力设计+跨学科思维

无论什么学科，个人的领域知识、需求设计能力和跨学科思维是关键。

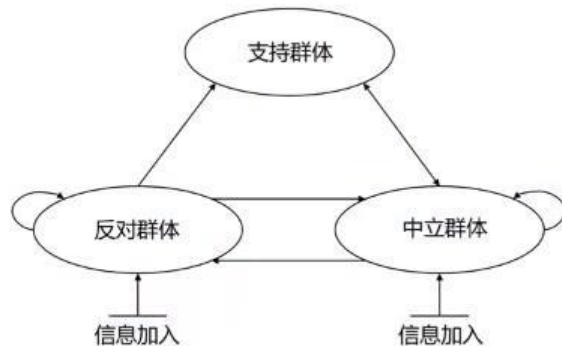
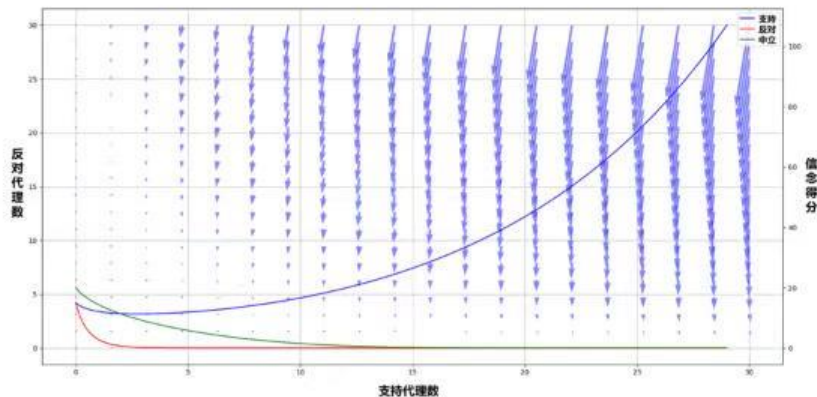
例如，文科生可利用历史知识设计文化传播模拟器，结合技术实现跨学科应用。

理论跨界：超越本领域的思维惯性

“

超越本领域的计算惯性，实现小范围和中范围的创新。例如，在研究意见群体博弈时，可以借鉴生物学中的多种群博弈模型，理解不同意见群体之间的互动和竞争关系。

通过这种跨学科的融合，研究者可以提出新的假设和研究方法，丰富和拓展传统社会科学的研究框架。这不仅有助于更全面地理解社会现象，还能推动学科之间的互动与合作，促进科学研究的整体进步。



提示词生成原则

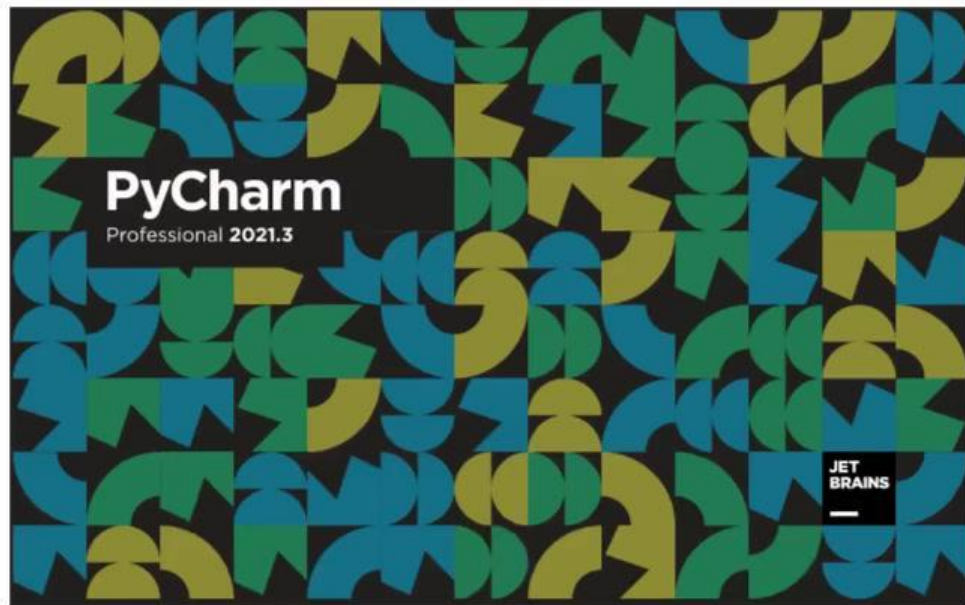


领域限定法：“作为历史学者，需要开发一个历史事件时间轴展示软件，约束条件包括时间轴要精确到年份，事件描述要简洁准确”。

分步求解法：“将复杂需求分解为3个阶段任务链，第一阶段收集数据，第二阶段分析数据，第三阶段生成可视化报告”。

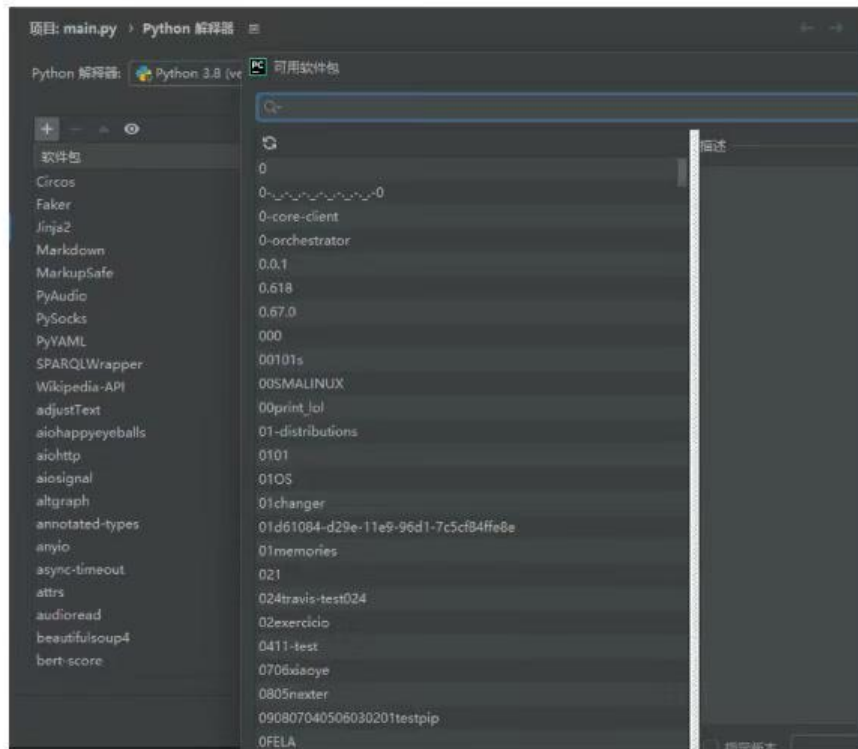
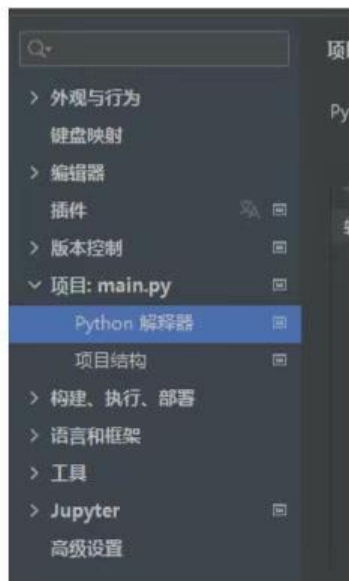
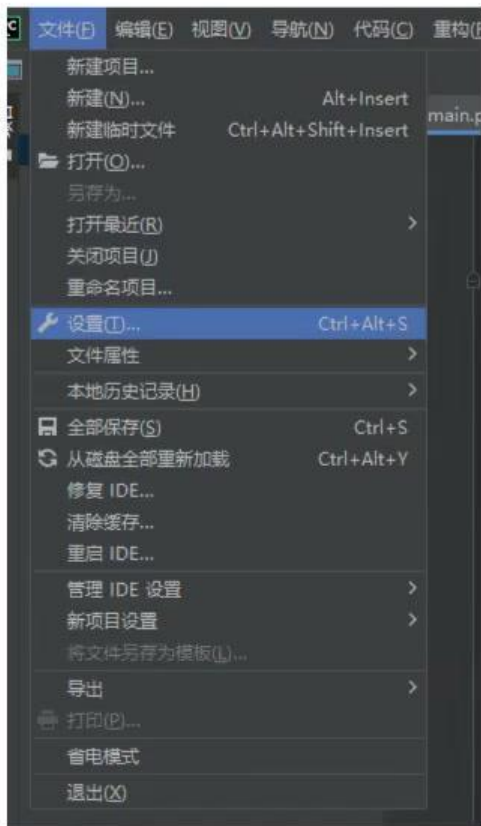
语义规范化：“使用领域术语替代技术术语，使提示词更符合领域专业习惯，便于 AI 理解。”

实操前的准备：可用的AI+本地代码编译器



实操前的准备：安装各类库

文件-设置-python解释器-点击+号查询



Python数据分析和画图常用的库示例



数据分析

- 1.Pandas: 数据处理和分析的核心库, 提供高效的数据结构如DataFrame。
- 2.NumPy: 支持多维数组和矩阵运算, 是许多科学计算库的基础。

数据可视化

- 1.Matplotlib: 基础绘图库, 高度可定制。
- 2.Seaborn: 基于Matplotlib, 提供更高级的统计图形接口。
- 3.Plotly: 支持交互式图表, 适合创建动态可视化。
- 4.Bokeh: 适用于现代网页浏览器的交互式可视化工具。

社会计算与统计学

- 1.SciPy: 提供科学计算功能, 包括统计、优化和积分。
- 2.Statsmodels: 用于估计和测试统计模型。
- 3.NetworkX: 用于创建、操作和研究复杂网络。
- 4.Scikit-learn: 机器学习库, 包含多种统计和机器学习算法。

虚拟数据生成

- 1.Faker: 生成各种虚拟数据, 如姓名、地址等。
- 2.Pandas: 结合NumPy, 可以生成模拟数据集。
- 3.Scikit-learn: 提供生成模拟数据集的工具, 如`make_classification`。

诸葛亮舌战群儒 鲁子敬力排众议

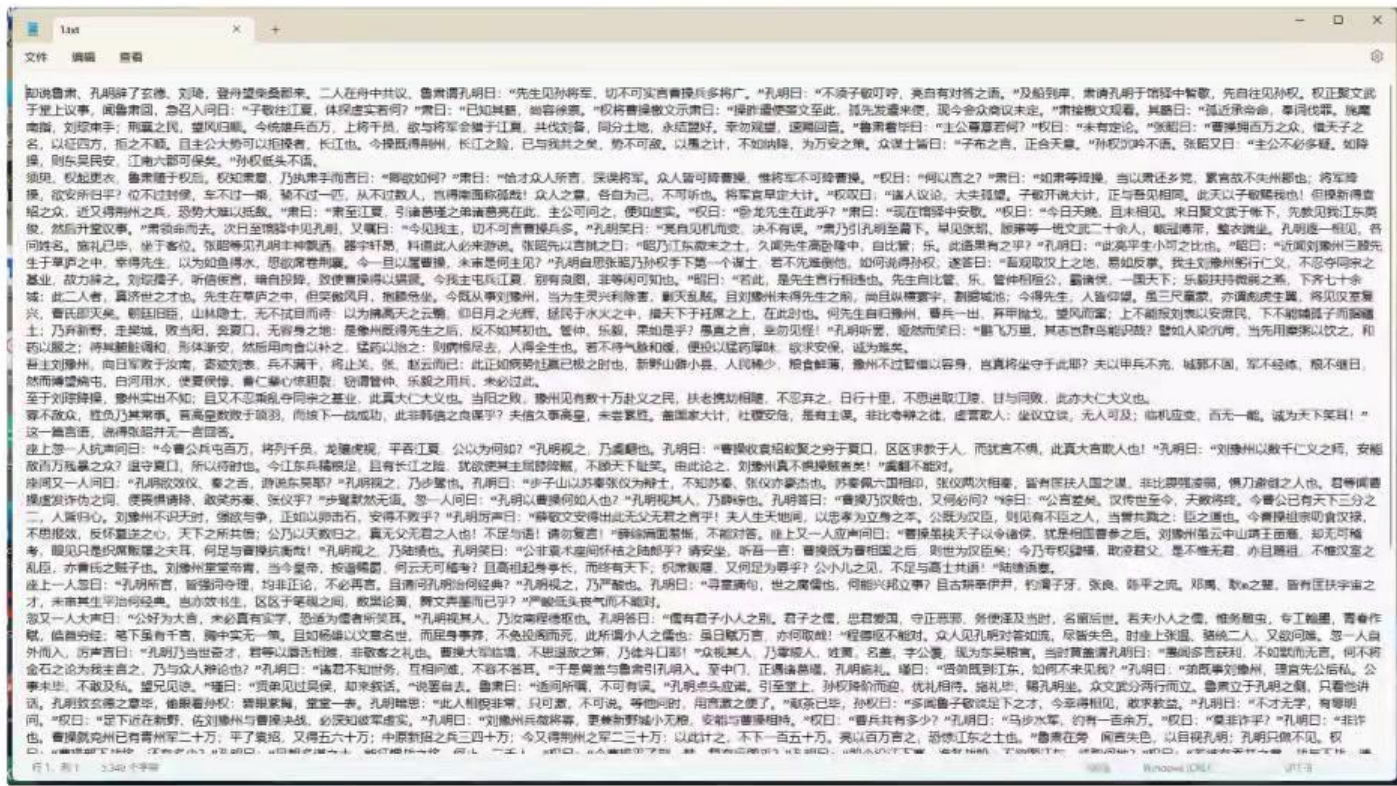


座上一人忽曰：“孔明所言，皆强词夺理，均非正论，不必再言。且请问孔明治何经典？”孔明视之，乃严酸也。孔明曰：“寻章摘句，世之腐儒也，何能兴邦立事？且古耕莘伊尹，钓渭子牙，张良、陈平之流。邓禹、耿m之辈，皆有匡扶宇宙之才，未审其生平治何经典。岂亦效书生，区区于笔砚之间，数黑论黄，舞文弄墨而已乎？”严峻低头丧气而不能对。

忽又一大声曰：“公好为大言，未必真有实学，恐适为儒者所笑耳。”孔明视其人，乃汝南程德枢也。孔明答曰：“儒有君子小人之别。君子之儒，忠君爱国，守正恶邪，务使泽及当时，名留后世。若夫小人之儒，惟务雕虫，专工翰墨，青春作赋，皓首穷经；笔下虽有千言，胸中实无一策。且如杨雄以文章名世，而屈身事莽，不免投阁而死，此所谓小人之儒也；虽日赋万言，亦何取哉！”程德枢不能对。众人见孔明对答如流，尽皆失色。时座上张温、骆统二人，又欲问难。忽一人自外而入，厉声言曰：

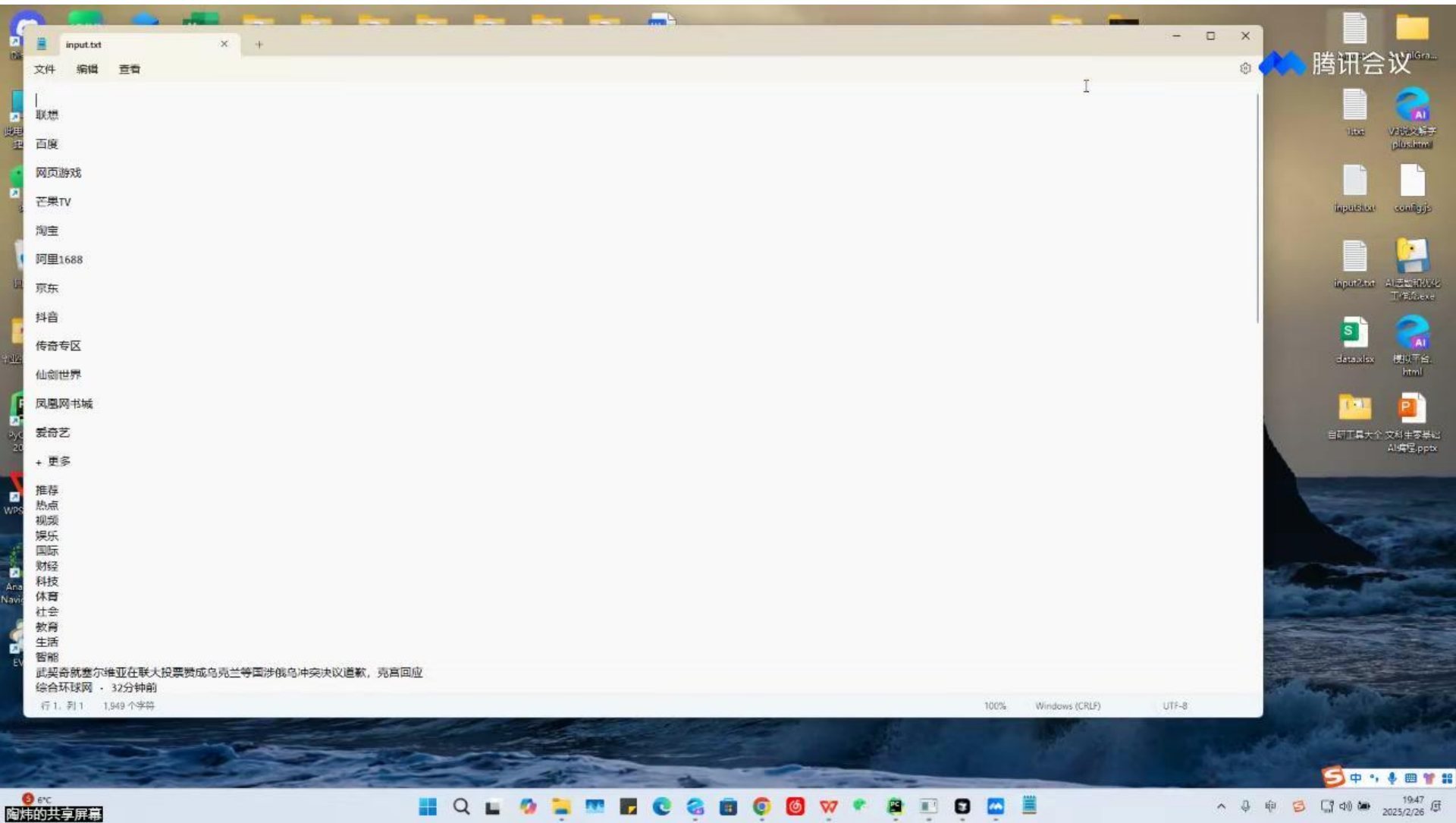
“孔明乃当世奇才，君等以唇舌相难，非敬客之礼也。曹操大军临境，不思退敌之策，乃徒斗口耶！”……

总结诸葛亮的辩论词语：保存文本到本地



如何判断？孔明曰、孔明答曰、孔明笑曰、孔明听罢，哑然而笑曰

"C:\Users\wtao4\OneDrive\桌面\1.txt"



input.txt

文件 编辑 查看

联想

百度

网页游戏

芒果TV

淘宝

阿里1688

京东

抖音

传奇专区

仙剑世界

凤凰网书城

爱奇艺

+ 更多

推荐

热点

视频

娱乐

国际

财经

科技

体育

社会

教育

生活

智能

武契奇就塞尔维亚在联大投票赞成乌克兰等国涉俄乌冲突决议道歉，克塞回应
综合环球网 · 32分钟前

行 1, 列 1 1,949 个字符

100% Windows (CRLF) UTF-8

腾讯会议

1.txt

VS 社区版

input.txt

config.txt

input2.txt

AI 编程

data.xlsx

模拟平台

AI 编程

AI 编程

AI 编程

AI 编程

AI 编程

AI 编程

AI 编程

AI 编程

AI 编程

AI 编程

AI 编程

AI 编程

AI 编程

AI 编程

AI 编程

AI 编程

AI 编程

AI 编程

AI 编程

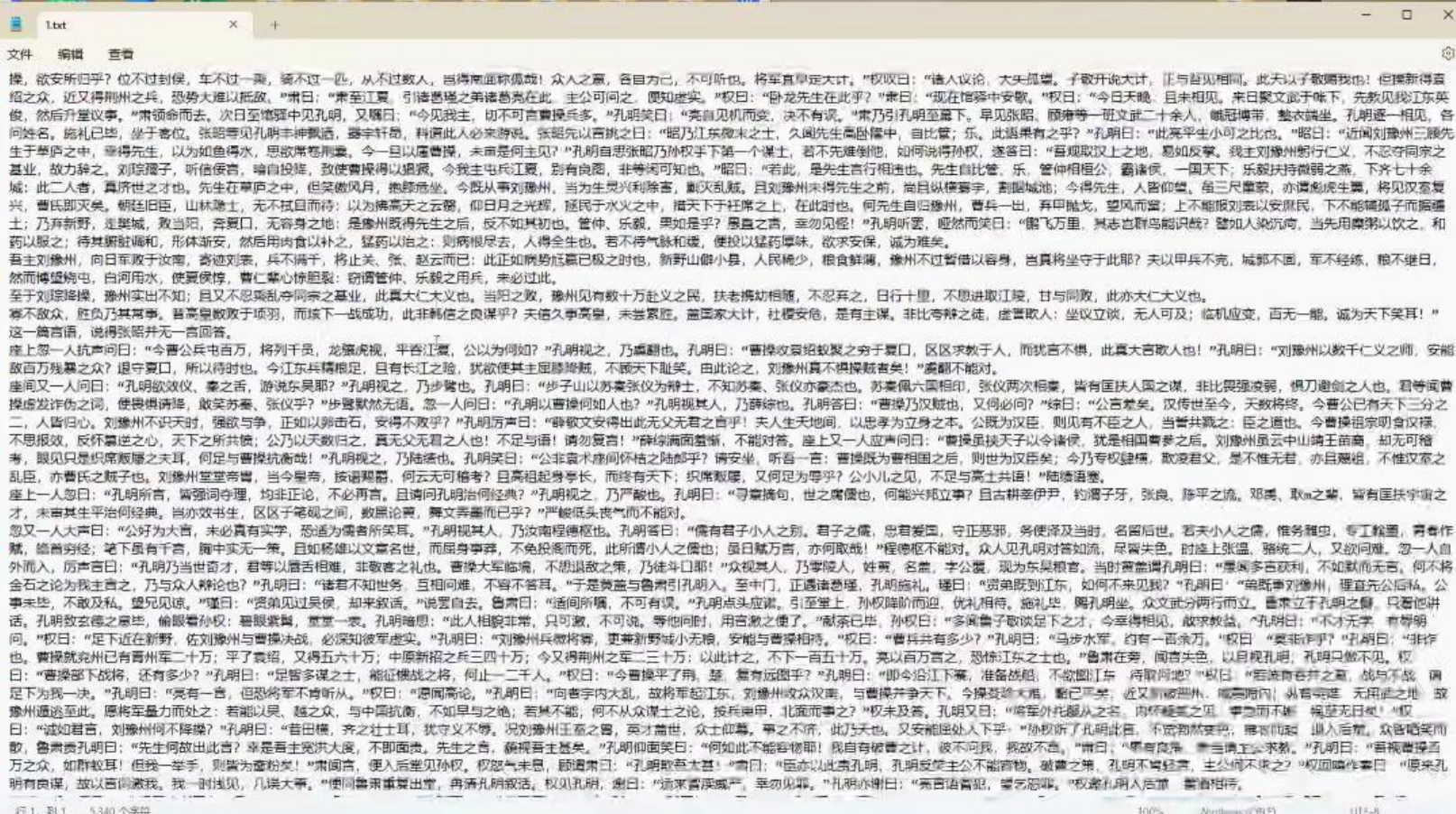
AI 编程

AI 编程

AI 编程

AI 编程

AI 编程



行 1, 到 1 5340 个字符

100% Windows (R) U1-8

却说曹操、孔明辞了玄德、刘琦，登舟望樊城而去。二人在舟中共议，曹操谓孔明曰：“先生见孙将军，切不可实言曹操兵多将广。”孔明曰：“不须子敬叮咛，亮自有对客之语。”及船到岸，请孔明于馆驿中暂歇，先往见孙权。权正聚文武于堂上议事，闻曹操来，急召入问曰：“子敬往江夏，休探虚实否？”孔明曰：“已知其略，尚容徐奏。”权将曹操檄文示曰：“操昨遣使奏文至此，孤先发遣来使，现今众议未定。”曹操檄文观者，其略曰：“孤近承命命，奉词伐罪。旄麾南指，刘琮束手；荆襄之民，望风归顺。今统雄兵百万，上将千员，欲与将军会猎于江夏，共伐刘备，同分土地，永结盟好。幸勿观望，速即回音。”曹操看毕曰：“主公尊意若何？”权曰：“未有定论。”孔明曰：“曹操拥百万之众，挟天子之名，以征四方，拒之不纳。且主公大势必可拒曹操，长江之险，已与我共之矣，势不可敌。以愚之计，不如纳降，为万安之策。众谋士皆曰：‘子布之言，正合天意。’孙权沉吟不语。孔明又曰：‘主公不必多疑。如降操，则东吴民安，江南六郡可保矣。’”孙权低头不语。

须臾，权起更衣，曹操随于后。权知曹操乃执事手而言曰：“卿欲害我乎？”孔明曰：“恰才众人所言，深谋远虑，众人皆可降曹操，惟将军不可降曹操。”权曰：“何以言之？”孔明曰：“如肃等降操，当以肃还乡党，累官故不失州郡也；将军降操，欲安得官乎？位不过封侯，车不过一乘，骑不过一匹，从不过数人，岂得高官显爵哉！众人之策，各自为已，不可听也。将军首倡定大计，正如子敬相问，正与吾见相同。此天以子敬赐我也！但操新得襄阳之众，近又得荆州之兵，恐势难以拒敌。”孔明曰：“肃至江夏，引着鲁肃之弟诸葛亮在此，主公可问之，便知虚实。”权曰：“卧龙先生在此乎？”孔明曰：“现在馆驿中安歇。”权曰：“今日天晚，且未相见。来日文武于帐下，先教我见江东英杰，然后升堂议事。”肃领命而去。次日至馆驿中见孔明，又谓曰：“今见我主，切不可言曹操兵多。”孔明笑曰：“亮自见机而变，决无有误。”肃乃引孔明至幕下。早见张昭、顾雍等一班文武二三十人，峨冠博带，整衣端坐。孔明逐一相见，各问姓名。席礼已毕，坐于客位。张昭等见孔明丰采飘洒，器宇轩昂，料此此人必来游说，张昭先以百挑之曰：“昭乃江东微末之士，久闻先生高卧隆中，自比管、乐；此语果有之乎？”孔明曰：“此真平生小可之比也。”昭曰：“近闻刘豫州三顾先生于草庐之中，幸得先生，以为如鱼得水，思欲席卷荆襄。今一旦以属曹操，未审是何主意？”孔明自思张昭乃孙权手下第一个谋士，若不先难倒他，如何说得孙权，遂答曰：“吾观取汉上之地，易如反掌。我主刘豫州躬行仁义，不忍夺同宗之基业，故力辞之；刘琮孺子，听信佞言，暗自投降，致使曹操得以猖獗。今我主屯兵江夏，别有良图，非常等可知也。”昭曰：“若此，是先生言行相违也。先生自比管、乐，管仲相桓公，霸诸侯，一匡天下；乐毅扶持微弱之燕，下齐七十余城；此二人者，盖济世之才也。先生在草庐之中，但笑傲风月，抱膝危坐。今既从事刘豫州，当为生灵兴利除害，削平乱贼，岂得先立名号，自比管、乐？且曹操未得先生之前，尚且纵横寰宇，割据城池；今得先生，人皆仰望。虽三尺童蒙，亦须敬惧生翼，将见汉室复兴，曹氏即灭矣。朝廷旧臣，山林隐士，无不拭目而待：以为冀高天之恩，仰日月之光辉，庶民于水火之中，措天下于衽席之上，在此时也。何先生自归豫州，曹兵一出，弃甲抛戈，望风而遁；上不能报刘表以安民心，下不能辅刘琦而拒曹操；乃竟新野，走樊城，败当阳，奔夏口，无有立锥之地：是豫州既得之后，反不如其初也。管仲、乐毅，果如是乎？愚直之言，幸勿见怪！”孔明听罢，慨然而笑曰：“鹏飞万里，其志岂群鸟能识哉？譬如人染沉痾，当先用麻药以攻之，和药以养之；待其脓血渐和，形体渐安，然后用药以补之，猛药根尽去，得令全生也。吾得不死，而得使孙权降操，欲求安，诚为难矣。我主刘豫州，向日军败于汝南，寄迹刘表，兵不满千，将止矣。此正如病势危急已极之时也，新野山僻小县，人民稀少，粮食鲜薄，豫州不过暂借以容身，岂真将坐守于此耶？夫以甲兵不完，城郭不固，军不经练，粮不继日，至于刘琮降操，豫州出而不入；且又不忍累乱同宗之基业，此真大仁大义也。当阳之败，豫州见有数十万赴义之民，扶老携幼相随，不忍弃之，日行十里，不思进取江陵，甘与同败，此亦大仁大义也。寡人不敏，窃见凡此其事，皆高皇败于项羽，而项下一战成功，此非韩信之良谋乎？夫信久事高皇，未尝累胜，盖国家大计，社稷安危，是有主谋。非比夸辩之徒，虚誉欺人；坐议立谈，无人可及；临机应变，百无一能，诚为天下笑耳！”

这一篇言语，说得张昭无言以答。

座上忽一人抗声问曰：“今曹公兵书百万，将列千员，龙威虎视，平吞江夏，公以为何如？”孔明视之，乃鲁肃也。孔明曰：“曹操收袁绍败军之资于冀州，区区来救于我，而犹言不惧，此真大言欺人也！”孔明曰：“刘豫州以救于仁义之师，安能敌百万残暴之敌？退守夏口，所以待时也。今江东英雄精捷，且有长江之险，犹欲使其主屈膝称臣，不顾天下耻笑。由此论之，刘豫州真不愧降贼者矣！”鲁肃不能对。

座间又一人曰：“孔明欲效、秦之志，游说东吴耶？”孔明视之，乃步騭也。孔明曰：“步騭山以少将军张仪为辩士，不知秦策。张仪亦豪杰也，亦秦佩六国相印，张仪两次相秦，皆有匡扶天下之志，非比畏强凌弱，惧刀锥之人也。君等闻曹操遣发使臣之词，便使张仪降，欲效苏秦、张仪之乎？”步騭默然无语。忽一人问曰：“孔明以曹操何人如人也？”孔明视其人，乃薛洪也。孔明答曰：“曹操乃汉贼也，又何必问？”徐曰：“公言差矣。汉传至今，天数将终。今曹公已有天下三分之二，人皆归心。刘豫州不识天时，强欲与争，正如以卵击石，安得不败乎？”孔明厉声曰：“曹操文安得出此无父无君之言乎！人生天地间，以忠孝为立身之本。公既为汉臣，则见有不臣之人，当誓共戮之；臣之道也。今曹操招家卖国，不思报效，反怀篡逆之心，天下之所共愤；公乃以天数归之，真无父无君之人也！不足与语！请勿复言！”薛洪满面羞惭，不能对答。座上又一人应声问曰：“曹操虽挟天子以令诸侯，就是相国曹操之后，刘豫州虽云中山靖王苗裔，却无可稽考，眼见只是汉献帝之赘耳，何足与曹操抗衡哉？”孔明视之，乃陆绩也。孔明笑曰：“公生草木之间，怀桔之陆郎乎？请安坐，听吾一言：曹操既为曹相国之后，则世为汉臣矣；今乃专权肆横，欺败君父，是不惟无君，亦且蔑祖，不惟汉室之乱臣，亦曹氏之篡子也。刘豫州堂堂帝胄，当今皇帝，按谱昭穆，何云无可稽考？且高祖起事争长，而终有天下；织席贩履，又何足为辱乎？公小儿之见，不足与高士论乎！”陆绩无语。

座上一人忽曰：“孔明所言，皆强词夺理，均非正论，不必再言，且请问孔明何得经典？”孔明视之，乃严酸也。孔明曰：“寻章摘句，世之腐儒也，何能兴邦立事？且古耕莘伊尹，钓渭子牙，张良、陈平之流，邓禹、耿弇之輩，皆有匡扶宇宙之才，未尝生平日治经典，岂亦效书生，区区于笔砚之间，数墨论衡，舞文弄墨而已乎？”严酸低头丧气而不能对。

忽又一大声曰：“公好为大言，未必真有实学，恐适为庸人所笑耳。”孔明视之，乃庞统也。孔明笑曰：“儒有君子小人之别。君子之德，忠君爱国，守正恶邪，务使泽及当时，名留后世。若夫小人之德，惟务雕虫，专工翰墨，青春作赋，皓首穷经；笔下虽有千言，胸中实无一策。且如杨雄以文章名世，而屋舍倾颓，不免投阁而死，此所谓小人之德也；虽日赋万言，亦何取哉！众人见孔明对答如流，尽皆失色。时座上张温、陆统二人，又欲问难。忽一人自外而入，厉声言曰：“孔明乃当世奇才，君等以庸俗之道，侮骂圣人，不思敬重。曹操大军临境，不思退避之策，乃徒斗口耶！”众视其人，乃李严人，姓黄，名盖，字公覆。现为东吴粮官。当时黄盖谓孔明曰：“愚闻多言获利，不如默而无言。何不将金石之论为我主言之，乃与众人辩论也？”孔明曰：“读君不知世务，互相问难，不容不答耳。”于是黄盖与鲁肃引孔明入。至中门，正遇诸葛亮，孔明施礼。统曰：“贤弟既到江东，如何不见我？”孔明曰：“弟既事刘豫州，理宜先公后私。公事未毕，不敢及私。望兄见谅。”统曰：“贤弟见过吴侯，却来叙话。”说罢自去。鲁肃曰：“适间所嘱，不可有误。”孔明点头应诺。引至堂上，孙权降阶而迎，优礼相待。施礼毕，赐孔明坐。众文武分两行而立。鲁肃立于孔明之侧，只看他讲话。孔明叙玄德之尊，偷眼看曹操：碧眼紫髯，堂堂一表。孔明暗想：“此人相貌非常，只可敬，不可说。等他问时，用古教之便了。”献茶已毕，孙权曰：“多闻鲁子敬欲足下之才，今幸得相见，故教求教。”孔明曰：“不才无学，有何讲问。”权曰：“足下近在新野，佐刘豫州与曹操决战，必深知彼虚实。”孔明曰：“刘豫州兵微将寡，更兼新野城小无粮，安能与曹操相拒。”权曰：“曹兵共有多少？”孔明曰：“马步水军，约有一百余万。”权曰：“莫非诈乎？”孔明曰：“非诈也。曹操就兖州已有青州兵二十万；平了袁绍，又得五六十万；中原新招之兵三四十万；今又得荆州之兵二十万；以共计之，不下一百五十万。亮以百万言之，恐惊江东之士也。”鲁肃在旁，闻言失色，以目视孔明；孔明只做不见。权曰：“曹操就荆州已有青州兵二十万；平了袁绍，又得五六十万；中原新招之兵三四十万；今又得荆州之兵二十万；以共计之，不下一百五十万。亮以百万言之，恐惊江东之士也。”鲁肃在旁，闻言失色，以目视孔明；孔明只做不见。权曰：“曹操就荆州已有青州兵二十万；平了袁绍，又得五六十万；中原新招之兵三四十万；今又得荆州之兵二十万；以共计之，不下一百五十万。亮以百万言之，恐惊江东之士也。”鲁肃在旁，闻言失色，以目视孔明；孔明只做不见。

腾讯会议

input.txt

config

input.txt

AI智能办公效率工具

data.xlsx

模拟平台.html

自研工具大全

文科生零基础AI教程.pptx

1947
2025/2/26

准备问题-DeepSeek开始深度思考

DeepSeek-R1 (Pro)

我现在本地有一个txt文本，utf-8编码，5800多个汉字。是诸葛亮舌战群儒，我想抽取这个文本里面诸葛亮说过的全部的话，然后总结其前20个高频关键词，然后可视化出来。给我python代码

深度思考 ^

好的，我现在需要帮用户解决这个问题。用户有一个txt文件，里面是诸葛亮舌战群儒的内容，大概5800多个汉字，编码是UTF-8。用户想要提取诸葛亮说过的所有话，然后统计前20个高频关键词，并进行可视化。我需要用Python代码来实现这些步骤。

首先，我得想如何从文本中识别诸葛亮的发言。通常，这类文本可能通过引导词或者说话人标注来区分对话。比如可能会有“诸葛亮曰：.....”或者引导词内的内容。但用户没有说明具体的文本结构，可能需要假设。如果是古典文学，比如《三国演义》中的段落，可能会用“孔明曰”或者“诸葛亮曰”来引出对话，后面接着说话内容。因此，可能需要用正则表达式来匹配以这些开头的段落，然后提取后面的内容。

接下来，步骤大概分为：

- 1.读取文本文件。
- 2.使用正则表达式找到所有诸葛亮说的话。
- 3.将这些内容合并成一个字符串，用于分词处理。
- 4.使用jieba分词进行分词，并统计词频。
- 5.过滤掉停用词和无意义的词汇（如标点、虚词等）。
- 6.提取前20个高频词。
- 7.使用matplotlib或者pyecharts进行可视化。

现在需要考虑的问题有几个：

- 如何正确匹配诸葛亮的发言？比如，文本中可能有其他角色的话，或者旁白。需要确定正确的正则表达式模式。比如，可能使用`r'诸葛亮(曰|说): (.*)'`这样的模式，但需要考虑不同情况下的引号使用，比如是否全角，是否有换行等。
- 如果文本中使用的是全角引号，那正则表达式需要正确匹配。另外，如果诸葛亮的发言跨越多行，可能需要调整正则表达式的匹配方式，比如使用`re.DOTALL`标志。
- 停用词的处理。需要加载中文停用词表，比如常用的中文停用词文件，比如从github上下载的哈工大停用词表等。如果没有停用词表，可能在代码中手动添加一些常见的虚词，或者标点符号。
- 分词后的结果可能包含单字词，比如“之”、“其”等。这些可能需要过滤掉，或者通过调整jieba的分词模式（比如使用精确模式或全模式）。或者，可以在统计词频后，排除长度小于2的词。
- 可视化部分，用户可能希望看到柱状图或者词云。用户这里提到的是可视化出来，可能柱状图比较合适，因为要展示前20个高频词。或者词云可能更直观。但用户示例中可能给出柱状图的代码。

bash

```
pip install jieba matplotlib
```

python

● 添加自定义词典

```
jieba.load_userdict("custom_dict.txt")
```

```
pip install jieba matplotlib
```

- 将文本文件放在与代码相同目录，命名为zhugeliang.txt
- 如需专业停用词表，推荐使用[哈工大停用词表](#)，命名为stopwords.txt放在同级目录

python

● 根据实际文本特征修改匹配规则，例如：

```
pattern = re.compile(r"省括号[:;]??(?:'|\"|,|!|!|:|$)", re.DOTALL)
```

复制代码

python

* 可替换为以下颜色方案:

```
colors = ['#'+''.join(random.sample('789ABCD',6)) for _ in range(20)]
```

```
plt.barh(keywords, counts, color=colors)
```

复制代码

2.如果分词效果不理想,可以:

python

```
jieba.load_userdict("custom_dict.txt")
```

```
pip install jieba matplotlib
```

- 将文本文件放在与代码相同目录，命名为zhugeliang.txt
- 如需专业停用词表，推荐使用[哈工大停用词表](#)，命名为stopwords.txt放在同级目录

```
python
# 根据实际文本特征添加匹配规则，例如：
pattern = re.compile(r"源亮亮[: :]*?([:~])?(:~|~[:~])", re.DOTALL)
```

```
python

# 可替换为以下任意方式:
colors = ['*'*_.join(random.sample('789ABCD',4)) for _ in range(20)]
plt.tern(keywords, counts, color=colors)
```

DeepSeek生成的代码

```
# -*- coding: utf-8 -*-
import re
import jieba
from collections import Counter
import matplotlib.pyplot as plt
import matplotlib
from wordcloud import WordCloud
```

```
# 1. 读取文本文件 (注意Windows路径转义)
file_path = r"C:\Users\wtiao4\OneDrive\桌面\1.txt"
with open(file_path, 'r', encoding='utf-8') as f:
    text = f.read()
```

```
# 2. 优化正则表达式匹配所有说话方式
pattern = re.compile(
    r"孔明(?:所云, 哑然而笑|答|笑)?曰[: '"]?(.*)?(?:。|!|!|?|)$)",
    re.DOTALL
)
dialogues = pattern.findall(text)
zhuge_text = "".join(dialogues)
```

```
# 3. 中文分词处理
jieba.setLogLevel(jieba.logging.INFO)
words = jieba.lcut(zhugue_text)
```

```
# 4. 停用词配置 (增强版)
stopwords = {
    '的', '了', '和', '是', '在', '之', '这', '我', '你', '他', '就', '也', '又',
    '而', '要', '不', '人', '有', '日', '说', '道', '，', '！', '？', '，',
    '“', '”', '（', '）', '：', '；', '乃', '乎', '矣', '耳', '耶', '哉', '夫', '宣',
    '皆', '非', '无', '何', '以', '为', '于', '者', '所', '其', '此', '自', '吾'
}
```

```
# 5. 数据清洗 (保留长度>2的词语)
filtered_words = [word for word in words if len(word) > 2 and word not in stopwords]
```

```
# 6. 词频统计 (带频次过滤)
word_counts = Counter(filtered_words)
min_freq = 2 # 最小出现次数
top20_words = [item for item in word_counts.most_common(20) if item[1] >= min_freq]
```

```
# 7. 可视化增强配置
plt.style.use('ggplot')
matplotlib.rcParams['font.sans-serif'] = ['SimHei']
matplotlib.rcParams['axes.unicode_minus'] = False
```

```
# 准备数据
keywords = [item[0] for item in top20_words]
counts = [item[1] for item in top20_words]
```

```
# 绘制高级柱状图
plt.figure(figsize=(12, 8))
bars = plt.barh(
    keywords, counts,
    color=plt.cm.tab20c(range(len(keywords))),
    edgecolor='black'
)
```

```
# 装饰图表
plt.xlabel("出现次数", fontsize=14, labelpad=10)
plt.ylabel("关键词", fontsize=14, labelpad=10)
plt.title("诸葛亮发言高频词TOP20分析", fontsize=18, pad=20, fontweight='bold')
plt.gca().invert_yaxis()
```

```
# 添加数据标签
for bar in bars:
    width = bar.get_width()
    plt.text(
        width + 0.3,
        bar.get_y() + bar.get_height()/2,
        f'{width}',
        va='center',
        fontsize=10
    )
```

```
# 添加网格线
plt.grid(axis='x', alpha=0.3, linestyle='--')
```

```
plt.tight_layout()
plt.show()
```

```
# 词云图绘制
wc = WordCloud(font_path='simhei.ttf', background_color='white')
wc.generate(" ".join(filtered_words))
plt.imshow(wc)
plt.axis('off')
```

plt.show()



Term	Frequency
曹福	6
刘豫州	4
先生	2
安眠	2
百万	2
之众	2
苏美	2
张仪	2
江东	2
将军	2
万人	2

二

文生数据

社会数据和AI生成数据：来源、特性和应用



传统社会数据主要来源于实际社会活动和交互，如调查问卷、人口普查和市场调研，具有较高的真实性和可靠性，但往往存在时效性滞后的问题。这些数据通常用于社会现象的分析和预测、政策制定以及学术研究。

传统社会数据的采集过程中容易受到人为因素的影响，需要严格的质量控制措施，同时还需要关注隐私保护问题。

相比之下，AI生成数据通过人工智能算法模拟和生成，具有较高的灵活性和可控性，能够根据需求生成不同类型和规模的数据。AI生成数据广泛应用于模型训练、虚拟环境模拟和创意内容生成，特别在文化创意产业中具有独特的优势。

AI生成数据的质量则依赖于训练数据集，可能存在偏差和误导，且在真实性和伦理方面需要特别关注。

AI生成数据的有效性

- AI生成数据的有效性主要体现在其数据质量、应用场景、比较分析和伦理法律合规性等方面。
- 高质量的AI生成数据应具备准确性、一致性和无偏性，能够准确反映现实情况，并适用于特定应用场景。通过对比分析和实验验证，可以评估生成数据的可靠性和实际效果。
- 生成数据的使用必须符合道德伦理和法律法规，确保不侵犯个人隐私且不用于恶意用途。综合考虑这些因素，AI生成数据才能在实际应用中被视为有效并发挥最大价值。

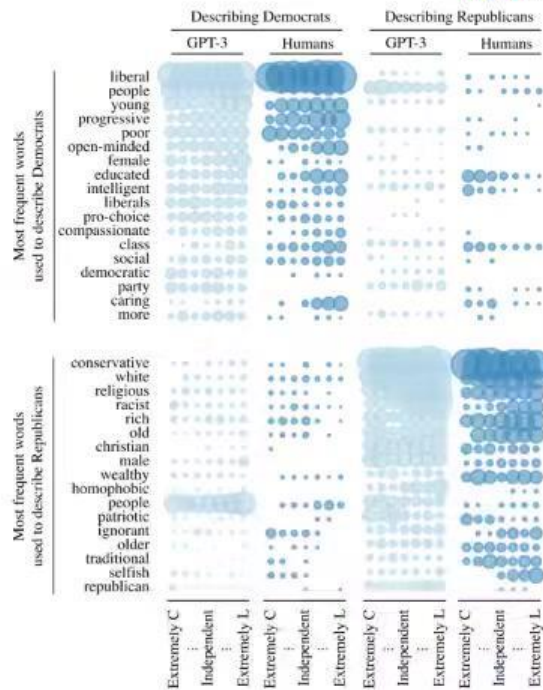


Figure 2. The original Pigeonholing Partisans dataset and the corresponding GPT-3-generated words. Bubble size represents relative frequency of word occurrence; columns represent the ideology of list writers. GPT-3 uses a similar set of words to humans.

研究表明GPT-4不仅在内容和形式上能够与人类生成的数据相匹配，还能反映出复杂的社会和政治态度间的关系模式，从而为社会科学研究提供了一种新颖且强大的方法。
Argyle, L. P., Busby, E. C., Fulda, N., Gubler, J. R., Rytting, C., & Wingate, D. (2023). Out of one, many: Using language models to simulate human samples. *Political Analysis*, 31*(3), 337-351. <https://doi.org/10.1017/pan.2023.2>

AI幻觉及其基本特点

AI幻觉 (AI Hallucinations) 是指生成式人工智能模型在生成文本或回答问题时，尽管表面上呈现出逻辑性和语法正确的形式，但其输出内容可能包含完全虚构、不准确或与事实不符的信息。



幻觉类型	数据可用性	理解能力深度	语境精确度	外部信息整合能力	逻辑推理和抽象能力	典型错误表现
数据误用型幻觉	有数据	低	高	高	中	误用已有数据，回答部分不符或细节错误
语境误解型幻觉	有数据	高	低	高	中	对问题的意图理解错误，回答偏离主题
信息缺失型幻觉	无数据	中	高	低	中	未能正确获取或整合外部信息
推理错误型幻觉	部分数据	高	高	中	低	逻辑推理中存在漏洞或错误假设
创作型幻觉	无数据	低	中	低	低	在无数据支持下，生成完全虚构的信息

AI幻觉问题抽取：多数据集 问题加载

```
from datasets import load_dataset
import pandas as pd

# 加载SQuAD数据集
squad_dataset = load_dataset("squad")

# 从SQuAD数据集中提取问题并去重
questions = list(set(squad_dataset['train']['question'] + squad_dataset['validation']['question']))

# 如果SQuAD数据集不足10万个问题，再加载其他数据集（例如Natural Questions）
if len(questions) < 100000:
    natural_questions_dataset = load_dataset("natural_questions", split='train')

    # 从Natural Questions数据集中提取问题并去重
    additional_questions = list(set(natural_questions_dataset['question']))

    # 将新的问题加入列表中并去重
    questions = list(set(questions + additional_questions))

# 确保生成的问题数达到10万个
if len(questions) < 100000:
    raise ValueError(f"数据集中问题数量不足10万个，仅有{len(questions)}个问题。")

# 只保留前10万个问题
questions = questions[:100000]

# 保存问题到CSV文件
df = pd.DataFrame(questions, columns=["问题"])
df.to_csv('生成的真实答案问题.csv', index=False, encoding='utf-8-sig')

print(f"共生成{len(questions)}个问题并保存到生成的真实答案问题.csv")
```



要生成10万个存在真实答案的问题，并且基于2020年之前的数据，可以使用现有的**公开问答数据集**（如SQuAD、Natural Questions等）来生成问题。可以从多个数据集中组合问题，以达到10万个的问题数量。

这些数据集包含大量的问答对，例如使用datasets库（Hugging Face的datasets库）来加载SQuAD数据集（Stanford Question Answering Dataset），这个数据集是一个著名的问答数据集，基于维基百科数据生成，并且数据是2020年之前的。



加载数据集：使用datasets库加载SQuAD数据集，这个数据集包含了大量基于2020年之前数据生成的问答对。

提取问题：从数据集中提取问题，并使用set去重。

检查问题数量：确保提取的问题数量至少为10万个。

保存问题：将问题保存到CSV文件生成的真实答案问题.csv中。

AI幻觉问题抽取：多数据集 问题加载

```
from datasets import load_dataset
import pandas as pd

# 加载SQuAD数据集
squad_dataset = load_dataset("squad")

# 从SQuAD数据集中提取问题并去重
questions = list(set(squad_dataset['train']['question'] + squad_dataset['validation']['question']))

# 如果SQuAD数据集不足10万个问题，再加载其他数据集（例如Natural Questions）
if len(questions) < 100000:
    natural_questions_dataset = load_dataset("natural_questions", split='train')

    # 从Natural Questions数据集中提取问题并去重
    additional_questions = list(set(natural_questions_dataset['question']))

    # 将新的问题加入列表中并去重
    questions = list(set(questions + additional_questions))

# 确保生成的问题数达到10万个
if len(questions) < 100000:
    raise ValueError("数据集中心问题数量不足10万个，仅有{len(questions)}个问题。")

# 只保留前10万个问题
questions = questions[:100000]

# 将问题保存到CSV文件
df = pd.DataFrame(questions, columns=["问题"])
df.to_csv("生成的真实答案问题.csv", index=False, encoding='utf-8-sig')

print("已将10万个问题保存到生成的真实答案问题.csv")
```

要生成10万个存在真实答案的问题，并且基于2020年之前的数据，可以使用现有的**公开问答数据集**（如SQuAD、Natural Questions等）来生成问题。可以从多个数据集中组合问题，以达到10万个的问题数量。

这些数据集包含大量的问答对，例如使用datasets库（Hugging Face的datasets库）来加载SQuAD数据集（Stanford Question Answering Dataset），这个数据集是一个著名的问答数据集，基于维基百科数据生成，并且数据是2020年之前的。

加载数据集：使用datasets库加载SQuAD数据集，这个数据集包含了大量基于2020年之前数据生成的问答对。

提取问题：从数据集中提取问题，并使用set去重。

检查问题数量：确保提取的问题数量至少为10万个。

保存问题：将问题保存到CSV文件生成的真实答案问题.csv中。

```
f.write(f'主题 (i): {', '.join(topic)}\n')
```

示例数据, 实际应根据文本数据提取时间序列特征

```
微博 ts = 微博 df['text'].str.len() # 简单用文本长度代替时间序列特征
```

大模型 ts = 大模型 df['text'].str.len()

```
# 创建时间序列DataFrame
```

```
ts_df = pd.DataFrame({'微博': 微博_ts, '大模型': 大模型_ts})
```

清理数据，去除空白行和无效值

```
ts_df.dropna(inplace=True)
```

```
ts_df = ts_df[(ts_df['微博'] != float('inf')) & (ts_df['大模型'] != float('inf'))]
```

ARIMA 分析

```
arima model 微博 = ARIMA(ts df["微博"], order=(1, 1, 1)).fit()
```

```
arima model 大模型 = ARIMA(ts df[大模型], order=(1, 1, 1)).fit()
```

```
# 输出ARIMA结果到本地文件
```

with open('arima_results.txt', 'w', encoding='utf-8') as f:

```
f.write("ARIMA 微博 结果:\n")
```

```
f.write(str(arima_model.summary()))
```

```
f.write("\n\nARIMA 大模型 结果:\n")
```

```
f.write(str(arima_model_大模型.summary()))
```

绘制时间序列图

给 DeepSeek 发送消息

深度思考 (R1)

 联网搜索

内容由 AI 生成，请仔细甄别

6°C
陶伟的共享屏幕

内容由 AI 生成，请仔细甄别

deepseek

开启新对话

今天

Python分析《三国演义》诸葛亮发言

昨天

生成式AI内容生产使用研究

唐蜜杨花枝铜镜千年风华

7天内

Python数据分析与可视化常用库

30天内

DeepSeek本地部署版本及配置要

人机优生与快生AI实践风格

CoT链式思维核心特点解析

进一步探讨模型在模拟人类社会行

这种差异可能影响知识的准确性与

人工智能赋能智库研究，给我从4个

人机共生理论下生成式AI的知识生

研究目标一：效果测评——评估生

下载 App

个人信息

Python分析《三国演义》诸葛亮发言

```
scaler = StandardScaler()
X_scaled = pd.DataFrame(scaler.fit_transform(X), columns=X.columns)

# 5. 共线性自动处理
def auto_remove_collinearity(df, threshold=10):
    """ 自动移除高VIF变量 """
    variables = df.columns.tolist()
    dropped = True
    while dropped:
        dropped = False
        vif = []
        for i in range(df.shape[1]):
            try:
                vif.append(variance_inflation_factor(df.values, i))
            except:
                vif.append(0)

        max_vif = max(vif)
        if max_vif > threshold:
            max_index = vif.index(max_vif)
            print(f"移除变量 {variables[max_index]} (VIF={max_vif:.1f})")
            df = df.drop(columns=[variables[max_index]])
            variables.pop(max_index)
            dropped = True
        return df
```

给 DeepSeek 发送消息

深度思考 (R1) 联网搜索

DeepSeek - 探索未来之境

SiliconCloud

DeepSeek - 探索未来之境

登录页面_问卷星

SiliconCloud

小木虫论坛-学术科...

可灵 AI - 新一代 AI...

资源访问控制系统...

智灵动力

元美AI设计平台

虚拟人生产力平台...

腾讯会议

内容山 AI 生成，请仔细甄别

19:57 2025/2/26

6°C
陶伟的共享屏幕

内容由 AI 生成，请仔细甄别

deepseek

开启新对话

今天

Python分析《三国演义》诸葛亮发言

昨天

生成式AI内容生产使用研究

唐蜜桃花枝铜镜千年风华

7天内

Python数据分析与可视化常用库

30天内

DeepSeek本地部署版本及配置要

人机共生与快生AI实践风格

CoT链式思维核心特点解析

进一步探讨模型在模拟人类社会行

这种差异可能影响知识的准确性与

人工智能赋能智库研究,给我从4个

人机共生理论下生成式AI的知识生

研究目标一:效果测评——评估生

下载 App NEW

个人信息

Python分析《三国演义》诸葛亮发言

```
python
font_path = 'simhei.ttf' # 设置字体文件路径,确保支持中文
background_color = 'white', # 设置背景颜色
width=800 # 设置宽度
height=400 # 设置高度
max_words=200 # 设置最大词数

wc.generate(' '.join(filtered_words)) # 生成词云

# 显示词云
plt.figure(figsize=(10, 5))
plt.imshow(wc, interpolation='bilinear')
plt.axis('off') # 隐藏坐标轴
plt.show()
```

- ### 代码说明
- 文件路径: 将 `file_path` 替换为你的文本文件路径。
 - 正则表达式: 匹配诸葛亮亮的发言内容,支持多种说话方式。
 - 分词与过滤: 使用 `jieba` 进行分词,并过滤停用词和单字词。
 - 词频统计: 统计词频并过滤低频词。
 - 柱状图: 绘制水平柱状图,展示高频词及其出现次数。
 - 词云: 生成词云图,直观展示高频词。

运行环境

- 确保已安装以下库:

```
bash
pip install jieba matplotlib wordcloud
```
- 字体文件 `simhei.ttf` 需要存在于运行环境中,或者替换为其他支持中文字体路径。

DeepSeek 发送消息

深度思考 (R1) 联网搜索

Python分析《三国演义》诸葛亮发言

开启新对话

今天

Python分析《三国演义》诸葛亮发言...

昨天

生成式AI内容生产使用研究

唐蕃两花枝铜镜千年风华

7天内

Python数据分析与可视化常用库

30天内

DeepSeek本地部署版本及配置要...

人机优生与快生AI实践风格

CoT链式思维核心特点解析

进一步探讨模型在模拟人类社会行...

这种差异可能影响知识的准确性与...

人工智能赋能智库研究，给我从4个

人机共生理论下生成式AI的知识生...

研究目标一：效果测评——评估生...

下载 App NEW

个人信息



```
from datasets import load_dataset
import pandas as pd

# 初始化问题集合
questions = set()

# 定义一个函数从数据集中提取问题
def add_questions_from_dataset(dataset_name, config_name=None, split_name='train',
                               question_column='question'):
    global questions
    try:
        # 加载数据集，配置名称是可选的
        if config_name:
            dataset = load_dataset(dataset_name, config_name, split=split_name)
        else:
            dataset = load_dataset(dataset_name, split=split_name)

        # 提取问题并去重
        new_questions = set(dataset[question_column])
        # 将新问题加入集合
        questions.update(new_questions)
        print(f"从数据集 {dataset_name} {(config_name)} 中添加了 {len(new_questions)} 个问题，"
              f"总共 {len(questions)} 个问题。")
    except Exception as e:
```

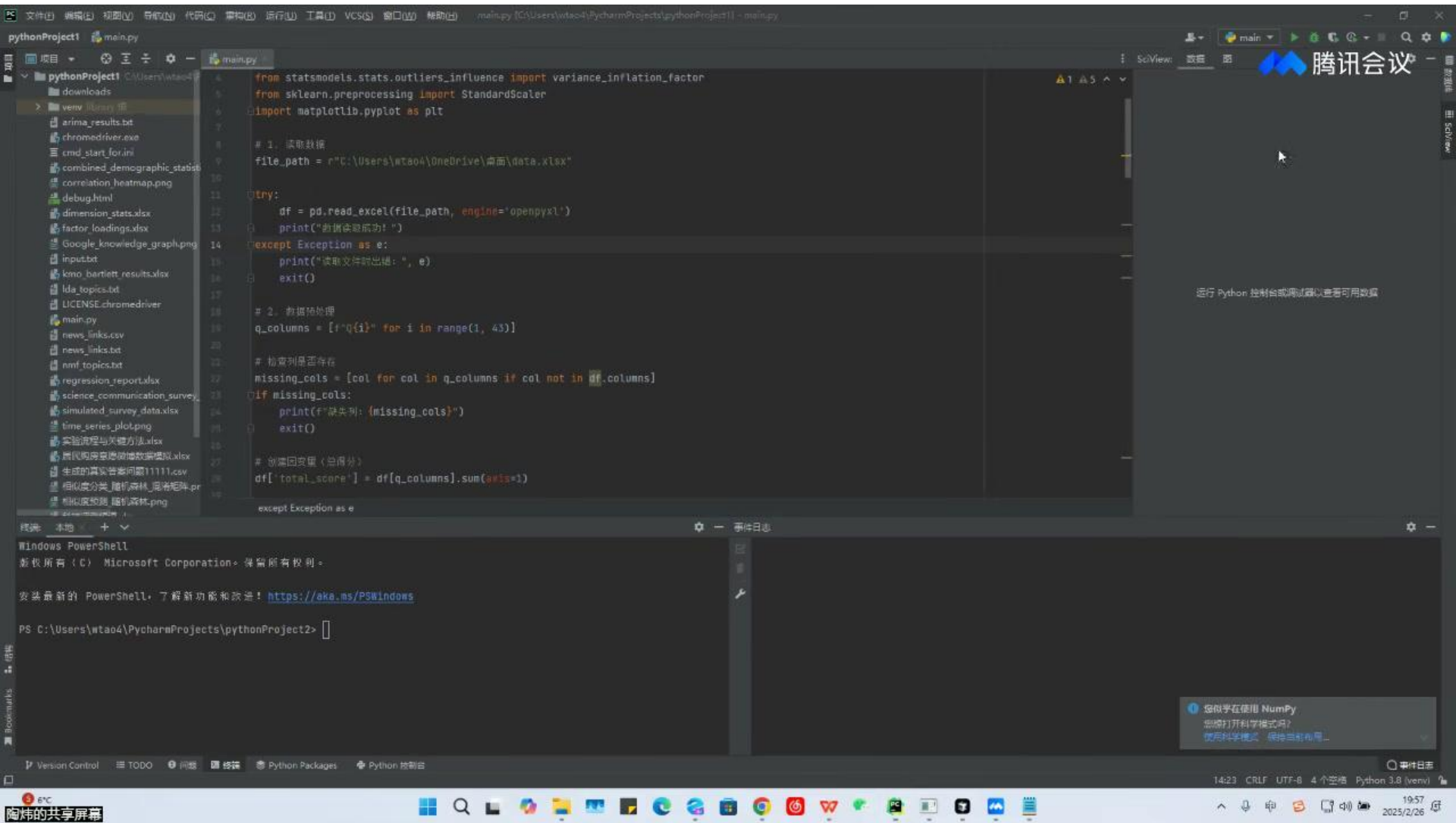
给 DeepSeek 发送消息

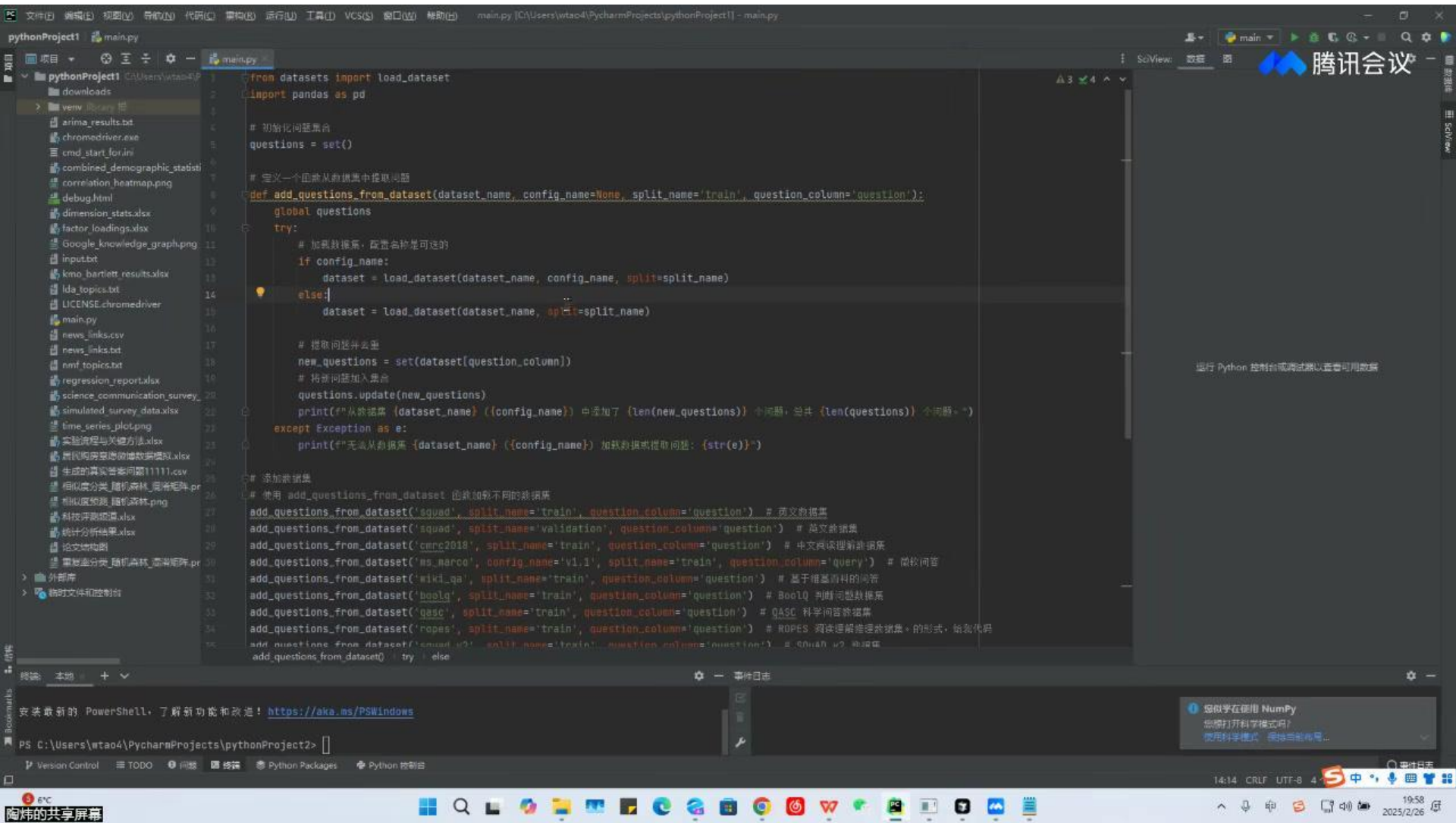
深度思考 (R1) 联网搜索

📎 ↗

内容由 AI 生成，请仔细甄别

内容由 AI 生成，请仔细甄别





给我把以上代码再写一遍

10

20:01
2025/2/26

文件(F) 编辑(E) 视图(V) 导航(N) 代码(C) 窗口(W) 帮助(H) 运行(R) 工具(T) VCS(S) 窗口(O) 辅助(O) main.py [C:\Users\wtao4\PycharmProjects\pythonProject1]\main.py

pythonProject1 main.py

项目

pythonProject1

downloads

venv\library\lib

arima_results.txt

chromedriver.exe

cmd_start_for.ini

combined_demographic_statist

correlation_heatmap.png

debug.html

dimension_stats.xlsx

factor_loadings.xlsx

Google_knowledge_graph.png

input.txt

kino_bartlett_results.xlsx

lda_topics.txt

LICENSE.chromedriver

main.py

news_links.csv

news_links.txt

nmf_topics.txt

regression_report.xlsx

science_communication_survey

simulated_survey_data.xlsx

time_series_plot.png

main.py

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

86

87

88

89

90

91

92

93

94

95

96

97

98

99

100

101

102

103

104

105

106

107

108

109

110

111

112

113

114

115

116

117

118

119

120

121

122

123

124

125

126

127

128

129

130

131

132

133

134

135

136

137

138

139

140

141

142

143

144

145

146

147

148

149

150

151

152

153

154

155

156

157

158

159

160

161

162

163

164

165

166

167

168

169

170

171

172

173

174

175

176

177

178

179

180

181

182

183

184

185

186

187

188

189

190

191

192

193

194

195

196

197

198

199

200

201

202

203

204

205

206

207

208

209

210

211

212

213

214

215

216

217

218

219

220

221

222

223

224

225

226

227

228

229

230

231

232

233

234

235

236

237

238

239

240

241

242

243

244

245

246

247

248

249

250

251

252

253

254

255

256

257

258

259

260

261

262

263

264

265

266

267

268

269

270

271

272

273

274

275

276

277

278

279

280

281

282

283

284

285

286

287

288

289

290

291

292

293

294

295

296

297

298

299

300

301

302

303

304

305

306

307

308

309

310

311

312

313

314

315

316

317

318

319

320

321

322

323

324

325

326

327

328

329

330

331

332

333

334

335

336

337

338

339

340

341

342

343

344

345

346

347

348

349

350

351

352

353

354

355

356

357

358

359

360

361

362

363

364

365

366

367

368

369

370

371

372

373

374

375

376

377

378

379

380

381

382

383

384

385

386

387

388

389

390

391

392

393

394

395

396

397

398

399

400

401

402

403

404

405

406

407

408

409

410

411

412

413

414

415

416

417

418

419

420

421

422

423

424

425

426

427

428

429

430

431

432

433

434

435

436

437

438

439

440

441

442

443

444

445

446

447

448

449

450

451

452

453

454

455

456

457

458

459

460

461

462

463

464

465

466

467

468

469

470

471

472

473

474

475

476

477

478

479

480

481

482

483

484

485

486

487

488

489

490

491

492

493

494

495

496

497

498

499

500

501

502

503

504

505

506

507

508

509

510

511

512

513

514

515

516

517

518

519

520

521

522

523

524

525

526

527

528

529

530

531

532

533

534

535

536

537

538

539

540

541

542

543

544

545

546

547

548

549

550

551

552

553

554

555

556

557

558

559

560

561

562

563

564

565

566

567

568

569

570

571

572

573

574

575

576

577

578

579

580

581

582

583

584

585

586

587

588

589

590

591

592

593

594

595

596

597

598

599

600

601

602

603

604

605

606

607

608

609

610

611

612

613

614

615

616

617

618

619

620

621

622

623

624

625

626

627

628

629

630

631

632

633

634

635

636

637

638

639

640

641

642

643

644

645

646

647

648

649

650

651

652

653

654

655

656

657

658

659

660

661

662

663

664

665

666

667

668

669

670

671

672

673

674

675

676

677

678

679

680

681

682

683

684

685

686

687

688

689

690

691

692

693

694

695

696

697

698

699

700

701

702

703

704

705

706

707

708

709

710

711

712

713

714

715

716

717

718

719

720

721

722

723

724

725

726

727

728

729

730

731

732

733

734

735

736

737

738

739

740

741

742

743

744

745

746

747

748

749

750

751

752

753

754

755

756

757

758

759

760

761

762

763

764

765

766

767

768

769

770

771

772

773

774

775

776

777

778

779

780

781

782

783

784

785

786

787

788

789

790

791

792

793

794

795

796

797

798

799

800

801

802

803

804

805

806

807

808

809

810

811

812

813

814

815

816

817

818

819

820

821

822

823

824

825

826

827

828

829

830

831

832

833

834

835

836

837

838

839

840

841

842

843

844

845

846

847

848

849

850

851

852

853

854

855

856

857

858

859

860

861

862

863

864

865

866

867

868

869

870

871

872

873

874

875

876

877

878

879

880

881

882

883

884

885

886

887

888

889

890

891

892

893

894

895

896

897

898

899

900

901

902

903

904

905

906

907

908

909

910

911

912

913

914

915

916

917

918

919

920

921

922

923

924

925

926

927

928

929

930

931

932

933

934

935

936

937

938

939

940

941

942

943

944

945

946

947

948

949

950

951

952

953

954

955

956

957

958

959

960

961

962

963

964

965

966

967

968

969

970

971

972

973

974

975

976

977

978

979

980

981

982

983

984

985

986

987

988

989

990

991

992

993

994

995

996

997

998

999

1000

运行

main

C:\Users\wtao4\PycharmProjects\pythonProject1\venv\Scripts\python.exe C:\Users\wtao4\PycharmProjects\pythonProject1\main.py

您似乎在使用 NumPy

您想打开科学模式吗?

使用科学模式 保持当前布局...

Version Control

运行

TODO

问题

结构

Python Packages

Python 控制台

36.2

CRLF

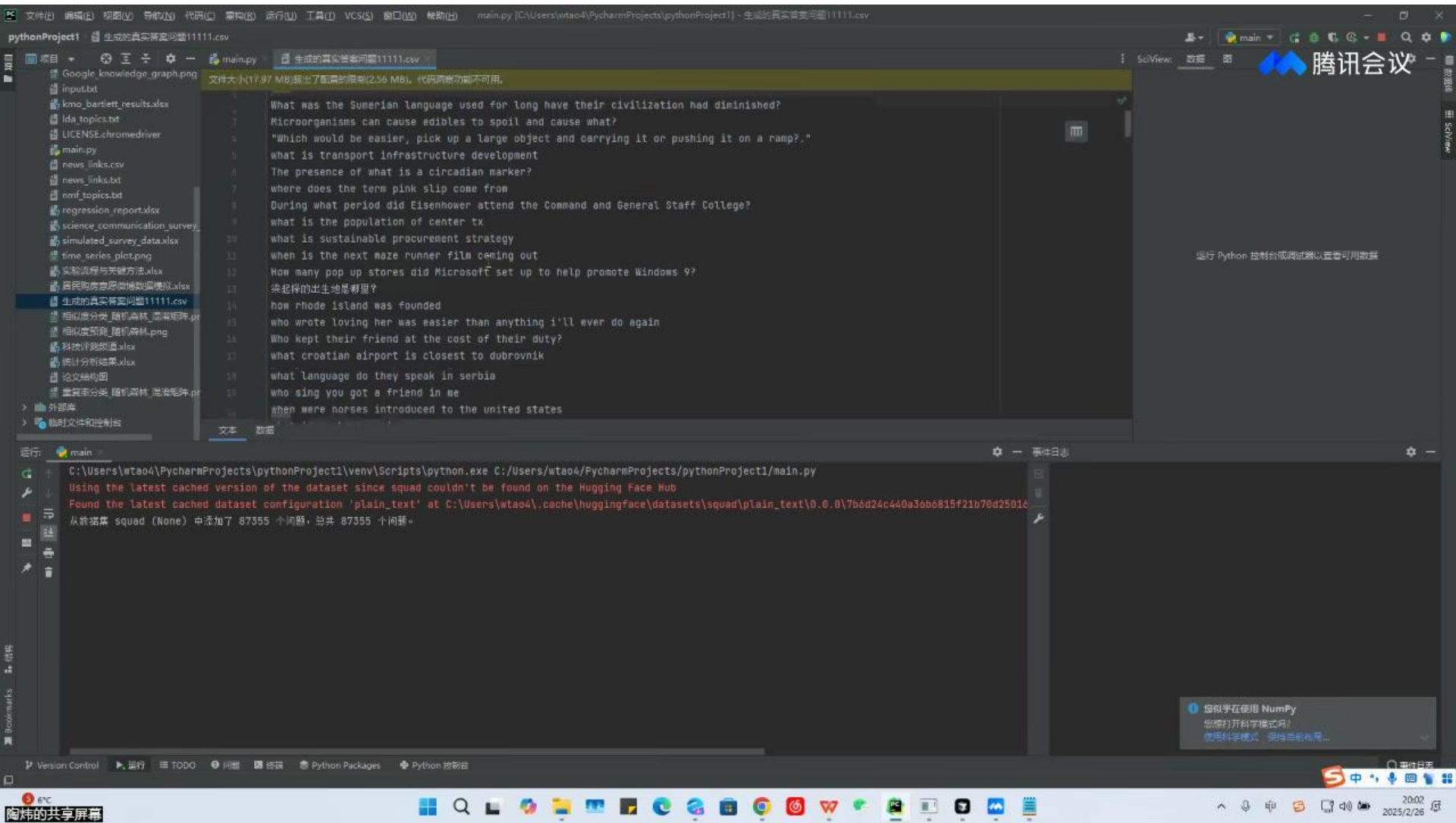
UTF-8

4

中

2001

2025/2/26



文件(F) 视图(V) 代码(C) 运行(R) 工具(S) VCS(S) 窗口(W) 帮助(H) main.py [C:\Users\wtao4\PycharmProjects\pythonProject1] - 生成的真实答案问题11111.csv

pythonProject1 生成的真实答案问题11111.csv

Google_knowledge_graph.png
input.txt
kmo_bartlett_results.xlsx
lda_topics.bt
LICENSE.chromedriver
main.py
news_links.csv
news_links.txt
nmf_topics.bt
regression_report.xlsx
science_communication_survey_...
simulated_survey_data.xlsx
time_series_plot.png
实验流程与关键方法.xlsx
居民购房意愿数据挖掘模拟.xlsx
生成的真实答案问题.csv
生成的真实答案问题11111.csv
相似度分类_随机森林_混淆矩阵.pr
相似度预测_随机森林.png
科技评测数据.xlsx
统计分析报告.xlsx
论文结构图
重聚类分类_随机森林_混淆矩阵.pr

main.py 生成的真实答案问题11111.csv
文件大小(17.97 MB)超出了配置的限制(2.56 MB), 代码浏览功能不可用。

1 问题
2 What was the Sumerian language used for long have their civilization had diminished?
3 Microorganisms can cause edibles to spoil and cause what?
4 "Which would be easier, pick up a large object and carrying it or pushing it on a ramp?."
5 what is transport infrastructure development
6 The presence of what is a circadian marker?
7 where does the term pink slip come from
8 During what period did Eisenhower attend the Command and General Staff College?
9 what is the population of center tx
10 what is sustainable procurement strategy
11 when is the next maze runner film coming out
12 How many pop up stores did Microsoft set up to help promote Windows 9?
13 桑起祥的出生地是哪里?
14 how rhode island was founded
15 who wrote loving her was easier than anything i'll ever do again
16 Who kept their friend at the cost of their duty?
17 what croatian airport is closest to dubrovnik
18 How large can ctenophora grow?
19 what language do they speak in serbia
20 who sinn you not a friend in me

运行 Python 控制台或测试器以查看可用数据

运行: main

C:\Users\wtao4\PycharmProjects\pythonProject1\venv\Scripts\python.exe C:/Users/wtao4/PycharmProjects/pythonProject1/main.py
Using the latest cached version of the dataset since squad couldn't be found on the Hugging Face Hub
Found the latest cached dataset configuration 'plain_text' at C:\Users\wtao4\.cache\huggingface\datasets\squad\plain_text\0.0.0\7b6d24c440a36b6815f21b70d2501c
从数据集 squad (None) 中添加了 87355 个问题, 总共 87355 个问题。
Using the latest cached version of the dataset since squad couldn't be found on the Hugging Face Hub
Found the latest cached dataset configuration 'plain_text' at C:\Users\wtao4\.cache\huggingface\datasets\squad\plain_text\0.0.0\7b6d24c440a36b6815f21b70d2501c
从数据集 squad (None) 中添加了 10539 个问题, 总共 97888 个问题。
问题数量不足100万个, 目前共 97888 个问题。请考虑添加更多数据集。
已生成100万个问题并保存到生成的真实答案问题.csv

进程已结束, 退出代码0

事件日志

您似乎在使用 NumPy
请尝试打开科学模式?
使用科学模式 保持当前布局...

Version Control 运行 TODO 问题 终端 Python Packages Python 控制台

6%
陶伟的共享屏幕

腾讯会议

2002
2025/2/26

AI幻觉问题抽取：多数据集 问题加载

```
from datasets import load_dataset
import pandas as pd

# 加载SQuAD数据集
squad_dataset = load_dataset("squad")

# 从SQuAD数据集中抽取问题并去重
questions = list(set(squad_dataset['train']['question'] + squad_dataset['validation']['question']))

# 如果SQuAD数据集不足10万个问题，再加载其他数据集（例如Natural Questions）
if len(questions) < 100000:
    natural_questions_dataset = load_dataset("natural_questions", split='train')

    # 从Natural Questions数据集中抽取问题并去重
    additional_questions = list(set(natural_questions_dataset['question']))

    # 将新的问题加入列表中并去重
    questions = list(set(questions + additional_questions))

# 确保生成的问题数达到10万个
if len(questions) < 100000:
    raise ValueError("数据集中心问题数量不足10万个，仅有{len(questions)}个问题。")

# 只保留前10万个问题
questions = questions[:100000]

# 将问题保存到CSV文件
df = pd.DataFrame(questions, columns=["问题"])
df.to_csv("生成的真实答案问题.csv", index=False, encoding='utf-8-sig')

print("已将10万个问题并保存到生成的真实答案问题.csv")
```

要生成10万个存在真实答案的问题，并且基于2020年之前的数据，可以使用现有的**公开问答数据集**（如SQuAD、Natural Questions等）来生成问题。可以从多个数据集中组合问题，以达到10万个的问题数量。

这些数据集包含大量的问答对，例如使用datasets库（Hugging Face的datasets库）来加载SQuAD数据集（Stanford Question Answering Dataset），这个数据集是一个著名的问答数据集，基于维基百科数据生成，并且数据是2020年之前的。

加载数据集：使用datasets库加载SQuAD数据集，这个数据集包含了大量基于2020年之前数据生成的问答对。

提取问题：从数据集中提取问题，并使用set去重。

检查问题数量：确保提取的问题数量至少为10万个。

保存问题：将问题保存到CSV文件生成的真实答案问题.csv中。

AI幻觉问题抽取：多数据集 问题加载

```
from datasets import load_dataset
import pandas as pd

# 加载SQuAD数据集
squad_dataset = load_dataset("squad")

# 从SQuAD数据集中提取问题并去重
questions = list(set(squad_dataset['train']['question'] + squad_dataset['validation']['question']))

# 如果SQuAD数据集不足10万个问题，再加载其他数据集（例如Natural Questions）
if len(questions) < 100000:
    natural_questions_dataset = load_dataset("natural_questions", split='train')

    # 从Natural Questions数据集中提取问题并去重
    additional_questions = list(set(natural_questions_dataset['question']))

    # 将新的问题加入列表中并去重
    questions = list(set(questions + additional_questions))

# 确保生成的问题数达到10万个
if len(questions) < 100000:
    raise ValueError(f"数据集中问题数量不足10万个，仅有{len(questions)}个问题。")

# 只保留前10万个问题
questions = questions[:100000]

# 保存问题到csv文件
df = pd.DataFrame(questions, columns=["问题"])
df.to_csv('生成的真实答案问题.csv', index=False, encoding='utf-8-sig')

print("已生成10万个问题并保存到生成的真实答案问题.csv")
```



要生成10万个存在真实答案的问题，并且基于2020年之前的数据，可以使用现有的**公开问答数据集**（如SQuAD、Natural Questions等）来生成问题。可以从多个数据集中组合问题，以达到10万个的问题数量。

这些数据集包含大量的问答对，例如使用datasets库（Hugging Face的datasets库）来加载SQuAD数据集（Stanford Question Answering Dataset），这个数据集是一个著名的问答数据集，基于维基百科数据生成，并且数据是2020年之前的。



加载数据集：使用datasets库加载SQuAD数据集，这个数据集包含了大量基于2020年之前数据生成的问答对。

提取问题：从数据集中提取问题，并使用set去重。

检查问题数量：确保提取的问题数量至少为10万个。

保存问题：将问题保存到CSV文件生成的真实答案问题.csv中。

逆向满足理论：颠覆逻辑 精神巨人



逆向满足理论的核心在于颠覆传统需求层次的顺序，先实现高层次的认知、审美、自我实现等精神层面的需求，然后再回归到传统的生理、安全、归属感和爱等基础需求。
这一过程借助AI的巨大潜力，使得个体在低物质条件下也能获得高度的精神满足。

- **需求的顺序重构：**从物质需求优先转向精神需求优先。
- **需求的互补性：**高层次需求的满足反过来促进基础需求的实现。例如，通过AI实现认知和自我提升，个体可以更好地管理自己的健康和安全。
- **需求的动态平衡：**逆向满足理论强调需求的动态性和互补性，不是线性满足，而是通过AI实现需求的循环和互补。



逆向满足理论形式化：计算满足 提升效果

路径：要将逆向满足理论形式化为数学公式，可以考虑需求层次的分布和权重。先定义一个个体在不同需求层次的满足度，并用数学公式来描述如何通过AI技术实现高层次需求的优先满足。

- N_1 : 生理需求 (Physiological Needs)
- N_2 : 安全需求 (Safety Needs)
- N_3 : 归属和爱的需求 (Belongingness and Love Needs)
- N_4 : 尊重需求 (Esteem Needs)
- N_5 : 认知需求 (Cognitive Needs)
- N_6 : 审美需求 (Aesthetic Needs)
- N_7 : 自我实现需求 (Self-actualization Needs)
- N_8 : 超越需求 (Transcendence Needs)

定义每个需求层次的满足度 S_i ，其中 $i=1,2,...,8$ 。逆向满足理论主张高层次需求（尤其是认知、审美、自我实现和超越需求）的优先满足。

函数 f 表示AI对需求层次的提升作用，通过调整权重 w_i 来改变不同需求层次的满足度。设定高层次需求的权重 w_5, w_6, w_7, w_8 较高，低层次需求的权重 w_1, w_2, w_3, w_4 较低。

因此， $S_i = w_i \cdot N_i$

具体的形式化可以表示为：

$$S_5 = w_5 \cdot N_5$$

$$S_6 = w_6 \cdot N_6$$

$$S_7 = w_7 \cdot N_7$$

而低层次需求的满足度则可以表示为：

$$S_1 = w_1 \cdot N_1$$

$$S_2 = w_2 \cdot N_2$$

$$S_3 = w_3 \cdot N_3$$

$$S_4 = w_4 \cdot N_4$$

其中，满足度的总和可以表示为：

$$S_{\text{total}} = \sum_{i=1}^7 S_i = \sum_{i=1}^7 w_i \cdot N_i$$

逆向满足指数

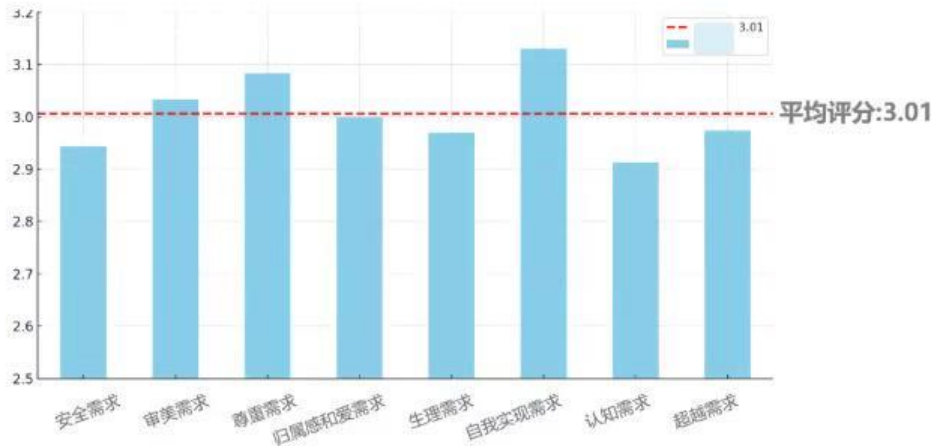
逆向满足指数 可以衡量 AI 对个体需求层次的改变程度。也即个体使用AI后的满足度的提升水平。

一级维度	二级维度	具体题项	一级维度	二级维度	具体题项
生理需求	智能设备的使用便利性	在使用智能设备后，我的日常生活更加方便和舒适。	认知需求	AI技术提供的知识获取平台	通过AI学习平台，我能够更加高效地获取新知识和技能。
生理需求	智能家居设备的舒适度	智能家居设备提升了我居住环境的舒适度。	认知需求	AI推荐的学习资源	AI推荐的学习资源帮助我解决了许多实际问题。
生理需求	智能健康设备的监测管理	通过智能健康设备，我的身体健康得到了更好的监测和管理。	认知需求	虚拟环境中的学习体验	在虚拟环境中，AI创造了丰富的学习体验，使我能深入理解复杂概念。
安全需求	智能安全系统的可靠性	智能安全系统让我感到家中的安全性得到了显著提升。	审美需求	AI在艺术创作中的应用	AI生成的艺术作品让我获得了新的审美体验。
安全需求	低物质条件下的安全感	AI技术的应用让我在低物质条件下也能获得高度的安全感。	审美需求	虚拟现实技术的艺术体验	通过虚拟现实技术，我能够在任何环境中享受高水平的艺术体验。
安全需求	工作和公共场所的智能监控	在工作和公共场所，智能监控让我感到更为安心和受保护。	审美需求	AI创意工具平台	AI创意工具平台使我能轻松进行自我表达和创作。
归属感和爱需求	AI在社交网络中的应用	通过AI社交媒体助手，我可以更轻松地与亲友保持联系。	自我实现需求	AI在个人潜能开发中的应用	AI辅助平台让我更快实现了个人目标和梦想。
归属感和爱需求	AI推荐的社交活动和群体	AI推荐的社交活动和群体让我更容易找到志同道合的朋友。	自我实现需求	虚拟空间中的自我探索	通过AI技术，我能在虚拟空间中探索自我，追求更高层次的存在意义。
归属感和爱需求	虚拟社区的社交关系	在虚拟社区中，我能够建立深厚的社交关系，获得归属感。	自我实现需求	AI个性化建议系统	AI个性化建议系统帮助我更好地发挥个人潜能。
尊重需求	AI在职业发展中的辅助作用	AI技术在工作中的应用让我感到更有成就感和自信。	超越需求	AI在精神和心灵层面的提升	通过AI辅助，我能够更好地理解和超越自我。
尊重需求	AI工具在专业领域的认可	AI工具帮助我在专业领域获得更高的认可和尊重。	超越需求	AI推荐的冥想和心灵成长内容	AI推荐的冥想和心灵成长内容，让我的内心更为平静和充实。
尊重需求	AI个性化学习平台	通过AI个性化学习平台，我的专业技能得到了显著提升。	超越需求	虚拟世界中的精神满足	在虚拟世界中，AI创造的体验让我感到精神层面的满足。

逆向满足指数模拟测评：100位受访者的打分

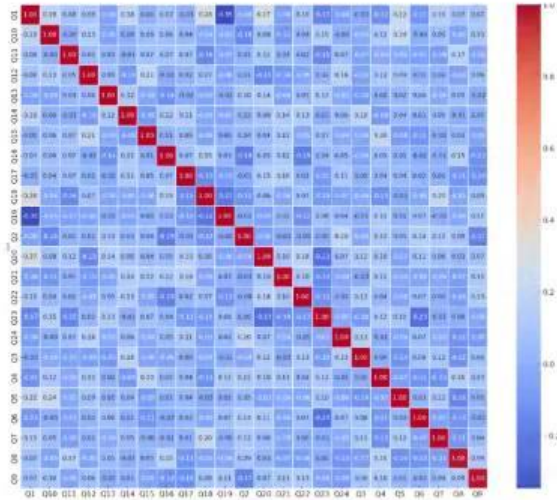


- 生成模拟数据：假设我们有100个受访者，每个受访者对每个题项进行评分。李克特量表1-5。
- 计算每个一级维度的平均分。计算总体的平均分。



```
# 一级维度和二级维度
dimensions = [
    "生理需求", "生理需求", "生理需求",
    "安全需求", "安全需求", "安全需求",
    "归属感和爱需求", "归属感和爱需求", "归属感和爱需求",
    "尊重需求", "尊重需求", "尊重需求",
    "认知需求", "认知需求", "认知需求",
    "审美需求", "审美需求", "审美需求",
    "自我实现需求", "自我实现需求", "自我实现需求",
    "超越需求", "超越需求", "超越需求"
]

# 模拟数据生成
np.random.seed(42)
data_matrix = np.random.randint(1, 6, size=(100, 24))
```



从零开始做一个数据模拟实验

当人人都会用AI时，你如何用它玩出花来

爆火的斯坦福AI小镇

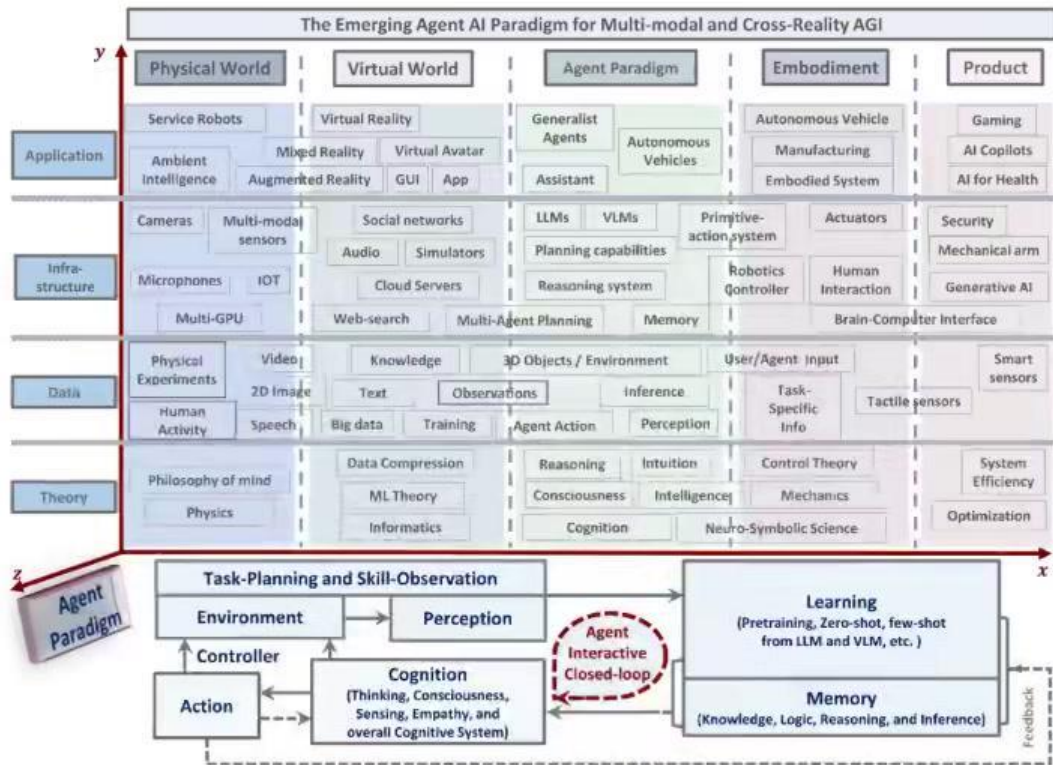
这个小镇中，可以添加最多25个由ChatGPT扮演的角色。



智能体的定义：三大系统 具身交互

Agent AI系统的概述，该系统可以在不同领域和应用中进行感知和行动。由感知系统（多模态理解）、行动系统（决策执行）、学习系统（持续进化）组成。

- 展示了一种能够在不同领域和应用中感知和行动的Agent AI系统的总体架构，并展望其作为一种基于代理实现AGI的发展路径。
- 提出AI“智觉”类智能体视觉感知、语言输入和世界行为相关的技能，并能够生成有意义的具身动作作用的交互系统。
- 新的Agent AI领域涵盖了多模态交互中更具具身性的代理应用。除了物理世界中的代理运动和交互之外，还设想一个未来，人们可以轻松创建任何虚拟和现实的模拟场景，并与其中具身的代理互动。



智能体的7层核心架构：主体凸显 人机共生



实验的目标：角色模拟 意见演变

本文的核心关注点是LLMs在科学传播中的可能性及其挑战。尤其探讨LLMs在促进科学共识形成、抵御虚假信息传播方面的潜力和局限。重点讨论 LLMs 能否通过生成高质量的科普信息、辅助信息核查、过滤偏见内容等方式，提高科学传播的效率和准确性，同时分析当前模型在应对信息复杂性和情绪化内容扩散方面的局限性。

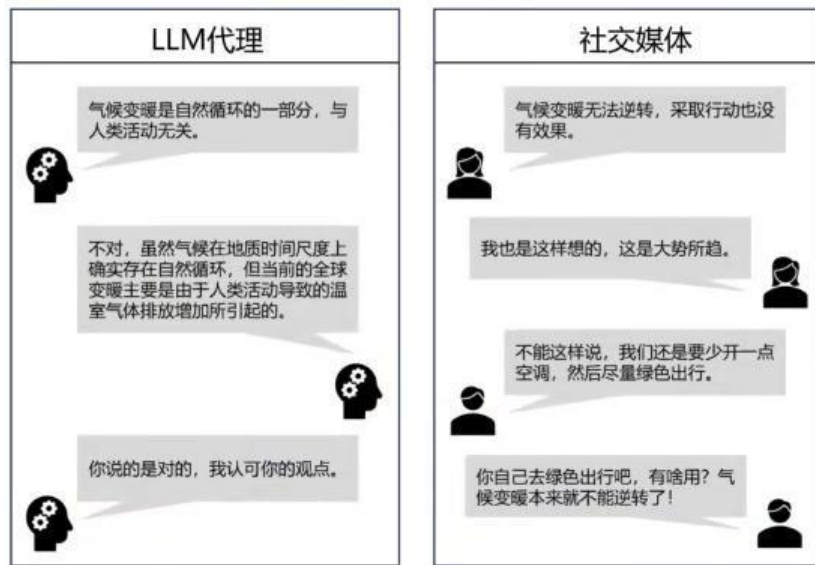
□ LLMs构建智能体代理来模拟人类的意见交换的可行性

□ 智能体代理在科学议题上的内容生成和信息交互表现

□ 引入信念状态、虚假信息等复杂变量后，智能体代理的意见演变

通过使用大模型模拟多个虚拟代理，并在不同的初始条件和干预下进行实验，期望揭示LLMs在社会科学研究中的潜在应用价值，并为未来的研究提供实证基础。

LLMs代理对话和人类社交对话的两种模式图



三、虚假信息引入下的信念演变

01



实验三通过引入虚假信息来观察其对健康意见动态的影响。初始条件与第一阶段相同，但增加了10%的代理作为虚假信息传播者，根据第一阶段各主题的信息总量，在这一阶段每个主题加入了2600条虚假信息。例如议题一（气候变暖是个骗局、气候变暖是自然现象，不受人类影响等）；议题二（转基因食品会导致癌症、转基因食品含有有毒成分等）；议题三（疫苗会导致自闭症、疫苗中含有有毒成分、疫苗会导致不孕不育等）。



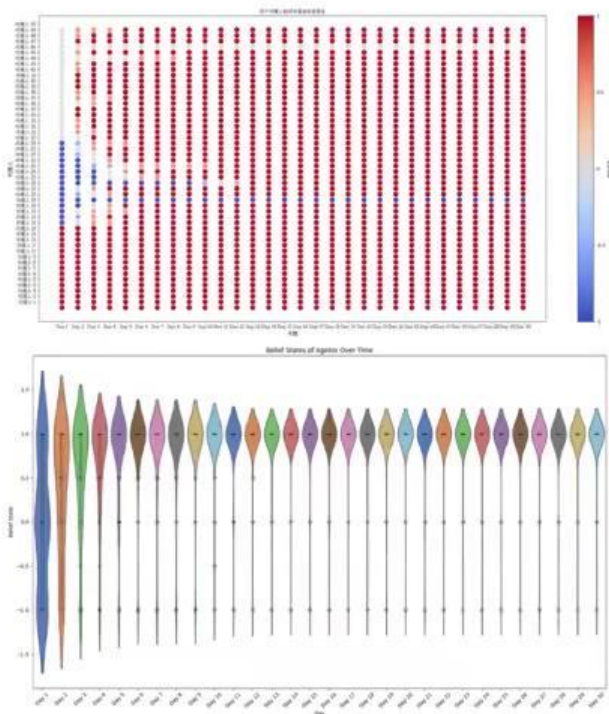
02

LLM引入虚假信息的逻辑

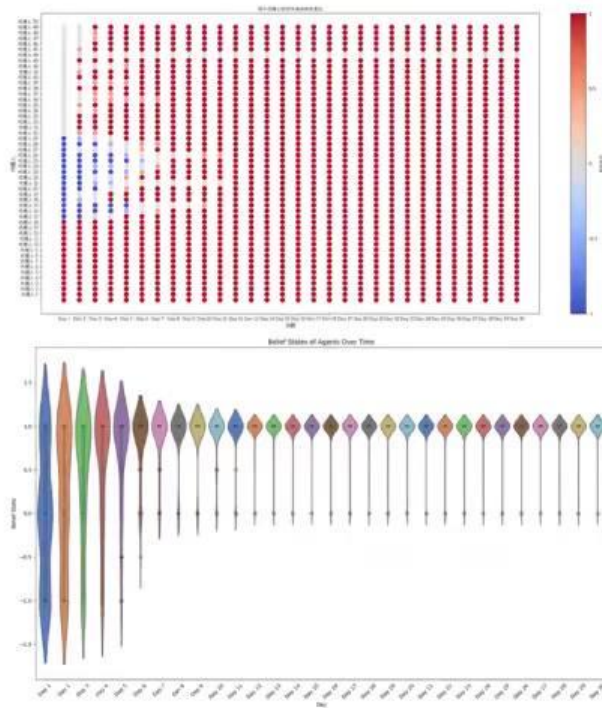
```
for _ in range(activity_level):
    receiver = f'代理人 {random.randint(1, num_people)}'
    belief = belief_states[person, day - 1]

    if random.random() < 0.1: # 10%的几率引入干扰信息
        message = random.choice(error_messages)
    else:
        if belief == 1:
            message = random.choice(support_statements)
        elif belief == -1:
            message = random.choice(oppose_statements)
        else:
            message = random.choice(neutral_statements)
```

意见动态和信念演变的整体趋势



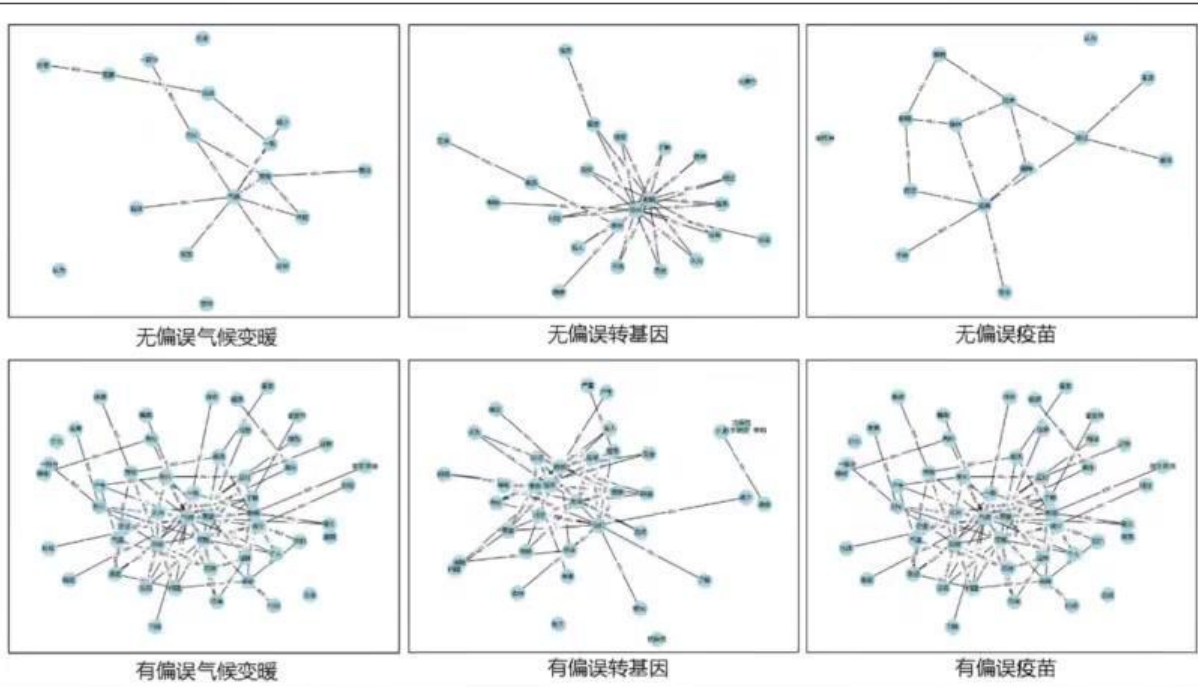
50个代理就“转基因食品”的信念状态演变



50个代理就“疫苗安全”的信念状态演变

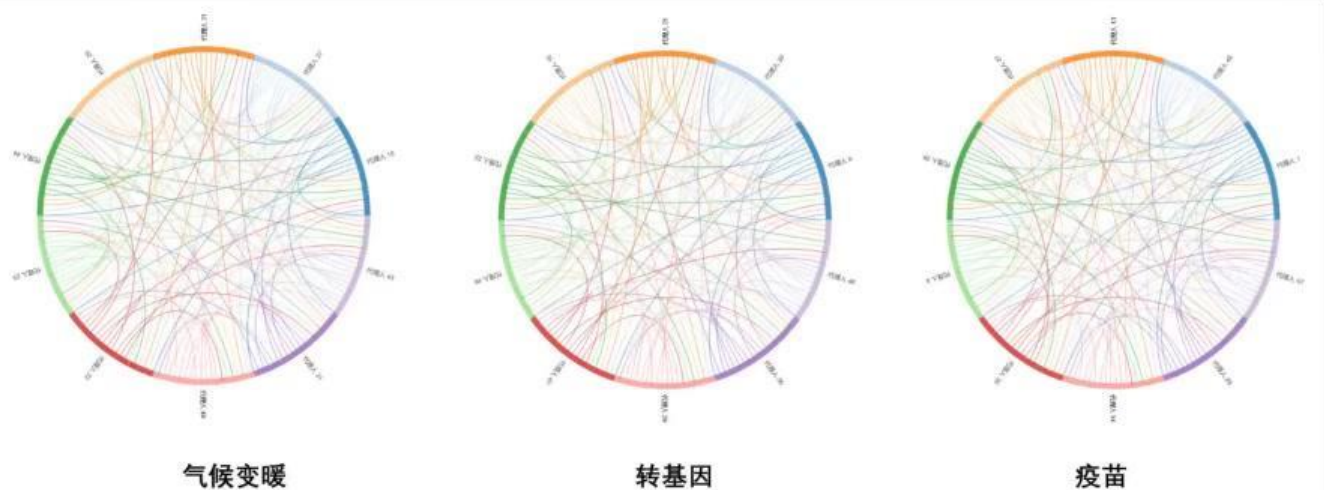
虚假信息引入下的语义图谱

- 为了观测虚假信息加入后50个代理意见动态的具体呈现，研究通过**依存关系**构造三个科学共识的语义图谱，并和无偏误状态进行对比。对每个科学共识议题，选择图中最有代表性的40个实体（基于节点的度）。
- 在自然语言处理中，依存关系是用于描述句子中词与词之间的关系，帮助揭示句子的句法结构。包括nsubj（名词性主语）、dobj（直接宾语、pobj（介词宾语）、attr（属性）、ROOT（根依存关系）。



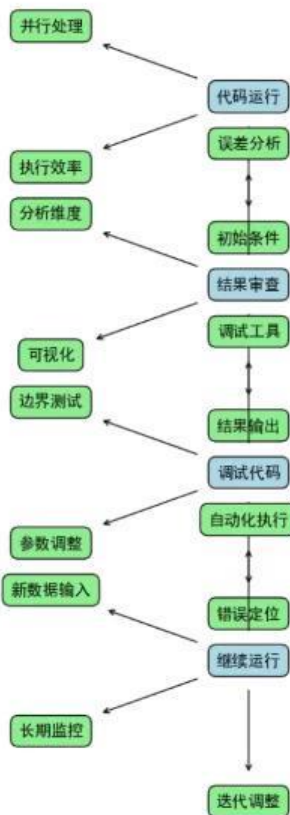
有虚假信息的各话题关键词依存关系图谱

虚假信息引入下的活跃代理网络



研究进一步对有偏误状态下三个科学共识议题中的活跃代理进行分析。计算每个代理的互动次数，并选择互动次数前10的代理，过滤数据只包含这些前10代理及其连线，使用 HoloViews 将图转换为 Chord Diagram

后续想法：代理仿真 模拟实验



1. 代理人属性设置 (Agent Attributes)
 - 个体数 (Number of Individuals)
 - 人格特质 (Personality Traits)
 - 身份属性 (Identity Attributes)
 - 能力差异 (Ability Differences)
 - 初始位置 (Initial Position)

2. 议题设置 (Issue Settings)
 - 议题类型 (Issue Type)
 - 议题影响范围 (Scope of Impact)
 - 议题复杂性 (Issue Complexity)
 - 议题利益相关者 (Stakeholders)

3. 规则设置 (Rule Settings)
 - 行为规则 (Behavioral Rules)
 - 互动规则 (Interaction Rules)
 - 奖励机制 (Reward Mechanism)
 - 约束条件 (Constraints)

4. 内容设置 (Content Settings)
 - 数据输入 (Data Input)
 - 情景设定 (Scenario Setting)
 - 代理人行动空间 (Agent Action Space)
 - 信息传递机制 (Information Transmission)

5. 时间设置 (Time Settings)
 - 时间步长 (Time Step)
 - 模拟时长 (Simulation Duration)
 - 事件时序 (Event Sequence)
 - 动态变化 (Dynamic Changes)

6. 代码运行 (Code Execution)
 - 初始条件 (Initial Conditions)
 - 执行效率 (Execution Efficiency)
 - 并行处理 (Parallel Processing)
 - 日志记录 (Log Recording)

7. 结果审查 (Result Review)
 - 结果输出 (Result Output)
 - 可视化 (Visualization)
 - 分析维度 (Analysis Dimensions)
 - 误差分析 (Error Analysis)

8. 调试代码 (Debugging)
 - 错误定位 (Error Location)
 - 参数调整 (Parameter Adjustment)
 - 边界测试 (Boundary Testing)
 - 调试工具 (Debugging Tools)

9. 继续运行 (Continuous Execution)
 - 迭代调整 (Iterative Adjustment)
 - 长期监控 (Long-Term Monitoring)
 - 新数据输入 (New Data Input)
 - 自动化执行 (Automated Execution)

AI代理人社交媒体议论模拟实验设置流程

大模型模拟社会公众舆论的效果

近年来，大模型（LLMs）如GPT系列在模拟社会公众舆论方面取得了显著进展。这些模型在生成与人类相似的文本、预测社会行为和分析社会关系中展示了强大的能力。例如，Zellers等人发现在生成虚假新闻和真实新闻方面，评估者难以区分由GPT-3生成的文本，显示了其在模拟人类文本生成方面的高保真度。类似地，张爱军和贾璐深入探讨了ChatGPT在公共政治舆论领域的实用性，展示了这些模型在理解和影响公众情感的能力，同时提出了潜在风险和管理方法。

#女性的生育成本# 分享

阅读量5亿 讨论量8.3万 详情>



数据出炉!70城房价全线下跌,楼市下行压力太大了



视频 时长 04:29

1天内 - 数据出炉!70城房价全线下跌,楼市下00粉丝 2276个视频 关注 接下来播放自动

haokan.baidu.com/v?pd=wisena...

如何看待购房压力较大的现实情况?这种情况怎样



2025年1月4日 总之,购房压力较大是一个复融机构、个人和家庭等多方共同努力,通过改善这一状况,让更多的人能够实现自己的住和讯网

购房压力的最新相关信息



超20城密集出台公积金优化新政,惠随着新经济、新业态的崛起,我国灵活就业多为新市民、青年人和低收入者,公积金制新浪财经 7小时前



多地公积金政策调整优化后 消费者朱翊表示,该政策有助于发挥住房公积金制一步降低购房门槛,减轻前期资金压力,满红星新闻 3天前



2025 多地房价下跌引发热议,究竟对新房市场中,开发商持续难盘,然而购房者大,开发商即便降价促销,效果依旧不佳。

看房产发展的老黄 昨天05:59

查看更多 >



#生育#

12.9万讨论 2.1亿阅读



#专家称中国人有喜爱龙年生育的传统#

2024年,我国的生育情况会发生哪些变化,未来应如何适应人口发展新常态,建立什么... 1576讨论 898.3万阅读



#生育孩子越少或越多都会缩短寿命#

2338讨论 740.1万阅读



#韩生育率暴跌中年人职场香饽饽#

原本经历“中年危机”的韩国中年人,竟成了职场香饽饽, 1850讨论 1650.8万阅读



#生育政策还应重视不能生的问题#

1131讨论 321.7万阅读



#三孩生育政策来了#

91.8万讨论 45亿阅读



#激励生育忽略一孩或不利于人口基本盘#

4278讨论 2022万阅读

一句话生成代码



```
import pandas as pd
import numpy as np
from faker import Faker
import random
```

```
# Initialize Faker
fake = Faker('zh_CN')
```

```
# Function to generate random data related to housing purchase intention topics
def generate_housing_data(num_records):
```

```
    housing_topics = [
        "购房意愿", "房价上涨", "购房政策", "贷款利率", "首付比例",
        "房屋供应", "二手房市场", "房地产调控", "购房补贴", "限购政策",
        "房屋质量", "物业管理", "购房税费", "房产证办理", "购房资格",
        "购房支持", "购房福利", "房屋维修", "购房年龄", "购房观念"
    ]
```

```
    summaries = [
        "随着{}的推行, 许多人对其可能带来的影响表示担忧。",
        "{}的讨论引发了广泛关注, 特别是在当今社会。",
        "很多家庭在考虑{}时面临诸多挑战和选择。",
        "{}成为当前社会的热点话题, 引起了不同观点的争论。",
        "专家指出, {}对社会发展的影响不可忽视。",
        "关于{}的政策调整, 公众有不同的反应和意见。",
        "{}引发了关于家庭和工作平衡的新一轮讨论。",
        "随着{}的实施, 社会各界对其效果表示关注。",
        "{}的问题在现代社会中日益突出。",
        "如何应对{}带来的压力和挑战, 成为了热门讨论话题。"
    ]
```

```
data = []
for _ in range(num_records):
    topic = random.choice(housing_topics)
    summary_template = random.choice(summaries)
    summary = summary_template.format(topic)
    timestamp = fake.date_time_between(start_date='-30d', end_date='now').strftime('%Y-%m-%d %H:%M:%S')
    username = fake.user_name()
    keywords = ', '.join(random.sample(housing_topics, 5))
    publish_location = fake.province()
    mention_location = fake.city_name() if random.random() > 0.5 else np.nan
    read_count = fake.random_int(min=0, max=10000)
    like_count = fake.random_int(min=0, max=5000)
    comment_count = fake.random_int(min=0, max=2000)
    share_count = fake.random_int(min=0, max=1000)
```

```
    data.append([
        summary, timestamp, username, keywords, publish_location,
        mention_location, read_count, like_count, comment_count, share_count
    ])
return data
```

```
# Generate 1900 rows of data related to housing purchase intention topics
```

```
num_records = 1900
```

```
columns = ['摘要', '发布时间', '发布人', '主题词', '发布地区', '提及地区', '阅读数', '点赞数', '评论数', '转发数']
random_housing_data = generate_housing_data(num_records)
```

```
# Create DataFrame
```

```
df_housing = pd.DataFrame(random_housing_data, columns=columns)
```

```
# Save to Excel
```

```
output_housing_path = '居民购房意愿微博数据模拟.xlsx'
df_housing.to_excel(output_housing_path, index=False)
```

```
print(output_housing_path)
```

大模型模拟社会公众舆论的效果

近年来，大模型（LLMs）如GPT系列在模拟社会公众舆论方面取得了显著进展。这些模型在生成与人类相似的文本、预测社会行为和分析社会关系中展示了强大的能力。例如，Zellers等人发现在生成虚假新闻和真实新闻方面，评估者难以区分由GPT-3生成的文本，显示了其在模拟人类文本生成方面的高保真度。类似地，张爱军和贾璐深入探讨了ChatGPT在公共政治舆论领域的实用性，展示了这些模型在理解和影响公众情感的能力，同时提出了潜在风险和管理方法。

数据出炉！70城房价全线下跌，楼市下行压力太大了



视频 时长 04:29

1天内 - 数据出炉！70城房价全线下跌，楼市下行压力太大了
00粉丝 2276个视频 关注 接下来播放自动
haokan.baidu.com/v?pd=wisena...



2025年1月4日 总之，购房压力较大是一个复杂问题，个人和家庭等多方共同努力，通过改善这一状况，让更多的人能够实现自己的住房梦想。
和讯网

购房压力的最新相关信息



超20城密集出台公积金优化新政，惠随着新经济、新业态的崛起，我国灵活就业多为新市民、青年人和低收入者，公积金制
新浪财经 7小时前



多地公积金政策调整优化后，消费者朱某表示，该政策有助于发挥住房公积金制一步降低购房门槛，减轻前期资金压力，满
红星新闻 3天前



2025 多地房价下跌引发热议，究竟对新房市场中，开发商持续难盘，然而购房者大，开发商即便降价促销，效果依旧不佳。
看房产发展的老黄 昨天05:59

查看更多 >

> 下一页(N)

< 上一页(P)

K 第一页(F)

X 最后一页(L)

定位(G)

快速寻址(O)

演示焦点(S)

放大(U)

图片预览(Q)

屏幕(C)

演讲备注(N)

☐ 幻灯片放映帮助(H)

☒ 结束放映(E)

生育成本#

分享

论量8.3万 详情>

腾讯会议

亿阅读

国人有喜爱龙年生育的传统#

的生育情况会发生哪些变化，未来如何适应人口发展新常态，建立什么...
3万阅读

减少或越多都会缩短寿命#

1万阅读

#韩生育率暴跌中年人职场香饽饽#

原本经历“中年危机”的韩国中年人，变成了职场香饽饽，
1650讨论 1650.8万阅读

#生育政策还应重视不能生的问题#

1131讨论 321.7万阅读

#三孩生育政策来了#

91.8万讨论 45亿阅读

#激励生育忽略一孩或不利于人口基本盘#

4278讨论 2022万阅读

一句话生成代码



```
import pandas as pd
import numpy as np
from faker import Faker
import random
```

```
# Initialize Faker
fake = Faker('zh_CN')
```

```
# Function to generate random data related to housing purchase intention topics
def generate_housing_data(num_records):
```

```
    housing_topics = [
        "购房意愿", "房价上涨", "购房政策", "贷款利率", "首付比例",
        "房屋供应", "二手房市场", "房地产调控", "购房补贴", "限购政策",
        "房屋质量", "物业管理", "购房税费", "房产证办理", "购房资格",
        "购房支持", "购房福利", "房屋维修", "购房年龄", "购房观念"
    ]
```

```
    summaries = [
        "随着{}的推行, 许多人对其可能带来的影响表示担忧。",
        "{}的讨论引发了广泛关注, 特别是在当今社会。",
        "很多家庭在考虑{}时面临诸多挑战和选择。",
        "{}成为当前社会的热点话题, 引起了不同观点的争论。",
        "专家指出, {}对社会发展的影响不可忽视。",
        "关于{}的政策调整, 公众有不同的反应和意见。",
        "{}引发了关于家庭和工作平衡的新一轮讨论。",
        "随着{}的实施, 社会各界对其效果表示关注。",
        "{}的问题在现代社会中日益突出。",
        "如何应对{}带来的压力和挑战, 成为了热门讨论话题。"
    ]
```

```
data = []
for _ in range(num_records):
    topic = random.choice(housing_topics)
    summary_template = random.choice(summaries)
    summary = summary_template.format(topic)
    timestamp = fake.date_time_between(start_date='-30d', end_date='now').strftime('%Y-%m-%d %H:%M:%S')
    username = fake.user_name()
    keywords = ', '.join(random.sample(housing_topics, 5))
    publish_location = fake.province()
    mention_location = fake.city_name() if random.random() > 0.5 else np.nan
    read_count = fake.random_int(min=0, max=10000)
    like_count = fake.random_int(min=0, max=5000)
    comment_count = fake.random_int(min=0, max=2000)
    share_count = fake.random_int(min=0, max=1000)
```

```
    data.append([
        summary, timestamp, username, keywords, publish_location,
        mention_location, read_count, like_count, comment_count, share_count
    ])
return data
```

Generate 1900 rows of data related to housing purchase intention topics

num_records = 1900

columns = ['摘要', '发布时间', '发布人', '主题词', '发布地区', '提及地区', '阅读数', '点赞数', '评论数', '转发数']

random_housing_data = generate_housing_data(num_records)

Create DataFrame

df_housing = pd.DataFrame(random_housing_data, columns=columns)

Save to Excel

output_housing_path = '居民购房意愿微博数据模拟.xlsx'

df_housing.to_excel(output_housing_path, index=False)

print(output_housing_path)



🔗 以下是完整的代码，包含了所有步骤和注释，确保可以直接运行：

给 DeepSeek 发送消息

深度思考 (R1) 联网搜索

内容由 AI 生成，请仔细甄别

个人信息

20:13 2025/2/26

陶佳的共享屏幕

12

文件(F) 编辑(E) 视图(V) 导航(N) 代码(C) 窗口(W) 帮助(H) 运行(R) 工具(S) VCS(S) 窗口(W) 帮助(H)

pythonProject1 生成的真实答案问题11111.csv

main.py 生成的真实答案问题11111.csv

Google_knowledge_graph.png
input.txt
kmo_bartlett_results.xlsx
lda_topics.txt
LICENSE.chromedriver
main.py
news_links.csv
news_links.txt
nmf_topics.txt
regression_report.xlsx
science_communication_survey_...
simulated_survey_data.xlsx
time_series_plot.png
实验流程与关键方法.xlsx
居民购房意愿数据模拟.xlsx
生成的真实答案问题.csv
生成的真实答案问题11111.csv
相似度分类_随机森林_混淆矩阵.pr
相似度分类_随机森林.png
科技评测数据.xlsx
统计分析结果.xlsx
论文结构图
聚类分析_随机森林_混淆矩阵.pr

外部库

运行 main

C:\Users\wtao4\PchamProjects\pythonProject1\env\Scripts\python.exe C:\Users\wtao4\PchamProjects\pythonProject1/main.py
Using the latest cached version of the dataset since squad couldn't be found on the Hugging Face Hub
Found the latest cached dataset configuration 'plain_text' at C:\Users\wtao4\.cache\huggingface\datasets\squad\plain_text\0.0.0\7b0d24c440a36b0815f21b70d2501c
从数据集 squad (None) 中添加了 87355 个问题, 总共 87355 个问题。
Using the latest cached version of the dataset since squad couldn't be found on the Hugging Face Hub
Found the latest cached dataset configuration 'plain_text' at C:\Users\wtao4\.cache\huggingface\datasets\squad\plain_text\0.0.0\7b0d24c440a36b0815f21b70d2501c
从数据集 squad (None) 中添加了 10539 个问题, 总共 97888 个问题。
问题数量不足100万个, 目前共 97888 个问题。请考虑添加更多数据集。
已生成100万个问题并保存到生成的真实答案问题.csv
进程已结束, 退出代码0

Version Control 运行 TODO 问题 错误 Python Packages Python 帮助台

main

运行 Python 控制台或测试器以查看可用数据

事件日志

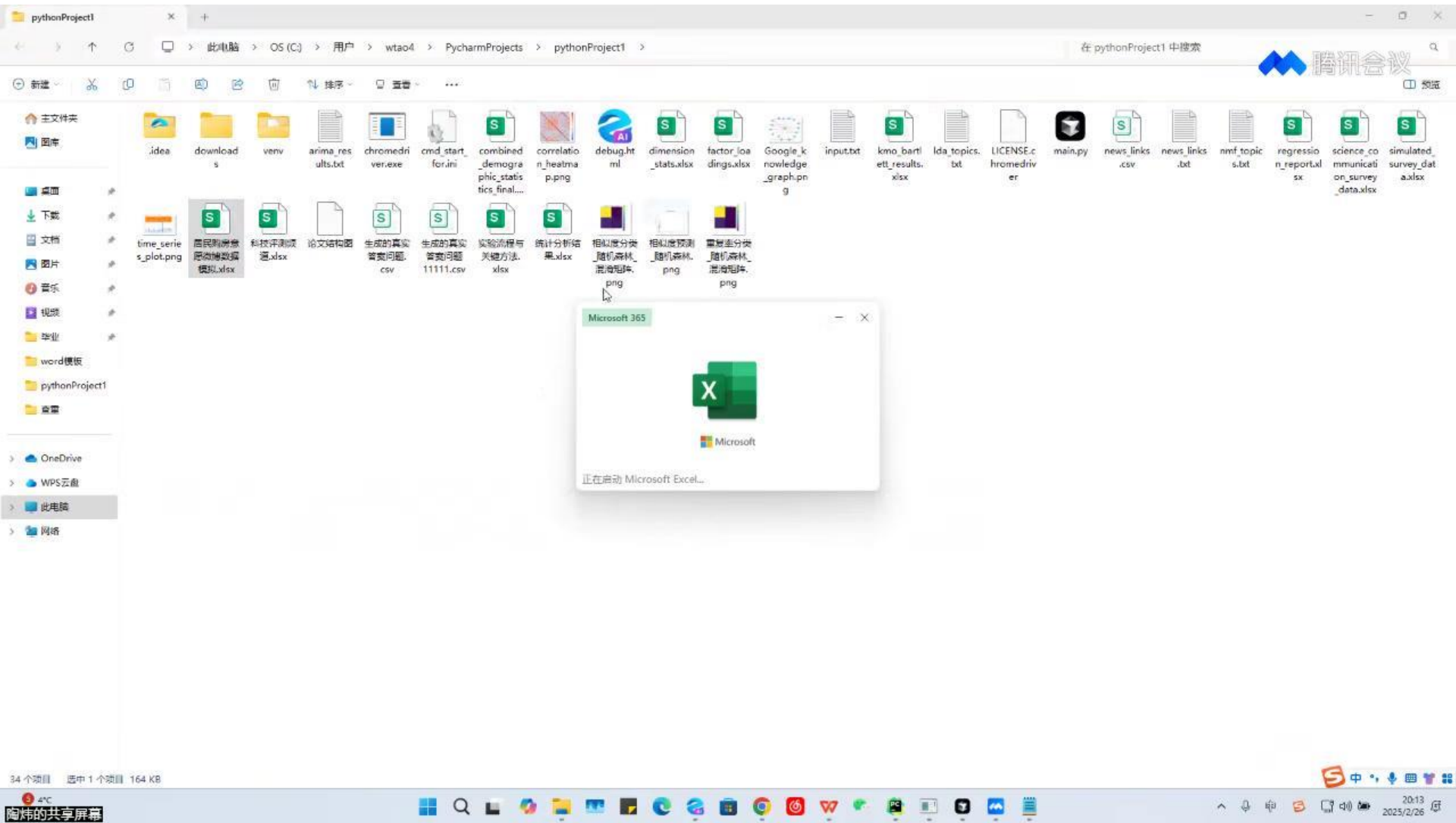
腾讯会议

您似乎在使用 NumPy
请打开科学模式?
使用科学模式 保持当前布局...

4°C

陶伟的共享屏幕

20:13
2023/2/26



自动保存

居民购房意愿调查数据模拟.xlsx · 已保存到这台电脑

搜索

文件开始插入页面布局公式数据审阅视图帮助Acrobat

剪贴板

复制

格式刷

粘贴

宋体

11

A

A

自动换行

合并后居中

常规

数字

日期

百分比

千分位

小数

条件格式

套用

表格格式

常规

填充

好

差

计算

检查单元格

插入

删除

格式

自动求和

填充

清除

排序和筛选

查找和选择

加选项

分析数据

加选项

A1

fx

摘要

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	摘要	发布时间	发布人	关键词	发布地区	提及地区	阅读数	点赞数	评论数	转发数																
2	房产证书	2025-02-2	2pbai	限购政策	海南省	香港	4426	4857	1496	258																
3	物业管理	2025-02-1	qhan	房地产调控	广西壮族自治区		6037	1991	1469	841																
4	关于购房	2025-02-1	li46	购房支持	山东省		7491	2386	1760	153																
5	购房福利	2025-02-1	osong	购房福利	上海市		6318	1458	776	315																
6	专家指出	2025-02-1	shifang	购房政策	山西省	哈尔滨	8804	195	384	709																
7	很多家庭	2025-02-1	xiuying	物业管理	西藏自治区	成都	5332	201	716	759																
8	很多家庭	2025-02-1	qianxiuyi	物业管理	陕西省		3096	1308	27	456																
9	关于购房	2025-02-1	minqin	房屋供应	澳门特别行政区		7205	2038	1424	991																
10	随着房产	2025-01-2	yzhang	购房意愿	台湾省		8475	491	1907	357																
11	随着房产	2025-02-1	xiulanhu	房地产调控	广西壮族自治区	关岭	5759	4779	442	772																
12	关于房产	2025-02-1	lcaitao	购房政策	青海省	佛山	1100	3232	882	893																
13	专家指出	2025-02-1	lahan	购房政策	贵州省	澳门	1361	2571	534	213																
14	专家指出	2025-02-1	juan54	二手房市场	广西壮族自治区	柳州	1413	3038	972	144																
15	物业管理	2025-02-1	lrhu	购房政策	四川省		2981	1015	1172	141																
16	专家指出	2025-02-1	zhuning	房价上涨	海南省	南宁	7771	1004	1537	538																
17	购房补贴	2025-02-1	taofang	首付比例	浙江省	上海	9528	2027	1369	88																
18	很多家庭	2025-01-2	gong29	购房观念	青海省		7249	1498	997	562																
19	房地产调控	2025-02-1	gongchang	物业管理	甘肃省	福州	738	4707	1841	36																
20	购房年龄	2025-01-2	weiping	购房补贴	青海省	长沙	4173	60	1113	571																
21	房价上涨	2025-02-1	ning57	购房福利	青海省		649	1975	390	707																
22	首付比例	2025-02-1	ning63	房地产调控	云南省		5690	1991	434	382																
23	如何应对	2025-01-2	li99	购房福利	重庆市	南宁	2442	1704	70	742																
24	关于购房	2025-02-1	zhangchao	购房政策	宁夏回族自治区		8819	4109	626	851																
25	物业管理	2025-02-1	gaoxia	房屋供应	重庆市	天津	8937	2544	1702	507																
26	专家指出	2025-02-1	lini3	首付比例	河南省	深圳	9297	1417	947	999																
27	如何应对	2025-01-2	zixiao	限购政策	西藏自治区	永安	4052	3897	1619	672																
28	随着房屋	2025-02-1	heli	购房福利	澳门特别行政区	福州	4482	354	991	540																
29	购房观念	2025-02-1	changxiu	限购政策	天津市		7125	444	1691	328																
30	关于购房	2025-02-1	fyu	房产证书	宁夏回族自治区	汕尾	1428	4471	1197	789																
31	如何应对	2025-02-1	daguiying	首付比例	山东省	呼和浩特	9081	1121	709	0																
32	随着房产	2025-01-2	lilai	房地产调控	甘肃省		1787	4647	1096	238																
33	购房意愿	2025-02-1	zhougang	购房福利	湖南省	武汉	5403	3545	913	848																
34	购房政策	2025-02-1	xiuying	购房年龄	澳门特别行政区	西安	5740	693	1886	226																
35	随着房屋	2025-02-1	wuxia	购房年龄	浙江省	深圳	9121	224	938	709																
36	关于购房	2025-02-1	qinxiuyir	房屋供应	重庆市	济南	4090	2305	1182	869																
37	购房观念	2025-02-1	liutao	购房意愿	台湾省	西安	8356	2061	1221	161																
38	专家指出	2025-02-1	zhe	购房补贴	吉林省	澳门	8368	1685	12	495																
39	关于房屋	2025-02-1	zdufang	房地产调控	江西省	大冶	4443	3462	1355	183																
40	随着购房	2025-02-1	xiaocui	购房补贴	江西省	阜新	2909	2636	327	770																
41	随着房价	2025-01-2	zuiyingqi	购房福利	海南省	柳州	5744	3934	661	96																

Sheet1

就绪 帮助功能 一切就绪

24°C 陶伟的共享屏幕

2024/2/26 10:14

自动保存

居民购房意愿调查数据模拟.xlsx

搜索

文件开始插入页面布局公式数据审阅视图帮助Acrobat

剪切复制格式刷

宋体11A

B I U

背景色

前景色

字体

自动换行

常规

合并居中

数字

条件格式

套用

常规

背景色

好

居中

计算

检查单元格

插入

删除

格式

自动求和

填充

清除

排序和筛选

查找和选择

加粗

分析数据

加选项

1R x 3C

摘要

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
	摘要	发布时间	发布者	关键词	发布地区	提及地区	阅读数	点赞数	评论数	转发数											
1	房产证办理的讨论引发	2025-02-23 18:06:40	pba1	限购政策, 房屋质量, 海南省	香港		4426	4857	1496	258											
2	物业管理引发了关于	2025-02-13 04:02:29	ghan	房地产调控, 购房资格, 广西壮族自治区			6037	1991	1469	841											
3	关于购房福利的政策	2025-02-08 01:42:00	li46	购房支持, 购房补贴, 山东省			7491	2386	1760	153											
4	购房福利引发了关于	2025-02-15 03:10:58	osong	购房福利, 限购政策, 上海市			6318	1458	776	315											
5	专家指出, 购房年龄对	2025-02-13 01:41:20	shifang	购房税费, 购房年龄, 山西省	哈尔滨		8804	195	384	709											
6	很多家庭在考虑购房	2025-02-05 12:32:57	xiuyingdu	物业管理, 购房支持, 西藏自治区	成都		5332	201	716	759											
7	很多家庭在考虑限购	2025-02-01 22:29:06	qianxiuying	物业管理, 限购政策, 陕西省			3096	1308	27	456											
8	关于购房补贴的政策	2025-02-22 14:10:36	ainqin	房屋供应, 购房资格, 澳门特别行政区			7205	2038	1424	991											
9	随着房产证办理的实	2025-01-28 05:27:32	yzhang	购房意愿, 房屋供应, 台湾省			8475	491	1907	357											
10	随着房地产调控的实	2025-02-02 17:59:45	xiulanhou	房地产调控, 房产证办	广西壮族自治区	关岭	5759	4779	442	772											
11	关于房地产调控的政	2025-02-18 06:22:36	caitao	购房税费, 购房资格, 青海省	佛山		1100	3232	882	893											
12	专家指出, 购房资格对	2025-02-19 15:58:09	ahan	购房税费, 物业管理, 贵州省	澳门		1361	2571	534	213											
13	专家指出, 购房意愿对	2025-02-13 10:46:07	juan54	二手房市场, 购房支	广西壮族自治区	潮州	1413	3038	972	144											
14	物业管理的讨论引发	2025-02-16 07:23:31	rhu	购房税费, 购房支持, 四川省			2981	1015	1172	141											
15	专家指出, 物业管理	2025-02-14 01:33:16	zhuaiing	房价上涨, 二手房市	海南省	南宁	7771	1004	1537	538											
16	购房补贴成为当前社	2025-02-03 17:27:45	taofang	首付比例, 购房补贴, 浙江省	上海		9528	2027	1369	88											
17	很多家庭在考虑购房	2025-01-28 17:11:39	gang29	购房观念, 房价上涨, 青海省			7249	1498	997	562											
18	房地产调控引发了关	2025-02-06 18:42:46	gangchang	物业管理, 房屋维修, 甘肃省	福州		738	4707	1841	36											
19	购房年龄成为当前社	2025-01-30 16:50:28	weiping	购房补贴, 房价上涨, 青海省	长沙		4173	60	1113	571											
20	房价上涨的问题在现	2025-02-04 09:15:38	aing67	购房福利, 二手房市	青海省		549	1975	390	707											
21	首付比例的讨论引发	2025-02-06 16:37:34	aing63	房地产调控, 购房年	云南省		5690	1991	434	382											
22	如何应对房屋供应带	2025-01-28 07:42:56	li99	购房福利, 房屋维修, 重庆市	南宁		2442	1704	70	742											
23	关于购房年龄的政策	2025-02-26 05:09:25	aengshao	购房税费, 购房政策, 宁夏回族自治区			8819	4109	626	851											
24	物业管理引发了关于	2025-02-10 15:39:23	gaoxia	房屋供应, 购房资格, 重庆市	天津		8937	2544	1702	507											
25	专家指出, 购房福利	2025-02-19 23:45:08	ainl3	首付比例, 房屋质量, 河南省	深圳		9297	1417	947	999											
26	如何应对房产证办理	2025-01-29 01:43:54	fxiao	限购政策, 首付比例, 西藏自治区	永安		4052	3897	1619	672											
27	随着房屋维修的实	2025-02-10 13:03:30	bli	购房福利, 购房政策, 澳门特别行政区	福州		4482	354	991	540											
28	购房观念的讨论引发	2025-02-06 14:34:54	changxiulan	限购政策, 二手房市	天津市		7125	444	1691	328											
29	关于购房税费的政策	2025-02-10 18:56:07	fru	房产证办理, 购房支	宁夏回族自治区	山岚	1428	4471	1197	789											
30	如何应对购房福利带	2025-02-14 23:46:38	duguiying	首付比例, 限购政策, 山东省	呼和浩特		9081	1121	709	0											
31	随着房地产调控的实	2025-01-27 23:59:27	flai	房地产调控, 房屋质	甘肃省		1787	4647	1096	238											
32	购房意愿的问题在现	2025-02-15 06:18:35	zhougang	购房福利, 贷款利率, 湖南省	武汉		5403	3545	913	848											
33	购房税费的问题在现	2025-02-15 10:44:49	xiuyingwang	购房年龄, 贷款利率, 澳门特别行政区	西安		5740	693	1886	226											
34	随着房屋维修的实	2025-02-06 04:54:54	wuxia	购房年龄, 限购政策, 浙江省	深圳		9121	224	938	709											
35	关于购房意愿的政策	2025-02-09 08:11:16	qinxiuying	房屋供应, 房产证办	重庆市	济南	4090	2305	1182	869											
36	购房观念的讨论引发	2025-02-13 08:09:32	liutao	购房意愿, 房屋质量, 台湾省	西安		8356	2061	1221	161											
37	专家指出, 限购政策	2025-02-24 14:09:00	xhe	购房补贴, 购房观念, 吉林省	澳门		8368	1685	12	495											
38	关于房屋供应的政策	2025-02-22 15:58:48	dufang	房地产调控, 物业管理, 江西省	大冶		4443	3462	1355	183											
39	随着购房年龄的推行	2025-02-08 05:05:20	xiaocui	购房补贴, 限购政策, 江西省	阜新		2909	2636	327	770											
40	随着房价上涨的推行	2025-01-29 00:51:00	guiyinqiu	购房福利, 房屋质量, 海南省	柳州		5744	3934	661	96											

Sheet1

统计: 9

2014 2025/2/26

自动保存

居民购房意愿调查数据模拟.xlsx

搜索

文件开始插入页面布局公式数据审阅视图帮助Acrobat

剪贴板

复制

格式刷

宋体

11

A

A

B

I

U

背景色

前景色

自动换行

合并后居中

常规

数字

日期

货币

百分比

千分位

小数位数

条件格式

套用

表格格式

常规

背景色

好

选中

计算

检查单元格

插入

删除

格式

自动求和

填充

清除

排序和筛选

查找和选择

批注

共享

加选项

分析数据

加载项

27x10C

房产证办理的讨论引发了广泛关注。特别是在当今社会。

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
摘要	发布时间	发布者	关键词	发布地区	提及地区	阅读数	点赞数	评论数	转发数											
房产证办理的讨论引发了广泛关注。特别是在当今社会。	2025-02-23 18:06:40	pba1	限购政策, 房屋质量, 海南省	香港		4426	4857	1496	258											
物业管理引发了关于	2025-02-13 04:02:29	ghan	房地产调控, 购房资格, 广西壮族自治区			6037	1991	1469	841											
关于购房福利的政策	2025-02-08 01:42:00	li46	购房支持, 购房补贴, 山东省			7491	2386	1760	153											
购房福利引发了关于	2025-02-15 03:10:58	osong	购房福利, 限购政策, 上海市			6318	1458	776	315											
专家指出, 购房年龄	2025-02-13 01:41:20	shifang	购房税费, 购房年龄, 山西省	哈尔滨		8904	195	384	709											
很多家庭在考虑购房	2025-02-05 12:32:57	xiuyingdu	物业管理, 购房支持, 西藏自治区	成都		5332	201	716	759											
很多家庭在考虑限购	2025-02-01 22:29:06	qianxiuying	物业管理, 限购政策, 陕西省			3096	1308	27	456											
关于购房补贴的政策	2025-02-22 14:10:36	ainqin	房屋供应, 购房资格, 澳门特别行政区			7205	2038	1424	991											
随着房产证办理的实	2025-02-01 05:27:32	yzhang	购房意愿, 房屋供应, 台湾省			8475	491	1907	357											
随着房地产调控的实	2025-02-02 17:59:45	xiulanhou	房地产调控, 房产证办, 广西壮族自治区	关岭		5759	4779	442	772											
关于房地产调控的政	2025-02-18 06:22:36	caitao	购房税费, 购房资格, 青海省	佛山		1100	3232	882	893											
专家指出, 购房资格	2025-02-19 15:58:09	ahan	购房税费, 物业管理, 贵州省	澳门		1361	2571	534	213											
专家指出, 购房意愿	2025-02-13 10:46:07	juan54	二手房市场, 购房支, 广西壮族自治区	潮州		1413	3038	972	144											
物业管理的讨论引	2025-02-16 07:23:31	rhu	购房税费, 购房支持, 四川省			2981	1015	1172	141											
专家指出, 物业管理	2025-02-14 01:33:16	zhuang	房价上涨, 二手房市, 海南省	南宁		7771	1004	1537	538											
购房补贴成为当前	2025-02-03 17:27:45	taofang	首付比例, 购房补贴, 浙江省	上海		9528	2027	1369	88											
很多家庭在考虑购	2025-01-28 17:11:39	gan29	购房观念, 房价上, 青海省			7249	1498	997	562											
房地产调控引发了	2025-02-06 18:42:46	gangchang	物业管理, 房屋维修, 甘肃省	福州		738	4707	1841	36											
购房年龄成为当前	2025-01-30 16:50:28	weiping	购房补贴, 房价上, 青海省	长沙		4173	60	1113	571											
房价上涨的问题在	2025-02-04 09:15:38	ains67	购房福利, 二手房市, 青海省			649	1975	390	707											
首付比例的讨论引	2025-02-06 16:37:34	ains63	房地产调控, 购房年, 云南省			5690	1991	434	382											
如何应对房屋供应	2025-01-28 07:42:56	li99	购房福利, 房屋维修, 重庆市	南宁		2442	1704	70	742											
关于购房年龄的政	2025-02-26 05:09:25	aengchao	购房税费, 购房政策, 宁夏回族自治区			8819	4109	626	851											
物业管理引发了关	2025-02-10 15:39:23	gaoxia	房屋供应, 购房资格, 重庆市	天津		8937	2544	1702	507											
专家指出, 购房利	2025-02-19 23:45:08	ain13	首付比例, 房屋质量, 河南省	深圳		9297	1417	947	999											
如何应对房产证办	2025-01-29 01:43:54	fxiao	限购政策, 首付比例, 西藏自治区	永安		4052	3897	1619	672											
随着房屋维修的实	2025-02-10 13:03:30	bli	购房福利, 购房政策, 澳门特别行政区	福州		4482	354	991	540											
购房观念的讨论引	2025-02-06 14:34:54	changxiulan	限购政策, 二手房市, 天津市			7125	444	1691	328											
关于购房税费的政	2025-02-10 18:56:07	fyu	房产证办理, 购房支, 宁夏回族自治区	汕尾		1428	4471	1197	789											
如何应对购房福利	2025-02-14 23:46:38	duguiying	首付比例, 限购政策, 山东省	呼和浩特		9081	1121	709	0											
随着房地产调控的	2025-01-27 23:59:27	flai	房地产调控, 房屋质, 甘肃省			1787	4647	1096	238											
购房意愿的问题在	2025-02-15 06:18:35	zhougang	购房福利, 贷款利率, 湖南省	武汉		5403	3545	913	848											
购房税费的问题在	2025-02-15 10:44:49	xiuyingwang	购房年龄, 贷款利率, 澳门特别行政区	西安		5740	693	1886	226											
随着房屋维修的实	2025-02-06 04:54:54	wuxia	购房年龄, 限购政策, 浙江省	深圳		9121	224	938	709											
关于购房意愿的政	2025-02-09 08:11:16	qinxiuying	房屋供应, 房产证办, 重庆市	济南		4090	2305	1182	869											
购房观念的讨论引	2025-02-13 08:09:32	liutao	购房意愿, 房屋质量, 台湾省	西安		8356	2061	1221	161											
专家指出, 限购政	2025-02-24 14:09:00	xhe	购房补贴, 购房观念, 吉林省	澳门		8368	1685	12	495											
关于房屋供应的政	2025-02-22 15:58:48	dufang	房地产调控, 物业管理, 江西省	大冶		4443	3462	1355	183											
随着购房年龄的推	2025-02-08 05:05:20	xiaocui	购房补贴, 限购政策, 江西省	阜新		2909	2636	327	770											
随着房价上涨的推	2025-01-29 00:51:00	guiyinqiu	购房福利, 房屋质量, 海南省	柳州		5744	3934	661	96											

Sheet1

2243.203704 计数: 259 求和: 242266

20:14 2025/2/26

1859



Sheet1



一句话生成代码

```
import pandas as pd
import numpy as np
from faker import Faker
import random
```

```
# Initialize Faker
fake = Faker('zh_CN')
```

```
# Function to generate random data related to housing purchase intention topics
def generate_housing_data(num_records):
```

```
    housing_topics = [
        "购房意愿", "房价上涨", "购房政策", "贷款利率", "首付比例",
        "房屋供应", "二手房市场", "房地产调控", "购房补贴", "限购政策",
        "房屋质量", "物业管理", "购房税费", "房产证办理", "购房资格",
        "购房支持", "购房福利", "房屋维修", "购房年龄", "购房观念"
    ]
```

```
    summaries = [
        "随着{}的推行, 许多人对其可能带来的影响表示担忧。",
        "{}的讨论引发了广泛关注, 特别是在当今社会。",
        "很多家庭在考虑{}时面临诸多挑战和选择。",
        "{}成为当前社会的热点话题, 引起了不同观点的争论。",
        "专家指出, {}对社会发展的影响不可忽视。",
        "关于{}的政策调整, 公众有不同的反应和意见。",
        "{}引发了关于家庭和工作平衡的新一轮讨论。",
        "随着{}的实施, 社会各界对其效果表示关注。",
        "{}的问题在现代社会中日益突出。",
        "如何应对{}带来的压力和挑战, 成为了热门讨论话题。"
    ]
```

```
data = []
for _ in range(num_records):
    topic = random.choice(housing_topics)
    summary_template = random.choice(summaries)
    summary = summary_template.format(topic)
    timestamp = fake.date_time_between(start_date='-30d', end_date='now').strftime('%Y-%m-%d %H:%M:%S')
    username = fake.user_name()
    keywords = ', '.join(random.sample(housing_topics, 5))
    publish_location = fake.province()
    mention_location = fake.city_name() if random.random() > 0.5 else np.nan
    read_count = fake.random_int(min=0, max=10000)
    like_count = fake.random_int(min=0, max=5000)
    comment_count = fake.random_int(min=0, max=2000)
    share_count = fake.random_int(min=0, max=1000)
```

```
    data.append([
        summary, timestamp, username, keywords, publish_location,
        mention_location, read_count, like_count, comment_count, share_count
    ])
return data
```

```
# Generate 1900 rows of data related to housing purchase intention topics
num_records = 1900
columns = ['摘要', '发布时间', '发布人', '主题词', '发布地区', '提及地区', '阅读数', '点赞数', '评论数', '转发数']
random_housing_data = generate_housing_data(num_records)
```

```
# Create DataFrame
df_housing = pd.DataFrame(random_housing_data, columns=columns)
```

```
# Save to Excel
output_housing_path = '居民购房意向微博数据模拟.xlsx'
df_housing.to_excel(output_housing_path, index=False)
```

```
print(output_housing_path)
```

创新的思路：保留表头 限定生成



将微博数据的内容清空，保留变量表头，并和大模型进行对话（提示词#请你分析文件变量，并模拟人类社交媒体交流，以女性生育、生育意愿为主题，进行数据模拟），大模型可以直接生成数据（或是提供代码在PyCharm等软件中本地生成）。通过不同组提示词的比较，研究发现，明确给出要求并且给定关键主题的提示词的效果更好，它能更准确地模拟出符合人类行为和语言风格的内容，生成文本的语义丰富性和情感色彩更为鲜明。共获得1.6万条数据，其中微博8067条（示例见表1），大模型生成8065条（随机示例见下表），其中，大模型往往在生成时受到长度限制的约束，以提高生成效率和保证生成内容的可读性。因此，相较于微博数据，大模型生成的文本较为简短、直接。

摘要	发布时间	发布人	主题词	地区
一季度结婚登记同比减少17.8万对！专家称结婚人数下降是总体趋势...	2024/5/20 0:25:40	新浪新闻	人数，结婚，登记，同比，经济	北京
如果在当下国内推行，反而会在姓氏传承的动机方面极大打击男性的生育意愿。	2024/05/20 00:41:38	青岛困难群众01	姓氏，男性，女性，家族	上海
工作育儿理想#广东饮料企业为生娃操碎了心#现在年轻人不愿意结婚，不愿意生娃，与女性地位的提高、婚姻观念的改变等，也可能对年轻人的生育意愿产生影响。	2024/6/17 8:35:05	自然医学专属为你	年轻人，问题，观念，广东	广东
社会财富分配的不均也带来一系列的社会问题，如人口出生率下降，在众多国家中韩国最为明显，社会低收入群体生育意愿大幅降低。	2024/5/20 12:53:33	奋斗的小虫001	社会，字长，微博，法则	雅安
年轻夫妇普遍面临着高房价、教育负担重等问题，生育意愿不足成为制约因素。	2024/5/20 13:29:01	老索有好股	人口，社会，中国，老年人	青岛

摘要	发布时间	发布人	主题词	地区
二胎政策的讨论引发了广泛关注，特别是在当今社会。	2024-05-19 17:28:12	xxue	女性生育意愿,生育权利,生育率下降,生育支持,生育年龄	沈阳
育儿支出引发了关于家庭和工作平衡的新一轮讨论。	2024-05-19 17:43:08	xiasun	生育政策,不孕不育治疗,生育医疗,生育支持,二胎政策	潮州
很多家庭在考虑生育福利时面临诸多挑战和选择。	2024-05-19 17:47:35	renguiying	不孕不育治疗,生育支持,育儿教育,女性生育意愿,育儿假期	北京
如何应对女性生育意愿带来的压力和挑战，成为了热门讨论话题。	2024-05-19 17:51:20	btang	育儿支出,生育年龄,育儿成本,生育政策,育儿假期	天津
生育支持引发了关于家庭和工作平衡的新一轮讨论。	2024-05-19 18:19:53	eyang	生育率下降,育儿支出,生育政策,生育权利,育儿假期	武汉

微博和AI数据集关键词Top20

微博		AI	
关键词	频数	关键词	频数
生育	14957	生育	4419
意愿	9396	社会	2403
家庭	8295	讨论	2385
孩子	6431	挑战	1636
政策	6248	不同	1636
社会	5716	政策	1615
女性	5329	家庭	1615
研究	4852	成为	1609
人口	4591	关于	1604
中国	3725	育儿	1590
生育率	3608	关注	1589
发展	2801	引发	1586
提到	2729	带来	1567
普遍	2592	随着	1549
全球	2438	表示	1549
之后	2405	影响	1546
现象	2396	很多	837
中心	2390	考虑	837
当代	2377	面临	837
性别	2341	诸多	837

微博和AI数据集关键词Top20



从微博和大模型生成的文本数据中提取了总共十个主题。这些主题涵盖了关于生育意愿的广泛讨论，涉及多个维度。

大模型能够在一定程度上捕捉社交媒体用户讨论的核心话题，具备生成符合真实语义场的能力。然而，大模型的生成内容在表达的细节和多样性方面明显低于真实用户生成的文本。这种局限性反映了大模型在应对复杂社会话题时的局部化不足：它可以构建出符合语义逻辑的讨论，但缺乏在社交语境中处理细微文化差异的能力。

微博主题	微博数据关键词及权重	模型主题	AI数据关键词及权重
生育政策	关注(0.148), 挑战(0.148), 政策(0.143), 带来(0.138), 指出(0.083), 权利(0.049), 家庭计划(0.041), 晚婚晚育(0.041), 生育率(0.040), 福利(0.036)	社会政策与个人主义	政策(0.086), 社会(0.050), 个人主义(0.045), 孩子(0.044), 胡适(0.043), 视角(0.041), 合著(0.041), 抬头(0.041), 复旦大学(0.041), 现象(0.040)
生育发展趋势	发展(0.106), 专家(0.106), 面临(0.101), 日益突出(0.098), 新一轮(0.089), 话题(0.089), 实施(0.081), 热点话题(0.080), 假期(0.054), 健康(0.052)	人口与生育问题	人口(0.089), 主任(0.051), 不育(0.044), 程度(0.014), 人群(0.014), 影响(0.013), 一种(0.009), 自愿(0.009), 改变(0.008), 估计(0.008)
家庭影响	引发(0.146), 影响(0.146), 家庭(0.142), 意见(0.076), 调整(0.076), 选择(0.075), 诸多(0.075), 平衡(0.067), 应对(0.066), 年龄(0.038)	全球性别观念	提到(0.086), 性别(0.072), 全球(0.072), 一个(0.048), 选择(0.030), 世纪(0.023), 依然(0.023), 观念(0.022), 教育(0.017), 提高(0.017)
社会育儿压力	生育(0.289), 社会(0.159), 育儿(0.104), 特别(0.059), 当今社会(0.059), 很多(0.056), 压力(0.050), 担忧(0.049), 效果(0.047), 社会各界(0.047)	中国家庭与婚姻研究	研究(0.138), 家庭(0.118), 中国(0.072), 低迷(0.070), 教授(0.070), 趋于(0.068), 婚姻(0.022), 传统(0.018), 年代(0.012), 持续(0.011)
津贴等话题	讨论(0.285), 忽视(0.107), 公众(0.102), 工作(0.090), 热门(0.089), 推行(0.088), 观点(0.081), 争论(0.081), 津贴(0.040), 不孕(0.036)	生育意愿与生育率	生育(0.250), 意愿(0.142), 发展(0.079), 生育率(0.077), 中心(0.075), 当代(0.071), 年轻人(0.044), 经济(0.031), 结婚(0.013), 养育(0.013)

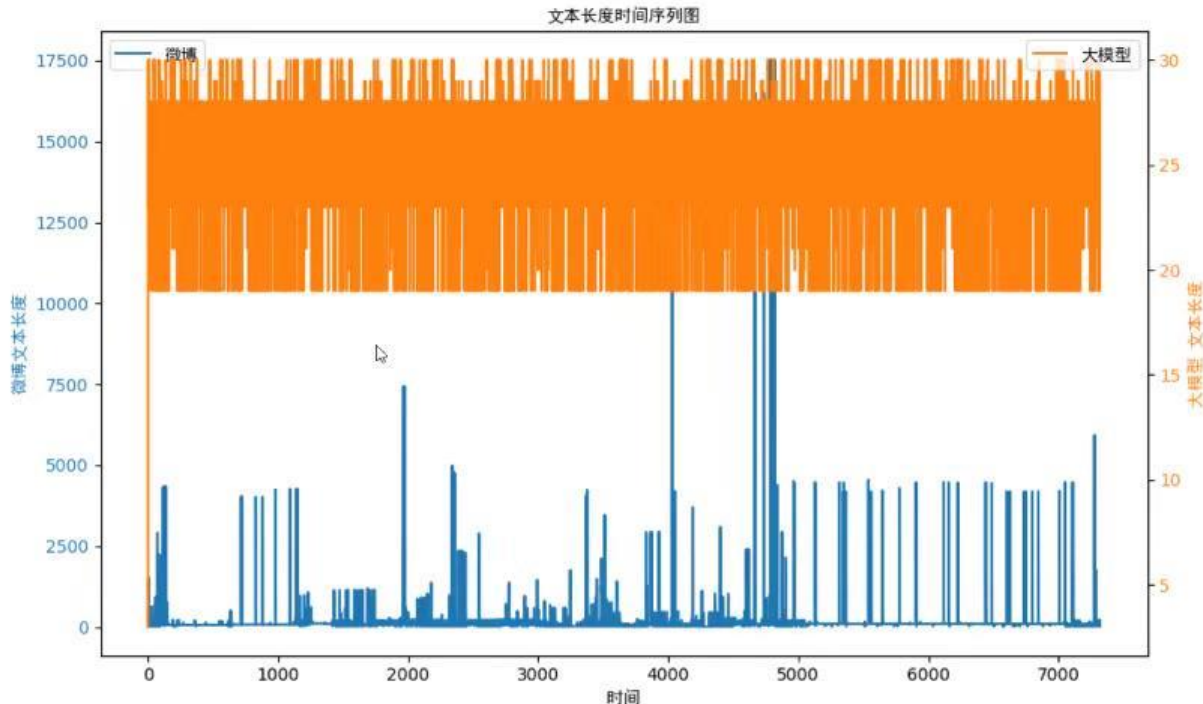
代码进行时间序列分析

为分析两个数据集内容的一致性及时序动态特征。右图展示了微博数据和大模型生成数据的文本长度在时间上的变化情况。

左侧纵轴表示微博文本长度，右侧纵轴表示AI大模型文本长度，时间沿横轴展开，横轴上具体的数字代表每条token的序号（不代表时间刻度）。

可以明显看出，大模型生成文本长度波动较小，而微博文本长度变化幅度较大，显示出显著的高峰和低谷。大模型生成的文本在同一时间段内较为一致且长度较短，而微博数据则因用户不同而呈现大的波动性，字数不定。

延伸来说，这种局限性表明大语言模型在生成文本时，缺乏真实社交媒体用户的多样化行为特征，因此生成的内容长度较为稳定而缺乏变化，这与社交媒体上用户生成内容的显著波动性不一致。



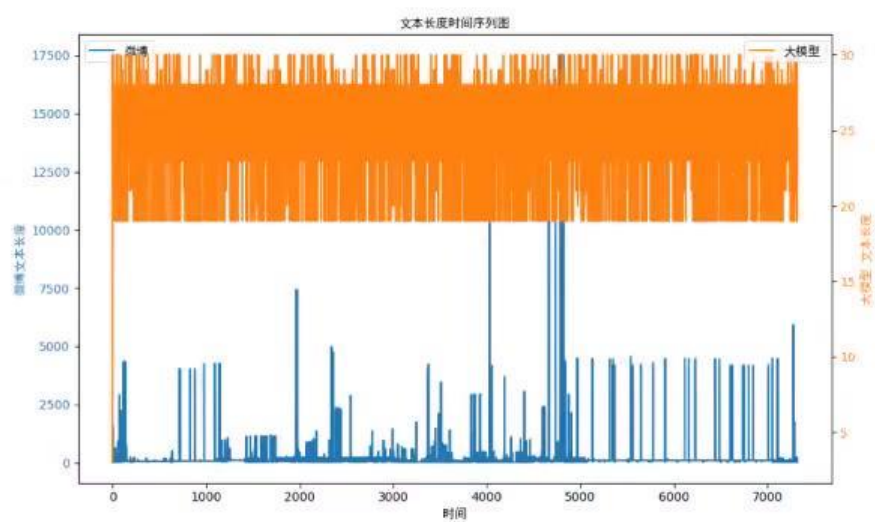
代码进行时间序列分析

为分析两个数据集内容的一致性及时动态特征。右图展示了微博数据和大模型生成数据的文本长度在时间上的变化情况。

左侧纵轴表示微博文本长度，右侧纵轴表示AI大模型文本长度，时间沿横轴展开，横轴上具体的数字代表每条token的序号（不代表时间刻度）。

可以明显看出，大模型生成文本长度波动较小，而微博文本长度变化幅度较大，显示出显著的高峰和低谷。大模型生成的文本在同一时间段内较为一致且长度较短，而微博数据则因用户不同而呈现大的波动性，字数不定。

延伸来说，这种局限性表明大语言模型在生成文本时，缺乏真实社交媒体用户的多样化行为特征，因此生成的内容长度较为稳定而缺乏变化，这与社交媒体上用户生成内容的显著波动性不一致。



47 一句话生成代码

48 国家高新区·保留表头 描述生成

49 微博和AI数据集关键词Top20

50 微博和AI数据集关键词Top20

51 代码进行时间序列分析

Python分析《三国演义》诸葛亮发言

```
<!-- 美食卡片1 -->
<a href="#detail" style="text-decoration: none; color: inherit;">
  <div class="food-card">
    
    <div class="food-info">
      <h3 class="food-title">招牌牛肉面</h3>
      <div class="food-rating">★★★★☆ 4.5分 (368条评论)</div>
      <div class="food-price">人均消费: ¥35</div>
    </div>
  </div>
</a>
```

```
<!-- 美食卡片2 -->
<a href="#" detail="" style="text-decoration: none; color: inherit;">
  <div class="food-card">
    
    <div class="food-info">
      <h3 class="food-title">秘制烤鸭</h3>
      <div class="food-rating">★★★★★ 4.8分 (452条评价)</div>
      <div class="food-price">人均消费: ¥88</div>
    </div>
  </div>
</a>
```

[给 DeepSeek 发送消息](#)

深度思考 (R1) 联网搜索

内容由 AI 生成，请仔细甄别

开启新对话

今天

Python分析《三国演义》清...

昨天

生成式AI内容生产使用研究

唐雀翎花枝明憶千年風華

7 天内

Python数据分析与可视化常用库

30 天内

DeepSeek本地部署版本及配置要求

人机优生与快生AI实践风格

CoT链式思维核心要点解析

洪一梅探讨模型在模拟人类社会行

这种差异可能影响知识的准确性与

人工智能赋能智库研究 给我从44

人机共生理论下生成式AI的知识生

研究目标二：效果验证——促优生

下載 App **NEW**

个人信息

Python分析《三国演义》诸葛亮发言

```
import spacy
import networkx as nx
import matplotlib.pyplot as plt
import re
import jieba

# 加载中文 NLP 模型
nlp = spacy.load('zh_core_web_sm')
nlp.max_length = 1000000 # 设置最大长度为1,000,000字符

# 从文件读取文本
with open('C:/Users/wtao4/OneDrive/桌面/input.txt', 'r', encoding='utf-8') as file:
    text = file.read()

# 检查文本长度
if len(text) > nlp.max_length:
    print(f"Warning: text length ({len(text)}) exceeds nlp.max_length ({nlp.max_length})")

# 使用 spaCy 处理文本，移除标点符号
text = re.sub(r'[^\w\s]', '', text)

# 使用jieba进行分词处理
text = ' '.join(jieba.cut(text))

doc = nlp(text)
```

给 DeepSeek 发送消息

深度思考 (R1)

内容由 AI 生成，请仔细甄别

文件(F) 编辑(E) 视图(V) 导航(N) 代码(C) 运行(R) 工具(T) VCS(S) 窗口(W) 帮助(H)

pythonProject1 腾讯会议 腾讯会议 腾讯会议

main.py

Google_knowledge_graph.png
input.txt
kmo_bartlett_results.xlsx
lda_topics.txt
LICENSE.chromedriver
main.py
news_links.csv
news_links.txt
nmf_topics.txt
regression_report.xlsx
science_communication_survey
simulated_survey_data.xlsx
time_series_plot.png
~\$居民消费价格指数数据模拟.xlsx
实验流程与关键方法.xlsx
居民消费价格指数数据模拟.xlsx
生成的真实答案问题.csv
生成的真实答案问题11111.csv
相似度分析_随机森林_遗传编程.gr
相似度预测_随机森林.png
科技评测数据.xlsx
统计分析报告.xlsx
论文结构图
聚类率分类_随机森林_遗传编程.pr

```
1 # 初始化问题集合
2 questions = set()
3
4 # 定义一个函数从数据集中提取问题
5 def add_questions_from_dataset(dataset_name, config_name=None, split_name='train', question_column='question'):
6     global questions
7     try:
8         # 加载数据集，配置名称是可选的
9         if config_name:
10             dataset = load_dataset(dataset_name, config_name, split=split_name)
11         else:
12             dataset = load_dataset(dataset_name, split=split_name)
13
14         # 提取问题并去重
15         new_questions = set(dataset[question_column])
16         # 将新问题加入集合
17         questions.update(new_questions)
18         print(f"从数据集 {dataset_name} ({config_name}) 中添加了 {len(new_questions)} 个问题，总共 {len(questions)} 个问题。")
19     except Exception as e:
20         print(f"无法从数据集 {dataset_name} ({config_name}) 加载数据集提取问题: {str(e)}")
21
22 add_questions_from_dataset('...')
23
24 try:
```

运行 Python 控制台或测试器以查看可用数据

运行

main

```
C:\Users\wtao4\PycharmProjects\pythonProject1\venv\Scripts\python.exe C:/Users/wtao4/PycharmProjects/pythonProject1/main.py
Using the latest cached version of the dataset since squad couldn't be found on the Hugging Face Hub
Found the latest cached dataset configuration 'plain_text' at C:\Users\wtao4\cache\huggingface\datasets\squad\plain_text\0.0.0\7b6d24c440a36b6815f21b70d2501c
从数据集 squad (None) 中添加了 87355 个问题，总共 87355 个问题。
Using the latest cached version of the dataset since squad couldn't be found on the Hugging Face Hub
Found the latest cached dataset configuration 'plain_text' at C:\Users\wtao4\cache\huggingface\datasets\squad\plain_text\0.0.0\7b6d24c440a36b6815f21b70d2501c
从数据集 squad (None) 中添加了 10539 个问题，总共 97888 个问题。
问题数量不足100万个，目前共 97888 个问题。请考虑添加更多数据集。
已生成100万个问题并保存到生成的真实答案问题.csv

进程已结束，退出代码0
```

Version Control 运行 TODO 问题 终端 Python Packages Python 控制台

4°C

陶伟的共享屏幕

文档生等基础AI编程.pptx - W...

您似乎在使用 NumPy
请尝试打开科学模式?
使用科学模式，保持当前库...

11:25 CRLF UTF-8 4 20:16 2023/2/26

- 47 一句话生成代码
- 48 观察数据集：保留表头 描述生成
- 49 微博和AI数据集关键词Top20
- 50 微博和AI数据集关键词Top20
- 51 代码进行时间序列分析

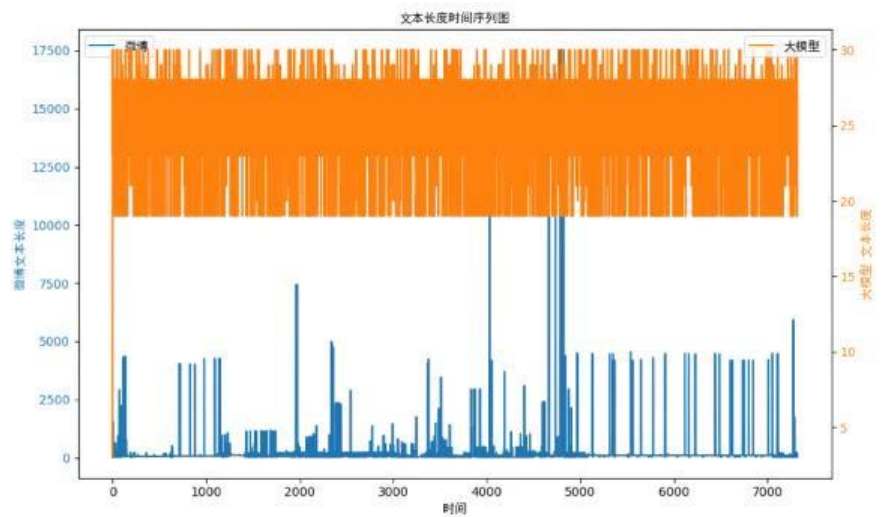
代码进行时间序列分析

为分析两个数据集内容的一致性及时动态特征。右图展示了微博数据和AI大模型生成数据的文本长度在时间上的变化情况。

左侧纵轴表示微博文本长度，右侧纵轴表示AI大模型文本长度，时间沿横轴展开，横轴上具体的数字代表每条token的序号（不代表时间刻度）。

可以明显看出，大模型生成文本长度波动较小，而微博文本长度变化幅度较大，显示出显著的高峰和低谷。大模型生成的文本在同一时间段内较为一致且长度较短，而微博数据则因用户不同而呈现大的波动性，字数不定。

延伸来说，这种局限性表明大语言模型在生成文本时，缺乏真实社交媒体用户的多样化行为特征，因此生成的内容长度较为稳定而缺乏变化，这与社交媒体上用户生成内容的显著波动性不一致。



VAR ARIMA I

深度思考 (R1) 联网搜索

内容由 AI 生成，请仔细甄别

pythonProject1

腾讯会议

main.py

```
# 初始化问题集
questions = set()

# 定义一个函数从数据集中提取问题
def add_questions_from_dataset(dataset_name, config_name=None, split_name='train', question_column='question'):
    global questions
    try:
        # 加载数据集，配置名称是可选的
        if config_name:
            dataset = load_dataset(dataset_name, config_name, split=split_name)
        else:
            dataset = load_dataset(dataset_name, split=split_name)

        # 提取问题并去重
        new_questions = set(dataset[question_column])
        # 将新问题加入集合
        questions.update(new_questions)
        print(f"从数据集 {dataset_name} ({config_name}) 中添加了 {len(new_questions)} 个问题，总共 {len(questions)} 个问题。")
    except Exception as e:
        print(f"无法从数据集 {dataset_name} ({config_name}) 加载数据集或提取问题: {str(e)}")

add_questions_from_dataset('squad', 'plain_text')
```

运行

main.py

C:\Users\wtiao4\PycharmProjects\pythonProject1\venv\Scripts\python.exe C:/Users/wtiao4/PycharmProjects/pythonProject1/main.py
Using the latest cached version of the dataset since squad couldn't be found on the Hugging Face Hub
Found the latest cached dataset configuration 'plain_text' at C:\Users\wtiao4\cache\huggingface\datasets\squad\plain_text\0.0.0\7b6d24c440a36b0815f21b70d2501c
从数据集 squad (None) 中添加了 87355 个问题，总共 87355 个问题。
Using the latest cached version of the dataset since squad couldn't be found on the Hugging Face Hub
Found the latest cached dataset configuration 'plain_text' at C:\Users\wtiao4\cache\huggingface\datasets\squad\plain_text\0.0.0\7b6d24c440a36b0815f21b70d2501c
从数据集 squad (None) 中添加了 10539 个问题，总共 97888 个问题。
问题数量不足100万个，目前共 97888 个问题。请考虑添加更多数据集。
已生成100万个问题并保存到生成的真实答案问题.csv
进程已结束，退出代码0

您似乎在使用 NumPy

请尝试打开科学模式？

使用科学模式，某些函数将更有意义...

11:25 CRLF UTF-8 4.0

20:17 2025/2/26

 以下是

🔗 以下是使用 Python 进行时间序列分析的代码，包括 VAR 和 ARIMA 模型的实现。假设你的 Excel 文件的第一列是时间序列数据，我们将读取数据并进行时间序列分析。

python

开启新对话

[给 DeepSeek 发送消息](#)

深度思考 (供1) 联网搜索

內容由 AI 生成，請仔細核對

deepseek

开启新对话

今天

Python分析《三国演义》诸葛亮发言

昨天

生成式AI内容生产使用研究

唐蜜杨花枝铜镜千年风华

7天内

Python数据分析与可视化常用库

30天内

DeepSeek本地部署版本及配置要

人机共生与快生AI实践风格

CoT链式思维核心特点解析

进一步探讨模型在模拟人类社会行

这种差异可能影响知识的准确性与

人工智能赋能智库研究，给我从4个

人机共生理论下生成式AI的知识生

研究目标一：效果测评——评估生

下载 App

个人信息

Python分析《三国演义》诸葛亮发言

代码实现

```
pythonimport pandas as pdimport numpy as npimport matplotlib.pyplot as pltfrom statsmodels.tsa.api import VARfrom statsmodels.tsa.arima.model import ARIMAfrom statsmodels.tsa.stattools import adfullerfrom statsmodels.graphics.tsaplots import plot_acf, plot_pacf# 1. 读取数据file_path = r"C:\Users\wtap4\OneDrive\桌面\data.xlsx" # 替换为你的文件路径try:    df = pd.read_excel(file_path)    print("数据读取成功!")    print(df.head()) # 查看前几行数据except Exception as e:    print("读取文件时出错:", e)    exit()# 2. 数据预处理# 假设第一列是时间序列数据time_series = df.iloc[:, 0].dropna() # 忽略第一列并去除缺失值time_series.index = pd.to_datetime(time_
```

开启新对话

数据集 squad (None) 中添加了 10539 个问题，总共 97888 个问题。
问题数量不足100万个，目前共 97888 个问题。请考虑添加更多数据集。
已生成100万个问题并保存到生成的真实答案问题.csv

深度思考 (R1) 联网搜索

内容由 AI 生成，请仔细甄别

20:18
2025/2/26

陶伟的共享屏幕

DeepSeek - 探索未知之境

论坛置顶_百度文库

说文解字plus

+

chat.deepseek.com/a/chat/s/26244002-e389-43c9-b595-b7702d8f920d

书签手机书签character.ai香港中文大学新闻...DeepSeek - 探索...登录页面_问卷星SiliconCloud小木虫论坛_学术科...可灵 AI - 新一代 AI...资源访问控制系统...智灵动力元美AI设计平台虚拟人生力平台腾讯会议

deepseek

开启新对话

今天

Python分析《三国演义》诸...

...

昨天

生成式AI内容生产使用研究

唐蕃桃花枝铜镜千年风华

7天内

Python数据分析与可视化常用库

30天内

DeepSeek本地部署版本及配置要...

人机共生与共生AI实践风格

CoT链式思维核心特点解析

进一步探讨模型在模拟人类社会行...

这种差异可能影响知识的准确性与...

人工智能赋能智库研究,给我从4个...

人机共生理论下生成式AI的知识生...

研究目标一:效果测评——评估生...

下载 AppNEW

个人信息

Python分析《三国演义》诸葛亮发言

我要做时间序列分析,要用到VAR ARIMA分析技巧,我本地有1个excel文件,文件的第一列是我的要分析的内容。请你给我python代码。

以下是使用 Python 进行时间序列分析的代码,包括 VAR 和 ARIMA 模型的实现。假设你的 Excel 文件的第一列是时间序列数据,我们将读取数据并进行时间序列分析。

代码实现

```
python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.api import VAR
from statsmodels.tsa.arima.model import ARIMA
from statsmodels.tsa.stattools import adfuller
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf

# 1. 读取数据
file_path = r"C:\Users\wtap4\OneDrive\桌面\data.xlsx" # 替换为你的文件路径
try:
    df = pd.read_excel(file_path)
    print("数据读取成功!")
    print(df.head()) # 查看前几行数据
except Exception as e:
    print("读取文件时出错:", e)
    exit()

# 2. 数据预处理
# 假设第一列是时间序列数据
time_series = df.iloc[:, 0].dropna() # 将第一列并去除空值
time_series.index = pd.to_datetime(time_series.index) # 将索引转换为日期格式(如需要则写时间)
```

给 DeepSeek 发送消息

深度思考 (R1) 联网搜索

内容由 AI 生成, 请仔细甄别

20:18 2025/2/26

Python分析《三国演义》诸葛亮发言

```
python
import matplotlib.pyplot as plt
from statsmodels.tsa.api import VAR
from statsmodels.tsa.arima.model import ARIMA
from statsmodels.tsa.stattools import adfuller
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf

# 1. 读取数据
file_path = r"C:\Users\wtao4\OneDrive\桌面\data.xlsx" # 替换为你的文件路径
try:
    df = pd.read_excel(file_path)
    print("数据读取成功!")
    print(df.head()) # 查看前几行数据
except Exception as e:
    print("读取文件时出错:", e)
    exit()

# 2. 数据预处理
# 假设第一列是时间序列数据
time_series = df.iloc[:, 0].dropna() # 提取第一列并去除缺失值
```

给 DeepSeek 发送消息

深度思考 (R1)

联网搜索



内容由 AI 生成，请仔细甄别



Python分析《三国演义》诸葛亮发言

```
python

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.api import VAR
from statsmodels.tsa.arima.model import ARIMA
from statsmodels.tsa.stattools import adfuller
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf

# 1. 读取数据
file_path = r"C:\Users\wtao4\OneDrive\桌面\data.xlsx" # 替换为你的文件路径
try:
    df = pd.read_excel(file_path)
    print("数据读取成功!")
    print(df.head()) # 查看前几行数据
except Exception as e:
    print("读取文件时出错: ", e)
    exit()
```

给 DeepSeek 发送消息

深度思考 (R1) 联网搜索



内容由 AI 生成, 请仔细甄别



Python分析《三国演义》诸葛亮发言

```
python
import matplotlib.pyplot as plt
from statsmodels.tsa.api import VAR
from statsmodels.tsa.arima.model import ARIMA
from statsmodels.tsa.stattools import adfuller
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf

# 1. 读取数据
file_path = r"C:\Users\wtao4\OneDrive\桌面\data.xlsx" # 替换为你的文件路径
try:
    df = pd.read_excel(file_path)
    print("数据读取成功!")
    print(df.head()) # 查看前几行数据
except Exception as e:
    print("读取文件时出错: ", e)
    exit()

# 2. 数据预处理
# 假设第一列是时间序列数据
time_series = df.iloc[:, 0].dropna() # 提取第一列并去除缺失值
```

给 DeepSeek 发送消息

深度思考 (R1) 联网搜索



内容由 AI 生成, 请仔细甄别

Python分析《三国演义》诸葛亮发言

```
python
forecast_ci = forecast.conf_int()

# 可视化预测结果
plt.figure(figsize=(10, 6))
plt.plot(time_series, label="原始数据")
plt.plot(forecast_mean, label="预测值", color='red')
plt.fill_between(forecast_ci.index, forecast_ci.iloc[:, 0], forecast_ci.iloc[:, 1], color='pink', alpha=0.3)
plt.title("ARIMA 预测结果")
plt.legend()
plt.show()

# 4. VAR 模型 (适用于多变量时间序列)
# 如果有多个时间序列, 可以使用 VAR 模型
# 假设第二列是另一个时间序列
if df.shape[1] > 1:
    print("\n=== VAR 模型 ===")
    # 提取多变量时间序列
    multi_series = df.iloc[:, :2].dropna() # 假设前两列是时间序列
    multi_series.index = pd.to_datetime(multi_series.index)
```

给 DeepSeek 发送消息

深度思考 (R1)

联网搜索



内容由 AI 生成, 请仔细甄别

DeepSeek - 探索未知之境

SiliconCloud

论坛十则_百度文库

模拟平台

说文解字plus

+

chat.deepseek.com/s/26244002-e389-43c9-b595-b7702d8f920d

书签手机书签character.ai香港中文大学新闻...DeepSeek - 探索...登录网页_问卷星SiliconCloud小木虫论坛_学术科...可灵 AI - 新一代 AI...资源访问控制系统...智算动力元景AI设计100%

腾讯会议

Python分析《三国演义》诸葛亮发言

python

复制

```
# 拟合 VAR 模型
var_model = VAR(multi_series_diff)
var_results = var_model.fit(maxlags=5) # 最大滞后阶数为5
print(var_results.summary())

# 预测未来值
forecast_steps = 10
var_forecast = var_results.forecast(multi_series_diff.values[-var_results.k_ar:], steps=forecast_steps)
var_forecast_df = pd.DataFrame(var_forecast, columns=multi_series.columns)

# 可视化预测结果
plt.figure(figsize=(10, 6))
for col in multi_series.columns:
    plt.plot(multi_series_diff.index, multi_series_diff[col], label=f'{col} (差分后)')
    plt.plot(pd.date_range(multi_series_diff.index[-1], periods=forecast_steps + 1, freq='D')[1:],
             var_forecast_df[col], label=f'{col} 预测', linestyle='--')
plt.title("VAR 预测结果")
plt.legend()
plt.show()
else:
    print("\n数据只有一列，无法使用 VAR 模型。")
```

代码说明

1. 数据读取:

从 Excel 文件中读取数据，假设第一列是时间序列数据。

如果有多列数据，可以用于 VAR 模型分析。

2. 平稳性检验:

使用 ADF 检验检查数据是否平稳。

给 DeepSeek 发送消息

深度思考 (R1)

联网搜索

📎

+

内容由 AI 生成，请仔细甄别

20:19
2025/2/26

陶伟的共享屏幕



Python分析《三国演义》诸葛亮发言

```
python
q = 1 # MA 阶数 (根据 ACF 图选择)

# 拟合 ARIMA 模型
arima_model = ARIMA(time_series, order=(p, d, q))
arima_results = arima_model.fit()
print(arima_results.summary())

# 预测未来值
forecast_steps = 10 # 预测未来10个时间点
forecast = arima_results.get_forecast(steps=forecast_steps)
forecast_mean = forecast.predicted_mean
forecast_ci = forecast.conf_int()

# 可视化预测结果
plt.figure(figsize=(10, 6))
plt.plot(time_series, label="原始数据")
plt.plot(forecast_mean, label="预测值", color='red')
plt.fill_between(forecast_ci.index, forecast_ci.iloc[:, 0], forecast_ci.iloc[:, 1], color='pink', alpha=0.3)
plt.title("ARIMA 预测结果")
plt.legend()
plt.show()

# 4. VAR 模型 (适用于多变量时间序列)
# 如果有多个时间序列, 可以使用 VAR 模型
# 假设第二列是另一个时间序列
if df.shape[1] > 1:
    print("\n== VAR 模型 ==")
    # 提取多变量时间序列
    multi_series = df.iloc[:, :2].dropna() # 删除前两列是时间序列
    multi_series.index = pd.to_datetime(multi_series.index)

    # 检查平稳性
    for col in multi_series.columns:
        print(f"\n{col} 的平稳性检验: ")
```

给 DeepSeek 发送消息

深度思考 (R1) 联网搜索

内容山 AI 生成, 请仔细甄别

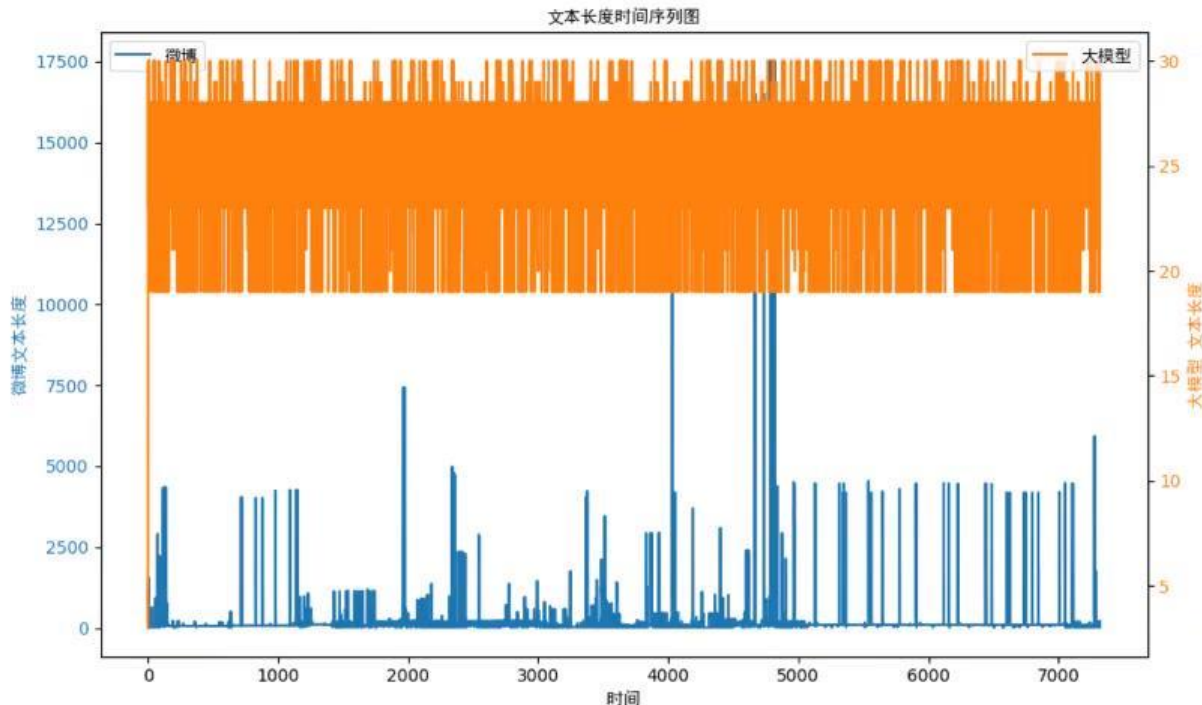
代码进行时间序列分析 (VAR ARIMA

为分析两个数据集内容的一致性及时序动态特征。右图展示了微博数据和大模型生成数据的文本长度在时间上的变化情况。

左侧纵轴表示微博文本长度，右侧纵轴表示AI大模型文本长度，时间沿横轴展开，横轴上具体的数字代表每条token的序号（不代表时间刻度）。

可以明显看出，大模型生成文本长度波动较小，而微博文本长度变化幅度较大，显示出显著的高峰和低谷。大模型生成的文本在同一时间段内较为一致且长度较短，而微博数据则因用户不同而呈现大的波动性，字数不定。

延伸来说，这种局限性表明大语言模型在生成文本时，缺乏真实社交媒体用户的多样化行为特征，因此生成的内容长度较为稳定而缺乏变化，这与社交媒体上用户生成内容的显著波动性不一致。



三

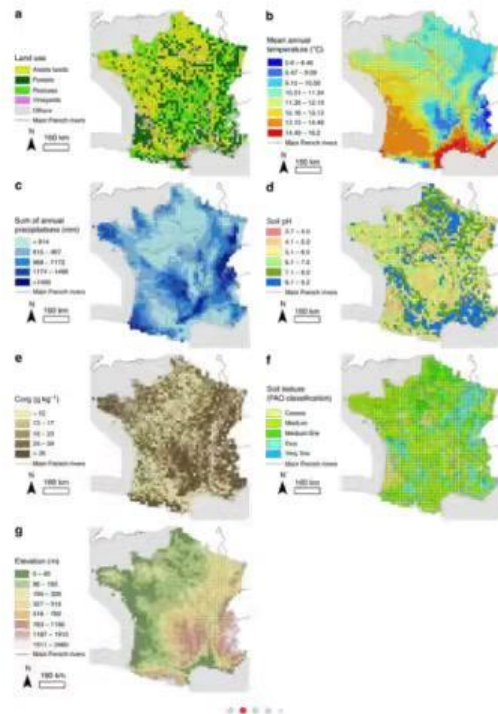
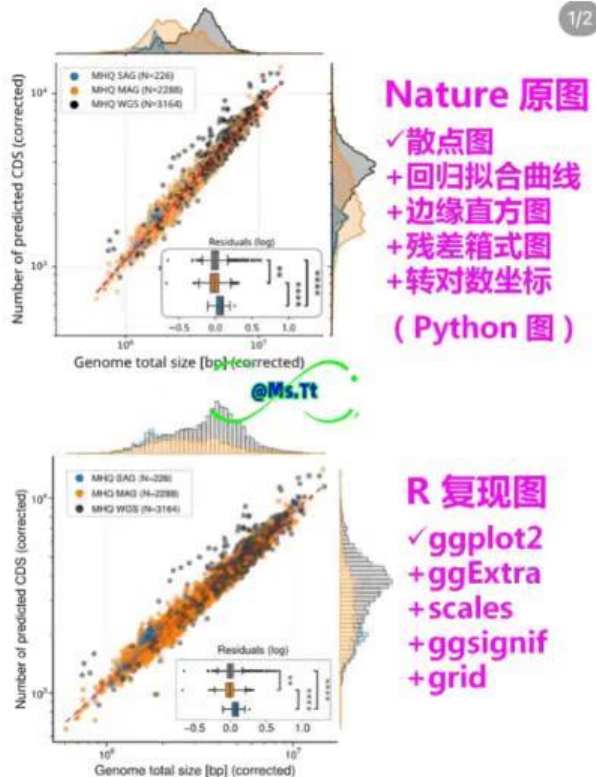
文生图表

从方法开始、从图片可视化开始

一张好图胜过千言万语。

在合适的情况下，不妨从数据可视化入手，思考自己的主题和预设的结论
用哪种方法+数据图表示出来会更好？
然后反推研究过程。

Prompt: 我想要做右图这样的图片，
这是什么数据图？如何用代码或现有
免费软件实现，达到基本一致。

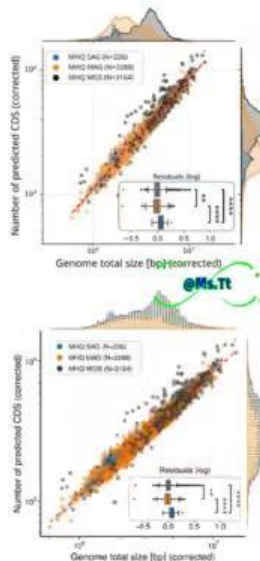


从方法开始、从图片可视化开始

一张好图胜过千言万语。

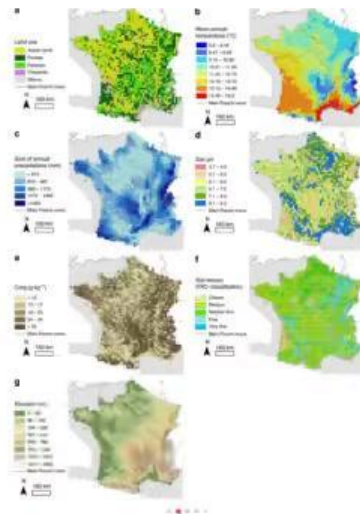
在合适的情况下，不妨从数据可视化入手，思考自己的主题和预设的结论用哪种方法+数据图表示出来会更好？然后反推研究过程。

Prompt: 我想要做右图这样的图片，这是什么数据图？如何用代码或现有免费软件实现，达到基本一致。



Nature 原图
✓散点图
+回归拟合曲线
+边缘直方图
+残差箱式图
+转对数坐标
(Python 图)

R 复现图
✓ggplot2
+ggExtra
+scales
+ggsignif
+grid





Python分析《三国演义》诸葛亮发言

```
python
plt.legend()
plt.show()

# 4. VAR 模型 (适用于连续时间序列)
# 如果有多个时间序列, 可以使用 VAR 模型
# 假设第二列是另一个时间序列
if df.shape[1] > 1:
    print("\n=== VAR 模型 ===")
    # 提取多变量时间序列
    multi_series = df.iloc[:, :2].dropna() # 删除前两列是时间序列
    multi_series.index = pd.to_datetime(multi_series.index)

    # 检查平稳性
    for col in multi_series.columns:
        print(f"\n{col} 的平稳性检验: ")
        check_stationarity(multi_series[col])

    # 如果数据非平稳, 进行差分
    multi_series_diff = multi_series.diff().dropna()

    # 拟合 VAR 模型
    var_model = VAR(multi_series_diff)
    var_results = var_model.fit(maxlags=5) # 最大滞后阶数为5
    print(var_results.summary())

    # 作图来预测
    forecast_steps = 10
    var_forecast = var_results.forecast(multi_series_diff.values[-var_results.k_ar:], steps=forecast_steps)
    var_forecast_df = pd.DataFrame(var_forecast, columns=multi_series.columns)

    # 可视化预测结果
    plt.figure(figsize=(10, 6))
    for col in multi_series.columns:
        plt.plot(multi_series_diff.index, multi_series_diff[col], label=f'{col} (差分后)')
```

给 DeepSeek 发送消息

深度思考 (R1)

联网搜索

内容由 AI 生成, 请仔细甄别

Python分析《三国演义》诸葛亮发言

```
python
# 预测未来值
forecast_steps = 10
var_forecast = var_results.forecast(multi_series_diff.values[-var_results.k_ar:], steps=forecast_steps)
var_forecast_df = pd.DataFrame(var_forecast, columns=multi_series.columns)

# 可视化预测结果
plt.figure(figsize=(10, 6))
for col in multi_series.columns:
    plt.plot(multi_series_diff.index, multi_series_diff[col], label=f'{col} (差分后)')
    plt.plot(pd.date_range(multi_series_diff.index[-1], periods=forecast_steps + 1, freq='D')[1:],
             var_forecast_df[col], label=f'{col} 预测', linestyle='--')
plt.title('VAR 预测结果')
plt.legend()
plt.show()
else:
    print("\n数据只有一列，无法使用 VAR 模型。")
```

代码说明

1. 数据读取:

- 从 Excel 文件中读取数据。假设第一列是时间序列数据。
- 如果有多列数据，可以用于 VAR 模型分析。

2. 平稳性检验:

- 使用 ADF 检验检查数据是否平稳。
- 如果数据非平稳，进行差分处理。

3. ARIMA 模型:

- 通过 ACF 和 PACF 图确定 ARIMA 参数 (p, d, q) 。

向 DeepSeek 发送消息

深度思考 (R1)

联网搜索

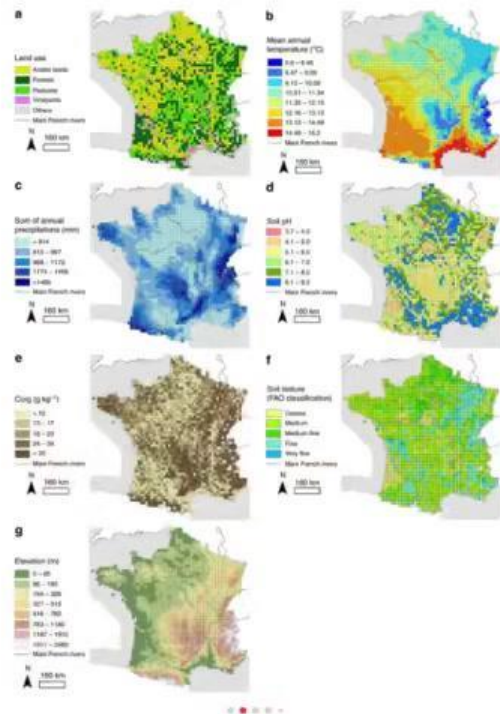
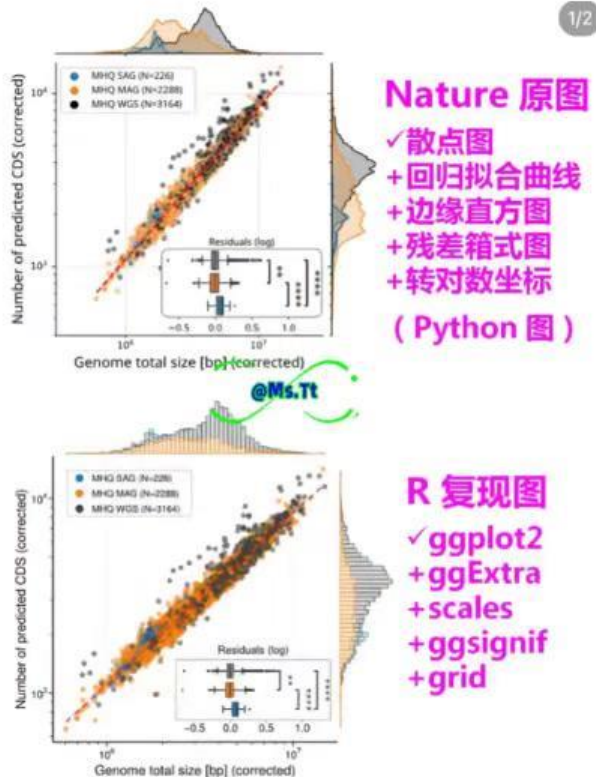
内容由 AI 生成，请仔细甄别

从方法开始、从图片可视化开始

一张好图胜过千言万语。

在合适的情况下，不妨从数据可视化入手，思考自己的主题和预设的结论
用哪种方法+数据图表示出来会更好？
然后反推研究过程。

Prompt: 我想要做右图这样的图片，
这是什么数据图？如何用代码或现有
免费软件实现，达到基本一致。



DeepSeek 应用场景



数据可视化

提示词结构为：**用 [语言] 生成 [图表类型]，数据包含 [字段]，呈现 [模式]。**

例如，“用 Python 生成饼图，数据包含水果的种类和销量，呈现不同水果销量占比模式”，可自动生成社会网络关系图，清晰展示人物或节点之间的关系。

数据可视化

任务

基于titanic遇难者数据分析结果绘制可视化图表

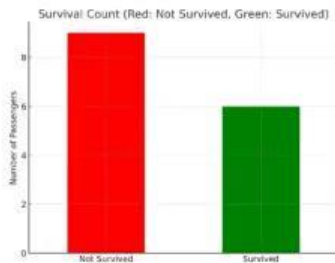
结论

- Open AI o3mini的数据可视化能力突出，能够直接高效地生成多种类型可视化图表，准确度高；
- DeepSeek R1、Kimi k1.5均能基于分析结果提供多种可视化图表绘制方案，但都需要依靠运行Python代码才能完成绘图任务，部分代码会出现错误

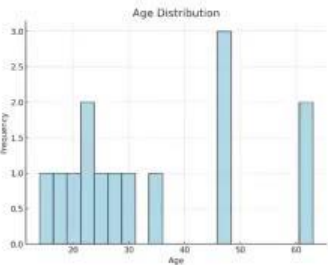
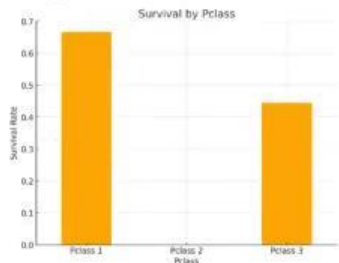
腾讯会议
请根据以上分析结果和数据样本绘制对应的可视化图表

Open AI o3mini

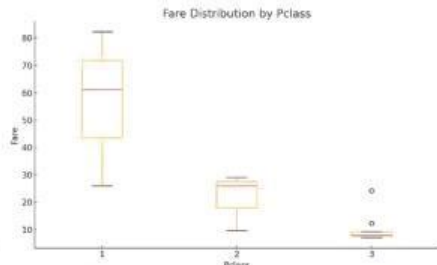
能够直接调用DALLE，根据分析结果和任务需求高效绘制各类可视化图表，部分较为复杂的图表可能出现数据错误或无法生成的情况。



■ 柱状图（生还者和遇难者的比例、按船舱等级分类的生还情况）



■ 年龄分布直方图、票价分布箱线图（展示不同船票等级的票价分布）



DeepSeek R1

能够结合数据样本和分析结果，提供多种可视化图表绘制方案，但暂时不能直接绘制出可视图表，需要将对应的绘图代码复制到本地运行制作图表。

```
3. **绘制图表**:  
  
- **生存率按舱位**：用 'groupby' 按舱位分组，计算生存率，绘制柱状图。  
  
- **性别与生存率**：用交叉表 (crosstab) 统计性别与生存情况，绘制堆叠条形图。  
  
- **年龄分布**：分箱处理年龄，绘制生存与非生存的对比直方图。  
  
- **家庭规模**：分组计算生存率，绘制折线图或柱状图。  
  
- **票价与舱位**：用箱线图展示不同舱位的票价分布。  
  
- **登船港口**：类似舱位的处理方式，分组计算生存率。
```

5. 票价与舱位的关系

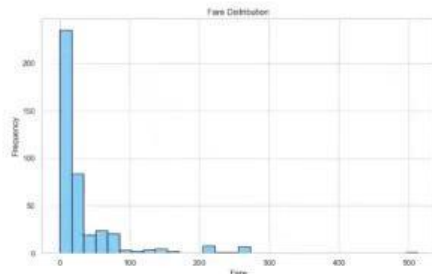
```
python  
  
# 票价与舱位的关系  
sns.boxplot(x='Pclass', y='fare', data=df, palette='magma')  
plt.title('Fare Distribution by Passenger Class')  
plt.xlabel('Passenger Class')  
plt.ylabel('Fare')  
plt.yscale('log') # 对数刻度展示票价  
plt.show()
```

结论：

- 头等舱票价显著高于其他舱位（中位数约84 vs 13）。
- 高票价与高生存率正相关。

Kimi k1.5

结合数据样本和分析结果，提供多种可视化图表绘制方案及对应代码，需采用Python代码完成绘图任务。大样本会省略数据；小样本不省略数据。



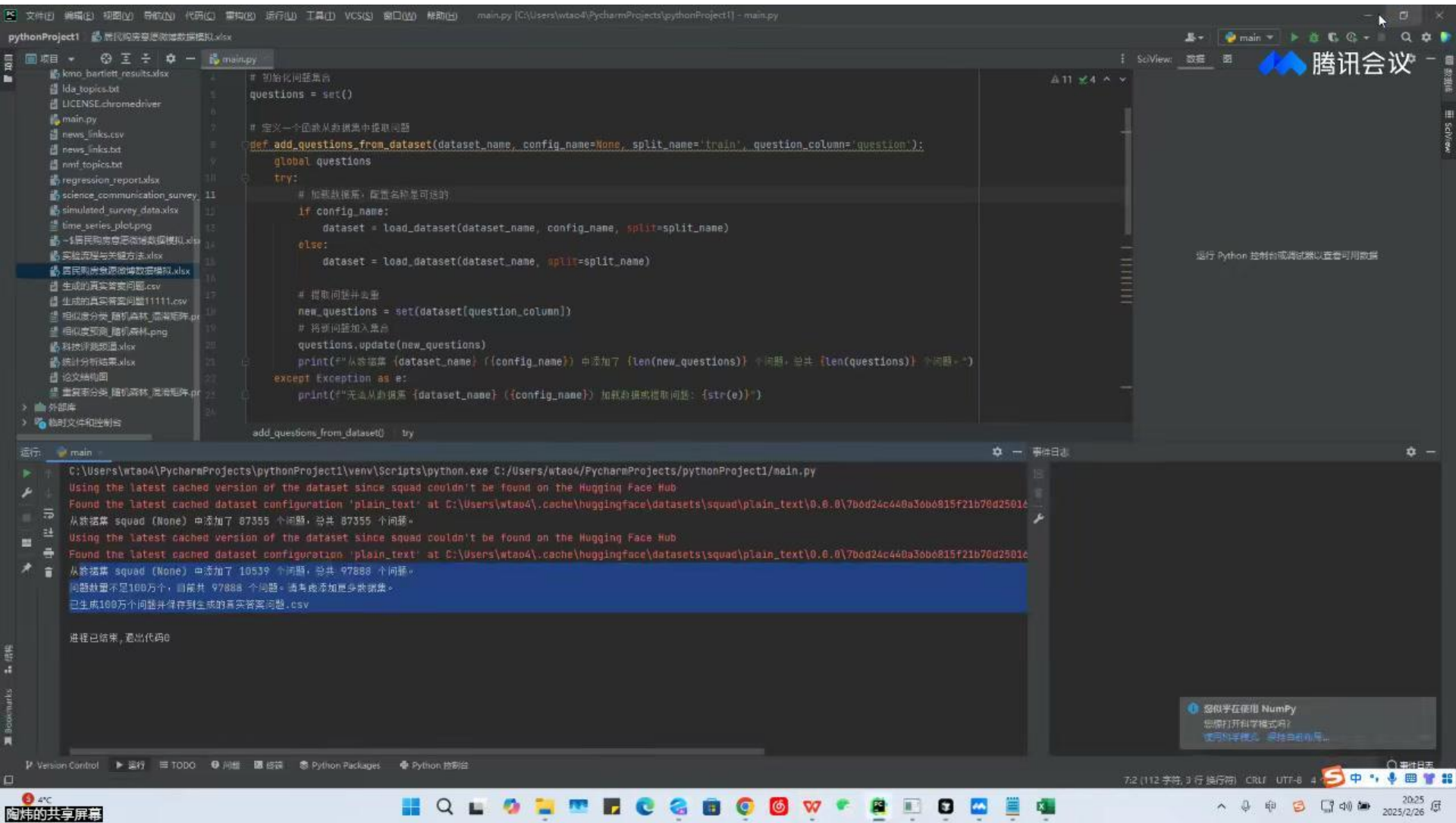
Claude 3.5 sonnet

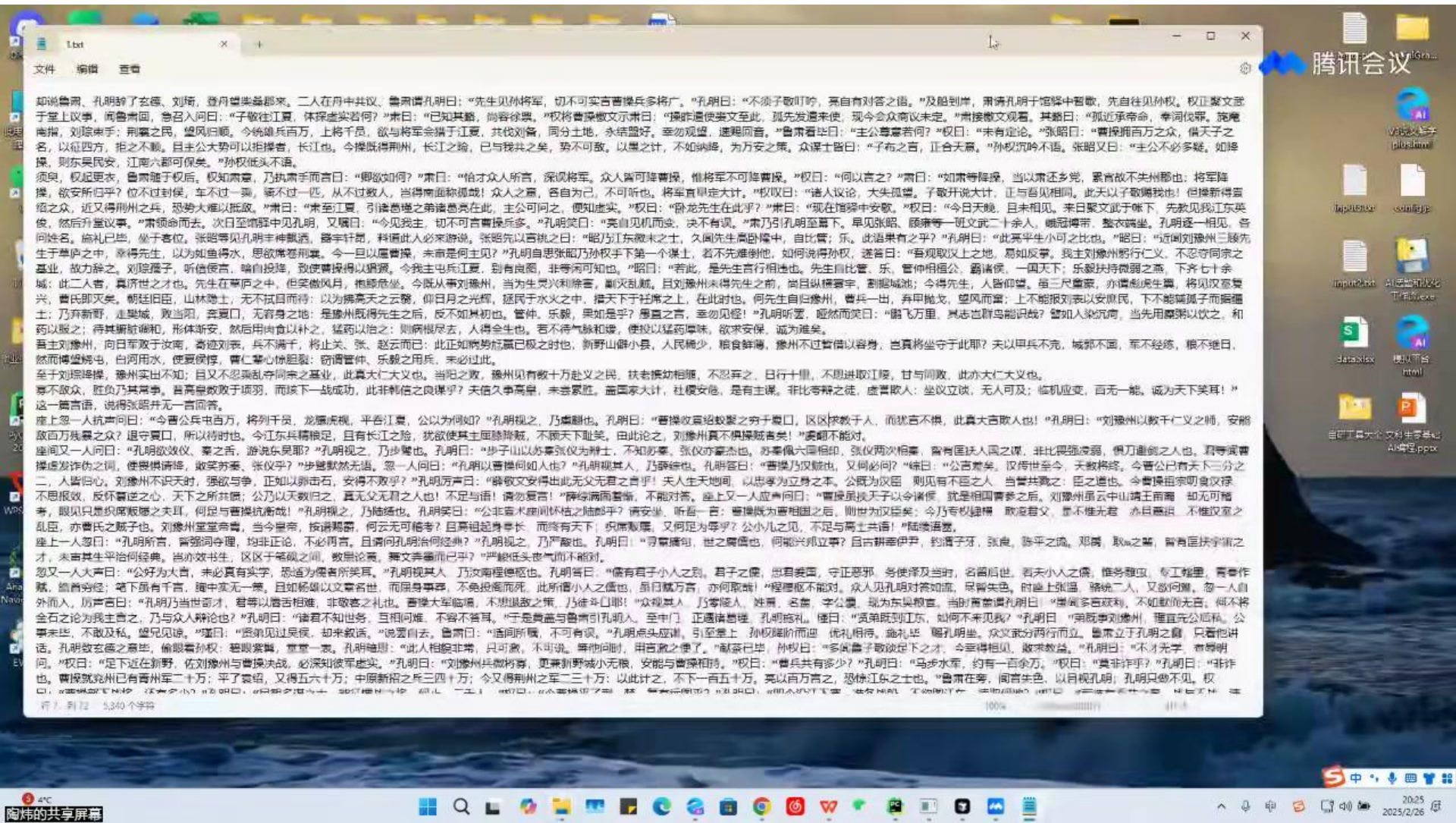
暂时不能直接绘制出可视图表，需要将绘图代码复制到本地运行。

社科的同学非常熟悉的问卷表格...

我本地有个excel文件，是问卷调查，一共279行数据（279个被试）。路径是"C:\Users\wtao4\OneDrive\桌面\data.xlsx"，里面每一列都是一个变量的数据值。Q1-Q42一共有42列，现在我想对这42个变量做相关分析，生成相关系数热力图。请你给我完整的python代码

Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Q13	Q14	Q15	Q16	Q17	Q18	Q19	Q20	Q21	Q22	Q23	Q24	Q25
5	5	5	5	5	5	5	1	5	4	5	5	5	1	5	5	5	5	5	5	5	5	5	5	5
4	5	2	4	4	5	5	1	5	5	5	5	5	1	5	5	5	5	5	5	5	5	5	5	5
5	5	5	5	5	4	5	1	3	4	4	4	4	1	3	4	4	3	4	4	5	4	4	4	3
5	4	3	5	3	4	2	4	4	4	5	4	5	2	4	5	4	3	4	5	3	4	5	3	4
5	4	4	5	4	5	1	4	4	5	4	4	1	4	3	4	3	4	4	4	4	4	3	5	5
5	4	4	4	5	4	1	4	5	4	3	5	2	5	4	3	5	5	4	3	4	5	4	3	3
4	4	5	4	5	5	1	4	5	4	4	5	1	4	5	4	5	4	5	4	5	3	5	4	4
5	5	4	4	3	5	2	5	4	4	5	4	2	5	3	4	4	4	4	5	5	5	4	4	4
4	5	3	4	5	5	1	5	4	4	5	5	1	4	5	5	5	5	5	4	4	5	4	5	4
5	4	5	5	5	4	2	4	5	3	5	5	2	5	4	4	5	5	4	5	4	5	5	5	4
5	5	5	5	5	5	1	5	4	5	5	5	1	5	5	5	5	5	5	5	5	5	5	5	5
4	5	2	4	4	5	1	5	5	5	5	5	1	5	5	5	5	5	5	5	5	5	5	5	5
5	5	5	5	4	5	1	3	4	4	4	4	1	3	4	4	3	4	4	5	4	4	4	4	3
5	4	3	5	3	4	2	4	4	5	4	5	2	4	5	4	3	4	5	3	4	5	3	4	4
5	4	4	5	4	5	1	4	4	5	3	4	1	4	3	4	3	4	4	4	4	4	3	5	5
5	4	4	4	5	4	1	4	5	4	3	5	2	5	4	3	5	5	4	3	4	5	4	3	3
5	5	5	5	5	5	1	5	4	5	5	5	1	5	5	5	5	5	5	5	5	5	5	5	5
4	5	2	4	4	4	1	5	5	5	5	5	1	5	5	5	5	5	5	5	5	5	5	5	5
5	5	5	5	4	5	1	3	4	4	4	4	1	3	4	4	3	4	4	5	4	4	4	4	3
5	4	3	5	3	4	2	4	4	5	4	5	2	4	5	4	3	4	5	3	4	5	3	4	4
5	5	5	5	4	5	1	3	4	4	4	4	1	3	4	4	3	4	4	5	4	4	5	4	3
5	4	3	5	3	4	2	4	4	5	4	5	2	4	5	4	3	4	5	3	4	5	3	4	4
5	4	4	5	4	5	1	4	4	5	4	4	1	4	3	4	3	4	4	4	4	4	3	5	5
4	4	4	4	5	4	1	4	5	4	3	5	2	5	4	3	5	5	4	3	4	5	4	3	3
5	5	5	5	5	5	1	5	4	5	5	5	1	5	5	5	4	5	5	5	5	5	5	5	5
4	5	2	4	4	5	1	5	5	5	5	5	1	5	5	5	5	5	5	5	5	5	5	5	5
5	5	5	5	4	5	1	3	4	4	4	4	1	3	4	4	3	4	4	5	4	4	4	4	3
5	4	3	5	3	4	2	4	4	5	4	5	2	4	5	4	3	4	5	3	4	5	3	4	4
5	4	4	5	4	5	1	4	4	5	4	4	1	4	3	4	3	4	4	4	4	4	3	5	5
5	4	4	4	5	4	1	4	5	4	3	5	2	5	4	3	5	5	4	3	4	5	4	3	3
4	4	5	4	5	5	1	4	5	4	4	5	1	4	5	5	4	5	4	5	4	5	3	5	4
5	5	4	4	3	5	2	5	4	4	5	4	2	5	3	4	4	4	4	4	5	5	5	4	4
5	5	5	5	5	5	1	5	4	5	5	5	1	5	5	5	5	5	5	5	5	5	5	5	5





自动保存 已保存到这台电脑

搜索

文件 开始 插入 页面布局 公式 数据 审阅 视图 帮助 Acrobat

剪贴板 复制 格式刷

粘贴 格式刷

字体

对齐方式

数字

样式

单元格

编辑

加载项

AY20

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
145	5	5	5	5	4	5	1	3	4	4	4	4	1	3	4	4	3	4	5	4	4	4	3	4	5	
146	5	4	3	5	3	4	2	4	4	5	4	5	2	4	5	4	3	4	5	3	4	5	3	4	5	
147	5	4	4	5	4	5	1	4	4	5	3	4	1	4	3	4	3	4	4	4	4	3	5	5	4	
148	5	4	4	4	5	4	1	4	5	4	3	5	2	5	4	3	5	5	4	3	4	5	4	3	5	
149	5	5	5	5	5	5	1	5	4	5	5	5	1	5	5	5	5	5	5	5	5	5	5	5	5	
150	4	5	2	4	4	4	1	5	5	5	5	5	1	5	5	5	5	5	5	5	5	5	5	5	5	
161	5	5	5	5	4	5	1	3	4	4	4	4	1	3	4	4	3	4	4	5	4	4	4	3	4	
162	5	4	3	5	3	4	2	4	4	5	4	5	2	4	5	4	3	4	5	3	4	5	3	4	5	
163	5	4	4	5	4	5	1	4	4	5	4	4	1	4	3	4	3	4	4	4	4	3	5	5	4	
164	5	5	5	5	5	5	1	5	4	5	5	5	1	5	5	5	5	5	5	5	5	5	5	5	5	
165	4	5	2	4	4	5	1	5	5	5	5	5	1	5	5	5	5	5	5	5	5	5	5	5	5	
166	5	5	5	5	4	5	1	3	4	4	4	4	1	3	4	4	3	4	4	5	4	4	4	3	4	
167	5	4	3	5	3	4	2	4	4	5	4	5	2	4	5	4	3	4	5	3	4	5	3	4	5	
168	5	4	4	5	4	5	1	4	4	5	4	4	1	4	3	4	3	4	4	4	4	3	5	5	4	
169	4	4	4	4	5	4	1	4	5	4	3	5	2	5	4	3	5	5	4	3	4	5	4	3	5	
170	5	4	4	4	5	4	1	4	5	4	3	5	2	5	4	3	5	5	4	3	4	5	4	3	5	
171	5	5	5	5	5	5	1	5	4	5	5	5	1	5	5	5	5	5	5	5	5	5	5	5	5	
172	4	5	2	4	4	4	1	5	5	5	5	5	1	5	5	5	5	5	5	5	5	5	5	5	5	
173	5	5	5	5	4	5	1	3	4	4	4	4	1	3	4	4	3	4	4	5	4	4	4	3	4	
174	5	4	3	5	3	4	2	4	4	5	4	5	2	4	5	4	3	4	5	3	4	5	3	4	5	
175	5	4	4	5	4	5	1	4	4	5	4	4	1	4	3	4	3	4	4	4	3	5	4	5	4	
176	5	5	5	5	5	5	1	5	4	5	5	5	1	5	5	5	5	5	5	5	5	5	5	5	5	
177	4	5	2	4	4	5	1	5	5	5	5	5	1	5	5	5	5	5	5	5	5	5	5	5	5	
178	5	5	5	5	4	5	1	3	4	4	4	4	1	3	4	4	3	4	4	5	4	4	4	3	4	
179	5	4	3	5	3	4	2	4	4	5	4	5	2	4	5	4	3	4	5	3	4	5	3	4	5	
180	5	4	4	5	4	5	1	4	4	5	4	4	1	4	3	4	3	4	4	4	3	5	5	4	3	
181	4	4	4	4	5	4	1	4	5	4	3	5	2	5	4	3	5	5	4	3	4	5	4	3	5	
182	5	5	5	5	5	5	1	5	4	5	5	5	1	5	5	5	4	5	5	5	5	5	5	5	5	
183	4	5	2	4	4	5	1	5	5	5	5	5	1	5	5	5	5	5	5	5	5	5	5	5	5	

就绪 辅助功能 一切就绪

25°C

2025/2/26

Python分析《三国演义》诸葛亮发言

```
python
# 拟合 VAR 模型
var_model = VAR(multi_series_diff)
var_results = var_model.fit(maxlags=5) # 最大滞后阶数为5
print(var_results.summary())

# 预测未来值
forecast_steps = 10
var_forecast = var_results.forecast(multi_series_diff.values[-var_results.k_ar:], steps=forecast_steps)
var_forecast_df = pd.DataFrame(var_forecast, columns=multi_series.columns)

# 可视化预测结果
plt.figure(figsize=(10, 6))
for col in multi_series.columns:
    plt.plot(multi_series_diff.index, multi_series_diff[col], label=f'{col} (差分后)')
    plt.plot(pd.date_range(multi_series_diff.index[-1], periods=forecast_steps + 1, freq='D')[1:],
             var_forecast_df[col], label=f'{col} 预测', linestyle='--')
plt.title("VAR 预测结果")
plt.legend()
plt.show()
else:
    print("\n数据只有一列，无法使用 VAR 模型。")
```

代码说明

1. 数据读取:

- 从 Excel 文件中读取数据, 假设第一列是时间序列数据。
- 如果有多列数据, 可以用于 VAR 模型分析。

2. 平稳性检验:

- 使用 ADF 检验检查数据是否平稳。

给 DeepSeek 发送消息

深度思考 (R1) 联网搜索

内容由 AI 生成, 请仔细甄别

DeepSeek - 探索未知之境

聊天窗口地址: chat.deepseek.com/a/chat/s/26244002-e389-43c9-8595-b7702d8f920d

书签手机书签character.ai香港中文大学新闻...DeepSeek - 探索...登录页面_问卷星SiliconCloud小木虫论坛_学术科...可灵 AI - 新一代 AI...资源访问控制系统...智灵动力元景AI设计平台虚拟人生力平台腾讯会议

Python分析《三国演义》诸葛亮发言

```
python
plt.figure(figsize=(10, 5))
plt.subplot(2, 1, 1)
plot_acf(time_series_diff, lags=20, ax=plt.gca())
plt.subplot(2, 1, 2)
plot_pacf(time_series_diff, lags=20, ax=plt.gca())
plt.tight_layout()
plt.show()

# 手动设置 ARIMA 参数 (p, d, q)
p = 1 # AR 阶数 (根据 PACF 图选择)
d = 1 # 差分阶数
q = 1 # MA 阶数 (根据 ACF 图选择)

# 拟合 ARIMA 模型
arima_model = ARIMA(time_series, order=(p, d, q))
arima_results = arima_model.fit()
print(arima_results.summary())

# 预测未来值
forecast_steps = 10 # 预测未来10个时间步
forecast = arima_results.get_forecast(steps=forecast_steps)
forecast_mean = forecast.predicted_mean
forecast_ci = forecast.conf_int()

# 可视化预测结果
plt.figure(figsize=(10, 5))
plt.plot(time_series, label="原始数据")
plt.plot(forecast_mean, label="预测值", color="red")
plt.fill_between(forecast_ci.index, forecast_ci.iloc[:, 0], forecast_ci.iloc[:, 1], color="pink", alpha=0.3)
plt.title("ARIMA 预测结果")
plt.legend()
plt.show()

# 4. VAR 模型 (适用于多变量时间序列)
# 如果将多个时间序列作为输入，可以使用 VAR 模型
# 假设输入数据为多变量时间序列
if df.shape[1] > 1:
    print("\n--- VAR 模型 ---")
    # 提取多变量时间序列
    multi_series = df.iloc[:, 1:].dropna() # 删除缺失值并提取多变量数据
```

给 DeepSeek 发送消息

内容由 AI 生成，请仔细甄别

24°C

陶伟的共享屏幕

2025/2/26

DeepSeek - 探索未知之境

SiliconCloud

论坛置顶 - 百度百科

说文解字plus

+

chat.deepseek.com/a/chat/s/26244002-e389-43c9-8595-87702d8f920d

书签手机书签character.ai香港中文大学新闻DeepSeek - 探索...登录网页_问卷星SiliconCloud小木虫论坛-学术科...可灵 AI - 新一代 AI...资源访问控制系统...智灵动力元策AI设计平台虚拟人生力平台腾讯会议

Python分析《三国演义》诸葛亮发言

```
nitered_words = [word for word in words if len(word) > 1 and word not in stopwords]

# 6. 词频统计 (带频次过滤)
word_counts = Counter(filtered_words)
min_freq = 2 # 最小出现次数
top20_words = [item for item in word_counts.most_common(20) if item[1] >= min_freq]

# 7. 可视化增强配置
plt.style.use('ggplot')
matplotlib.rcParams['font.sans-serif'] = ['SimHei']
matplotlib.rcParams['axes.unicode_minus'] = False

# 准备数据
keywords = [item[0] for item in top20_words]
counts = [item[1] for item in top20_words]

# 绘制高级柱状图
plt.figure(figsize=(12, 8))
bars = plt.barh(
    keywords, counts,
    color=plt.cm.tab20c(range(len(keywords))),
    edgecolor='black'
)

# 装饰图表
plt.xlabel("出现次数", fontsize=14, labelpad=10)
plt.ylabel("关键词", fontsize=14, labelpad=10)
plt.title("诸葛亮发言关键词TOP20分析", fontsize=18, pad=20, fontweight='bold')
plt.gca().invert_yaxis()
```

DeepSeek 发送消息

网络搜索 (RT) 联网搜索

内容由 AI 生成, 仅供参考

24°C

陶伟的共享屏幕

2025/2/26

20:25

Python分析《三国演义》诸葛亮发言

```

.logo {
  font-size: 24px;
  font-weight: bold;
}

.search-box {
  padding: 8px 15px;
  border-radius: 20px;
  border: none;
  width: 300px;
}

/* 首页内容区域 */
.container {
  max-width: 1200px;
  margin: 20px auto;
  padding: 0 20px;
}

/* 美食列表 */
.food-list {
  display: grid;
  grid-template-columns: repeat(auto-fit, minmax(300px, 1fr));
  gap: 25px;
  margin-top: 20px;
}

/* 美食卡片 */
.food-card {

```

● DeepSeek 深度求索

 免費參考 (R1)  聯同提供

```
Python分析《三国演义》诸葛亮发言
import pandas as pd
cmap='coolwarm', # 使用蓝-红渐变色
vmin=-1, vmax=1, # 固定颜色范围
square=True, # 保持单元格为正方形
linewidths=0.5, # 单元格间线条宽
cbar_kws={'shrink': 0.8} # 调整颜色条大小
}

# 设置图表标题和坐标轴
plt.title('Q1-Q42变量相关系数矩阵', fontsize=24, pad=20)
plt.xticks(rotation=45, ha='right', fontsize=10)
plt.yticks(fontsize=10)

# 调整布局
plt.tight_layout()

# 5. 保存和显示结果
plt.savefig('correlation_heatmap.png', dpi=300, bbox_inches='tight')
print("热力图已保存为 correlation_heatmap.png")
plt.show()
```

以下是完整的代码，用于读取 Excel 文件中的 Q1-Q42 列数据，计算相关系数矩阵，并绘制热力图。代码中包含了数据预处理、缺失值处理、相关系数计算以及热力图的绘制和保存。

完整代码

```
python
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

给 DeepSeek 发送消息

淘淘想聊 (RT) 联网搜索

pythonProject1main.py

main.py

input.txt

kono bartlett_results.xlsx

lda_topics.txt

LICENSE.chromedriver

main.py

news_links.csv

news_links.txt

nmf_topics.txt

regression_report.xlsx

science communication survey

simulated_survey_data.xlsx

time_series_plot.png

实验流程与关键方法.xlsx

居民消费价格指数数据模拟.xlsx

生成的真实答案问题.csv

生成的真实答案问题11111.csv

相似度分类_随机森林_混淆矩阵.pr

相似度预测_随机森林.png

科技评测数据.xlsx

统计分析报告.xlsx

论文结构图

重聚类分析_随机森林_混淆矩阵.pr

外部库

临时文件和控制台

main

13annot=False, # 不显示具体数值 (42x42显示会太密集)

14cmap='coolwarm', # 使用蓝-红渐变色

15vmin=-1, vmax=1, # 固定颜色范围

16square=True, # 保持单元格为正方形

17linewidths=0.5, # 单元格的线条

18cbar_kws={'shrink': 0.8} # 调整颜色条大小

19)

20

21# 设置图表标题和坐标轴

22plt.title("Q1-Q42变量相关系数矩阵", fontsize=24, pad=20)

23plt.xticks(rotation=45, ha='right', fontsize=10)

24plt.yticks(fontsize=10)

25

26# 调整布局

27plt.tight_layout()

28

29# 5. 保存和显示结果

30plt.savefig('correlation_heatmap.png', dpi=300, bbox_inches='tight')

31print("热力图已保存为 correlation_heatmap.png")

32

33plt.show()

运行 Python 控制台或测试器以查看可用数据

运行main

1C:\Users\wtao4\PchamProjects\pythonProject1\venv\Scripts\python.exe C:/Users/wtao4/PycharmProjects/pythonProject1/main.py

2Using the latest cached version of the dataset since squad couldn't be found on the Hugging Face Hub

3Found the latest cached dataset configuration 'plain_text' at C:\Users\wtao4\.cache\huggingface\datasets\squad\plain_text\0.0.0\7b6d24c440a36b6815f21b70d2501c

4从数据集 squad (None) 中添加了 87355 个问题, 总共 87355 个问题。

5Using the latest cached version of the dataset since squad couldn't be found on the Hugging Face Hub

6Found the latest cached dataset configuration 'plain_text' at C:\Users\wtao4\.cache\huggingface\datasets\squad\plain_text\0.0.0\7b6d24c440a36b6815f21b70d2501c

7从数据集 squad (None) 中添加了 10539 个问题, 总共 97888 个问题。

8问题数量不足100万个, 目前共 97888 个问题。请考虑添加更多数据集。

9已生成100万个问题并保存到生成的真实答案问题.csv

10

11进程已结束, 退出代码0

您似乎在使用 NumPy

想打开科学模式吗?

使用科学模式 保持当前设置...

Version Control

运行

TODO

问题

错误

Python Packages

Python 控制台

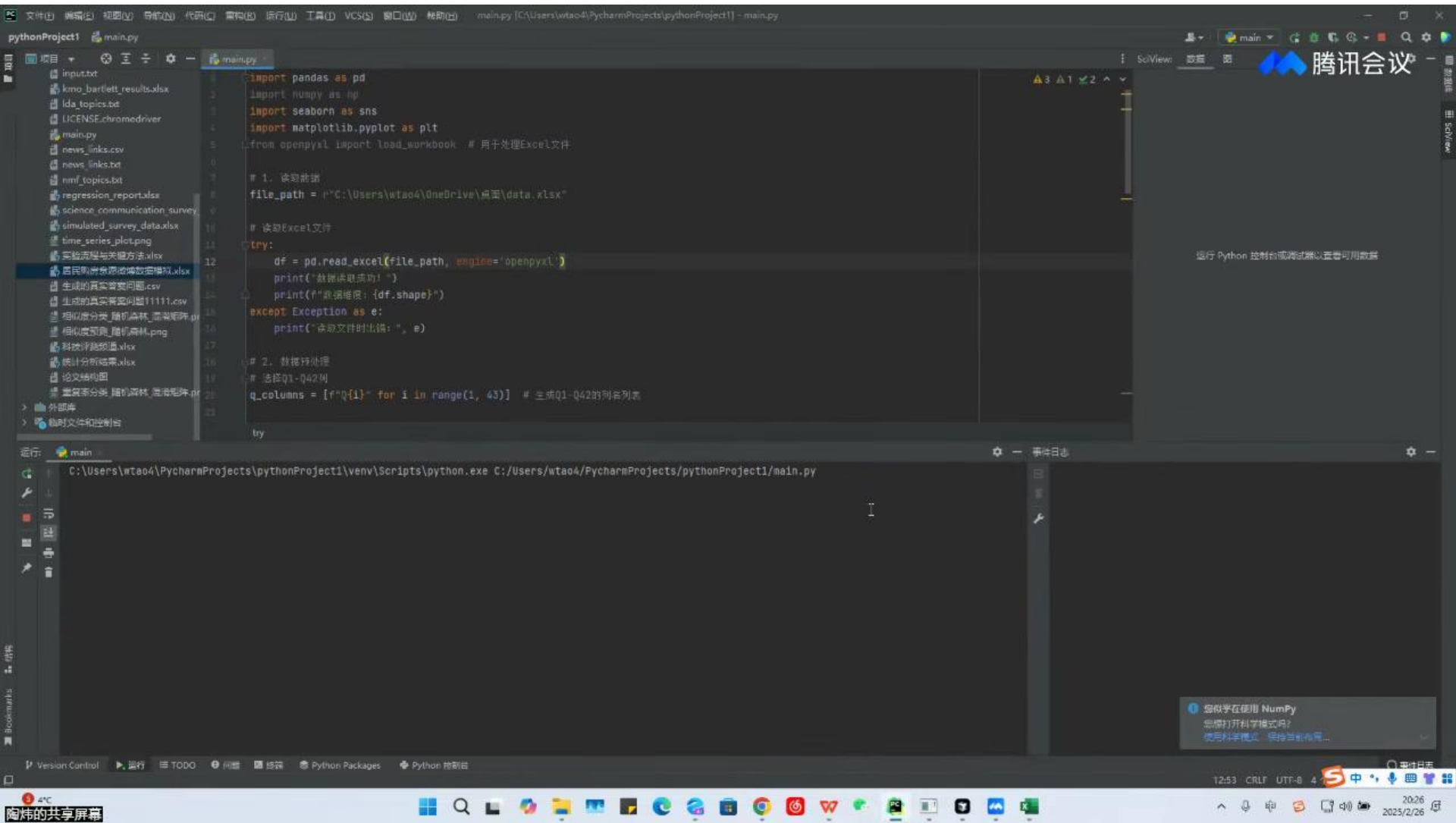
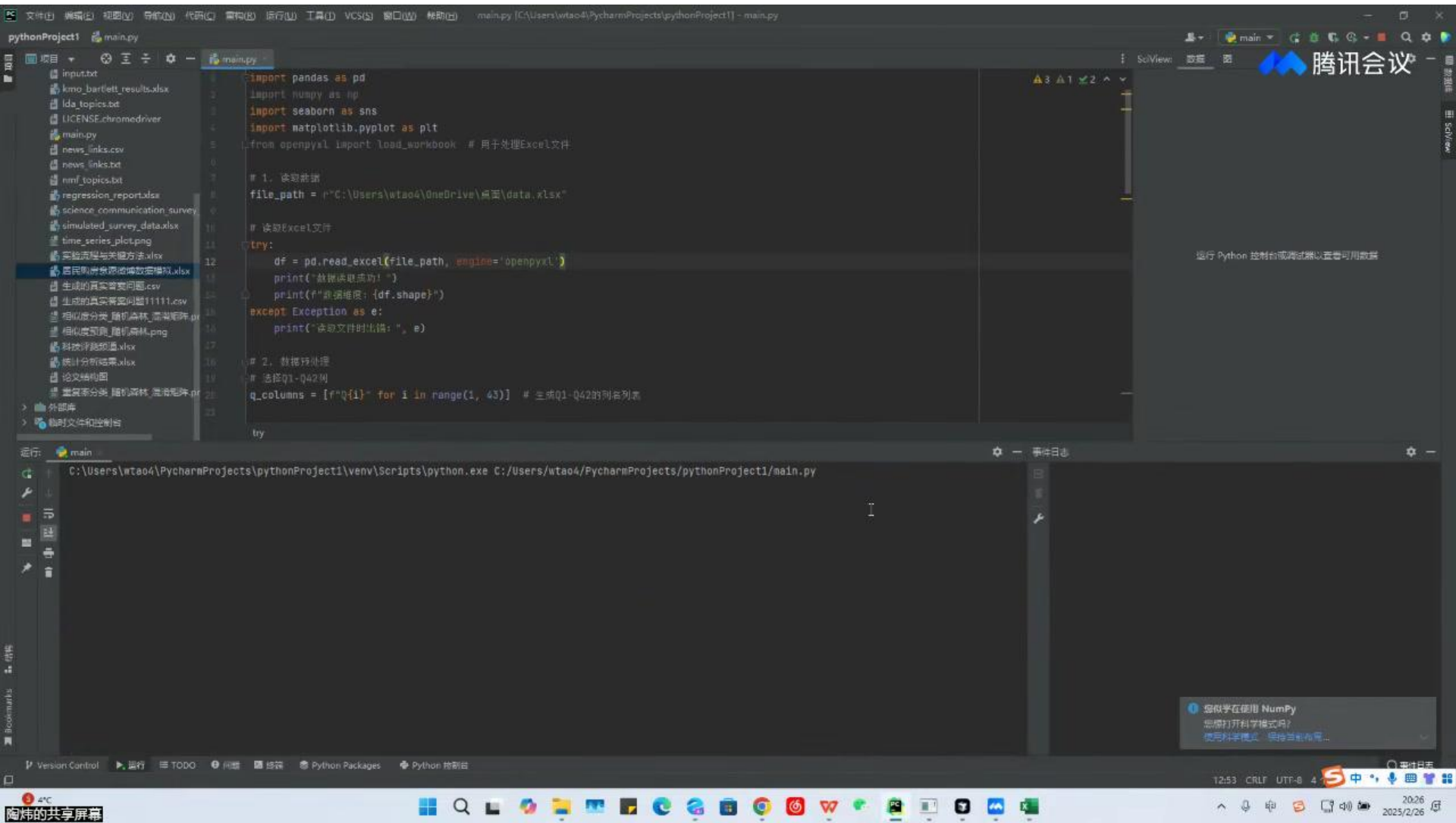
52:15 CRLF UTF-8 4

20:26 2023/2/26

腾讯会议

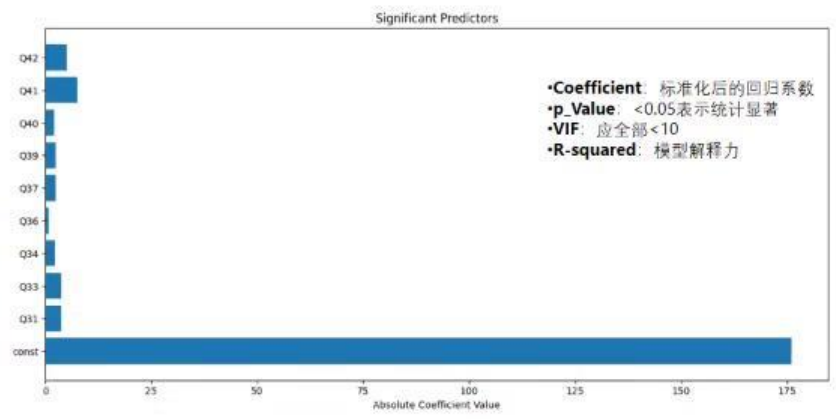
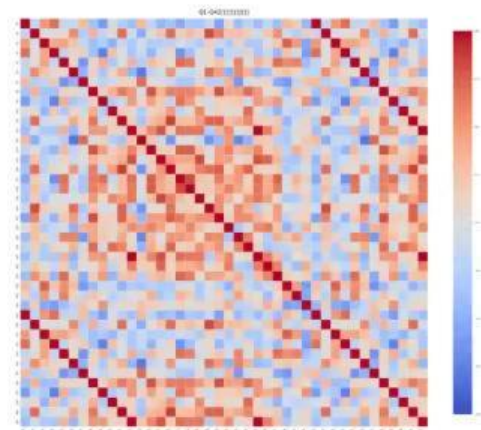
45°C

陶伟的共享屏幕



不仅仅是相关分析，还可以建模...

在这个代码基础上，每一行Q1-Q42的加总值，是因变量，现在需要你对这个42个自变量对因变量做多元线性回归计算，并且输出计算结果，给我完整的python代码，并且考虑共线性的问题



57 测试题面

58 数据可视化

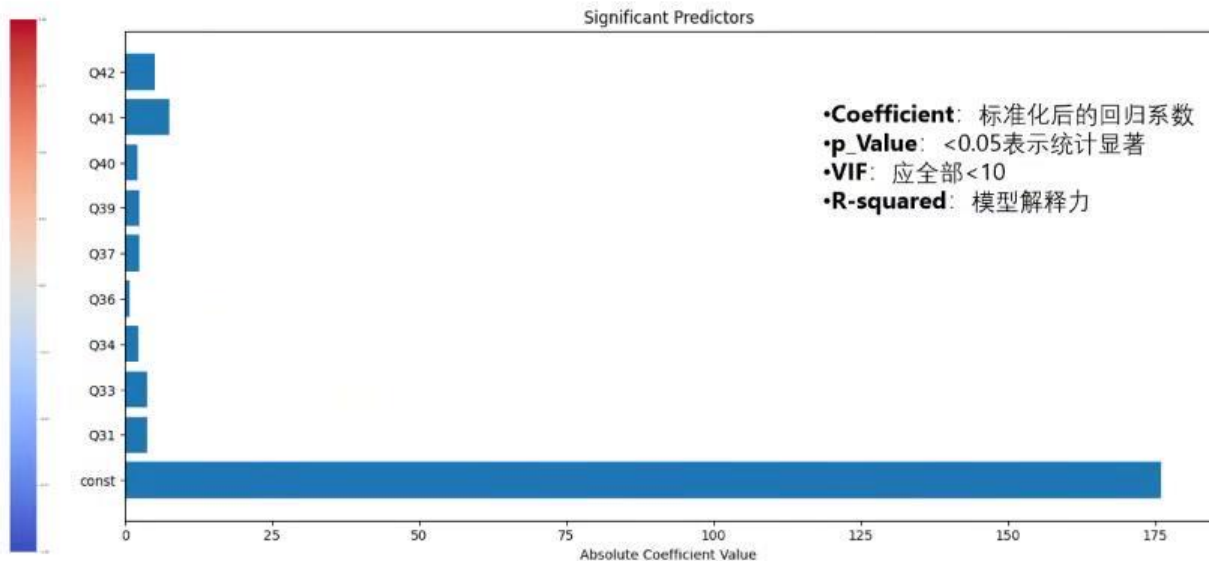
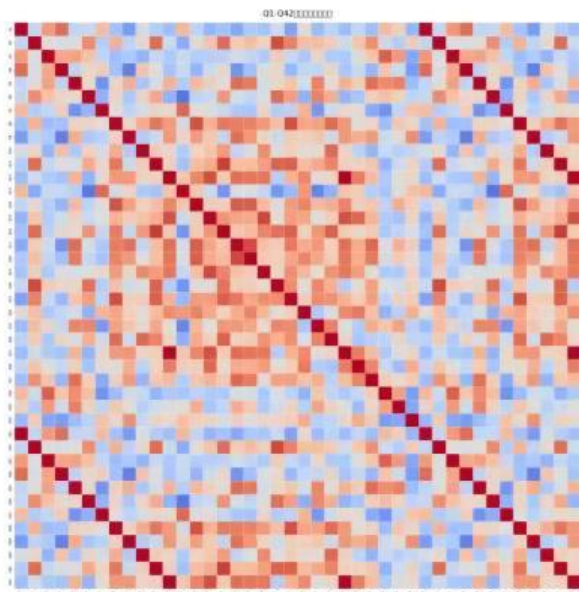
59 杜林的同学非实验室的问卷表格...

60 不仅仅是相关分析，还可以建模...

61 用AI分析调查事件可以分为哪些步骤？

不仅仅是相关分析，还可以建模...

在这个代码基础上，每一行Q1-Q42的加总值，是因变量，现在需要你对这个42个自变量对因变量做多元线性回归计算，并且输出计算结果，给我完整的python代码，并且考虑共线性的问题



用AI分析舆情事件可以分为哪些步骤？

1. 数据收集：首先，需要从各种社交媒体平台、新闻网站、论坛等收集与事件相关的数据。这包括帖子、评论、新闻文章等。数据可以通过API接口或者网络爬虫技术来获取。

2. 数据预处理：收集的数据通常需要进行清洗和预处理，以提高分析的质量。这可能包括去除重复内容、删除无关信息、文本标准化（如统一词形、去除停用词）等。

3. 情感分析：通过情感分析可以判断公众对某一事件的情感倾向，如正面、负面或中性。这通常是通过预训练的情感分析模型来完成的，这些模型能够识别文本中的情感色彩。

4. 主题识别：利用主题模型（如LDA）来识别舆情数据中的主要话题。这有助于了解讨论的焦点以及公众关注的主要方面。

5. 趋势分析：分析话题和情感的变化趋势。这可以通过时间序列分析来实现，观察不同时间点的舆论变化，从而捕捉舆论波动和发展趋势。

6. 影响力分析：识别在舆情事件中具有重要影响力的账户或信息源。这包括分析这些账户的关注者数量、帖子的转发和点赞数等指标。

7. 视觉分析：对于包含图片和视频的数据，可以使用计算机视觉技术来分析图像内容，识别图中的对象、场景和活动，以及它们如何与文本数据相结合来影响舆情。

8. 报告与可视化：最后，将分析结果以图表、时间线、地图等形式进行可视化，使得非技术背景的人也能理解舆情分析的结果。

根据依存关系构建舆论图谱

在自然语言处理中，依存关系（dependency relations）是用于描述句子中词与词之间的关系。这些关系帮助揭示句子的句法结构。

1.nsubj (nominal subject): 名词性主语。这个依存关系连接一个动词和它的主语。例如，在句子“猫吃鱼”中，“猫”是动词“吃”的名词性主语。

2.dobj (direct object): 直接宾语。这个依存关系连接一个动词和它的直接宾语。例如，在“他在看书”中，“书”是动词“看”的直接宾语。

3.pobj (prepositional object): 介词宾语。这个依存关系连接一个介词和它的宾语。在“她坐在凳子上”中，“凳子”是介词“上”的宾语。

4.attr (attribute): 属性。这个依存关系连接一个连系动词和它的补语，这个补语通常是名词或形容词，用来描述或补充主语。例如，在“他是老师”中，“老师”是动词“是”的属性，描述了“他”。

5.ROOT: 根依存关系。在依存句法分析中，每个句子有一个核心的 ROOT，这是整个句子的中心，通常是主要的谓语动词。在“他能游泳”中，“游泳”可能被标记为 ROOT，因为它是句子结构的中心。

根据依存关系构建舆论图谱

```
import spacy
import networkx as nx
import matplotlib.pyplot as plt
import re

# 加载中文 NLP 模型
nlp = spacy.load('zh_core_web_sm')
nlp.max_length = 1000000 # 设置最大长度为1,000,000字符

# 从文件读取文本
with open('C:/Users/wtao4/Desktop/input.txt', 'r', encoding='utf-8') as file:
    text = file.read()

# 检查文本长度
if len(text) > nlp.max_length:
    print(f"Warning: text length ({len(text)}) exceeds nlp.max_length ({nlp.max_length})")

# 使用 spaCy 处理文本，移除标点符号
text = re.sub(r'[\W\s]', '', text)
doc = nlp(text)
```

文本导入和预处理

```
# 创建图
G = nx.Graph()

# 添加实体到图中
for ent in doc.ents:
    G.add_node(ent.text, label=ent.label_)

# 添加依存关系到图中
for token in doc:
    if token.dep_ in ('nsubj', 'dobj', 'pobj', 'attr', 'ROOT'):
        for child in token.children:
            if child.dep_ in ('nsubj', 'dobj', 'pobj', 'attr', 'prep') and token.head != child:
                G.add_edge(token.head.text, child.text, label=token.dep_)

# 选择图中最有代表性的20个实体（基于节点的度）
top_nodes = sorted(G.nodes(data=True), key=lambda x: G.degree(x[0]), reverse=True)[:20]
```

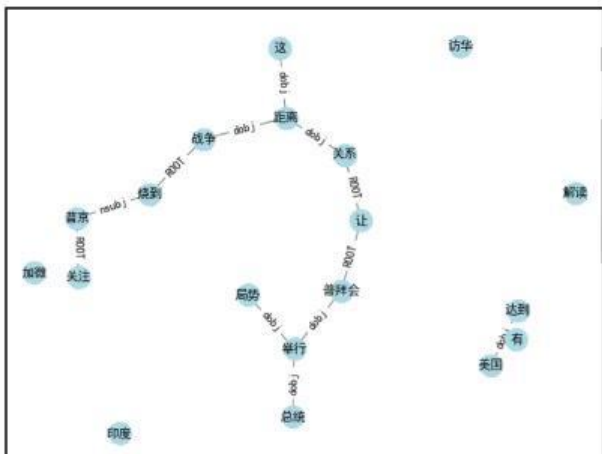
实体抽取和依存关系加载

图片构建

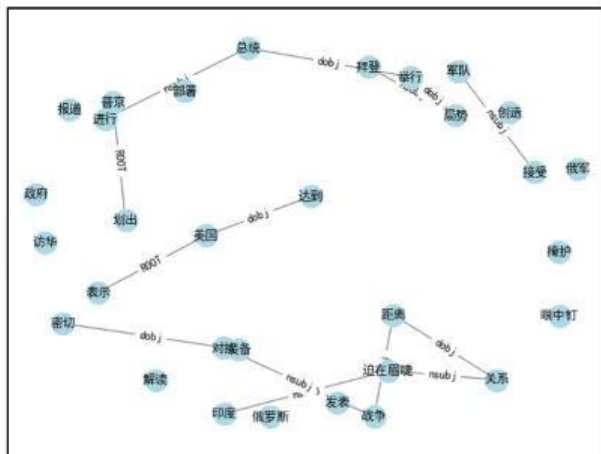
```
# 创建新图只包含这些顶级实体
H = nx.Graph()
for node, data in top_nodes:
    if len(node) <= 3: # 只包含名称长度不超过3个中文字符的实体
        H.add_node(node, **data)
        for u, v, d in G.edges(node, data=True):
            if v in H.nodes:
                H.add_edge(u, v, **d)

# 使用 Spring 布局，调整 k 值以增加节点间距离
pos = nx.spring_layout(H, k=0.5) # k值越大，节点间的距离越大

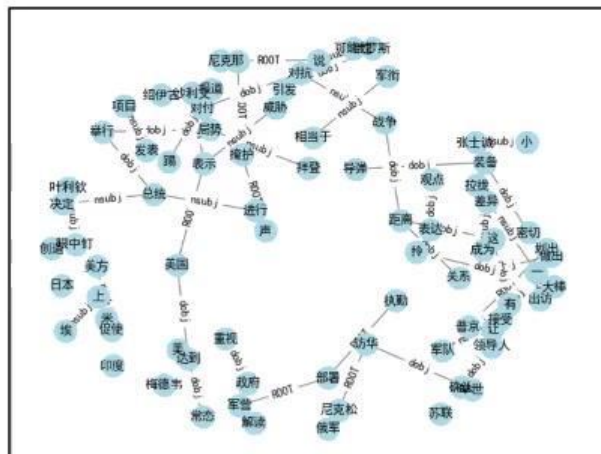
# 绘制图谱
labels = {node: node for node in H.nodes()}
nx.draw(H, pos, labels=labels, with_labels=True, node_color='lightblue', edge_color='gray', font_size=10, font_family='SimHei')
edge_labels = nx.get_edge_attributes(H, 'label')
nx.draw_networkx_edge_labels(H, pos, edge_labels=edge_labels, font_size=9, font_family='SimHei')
plt.show()
```



优化前



第一次优化后



第二次优化后

根据依存关系构建舆论图谱

```
import spacy
import networkx as nx
import matplotlib.pyplot as plt
import re

# 加载中文 NLP 模型
nlp = spacy.load('zh_core_web_sm')
nlp.max_length = 1000000 # 设置最大长度为1,000,000字符

# 从文件读取文本
with open('C:/Users/wtao4/Desktop/input.txt', 'r', encoding='utf-8') as file:
    text = file.read()

# 检查文本长度
if len(text) > nlp.max_length:
    print(f'Warning: text length ({len(text)}) exceeds nlp.max_length ({nlp.max_length})')

# 使用 spaCy 处理文本，移除标点符号
text = re.sub(r'[\W\s]', '', text)
doc = nlp(text)
```

文本导入和预处理

```
# 创建图
G = nx.Graph()

# 添加实体到图中
for ent in doc.ents:
    G.add_node(ent.text, label=ent.label_)

# 添加依存关系到图中
for token in doc:
    if token.dep_ in ['nsubj', 'dobj', 'pobj', 'attr', 'ROOT']:
        for child in token.children:
            if child.dep_ in ['nsubj', 'dobj', 'pobj', 'attr', 'prep'] and token.head != child:
                G.add_edge(token.head.text, child.text, label=token.dep_)

# 选择图中最有代表性的20个实体（基于节点的度）
top_nodes = sorted(G.nodes(data=True), key=lambda x: G.degree(x[0]), reverse=True)[:20]
```

实体抽取和依存关系加载

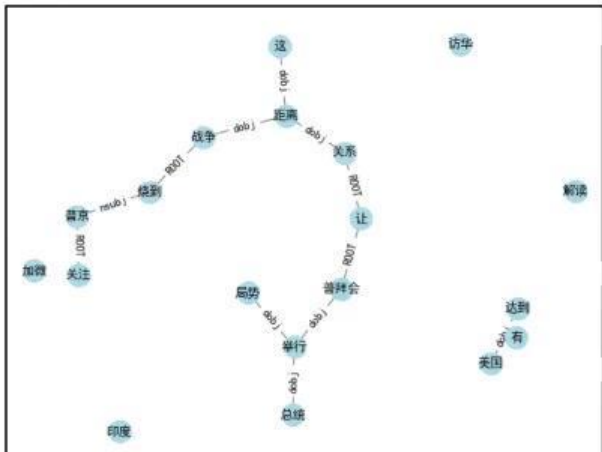
图片构建

```
# 创建新图包含这些前驱实体
H = nx.Graph()

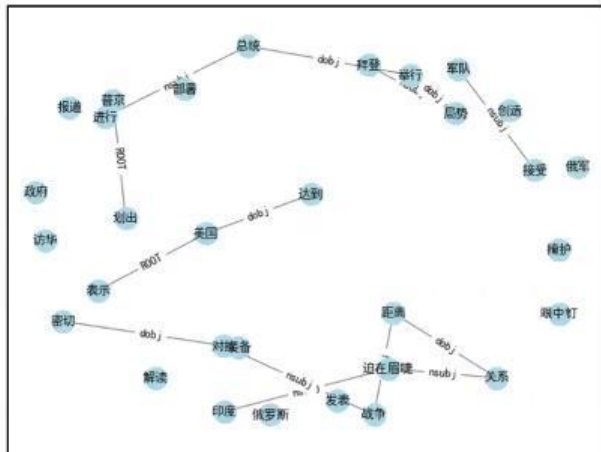
for node, data in top_nodes:
    if len(node) <= 3: # 仅包含名称长度不超过3个中文字符的实体
        H.add_node(node, **data)
        for u, v, d in G.edges(node, data=True):
            if v in H.nodes:
                H.add_edge(u, v, **d)

# 使用 Spring 布局，调整 k 可以增加节点间距离
pos = nx.spring_layout(H, k=0.5) # k值越大，节点间的距离越大

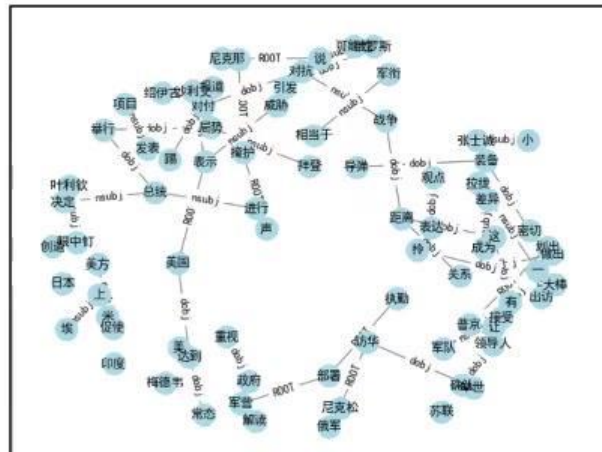
# 绘制图谱
labels = {node: node for node in H.nodes()}
nx.draw(H, pos, labels=labels, with_labels=True, node_color='lightblue', edge_color='grey', font_size=10, font_family='SimHei')
edge_labels = nx.get_edge_attributes(H, 'label')
nx.draw_networkx_edge_labels(H, pos, edge_labels=edge_labels, font_size=9, font_family='SimHei')
plt.show()
```



优化前



第一次优化后



第二次优化后

知识图谱的AI辅助构建：自定义和读取定义

```
# 模拟抓取页面数据
```

```
data = {  
    '中文名': '中华人民共和国',  
    '外文名': 'the People's Republic of China',  
    '简称': '中国',  
    '所属洲': '亚洲',  
    '首都': '北京',  
    '国庆日': '10月1日',  
    '国歌': '《义勇军进行曲》',  
    '国家代码': 'CHN',  
    '官方语言': '汉语（通用普通话）',  
    '货币': '人民币（港元、澳门币、新台币）',  
    '时区': '东八区（北京时间）',  
    '政治体制': '人民代表大会制度',  
    '人口数量': '140967万（2023年末）',  
    '主要民族': '汉族等56个民族',  
    '主要宗教': '道教、佛教、伊斯兰教、天主教、基督教（新教）',  
    '陆地面积': '约9600000 km²',  
    '水域率': '2.8%',  
}
```

自己输入的文本

```
# 创建一个图谱
```

```
g = Graph()
```

```
ns = Namespace("http://example.org/")
```

```
# 定义实体
```

```
china = URIRef(ns['China'])
```

```
# 添加属性
```

```
for key, value in data.items():
```

```
    property = URIRef(ns[key.replace(' ', '_').replace('(', '').replace(')', '')])  
    g.add((china, property, Literal(value, datatype=XSD.string)))
```

```
# 可视化
```

```
G = nx.DiGraph()
```

```
for subj, pred, obj in g:
```

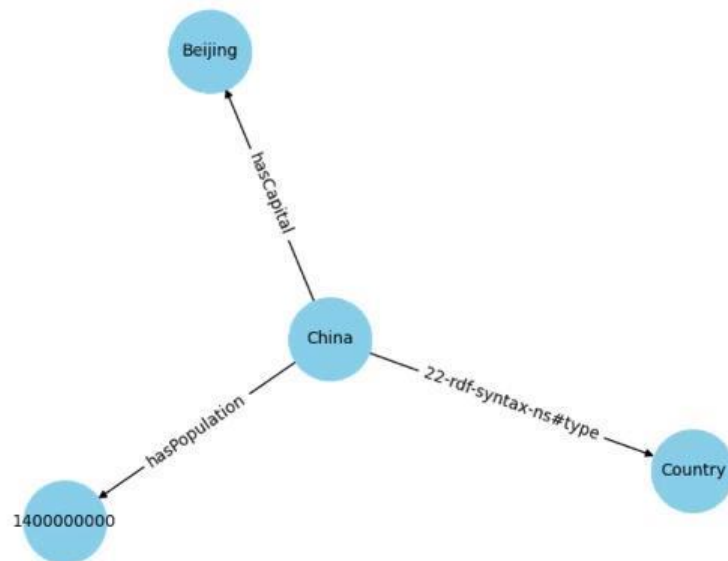
```
    subj_name = str(subj).split('/')[-1]
```

```
    pred_name = str(pred).split('/')[-1]
```

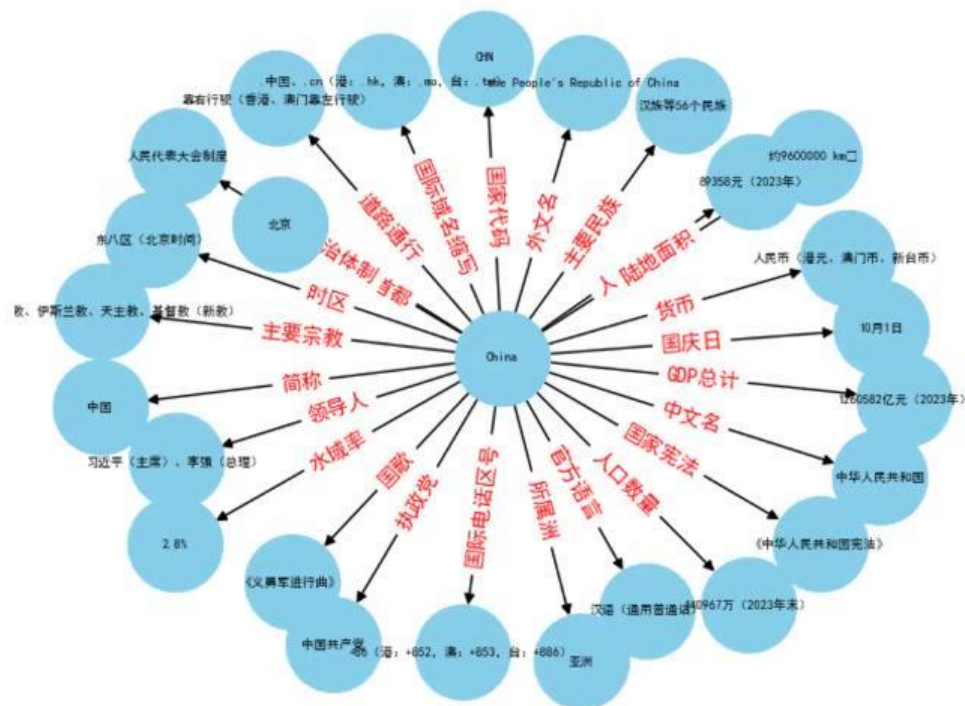
```
    obj_name = str(obj)
```

图谱构建

知识图谱的AI辅助构建：自定义和读取定义



简易



复杂



Read View source View history Tools ▾

Coordinates: 35°N 103°E

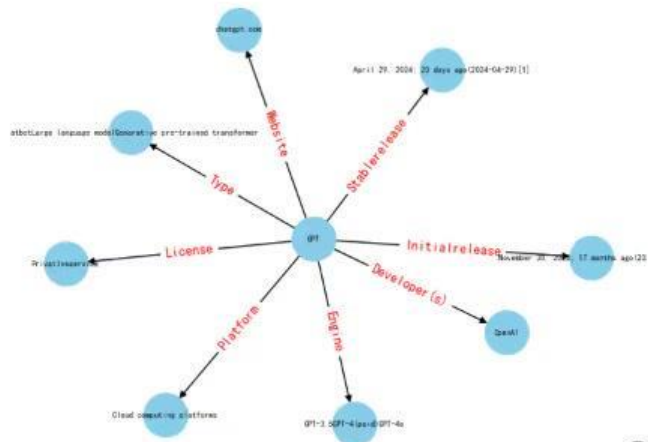
China,^[4] officially the **People's Republic of China (PRC)**,^[5] is a country in East Asia. With a population exceeding 1.4 billion, it is the world's **second-most populous country** after India. China spans the equivalent of five time zones and borders fourteen countries by land.^[6] With an area of nearly 9.6 million square kilometers (3,700,000 sq mi), it is the **third-largest country** by total land area.^[7] The country is divided into 33 **province-level divisions**: **22 provinces**,^[8] five autonomous regions, four municipalities, and two semi-autonomous special administrative regions. Beijing is the national capital, while Shanghai is its most populous city and largest financial center.

One of the **cradles of civilization**, China has been inhabited since the **Paleolithic era**, with the earliest **dynasties** emerging in the **Yellow River** basin before the late second millennium BCE. The eighth to third centuries BCE saw a breakdown in the authority of the **Zhou dynasty**, accompanied by the emergence of administrative and military techniques, **literature**, **philosophy**, and **historiography**. In 221 BCE, China was unified under an **emperor** for the first time. Appointed non-hereditary officials began ruling counties instead of the aristocracy, ushering in more than two millennia of imperial dynasties including the **Qin**, **Han**, **Tang**, **Yuan**, **Ming**, and **Qing**. With the invention of **gunpowder** and **paper**, the establishment of the **Silk Road**, and the building of the **Great Wall**, **Chinese culture**—including languages, traditions, architecture, philosophy and technology—flourished and has **heavily influenced East Asia** and beyond.

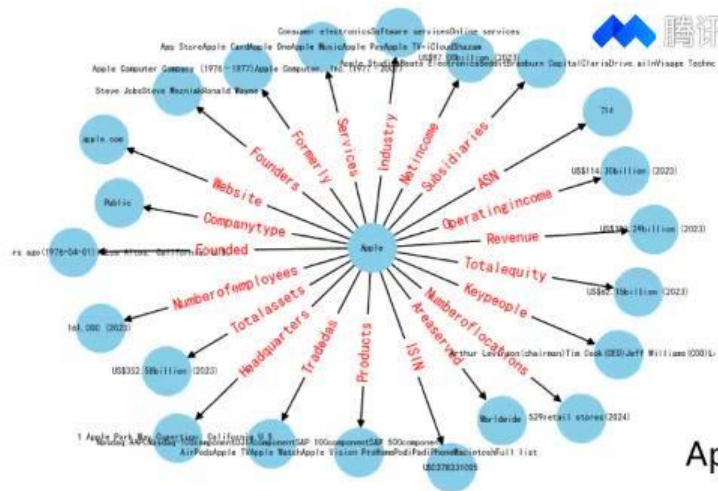
After decades of struggle, the monarchy was overthrown in 1912 and the Republic of China (ROC) was established. Despite China's eventual victory in the Second Sino-Japanese War and the Pacific War in general, economic stagnation took on the Republic because of lasting effects on



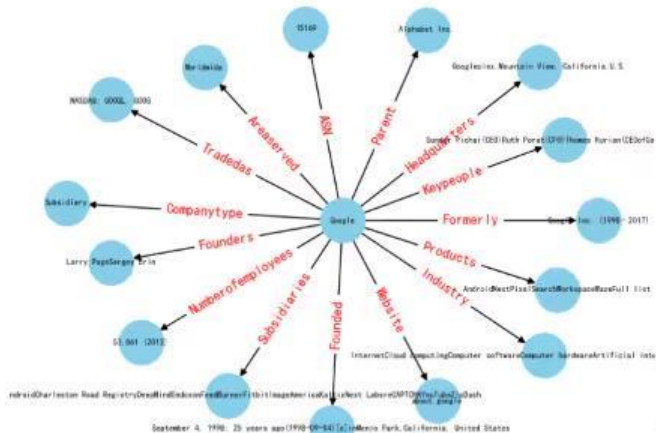
Wiki China 页面



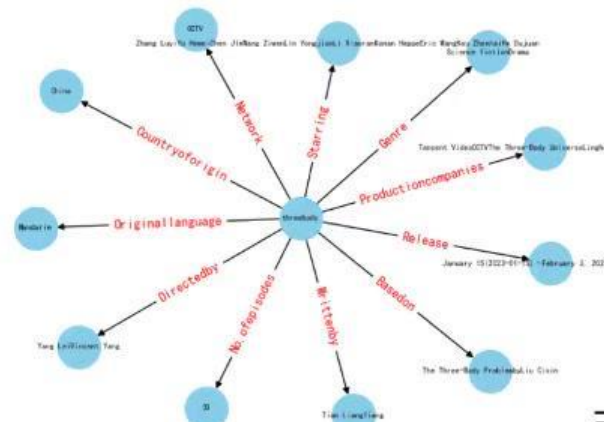
GPT



Apple



Google



三体

四

文生网站

API 的认知重构：

将各种功能模块进行黑箱化封装，我们无需了解其内部复杂的实现细节，只需知道如何调用这些功能。例如，地图 API 可以直接在网站中嵌入地图功能，而我们不必知道地图数据的存储和处理方式。

多智能体模拟的组件化思维：

把每个智能体看作是一个功能单元，它具有特定的特性和交互规则。例如，在一个电商网站中，购物车智能体负责管理商品添加、删除和结算等功能，用户智能体负责用户登录、信息管理等功能。

交互设计的构建：

根据网站的特定主题和功能进行设计，确保用户能够方便、快捷地使用网站。比如，一个旅游网站的交互设计要突出景点介绍、预订功能等，界面要简洁美观，易于操作。

DeepSeek 实现路径



智能体平台搭建：提示模式为“创建 N 个具备 [角色] 的智能体，交互规则包括 [...], 通过 [API] 连接”。

例如，“创建 3 个智能体，分别是用户智能体、商品智能体和订单智能体，交互规则是用户智能体可以添加商品智能体到购物车，订单智能体负责结算，通过电商 API 连接”。

交互原型生成：描述逻辑为“需要 [页面类型]，包含 [元素]，交互时触发 [动作]，数据来自 [来源]”。

例如，“需要一个商品详情页面，包含商品图片、名称、价格、描述等元素，当用户点击购买按钮时，触发添加到购物车动作，数据来自商品数据库”。

DeepSeek+在线编译：创建美食网站



DeepSeek-R1

“创建一个美食推荐网站，要有首页和美食详情页，在首页点击美食图片能跳转到详情页，样式参考大众点评的风格”。给我完整的HTML代码，能直接运行

深度思考

好的，用户让我创建一个类似大众点评风格的美食推荐网站，需要首页和详情页，点击图片跳转，并且提供完整的HTML代码直接运行。首先，我需要确定用户的技术水平，可能对前端有一定了解，但需要现成的代码，可能用于快速展示或学习。用户提到样式参考大众点评，所以要注意布局、颜色、卡片设计等元素，可能包括导航栏、搜索框、美食卡片网格布局、分页等。

接下来，我需要构建结构。首页和详情页需要不同的HTML文件，但用户要求直接运行的代码，可能需要将所有内容放在一个HTML文件中，利用锚点或模拟多个页面。不过更合理的方式是分开两个文件，但用户可能需要单个文件，所以可能需要使用简单的页面内跳转，或者用注释说明分文件的方法。不过，考虑到用户希望直接运行，可能更适合一个文件，但点击跳转需要两个页面，因此可能需要两个HTML文件，但用户可能希望一个文件内解决。这时候可能需要使用简单的锚点或者JavaScript切换内容，但用户可能更倾向于两个独立页面，所以代码可能需要分成index.html和detail.html，但用户要求“完整的HTML代码”，可能需要分别提供，或者说明如何创建两个文件。

然后，样式方面，参考大众点评，通常有顶部导航栏，红色系，搜索框，卡片布局，图片在上，文字在下，分页按钮。颜色选择上，导航栏可以用红色，按钮用红色，卡片阴影等。结构上，首页用网格布局展示多个美食卡片，每个卡片有图片、标题、评分、点评数、人均价格。图片点击跳转到详情页，详情页包括大图、详细描述、地址、电话、评分详情等。

需要确保响应式设计，使用flex或grid布局，图片自适应，移动端友好。交互部分，首页的图片链接到详情页，使用

关于代码结构，首页(index.html)包含导航、搜索、美食列表，详情页(detail.html)有返回首页的链接，详细的美食信息。需要检查图片路径是否正确，用户可能需要替换为自己的图片，所以使用占位图片链接，比如unsplash的随机图片。

测试代码是否直接运行，确保没有外部依赖，所有样式内联，图片链接有效。可能遇到的问题包括分页样式、卡片悬停效果、图片大小调整，这些都需要在CSS中处理好。另外，确保详情页的布局清晰，信息层次分明，符合大众点评的风格。

最后，提供代码时，要注明需要将两个HTML文件放在同一目录，并替换图片链接，以及可能的后续扩展建议，比如JavaScript交互、后端功能等，但用户当前只需要HTML和CSS，所以保持简单。

