2025 第二届信息技术应用创新大赛

基于大模型文献数据应用创新赛

参赛指南

大赛组委会

2025 年 5 月

目录

基于大模型文献数据应用创新赛
参赛指南

【赛题理解】

如下场景示例仅供参考，本次为发散性作品赛，参赛者可根据主办方提供的数据自由发挥。

（1）参考方向：洞察分析

涵盖研究趋势、研究热点、主题变迁、合作网络、人才画像及技术成熟度预测，结合 AI 动态展示与文献溯源体系，全面解析技术领域发展脉络与未来方向。

（2）参考方向：个人 AI 知识库

支持多技术领域文献管理，具备主题分类、AI 综述、全库问答功能，可解读单篇论文并辅助阅读，提供智能摘要、相似论文推荐及永久对话记录，助力高效科研与学习。

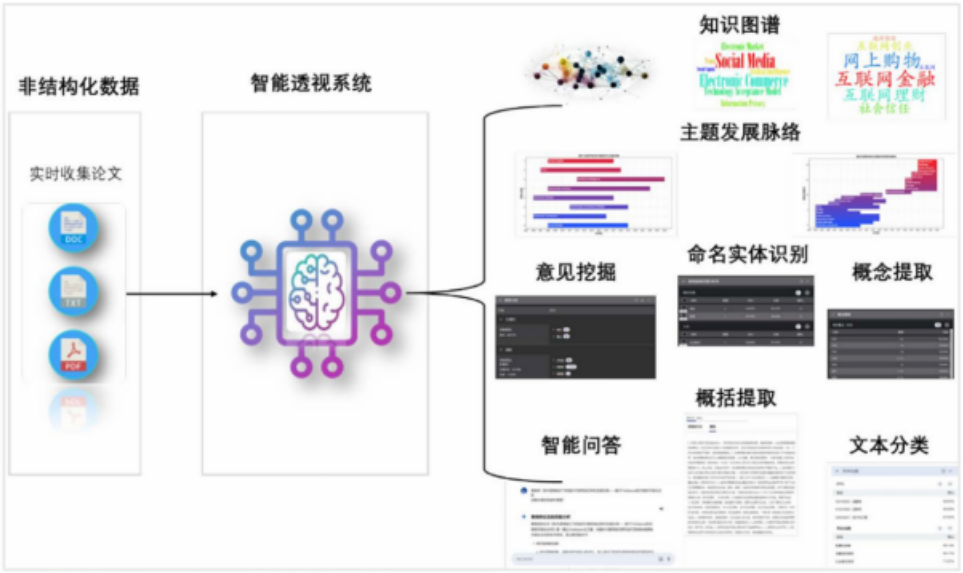

图 1. 赛题理解图示

【示例】

题目：从以上数据中遴选所需要的数据，聚焦您熟悉的某一领域（如人工智能），通过分析以上论文题录信息（如，论文名称，作者，作者单位，关键词，摘要等）。尝试找出该领域 20 位核心作者，并梳理作者之间关系，详细分析发文排名第一作者的教育、工作背景及研究脉络。

解决方案：

标题：基于 DeepSeek 大模型的挖掘与学术脉络分析

一、 研究目标

核心作者筛选：基于文献计量学与 DeepSeek 语义分析，精准识别作者和单位或主题信息。

合作网络构建：揭示作者间的跨机构合作模式及研究方向关联性。

脉络深度解析：追踪作者或单位的研究主题演化路径，生成技术突破时间轴。

二、 工具与技术

核心工具：DeepSeek：用于数据清洗、作者消歧、文本语义分析、研究脉络生成。

Excel：用于数据的筛选、统计分析。

辅助工具：Python（Pandas、NetworkX）、Gephi（可视化）。

数据：200 篇高引用量论文题录信息（标题、作者、单位、关键词、摘要）。

三、 实施流程

1） 数据结构化

基于 json 文件，利用 java 或 python 编程软件对 json 文件做结构化处理，按标题、摘要、关键词、作者、年度、作者单位、参考文件等字段生成 exel 表格进行结构化存储。

```python
import pandas as pd
import json

def load_data(files):
    all_articles = []
    for file in files:
        with open(file, 'r') as f:
            data = json.load(f)
            all_articles.extend(data)
    return pd.DataFrame(all_articles)

# 关键字段处理
df = load_data(['doc1.json', 'doc2.json', 'doc3.json'])[[
    'Title', 'Authors', 'PubYear', 'JournalTitle', 'Volume', 'Issue', 'DOI', 'Abstract', 'Keywords'
]]

# 作者字段格式化
df['Authors'] = df['Authors'].apply(
    lambda x: '; '.join([f"{a['Name']} ({a['Affiliation'].split(',')[0]})" for a in x])
)

# 输出Excel
df.to_excel("Merged_Articles.xlsx", index=False)
```

图 2 数据结构化处理

2） 数据筛选与清洗

基于 excel 数据表做初步分析，按研究专题、高频关键字、作者或作者单位等遴选出具有相同属性（如具有相同的研究领域、单位、作者等）的数据，保证数据聚焦，便于做数据分析。可以利用大模型、Excel 等工具对作者、研究单位、甚至关键字等内容进行统一化，消歧处理。

注：数据的数量可以根据原始数据的数量、或选择的主题自行确定，如 200 篇、500 篇、700 篇等。

3） 专题分析

基于以上遴选的数据确定分析专题，如若遴选的是同一个单位的数据，可以分析该单位重点研究内容、边缘研究内容、研究人员的合作关系、研究时序图，以及主要研究人员等。

- **逐年发文量分析：**

代码：

```
import pandas as pd
import matplotlib.pyplot as plt


# 假设数据已读取为 DataFrame，列名为'Year'
year_counts = df['Year'].value_counts().sort_index()


plt.figure(figsize=(10, 6))
```

```
plt.bar(year_counts.index, year_counts.values, color='#2c7bb6')
plt.title('发文数量趋势（2019-2025）', fontsize=14)
plt.xlabel('年份', fontsize=12)
plt.ylabel('发文数量', fontsize=12)
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.show()
```

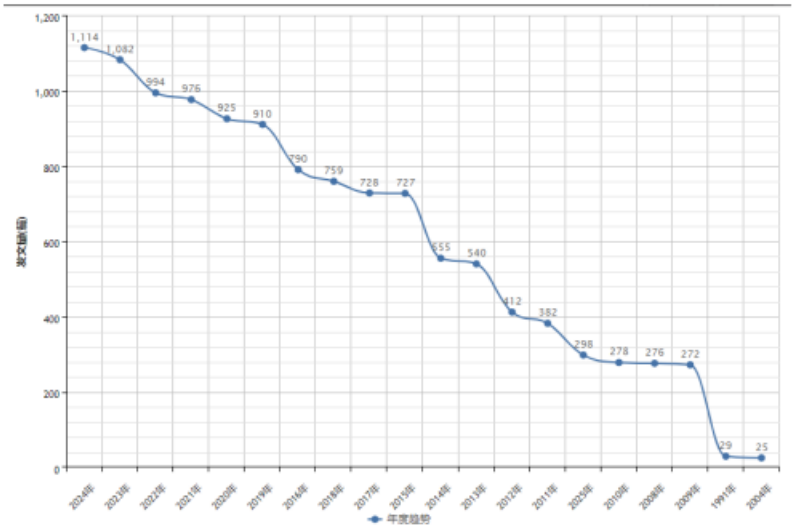

图 3. 年度发文趋势分析

- **研究主题分析**

代码：

```
from wordcloud import WordCloud


keywords = ';'.join(df['Keyword-关键词'].dropna()).split(';;')
text = ' '.join([kw.strip() for kw in keywords if kw])


wordcloud = WordCloud(font_path='simhei.ttf', width=800,
height=400, background_color='white').generate(text)
plt.figure(figsize=(12, 8))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
```

plt.title('研究主题关键词分布', fontsize=14)

plt.show()

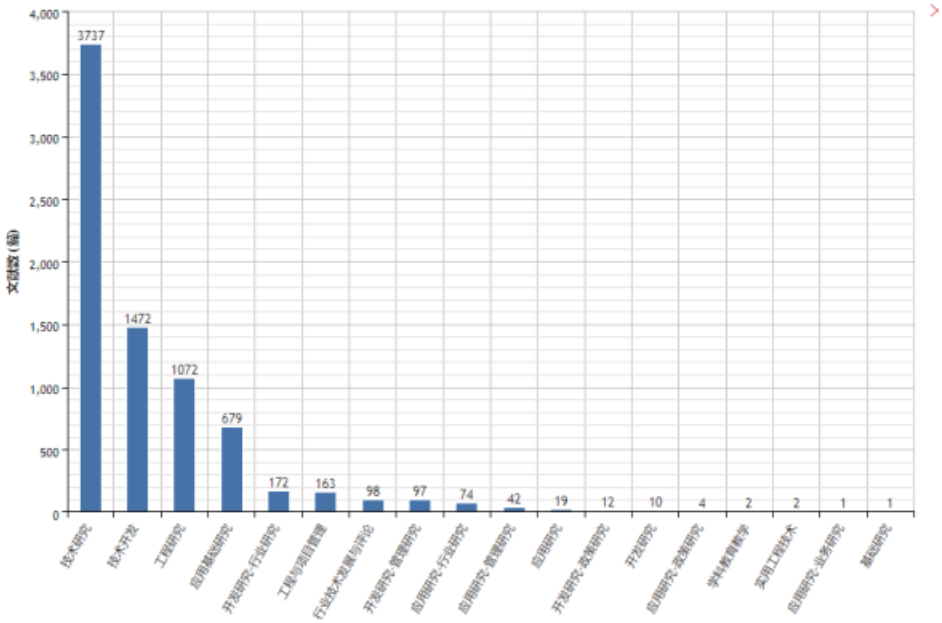

图 4. 主要研究内容分析

- **合作网络分析**

提取单位合作关系（示例：南京信息工程大学、西南交通大学等）

collaborations = df['Organ-单位'].str.split(';').explode().str.strip().value_counts().head(10)

plt.figure(figsize=(10, 6))

collaborations.plot(kind='barh', color='#d7191c')

plt.title('主要合作单位发文数量', fontsize=14)

plt.xlabel('发文量', fontsize=12)

plt.ylabel('单位', fontsize=12)

plt.gca().invert_yaxis()

plt.grid(axis='x', linestyle='--', alpha=0.7)

```
plt.show()
```

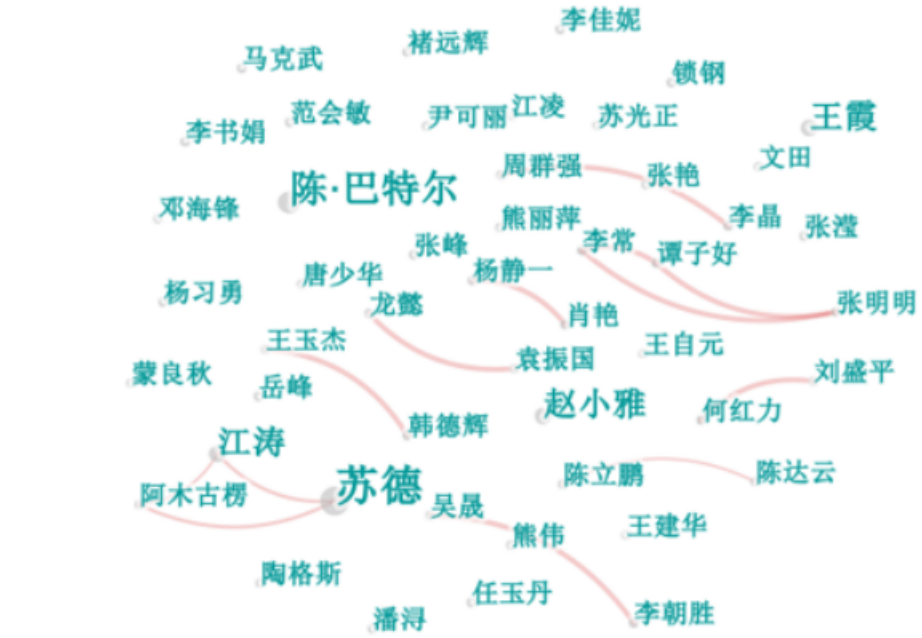

图 5. 作者分布分析

四、 输出成果

基于以上分析可利用 Python、Gephi、Echarts 相关技术进行绘图展示，并按图说话，对图进行解析。解读参考案例见 2017 年《铁道运输与经济》期刊论文《基于知识图谱的铁路运输管理工程领域研究热点及趋势分析》。