自然语言处理第一次作业（A）

作者：张宗桁
学号：2023K8009991013

2025 年 11 月 8 日

目录

1 样本的爬取过程

本次实验使用 Python 的 Scrapy 框架构建两个定制爬虫，分别采集英文科研论文摘要与中文科技新闻文本语料。本实验的主要采集逻辑由两个 Spider 类（`arxiv_spider.py` 与 `sina_tech.py`）实现，对应两个异质数据源：学术论文摘要与新闻报道。

1.1 $arXiv$ 论文数据爬取 (arxiv_spider.py)

采集目标：获取计算机科学领域近期论文摘要，限定在人工智能 (`cs.AI`)、计算语言学 (`cs.CL`) 与计算机视觉 (`cs.CV`) 等分类下的最新条目。整体流程概述如下：

1. 入口解析 (`parse`): 从 `start_urls` 指向的 `/recent` 页面启动，利用 CSS 选择器提取指向具体论文摘要页（`/abs/...`）的链接，并通过 `response.follow` 派发至摘要解析回调。
2. 摘要抽取 (`parse_abstract`): 在摘要页中定位 `<blockquote>` 标签并提取其纯文本内容，只抓取摘要。
3. 结构规整: 在抽取后进行基础清洗，以便后续英文文本规范化与统计处理。

1.2 新浪新闻数据爬取 (sina_tech.py)

该 Spider 针对新浪科技频道最新页面的资讯类内容：

1. 链接发现: 在首页使用 XPath 表达式筛选包含特征片段（`/doc-` 与 `.shtml`）的文章 URL，规避导航与多媒体类非正文页面。
2. 正文解析 (`parse_article`): 对每篇文章提取标题、发布时间、来源与主体段落等字段，确保生成结构化条目；解析时通过 XPath 精准定位，减少对页面样式变动的敏感性。

1.3 核心爬取配置 (settings.py)

在 `settings.py` 中配置关键参数：

1. 访问友好策略: `ROBOTSTXT_OBEY = True` 强制遵守目标站点的爬取协议； `DOWNLOAD_DELAY = 1` 设置 1 秒请求延迟，减少被限速或封禁风险。
2. User-Agent 伪装: 设置模拟常见 Chrome 浏览器标识的 `USER_AGENT`，提升页面响应一致性并减少触发反爬策略的概率。
3. 编码统一: 明确 `FEED_EXPORT_ENCODING = "utf-8"`，保证中文文本在导出阶段不出现乱码，方便后续跨平台处理与统计。

2 样本的清洗

本实验针对中英文语料的语言特性差异与噪音模式，分别设计差异化清洗策略，主要依托 Python 的 BeautifulSoup 库（HTML 结构解析）与正则表达式库（模式匹配与文本规整）实现。

清洗流程遵循"粗清洗 → 精清洗 → 标准化"的层次策略：首先移除明显的非文本元素与格式标记，其次针对语言特性进行精细化过滤，最后统一编码与空白符。

2.1 中文文本处理流程

针对新浪科技新闻的中文语料，采取如下精细化清洗策略：

1. 结构化噪音过滤：使用正则表达式移除网页特有的冗余信息，包括：
   - 商业广告植入（如"炒股就看金融界"等营销文案）
   - 技术残留标识（如"Flash Player"、"JavaScript"等系统提示）
   - 法律免责条款与版权声明
2. 元信息剔除：移除新闻生产流程中的编辑元数据，具体包括：
   - 来源标注：（来源：[机构名称]）
   - 责编信息：（责编|记者：[姓名]）
   - 时间戳与 URL 链接
3. 字符规范化：清理格式控制符与异常字符：
   - 移除零宽字符、制表符及非打印控制字符
   - 压缩连续空白为单一空格

2.2 英文文本处理流程

arXiv 论文摘要的英文文本清洗更注重学术写作规范化与统计一致性：

1. LaTeX 残留清除：移除论文排版过程中的格式化标记：
   - 移除反斜杠转义序列（`\s*` 等）
   - 清理 `\textbf`、`\textit` 等字体控制命令
2. 大小写标准化：将全文转换为小写，消除因首字母大写、专有名词等导致的词频统计偏差。
3. 字符集约束：
   - 仅保留英文字母（a-z）与空白符

- 移除数字、标点符号与特殊符号

4. 空白符规整：将连续空白字符（空格、制表符）压缩为单一空格，避免词间距不一致影响切分。

5. 去重优化：采用集合数据结构（`set`）记录已处理行，避免重复内容对词频统计造成偏差。

2.3 语料库构建结果

经过上述差异化预处理，最终构建了六个分层语料库，形成中英双语、三级规模的矩阵化数据集：

- 小规模语料：中文 $\approx$ 150KB，英文 $\approx$ 700KB
- 中等规模语料：中文 $\approx$ 250KB，英文 $\approx$ 900KB
- 大规模语料：中文 $\approx$ 350KB，英文 $\approx$ 1.5MB

3 符号概率分布与信息熵分析

信息熵的数学定义为：

$$H(X) = -\sum_{i=1}^{n} P(x_i) \log_2 P(x_i)$$

其中 $P(x_i)$ 表示符号 x_i（汉字或英文单词）在语料中的出现概率，n 为不重复符号的总数。熵值以比特（bit）为单位，数值越高表示符号分布越均匀，文本的不确定性与信息密度越大。

本实验对中英文采用不同的统计单元：中文以"汉字"为原子符号进行字级统计，英文以"单词"为基本单元进行词级统计。

3.1 中文汉字熵分析

3.1.1 结果

排名	汉字	次数	概率
1	的	1503	2.7301%
2	在	531	0.9645%
3	为	449	0.8156%
4	一	435	0.7902%
5	能	366	0.6648%
6	中	359	0.6521%
7	是	309	0.5613%
8	了	308	0.5595%
9	大	302	0.5486%
10	成	281	0.5104%
11	人	272	0.4941%
12	年	268	0.4868%
13	同	263	0.4777%
14	业	258	0.4686%
15	发	258	0.4686%
16	有	256	0.4650%
17	用	245	0.4450%
18	新	244	0.4432%
19	国	239	0.4341%
20	品	239	0.4341%

图 1: 小规模样本的汉字频率和信息熵

排名	汉字	次数	概率
1	的	2186	2.6710%
2	在	690	0.8431%
3	业	637	0.7783%
4	一	608	0.7429%
5	为	578	0.7062%
6	年	566	0.6916%
7	不	557	0.6806%
8	是	555	0.6781%
9	行	491	0.5999%
10	中	490	0.5987%
11	发	470	0.5743%
12	人	441	0.5388%
13	生	441	0.5388%
14	有	434	0.5303%
15	能	426	0.5205%
16	上	414	0.5058%
17	大	389	0.4753%
18	了	381	0.4655%
19	国	379	0.4631%
20	学	368	0.4496%

图 2: 中规模样本的汉字频率和信息熵

排名	汉字	次数	概率
1	的	3382	3.1309%
2	在	1047	0.9693%
3	一	901	0.8341%
4	是	855	0.7915%
5	业	854	0.7906%
6	年	757	0.7008%
7	为	688	0.6369%
8	大	635	0.5878%
9	不	608	0.5629%
10	有	607	0.5619%
11	中	590	0.5462%
12	能	581	0.5379%
13	人	574	0.5314%
14	了	561	0.5193%
15	上	550	0.5092%
16	资	538	0.4981%
17	理	496	0.4592%
18	场	474	0.4388%
19	产	466	0.4314%
20	成	460	0.4258%

图 3: 大规模样本的汉字频率和信息熵

3.1.2 横向对比

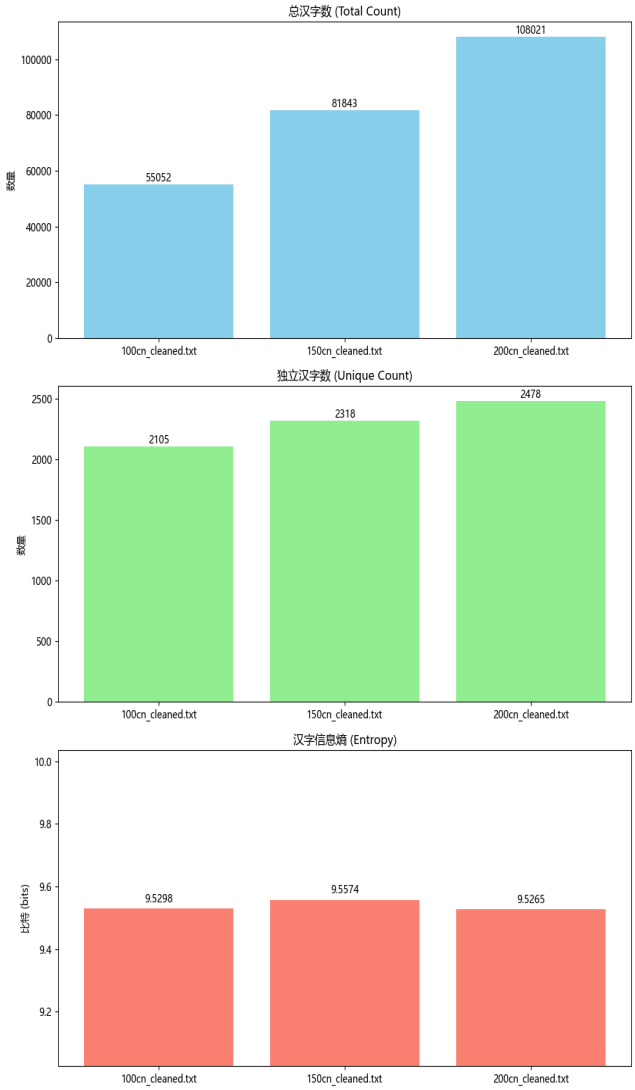

图 4: 横向对比

分析 根据图表数据分析，`100cn_cleaned.txt`、`150cn_cleaned.txt` 和 `200cn_cleaned.txt` 三个文本文件的汉字信息熵变化呈现先升后降（9.5298 → 9.5574 → 9.5265）的非线性趋势。

这一现象清晰地表明，影响信息熵的核心因素是文本的“主题内容”及其决定的“汉字概率分布”，而非文本规模本身。下面我们分阶段进行详细分析：

第一阶段：从 100cn 到 150cn

1. 数据表现：在此阶段，总汉字数（55,052 → 81,843）、独立汉字数（2,105 → 2,318）和信息熵（9.5298 → 9.5574）均呈现增长。
2. 分布变化：这是一个符合预期的增长。文本规模扩大，带来了 213 个新的独立汉字，引入了更

多样化的词汇和表达。关键证据是，最常用字“的”的概率反而轻微下降了（从 2.7301% → 2.6710%）。

3. 结论：当一个文本库在扩展时，如果新加入的内容主题多样，会引入新的词汇，同时稀释最常见词汇的占比。这种多样化使得汉字分布更趋于均匀，系统的不确定性随之增加，因此信息熵上升。

第二阶段：从 150cn 到 200cn

1. 数据表现：在此阶段，总汉字数（81,843 → 108,021）和独立汉字数（2,318 → 2,478）依然在增加，但信息熵却反常下降了（从 9.5574 → 9.5265）。

2. 分布变化：熵下降的直接原因是汉字概率分布变得更不均匀。最显著的证据是：

   - 排名第一的汉字“的”，其出现概率从 2.6710% 显著飙升至 3.1309%。
   - 排名第二的“在”，概率也从 0.8431% 上升到 0.9693%。

3. 理论解释：根据信息熵的定义 $H = -\sum p(x) \log p(x)$，当某个单一事件（如“的”字）的概率 p 显著提高时，它对总熵的贡献会减小，导致整体系统的不确定性（熵）降低。换言之，当“的”字出现得如此频繁，系统就变得“更可预测”了。

4. 主题偏移：词频表（Top 20）的变化揭示了熵下降的根本原因——“主题偏移”。

   - `150cn` 的高频词包括“行”（第9）、“生”（第13）、“学”（第20）。
   - `200cn` 中，这些词跌出前20，取而代之的是一组主题高度集中的新词：“资”、“理”、“场”、“产”。

5. 结论：这表明从 150cn 到 200cn 新增的文本内容，大概率是高度集中在“商业”、“经济”或“管理”领域。这种高度专业化、主题趋同的文本，不仅引入了自己领域的高频词，也导致了“的”、“在”等通用字的用法频率激增。这种现象使得整个文本的汉字分布变得更加倾斜和不均，最终导致了信息熵的下降，尽管文本的总量和词汇量仍在增加。

3.2 英文

对于英文，我们以“单词”（Word）为单位进行统计。

3.2.1 结果

排名	单词	次数	概率
1	and	3469	3.5039%
2	the	3262	3.2948%
3	a	2926	2.9554%
4	to	2241	2.2635%
5	of	2096	2.1171%
6	in	1527	1.5423%
7	we	1245	1.2575%
8	for	1224	1.2363%
9	that	1158	1.1696%
10	model	1148	1.1595%
11	on	912	0.9212%
12	this	818	0.8262%
13	with	798	0.8060%
14	is	675	0.6818%
15	by	618	0.6242%
16	our	469	0.4737%
17	from	444	0.4485%
18	llm	442	0.4464%
19	reasoning	426	0.4303%
20	are	418	0.4222%

图 5: 小规模样本的单词频率和信息熵

排名	单词	次数	概率
1	and	4459	3.5477%
2	the	4075	3.2422%
3	a	3648	2.9024%
4	to	2875	2.2874%
5	of	2539	2.0201%
6	in	1962	1.5610%
7	that	1544	1.2284%
8	we	1516	1.2062%
9	for	1501	1.1942%
10	model	1487	1.1831%
11	on	1126	0.8959%
12	with	1067	0.8489%
13	this	1017	0.8091%
14	is	783	0.6230%
15	by	760	0.6047%
16	llm	637	0.5068%
17	our	620	0.4933%
18	reasoning	616	0.4901%
19	from	589	0.4686%
20	task	542	0.4312%

图 6: 中规模样本的单词频率和信息熵

排名	单词	次数	概率
1	and	6863	3.4658%
2	the	6658	3.3623%
3	a	5703	2.8800%
4	to	4404	2.2240%
5	of	4142	2.0917%
6	in	3040	1.5352%
7	for	2459	1.2418%
8	we	2432	1.2282%
9	model	2372	1.1979%
10	that	2318	1.1706%
11	on	1806	0.9120%
12	this	1646	0.8312%
13	with	1633	0.8247%
14	is	1315	0.6641%
15	by	1207	0.6095%
16	our	934	0.4717%
17	from	893	0.4510%
18	method	816	0.4121%
19	data	809	0.4085%
20	an	807	0.4075%

图 7: 大规模样本的单词频率和信息熵

3.2.2 横向对比

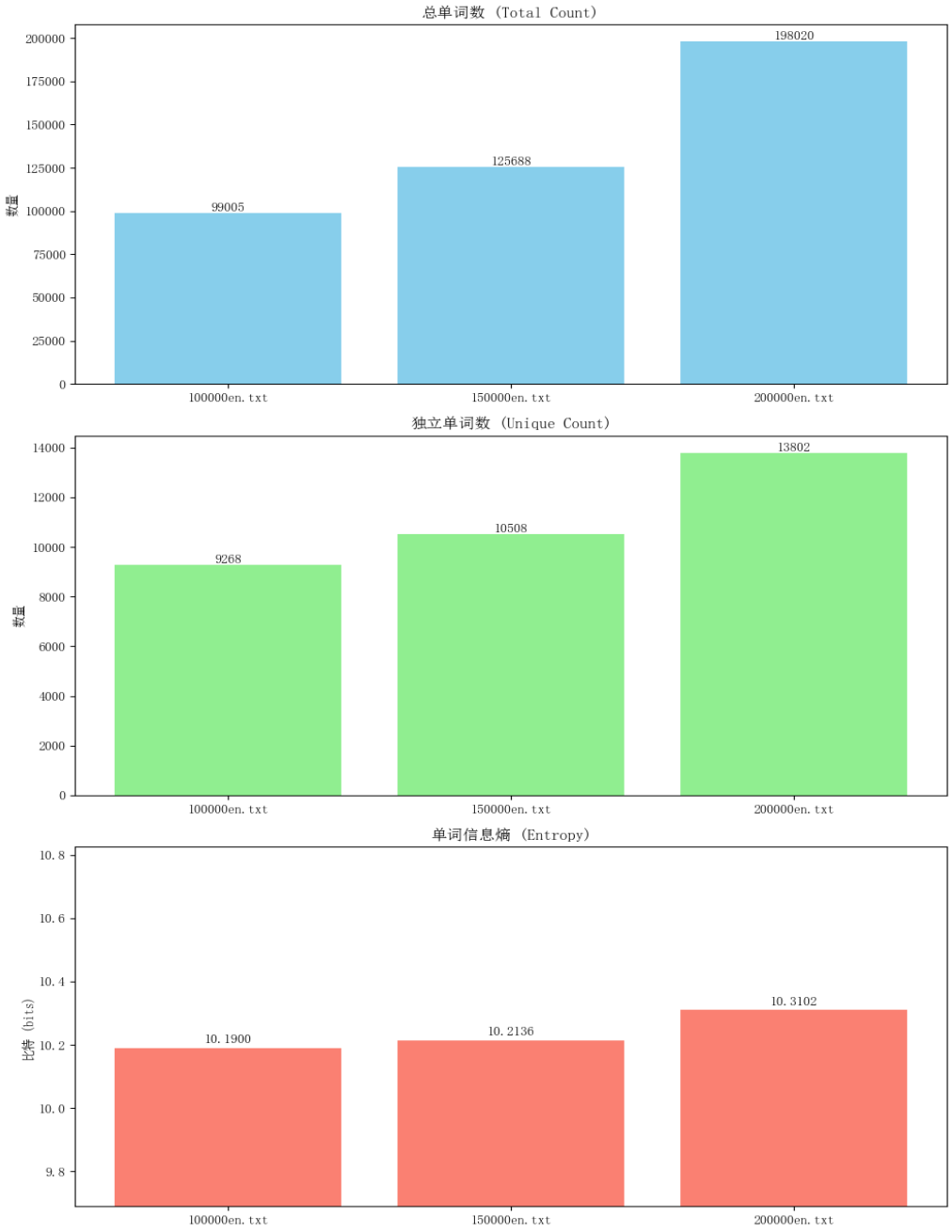

图 8: 横向对比

分析 基于三组英文样本（100K/150K/200K）的统计结果，可以得到以下观察与结论：

1. 信息熵：单词信息熵 10.1900 → 10.2136 → 10.3102，稳步上升，说明词频分布整体更加均匀，即不确定性增加。

2. 高频词结构： Top-20 长期由功能词 and, the, a, to, of, in 等主导。随着规模扩大，model, llm, reasoning, method, data, task 等领域词逐步进入Top-20，显示样本主题对 AI、LLM 等研究语

域的偏向。

3. 头尾部变化：头部功能词的相对占比基本稳定或略有下降，而尾部长尾扩展更明显——这是熵上升的主要来源。

综上，随着语料规模的扩大，英文样本呈现出“熵上升、主题更加凸显”的特征；这符合语言统计的一般规律，也反映了采样来源对语域的影响。

4 验证齐夫定律 (Zipf's Law)

齐夫定律指出，在一个大型语料库中，任意单词的出现频率 f 与其在频率表中的排名 r 成反比，即：

$$f \propto \frac{1}{r}$$

取对数后，$\log(f)$ 和 $\log(r)$ 应呈线性关系。我们仅对英文样本进行单词级别的验证。

验证方法 使用 Python 实现验证逻辑：首先统计单词频率并按频率降序排列。然后计算词频与排名的对数值，通过 `numpy.polyfit` 进行线性拟合，获得斜率参数。

4.1 小规模样本

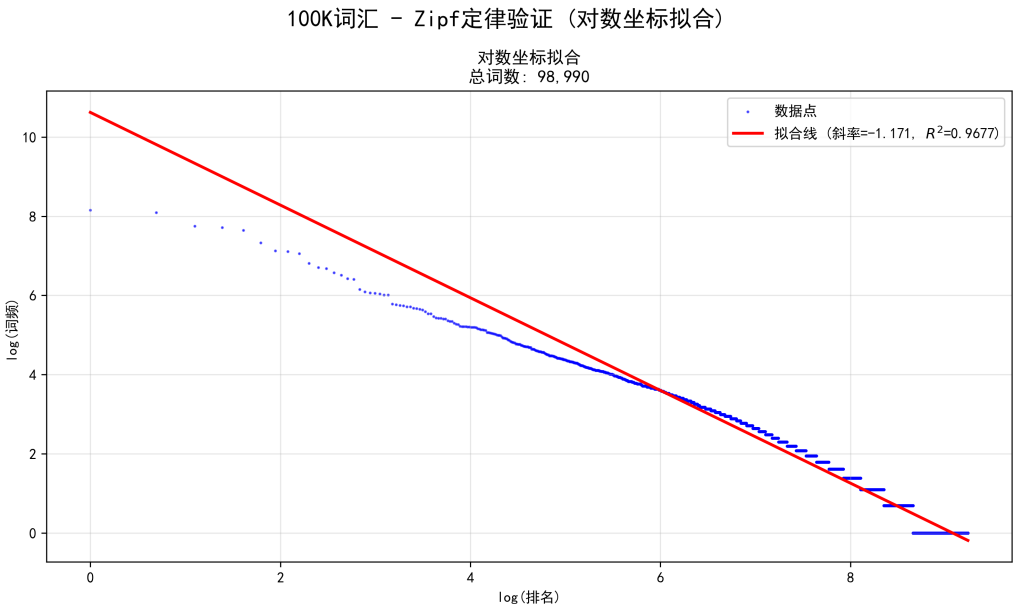

图 9: 小规模样本的 Zipf 定律对数拟合图

4.2 中规模样本

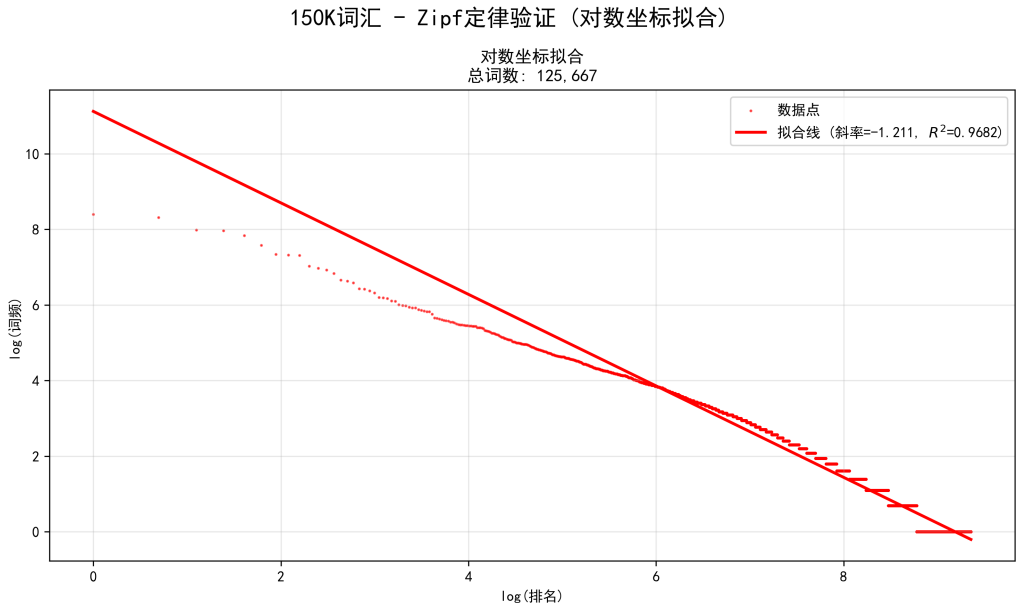

图 10: 中规模样本的 Zipf 定律对数拟合图

4.3 大规模样本

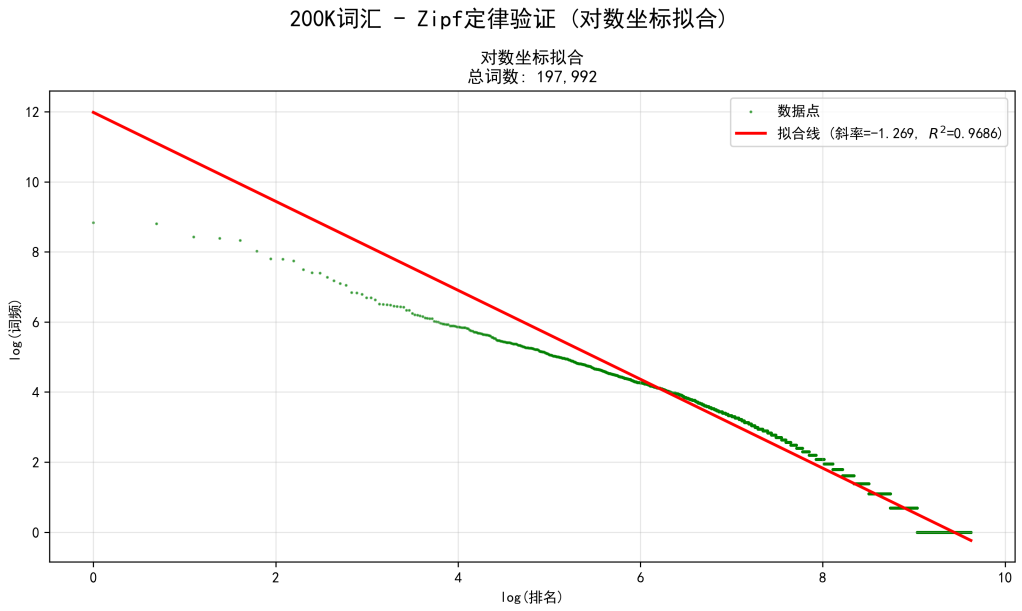

图 11: 大规模样本的 Zipf 定律对数拟合图

分析 从对数坐标的拟合结果看，三个规模的语料库均表现出词频与排名近似线性相关的特性，这与齐夫定律的预测一致。具体而言，100K、150K 和 200K 词汇语料的回归斜率分别为 -1.171 ($R^2 = 0.9677$)、-1.211 ($R^2 = 0.9682$) 和 -1.269 ($R^2 = 0.9686$)。

实验结论与偏差分析：

1. 基本符合理论：所有样本的拟合直线斜率都接近理论值 -1，同时 R^2 均接近 1，表明词频与排名的对数关系具有较高的线性相关性。
2. 高频词区域：在所有图中，曲线的头部（高频词区域）都明显高于拟合的直线。这主要是因为大量的冠词、介词等功能词（如 "the", "of", "a"）占据了频率的顶端，其出现频率远超普通内容词，导致了分布的陡峭化。
3. 低频词区域：曲线的尾部出现了明显的向下弯曲和“台阶”状结构。这种现象源于以下几个原因：
    - 有限样本：在有限的语料中，许多稀有词只出现了一次或极少数次，导致在对数坐标尾部形成离散的、水平的“台阶”。
    - 数据离散化：排名越靠后，词频越低，频率值的变化不再是连续的，这种离散效应在对数图上被放大。