哈爾濱工業大學
HARBIN INSTITUTE OF TECHNOLOGY

自然语言处理

实验一：汉语分词系统

School of Computer Science and Technology

Harbin Institute of Technology

1 实验目标

本次实验目的是构建一个汉语分词系统。在课堂学习汉语自动分词原理和方法基础上，全面掌握汉语分词的若干关键技术，包括从词典的建立、分词算法的实现、性能优化和评价等环节。

本次实验所要用到的知识和技能如下：

语料库相关知识
正反向最大匹配分词算法
N 元语言模型相关知识
分词性能评价常用指标
基本编程能力（文件处理、数据统计等）
相关的（查找）算法及数据结构实现能力

2 实验环境

编程语言：C/C++、python、或者Java(任选)
其他无特殊要求

3 实验内容及要求

训练集：199801_seg&pos.txt（1998 年1 月《人民日报》的分词语料库）
2021 年秋季学期开始增加1个月的的训练数据；
人名地名资源

可以使用编码表（如Unicode,GBK，ASCII）里的所有字符。
不得使用第三方的数据和词典。

最终测试集：格式参见199801_sent.txt，内容来自多源多领域文本。均经过手工标注。

注意：因数据版权约定，同学不可将所有接收到的数据透露给任何其他同学和单位，禁止用于任何非学习目的。原则上应在完成实验后，删除所接收到的全部训练数据。

3.1 词典的构建

输入文件：199801_seg&pos.txt
输出：dic.txt（自己形成的分词词典）
提交要求：
1）本节需提交dic.txt；
2）最终实验报告中须说明分词单位的标准、以及词典文件格式；同时，要求对自己所构建的词典进行分析；

{提示：所提取词典没有要求一定写代码完成；

对词典的分析这里设置为一个开放问题，只要从实用的角度进行分析，分析手段得当，结果有价值即可。}

3.2 正反向最大匹配分词实现

输入文件：199801_sent.txt（1998 年1 月《人民日报》语料，未分词）
dic.txt(自己形成的分词词典)
输出：seg_FMM.txt 和seg_BMM.txt(正反向最大匹配分词结果，格式参照分词语料格式，形如“词/_词/_.....”，“_代表空格”)

编程要求：
自己定义词典的数据结构，编写写词典查找算法。不允许使用类似list, dict (python 特例允许使用list)等编程语言内置的数据结构。
鼓励最少代码量的系统实现

提交要求：1）本节需提交seg_FMM.txt 和seg_BMM.txt；
2）本节需提交程序源代码；
3）最终实验报告中，须说明程序实现过程中的收获；

{提示：写最少的代码，快速通过}

3.3 正反向最大匹配分词效果分析

输入文件：199801_seg&pos.txt（1998 年1 月《人民日报》的分词语料库）seg_FMM.txt（3.2节输出）、seg_BMM.txt（3.2节输出）
输出：score.txt，包括精确率(precision)、召回率(recall)、F 值结果
编程要求：
自己编写评价代码
保证评价结果的正确性
提交要求：1）本节需提交score.txt；
2）本小节不检查代码，不用提交评价工具；
3）最终实验报告中，须分析正反向对大匹配在分词精度上的差异，分析角度独特有加分（最终实验成绩上最多加3 分）；

评分提示：评价结果的误差,将影响本次实验最终成绩。例如,在精确率指标上,自己计算结果为0.96,最终核查结果为0.97,|0.96-0.97|*100=1,则本次实验成绩最终得分将被扣除1 分。这里的误差包括“精确率误差+召回率误差”，不再考虑F 值的误差。

{提示：看似简单，但是很多同学修改这段代码的时间比初次完成的时间要长---如果自己写，不用内置的函数。祝早日通过}
{针对实验环节中出现的时机问题，补充提示如下：如果采用本小节输出的

score.txt 来讨论分词性能，是很不严谨的，将被扣除本小节50%的得分}

3.4 基于机械匹配的分词系统的速度优化

输入文件：199801_sent.txt（1998 年1 月《人民日报》语料，未分词）

输出：TimeCost.txt（内含分词所用时间）

提交要求：1）本节需提交TimeCost.txt（应包含优化前后的分词耗时）；

2）本节需提交程序源代码；

3）最终实验报告中，须详细描述所实现的优化方案，分析优化技术的效果，尝试提出自己现有代码基础上进一步优化分词速度的思路；

编程要求：

任选前后向最大匹配分词算法其中之一，尽可能对分词系统速度优化，底线要求是实现二分查找，鼓励实现更优的方案；

禁止使用开发环境内置的数据结构，查找算法和数据结构都要求独立实现；

程序初始化时间不考虑在内，仅计算从分词过程开始到分词结果输出完成的耗时；

{提示：挑战索引结构，比如哈希什么的（找到恰当的哈希函数不太容易）；有同学直接手写实现了双Trie 树结构，很惊艳；}

3.5 基于统计语言模型的分词系统实现

输入文件：test.txt（未分词的最终测试集，多种来源格式形如199801_sent.txt）

训练数据：199801_seg&pos.txt（注：2021 年秋季开始加发另一个月的标注数据）、汉语人名资源

输出：seg_LM.txt（利用统计语言模型分词结果，格式参照分词语料）

编程要求：

根据训练语料 建立随后需要使用的统计语言模型；

使用动态规划，实现全切分有向图的搜索；

至少使用一元语言模型（最大词频分词），这是本节的及格线。

期待实现基于二元语言模型的分词系统（实现此环节，可以不用实现一元语言模型的分词系统）；

鼓励实现更稿阶的n元语言模型（可不用实现低阶的语言模型）的分词系统

提交要求：1）能够读入指定的测试文件，输出文件为seg_LM.txt；

2）分词程序涉及的全部源代码；

3）实验报告：须对程序中的重点实现代码进行说明（可用流程图对算法进行辅助说明）；对比分析所使用的不同分词方法的性能；

{提示：一元文法挺有效。二元文法难在参数平滑，程序实现也更复杂，但是性能如果没超过1元文法，肯定是做的有问题；最大的福利：所有编程的限制取消，编程语

言的内置函数、库，放开使用}

3.6 分词结果的再优化（本节简称刷榜环节）

这一节不是强制完成环节，属于分词处理中的高水平要求，请同学们根据自己的时间和精力安排。鼓励同学尝试通关。

实现未登录词识别功能；

鼓励通过分词结果后处理实现性能优化；

鼓励通过多分词系统的系统集成实现性能优化；

本节可以使用任何你能驾驭的方法模型，只要是能证明是自己的方法（关键的环节是自己的实现），而不是纯使用第三方工具（调API、封装别人的接口）。

特别的，最终输出结果的技术环节，必须是自己编程实现的，而不能由第三方工具或接口来完成。

提交：1）含有未登录词识别功能的代码应单独提交一个版本，作为评分依据；

2）其他环节分词程序涉及的全部源代码；

3）所涉及的第三方工具及源码（如有）；

4）必要的系统使用说明（如果需要）

5）本节可以提交最多不超过3个分词优化系统，取性能最高作为最终评分依据。

特别提示：

1）本实验不是开放竞赛，3.6节最终提交的代码中，必须包含uni-gram 或者bi-gram分词结果，包含自己实现的未登录词识别模块，是在这些工作上的优化；

2）在最终的性能冲刺中，单独调用某个的第三方分词工具、某个工具包，是不被认可的；

3）在系统集成中，如果实际上输出的就是其中某个分词工具的结果，性能将不被认可。

4 实验报告

不要流水账；不要缺失应有的参考文献；

推荐按照ACL 论文的内容安排撰写，凝练自己工作的核心（发现、贡献），巧妙地将上述实验结果，自己的设计、心得，写出来。

使用按照ACL 会议论文排版格式，网上有模板。

正文部分不允许出现源代码，在说明问题时可使用伪代码（如需附代码，请使用附录）

请确保实验报告格式清晰、一致，内容的条理性和完整性。

每个小组单独提交1份实验报告，其中要求个人独立完成的部分，没人1节分别撰写自己的内容。

5 实验评分

1）该实验成绩=编程实现成绩+报告成绩

2）编程实现考评环节：满分12 分

6 分：3.3 完成，要求个人独立完成；
7 分：3.4 完成，要求个人独立完成；
8 分及以上：3.5 完成，小组成员不超过3 人，根据完成度和贡献度确定分数；

完成度评分：正确完成动态规划，以1 元语言模型输出结果，评分8；
在上述基础上，以2 元语言模型实现分词系统，评分9；
在上述基础上，正确进行了未登录词识别（至少性能获得进一步提升），评分10；
在上述基础上，进一步采取了性能优化手段，以最高性能记为12 分，其余根据性能差异，按比例取得；
（即，实现未登录词后，才能按照性能获得10.5，11，11.5，12分这四档评分）。

贡献度评分：小组内每人预分配3 分，根据组内贡献度，最终决定每人得分；要求每人贡献度得分不能相同，分数总和等于3*n（n 为小组人数）；
最后每人的贡献度得分转换为最高分为[0,3]的标准分。

3）报告成绩:5 分
内容完整
格式规范
包含所使用的参考文献[重要]
参考ACL 会议论文模板，内容安排和呈现方式越贴近ACL 要求，得分越高没有一定的页数限制，建议排版后正文在4-8 页之间。

最后的提示：独立完成，认定的依据是“应提交的代码和报告都应分别有物理存在，并按时提交”。

6 提交方式（2021秋）

中期验收：11.27,23:59’59”(完成并提交到3.4 节的实验成果)
最终截止日期：12.5,23:59’59” (完成并提交到3.5 节及其后所要求的代码及实验报告)
提交方式：提供所要求的代码和报告,可以通过邮件、QQ 发送,或者网盘云文件的形式。
由助教确认收取。