ICTCLAS Free 版本修正报告

中科院计算技术研究所多语言交互技术评测实验室

黄瑾 huangjin@ict.ac.cn

1． 原始版本信息

ICTCLAS Free版本于 2002 年 8 月 16 日发布于中文自然语言处理开放平台（http://www.nlp.org.cn/）并于 2002 年 9 月发布相应的论文及测试报告。测试报告主要包括国家 973 英汉机器翻译第二阶段的评测报告及在 1998 年 1 月标注人民语料库上的自评结果。

2． 修正 bug 类型及修正结果

ICTCLAS Free 版本的错误修正主要根据论坛(http://www.nlp.org.cn/tree.php?cid=12&topic=词法分析)上的若干错误报告进行针对性的修改。具体 bug 见下表：

序号	bug 描述	引起 bug 的原因	解决办法	状态
1	连续的"//////////"字符串会引起系统崩溃。(事实上两个以上连续的/字符就会引起系统崩溃)	对于/没有处理引起数组越界	加入对于字符'/'的判断	已修正
2	部分"<.+?>"格式的字符串会引起系统崩溃	对于这些特殊字符没有处理引起个别时候数组越界	每个句子处理之前重置 buffer 的值避免以前的处理结果影响新的句子处理结果	已修正，实验中使用特殊字符未重现该错误
3	");D(["ce"]"));D(["ms" 处理上述存在特殊字符的句子时系统崩溃	同上	同上	已修正，使用这个输入系统正常输出
4	分解字符串包含大于 15 个长度的"="系统发生崩溃	为切分标注结果字符串分配存储空间时分配空间不足导致数组越界及溢出	根据最坏情况以 4 倍长度重新分配空间	已修正
5	文本比较大时系统发生	部分分配内存未	部分以 new XX[]	已修正

	崩溃	能正常释放	分配的内存没有使用 delete []XX 的格式进行释放，修正此类问题。	
6	读文件,每次读取 64K, 如果刚好把一个句子弄断了,没有处理.	读入时未判断是否存在截断情况	当读入发生截断时,回退一个字符而不是多读一个字符(为了避免重新分空间和因此可能引起的其他 buffer 大小不足而溢出)	已修正但不能真正解决问题，可能由于切割造成句子切分错误。另外在极端的输入情况下原先的 buffer 大小仍然会造成其他 buffer 的溢出，因此减小了读入段落的大小。

其他在代码查看及调试过程中发现的 bug

1	对……的切分结果有误	读入段落并进行句子切割时将……切开了因此输出结果中识别为…/w …/w	切割段落中的句子时对……特别处理，不要切开	已修正
2	19980101-01-001-007 在新版中的被切散	原始版本不区分-和半角数字因此类似于 19980101-01-001-007 的输入串不会被分割，修正后区分这两种类型的字符会造成被切散	添加半角分割符一项，在 GenerateWordNet 中再将其正确合并	已修正
3	月份、点钟、刻钟一类词被切散	词典概率“月份“竞争不过“X月”而被切开，同时在判断是否为时间类型时也未处理这几个后缀造成切散	在 GenerateWordNet 函数中处理不令其切割开，同时加入判断使得此种后缀被识为时间词后缀	已修正
4	１５２．４１５分/t １５２．４１５/m 分/q	判断是否为时间词串时只是简单的判断是否具有时间词串后缀“月日时分秒”	在判断为时间词串之前要求上述关键词前的数字串中应该不含有“．．”以及长度应该不超过6个字符。	长度不超过6个字符只是简单的规则，更好的解决办

				法是判断是否为合法的时间类型词数字串，例如“月”的取值应该在1到12之间，也有可能是“一”到“十二”，或者全角的数字1到12
5	部分结构体未初始化	含有指针变量的结构体变量没有初始化可能造成对非法指针地址的引用	添加结构体的构造函数，将成员变量置零	已修正，主要针对含有指针变量的结构体
6	部分数组变量未初始化	主要是字符串数组未进行初始化可能造成字符串操作时非法访问	对部分字符串数组在定义时即置为空	已修正
7	log 函数在.net 环境下编译不通过	新版本在.net 环境下进行调试，由于 log 函数中数据类型不定造成编译不通过	将操作数强制转换为 double 型即可，不影响程序运行结果	已修正
8	对半角数字和全角数字混排时处理有误。	原始版本不特别区别半角数字串，对于“19980101-01-001-007１９９７年”这样的输入半角数字串和全角数字串被连了起来而不是正常切开	在判断类型时加入了半角数字并同时修正判断是否均为合法数字串的函数 IsAllNum	已修正，全角和半角数字混排时会被切割开识别为不同的数字值。

9	部分数词切割结果有误	在原子切割时被分开而合并时由于为部分串不能正确被识别为符合数字格式的串而被切开	在 AtomSegment 函数中就进行数字串的基本判断，避免被切割	已修正，但只针对全角半角阿拉伯数字和符号的情况
10	二 〇 〇 三 年 二〇〇三年	判断数字串时漏掉了“ 〇 ”，注意与“ ○ ”不是同一个字符	在判断数字串的相应地方添加“ 〇 ”的情况	已修正
11	－/w ５．３/m －５．３/m	判断数字串时漏掉了“ － ”，注意与“ — ”不是同一个字符	在判断数字串的相应地方添加“ － ”的情况	已修正
12	４２万/m 亿/m ４２万亿/m	函数 IsAllNum 中对于多个汉字后缀的情况没有处理	修改 IsAllNum 函数，添加这种情况的判断	已修正
13	浇/v 上一/m 浇上/v 一/m	函数 IsAllChineseNum 中对于”上”为前缀的情况没有特别处理，使得数字前缀+数字被识别为一个数词	对“ 上 ”和“ 成 ”这两个常见的数字串前缀，要求其后面跟的数字汉字为“ 百千万亿佰仟 ”	已修正
14	19980118-04-008-007 ＊＊/m 19980118-04-008-007/m ＊/w ＊/w 19980120-09-004-001 强强/m 19980120-09-004-001/m 强强/nr	在针对重叠词进行扩展的规则部分，使用 ABB 的规则将诸如“ 一段段、一片片 “ 的词合起来，所使用的规则仅为前面一个为数字，后面两字相同。	增加规则，使得前面的所谓数词应该为汉字且长度不应该超过一个汉字；同样发生修改的还有 AA 型重叠词，要求必须为汉字(否则多个＊就会被两个两个切成一对)	已修正
15	部分数词串后面跟了明显不为数词串的其他汉字	在判断数字串后缀时，使用了 strstr 函数而不是 CC_Find 函数来查看是否存在某个特定的全角字符，由于多个字符串错位组合形成了一个真正的汉字而被误判	将应该使用全角判断的地方均改为 CC_Find。	已修正

16	部分非数字单个汉字被识别为 m 类型，例如第、上、成等	主要原因为判断是否为数字串时允许这种前缀单独出现的情况而造成单字也被识别为数字类型	对最终生成的候选数字串进行判断，如果只有一个这种前缀字则重新生成其词性	已修正
17	CDynamicArray 的赋值操作符没有对自赋值的判断	operator =重载应该首先判断是否为 this	加入对 this 的判断	已修正
18	连续的六个或者三个半角点被全部切开	属于不规范的输入，因此全部被判断为普通的分隔符并切割开	加入特殊处理使得此类连续的点号被识别为省略号	已修正
19	两三年工夫医院就悄没声儿地火爆起来。 两三年/t 工夫/n 医/j 院/j 就 悄 没 声 儿 地 火爆 起来 。/w 另外一个例子： 演的是王允争取吕布共诛董卓的著名历史故事。 演/v 的/u 是/v 王/nr 允/nr 争取/v 吕/nr 布/nr 共/u 诛 董卓/拃 的/u 著名/a 历史/n 故事/n 。/w	丢失部分词性标注信息，其原因是“悄”字没有单字的词性，在词性标注的 Viterbi 算法解码时数组索引越界。虽然在生成 1-best 时全部标注了词性信息，但是在生成 10-best 时最好结果丢失部分词性标记。 综合另外一个实例，对于没有查到相应词性的标注结果不可预料，有可能丢失有可能是乱码。	最后进行词性标注时（已进行了命名实体识别）如果没有词性信息则判断为’x’标记。	使用默认词性’x’的方法虽然避免了词性丢失，但是默认的’x’词性并不一定准确。
20	比较测试比较测试……测试比较测试比较测试比较 结果为 比较 测试 比较 测试 …… 测试 比较 测试 比较/d 测试/v 比较/d	丢失部分词性标注信息，这是一个测试特例，省略号中的内容为多个“比较测试”的重复串达到两百个以上，由于一句中连续出现的有歧义词性的的词个数超过限制的 MAX_WORDS_PER_SENTENCE，在设置结束符号时数组下标越界造成中间结果所有词性均丢失。	对长度进行了限制并正确的设置了下标。	

3． 实验分析及性能报告

	原始版本 (发布结果)	原始版本 (实测结果)	更新版本 (实测结果)
分词正确率(按词统计)	98.231178%	98.7253%	98.8313%
上位词性标注正确率(按词统计)	95.526672%	95.6445%	96.1989%
上位词性标注相对正确率(按词统计)	97.246795%	96.8794%	97.3365%
下位词性标注正确率(按词统计)	93.335495%	93.4519%	94.0148%
下位词性标注相对正确率(按词统计)	95.016162%	94.6585%	95.1265%
分词正确率(按句统计)	88.877231%	69.5802%	71.6896%
上位词性标注正确率(按句统计)	79.072470%	34.4642%	37.3024%
上位词性标注相对正确率(按句统计)	88.968197%	49.5316%	52.0332%
下位词性标注正确率(按句统计)	33.106805%	25.0411%	26.6732%
下位词性标注相对正确率(按句统计)	75.264715%	35.9888%	37.2065%

表格 1　1998 年 1 月的《人民日报语料库》测试结果

	规模	原始版本(实测结果)		更新版本(实测结果)	
		分词正确率 (按词统计)	速度	分词正确率 (按词统计)	速度
1998.1 人民日报	3998K 19484 句	98.7253%	98360ms 40.65K/s	98.8313%	92562ms 43.19K/s
北大测试集	55K 380 句	95.6671%	1437ms 38.2K/s	95.7776%	1406ms 39.12K/s
Finace_News测试集[1]	913K 23859 句	91.2349%	25547ms 35.74K/s	91.337%	22578ms 40.44K/s

表格 2 不同测试集结果比较

a) 由于未能获取 973 专家评测数据，因此主要以 1998 年 1 月人民日报语料库进行自评；

b) 使用其他分词评测的数据比较原始版本与修正后版本的性能差异(只有切词结果无词性标记)

c) 修正新版后，系统的稳定性、正确性及速度都有提高。

i. 稳定性提高主要是由于修正了系统内部使用数组的访问越界或者溢出情况，对未做初始化的部分变量赋了初值避免访问非法内存，对于内存的分配及释放也做了更合理的管理；

ii. 正确性的提高主要在于重写了关于数词及特殊符号的相关处理函数便得一些明显的切分错误结果得到修正；另外，由于变量未初始化可能造成的切分过程漏词、多词等现象有所改善；部分乱码或者无词性单字造成的随机结果得到改善；

iii. 速度的提高并不明显，主要提高点为在 AtomSegment 函数中提前处理了部分汉字数字串的情况，使得后期处理的词数减少，速度有所提高。

4． 其他需要说明的事宜

a) 我使用 1998 年 1 月人民日报语料库对原始版本进行自测的结果与邹纲发布的结果不同，具体有以下几点：

i. 从网上免费下载的标准答案共有词 1140931 个，而邹纲的结果中，切分正确的词就达 1147791，即比原始的词还多，除非使用数据不同，否则数据必然有误；

ii. 发布的“ 下位词性标注相对正确率(按句统计)“ 计算明显有误，应为 37.2500%而不是

[1] 由于原始版本处理该测试集时系统会发生崩溃，在实验时去掉了其中十来个句子。

75.264715%。(下位词性标注相对正确率(按句统计)=下位词性标注正确率(按句统计)/分词正确率(按句统计))

iii. 使用我个人编写的评测程序比较原始版本与修正版本的性能，在 1998 年 1 月人民日报语料库中按词统计的数据均有所提高(即使是原始版本的数据也比邹纲发布结果要高)，但是按句统计性能严重低于发布结果，怀疑其中计算方法不一致。