

A Semi-Automated Method of Extracting COSMIC Functional Users

Hao Liu, Song Wu

College of Computer Science, National University of Defense Technology, Changsha Hunan
Email: 87647495@qq.com

Received: Jan. 29th, 2016; accepted: Feb. 22nd, 2016; published: Feb. 25th, 2016

Copyright © 2016 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

COSMIC is one of the mainstream function point size measurements, but in practice will lead a certain deviation because of the lack of guidance which is easy to operate or the different comprehensions of the natural language in requirement documents. We subdivide parts of principles and rules in COSMIC official manual into ten specific processing steps. We present a semi-automated need pretreatment method based on smart tools. To some extent, we solve the problems of not giving a specific implementation process and measurement inconsistency caused by the personnel subjective measure in the measurement manual. Practice shows that by using the method we can complete all the work in the stage of strategies, be able to extract the candidate functional users and provide some reference for automated measurement of COSMIC in the future.

Keywords

Semi-Automatic, COSMIC, Functional User, Method

一种半自动化提取COSMIC功能用户的方法

刘浩, 吴颂

国防科学技术大学计算机学院, 湖南 长沙
Email: 87647495@qq.com

收稿日期: 2016年1月29日; 录用日期: 2016年2月22日; 发布日期: 2016年2月25日

摘要

COSMIC方法是当今功能点规模度量方法的主流之一，但是在实践中常因为缺少易于操作的方法指导，或者对需求文档中自然语言的理解不同造成一定的偏差。本文将COSMIC官方手册中的部分原则和规则，细化为具体的十个处理步骤，提出了一种基于智能工具的半自动化需求预处理方法，一定程度上解决了度量手册中没有给出具体的实施流程，以及度量人员主观臆断引起的度量不一致等问题。实践证明，该方法能够完成COSMIC方法度量策略阶段的所有工作并能够提取出候选功能用户，同时为下一步COSMIC度量的自动化提供了一定的参考。

关键词

半自动化, COSMIC, 功能用户, 方法

1. COSMIC 方法简介

软件规模度量是度量软件开发的工作量、成本与资源需求的基础。目前流行的度量方法有代码行估算方法和功能点分析方法(Function Points Analysis, 简称FPA)。代码行估算是一种经验方法，其结果与度量人员、使用的开发工具紧密相关。FPA方法的代表有IFPUG(The International Function Point Users' Group, 国际功能点用户协会)、COSMIC(Common Software Measurement International Consortium, 通用软件度量国际联盟)等。COSMIC方法从用户功能的视角入手，起源于客户可以理解的术语，不需要调整因子，简单易行，因而受到越来越多的软件公司的推崇。

COSMIC方法假设软件的功能规模大小是通过累计“数据移动”的个数来度量的，一个数据移动是一个数据组的传输，一个数据组是一个有唯一的、非空的、无序且没有冗余的数据属性的集合。数据移动有4种子类型：输入、输出、写和读。输入是从功能用户跨越边界传输数据到系统内部，这里的功能用户既包括使用系统的自然人，也包括其他软件或者硬件设备；输出是一个数据组从系统跨越边界移动到功能用户；写是存储数据到持久存储器；读是从持久存储器读取数据。一次数据移动记为一个COSMIC功能点(COSMIC Function Point, 简称CFP)，CFP是COSMIC方法中标准的测量单位。通过统计系统中所有CFP的个数就可以得到系统的功能规模大小[1]。

2. COSMIC 方法度量策略阶段存在的困难

目前，COSMIC方法的指导依据是官方给出的COSMIC功能规模度量方法4.0.1版[2](The COSMIC Functional Size Measurement Method v4.0.1, 下文简称度量手册)。度量手册给出了COSMIC方法的10个原则和17条规则，分三个阶段(度量策略阶段、映射阶段和度量阶段)实施。度量策略阶段是COSMIC方法的第一个阶段，在该阶段，准确识别功能用户以及其他度量元素是确保度量一致性的前提和保证，但是在实践中往往存在这样或那样的问题。

2.1. COSMIC 方法度量策略阶段需要完成的工作

COSMIC方法度量策略阶段的起点是软件需求文档，终点是确定功能用户、粒度级别和层。此阶段需要完成以下工作：

明确度量目的，由度量目的导出度量范围，分解软件层级，识别粒度级别，识别功能用户。

2.2. COSMIC 方法度量策略阶段存在的困难

在运用 COSMIC 方法进行度量实践时, 在其策略阶段仍然存在以下问题:

2.2.1. 缺少可实施的方法指导

度量手册给出了 COSMIC 方法策略阶段的 3 个原则和 3 条规则, 但对于初学者往往无从下手。

2.2.2. 需求理解不同导致度量不一致

对于软件需求文档的理解, 取决于客观和主观两个方面: 客观方面, 不可否认的是软件需求文档的质量问题广泛存在着, 度量人员在面对一份质量较差的需求时, 往往花费大量精力去阅读和分析, 仍不能正确地理解需求描述, 度量结果也就很难达成一致; 主观方面, 虽然通过主观臆测进行度量的做法是不科学的, 而且这也不符合 COSMIC 方法的基本要求, 但是实践中, 在对软件需求文档进行分析时, 往往不可避免地需要进行人工逻辑推理。这是因为需求文档本身离不开自然语言的描述, 也就避免不了自然语言的模糊性[3]。

3. 基于分词工具的半自动化文档分析方法步骤

周泽龙在《可实施的 COSMIC 方法研究》[3]一文中, 使用汉语句法分析功能用户需求(Function User Requirement, 下文简称 FUR), 能够得到部分度量元素, 但是实践发现在个别处理步骤中, 人工操作只是做了简单的重复性工作, 工作量大且易纰漏, 基于此, 本文作者尝试将智能分词工具引入该方法, 替代部分人工步骤, 具体方法步骤及处理规则如下 3.1~3.10。

3.1. 从需求文档中找出功能需求章节

GJB438B-2009《军用软件开发文档通用要求》, 规范了软件需求文档的正式格式, 其中需求文档内容应当包括: 范围、引用文件、需求、合格性规定、需求可追踪性、尚未解决的问题和注解等七个部分[4]。通常我们需要找到需求部分, 从中提取描述软件功能需求的相应章节。

3.2. 将功能需求章节分割成独立的 FUR 短句

规则: 将文档中的“换行符”“,”“。”“;”处断开, 分割成独立的 FUR 短句。

使用表 1 所示的正则表达式操作字符串, 能够得到 FUR 短句。

3.3. 手工调整 FUR 短句

规则 3.3.1: 语态变换。将被动语态的 FUR 短句转换成主动语态。

规则 3.3.2: 拆分规则。含有两个或多个并列动词的复句应分解为两个或多个短句。

规则 3.3.3: 合并规则。下列情况中的“,”不应作为 FUR 短句的结束标识符, 应将多个短句合并为一句:

- 用于并列词语之间, 表并列关系的“,”。
- 用于句子内部, 表语气停顿的“,”。

规则 3.3.4: 排除规则。排除下列非功能需求语句:

- 排除介绍软件质量、性能、实现功能类语句。如“主控程序收到自检指令完成检查并返回结果的时间小于 30 秒”。
- 排除概括性、总结性语句和章节标识。如“1.登陆 1.1 简要描述 这个用例描述用户是如何登陆到课程注册系统的”。

Table 1. Use java language to punctuate
表 1. 使用 java 语句实现断句

```
File file = new File("");
FileWriter fw = new FileWriter("");
BufferedReader br = new BufferedReader(new InputStreamReader(new FileInputStream(file), "gb2312"));
String str = null;
while((str = br.readLine()) != null){
    //去除读入一行中的空格和回车
    String s = str.replaceAll("\\p{Z}", "");
    //匹配正则, 替换[, . ; ], 后加入回车符
    s = str.replaceAll("[, . ; ]", "$1+" + "\r\n");
    fw.write(s);
    System.out.println(s);
    fw.flush();
}
fw.close();
br.close();
```

- 排除数据运算类语句。如“将实时温度与目标温度进行比对”。
- 排除其他无数据移动动作的语句。如“XXXX 进入到软件定时中断”。

3.4. 结合上下文补全 FUR 短句成分

经过断句处理后的 FUR 短句，往往语法成分不全，在本步骤中，我们需要结合上下文，按照一般的行文语法规则将省略的句子成分尽可能地补全。

规则 3.4.1：补全主语。常见于待度量软件作为短句主语被省略的情况。

规则 3.4.2：补全状语。常见于短句表示对象的状态缺失等情况。

规则 3.4.3：无法补全。短句主语、状语均被省略。常见于动作的发起者是待度量软件，而接收者也是待度量软件或者是存在于它内部的存储器。这种情况，应结合上下文将显而易见的缺失补上，对于那些无法通过上下文判断的残缺成分，不能盲目补全。

3.5. 联系上下文找出用来替换代词的实体名词

为避免重复，需求文档除了采用“省略”手法外，还广泛地使用代词来代替前文中出现过的词或提到的内容，常使用的代词有人称代词（“他”、“它”等）和指示代词（“其”、“这个”等）。在本步骤中必须根据上下文的逻辑关系正确判断代词所替代的具体内容，如分析不当，往往会“张冠李戴”，导致功能用户识别错误。

规则 3.5.1：将短句中所有指代待度量软件的词替换为同一个词。

规则 3.5.2：通过上下文将其余代词的指代对象找出，用相应的实体名词替换。

3.6. 利用分词软件将 FUR 短句按照词性进行切分

本文选择了的 NLPIR 汉语分词系统作为分词工具用来辅助人工度量，同样功能的分词工具还有很多，如 Stanford 汉语分词工具、哈工大语言云(LTP-cloud)、庖丁解牛分词、盘古分词等。中文分词工具相对于以英文为代表的拉丁语系语言分词工具在分词的准确性上还有不小的差距，但多数软件都开发了用户自定义词组等功能方便人工校订。

中国科学院计算技术研究所研发的 NLPIR 汉语分词系统(ICTCLAS2013 及以后版本)，主要包括中文分词、词性标注、用户词典、新词发现和关键词提取等功能[5]。本步骤使用 NLPIR2015 分词工具将处理好的 FUR 短句进行分词。

3.7. 利用工具将切分好的词进行词性标注

NLPIR2015 不但可以将 FUR 短句按词性进行切分, 而且对切分好的词语都会进行词性标注, 标注依据计算所汉语词性标记集 Version3.0 [5], 表 2 列出了常见的词性标记:

分析输出结果发现, 智能工具的优点很明显, 就是处理文档的速度快, 也不会因为疏忽而出现遗漏, 同样的, 工具的缺点也很突出, 如无法结合上下文断句, 生僻词不能够准确划分, 度量不关心的词语和标点也进行了词性标注等。其实, 如果需要更加准确地进行机器识别, 那么就要增加识别规则, 必然将增大计算机的开销, 然而在使用工具处理之后, 再利用人工进行校对处理, 却能收到很好的效果。

3.8. 人工纠正划分错误的词语、删除标点符号的词性标注

规则 3.8.1: 一般词拆分错误, 手工逐个更正。对于更正后的词语, 应联系上下文手工标注词性。

规则 3.8.2: 领域专用词拆分错误, 使用 NLPIR 分词系统中的“用户词典”功能和“自定义词性”功能, 批量修改文档中出现的分词错误并重新标定词性。

NLPIR 提供了可视化操作界面, 其中“用户词典”功能演示效果如图 1 所示。

规则 3.8.3*(可选): 删除副词、助词、字符串、标点符号等与度量无关的词性标注。

Table 2. Common speech list
表 2. 常见词性列表

标记符号	中文词性
n	名词
v	动词
vn	名动词
rr	人称代词
p	介词
c	连词
cc	并列连词



Figure 1. “User Dictionary” and “Noun Speech” in NLPIR
图 1. NLPIR 分词系统中的“用户词典”功能和“自定义词性”功能

COSMIC 方法往往只关心描述数据移动的动词以及承担动作发起者或者接收者的名词, 为了避免干扰, 可以将对度量没有影响的副词(如 d)、助词(如 u、ude1 的、ule 了等), 以及字符串(如 x)和标点符号(如 wj 句号、wd 逗号、wh 单位符号)等词性标注删除。

3.9. 依据汉语语法规则划分 FUR 短句句子成分

该步骤通过语法分析, 将每一个 FUR 短句中重要的句子成分划分出来。我们知道, 一个完整句子其成分的排列顺序应该为: 修饰主语的定语 + 主语 + 状语 + 谓语 + 补语 + 修饰宾语的定语 + 宾语, 具体应用时各个成分的组合是非常灵活的, 尽管汉语语句浩如烟海, 写作运用的自然句子无穷无尽, 但软件需求文档的写作风格却相对固定, 使用的句子结构类型也十分有限, 其规律也是可循的[6]。经验说明, 经过处理后的 FUR 短句满足以下三种句子模型:

(修饰主语的定语) + 主语 + 状语 + 谓语 + (修饰宾语的定语) + 宾语。

(修饰主语的定语) + 主语 + 谓语 + 宾语 + 宾语补足语。

(修饰主语的定语) + 主语 + 谓语 + 直接宾语 + (定语) + 间接宾语。

如果忽略对度量无影响的定语, 以上三种模型可简化为: “主状谓宾”结构、“主谓宾补”结构和“主谓双宾语”结构。

《现代汉语句子成分的分析》[7]一书中介绍了句子成分分析的一般方法, 有图解法、文字说明法和加线法等。本文使用 Excel 软件, 将上述三种结构中每个句子成分作为一列, 填入表头作为列名称, 删除对度量没有影响的定语列、补语列等列, 将每个 FUR 短句中充当相同句子成分的词语进行词对齐操作, 填入如表 3 所示的相应列中。

3.10. 提取候选功能用户

规则 3.10.1: 待度量软件、功能用户以及持久存储器, 在 FUR 短句中可均充当主语、状语、补语或直接宾语。

我们依据该规则, 在 Excel 中将主语列、状语列、补语列、直接宾语列合并, 得到一个“混合列”。再剔除“混合列”中内容重复的单元格、待度量软件和“空值”, 就可以得到全部功能用户的集合了。

值得注意的是, 在 3.4 节中, 我们将显而易见的待度量软件补入句中, 而无法联系上下文判断的持久存储器(FUR 的描述语句通常不会刻意提及“持久存储器”这个概念[3])没有盲目补全, 这就导致 3.9 节列出的句子成分中, 有部分单元格为“空值”, 这些“空值”一般是指持久存储器, 但也不排除是功能用户的特殊情况, 故最终得到的全部功能用户集合, 严格意义上说是一个候选功能用户集合。

4. 实验验证

电饭煲系统(Rice Cooker application) [8]是 COSMIC 官方给出的实时软件案例, 用于控制智能电饭煲的自动烹饪。该软件的功能需求章节的描述如表 4。

使用本文方法提取该系统候选候选功能用户的流程如下。

Table 3. Division sentence elements

表 3. 划分句子成分

FUR 短句	划分句子成分						
	主语	状语	谓语	宾语	补语	直接宾语	间接宾语

4.1. 智能断句处理

按照步骤 3.2 将功能需求章节进行断句, 结果如图 2 所示。

4.2. 人工处理

按照步骤 3.3~3.5, 将短句进行“手工调整”、“补齐句子成分”和“替换代词”等操作, 得到 FUR 短句如表 5 所示。

4.3. 分词工具处理

使用 NLPiR 分词工具, 对人工处理后得到的 FUR 短句进行分词和词性标注(步骤 3.6~步骤 3.7), 如图 3 所示。

4.4. 划分句子成分

对分词处理后需进行人工纠错(步骤 3.8), 并将划分好的句子成分(步骤 3.9)填入 Excel 相应列中, 如图 4 所示。

4.5. 提取候选功能用户

按照步骤 3.10 所述, 提取该软件的候选功能用户为: 用户、计时器、温度传感器、加热器、指示灯。

5. 结束语

本文在遵循 COSMIC 方法的基础上, 提出了一种半自动化的需求预处理方法。通过引入智能工具替

Table 4. Division sentence elements

表 4. 划分句子成分

电饭煲有三种烹饪模式: 快速、正常和煮粥模式。除非用户选定特定的烹饪模式并按下 START, 否则电饭煲将按照正常模式煮饭。

实时软件能够自动为加热器提供在给定的模式和给定持续时间的目标温度。每隔 30 秒, 一个新的目标温度将被设置; 每隔 5 秒, 实际温度将从外部温度传感器获取, 并于目标温度进行比对。目标温度减去实际温度的差值为正时, 加热器被设置为开, 否则为关。

煮饭和保温过程中有相应的指示灯亮着提示电饭煲状态。

Table 5. Short sentences of FUR

表 5. FUR 短句

人工处理(“调、补、替”)得到 FUR 短句

用户按下特定烹饪模式和 START 键, 向电饭煲发出烹饪指令

电饭煲写入特定烹饪模式。

电饭煲能够读取给定的烹饪模式

电饭煲能够读取给定持续时间的目标温度。

计时器每隔 30 秒向电饭煲发送一段时间信号,

电饭煲设置一个新的目标温度;

计时器每隔 5 秒向电饭煲发送一段时间信号,

电饭煲将从外部温度传感器获取实际温度,

电饭煲向加热器发送开/关指令。

电饭煲向相应的指示灯发送开/关指令。

```

package code;

import java.io.BufferedReader;

public class DuJu {

    public static void main(String[] args) throws Exception {

        //读取需要断句的文档
        File file = new File("E:\\分词\\myjavafenci\\importtxt\\电饭煲案例\\1案例
        //断句后要写入的文档
        FileWriter fw = new FileWriter("E:\\分词\\myjavafenci\\importtxt\\电饭煲
        //将字节流用gb2312编码转换为字节流
        BufferedReader br = new BufferedReader(new InputStreamReader(new File
        String str = null;
        //循环读入文档
        while((str = br.readLine())!=null){
            //去除读入一行中的空格和回车
            String s = str.replaceAll("\\p{Z}", "");
            //匹配正则, 替换[, . ; ], 后加入回车符
            s = str.replaceAll("[, . ; ]", "$1"+"\\r\\n");
            //写入目标文档
            fw.write(s);
            System.out.println(s);
            fw.flush();
        }
    }
}
    
```

<terminated> DuJu [Java Application] E:\JAVA\jre7\bin\jav
 电饭煲有三种烹饪模式：快速、正常和煮粥模式。
 除非用户选定特定的烹饪模式并按下START，
 否则电饭煲将按照正常模式煮饭。
 实时软件能够自动为加热器提供在给定的模式和给定持续时间的目标温度。
 每隔30秒，
 一个新的目标温度将被设置；
 每隔5秒，
 实际温度将从外部温度传感器获取，
 并于目标温度进行比对。
 目标温度减去实际温度的差值为正时，
 加热器被设置为开，
 否则为关。
 煮饭和保温过程中有相应的指示灯亮着提示电饭煲状态。

Figure 2. The result of smart punctuation
 图2. 智能断句结果

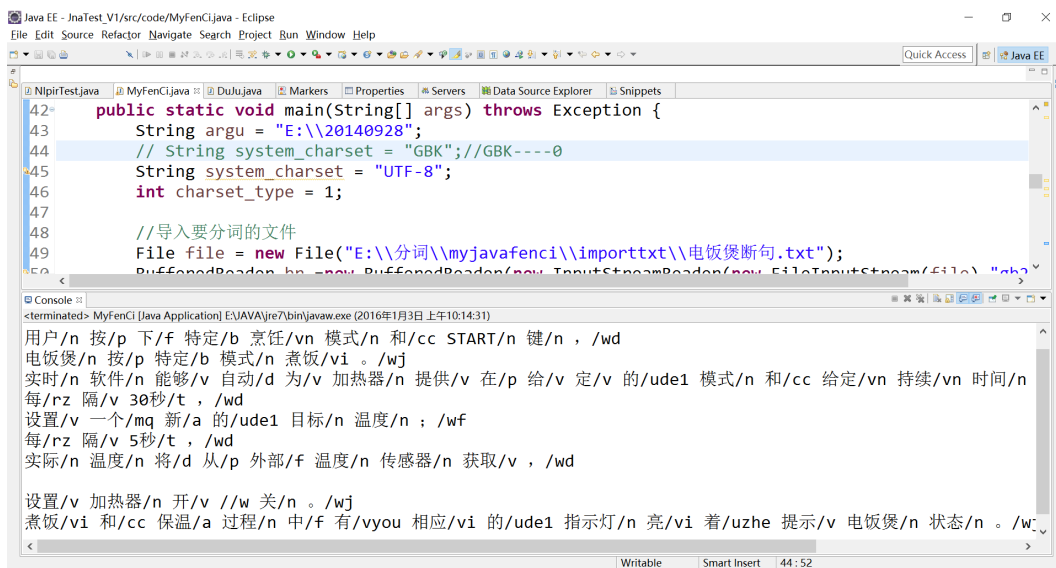


Figure 3. The result of NLP
 图3. NLP的分词结果

人工校对处理（“校”）后的FUR短句	划分子句成分					
	主语	状语	谓语	宾语	补语	直接宾语 间接宾语
用户/n按/p下/t特定/b烹饪模式/n和/START键/n, 向/p电饭煲/n发出/v烹饪指令/n	用户/n	电饭煲/n	发出/v	烹饪指令/n		
电饭煲/n写入/v特定/b烹饪模式/n。	电饭煲/n		写入/v	烹饪模式/n		
电饭煲/n能够/v读取/v给定的/烹饪模式/n	电饭煲/n		读取/v	烹饪模式/n		
电饭煲/n能够/v读取/v给定持续时间的/目标温度/n。	电饭煲/n		读取/v	目标温度/n		
计时器/n每/rz隔/v30秒/t向/p电饭煲/n发送/v一段/mq时间信号/n,	计时器/n	电饭煲/n	发送/v	时间信号/n		
电饭煲/n设置/v一个/mq新的/目标温度/n;	电饭煲/n		设置/v	目标温度/n		
计时器/n每/rz隔/v5秒/t向/p电饭煲/n发送/v一段/mq时间信号/n,	计时器/n	电饭煲/n	发送/v	时间信号/n		
电饭煲/n将/d从/p外部/b温度传感器/n获取/v实际温度/n,	电饭煲/n	温度传感器/n	获取/v	实际温度/n		
电饭煲/n向/p加热器/n发送/v开/关指令/n。	电饭煲/n	加热器/n	发送/v	开/关指令/n		
电饭煲/n向/p相应的/指示灯/n发送/v开/关指令/n。	电饭煲/n	指示灯/n	发送/v	开/关指令/n		

Figure 4. Divide the elements within the sentences
 图4. 划分子句成分

代部分手工操作, 将 COSMIC 方法度量策略阶段的原则和规则, 转化成为易于操作的步骤流程。需要说明的是, 本文只针对 COSMIC 方法的第一阶段进行了改进, 得出的结果是服务后续度量工作的, 并非软件的最终规模大小, 要想得到最终的软件规模, 还需要进行映射和度量等两个阶段的处理, 这也为作者今后的实验探索提出了新的挑战。实践证明, 本文方法能够更加准确、高效地完成 COSMIC 方法度量策略阶段的工作, 为 COSMIC 方法识别功能用户提供了一种有效的、易于实施的新思路, 也为 COSMIC 度量的自动化技术进行了一定的探索。

参考文献 (References)

- [1] 任甲林. 术以载道——软件过程改进实践指南[M]. 北京: 人民邮电出版社, 2014, 4: 152-152.
- [2] COSMIC (2015) The COSMIC Functional Size Measurement Method-Version 4.0.1 Measurement Manual. <http://www.cosmic-sizing.org>
- [3] 周泽龙. 可实施的 COSMIC 方法研究[D]: [硕士学位论文]. 长沙: 国防科技大学计算机学院, 2013: 6.
- [4] 程江涛. 一种改进的 COSMIC 实施方法研究[D]: [硕士学位论文]. 长沙: 国防科技大学计算机学院, 2014: 5.
- [5] NLPPIR 汉语分词系统. NLPPIR 简介[EB/OL]. <http://ictclas.nlpir.org>
- [6] 李裕德. 科技汉语语法[M]. 北京: 冶金工业出版社, 1985: 12.
- [7] 赵遵礼. 现代汉语句子成分的分析[M]. 西安: 陕西人民出版社, 1983: 5.
- [8] COSMIC-CASE STUDY (2000) Rice Cooker Application. <http://www.cosmic-sizing.org>