计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2018,27(4):254−258 [doi: 10.15888/j.cnki.csa.006280]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于相似度的软件工程中软件成本估算问题研究①

王　楠, 余建坤

(云南财经大学 信息学院, 昆明 650021)

摘　要: 软件工程概念从 1968 年被提出以来, 经历了近 50 年的发展, 软件系统规模和复杂程度日益加大, 然而从上个世纪 70 年代左右开始, 软件工程领域出现大量软件项目进度延期、预算超支和质量缺陷为典型特征的软件危机. 这体现出软件成本估算在软件工程开发过程的重要性. 精准的软件成本估算是软件工程按时完成的保证. 本文采用一种基于皮尔逊相关系数的相似度量方法, 结合 TOPSIS 方法软件成本进行类比估算以获取与之最接近项目的项目作为参考进行软件成本估算. 最后将该方法应用于 Desharnais 数据集进行实验, 并和其他方法进行比较, 实验结果表明, 本文采用的基于相关系数的软件成本度量方法较已有的相似性度量方法准确度更好.

关键词: 相关系数; 成本估算; 相似性度量

引用格式: 王楠,余建坤.基于相似度的软件工程中软件成本估算问题研究.计算机系统应用,2018,27(4):254–258. http://www.c-s-a.org.cn/1003-3254/6280.html

Software Cost Estimation in Software Engineering Based on Similarity

WANG Nan, YU Jian-Kun

(School of Information, Yunnan University of Finance and Economics, Kunming 650021, China)

Abstract: Since the introduction of software engineering concept in 1968, it has experienced the development of nearly 50 years. The scale and complexity of software system have been increasing day by day. However, since the 1970s, there has been a delay in the progress of a large number of software projects. And quality defects for the typical characteristics of the software crisis, is still frequent. Software cost estimation in the software engineering development process has been playing an important role. The precise software cost estimate is the guarantee of software engineering on time. In this study, a similarity measure based on Pearson correlation coefficient is used, and the software cost estimation is carried out by using the TOPSIS method to estimate the cost of the project with the closest project. Finally, the method is applied to the Desharnais dataset and compared with other methods. The experimental results show that the software cost measurement method based on the correlation coefficient has better accuracy than the existing similarity measure method.

Key words: correlation coefficient; cost estimation; similarity measure

软件工程过程是指开发或维护软件及其相关产品的一系列活动. 一个完整的软件工程开发过程包括软件概念提出、前期需求分析、软件结构设计、软件详细设计、编码过程、软件测试过程 6 个过程[1]. 随着科技的发展, 软件的系统规模也在不断扩大, 复杂程度日益加大, 需求分析作为整开发过程的基础, 在软件工程中的重要程度和地位越来越被人们所认可. 据统计、有超过半数以上的软件工程项目存在前期需求分析不当的问题, 许多项目也因此导致延期或失败[2]. 软件成本估算问题作为前期需求的重中之重, 也越来越被人们所重视.

如何对软件项目进行精准成本估算, 一直是软件

① 基金项目: 云南省高校商务智能科技创新团队基金 (42212217010)
收稿时间: 2017-07-10; 修改时间: 2017-07-24; 采用时间: 2017-07-28; csa 在线出版时间: 2018-03-31

工程和软件项目管理中最为重要且最具挑战问题之一. 软件成本估算是进行有效的项目计划、跟踪和控制的基础. 依照合理的估计结果, 不仅能够制定切实可行的目标, 还可对软件的成本、进度与质量进行权衡, 实施有效的风险管理, 并为项目管理者的决策提供有力的支撑和依据. 不准确的估计则会造成软件工程项目的延期、超支甚至项目失败, 严重的还会损害企业的商业形象和信誉.

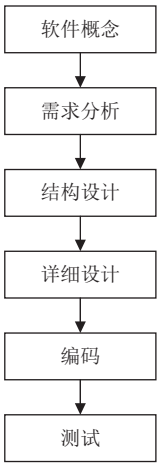

图 1 软件工程流程

Standish 组织在 1995 年公布的软件工程报告显示, 在来自国际上 350 个组织的 8000 个软件工程项目中, 只有 16.2% 项目被定义为“Succeeded”, 即该项目在预算和预期内完成; 31.1% 的项目被定义为“Falied”, 即该项目未能按时完成或者被取消; 剩下的 52.7% 被定义为“Challenged”, 即虽然该项目被完成, 但预算超出或者项目完成不达标[3]. 2004 年, Standish 组织的再次公布其统计数据, 统计项目数累计超过 50000 多个, 根据其公布的结果显示, “Succeeded”项目所占比例为 29%, “Falied”项目所占比例为 29%, 有所下降, 而“Challenged”项目比例仍有 53%[4].

虽然有许多学者和专家认为在 Standish 组织公布的报告中关于软件成本预算超支 89% 的数据被过分夸大, 但有一点却能够取得共识, 即不精准的软件成本估算与需求不稳定并列, 是造成软件工程项目失败和超期的最主要的的两大因素.

软件成本估算和方法和理论有很多种类和形式, 最早的软件成本估算是上 60 年代提出的 SDC (System Development Corporation) 方法, 一直到如今, 关于软件成本度量方法主要有两种方法: ① 基于模型的软件成本估算方法; ② 基于类比估算软件成本估算方法[5].

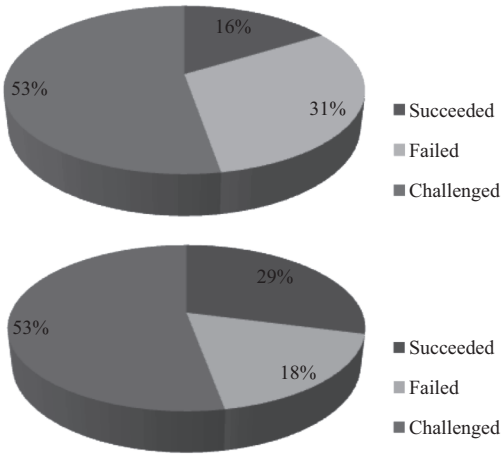

图 2 Standish 组织统计数据

基于模型的软件成本估算方法是通过将影响软件工程项目的相关因素如项目复杂性、相关管理经验、团队经验等与软件项目的相关指标例如工作量、工作环境、工作时长等之间存在着可用公式表示的确定关系, 并判定它对工作量所产生影响的程度, 再从参数得到成本估算的一系列规则、公式. 以期得到最佳的模型算法表达形式, 然而基于模型的软件成本估算方法难以用在没有前例的场合, 并且不能处理异常情况等的问题. 而且算法复杂度往往比较高, 所以基于模型的软件成本估算方法存在一定缺陷[6].

基于类比的方法是是采用基于相似性度量的方法进行软件成本的估算. 即通过对一个或多个已完成的软件工程项目项目与新的项目之间的对比来预测当前项目的成本与进度. 在软件成本估算中, 需要当前问题抽象为待估算的项目时, 每个实例即指已完成的软件项目, 通过案例识别、案例检索以及案例适配 3 个步骤进行软件实例之间成本相似性估算. 在软件成本估算问题上, 经常采用的相似性度量方法有欧式距离、熵度量、模糊度量等, 例如文献[7]提出一种基于基于协同过滤方法的软件成本度量方法, 通过用户评分来定义属性权重, 然而这种方法依然是一种基于专家评论的方法, 然而软件工程十分复杂, 单一的专家定义权重精确度并不高, 用户的个人偏好、经验差异与专业局限性都可能为估算的准确性带来风险, 文献[8]提出一种基于相似度的软件成本度量方法, 根据不同属性的类型采用不同的度量公式, 例如欧式距离, 余弦公式

等, 然而这种方法依然是采用单一的度量公式来度量软件成本相似性, 而且单一的采用欧式距离等公式在软件成本度量方面精度比较低. 文献[9]借鉴改进了 Sheperd 等人关于类比估算的方法, 提出来采用提取相似项目、决定最相似的项目等 4 个步骤来进行软件成本相似性度量, 并根据不同的属性类别选用不同相似性度量方法, 然而其依然是采用单一的度量公式进行软件成本相似性度量、并没有考虑如何使度量最优化的问题, 其方法精确度并不高[10].

本文采用类比估算方法, 提出一种采用皮尔逊相关系数的度量方法, 并结合 TOPSIS 决策方法, 采用专家评估和客观权重综合的方法, 提出一种的基于皮尔逊相关系数的软件成本相似性度量方法, 来进行软件成本的相似性估算, 通过并通过 Desharnais 数据集进行了实验验证, 证明本方法在软件成本估算问题上相对欧式距离和其他方法在检测精度上具有一定优势.

1 理论背景

1.1 相关系数

相关系数 (Correlation Coefficient) 又称皮尔逊相关系数. 是由著名英国数学家卡尔·皮尔逊于首次提出的一个统计学指标. 相关系数是用于反映所求变量之间相关关系密切程度的统计指标. 相关系数是按积差方法计算, 同样以两变量与各自平均值的离差为基础, 通过两个离差相乘来反映两变量之间相关程度. 皮尔逊相关系数并不是唯一的相关系数, 但是却是最常见的相关系数. 皮尔逊相关关系是一种非确定性的关系, 它是用来描述变量之间线性相关程度的量.

定义 1. 假设存在随机变量 X, Y 是两个随机分布, $E(X)$、$E(Y)$为 X, Y 的期望, 则 X 与 Y 的协方差$Cov(X,Y)$被定义为:

$$Cov(X,Y)=E\{[X-E(X)]\,[Y-E(Y)]\}$$

定义 2. 相关系数ρ_{xy}为:

$$\rho_{xy}=\frac{Cov(X,Y)}{\sqrt{D(X)}\sqrt{D(Y)}}$$

其中, $D(X)$, $D(Y)$为随机分布 X、Y 的方差.

相关系数ρ_{xy}取值在–1 到 1 之间, ρ_{xy}=0时, 称 X, Y 不相关; $|\rho_{xy}|$=1时, 称 X, Y 完全相关, 此时, X, Y 之间具有线性函数关系; $|\rho_{xy}|<1$时, X 的变动引起 Y 的部分变动, $|\rho_{xy}|$的值越大, X 的变动引起 Y 的变动就越大, $|\rho_{xy}|>0.8$时称为高度相关, 当$|\rho_{xy}|<0.3$时称为低度相关, 其它时候为中度相关.

1.2 TOPSIS

最优劣解距离法 (Technique for Order Preference by Similarity to an Ideal Solution, TOPSIS) 是由 C. L. Hwang 和 K. Yoon 于 1981 年首次提出. TOPSIS 是处理真实世界中的多属性或多标准决策 (MADM/MCDM) 问题的主要技术之一[11]. 它帮助决策者组织待解决的问题, 并对替代品进行分析, 比较和排名. 从而进行合理的选择.

Topsis 方法的具体过程如下:

1) 对特征矩阵进行规范化处理, 得到规格化向量n_{ij}, 建立关于规格化向量n_{ij}的规范化矩阵.

$$n_{ij}=x_{ij}/\sqrt{\sum_{j=1}^{m}x_{ij}^2},\quad j=1,\cdots,m;\ i=1,\cdots,n$$

2) 通过计算权重规格化值v_{ij}建立关于权重规范化值的v_{ij}权重规范化矩阵

$$v_{ij}=w_i n_{ij},\ j=1,\cdots,m;\ i=1,\cdots,n$$

其中, w_{ij}是第 j 个指标的权重.

3) 确定正理想解 Z^+和负理想理 Z^-

$$Z^+=\left(\max_i v_{ij}|j\in J_1\right),\left(\min_i v_{ij}|j\in J_2\right),\ i=1,2,\cdots,m$$
$$Z^-=\left(\min_i v_{ij}|j\in J_1\right),\left(\max_i v_{ij}|j\in J_2\right),\ i=1,2,\cdots,m$$

其中, J_1 为收益性指标集, 表示在第 i 个指标上的最优值. J_2 是损耗性指标集, 表示在第 i 个指标上的最劣值. 收益性指标越大, 对评估结果越有利; 损耗性指标越小, 对评估结果越有利. 反之, 则对评估结果不利.

4) 计算距离尺度, 即计算每个目标到正理想解和负理想解的距离:

$$D^+=\{\sum_{i=1}^{n}\left(v_{ij}\wedge v_i^+\right)\}$$
$$D^-=\{\sum_{i=1}^{n}\left(v_{ij}\wedge v_i^-\right)\}$$

5) 计算目标与理想解之间的的相对接近度.

$$p_i=\frac{Z^-}{Z^++Z^-}$$

6) 排列偏好顺序.

2 一种基于相关系数的相似性度量方法在软件成本评估中的应用

在软件成本的估算过程中, 不同指标之间权重的

确定问题尤为重要, 大多数相似性度量方法一般根据专家的判断或者由先验经验来确定[12−15], 但这种方法主观性太强, 本文采用基于主观权重和客观权重的混合权重确定方法, 既通过专家判断, 又采用基于信息熵的权重确定方法. 具体方法如下.

假设评估体系中具有 n 个评价对象, m 个评估指标, 评价矩阵为:

$$A=\begin{vmatrix} a_{11} & a_{12} & \cdots & a_1 \\ a_{21} & a_{22} & \cdots & a_{2j} \\ \vdots & \vdots & \vdots & \vdots \\ a_{i1} & a_{i2} & \cdots & a_{ij} \end{vmatrix}$$

其中 $M=\left(a_{i1},a_{i2},\cdots,a_{ij}\right)$ 是 i 个备选方案的合集, $N=\left(a_{1j},a_{ij},\cdots,a_{ij}\right)$ 是 j 个属性的合集.

1) 首先, 对评价矩阵 A 进行标准化处理.

$$n_{ij}=x_{ij}/\sqrt{\sum_{j=1}^{m}x_{ij}^2},\ \ j=1,\cdots,m;\ i=1,\cdots,n$$

2) 确定各个指标之间的权重. 在此我们采用熵权法进行客观权重的确定, 采用熵公式计算每个属性的平均信息量$E(A)$:

$$E(A)=E\left[-\log p_i\right]=-\sum_{i=1}^{n}p_i\log p_i$$

在$E(A)$的基础上, 定义第 i 个评价指标的客观熵权被定义为:

$$\omega_i=\frac{1-E(A)_i}{m-\sum\limits_{i=1}^{m}E(A)_i},\ \ i=1,2,\cdots,m$$

3) 设 m 个评价指标的主观权重分别为 $\theta_1,\theta_2,\cdots,\theta_j$, 则第 i 个指标的真实权重被定义为:

$$r_i=\frac{\theta_i\omega_i}{\sum\limits_{i=1}^{m}\theta_i\omega_i}$$

4) 确定正理想解 $(IFPIS,Z^+)$ 和负理想理 $(IFNIS,Z^-)$

$$Z^+=\left(\max_i v_{ij}|j\in J_1\right),\left(\min_i v_{ij}|j\in J_2\right),\ \ i=1,2,\cdots,m$$
$$Z^-=\left(\min_i v_{ij}|j\in J_1\right),\left(\max_i v_{ij}|j\in J_2\right),\ \ i=1,2,\cdots,m$$

5) 计算距离尺度, 即采用相关系数计算每个目标 A 到正理想解$(IFPIS,Z^+)$和负理想解$(IFNIS,Z^-)$之间的的距离$D_\rho(A,Z^+)$和$D_\rho(A,Z^+)$:

$$D_\rho(A,Z^+)=\frac{E\{[A-E(A)]\,[Z^+-E(Z^+)]\}}{\sqrt{D(A)}\,\sqrt{D(Z^+)}}$$
$$D_\rho(A,Z^-)=\frac{E\{[A-E(A)]\,[Z^--E(Z^-)]\}}{\sqrt{D(A)}\,\sqrt{D(Z^-)}}$$

6) 计算决策目标与正负理想的贴近度C_i.

$$C_i=\frac{D(A,Z^-)}{D(A,Z^+)+D(A,Z^-)}$$

7) 根据C_i大小排序, 并输出排序结果.

3 实验验证

为了检验基于本文采用的相关系数度量方法的软件成本估算方法, 设计试验, 并采用公开数据集 Desharnais 进行本测评实验, Desharnais 数据集是由加拿大软件行业的统计结果数据, 它最早是在 1989 年由 Jean-Marc Desharnais 在对项目开发功能点数据的统计分析中应用. 在 Desharnais 数据集中有 12 种项目属性.

表 1 Desharnais 数据集的项目属性

属性名	标识	属性含义
Project	0	项目编号
Team Exp.	1	团队经验
Manager Exp.	2	管理经验
Year end	3	项目结束年份
Length	4	耗时
Effort	5	实际工作量
Transact	6	事务数
Entities	7	实体数
Points ajust	8	调整后功能点数
Envergure	9	环境系数
Points non ajust	10	未调整功能点数
Langage	11	开发环境

在本实验中, 省略了 4 个数据不完整的项目, 只利用其中 77 个完整的项目的数据进行试验.

选取数据集中 Team Exp、Manager Exp、Length、Transact、Entities、Points ajust、Envergure、Points non ajust、Langage 9 种属性做为本文软件成本估算的属性, Effort 属性作为估算的目标属性对预测结果的误差进行计算.

为了验证成本估算的有效性与准确性, 需要引入适当的评价标准. 常用的评价标准有许多, 我们采用以下两种作为标准:

① 平均误差率 (MMRE), 用于评估软件成本估算的平均误差情况, 计算公式如下:

$$MMRE(\%) = \frac{1}{n}\sum_{i=1}^{n} MRE * 100\%$$

② Pred(x), 用于评价成本估算的补充标准, 用于计算标准误差低于 x 的项目数量在整个数据集中所占的比例. x 的值通常设置为 25%. 计算公式如下:

$$\text{Pred}(x) = \frac{k}{n}$$

其中, k 为标准误差低于 x 的项目数量.

为了验证本文所提方法在软件方法估算中的准确性, 与经典的欧式距离和文献[16]所提及的相似性度量方法进行比较实验, 比较结果如表 2.

表 2 实验结果

评价标准度量方法	MMRE(%)	Pred25(%)
相关系数	40.21	42.61
欧氏距离	64	42
未知度Vague集	52.46	39.8

从上述实验结果可以看出, 本文提出的基于相关系数度量方法的软件成本估算方法, 在结果上优于经典的欧式距离和文献[16]提出的相似性度量公式.

4 结论

本文针对软件工程项目中需求分析阶段软件成本估算问题进行研究, 在皮尔逊相关系数的基础上, 综合考虑了主观权重和客观权重, 采用 TOPSIS 方法建立相关模型, 提出一种基于相关系数的 TOPSIS 方法用于软件成本估算问题研究, 并采用公开数据集 Desharnais 进行试验验证. 实验结果表明, 本文所采用基于相关系数的相似性度量方法较以往的方法有更高的准确率. 然而, 本文所提出的方法和研究工作只是单纯聚焦于处理与工作量相关的项目特征而没有忽略团队特性、员工发展等主观因素方面的考虑. 因此, 在未来的研究中, 会应将包括软件工程成员的性格、潜力、团队氛围等诸多特性充分考虑在内以更加准确地进行软件成本估算.

参考文献

1 史济民, 顾春华, 郑红. 软件工程: 原理、方法与应用. 北京: 高等教育出版社, 2009.
2 杨芙清. 软件工程技术发展思索. 软件学报, 2005, 16(1): 1–7.
3 The Standish Group. CHAOS report. http://www.standishgroup.com. 1995.
4 The Standish Group. 2004 the 3rd quarter research report. http://www.standishgroup.com, 2004.
5 李明树, 何梅, 杨达, 等. 软件成本估算方法及应用. 软件学报, 2007, 18(4): 775–795.
6 Boehm BW, Valerdi R. Achievements and challenges in software resource estimation[Technical Report]. No.USC-CSE-2005-513. http://sunset.usc.edu/publications/TECHRPTS/2005/usccse2005-513/usccse2005-513.pdf, 2005.
7 任雪利. 协同过滤在软件成本估算中的应用. 计算机系统应用, 2014, 23(6): 246–249.
8 任雪利, 代余彪. 软件相似度在成本估算中的应用. 计算机应用与软件, 2015, 32(6): 34–36, 112.
9 曹冬生, 王强军, 张元忠, 等. 基于类比的软件成本估算及其一种改进方法. 计算机工程与科学, 2009, 31(5): 102–106.
10 何慧, 张宏莉, 张伟哲, 等. 一种基于相似度的 DDoS 攻击检测方法. 通信学报, 2004, 25(7): 176–184.
11 朱永松, 国澄明. 基于相关系数的相关跟踪算法研究. 中国图象图形学报, 2004, 9(8): 963–967. [doi: 10.11834/jig.200408184]
12 何晓阳, 王亚沙. 基于模型的软件成本估计方法. 计算机研究与发展, 2006, 43(5): 777–783.
13 Jahanshahloo GR, Lotfi FH, Izadikhah M. An algorithmic method to extend TOPSIS for decision-making problems with interval data. Applied Mathematics and Computation, 2006, 175(2): 1375–1384. [doi: 10.1016/j.amc.2005.08.048]
14 Shih HS, Shyur HJ, Lee ES. An extension of TOPSIS for group decision making. Mathematical and Computer Modelling, 2007, 45(7-8): 801–813. [doi: 10.1016/j.mcm.2006.03.023]
15 周启超. BP 算法改进及在软件成本估算中的应用. 计算机技术与发展, 2016, 26(2): 195–198.
16 赵雪芬. 基于未知度的 Vague 集相似度量方法研究. 计算机工程与应用, 2013, 49(14): 130–132, 216. [doi: 10.3778/j.issn.1002-8331.1111-0518]