基于聚类分析的业务流程模型抽象

孙善武[1,2] 王 楠[1,2] 欧阳丹彤[3]
(吉林财经大学管理科学与信息工程学院 长春 130117)[1]
(吉林财经大学物流产业经济与智能物流吉林省重点实验室 长春 130117)[2]
(吉林大学符号计算与知识工程教育部重点实验室 长春 130012)[3]

摘 要 业务流程模型抽象的一个最突出的用例是对包含大量元素的业务流程细节模型进行"简要视图"的构造,以便对流程进行快速理解。很多学者对流程抽象方法进行了研究,提出根据行为的语义相似性对行为进行聚合,其中多数研究基于k-means聚类分析,即根据事先指定的抽象行为个数对行为进行聚类,在将行为聚合到某一个行为簇时,选择距离该行为簇的图心最近的行为。但实际上,抽象行为(子流程)个数是一个未知的量,哪些行为属于同一个子流程往往取决于建模者的经验和抽象习惯,而且在聚合时,若行为从业务意义角度或建模者的抽象习惯角度并不属于该子流程,则合并往往会产生抽象错误。因此,引入虚拟文档表示行为和流程模型,以消除固定属性作为表示行为的向量空间维度带来的约束。并且设计算法从大量包含人工设计子流程的真实的业务流程模型库中获取行为与所在子流程的距离阈值,利用该阈值指导生成可能获得的抽象行为个数k。以k为参数对流程模型进行行为聚类,在聚类过程中,进一步利用距离阈值对聚合行为进行限制。对真实的流程模型库进行实验分析,结果表明提出的行为聚类方法更加接近人工设计的抽象结果。

关键词 业务流程模型抽象,聚类分析,虚拟文档,行为聚合
中图法分类号 TP391 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.5.035

Business Process Model Abstraction Based on Cluster Analysis

SUN Shan-wu[1,2] WANG Nan[1,2] OUYANG Dan-tong[3]
(College of Management Science and Information Engineering, Jilin University of Finance and Economics, Changchun 130117, China)[1]
(Laboratory of Logistics Industry Economy and Intelligent Logistics, Jilin University of Finance and Economics, Changchun 130117, China)[2]
(Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130012, China)[3]

Abstract A prominent business process model abstraction(BPMA) use case is a construction of a process "quick view" for rapidly comprehending a complex process. Some researchers propose the process abstraction methods to aggregate the activities on the basis of their semantic similarity. Most of these methods focus on k-means cluster analysis which partitions an activity set into k clusters(k is predefined by users) and assigns an activity to a cluster whose centroid is the closest to this activity. But in fact, the number of the abstract activities(clusters or subprocesses) is unknown so that which activities are assigned to the same subprocess tends to depend on the designers' experiences or modeling habits. Moreover, if the activities are not part of particular cohesive groups from the point of view of experienced modelers, the initial merging decisions may contain some errors. In this paper, the virtual document was introduced to represent activities and process models in order to eliminate the constraint of using the particular attributes of activities as the dimensions of vector spaces. And an approach was presented that exploits an industrial process model repository to derive a distance threshold between an activity and the subprocess it belongs to. According to this threshold, an algorithm was proposed to generate a possible number k of the clusters. By using k as a parameter and the distance threshold as a further constraint, a clustering procedure was given to aggregate the activities into different clusters. In an experimental validation, we compared both the proposed approach and k-means clustering(without the distance threshold as a further constraint) with actual model decisions, showing a stronger correlation between the proposed approach and actual model decisions. As such, this paper contributes to the development of modeling support for effective process model abstraction, easing the use of business process models in practice.

Keywords Business process model abstraction, Clustering analysis, Virtual document, Activity aggregation

到稿日期:2015-03-18 返修日期:2015-08-30 本文受国家自然科学基金(61402193,61272208,61133011,60973089,61003101,61170092),吉林省教育厅"十二五"科学技术研究项目(2014160),吉林省教育科学"十二五"规划课题(GH150285),吉林省科技发展计划项目(20130522177JH),吉林省教育厅"十三五"科学技术研究项目(2016105)资助。

孙善武 副教授,主要研究方向为计算机网络应用等,E-mail:ctusunshanwu@126.com;王 楠 副教授,主要研究方向为抽象理论、流程建模等,E-mail:ctuwangnan@126.com(通信作者);欧阳丹彤 教授,主要研究方向为基于模型的诊断与自动推理等。

1 引言

终端用户很难理解包含超过 50 个元素的流程模型[1]。业务流程模型抽象中一个最突出的用例是对包含大量元素的业务流程细节模型进行"简要视图"的构造,以便对流程进行快速理解[2,3]。在这个上下文中,抽象模型中的每个粗粒度行为(子流程)与初始模型中的一组细节行为相关联。显然,存在不同的方法对行为进行聚合,从用户的角度,语义上属于同组的行为更具有特定的聚合价值[4]。基于结构的业务流程模型抽象[5-7]仅根据控制流发现粗粒度行为,而不考虑行为的域语义,因此无法回答诸如"怎样发现域相关的行为集合"的问题。为了克服基于结构的业务流程模型抽象的局限,一些学者提出根据行为的业务域语义对行为进行聚合[2,8-10]。文献[8]和文献[10]利用 k-means 聚类算法将行为集合划分成 k (k 是一个由用户事先指定的固定值)个簇,每个簇作为一个候选子流程。但是实际上,子流程(抽象行为)个数是一个未知的量,哪些行为应该属于同一个子流程,往往取决于建模者的经验和抽象习惯。文献[8]构建了流程模型集合抽象指纹,根据已经存在的包含子流程层次的模型,模拟有一定经验的建模者的抽象决策,对业务流程模型进行自动抽象。但是该方法存在以下问题:基于流程模型库包含的知识,用预设的行为属性来定义行为的向量空间的各个维度,对流程模型域约束较高;k-means 聚类算法将行为聚合到某一个行为簇时,选择距离该行为簇的图心最近的行为,如果该行为从业务意义角度或建模者的抽象习惯角度并不属于该子流程,则该合并往往会产生抽象错误。

本文引入虚拟文档[11]来表示行为或行为集合,以消除固定属性作为行为的向量空间维度带来的约束。设计算法从大量包含人工设计子流程的真实的业务流程模型库中获取行为与所在子流程的距离阈值,利用该阈值,采用贪心策略近似估算可能获得的抽象行为(子流程)个数 k。以 k 为参数对流程模型进行行为聚类,在聚类过程中,进一步用距离阈值对聚合行为进行限制,减少 k-means 聚类算法可能产生的抽象误差。

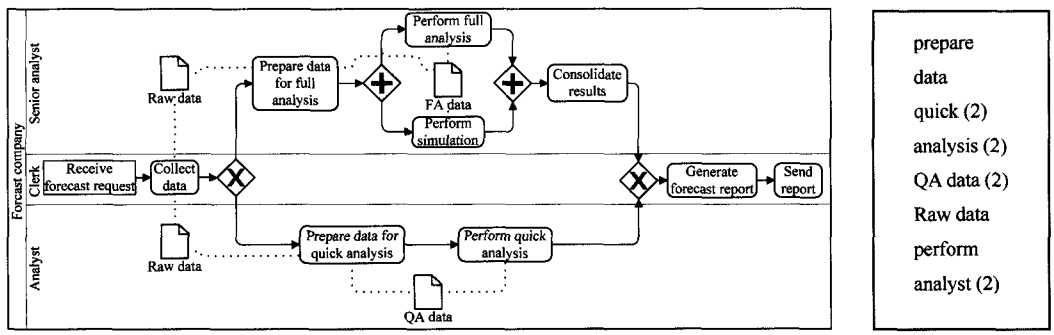
本文第 2.1 节引入虚拟文档的概念表示行为,并给出了一个运行实例;第 2.2 节设计了行为聚类算法,并给出了行为与行为簇间的距离计算方法以及距离阈值的确定过程。第 3 节用真实的流程模型进行了实验,并给出了实验结果分析。

2 行为聚类

2.1 行为的文档表示

为了计算两个行为之间或行为集合(簇)之间的距离,引入虚拟文档[11]来表示行为或行为集合。一个行为的虚拟文档由一些词构成,这些词来自于与该节点相关联的所有文本信息。在业务流程模型上下文中,一个行为的虚拟文档由一些术语的集合构成,这些术语由行为标签、执行角色标签(如果该信息可用)、输入/输出数据以及行为的文本描述生成[9]。一组行为的虚拟文档则通过合并所有行为的文档生成。虚拟文档的生成包含了术语的标准化、连接词过滤以及词干提取等[12]。给定两个虚拟文档,可以基于它们向量空间的距离计算其相似性,其中,维度就是出现在文档中的术语,各个维度的值使用术语出现的频率计算得到[13]。

例 1 引入文献[8]中的"生成预测报告"流程的例子,并对其进行了修改,如图 1(a)所示。该流程由 10 个行为、4 个子流程(每个子流程中包含的行为用不同颜色标识)构成,具体如表 1 所列。由于篇幅限制,仅给出子流程 S_3 对应的虚拟文档,如图 1(b)所示。



(a) (b)

图 1 流程实例[8]

表 1 图 1 中的流程模型细节描述

Receive forecast report	d_1	S_1
Collect data	d_2	
Prepare data for quick analysis	d_3	S_2
Perform quick analysis	d_4	
Perform simulation	d_5	
Consolidate results	d_6	
Prepare data for quick analysis	d_7	S_3
Perform quick analysis	d_8	
Generate forecast report	d_9	S_4
Send report	d_{10}	

2.2 流程模型抽象的聚类算法

将行为的聚合作为一个聚类分析问题,根据距离测量方法将相距较近的行为聚合生成行为簇,也就是流程抽象模型中的粗粒度行为(子流程)。聚类分析有很多不同的算法[14],k-means 聚类算法是比较容易实现并且可以很好地用于流程模型抽象的算法。但是正如本文前面所指出的,人工抽象流程模型获得的抽象行为个数是一个未知的量,预先设定参数 k 限制了抽象行为的生成。传统的利用迭代 k-means 算法[15]生成最佳 k 值的过程具有较高的复杂性。因此,本节设计贪心策略,根据事先确定的距离参考阈值 ω,动态生成抽象行为初始个数 k,并利用 ω 和 k 作为参数,实现行为聚类。具体过程如算法 1 和算法 2 所示。

算法 1 GenerateK

//根据事先指定的距离参考阈值 ω,生成待聚合行为簇的参考数 k 以及 k 个行为簇

1. A←行为集合
2. k←0

3. 循环，直到 A 为空

4. k←k+1

5. 任取 a∈A，C_k←{a}，A←A−{a}

6. 计算 D={d|d←dist(b,C_k)，b∈A}

7. 取 D 中的最小距离值 md←dist(mb,C_k)

8. 如果 md≤ω，则 C_k←C_k+{mb}，A←A−{mb}，转步骤 6

9. 否则，转步骤 3

10. 输出 k，C_1，…，C_k

算法 2 ActivitiesClustering

//行为聚类

//输入：k，ω，$C_1^{(0)}$，…，$C_k^{(0)}$

//输出：包含行为个数大于 1 的行为簇

1. t=0

2. 重复执行以下步骤，直到不发生任何行为的调整

2.1 对每个行为 a，计算 $d_j^{(t)} \leftarrow \min_{1\leqslant j\leqslant k}(\mathrm{dist}(a,C_j^{(t)}))$，若 $d_j^{(t)}\leqslant\omega$，则将 a 分配至 $C_j^{(t+1)}$；

2.2 t←t+1；

3. 输出包含行为个数大于 1 的行为簇

算法 GenerateK 随机选取待抽象模型中的行为 a 作为初始簇 C_1，利用贪心选择（在剩余行为中选择与 C_1 相距最近并且距离值不大于给定的距离参考阈值 ω 的行为）不断扩充 C_1，直到所有行为都处理过；然后重复该过程，直到行为集合 A 为空。此时，生成了 k 个簇（包括由单个行为构成的簇）。

以 k，C_1，…，C_k，以及 ω 作为输入，执行算法 ActivitiesClustering，生成最终行为簇。关于算法 ActivitiesClustering 的几点说明：

- 为了与 k 值生成过程保持一致，计算 $dist(a,C_j)$ 时，若 $a\in C_j$，计算 $dist(a,C_j-\{a\})$；
- 阈值 ω 用来限制每次并入簇的行为；
- 最终得到的行为簇的个数小于等于 k。

如果业务流程的抽象过程是由人工实现的，则抽象的结果中会隐含设计人员对于抽象的一些内在的标准[8]，即子流程内的行为与子流程外的行为到该子流程的距离应该存在一定差距。为了证实这种差距，并得到指导聚类算法运行的距离参考阈值，将行为 a 与行为簇 C 之间的包含关系定义为函数 $diff$：

$$diff(a,C)=\begin{cases}0, & \text{如果 } A\in C\\ 1, & \text{否则}\end{cases}$$

将行为 a 和行为簇 C 表示为虚拟文档后，分别用 4 种方法计算行为与行为簇之间的距离，即：

- SL(single-link)

$$dist_{SL}(a,C)=\min_{\substack{b\in C\\ b\neq a}} dist(d_a,d_b)$$

- CL(complete-link)

$$dist_{CL}(a,C)=\max_{\substack{b\in C\\ b\neq a}} dist(d_a,d_b)$$

- AL(average-link)，

$$dist_{AL}(a,C)=dist(d_a,d_b)$$

d_b 表示行为簇 C 的图心文档，即，$\vec{v_b}=\frac{1}{|C|}\sum_{\vec{v_d}\in C}\vec{v_d}$。

- VD(virtual-document)

直接计算行为的虚拟文档与行为簇的虚拟文档之间的距离，即 $dist_{VD}(a,C)=1-sim(d_a,d_C)$，其中 d_a 和 d_C 分别表示 a 和 C 对应的虚拟文档。

显然地，4 种计算行为与行为簇之间距离的过程根据行为簇表示的不同而不同，对于第一种和第二种方法，可以将每个子流程 S 表示为由多个单个虚拟文档构成的集合，即 $S=\{d_1,\cdots,d_m\}$；对于第三种方法，将每个子流程 S 对应的虚拟文档表示为 S 中所有虚拟文档的中心文档，即 $\vec{v_S}=\frac{1}{|S|}\sum_{\vec{v_d}\in S}\vec{v_d}$；对于第四种方法，将每个子流程 S 表示为 S 中所有虚拟文档的并集，如图 1(b)所示。

根据观测量和计算结果，利用斯皮尔曼相关系数讨论 $diff$ 值与 $dist$ 值之间的相关性，从中发现了与 $diff$ 值强关联的距离计算方法，并根据此方法从包含人工设计子流程的业务流程模型库中计算行为到所在子流程之间的距离阈值的可能参考范围。

设流程模型为 M，n 个行为对应的虚拟文档集合 $D=\{d_1,\cdots,d_n\}$，d_i 表示第 i 个行为对应的虚拟文档，其包含的子流程集合 $S=\{S_1,\cdots,S_k\}$，S_i 表示一个子流程。利用 4 种计算行为和子流程距离的方法求解对应的斯皮尔曼相关系数，公式如下：

$$\rho(dist,diff)=\frac{1}{k}\sum_{j=1}^{k}\rho_j(dist,diff)$$

其中，$\rho_j(dist,diff)=1-\frac{6\sum_{i=1}^{n}(dist(d_i,S_j)-diff(d_i,S_j))^2}{n(n-1)}$，$dist(d_i,S_j)$ 属于 SL、CL、AL 或 VD 中的某种距离计算公式。

例 2　以图 1(a)所示的流程模型为例，$dist_{AL}$ 值如表 2 所列。

表 2　图 1(a)中行为与各个子流程之间的 $dist_{AL}$ 值

$dist_{AL}$ activity \ Subprocess	S_1	S_2	S_3	S_4
d_1	0.5094	1.0000	1.0000	0.5477
d_2	0.5094	0.8356	0.7500	0.6985
d_3	0.7418	0.5286	0.6938	1.0000
d_4	1.0000	0.3333	0.6250	1.0000
d_5	1.0000	0.3333	0.8750	1.0000
d_6	1.0000	0.5286	1.0000	1.0000
d_7	0.7418	0.7315	0.2814	1.0000
d_8	1.0000	0.7534	0.2250	1.0000
d_9	0.5257	1.0000	1.0000	0.5955
d_{10}	0.6349	1.0000	1.0000	0.4296

计算 $\rho(dist_{AL},diff)=\frac{1}{4}\sum_{j=1}^{4}\rho_j(dist,diff)=0.9462$。由于本例行为数量较少，因此 4 种距离计算方法获得的 ρ 值相差不大，都表现出了强关联性[16]。

3　实验验证

为了验证提出的行为聚合方法对业务流程模型的抽象结果与人工抽象结果的相似程度，对真实的业务流程模型集合进行了理论验证，并对验证结果进行了解释。

3.1　实验构架

基于笔者所在的省重点实验室开放项目，从项目的合作单位——某大型合资汽车生产商及其伙伴物流公司获取了流程模型集合作为研究对象。该公司提供的模型表示规范，质量较高，很多节点数较多的模型已经进行了人工子流程设计。从中选取了 50 个行为标签描述规范、行为属性描述完整的模型（其中 40 个包含人工设计子流程层次结构）。为了利用出现在行为及其属性标签中的词，将其以向量空间的形式表示

为虚拟文档，我们与相关工作人员共同对术语进行了重新规范，并达成了共识。另外，为了获取尽量多的信息，进一步考虑了流程中的控制流信息，并将提取的词加入到相邻行为的虚拟文档中，单数字信息根据含义转化为变量名。由此，流程模型转化为了虚拟文档集合，其中保留子流程的层次关系。

需要指出的是，由于该公司的全局生产线的生产流程很复杂，流程模型涉及的域也较广，包括业务流程、生产流程、物流流程，并且生产流程中经常包含了各种零部件的物流子流程。因此，为了验证提出的方法不依赖于行为的具体域，保证实验结果的有效性，对于具有子流程层次的模型，按照以下原则对模型进行选取：①展开后规模适中；②包含尽量多的其它域子流程；③尽量不选取包含多于两层子流程的流程模型。表 3 给出了所选流程模型的相关属性。

表 3　实验流程模型的相关属性

	M_1-M_{40}			$M_{41}-M_{50}$
	行为数	子流程数	子流程包含行为数	行为数
平均	94.10	7.97	7.52	80.33
最大	127.00	20.00	10.50	109.00
最小	59.00	3.00	4.20	47.00

3.2　实验结果

对于 M_1-M_{30} 中的每个流程 M，计算 M 中的每个行为与 M 中包含的所有子流程之间的 $diff$ 值和距离 $dist$ 值。图 2 中给出了 30 个实验模型的 4 种 $dist$ 值（$dist_{SL}$，$dist_{CL}$，$dist_{AL}$，$dist_{VD}$）与 $diff$ 值的关联结果，$\rho(dist_{AL},diff)$ 与 $\rho(dist_{VD},diff)$ 的结果非常相似，其平均值分别为 0.7273 和 0.7230（$\rho(dist_{SL},diff)$ 与 $\rho(dist_{CL},diff)$ 结果的平均值分别为 0.4377和 0.2550），这个值的水平表示高度相关。因此，在 $dist_{AL}$ 与 $diff$（以及 $dist_{VD}$ 与 $diff$）测量值之间具有强关联性。

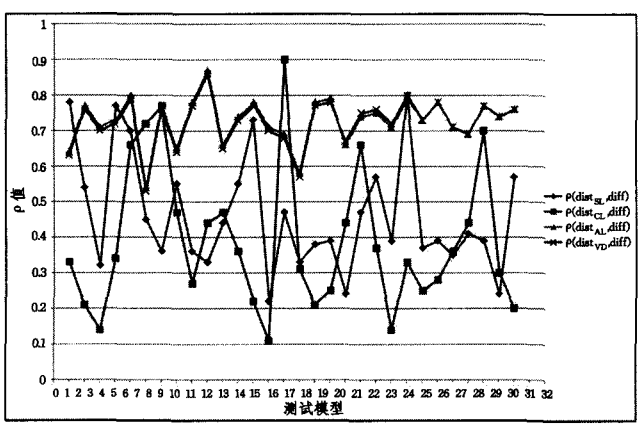


图 2　4 个 $dist$ 距离测量值与 $diff$ 值的关系

文中实验选取 $dist_{AL}$ 作为计算行为到行为集合的距离的方法，并在模型 M_1-M_{30} 中，对每个行为 $a_i\in S_j(1\leqslant i\leqslant m)$，计算 $d_i=dist_{AL}(a_i,S_j)$，分别用 $d_1,\cdots,d_m$ 的中位数、平均值、最小值和最大值作为距离参考阈值，分别记为 ω_1、ω_2、ω_3 和 ω_4，运行 GenerateK 算法生成初始 k 值，分别设为 k_1、k_2、k_3 和 k_4，与模型实际包含的子流程个数之间的值的对比情况如图 3 所示，其中 k_1 的最大值、最小值和平均值与实验模型库中的子流程数相差均不超过 35%（分别为 25%、33.33% 和 11.72%）。注意，由于 GenerateK 算法结果中包含由单个行为构成的簇，因此在图 3 的统计中事先将这种簇从生成的 k 值中排除。

聚类算法运行实验分为两部分：第一部分将 10 个包含人工设计子流程的流程模型（$M_{31}-M_{40}$）转换为对应的细节展开模型，分别用本文提出的算法 ActivitiesClustering 和文献[8]中使用的 k-means 聚类算法自动生成聚合行为簇，然后对比生成的行为簇与人工设计子流程之间的接近度；第二部分对 10 个不包含人工设计子流程的细节模型（$M_{41}-M_{50}$）同样运行这两个算法以自动生成行为簇，并将生成结果分发给参与本实验的相关人员，让其对各个行为簇进行人工评估和比较分析。

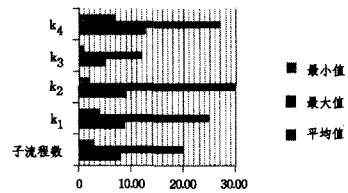


图 3　4 个获取的初始 k 值与实际的子流程数

引用文献[10]中定义的部分相关指标来比较人工设计的子流程分解与自动生成行为簇之间的各种特征，具体说明如下。

Precision：自动生成的并且同时是人工定义的子流程数除以自动生成的子流程数。

Recall：自动生成的并且同时是人工定义的子流程数除以人工定义的子流程数。

Overshoot：自动生成的候选子流程的节点中，不属于与该子流程匹配的人工定义的子流程的节点比例。

Undershoot：应该属于某个人工定义的子流程中但是在自动生成的候选子流程中没有生成的节点比例。

在比较自动生成的子流程与人工定义的子流程是否匹配时，采用节点匹配法，而不是流程的整体匹配法。各个测量指标的计算定义如下，详见文献[10]。

令 N 是一个流程中的所有节点集合（包括其子流程），$SP_M\subseteq P_N$ 是人工确定的子流程集合，$SP_A\subseteq P_N$ 是自动确定的候选子流程集合。$P_M\in SP_M$ 是一个人工确定的子流程，$P_A\in SP_A$ 是一个自动确定的候选子流程。P_A 与 P_M 之间的覆盖 Overlap 定义为[10]：

$$Overlap=\frac{|P_A\cap P_M|}{\max(|P_A|,|P_M|)}$$

如果 P_A 与 P_M 之间的覆盖 $Overlap>0$，并且不存在其他自动生成的子流程 $P_A'\in SP_A$ 与 P_M 之间的覆盖 $Overlap'>Overlap$，则称 P_A 是 P_M 的一个最相关匹配。令函数 $match$：$SP_M\rightarrow P_N$ 返回每个人工定义的子流程的最相关匹配，如果不存在，则返回空集。

Precision 和 Recall 定义如下[10]：

$$Precision=\frac{\sum_{P_M\in SP_M}|P_M\cap match(P_M)|}{\sum_{P_A\in SP_A}|P_A|}$$

$$Recall=\frac{\sum_{P_M\in SP_M}|P_M\cap match(P_M)|}{\sum_{P_M\in SP_M}|P_M|}$$

F 值定义为 Precision 和 Recall 两个值的调和平均数[10]：

$$F=\frac{2\cdot Precision\cdot Recall}{Precision+Recall}$$

Overshoot 和 Undershoot 分别定义如下[10]：

$$Overshoot=\frac{\sum_{P_M\in SP_M}|match(P_M)-P_M|}{\sum_{P_A\in SP_A}|P_A|}$$

$$Undershoot=\frac{\sum_{P_M\in SP_M}|P_M-match(P_M)|}{\sum_{P_M\in SP_M}|P_M|}$$

以对 M_1-M_{30} 挖掘获取的 ω_1 为阈值参数，对 10 个包含人工设计子流程的流程模型对应的细节展开模型运行算法 GenerateK，并以生成的 k 值作为参数执行算法 ActivitiesClustering 和文献[8]中使用的 k-means 聚类算法。由于篇幅限制，本文仅附上各个指标在 10 个模型获得的平均值结果统计，如表 4 所列。

表 4　模型 $M_{31}-M_{40}$ 的指标平均值

指标	算法 ActivitiesClustering	k-means 聚类算法[8]	原模型
子流程数	11.73	11.73	7.97
子流程的行为数	6.31	5.2	7.52
子流程行为数最大值	26.7	45.1	13.47
子流程行为数最小值	3	2.7	4.2
Precision	0.55	0.15	—
Recall	0.45	0.33	—
F 值	0.495	0.21	—
Overshoot 值	0.1	0.23	—
Undershoot 值	0.09	0.44	—

根据文献[10]，F 值是一个很重要的指标，它给出了自动子流程分解与人工设计的子流程分解之间的接近程度。根据表 4 的结果可以看出，算法 ActivitiesClustering 的抽象过程比文献[8]中使用的 k-means 聚类算法的划分方法要更加接近于人工划分结果。而文献[10]仅针对一个流程的实验结果中，连接性标准、标签相似性标准以及两者结合所获得的 F 值最大为 0.35。另外，算法 ActivitiesClustering 的抽象结果的 Overshoot 值和 Undershoot 值都非常低，说明该算法更好地对行为进行了归类。

第二部分实验对 10 个原始细节模型分别运行算法 ActivitiesClustering 和文献[8]中使用的 k-means 聚类算法，其输入的距离阈值参数仍为第一部分实验中获取的 ω_1；然后将自动抽象的结果模型分发给 10 个参与项目合作的工作人员，让其对每个抽象模型做人工分析和评估，即根据经验，每个工作人员独立在自动生成的行为簇的基础上进行节点扩充或删除，将其修正为符合人工设计经验的子流程。按照第一部分实验方法，对 10 个模型可以得到各个指标的 10 组不同的值，将 10 组值取平均后得到每个模型各个指标的实验评估结果。

为了简化，对每个行为簇 P_A，只统计工作人员人工添加节点的个数 n_{add}（表示本应属于该行为簇但是未被自动生成的节点）和删除节点的个数 n_{del}（表示自动生成但不属于该行为簇的节点），则 $|P_A|+n_{add}-n_{del}$ 表示人工修正后得到的子流程节点数。当 $|P_A|+n_{add}-n_{del}=0$ 时，表示该流程片段不构成有意义的子流程，即不匹配任何人工设计的子流程。

根据统计的数据，得到各个指标值的计算公式如下：

$$Overshoot^*=\left(\sum_{person\in Group}\frac{\sum_{P_A\in SP_A}n_{del}}{\sum_{P_A\in SP_A}|P_A|}\right)/|Group|$$

$$Undershoot^*=\left(\sum_{person\in Group}\frac{\sum_{P_A\in SP_A}n_{add}}{\sum_{P_A\in SP_A}(|P_A|+n_{add}-n_{del})}\right)/|Group|$$

$$Precision^*=\left(\sum_{person\in Group}\frac{\sum_{P_A\in SP_A}(|P_A|-n_{del})}{\sum_{P_A\in SP_A}|P_A|}\right)/|Group|$$

$$Recall^*=\left(\sum_{person\in Group}\frac{\sum_{P_A\in SP_A}(|P_A|-n_{del})}{\sum_{P_A\in SP_A}(|P_A|+n_{add}-n_{del})}\right)/|Group|$$

$$F^*=\left(\sum_{person\in Group}\frac{2\cdot Precision^*\,Recall^*}{Precision^*+Recall^*}\right)/|Group|$$

10 个模型的各项指标的平均值如表 5 所列。

表 5　对模型 $M_{41}-M_{50}$ 各个指标的实验评估结果

指标	算法 ActivitiesClustering	k-means 聚类算法[8]	人工修正后
子流程数	9.08	9.08	6.00
子流程的行为数	7.10	5.33	7.90
子流程行为数最大值	18.00	25.60	14.23
子流程行为数最小值	2.70	3.00	2.00
Precision*	0.76	0.31	—
Recall*	0.76	0.44	—
F* 值	0.76	0.36	—
Overshoot*	0.24	0.17	—
Undershoot*	0.24	0.35	—

在第二部分实验结果子流程中，为了把重点放在构建完整业务意义的子流程上，工作人员的修正过程只针对每个生成的行为簇，单独对每个行为簇进行节点添加或节点删除，而没有考虑最终的抽象流程连接逻辑，因此很多子流程在添加节点过程中，重用了相同的行为，甚至相同的其它子流程。从实验结果可以看出，虽然算法 ActivitiesClustering 得到的 Overshoot值的平均值有所上升，但是 F 值为 0.76，表明自动抽象结果非常接近人工的抽象结果。

结束语　本文利用虚拟文档表示业务流程模型，从包含人工设计子流程的业务流程模型库中分析行为与所在子流程之间距离的可能界限，利用该界限作为阈值参数动态生成细节模型可能产生的抽象行为个数，并以此为指导进行行为聚类，用大量真实的业务流程模型对聚类结果进行了比较分析。与目前其他基于聚类算法的业务流程模型抽象相比（比如文献[8,10]），本文的主要优势为：(1)用虚拟文档表示流程模型及构成行为，消除固定属性作为表示行为的向量空间维度带来的约束；(2)根据真实的业务流程模型库分析获取的阈值参数，以低复杂性的贪心方法自动生成可能的抽象行为个数 k，无需用户事先指定；(3)聚类过程中用阈值参数限制行为的聚合，减小由于不符合建模者的抽象习惯产生的抽象误差，如文献[8]中使用的 k-means 算法；(4)本文选取的实验模型数量和节点个数较多，并且不来源于单一域，因此更具有说服力（文献[10]仅使用一个模型获取实验结果，文献[8]使用 30 个模型，但最大节点数仅为 48 个）。

基于行为语义对行为进行聚类克服了基于结构的业务流程模型抽象方法[6,7]的某些局限，从聚合行为的语义信息角度生成具有独立业务语义的行为集合。但是，无论是从行为域本体角度选择强连接的行为集合[2,10]，还是利用非控制流

（下转第 229 页）

Clustering Algorithm Based on Adaptive Feature Selection[J]. Expert Systems with Applications, 2011, 38(3): 1393-1399

[6] Aggarwal C C, Han J W, Wang J Y, et al. A Framework for Clustering Evolving Data Streams[C]// Proc of the 29th International Conference on Very Large Data Bases. Berlin, Germany, 2003: 81-92

[7] Aggarwal C C, Yu P S. On Clustering Massive Text and Categorical Data Streams[J]. Knowledge and Information Systems, 2010, 24(2): 171-196

[8] Aggarwal C C, Han J W, Wang J Y, et al. A Framework for Projected Clustering of High Dimensional Data Streams[C]// Proc of the 30th International Conference on Very Large Data Bases. Toronto, Canada, 2004: 852 -863

[9] Liu Y B, Cai J R, Yin J, et al. Clustering Text Data Streams[J]. Journal of Computer Science & Technology, 2008, 23(1): 112-128

[10] Shi Zhong. Efficient Streaming Text Clustering[J]. Neural Networks, 2005, 18(5/6): 790-798

[11] Frey B J, Dueck D. Clustering by Passing Messages between Data Points[J]. Science, 2007, 315(5814): 972 -976

[12] Guo Xiu-juan, Chen Ying. Analysis and Application of AP Clustering algorithm [J]. Journal of Jilin Jianzhu University, 2013, 30(4): 58-61(in Chinese)
郭秀娟, 陈莹. AP 聚类算法的分析与应用[J]. 吉林建筑大学学报, 2013, 30(4): 58-61

[13] Strehl A, Ghosh J, Mooney R J. Impact of Similarity Measures on Web-page Clustering[C]// AAAI Workshop on AI for Web Search. 2000: 58-64

[14] Zhang Zhen, Wang Bin-qiang, Yi Peng, et al. A Hierarchical Combination of Semi Supervised Neighbor Propagation Clustering Algorithm [J]. Journal of Electronic & Information Technology, 2013, 35(3): 645-651(in Chinese)
张震, 汪斌强, 伊鹏, 等. 一种分层组合的半监督近邻传播聚类算法[J]. 电子与信息学报, 2013, 35(3): 645-651

[15] Zhang X L, Furtlehner C, Sebag M. Data Streaming with Affinity Propagation[C]// Proc of the European Conference on Machine Learning and Knowledge Discovery in Databases. Antwerp, Belgium, 2008: 628-643

[16] Karypis G. CLUTO-a clustering toolkit [OL]. 2002. http://www-users. cs. umn. edu/~karypis/cluto

(上接第 197 页)

模型元素的聚类分析[8]，都未考虑子流程构成行为之间的控制流相关性，生成的候选子流程中的构成行为结构松散，甚至无法保证模型的保序性[17]，因此，对生成的候选子流程进行控制流修正以及构建之间控制流关系是下一步的研究内容。另外，在第二部分实验中发现，人工修正后的流程片段之间出现很多重复的行为甚至其它流程片段，这也是本文方法存在的问题，即初始模型中的每个行为只属于一个抽象行为，因此，接下来的改进算法将重点研究解决行为或流程片段重用问题。

参 考 文 献

[1] Mendling J, Reijers H A, van der Aalst W M P. Seven Process Modeling Guidelines (7pmg) [J]. Information and Software Technology, 2010, 52(2): 127-136

[2] Smirnov S, Dijkman R, Mendling J, et al. Meronymy-based aggregation of activities in business process models[J]. Conceptual Modeling-ER 2010, Lecture Notes in Computer Science, 2010, 6412: 1-14

[3] Smirnov S, Reijers H A, Weske M H, et al. Business process model abstraction: a definition, catalog, and survey[J]. Distributed and Parallel Databases, 2012, 30(1): 63-99

[4] Smirnov S. Business Process Model Abstraction[D]. Germany: University of Potsdam, 2012

[5] Polyvyanyy A, Smirnov S, Weske M. Reducing Complexity of Large EPCs[C]// MobIS. Saarbrücken, Germany, 2008: 195-207

[6] Polyvyanyy A, Smirnov S, Weske M. On Application of Structural Decomposition for Process Model Abstraction[C]// Proceedings of the BPSC 2009. Leipzig, 2009: 110-122

[7] Vanhatalo J, Völzer H, Koehler J. The Refined Process Structure Tree[C]// Proceedings of the 6th International Conference on Business Process Management (BPM 2008). Milan, Italy, 2008: 100-115

[8] Smirnov S, Reijers H A, Weske M. A Semantic Approach for Business Process Model Abstraction[M]// Advanced Information Systems Engineering: 23rd International Conference(CAiSE 2011). London, UK, June 20-24, 2011. Springer, 2011: 497-511

[9] Weidlich M, Dijkman R, Mendling J. The ICoP framework-Identification of correspondences between process models, Advanced Information Systems Engineering[M]// Advanced Infromation Systems Engineering: 22nd International Conference (CAiSE 2010). Hammamet, Tunisia, June 7-9, 2010. Springer, 2010: 483-498

[10] Reijers H A, Mendling J, Dijkman R M. On the Usefulness of Subprocesses in Business Process Models: BPM Center Report BPM-10-03[R]. BPMcenter. org, 2010

[11] Qu Y, Hu W, Cheng G. Constructing virtual documents for ontology matching[C]// Proceedings of the 15th International Conference on World Wide Web. Edinburgh, Scetland, UK, 2006: 23-31

[12] Porter M F. An algorithm for suffix stripping [J]. Program, 1980, 14(3): 130-137

[13] Euzenat J, Shvaiko P. Ontology matching[M]. Springer-Verlag, 2007

[14] Schae_er S E. Graph Clustering[J]. Computer Science Review, 2007, 1(1): 27-64

[15] Zhou S, Xu Z, Tang X. New method for determining optimal number of clusters in K-means clustering algorithm [J]. Computer Engineering and Applications, 2010, 46(16): 27-31

[16] Franzblau A N. A Primer of Statistics for Non-statisticians [M]// Harcourt, Brace & World New York. 1958

[17] Liu D, Shen M. Workflow Modeling for Virtual Processes: an Order-preserving Process-view Approach[J]. Information Systems, 2003, 28(6): 505-532