

## プロセス改善のための イベントとリソースの関係に基づく形式概念解析による手法

池田 真土里      山本 章博

京都大学 情報学研究科

m.ikedai@iip.ist.i.kyoto-u.ac.jp    akihiro@i.kyoto-u.ac.jp

### 1 はじめに

本稿では、産業組織において実行されるプロセスの改善を目的として、実際のプロセスを記録したイベントログから形式概念解析を用いてプロセスの弱点を発見するための手法を提案する。

工場や病院など、あらゆる産業組織は生産や運営のためにプロセスを実行している。近年では、ほとんどの組織が実際のプロセスをイベント単位で記録しており、そのような記録をイベントログと呼ぶ [9]。例として、航空会社で欠航などを理由に発生する航空券の保障プロセスを記録したイベントログを表 1 に示す。表 1 の最初のレコードは「事例 1 においてイベントタイプ register request が (ヒューマン) リソース Pete によって 12 月 30 日 11:02 に実行された」というイベントを表す。一般に、イベントログは MXML [11], XES [13] といった半構造データとして記録される。本稿では、多くのイベントログに共通する構造に注目し、表 1 に示したように単純化し、 $L = (F, E)$  と定式化する。 $E$  と  $F$  はそれぞれイベントの集合、属性の集合を表し、イベント  $e \in E$  がとる属性  $f \in F$  の値を  $f(e)$  と表す。各イベント  $e \in E$  は全ての属性の値の組  $(f_1(e), f_2(e), \dots, f_n(e))$  として記録される。イベントログには決められた形式は存在しないものの、表 1 で用いられている 4 つの属性に加え、コストや場所などは、イベントを記録するために一般に採用されている主要な属性である。本研究では、これらのうちイベントタイプとリソースに注目し、これらの属性をそれぞれ  $f_e, f_r$  と表す。

イベントログを用いて、プロセスの発見、監視、改善を行うことで組織の生産性、信頼性の向上を目指す分野としてプロセスマイニングがある [6, 8, 9]。プロセス発見とはイベントログからプロセスモデルを構築することであり、プロセス中のイベントタイプの順番に注目したワークフローモデル [2, 3, 5, 12] や、リソース間のイベントタイプの共有関係などに注目したソーシャルネットワークモデル [7, 10] など、様々なものが提案されて

表 1: イベントログ  $L = (F, E)$ .

事例	イベントタイプ	リソース	時間
1	register request	Pete	30-12-2010.11:02
1	examine thoroughly	Sue	31-12-2010.10:06
1	check ticket	Mike	05-01-2011.15:12
1	decide	Sara	06-01-2011.11:18
1	reject request	Pete	07-01-2011.14:24
2	register request	Mike	30-12-2010.11:32
2	check ticket	Mike	30-12-2010.12:12
2	examine casually	Sean	30-12-2010.14:16
2	decide	Sara	05-01-2011.11:22
2	pay compensation	Ellen	08-01-2011.12:05
3	register request	Pete	30-12-2010.14:32
3	examine casually	Mike	30-12-2010.15:06
3	check ticket	Ellen	30-12-2010.16:34
3	decide	Sara	06-01-2011.09:18
3	reinitiate request	Sara	06-01-2011.12:18
3	examine thoroughly	Sean	06-01-2011.13:06
3	check ticket	Pete	08-01-2011.11:43
3	decide	Sara	09-01-2011.09:55
3	pay compensation	Ellen	15-01-2011.10:45

いる。しかし、これらのモデルはプロセスの把握や監視には有用であるものの、プロセスの改善の観点から「どこを改善すべきか？」という問いに答えることができない。そこで我々は、形式概念解析 [1, 4] を用いて、プロセスの改善のために補強すべきプロセスの弱点をイベントログから発見するための手法を提案する。

### 2 イベントログからの弱点発見

本手法では、まず、形式概念解析を用いるためにイベントログを形式文脈へと変換する。そして、形式文脈から形式概念を計算し、各形式概念の脆弱性を数値化した上で、値の大きい形式概念をプロセスの弱点として提示する。

形式文脈  $K = (G, M, I)$  とは、対象の集合  $G$  と属性の集合  $M$ 、それらの二項関係  $I \subseteq G \times M$  からなる。提案手法では、イベントタイプの値の集合を  $G$ 、リソースの値の集合を  $M$  とし、 $I = \{(g, m) \in G \times M \mid \exists e \in E. f_e(e) = g \wedge f_r(e) = m\}$  としてイベントログ  $L = (F, E)$  から形式文脈を作成する。また、 $\text{freq}((g, m)) = |\{e \in E \mid f_e(e) = g \wedge f_r(e) = m\}|$  とする。これは、イベントタイプが  $g$  か

表 2: 形式文脈  $K = (G, M, I)$ : 値は  $\text{freq}((g, m))$  を示す (ただし 0 は省略している).

	Pete	Sue	Mike	Sara	Sean	Ellen
register request	2		1			
examine thoroughly		1			1	
check ticket	1		2			1
decide				4		
reject request	1					
examine casually			1		1	
pay compensation						2
reinitiate request				1		

つりソースが  $m$  であるイベントの出現頻度を表す. 表 1 を変換し得られる形式文脈を表 2 に示す.

イベントタイプの部分集合  $A \subseteq G$  とリソースの部分集合  $B \subseteq M$  に関して,  $A^I = \{m \in M \mid \forall g \in A. (g, m) \in I\}$ ,  $B^I = \{g \in G \mid \forall m \in B. (g, m) \in I\}$  とする. このとき,  $(A, B)$  は  $A^I = B$  かつ  $A = B^I$  を満たすならば**形式概念**である. 例えば  $(\{\text{register request, check ticket}\}, \{\text{Pete, Mike}\})$  は形式概念である. 形式概念  $c = (A, B)$  について,  $\text{Ex}(c) = A$ ,  $\text{In}(c) = B$  とし,  $\text{freq}(c) = \sum_{g \in \text{Ex}(c)} \sum_{m \in \text{In}(c)} \text{freq}((g, m))$  とする. さらに  $\text{Ex}_\gamma(c) = \{g \in \text{Ex}(c) \mid \{g\}^I = \text{Ex}(c)\}$ ,  $\text{In}_\mu(c) = \{m \in \text{In}(c) \mid \{m\}^I = \text{In}(c)\}$  とする.

イベントタイプの集合とリソースの集合の組として得られる形式概念  $c$  に対して, その脆弱性  $\text{weak}(c)$  を

$$\text{weak}(c) = \left( \frac{1 + |\text{Ex}_\gamma(c)|}{1 + |\text{In}(c)|} \times \frac{1 + |\text{Ex}(c)|}{1 + |\text{In}_\mu(c)|} \right) \times \left( \frac{\text{freq}(c)}{|E|} \right)$$

と定義する. これは, イベントタイプ集合とリソース集合の組として表される部分プロセスのプロセス全体に対する影響力と出現頻度から構成されている. 第一項は部分プロセス  $c$  の持つ影響力を表し,  $c$  に関して, 多様なイベントタイプがわずかなリソースによってのみ実行可能な場合や, わずかなリソースが多様なイベントタイプに消費される場合に大きくなる. 直感的には, この項は部分プロセス  $c$  を実行する際の負担の大きさを表す. 第二項はイベントログ全体に対する部分プロセスの出現頻度である. つまり, 頻繁に実行される負担の大きい部分プロセス  $c$  の脆弱性  $\text{weak}(c)$  は大きくなる. 例えば, 表 2 に示した形式文脈から得られる形式概念  $(\{\text{register request, check ticket}\}, \{\text{Pete, Mike}\})$  の脆弱性は約 0.63 であり, これが最も脆弱な部分プロセスとなる. 実際, register request と check ticket は主に Pete と Mike によって実行されており, Pete と Mike はその他のイベントタイプの実行にも関わっている. 加えてこの部分プロセスは実行頻度が高い.

### 3 まとめ

本稿では, 形式概念解析を用いてイベントログからプロセスの弱点を発見する手法を提案した. 具体的には, イベントタイプとリソースの関係に注目することで, 部分プロセスを実行する際の負担の大きさを数値化し, 負担の大きさと実行頻度から部分プロセスの脆弱性を数値的に求めた. 今後の課題としては, プロセスの弱点を効率的に改善する方法の提案が挙げられる.

### 参考文献

- [1] B. A. Davey, and H. A. Priestley. *Introduction to Lattices and Order*. Cambridge University Press, 2002.
- [2] M. Dumas, and A. ter Hofstede. UML Activity Diagrams as a Workflow Specification Language. *« UML » 2001 – The Unified Modeling Language. Modeling Languages, Concepts, and Tools, Lecture Notes in Computer Science*, vol. 2185, pp. 76–90, 2001.
- [3] R. Flowers, and C. Edeki. Business Process Modeling Notation. *International Journal of Computer Science and Mobile Computing*, vol. 2, issue 3, pp. 35–40, 2013.
- [4] B. Ganter, and R. Wille. *Formal Concept Analysis: Mathematical Foundations*. Springer-Verlag New York Inc., 1997.
- [5] J. Mendling, and M. Nüttgens. EPC markup language (EPML): an XML-based interchange format for event-driven process chains (EPC). *Information Systems and e-Business Management*, vol. 4, issue 3, pp. 245–263, 2006.
- [6] A. Shtub, and R. Karni. *ERP - The Dynamics of Supply Chain and Process Management*. Springer, 2010.
- [7] M. Song, and W. van der Aalst. Towards Comprehensive Support for Organizational Mining. *Decis. Support Syst.*, vol. 46, issue 1, pp. 300–317, 2008.
- [8] W. van der Aalst, et al.. Process Mining Manifesto. *Business Process Management Workshops, Lecture Notes in Business Information Processing*, vol. 99, pp. 169–194, 2012.
- [9] W. van der Aalst. *Process Mining - Discovery, Conformance and Enhancement of Business Processes*. Springer, 2011.
- [10] W. van der Aalst, H. Reijers, and M. Song. Discovering Social Networks from Event Logs. *Comput. Supported Coop. Work*, vol. 14, issue 6, pp. 549–593, 2005.
- [11] W. van der Aalst, B. van Dongen, J. Herbst, L. Maruster, G. Schimm, and A. Weijters. Workflow Mining: A Survey of Issues and Approaches. *Data Knowl. Eng.*, vol. 47, issue 2, pp. 237–267, 2003.
- [12] B. van Dongen, A. Alves de Medeiros, and L. Wen. Process Mining: Overview and Outlook of Petri Net Discovery Algorithms. *Transactions on Petri Nets and Other Models of Concurrency II, Lecture Notes in Computer Science*, vol. 5460, pp. 225–242, 2009.
- [13] H. Verbeek, J. Buijs, B. van Dongen, and W. van der Aalst. XES, XESame, and ProM 6. *Information Systems Evolution, Lecture Notes in Business Information Processing*, vol. 72, pp. 60–75, 2011.