Applied Soft Computing 11 (2011) 1697–1710

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Process discovery in event logs: An application in the telecom industry

Stijn Goedertier[a], Jochen De Weerdt[a,*], David Martens[a,b], Jan Vanthienen[a], Bart Baesens[a,c]

[a] Department of Decision Sciences and Information Management, Katholieke Universiteit Leuven, Naamsestraat 69, B-3000 Leuven, Belgium
[b] Department of Business Administration and Public Management, Hogeschool Gent, Universiteit Gent, Voskenslaan 270, B-9000 Ghent, Belgium
[c] School of Management, University of Southampton, Highfield Southampton, SO17 1BJ, United Kingdom

ARTICLE INFO

Article history:
Received 14 December 2009
Received in revised form 4 February 2010
Accepted 30 April 2010
Available online 8 May 2010

Keywords:
Process discovery
AGNEs
HeuristicsMiner
Event logs
Genetic Miner
Data mining
Workflow management systems (WfMS)

ABSTRACT

The abundant availability of data is typical for information-intensive organizations. Usually, discerning knowledge from vast amounts of data is a challenge. Similarly, discovering business process models from information system event logs is definitely non-trivial. Within the analysis of event logs, process discovery, which can be defined as the automated construction of structured process models from such event logs, is an important learning task. However, the discovery of these processes poses many challenges. First of all, human-centric processes are likely to contain a lot of noise as people deviate from standard procedures. Other challenges are the discovery of so-called non-local, non-free choice constructs, duplicate activities, incomplete event logs and the inclusion of prior knowledge. In this paper, we present an empirical evaluation of three state-of-the-art process discovery techniques: Genetic Miner, AGNEs and HeuristicsMiner. Although the detailed empirical evaluation is the main contribution of this paper to the literature, an in-depth discussion of a number of different evaluation metrics for process discovery techniques and a thorough discussion of the validity issue are key contributions as well.

1. Introduction

Organizations currently face an information paradox: the more they automate their processes, the less they are capable of monitoring and understanding them. A good understanding of processes is nonetheless vital for fulfilling business requirements such as verifying and guaranteeing business process compliance [26], setting up a coherent access control policy [14] and optimizing and redesigning business processes [17]. A better understanding will eventually enable organizations to provide better, automated support for their business processes in flexible, process-aware information systems [8,42].

Traditionally, practitioners have been obtaining insight into processes using interviewing techniques. A new and promising way of acquiring insights into business processes is the analysis of the event logs of information systems [33]. In many organizations, such event logs conceal an untapped reservoir of knowledge about the way employees and customers conduct every-day business transactions. Event logs are already available in many organizations. Popular Enterprise Resource Planning (ERP) systems such as SAP R/3, Oracle e-Business Suite and workflow management systems (WfMS) such as ARIS, TIBCO and Microsoft Biztalk already keep track of these event logs.

The topic of process discovery is relatively new and can be situated at an intersection of the fields of Business Process Management (BPM) and data mining [27]. It is inherently related to data mining and to the more general domain of knowledge discovery in databases (KDD) since the nature of its objectives is extracting useful information from large data repositories. Likewise, process discovery is strongly associated with BPM because of its purpose of gaining insight into business processes. As a result, process mining fits flawlessly into the BPM life cycle framework [34,41,47].

Because of the rather novelty of process discovery, it is definitely valuable to discuss various state-of-the-art discovery algorithms and assess them in a real-life setting. In order to do so, the remainder of this paper is structured as follows. In Section 2, process discovery and its main challenges are discussed and some basic concepts of Petri net theory are briefly introduced. Section 3 outlines a number of state-of-the-art process discovery techniques. Section 4 provides a discussion on evaluation metrics. In Section 5, three of the discussed algorithms will be applied on a real-life event log. Finally, the conclusions are formulated in Section 6.

2. Preliminaries

2.1. Process discovery

The basic idea of process discovery or control-flow discovery is straightforward: given an event log, automatically compose a

* Corresponding author. Tel.: +32 16 32 68 87; fax: +32 16 32 66 24.
E-mail addresses: stijn.goedertier@econ.kuleuven.be (S. Goedertier), jochen.deweerdt@econ.kuleuven.be (J. De Weerdt), david.martens@econ.kuleuven.be (D. Martens), jan.vanthienen@econ.kuleuven.be (J. Vanthienen), bart.baesens@econ.kuleuven.be (B. Baesens).

1568-4946/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.asoc.2010.04.025

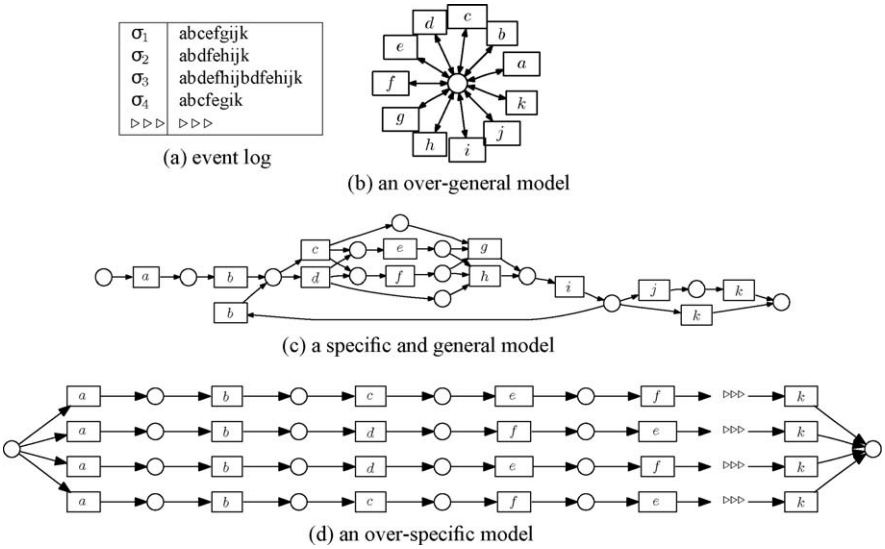

Fig. 1. DriversLicensel – discovery of a driver's license application process – (the transitions correspond to activity types that have the following meaning: *a* start, *b* apply for license, *c* attend classes cars, *d* attend classes motor bikes, *e* obtain insurance, *f* theoretical exam, *g* practical exam cars, *h* practical exam motor bikes, *i* get result, *j* receive license, and *k* end). (a) Event log, (b) an over-general model, (c) a specific and general model and (d) an over-specific model.

suitable process model that describes the behavior seen in the log. Auspiciously, processes occur in a more or less structured fashion, containing structures such as or-joins, or-splits, and-joins, and-splits, and loops. More accurately, the learning task can be formulated as follows: given an event log that contains the events about a finite number of process instances, find a model that correctly summarizes the behavior in the event log, striking the right balance between generality (allowing enough behavior) and specificity (not allowing too much behavior). For the purpose of process discovery, processes are often represented as workflow nets, a special subclass of Petri nets. Fig. 1 illustrates the learning problem for a Driver's License application process. Given the event log in Fig. 1(a), different process models can be conceived that portray similar behavior as the event log. The Petri net in Fig. 1(b) is capable of parsing every sequence in the event log. However, it can be considered to be overly general as it allows any activity to occur in any order. In contrast, the Petri net in Fig. 1(d) is overly specific, as it provides a mere enumeration of the different sequences in the event log. The Petri net in Fig. 1(c) is likely to be the more suitable process model. It is well-structured, and strikes a reasonable balance between specificity and generality, allowing for instance an unseen sequence *abcefgik*, but disallowing random behavior.

Process discovery is particularly useful in the context of human-centric processes for which an information system does not enforce the activities to be carried out in a particular order [30]. The extraction of useful knowledge from human-centric event logs often goes beyond the ability of descriptive statistics. What is additionally required is a process model that correctly summarizes the event log and that describes how business processes have actually taken place. Process discovery can provide answers to many business questions. In the first place, process discovery enables organizations to unveil implicit processes and tacit knowledge. Moreover, the availability of a model that accurately describes the behavior in an event log allows for advanced analysis such as the identification of performance bottlenecks or the localization of paths in the process model that are not compliant to existing regulations. The business value of process discovery is well illustrated by the plugins within the ProM Framework [31,37]. In analogy with the WEKA toolset for data mining [12,46], the ProM Framework consists of a large number of plugins for the analysis of event logs. The *Conformance Checker* plugin [25], for instance, allows identifying the discrepancies between an idealized process model and an event log. Moreover, with a model that accurately describes the event log, it becomes possible to use the time-information in an event log for the purpose of performance analysis, using, for instance, the *Performance Analysis with Petri nets* plugin.

2.2. Challenging problems

The discovery of process models from event logs faces many challenges [35]:

- **Incomplete logs**: Process discovery is inherently a descriptive learning task that aims at accurately summarizing an event log such that the discovered process model models all allowable behavior (recall) but does not allow for behavior that is not present in the event log (precision). Nonetheless, the ability to generalize beyond observed behavior can also be important to the problem of process discovery.
- **Noise**: Human-centric processes are prone to exceptions and logging errors. This causes additional low-frequent behavior to be present in the event log that is unwanted in the process model to be discovered. Consequently, process discovery algorithms face the challenge of not overfitting this noise.
- **Unsupervised learning**: An inherent difficulty of process discovery is that it is often limited to the much more difficult setting of unsupervised learning. Event logs rarely contain negative information about state transitions that were prevented from taking place.
- **History-dependent behavior**: Human-centric processes portray behavior that is dependent on the non-immediate history. An example is the occurrence of history-based joins in the control flow of a business process [39]. A history of related events can therefore be a strong predictor and is readily available in event logs. However, *non-local* behavior of business processes already presents many challenges for process modeling [2,40]. For process discovery, the inclusion of such non-local, non-free choice in the hypothesis space of process mining algorithms raises difficulties with regard to search space complexity and hypothesis visualization [45].
- **Inclusion of case data**: The routing choices that are made in business processes can also be dependent on the value of its data