Desire lines in big data : using event data for process discovery and conformance checking

3 Desire Lines in Big Data: Using Event Data for Process Discovery and Conformance Checking

■ **Wil van der Aalst**, Eindhoven University of Technology, Department of Mathematics and Computer Science, Eindhoven, The Netherlands, w.m.p.v.d.aalst@tue.nl

Abstract

Recently, the Task Force on Process Mining released the *Process Mining Manifesto*. The manifesto is supported by 53 organizations and 77 process mining experts contributed to it. The active contributions from end-users, tool vendors, consultants, analysts, and researchers illustrate the growing relevance of process mining as a bridge between data mining and business process modeling. This paper summarizes the manifesto and explains why process mining is a highly relevant, but also very challenging, research area. This way we hope to stimulate the broader IS (Information Systems) and KM (Knowledge Management) communities to look at *process-centric knowledge discovery*. This paper summarizes the manifesto and is based on a paper with the same title that appeared in the December 2011 issue of SIGKDD Explorations (Volume 13, Issue 2).

3.1 Process Mining

Process mining is a relatively young research discipline that sits between computational intelligence and data mining on the one hand, and process modeling and analysis on the other hand. The idea of process mining is to discover, monitor and improve real processes (i.e., not assumed processes) by extracting knowledge from event logs readily available in today's (information) systems (van der Aalst, 2011). Process mining includes (automated) process discovery (i.e., extracting process models from an event log), conformance checking (i.e., monitoring deviations by comparing model and log), social network/organizational mining, automated construction of simulation models, model extension, model repair, case prediction, and history-based recommendations.

Figure 6 illustrates the scope of process mining. Starting point for process mining is an *event log*. All process mining techniques assume that it is possible to *sequentially* record events such that each event refers to an *activity* (i.e., a well-defined step in some process) and is related to a particular *case* (i.e., a process instance). Event logs may store additional information about events. In fact, whenever possible, process mining techniques use extra information such as the *resource* (i.e., person or device) executing or initiating the activity, the timestamp of the event, or *data elements* recorded with the event (e.g., the size of an order).

Event logs can be used to conduct three types of process mining (van der Aalst, 2011; IEEE Task Force on Process Mining, 2011). The first type of process mining is *discovery*. A discovery technique takes an event log and produces a model without using any a-priori information. Process discovery is the most prominent process mining technique. For many organizations it is surprising to see that existing techniques are indeed able to discover real processes merely based on example executions in event logs. The second type of process mining is *conformance*. Here, an existing process model is compared with an event log of the same process. Conformance checking can be used to check if reality, as recorded in the log, conforms to the model and vice versa. The third type of process mining is *enhancement*. Here, the idea is to extend or improve an existing process model using information about the actual process recorded in some event log. Whereas conformance checking measures the alignment between model and reality, this third type of process mining aims at changing or extending the a-priori model. For instance, by using timestamps in the event log one can extend the model to show bottlenecks, service levels, throughput times, and frequencies.

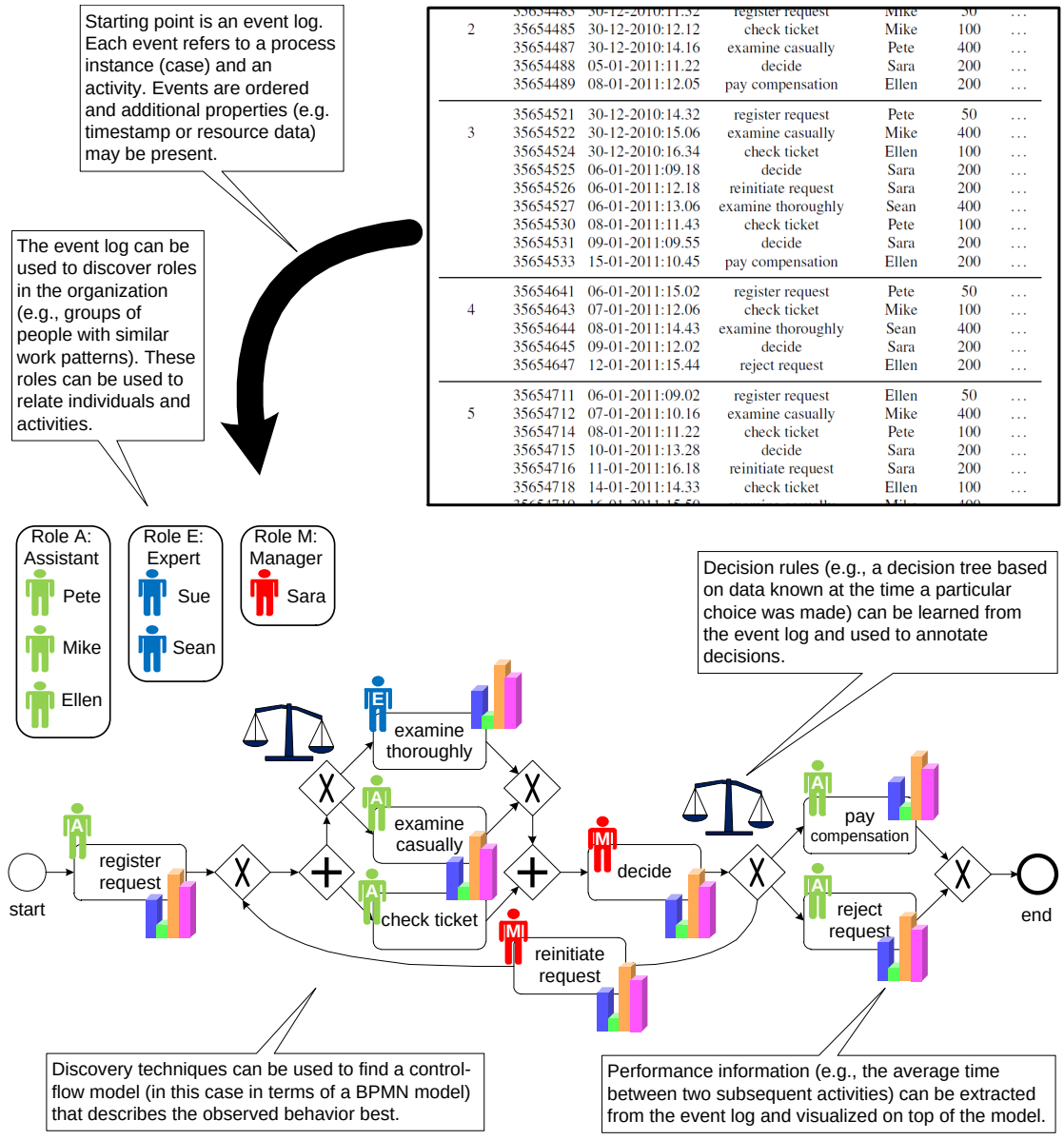

Figure 6: Process mining techniques extract knowledge from event logs in order to discover, monitor and improve processes

Figure 6 shows how first an end-to-end process model is discovered. The model is visualized as a BPMN (Business Process Modeling Notation) model, but internally algorithms are often using more formal notations such as Petri nets, C-nets, and transition systems (van der Aalst, 2011). By replaying the event log on the model it is possible to add information on bottlenecks, decisions, roles, and resources.

3.2 IEEE Task Force on Process Mining

The growing interest in log-based process analysis motivated the establishment of the IEEE Task Force on Process Mining. The goal of this task force is to promote the research, development, education, and understanding of process mining. The task force was established in 2009 in the context of the Data Mining Technical Committee of the Computational Intelligence Society of the IEEE. Members of the task force include representatives of more than a dozen commercial software vendors (e.g., Pallas Athena, Software AG, Futura Process Intelligence, HP, IBM, Fujitsu, Infosys, and Fluxicon), ten consultancy firms (e.g., Gartner and Deloitte) and over twenty universities.

Concrete objectives of the task force are: to make end-users, developers, consultants, managers, and researchers aware of the state-of-the-art in process mining, to promote the use of process mining techniques and tools, to stimulate new process mining applications, to play a role in standardization efforts for logging event data, to organize tutorials, special sessions, workshops, panels, and to publish articles, books, videos, and special issues of journals. For example, in 2010 the task force standardized XES (`www.xes-standard.org`), a standard logging format that is extensible and supported by the OpenXES library (`www.openxes.org`) and by tools such as ProM, XESame, Nitro, etc. See `http://www.win.tue.nl/ieeetfpm/` for recent activities of the task force.

3.3 Process Mining Manifesto

The IEEE Task Force on Process Mining recently released a manifesto describing *guiding principles* and *challenges* (IEEE Task Force on Process Mining, 2011). The manifesto aims to increase the visibility of process mining as a new tool to improve the (re)design, control, and support of operational business processes. It is intended to guide software developers, scientists, consultants, and end-users. As an introduction to the state-of-the-art in process mining, we briefly summarize the main findings reported in the manifesto (IEEE Task Force on Process Mining, 2011).

3.3.1 Guiding Principles

As with any new technology, there are obvious mistakes that can be made when applying process mining in real-life settings. Therefore, the six guiding principles listed in Table 2 aim to prevent users/analysts from making such mistakes. As an example, consider guiding principle *GP4*: “Events Should Be Related to Model Elements”. It is a misconception that process mining is limited to control-flow discovery, other perspectives such as the organizational perspective, the time perspective, and the data perspective are equally important. However, the control-flow perspective (i.e., the ordering of activities) serves as the layer connecting the different perspectives. Therefore, it is important to relate events in the log to activities in the model. Conformance checking and model enhancement heavily rely on this relationship. After relating events to model elements, it is possible to “replay” the event log on the model (van der Aalst, 2011). Replay may be used to reveal discrepancies between an event log and a model, e.g., some events in the

log are not possible according to the model. Techniques for conformance checking quantify and diagnose such discrepancies. Timestamps in the event log can be used to analyze the temporal behavior during replay. Time differences between causally related activities can be used to add average/expected waiting times to the model. These examples illustrate the importance of guiding principle GP4; the relation between events in the log and elements in the model serves as a starting point for different types of analysis.

GP 1	**Event Data Should Be Treated as First-Class Citizens**: Event should be *trustworthy*, i.e., it should be safe to assume that the recorded events actually happened and that the attributes of events are correct. Event logs should be *complete*, i.e., given a particular scope, no events may be missing. Any recorded event should have well-defined *semantics*. Moreover, the event data should be *safe* in the sense that privacy and security concerns are addressed when recording the event log.
GP 2	**Log Extraction Should Be Driven by Questions**: Without concrete questions it is very difficult to extract meaningful event data. Consider, for example, the thousands of tables in the database of an ERP system like SAP. Without questions one does not know where to start.
GP 3	**Concurrency, Choice and Other Basic Control-Flow Constructs Should be Supported**: Basic workflow *patterns* supported by all mainstream languages (e.g., BPMN, EPCs, Petri nets, BPEL, and UML activity diagrams) are *sequence, parallel routing* (AND-splits/joins), *choice* (XOR-splits/joins), and *loops*. Obviously, these patterns should be supported by process mining techniques.
GP 4	**Events Should Be Related to Model Elements**: Conformance checking and enhancement heavily rely on the relationship between *elements in the model* and *events in the log*. This relationship may be used to "replay" the event log on the model. Replay can be used to reveal discrepancies between event log and model (e.g., some events in the log are not possible according to the model) and can be used to enrich the model with additional information extracted from the event log (e.g., bottlenecks are identified by using the timestamps in the event log).
GP 5	**Models Should Be Treated as Purposeful Abstractions of Reality**: A model derived from event data provides a *view on reality*. Such a view should serve as a purposeful abstraction of the behavior captured in the event log. Given an event log, there may be multiple views that are useful.
GP 6	**Process Mining Should Be a Continuous Process**: Given the dynamical nature of processes, it is not advisable to see process mining as a one-time activity. The goal should not be to create a fixed model, but to breathe life into process models such that users and analysts are encouraged to look at them on a daily basis.

Table 2: Six guiding principles listed in the manifesto

3.3.2 Challenges

Process mining is an important tool for modern organizations that need to manage non-trivial operational processes. On the one hand, there is an incredible growth of event data. On the other hand, processes and information need to be aligned perfectly in order to meet requirements related to compliance, efficiency, and customer service. Despite the applicability of process mining there are still important challenges that need to be addressed; these illustrate that process mining is an emerging discipline. Table 3 lists the eleven challenges described in the manifesto (IEEE Task Force on Process Mining, 2011). As an example consider Challenge *C4*: "Dealing with Concept Drift". The term *concept drift* refers to the situation in which the process is changing while being analyzed. For instance, in the beginning of the event log two activities may be concurrent whereas later in the log these activities become sequential. Processes may change due to periodic/seasonal changes (e.g., "in December there is more demand" or "on Friday afternoon there are fewer employees available") or due to changing conditions (e.g., "the market is getting more competitive"). Such changes impact processes and it is vital to detect and analyze them. However, most process mining techniques analyze processes as if they are in steady-state.

C 1	**Finding, Merging, and Cleaning Event Data**: When extracting event data suitable for process mining several challenges need to be addressed: data may be *distributed* over a variety of sources, event data may be *incomplete*, an event log may contain *outliers*, logs may contain events at *different level of granularity*, etc.
C 2	**Dealing with Complex Event Logs Having Diverse Characteristics**: Event logs may have very different characteristics. Some event logs may be extremely large making them difficult to handle whereas other event logs are so small that not enough data is available to make reliable conclusions.
C 3	**Creating Representative Benchmarks**: Good benchmarks consisting of example data sets and representative quality criteria are needed to compare and improve the various tools and algorithms.
C 4	**Dealing with Concept Drift**: The process may be changing while being analyzed. Understanding such concept drifts is of prime importance for the management of processes.
C 5	**Improving the Representational Bias Used for Process Discovery**: A more careful and refined selection of the representational bias is needed to ensure high-quality process mining results.
C 6	**Balancing Between Quality Criteria such as Fitness, Simplicity, Precision, and Generalization**: There are four competing quality dimensions: (a) fitness, (b) simplicity, (c) precision, and (d) generalization. The challenge is to find models that score good in all four dimensions.
C 7	**Cross-Organizational Mining**: There are various use cases where event logs of multiple organizations are available for analysis. Some organizations work together to handle process instances (e.g., supply chain partners) or organizations are executing essentially the same process while sharing experiences, knowledge, or a common infrastructure. However, traditional process mining techniques typically consider one event log in one organization.
C 8	**Providing Operational Support**: Process mining is not restricted to off-line analysis and can also be used for online operational support. Three operational support activities can be identified: *detect*, *predict*, and *recommend*.
C 9	**Combining Process Mining With Other Types of Analysis**: The challenge is to combine automated process mining techniques with other analysis approaches (optimization techniques, data mining, simulation, visual analytics, etc.) to extract more insights from event data.
C 10	**Improving Usability for Non-Experts**: The challenge is to hide the sophisticated process mining algorithms behind user-friendly interfaces that automatically set parameters and suggest suitable types of analysis.
C 11	**Improving Understandability for Non-Experts**: The user may have problems understanding the output or is tempted to infer incorrect conclusions. To avoid such problems, the results should be presented using a suitable representation and the trustworthiness of the results should always be clearly indicated.

Table 3: Some of the most important process mining challenges identified in the manifesto

3.4 What Makes Process Discovery Challenging?

Although the process mining spectrum is much broader than just learning process models (see for example conformance checking and model enhancement), process discovery is by far the toughest problem. Discovering end-to-end processes is much more challenging than classical data mining problems such as classification, clustering, regression, association rule learning, and sequence/episode mining.

Why is process mining such a difficult problem? There are obvious reasons that also apply to many other data mining and machine learning problems, e.g., dealing with noise, concept drift, and a complex and large search space. However, there are also some specific problems:

- there are *no negative examples* (i.e., a log shows what has happened but does not show what could not happen);
- due to concurrency, loops, and choices the *search space has a complex structure* and the log typically contains only a *fraction* of all possible behaviors;
- there is no clear *relation between the size of a model and its behavior* (i.e., a smaller model may generate more or less behavior although classical analysis and evaluation methods typically assume some monotonicity property); and
- there is a need to balance between four (often) *competing quality criteria* (see Challenge C6): (a) *fitness* (be able to generate the observed behavior), (b) *simplicity* (avoid large and complex models), (c) *precision* (avoid "underfitting"), and (d) generalization (avoid "overfitting").

To illustrate the challenging nature of process mining we consider the process model shown in Figure 7. This Petri net models the process that starts with a and ends with d. In-between k activities can occur in parallel. For parallel branch i there is choice between b_i and c_i. The process model is able to generate $2^k k!$ different traces, i.e., for $k = 10$ there are 3,715,891,200 possible execution sequences. Two example traces are $a\ c5\ b3\ c1\ b2\ b4\ c6\ c8\ b7\ c9\ c10\ d$ and $a\ b1\ c2\ b3\ c4\ b5\ c6\ b7\ c8\ b9\ c10\ d$. Concurrency and choice typically result in heaps of possible traces. In fact, if there are loops, there are potentially infinitely many traces. Hence, it is completely unrealistic to assume that all possible traces will be observed in some event log. Even for smaller values of k and event logs with millions of cases, it is often still unlikely that all possible traces will be seen.

Fortunately, existing process discovery algorithms do not need to see all possible interleavings to learn a model with concurrency. For example, the classical α algorithm can learn the Petri net based on less than $4k(k-1)$ example traces. For the α algorithm it is sufficient to see all "direct successions" rather than all "interleavings", i.e., if x can be directly followed by y it should be observed at least once.

Traditional knowledge discovery techniques are unable to discover the process model shown in Figure 7. However, for organizations interested in process improvement and compliance it is essential to discover the actual processes and these exhibit the control-flow patterns used in Figure 7. Various management trends related to process improvement (e.g., Six Sigma, TQM, CPI, and CPM) and compliance (SOX, BAM, etc.) can benefit from process mining.

Therefore, we hope that the manifesto will stimulate the IS and KM communities to think about new techniques for process-centric knowledge discovery.

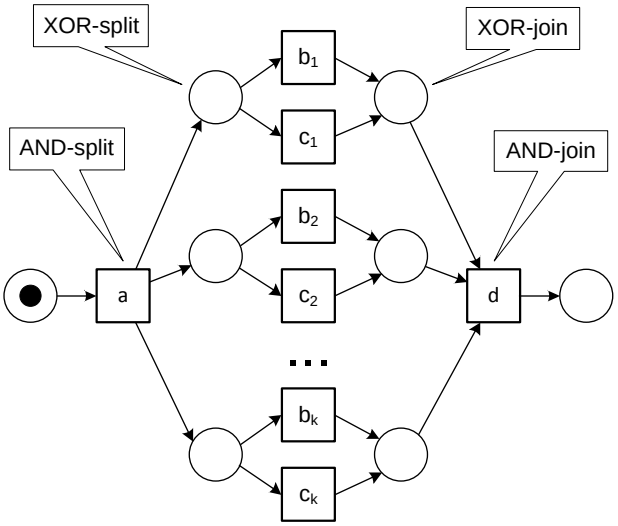

Figure 7: A Petri net with $2^k \cdot k!$ possible execution sequences

3.5 Learn More About Process Mining?

The process mining manifesto can be obtained from `http://www.win.tue.nl/ieeetfpm/`. The manifesto has been translated into Chinese, German, French, Spanish, Greek, Italian, Korean, Dutch, Portuguese, Turkish, and Japanese. The reader interested in process mining is also referred to the recent book on process mining (van der Aalst, 2011). Also visit `www.processmining.org` for sample logs, videos, slides, articles, and software.

- 30