Process Mining as a Modelling Tool: Beyond the Domain of Business Process Management

Antonio Cerone(✉)

IMT Institute for Advanced Studies, Lucca, Italy
antonio.cerone@imtlucca.it

Abstract. Process mining emerged in the field of business process management (BPM) as an innovative technique to exploit the large amount of data recorded by information systems in the form of event logs. It allows to discover not only relations and structure in data but also control flow, and produces a process model, which can then be visualised as a process map. In addition to discovery, process mining supports conformance analysis, a technique to compare an *a priori* model with the event logs to detect deviations and inconsistencies.

In this paper we go beyond the domain of BPM and illustrate how process mining and conformance analysis can be used in a number of contexts, in and across the areas of human-computer interaction and learning.

1 Introduction

Process mining is an emerging discipline based on model-driven approaches and data mining. It is a process management technique used to extract information from event logs consisting of activities (business activities, communication activities, collaboration activities, etc.) and then produce a graphical representation of the process control flow, detect relations between components/individuals involved in the process and infer data dependencies between process activities [11].

In order to be successfully processed, event logs have to meet a number of structural properties, that is, to contain adequately organised and clustered data. This level of structural organisation can be attained by applying text mining techniques, in particular semantic indexing, that is, by assigning a meaningful subject to the data. Recently, incremental fully automatic semantic mining algorithms have been developed within the semantic platform associated with the 2012 SQL Server: they produce weighted physical indexes, which can then be queried through the SQL interface [5]. Using the semantic platform associated with the SQL Server, queries can be defined based on a catalog in which keywords, key-phrases and conditional activities are categorised in terms of states of a process. The resultant output is a pre-processed log to be fed to a process mining tool, such as DISCO (Discover Your Process) [2], to produce a process model, presented as a Petri net-like visualisation (often called process map) and its associated PNML representation, possibly together with relevant statistical

D. Bianculli et al. (Eds.): SEFM 2015 Workshops, LNCS 9509, pp. 139–144, 2015.
DOI: 10.1007/978-3-662-49224-6_12

data. The capability of discovering statistical data about the analysed process makes process mining a useful tool in performance evaluation.

In the area of business process management (BPM), process mining has been used not only to discover a process model and represent it as a process map, but also to extend a pre-existing *a priori* model by enriching it with new aspects and perspectives illustrated by the discovered *a posteriori* process model, and to compare, by using a technique called *conformance analysis*, the *a priori* model with the event logs (and thus implicitly with the *a posteriori* process model).

Conformance analysis originated from Rozinat and van der Aalst's work in the area of BPM [7]. Conformance analysis, also called *conformance checking*, is the detection of deviations and inconsistencies between an *a priori* model, which is based on theoretical perspectives and/or data collected and analysed using social science research methods, and the traces generated by the event logs. In fact, conformance checking seems appropriate well beyond BPM. In particular, it can be applied to the analysis of social networks and peer-production systems, and the first attempts in this direction have being done in the areas of collaborative learning and Free/Libre Open Source Software (FLOSS) development [3,4].

In this paper we go beyond the domain of BPM and illustrate how process mining and conformance analysis can be used in a number of contexts, in and across the areas of human-computer interaction (HCI) and learning. Section 2 illustrates how to apply process mining to social networks in order to extract behavioral patterns that provide evidence of learning processes (Sect. 2.1) and skill acquisition (Sect. 2.2). Section 3 concerns real-time applications of process mining. Section 4 concludes the paper.

2 Modelling from Observed Behavioural Patterns

One important application of cognitive psychology to HCI is the observation of human behaviour, during interaction with interfaces, devices or within online communities, to extract behavioral patterns of users or control operators. In this section we present a case study on the extraction of learning processes from behavioral patterns of FLOSS contributors and propose how to extend our approach to modelling skill acquisition not only in FLOSS communities but, generally, in interacting with a specific device/interface/application.

2.1 Modelling Learning Processes: The FLOSS Case Study

Social Networks can be seen as collaborative environments in which interactions among peers support the building of knowledge both at individual and community level. Learning processes occur naturally within such environments and produce evidences of their existence in the contents of communications between community members and in the digital artifacts shared or produced by the community, such as web pages, documents, audio and video clips, software, etc.

FLOSS communities also present this learning potential. They are open participatory ecosystems in which actors not only create source code but also produce and organise a large variety of resources that include implicit and explicit knowledge, communication logs, documentation and tools. Collaboration in FLOSS projects is highly mediated by the usage of tools, such as versioning systems, mailing lists, reporting systems, etc. These tools serve as repositories which can be data mined to understand the identities of the individuals involved in a communication, the topics of their communication, the amount of information exchanged in each direction, as well as the amount of their contribution in terms of code commits, bug fixing, produced reports and documentation, sent emails and posted comments/messages. This large amount of data can be selectively collected and then analysed not only by using inferential statistics to identify activity patterns but also by using ontology engineering formalisms that support the extraction of semantic information [8,9].

In recent work [3], we identify three phases of the learning process occurring in a FLOSS environment, *initiation*, *progression* and *maturation*, and two categories of FLOSS contributors, *novice* and *expert*. For each phase and category of contributor, we make use of semantic search in SQL to retrieve data from posts and emails, in order to identify those activities, carried out by FLOSS members, that may contribute to the members' learning processes. The choice of the key-words and key-phrases that drive the semantic search is based on a number of studies that analyse FLOSS communities using social science means to identify questions and answers that normally occur during collaboration and communication in FLOSS environments [8]. States of the process are associated with lists of generic key-words/key-phrases while specific activities are associated with lists of more discriminative key-words/key-phrases. Examples of states are: *observation* and *contact establishment*, for the initiation phase; *revert*, *post* and *apply*, for the progression phase; *analyse*, *commit*, *develop*, *revert* and *review*, for the maturation phase. Example of activities are: *formulate question*, *identify expert* and *post message* as novice's activities of the *observation* state; *run source code* as expert's activity of the *apply* state; *submit code* and *submit bug report* as novice's activities of the *commit* state; *write source code* as novice's activity of the *develop* state. The resultant three catalogs, one for each phase of the learning process, are used to build organised event logs out of the unstructured data. Using DISCO process mining tool [2], a visual representation of a process model is extracted from the event log [3].

A number of pilot studies have analysed communications in FLOSS communities in terms of participants, quantity and sometimes topics by using questionnaires and surveys or written student reports describing the encountered risks as research instruments. These previous works were the basis for our definition of *a priori* models of the collaboration and learning processes occurring in FLOSS communities [1]. Using conformance analysis, these *a priori* models are compared with the event logs, thus detecting a number of deviations. Finally, such deviations are interpreted on the discovered *a posteriori* model in order to reconcile it with the corresponding original *a priori* model.

2.2 Modelling Skill Acquisition

The modelling and conformance analysis approach described in Sect. 2.1 refers to the learning process at a specific phase of the contributor's growth as a member of the FLOSS community, according to the two points of view of the novice looking for guidance and the expert providing support. However, transition between learning phases is not instantaneous but proceeds as a gradual evolution determined by the acquisition of new skills and their exploration in the social and productive contexts of the FLOSS community.

Understanding the aspects of skill acquisition, its individual variations and the social, technological and organisational factors that naturally encourage, constrain or hinder it is essential to design an appropriate learning model based on the exploitation of FLOSS projects. Given the diversity with which skill acquisition occurs for different individuals, and the consequent difficulties in collecting comprehensive data through social science research methods, it is hard to develop an *a priori* model of this important learning process.

In this context, process mining could be used as a primary modelling tool. As we have seen in Sect. 2.1, in order to produce catalogs for semantic search, appropriate key-words and key-phrases can be identified and associated with states and activities. However, states would now describe acquired skills, such as *coding*, *reviewing*, *testing* and *documenting*, while activities would still be the same as we identified in our previous work [3]. Furthermore, process mining could be used to associate quantitative information, such as frequency, number of repetition and approval rate, with activities. Quantitative information would be then integrated by functions that evaluate the level of skill acquisition. For example, the transition to the state *coding* can occur only if the ratio between the frequencies of *commit source code* and *write source code* is sufficiently high. Transition between states is triggered when the value of the function associated with the skill represented by the target state is above a given threshold.

Social networks are an important source of information about learning and other cognitive processes not only in the case of interaction within an online community, as in the case of FLOSS communities, but also in the context of the usage of a specific device/interface/application. Similarly to the diversity with which skill acquisition occurs for different individuals, users show large varieties in the modality of interacting with or using the device/interface/application/online resource. Moreover, the large number of features offered by these kinds of hardware and software artifacts gives users plenty of choices in developing strategies for using a specific artifact to achieve their goals. Furthermore, on the one hand, user's creativity results in modalities of use and exploitation that were not considered by the artifact designers and developers. For example, short message service (sms) was initially introduced in the 1980 s as an additional feature of mobile phones, but has nowadays become, for many users, the main or only purpose of using a mobile phone. On the other hand, artifact "pseudo-intelligence" tries to anticipate user's (unpredictable) behaviour, thus leading to unexpected errors. For example, a text processing program might continuously rearrange the order of the items in a toolbar depending on the frequency of their use, thus

confusing users and inducing errors. This complex situation cannot be captured by collecting data through social science research methods. As a consequence, also in modelling usage or tasks or task failures it is basically impossible to develop comprehensive *a priori* models. Thus, the application of process mining to online reviews and user community communications could extract important information about behavioral patterns underlying usage strategies, task performance and task failure.

Finally, online reviews of products contain a large amount of information about their standard and non-standard usage as well as their pitfalls and failures. Moreover, new hardware and software products give rise to new online communities of users who exchange opinions on the product, report their usage experiences, post requests for help, reply by providing advices and, most important, learn from each other, thus improving their skills and evolving from being novices to being experts. We claim that both online reviews and user community communications can be mined to extract information about user's skill acquisition in using the product. Therefore, a process mining based approach could be used also in this context to define a skill acquisition model to be used for evaluating the quality of the product and improving new releases in terms of learnability and usability.

3 Towards Real-Time Process Mining

In various domains a large amount of data is collected using geographical information system (GIS) in association with a wireless sensor network. This is the case, for example, in: *ethology*, by equipping individuals of animal species with tracking devices in order to monitor animal behaviour and migration routes; *transportation*, by exploiting GPS devices on cellphones or cars; *ecology*, where wireless sensor networks are used to collect real-time information about environmental conditions. In some cases, such as in ethology, data processing occurs normally only after data collection has been completed, whereas, in other cases, data must be processed in real time in order to decide corrective or emergency action to be carried out promptly. For example, in transportation, traffic may be redirected in real time to avoid congestion, while, in ecology, real-time data flow allows researchers to react rapidly to events, thus extending the laboratory to the field [6].

We envisage the use of real-time process mining to produce visual presentations of bottlenecks and alternative routes, from traffic event logs, for traffic management, as well as informed, visual presentations of the real-time situations, from sensor network and social network logs, for emergency management.

4 Conclusion

In this paper we have extensively discussed the possible use of process mining as an effective modelling and validation tool to support the design and validation of a number of frameworks and systems spanning across various application

domains. We have envisaged that process mining could be effectively applied to a variety of domains other than BPM: learning, HCI, cognitive modelling, traffic management and emergency management. As a support to our claims, we have shown the successful use of process mining and conformance analysis in the area of learning, by referring to our previous work on process mining FLOSS repositories, which aimed to validate an *a priori* model of the learning processes naturally occurring within FLOSS communities [3,4]. In our approach, process mining is applied to FLOSS communities to discover dynamic processes (learning processes, that is, processes that produce learning). This is different from van der Aalst and Song's approach to discover and analyse social networks from event logs [10]. In fact, their approach consists in extracting information about the activity performers described by the event logs, whereas ours consists in extracting information about control flow and building statistics about the occurrence of activities.

References

1. Cerone, A.: Learning and activity patterns in OSS communities and their impact on software quality. In: Proceedings of OpenCert 2011, ECEASST, vol. 48 (2012)
2. Günther, C., Rozinat, A.: DISCO: discover your process. In: Proceedings of the Demonstration Track of BPM 2012, CEUR Workshop Proceedings, vol. 940, pp. 40–44. CEUR-WS.org (2012)
3. Mukala, P., Cerone, A., Turini, F.: Mining learning processes from FLOSS mailing archives. In: Janssen, M., Mäntymäki, M., Hidders, J., Klievink, B., Lamersdorf, W., van Loenen, B., Zuiderwijk, A. (eds.) I3E 2015. LNCS, vol. 9373, pp. 287–298. Springer, Heidelberg (2015)
4. Mukala, P., Cerone, A., Turini, F.: Process mining event logs from FLOSS data: state of the art and perspectives. In: Canal, C., Idani, A. (eds.) SEFM 2014 Workshops. LNCS, vol. 8938, pp. 182–198. Springer, Heidelberg (2015)
5. Mukerjee, K., Porter, T., Gherman, S.: Linear scale semantic mining algorithms in Microsoft SQL server's semantics platform. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 213–221. ACM (2011)
6. Porter, J., et al.: Wireless sensor networks for ecology. BioScience **55**(7), 561–572 (2005)
7. Rozinat, A., van der Aalst, W.M.P.: Conformance checking of processes based on monitoring real behavior. Inf. Syst. **33**(1), 64–95 (2008)
8. Scacchi, W., Feller, J., Fitzgerald, B., Hissam, S.A., Lakhani, K.: Understanding free/open source software development processes. Softw. Process Improv. Pract. **11**(2), 95–105 (2006)
9. Sowe, S.K., Cerone, A.: Integrating data from multiple repositories to analyze patterns of contribution in FOSS projects. In: Proceedings of OpenCert 2010, ECEASST, vol. 33 (2010)
10. van der Aalst, W.M.P., Song, M.S.: Mining social networks: uncovering interaction patterns in business processes. In: Desel, J., Pernici, B., Weske, M. (eds.) BPM 2004. LNCS, vol. 3080, pp. 244–260. Springer, Heidelberg (2004)
11. van der Aalst, W.M.P., Stahl, C.: Modeling Business Processes: A Petri Net-Oriented Approach. The MIT Press, Cambridge (2011)