Discovering Business Rules in Knowledge-Intensive Processes Through Decision Mining: An Experimental Study

Júlio Campos(✉), Pedro Richetti, Fernanda Araújo Baião, and Flávia Maria Santoro

Federal University of the State of Rio de Janeiro (UNIRIO), Rio de Janeiro, Brazil
{julio.campos,pedro.richetti,fernanda.baiao,
flavia.santoro}@uniriotec.br

Abstract. Decision mining allows discovering rules that constraint the paths that the instances of a business process may follow during its execution. In Knowledge-intensive Processes (KiP), the discovery of such rules is a great challenge because they lack structure. In this context, this experimental study applies a decision mining technique in an event log of a real company that provides ICT infrastructure services. The log comprises structured data (ticket events) and non-structured data (messages exchanged among team members). The goal was to discover tacit decisions that could be potentially declared as business rules for the company. In addition to mining the decision points, we validated the discovered rule with the company w.r.t. their meaning.

Keywords: Decision mining · Business rules · Knowledge-intensive Processes

1 Introduction

Process mining techniques aim to discover business process models from events recorded in data logs. Most algorithms used for this purpose generate models that show the flow of activities, but do not identify or detail how decisions are made along it. Nonetheless, recent advances in techniques produce models that are more adjusted to the event log, since they contemplate decisions that regulate the activities flow [2]. Decision mining allows discovering decision points to explain how different paths are taken during a process execution [12]. However, the discovery of decisions is not trivial, especially for the so-called Knowledge-intensive Processes (KiP), which are weakly structured and are not driven by pre-established rules. KiPs are mostly carried out based on knowledge and experience of actors involved in its execution [11].

Despite these characteristics, some of the discovered decisions within a KiP are candidates to become business rules that might serve as strategic knowledge for the organization and support future decisions to be made. The literature shows that few works address the relationship between the logic of decisions made and the process. So, the problem investigated here is whether a decision mining technique allows to discover business rules within the flow of activities of a KiP. The main goal of this paper is to

E. Teniente and M. Weidlich (Eds.): BPM 2017 Workshops, LNBIP 308, pp. 556–567, 2018.
https://doi.org/10.1007/978-3-319-74030-0_44

discuss the results from an experimental study made with a log of a com-pany that provides IT services to several clients.

2 Background Knowledge

2.1 Knowledge-Intensive Processes

According to [11], business processes can be classified into three types: structured, semi-structured and unstructured. Structured processes are completely predefined, i.e., there are fixed rules that cannot be changed to perform each activity. Semi-structured processes contain structured and unstructured parts; not all the activities have pre-defined rules regarding the next steps in the flow. Unstructured processes are completely unpredictable and pose no pre-defined order for the execution of the activities, being commonly called knowledge-intensive processes (KiP). KiPs are not suitable for automation and are conferred to a great degree of freedom to achieve their goals. For [11], value is created within KIP by fulfilling the participants' knowledge requirements. Thus, the decisions that are made during the accomplishment of the tasks are directly influenced by the knowledge of who performs them.

Di Ciccio et al. [4] defined KiP as processes "whose conduct and execution are heavily dependent on knowledge workers performing various interconnected knowledge intensive decision-making tasks." The authors also elicited key typical characteristics of KiPs: knowledge-driven; collaboration-oriented; unpredictable; emergent; goal-oriented; event-driven; constraint-and rule-driven; and non-repeatable. Examples of KiPs include customer support, design of new products/services, marketing, IT governance or strategic planning. Besides, they concluded that the way organizations deal with this kind of processes has changed over time, e.g. the customer support processes in several organizations have evolved from highly structured to knowledge-intensive, and personalized, flexible individual cases.

França [5] proposed an ontology named KiPO (Knowledge-Intensive Process Ontology) aimed at comprising the key concepts and relationships that are relevant for understanding, describing and managing a knowledge-intensive process. KiPO provides a common, domain-independent understanding of KiPs and, as such, it may be used as a metamodel for structuring KiP concepts. KiPO is composed of 5 sub-ontologies, which reflect the main KiP perspectives. The Business Process Ontology (BPO) comprises the traditional elements of business process modeling (such as activ-ities, event flows, input/output data objects). The Collaboration Ontology (CO) de-picts concepts to explain how knowledge artifacts are exchanged among process participants, and how the collaboration takes place. The Decision Ontology (DO) aims at describing the rationale of the decisions made by the process agents (i.e., the "why" and "how" decisions were made by the people involved in the process) thus allowing the tracking of what motivated a decision and the outcomes from it. The Business Rules Ontology (BRO) provides the means to describe some parts of the KiP from a declarative perspective, since describing the rules that govern a KiP is especially useful for describing the parts of the process which are very flexible and not subject to predefined event flows. Finally, the Knowledge Intensive Process Core Ontology (KiPCO) comprises the core concepts of a KiP (mainly

Agents, Knowledge-intensive activities and contextual elements involved in their execution). KiPO argues that in a KiP the flow of activities (especially decision-making) is deeply influenced by tacit elements from its stakeholders. In this paper, we explore the decision making associated to business rules perspectives of KiPO.

2.2 Decision Mining

Business processes are established and structured upon business rules. A business rule "is a statement that defines or restricts some aspect of an organization's business" [6]. Process mining discovers how business processes are structured through two techniques: process discovery and conformance checking. The first one builds a process model that reflects the behavior observed in event logs. The second one tries to detect deviations in the existing model [12]. According to [12], in the early days of process mining, most algorithms supported only the control flow perspective. Slight attention has been given to values of data attributes which can affect the routing of an instance during a process execution. Decision mining research was advanced in this context. The term was first used by researchers who described so-called decision points in models. Decision points are "parts of the model in which the process is divided into alternative branches" [12]. The researchers created a decision tree algorithm, provided in ProM[1] framework, which retrieves test results at a split point to analyze how choices are made in a business process model.

De Leoni and Van der Aalst [2] stated that the technique proposed by [12] was not able to discover the conditions associated with split exclusive or and loops. Another limitation was that the event log required to be in full compliance with the flow control modeled, i.e., the order in which the activities are executed would never be different from the order of the idealized model. The authors proposed a new approach in which an alignment between an event log and a process model is performed first, and then a decision tree algorithm is applied. This solution was implemented in ProM framework through the Multi-Perspective Process Explorer (MPE) plugin [8]. The "Discover Data Perspective" mode of MPE allows discovering guards associated with a transition (process activity). A guard can be any Boolean expression that uses logical operators such as conjunction ($\wedge$), disjunction ($\vee$) and negation ($\neg$). The user selects one among five decision tree algorithms to discover the guards as well as the attributes to be considered. The user has also to configure two parameters: minimum of elements associated with the leaves of the decision tree (min cases) and the mini-mum adjustment of the flow of control for each instance to be considered. The min instances parameter is important because it influences whether the guards are over-adjusted, or poor-adjusted [8]. Recently, Mannhardt et al. [9] developed a new technique that allows discovering overlapping decision rules, since the algorithm proposed by [12] was able to mine only mutually exclusive splits (XOR). The solution was added to the Multi-Perspective Process Explorer (MPE) plugin.

De Smedt et al. [3] argue that most of the literature on decision mining focuses on increasingly refined techniques on the retrieval of decision information in business

[1] http://www.promtools.org/

process models. However, for the authors, only a few works are dynamically capable of discovering the stages of the decision-making process. The authors created a framework with four perspectives to evaluate decision mining techniques. They concluded that few studies cover this perspective that explains how the decisions discovered are related to the own decision-making process.

2.3 Decision Model and Notation (DMN)

Due to the emergence of decision mining, several organizations started to address a need for a standardized notation to represent decisions in business process models. In 2015, The Object Management Group (OMG) released the first version of the DMN (Decision Model and Notation). DMN goal is to ensure that a decision model is inter-changeable between entities through an XML representation [10]. Through the DMN, "decisions can be modeled so that a decision-making in an organization can be easily represented in diagrams, accurately by business analysts and (optionally) automated".

According to the [10], decision-making is addressed by two different perspectives by existing model standards: First in BPMN process models, which represent tasks in which decisions occur. The second perspective refers to a decision logic, i.e. a specific logic to make particular decisions, for example, in business rules, decision tables or in executable analytical models. For several authors, a decision making has an internal structure that is not conveniently captured by either of the two perspectives cited. Thus, the purpose of the DMN is to provide a third perspective, the Decision Requirements Diagram (DRD), which forms the bridge between business models and decision logic models. OMG (2015) suggests three possible uses of DMN in order to understand and define how decisions are made in a company or organization: (i) modeling human decision making, (ii) modeling requirements for automated decision making, (iii) implementing automated decision making.

3 Experimental Study

We accomplished an experimental study comprising using an event log of an ICT company. The company has around a hundred contracts with different firms to pro-vide ICT infrastructure services. One of their business processes is the resolution of ICT incidents related to client's assets, such as e-mail server outages and network connection problems. An incident is an unexpected, unplanned episode, which if not solved correctly can cause loss, damage or even some kind of accident. The activities to address the incidents involve the application of technical skills, troubleshooting abilities, collaboration, and information exchange among technicians and between the team of technicians and the client. There is no strictly structured process to be followed, since most of the problems are situational and several ad-hoc decisions may be taken. These points characterize knowledge-intensive aspects in such a way that traditional control flow oriented business process would be not adequate to manage the scenario. The goal of this experimental study was to evaluate decision mining techniques to discover decision points in a KiP. Due to the characteristics of KiP, we show the relevance of also considering textual content in the analysis. Thus, we divided the study into 2 steps.

3.1 Experimental Setting: The Log

The log contains records of 6.337 instances of the process and 246.283 events, distributed by 32 activities. We filtered a sample of the log with all tickets opened in the 2nd semester of 2015. This sample included structured data about the tickets logged by the process-aware CRM system (explored in the first step), together with all e-mail messages exchanged between employees and customers for discussions about the problem to be solved (explored in the second step).

3.2 First Step: Discovering Rules Within the Structured Log

Method. The first step was shaped to mine the log attempting to find decision points. Although we are dealing with a KiP, since our focus in this paper is on decisions, we did not choose techniques commonly used to discover unstructured models (such as declarative ones). The MPE plugin was chosen to support this task because the technique fits into the third perspective of the framework proposed by [3]. The content of the conversations was not included in this step because it consists unstructured data and, as so, not eligible for analysis with the technique. Before performing the process mining, we filtered the event log. Filtering is an important preprocessing procedure because it allows for the discovery of error-free process models. It is also useful for selecting a subset of data of greater relevance or interest for an analysis. In our case, the purpose was to select tickets that generated e-mail exchanges between employees and customers, as the examples in Fig. 1. Thus, all instances involving null fields in the column "article_id" were filtered out. After filtering, the number of instances remained in 6.337. The number of events dropped to 63.424 and activities to 13.

	A	B	C	D	E	F	G	H	I
1	ticket_id	eventName	article_id	priority_id	ticketState	serviceType	solution_time	SLAMissed	eventDateTime
2	160431	NewTicket		3	new	Redes::Diagnósticos	720	N	2015-06-01 07:20:09
3	160431	ServiceUpdate		3	new	Redes::Diagnósticos	720	N	2015-06-01 07:20:09
4	160431	SLAUpdate		3	new	Redes::Diagnósticos	720	N	2015-06-01 07:20:09
5	160431	CustomerUpdate		3	new	Redes::Diagnósticos	720	N	2015-06-01 07:20:09
6	160431	EmailCustomer	605130	3	new	Redes::Diagnósticos	720	N	2015-06-01 07:20:09
7	160431	SendAutoReply	605131	3	new	Redes::Diagnósticos	720	N	2015-06-01 07:20:10
8	160431	SendAgentNotification		3	new	Redes::Diagnósticos	720	N	2015-06-01 07:20:10
9	160431	SendAgentNotification		3	new	Redes::Diagnósticos	720	N	2015-06-01 07:20:11
10	160431	SendAgentNotification		3	new	Redes::Diagnósticos	720	N	2015-06-01 07:20:11
11	160431	Lock		3	new	Redes::Diagnósticos	720	N	2015-06-01 09:05:13
12	160431	Misc		3	new	Redes::Diagnósticos	720	N	2015-06-01 09:05:13
13	160431	OwnerUpdate		3	new	Redes::Diagnósticos	720	N	2015-06-01 09:05:13
14	160431	TypeUpdate		3	new	Redes::Diagnósticos	720	N	2015-06-01 09:08:10
15	160431	ServiceUpdate		3	new	Redes::Diagnósticos	720	N	2015-06-01 09:08:10
16	160431	SLAUpdate		3	new	Redes::Diagnósticos	720	N	2015-06-01 09:08:10
17	160431	AddNote	605208	3	new	Redes::Diagnósticos	720	N	2015-06-01 09:08:10
18	160431	TicketDynamicFieldUpdate		3	new	Redes::Diagnósticos	720	N	2015-06-01 09:08:10
19	160431	TicketDynamicFieldUpdate		3	new	Redes::Diagnósticos	720	N	2015-06-01 09:08:10
20	160431	SendAnswer	605210	3	new	Redes::Diagnósticos	720	N	2015-06-01 09:08:32

Fig. 1. Sample of the selected events by the filter

Among the most executed activities in the process, "TimeAccount" standed out as the top most frequent, and "AddNote" the second. The "TimeAccount" activity is a manual record of the time an employee spent to interact with a customer. The "Add-Note" activity is an internal note used to exchange information between employees. this activity reinforces classifying this process as knowledge-intensive, since it represented exchanges of knowledge among employees to guide decision-making about problems and incidents that occur during the provision of company services.

After filtering, the process mined from the event log was represented as a Petri net, using the "Petri Net Mine with Visual Inductive Mining" plugin. In this plugin, the only adjustment was setting the value of the "Noise threshold" parameter from 0.20 to 0.0, in order to guarantee a perfect adjustment of the log. The Petri net model and filtered event log served as inputs for a "Multi-Perspective Process Explorer" plug-in. After the plugin was executed, it generated the base model in Fig. 2.

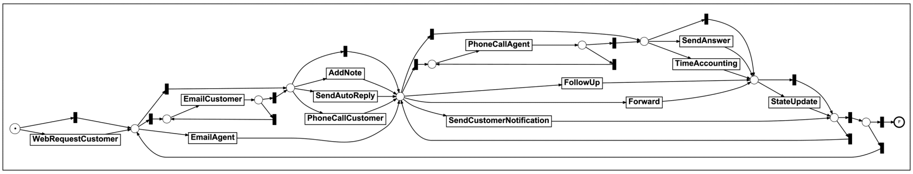

Fig. 2. Model generated by MPE

When selecting the "Discover Data Perspective" mode, the plugin performed a computational alignment between the event log and the Petri net model. To perform the alignment, the simple configuration mode, which uses standard parameters, has been chosen. A new base model was generated with an average adjustment rate of 100%, meaning no violations, missed or lost events. All 63.424 event log events have been correctly aligned to the Petri net model.

Results. Three scenarios were carried out to discover the rules on the Petri net model. In all of them, the value of the "min instances" parameter was modified, keeping the value of the "min fitness" parameter equal to 1. In the first scenario, the lowest possible value was selected for the "min instances" parameter (0.001), which allowed the discovery of very large and complex rules related to some activities. In the second scenario, the highest possible value was selected for the "min instances" parameter (0.5), and no rules were found in the model. In the third scenario, the value of the "min instances" parameter was changed between 0.001 and 0.5. In this scenario, guards were found in the "sink 6" position for the "AddNote" activity. AddNote is considered a knowledge intensive activity because it is when people interact (through messages) to discuss the problems. The results are shown in Table 1.

In Table 1, the accuracy of the rules found is measured by the parameter "Guard F-Measure". The higher the value of this parameter, the greater the accuracy of the uncovered guard. All rules discovered presented high accuracy. The difference among them lies in the degree of complexity. The algorithm could find simple rules for the "min instances" values below 0.2, but for the value 0.3 a large and complex rule related to the status of the ticket was found, as shown in Fig. 3.

Table 1. Rules related to the “AddNote” activity

AddNote	Decision tree (default false)		
Min instances	0.11	0.2	0.3
Min fitness	1.0	1.0	1.0
Guard F-measure	85.7%	81.7%	85.8%
Guard	article_id > 605709.0	article_id > 622246.0	((((((((ticketState == “Agendamento” ǁ ticketState == “closed successful”) ǁ ticketState == “closed with workaround”) ǁ ticketState == “merge d”) ǁ ticketState == “new”) ǁ ticketState == “open”) ǁ ticketState == “pendi ng auto close +”) ǁ ticketState == “pendi ng auto close-”) ǁ ticketState == “pendi ng reminder”)
Correct events	12570	12570	12570
Wrong events (data)	0	0	0
Missing events	0	0	0

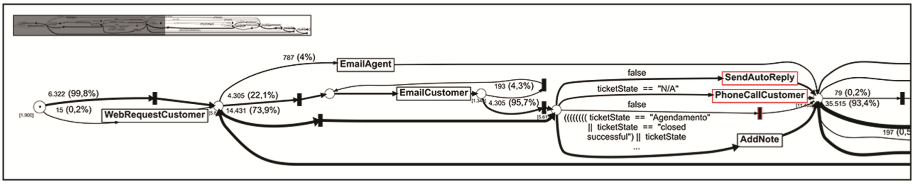

Fig. 3. Rule discovered by decision tree algorithm related to “AddNote” at “sink 6” (Color figure online)

In Fig. 3, we notice that some activities are highlighted in color. This is an indication that the discovery of rules for such activities is more difficult according to the value selected for the “min instances” parameter. Figure 4 shows the same model without the rules. “AddNote” is a transition of the “sink 6” location, as well as the “SendAutoReplay”, “PhoneCallCustomer” transitions, and also an invisible transition (black rectangle). According to the model, 18.736 events have passed from the local “sink 6”. Of these, 12.570 (67.1%) performed the “AddNote” activity. In the model, thicker arcs indicate the main flow followed by most instances of the process.

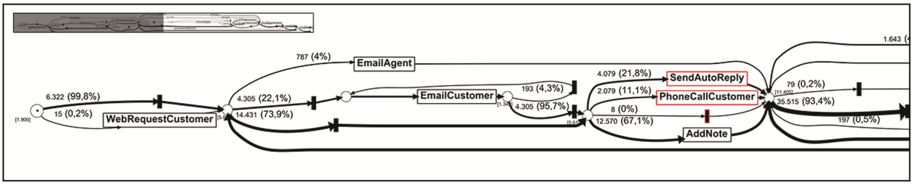

Fig. 4. Control flow with the percentage of events that executed each activity (Color figure online)

It is also possible to evaluate the accuracy of the base model, generated after the computational alignment. Table 2 shows a comparison of the accuracy of the base model with “sink 6”, from which the rules were found. The accuracy of the base mod-el is low (37.6%). The local precision of “sink 6” is somewhat lower (27.7%). Although the base model has 100% fitness, the accuracy is low, i.e., it allows a behavior not observed in the event log.

Table 2. Comparison between precision of the model and “sink 6” place

Sink 6		General	
Local place precision	27.7%	Avg precision	37.6%
# Observed locally	20.747	# Observed	243.736
# Possible locally	74.944	# Possible	647.571
Global place precision	27.7%	Avg fitness	100%

The rules discovered in all three scenarios evaluated with the structured log were then validated with the company staff. Two managers responsible for keeping up with the tasks executed by the technical team were interviewed. The rules were presented to them and they were asked to analyze them and answer about the meaning and appropriateness. The goal was to understand if the rules could be considered correct and as well if they are really applied within this process. Both agreed with the rules, telling that they make sense for them, but in fact, they are not surprising. We concluded that the method applied may possibly discover correct rules; however, this is not enough in this case to provide insights to the company staff. So, we proceeded to explore the unstructured information available in the log.

3.3 Second Step: Discovering Rules Within the Unstructured Log

Method. The fourth scenario of the experiment explored the event log through text mining techniques. The goal was to seek decision rules derived from the knowledge of the employees that guided the decision making during the execution of the activities of the process within the records of conversations exchanged between the employees and the clients. These conversations were extracted from the original event log and copied to a new text file. Using the free software R, the text file was loaded and scanned with the grep command through a regular expression to filter tickets in which the words “incident”, “rule”, “procedure” and the radical “amos” and “soluc” (in Portuguese) were mentioned in order to find records of incidents that were solved.

Results. We found 421 results related to the word incident. Among the results obtained, four related to the solution of a problem were selected for analysis (Table 3).

Table 3. Report of an incident discussion

Ticket	Article	Activity	Message
165027	623276	*EmailAgent*	"We arrived at the place where there was no internet access. We did an analysis of the environment to detect the source of the problem. We identified that the up-link cable did not allow connection to the internet. We used another preexisting connection in the store, changed the up-link and solved the incident."
218683	690455	*SendAnswer*	"(…) We inform you that your request regarding 'Printer has stopped working' was completed by the FOT team who took great pleasure in helping you. We performed environment analysis, and we detected divergence of configurations. The stations were pointing to an address that differed from the address set on the printer. We made the correction in the printer, entering the address to which the stations pointed. This procedure solved the problem reported by our client, who received us on the spot, validated the conclusion of our call with success."
234964	745175	*TimeAccounting*	"(…) According to the phone contact, a reboot procedure was performed on the server and it did not load the system correctly. After this episode, the server was shut down and reconnected without the physically connected off-board network cards. Since it was not successful, we are migrating the call to head-on service, which will be arranged by scheduling with the service center. We are aware of the criticality of the incident and are placing the call with a high urgency level."
166513	629237	*SendAnswer*	"(…) We have already corrected the Firewall rule that was identified by the support of yesterday. Rules loading tests were all performed successfully. The call will be ended."

The first incident, for example, records reports the decision made by company technician to solve a problem of internet access interruption. The need to analyze the environment indicates that the activity is knowledge-intensive, because it demands tacit knowledge of the employees. The analysis may have been based on a business rule defined by the company or derived from the employees' tacit knowledge. Business rule inferred by the conversation: "The first alternative to be tested in a case of internet interruption must be the pre-existent connection". The source of the problem was detected and a decision was made based on a workaround, that is, use another pre-existing connection in the client store, which solved the incident. Figure 5 shows a decision model of the treatment of that incident using the DMN notation.

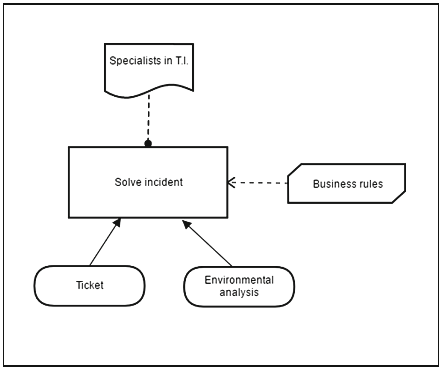

Fig. 5. Decision Requirement Diagram of the 1st and 2nd incident (Table 3).

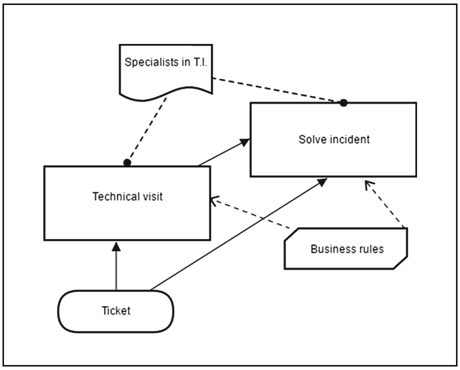

Fig. 6. Decision Requirement Diagram of the 3rd incident (Table 3).

In Fig. 5, the input data from the Decision Requirement Diagram (DRD) is the conversation about the ticket and the analysis of the environment. This data provides the information needed to guide the decision, a Business Knowledge Model, which may relate to a business rule predefined by the company or represent a situation based on the employee tacit knowledge. The decision is made by a Knowledge Source, which is an authority defined to make the decisions. In the case under analysis, the company employees detain technical and tacit knowledge to solve complex problems.

The record of the second incident describes a situation similar to the first one. A staff performed an on-site environment analysis to solve a printer operating problem. The result of the analysis allowed them to identify the source of the problem to apply an appropriate correction procedure. The problem solution was validated by the customer who contacted the team. For this incident, the same DRD of Fig. 5 applies.

The third incident log describes an episode where a remote procedure was performed to try to solve a troubleshoot on a server. After two unsuccessful attempts, the employee requests a head-on service with a high level of urgency. In Fig. 6, this incident is modeled in a DRD. We notice that the decision about how to solve an incident depends on another decision, i.e. assessing the need for a technical visit. Ac-cording to the report, the decision to request face-to-face attendance was made due to the failure of the procedure performed, in addition to the high degree of criticality. The procedure is a clear evidence that the employee followed a business rule established by the company to try to solve the problem. A rule that can be inferred from the decision made by the employee: "A technical visit must be requested for highly critical incidents". The last incident log is a response to a client to inform about fixing a problem detected in a Firewall rule. The problem was solved after testing. This scenario describes a situation that is very frequent in the company: the solution of problems depends on the adjustment of rules of service configuration, which demands a high level of tacit knowledge of the employees.

Once more, we conducted a validation with the company staff. The same two managers were invited to analyze those rules based on the same criteria: meaning and appropriateness. This time their perception about the results were very positive. They recognized the rules as tacit knowledge of the team; therefore, they agreed that it would be very relevant to make them explicit, and also the possibility to disseminate to the other technicians, and finally institutionalize them.

4 Discussion

The results obtained in the experiment point to limitations of business rules discovery in knowledge-intensive processes. The decision mining technique discovered few rules due to the low number of event attributes in the company log. In addition, the decision rules discovered by the decision tree algorithm only informed the necessary conditions for the execution of certain activities. The decision mining algorithm used could discover three distinct rules for the "AddNote" activity (two simple and one complex), with good accuracy. However, the model from which the decision rules were discovered presented low precision, which significantly affected the quality of the discovered model. In addition, little understanding, or knowledge relevant to decision making was obtained from the rules. To find out how decisions were made during the execution of a knowledge-intensive activity, we analyzed the conversations exchanged between the employees and clients of the company. In the last scenario, it was clear how decision-making is contextualized and can vary from instance to instance. In this scenario, the discovery of general rules that guides decision making and that could be established as business rules becomes a major research challenge.

Some rules could be identified from the incident records. These rules were based on procedures and guided the technicians' decision-making process. The DRDs of the incidents illustrate how to explicit that the decision to solve an incident depends on another decision: to evaluate the necessity of a technical visit to the place. According to the records, the technical visit is requested when the level of criticality of the incident is high. In this situation, an environmental analysis is performed and it requires a more intensive level of employee knowledge to identify the cause of the problem. It seems to be aligned with the conclusions of [7], for whom in a Kip scenario, a decision model is required, and it is more relevant than the BPMN model.

A few limitations of the experiment are observed. In the first step, we chose one technique to discover rules, but we did not compare with other approaches, as for example [1]. One clear threat to validity was the validation, which was made qualitatively through interviews with only two participants of the process. Although they are the most experienced participants in this process, collect the perception of other members of the staff could improve the conclusions.

5 Conclusions

Mining decisions in knowledge-intensive processes is not an easy task, as their activities are poorly structured. Moreover, each instance of a knowledge-intensive process is executed in a different way, which further complicates the automatic extraction of decision rules associated with the execution of its activities. In the experiments performed with the support of a decision mining technique, few rules were found for the activities of the process. The model in which the rules were discovered showed low precision, which significantly affected their quality. The scenario with text mining showed the existence of decision-making records cannot yet be easily incorporated by current process mining techniques in order to enrich the mining decision models. Future work is enriching log events with complementary Natural Language Processing and text mining techniques, besides applying other approaches such as [1] to compare the results.

References

1. Bazhenova, E., Buelow, S., Weske, M.: Discovering decision models from event logs. In: Abramowicz, W., Alt, R., Franczyk, B. (eds.) BIS 2016. LNBIP, vol. 255, pp. 237–251. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39426-8_19
2. De Leoni, M., van der Aalst W.M.P.: Data-aware process mining: discovering decisions in processes using alignments. In: Proceedings of the 28th Annual ACM Symposium on Applied Computing, pp. 1454–1461. ACM (2013)
3. De Smedt, J., vanden Broucke, S.K.L.M., Obregon, J., Kim, A., Jung, J.-Y., Vanthienen, J.: Decision mining in a broader context: an overview of the current landscape and future directions. In: Dumas, M., Fantinato, M. (eds.) BPM 2016. LNBIP, vol. 281, pp. 197–207. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-58457-7_15
4. Di Ciccio, C., Marrella, A., Russo, A.: Knowledge-intensive processes: characteristics, requirements and analysis of contemporary approaches. J. Data Semant. **4**(1), 29–57 (2015)
5. França, J.: Uma ontologia para definição de processos intensivos em conhecimento. Tese de Doutorado. M. Sc. dissertation, Departamento de Informática Aplicada (DIA), Universidade Federal do Estado do Rio de Janeiro (UNIRIO), Rio de Janeiro (2012)
6. Hay, D., et al.: Defining business rules-what are they really. Final report (2000)
7. Janssens, L., Bazhenova, E., De Smedt, J., Vanthienen, J., Denecker, M.: Consistent integration of decision (DMN) and process (BPMN) models. In: Proceedings of the CAiSE 2016 Forum at the 28th International Conference on Advanced Information Systems Engineering, Ljubljana, Slovenia (2016)
8. Manhardt, F., De Leoni, M., Reijers, H.A.: The multi-perspective process explorer. In: BPM (Demos), pp. 130–134 (2015)
9. Mannhardt, F., de Leoni, M., Reijers, H.A., van der Aalst, W.M.P.: Decision mining revisited - discovering overlapping rules. In: Nurcan, S., Soffer, P., Bajec, M., Eder, J. (eds.) CAiSE 2016. LNCS, vol. 9694, pp. 377–392. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39696-5_23
10. Object Management Group (OMG): Decision Model And Notation (DMN) version 1.1 (2016). Accessed 01 Jun 2016
11. Richter-Von Hagen, C., Ratz, D., Povalej, R.: Towards self-organizing knowledge intensive processes. J. Univ. Knowl. Manag. **2**, 148–169 (2005)
12. Rozinat, A., van der Aalst, W.M.P.: Decision mining in ProM. In: Dustdar, S., Fiadeiro, J.L., Sheth, A.P. (eds.) BPM 2006. LNCS, vol. 4102, pp. 420–425. Springer, Heidelberg (2006). https://doi.org/10.1007/11841760_33