FEDERAL UNIVERSITY OF THE STATE OF RIO DE JANEIRO
CENTER OF EXACT SCIENCES AND TECHNOLOGY
SCHOOL OF APPLIED INFORMATICS

Use of Decision Mining Technique to Discover Business Rules in Knowledge-intensive Processes

Júlio Campos

Supervisors
Flávia Maria Santoro
Fernanda Araujo Baião

RIO DE JANEIRO, RJ – BRAZIL
JULY OF 2018

Computerized cataloging by the author

C198 Campos, Júlio

Use of Decision Mining Technique to Discover Business Rules in Knowledge-Intensive Processes / Júlio Campos. -- Rio de Janeiro, 2018.

40

Orientadora: Flávia Maria Santoro.

Coorientadora: Fernanda Araujo Baião.

Trabalho de Conclusão de Curso (Graduação) - Universidade Federal do Estado do Rio de Janeiro, Graduação em Sistemas de Informação, 2018.

1. Decision Mining. 2. Business Rules. 3. Knowledge-intensive Processes. I. Santoro, Flávia Maria, orient. II. Baião, Fernanda Araujo, coorient. III. Título.

Use of Decision Mining Technique to Discover Business Rules
in Knowledge-intensive Processes

Júlio Campos

Undergraduate project presented at the School of Applied Informatics at Federal University of the State of Rio de Janeiro (UNIRIO) in order to obtain the title of bachelor's in information systems.

Approved by:

____

Flávia Maria Santoro, D. Sc. (UNIRIO)

____

Fernanda Araujo Baião, D. Sc. (UNIRIO)

____

Kate Cerqueira Revoredo, D. Sc. (UNIRIO)

RIO DE JANEIRO, RJ – BRAZIL
JULY OF 2018

Acknowledgments

First of all, I thank God for giving me strength, wisdom and protection during my stay in Rio de Janeiro.

To my parents for the education, care and unconditional support so that I could complete another stage in my life.

To the direction, coordination and body of professors of the course of Bachelor of Information Systems of the Unirio for the knowledge and values received that contributed greatly to my training as a systems analyst.

To my advisers Flávia Maria Santoro and Fernanda Araujo Baião for all the knowledge, accompaniment and support received in the last three years as a scientific initiation fellow. I have no words to thank!

To my co-advisers in the scientific initiation Pedro Richetti and João Carlos de Almeida Rodrigues Gonçalves who helped me and supported me a lot during the scientific initiation.

To Dr. Felix Mannhardt for the kindness and availability in solving some doubts that I had in relation to his technique of decision mining that he developed and used in this work.

To Kate Revoredo for teaching me the first concepts about data mining and her willingness to participate in the evaluation of this work.

To my graduation colleagues for the friendship and coexistence during my academic education.

ABSTRACT

Decision mining allows discovering rules that constraint and explain 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.

RESUMO

A mineração de decisão permite descobrir regras que restringem e explicam os caminhos que as instâncias de um processo de negócio podem seguir durante a sua execução. Nos processos intensivos em conhecimento (KiP), a descoberta de tais regras é um grande desafio, pois estes processos são geralmente pouco estruturados. Neste contexto, o objetivo deste trabalho de conclusão de curso é apresentar um estudo experimental que aplica uma técnica de mineração de decisão em um log de eventos de uma empresa que fornece serviços de infraestrutura de TIC. O log compreende dados estruturados (eventos de ticket) e dados não estruturados (mensagens trocadas entre funcionários). O objetivo foi descobrir decisões tácitas que poderiam ser potencialmente declaradas como regras de negócios para a empresa. Além de minerar os pontos de decisão, as regras descobertas foram validadas com a empresa em relação ao seu significado.

Palavras-chave: Mineração de Decisão, Regras de Negócios, Processos Intensivos em Conhecimento.

Index

Table Index

Image Index

1 **Introduction**

1.1 Motivation

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 (DE LEONI ; VAN DER AALST, 2013). Decision mining allows discovering decision points to explain how different paths are taken during a process execution (ROZINAT; VAN DER AALST, 2006). However, the discovery of decisions is not trivial, especially for the so-called Knowledge-intensive Processes (KiP), which generally 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 (ROZINAT; VAN DER AALST, 2006).

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.

1.2 Objective

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 undergraduate dissertation is to discuss the results from an experimental study made with a log of a company that provides Information Technology (IT) services to several clients.

1.3 Structure

The present work is structured in chapters and, in addition to the first one, will be developed as follows:

- Chapter II: presents the concepts and related work to this research.
- Chapter III: describes the experimental study.
- Chapter IV: provides a general discussion about the experimental study.
- Chapter V: concludes and presents future perspectives.

2 **Background Knowledge**

2.1 Business Process Management (BPM)

According to Dumas et. al (2013, p. 1), Business Process Management (BPM) is a science that studies the way work "is performed in an organization to ensure consistent outcomes and to take advantage of improvement opportunities". It is a multidisciplinary field of knowledge that incorporates principles, methods and tools to design, analyze, execute and monitor business processes.

These authors define a business process as " a collection of inter-related events, activities and decision points that involve a number of actors and objects, and that collectively lead to an outcome that is of value to at least one customer" (DUMAS et. al, 2013, p. 5). Events are the occurrences that activate a business process. They can trigger the execution of a sequence of activities in a process. Decision points refer to points in time when decisions are made that determine how a process is run. The results achieved at the end of a process can be determined by the decisions taken during its execution.

2.2 Knowledge-intensive Processes

According to Richter-Von Hagen, Ratz and Povalej (2005, p. 150), 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 predefined rules regarding the next steps in the flow. Unstructured processes are completely unpredictable and pose no predefined order for the execution of the activities, being commonly called knowledge-intensive processes (KiP).

KiPs are not easily suitable for automation, and they are conferred to a great degree of freedom to achieve their goals. For Richter-Von Hagen, Ratz and Povalej (2005, p.

149), value is created within KiP only 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, Marella and Russo (2015, p. 5) defined KiP as business processes "whose conduct and execution are heavily dependent on knowledge workers performing various interconnected knowledge intensive decision-making tasks".

The authors ranked the business processes along a spectrum based on the degree of structuring and predictability in order to better understand and position the KiPs in the BPM context, as shown in Figure 1.

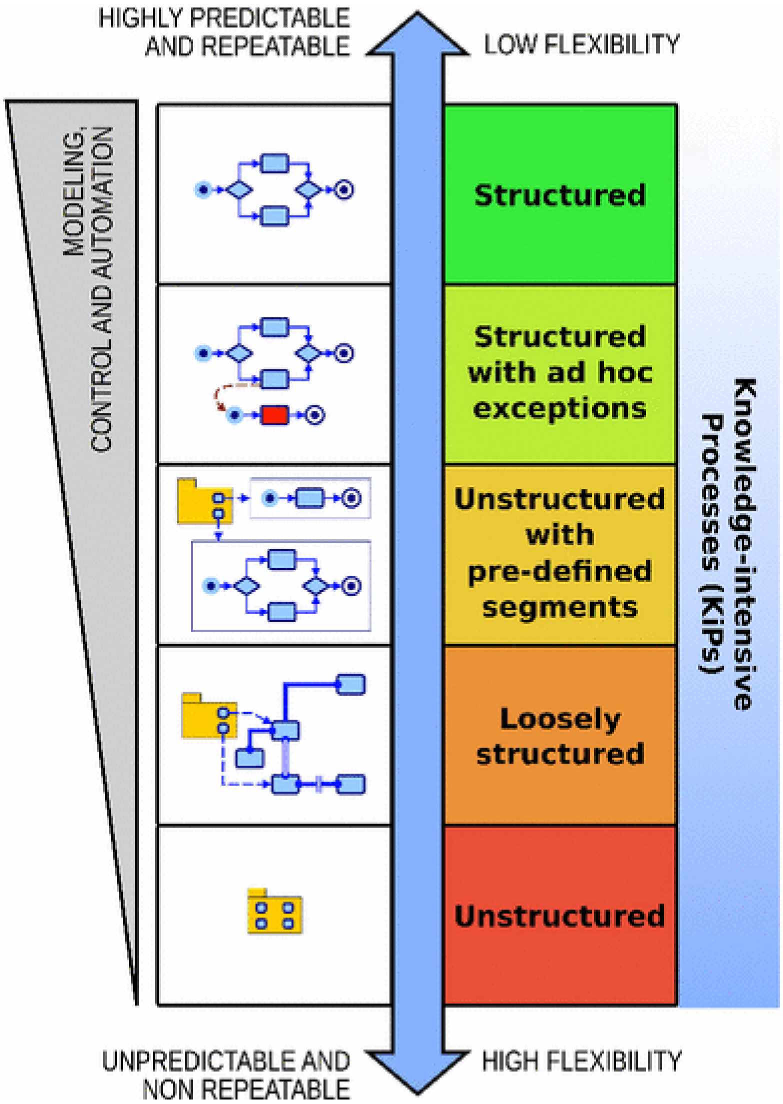


Figure 1 - The spectrum of process management. Source: Di Ciccio, Marella and Russo (2015, p. 6)

According to this spectrum, business processes can be classified into five classes that range from structured to unstructured. The more structured a process is, the more predictable is the work routine to be performed and the less flexible is the execution of its activities. On the other hand, the less structured a process is, the less predictable is the work routine to be performed and the more flexible is the execution of its activities. More predictable processes are more repeatable because the sequence of their activities is well known and controlled and vice versa. The class of KiPs is positioned transversely to the classifications presented by the authors, because according to them, "although the intensity of knowledge usually increases along the spectrum, almost all classes of processes (...) may include elements that make them intensive in knowledge".

Based on the definitions of KiPs available in the literature and in some representative application scenarios, the actors derived eight major KiPs characteristics: 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 (2012) 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 as show the Figure 2. The Business Process Ontology (BPO) comprises the traditional elements of business process modeling (such as activities, event flows, input/output data objects). The Collaboration Ontology (CO) depicts 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).

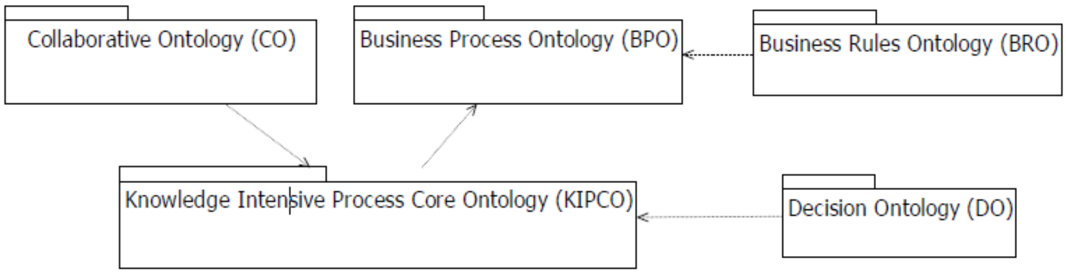


Figure 2 - Structure of KIPO. Source: França (2012)

KiPO argues that in a KiP the flow of activities (especially decision-making) is deeply influenced by tacit elements from its stakeholders. Decision makers in KiPO are called Agents, who accumulate previous Experiences about their work. Agents can be classified into two types: Impact Agent and Innovation Agent. While the Impact Agent performs a KiA, the Innovation Agent incorporates innovation in the execution of knowledge intensive tasks and proposes Alternatives to be considered in decision making. Unlike the Discarded Alternatives, the Chosen Alternative composes the decisions of the process.

According to França (2012, p. 116), in KiPO the alternatives of a KiP are discussed in a Socialization, that is, "a type of communicative interaction, composed by communication and perception, developed by an agent". In a Communicative Interaction, an Agent can assume two distinct roles: Sender, when sending messages, and Receiver, when receiving messages.

In KiPO, decision making can still be influenced by some constraint (Restriction), i.e., a Business Rule. In KiPO, business rules are defined by the Business Rules Ontology (BRO) subontology, and can be categorized into three types, as shown in Table 1.

Table 1 - Types of business rules defined in the Business Rules Ontology (BRO). Source: França (2012)

Source	Concept	Definition
Business Rules Ontology (BRO)	Reaction Rule	ECA (Event-Condition-Action) are statements that, in the event of a triggering event or if a set of conditions are satisfied, specify the execution of one or more actions. Optionally, after the execution of an action, post-conditions may be true.
	Derivation Rule	A rule that has condition and completion and which explains how an element of the model can be derived. It represents the derivation of new concepts in the domain from existing knowledge being modeled and presents a condition prior to derivation. When the state of the domain satisfies this condition, a conclusion will occur, adding a new element to the domain.
	Integrity Rule	This type of rule is structural. It does not change the domain, it does not create a new event or action that changes it. This rule only restricts something already existing between the concepts already foreseen.

In this undergraduate dissertation, we explore the decision making associated to business rules perspectives of KiPO.

2.3 Decision Mining

Business processes are stablished and structured based on business rules. According to Hay and Healy (2000, p. 4), a business rule is “a statement that defines or constraints some aspect of the business” within an organization. The main characteristic of a business rule is atomicity, meaning that it cannot be decomposed into other rules. The goal of a business rule is to state the business structure to control and influence its behavior.

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 (ROZINAT; VAN DER AALST, 2006, p. 420).

According to Rozinat and Van der Aalst (2006, p. 420), 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 has 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" (ROZINAT; VAN DER AALST, 2006, p. 421). The researchers created a decision tree algorithm, provided in ProM 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 (2013) stated that the technique proposed by Rozinat and Van der Aalst (2006) 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 by Manhardt, De Leoni, and Reijers (2013) in ProM[1] framework through the Multi-Perspective Process Explorer (MPE) plugin.

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 (∧), disjunction (∨) and negation (¬). 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: the minimum number of elements associated with decision-tree leaves (*min instances*) and the minimum control-flow fitness for each trace to be considered (*min fitness*). According Manhardt, De Leoni, and Reijers (2013, p. 132) “the *min instances* parameter is important as it influences whether the discovered guards are over-fitting (value too low), or under-fitting (value too high)”.

Recently, Mannhardt et al (2016) developed a new technique that allows discovering overlapping decision rules, since the algorithm proposed by Rozinat and Van der Aalst (2006) was able to mine only mutually exclusive splits (XOR). The solution was added to the Multi-Perspective Process Explorer (MPE) plugin.

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

De Smedt et al. (2016) argue that most of the literature on decision mining focuses on increasingly refined techniques on the retrieval of decision information in business 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, as shown in Figure 3.

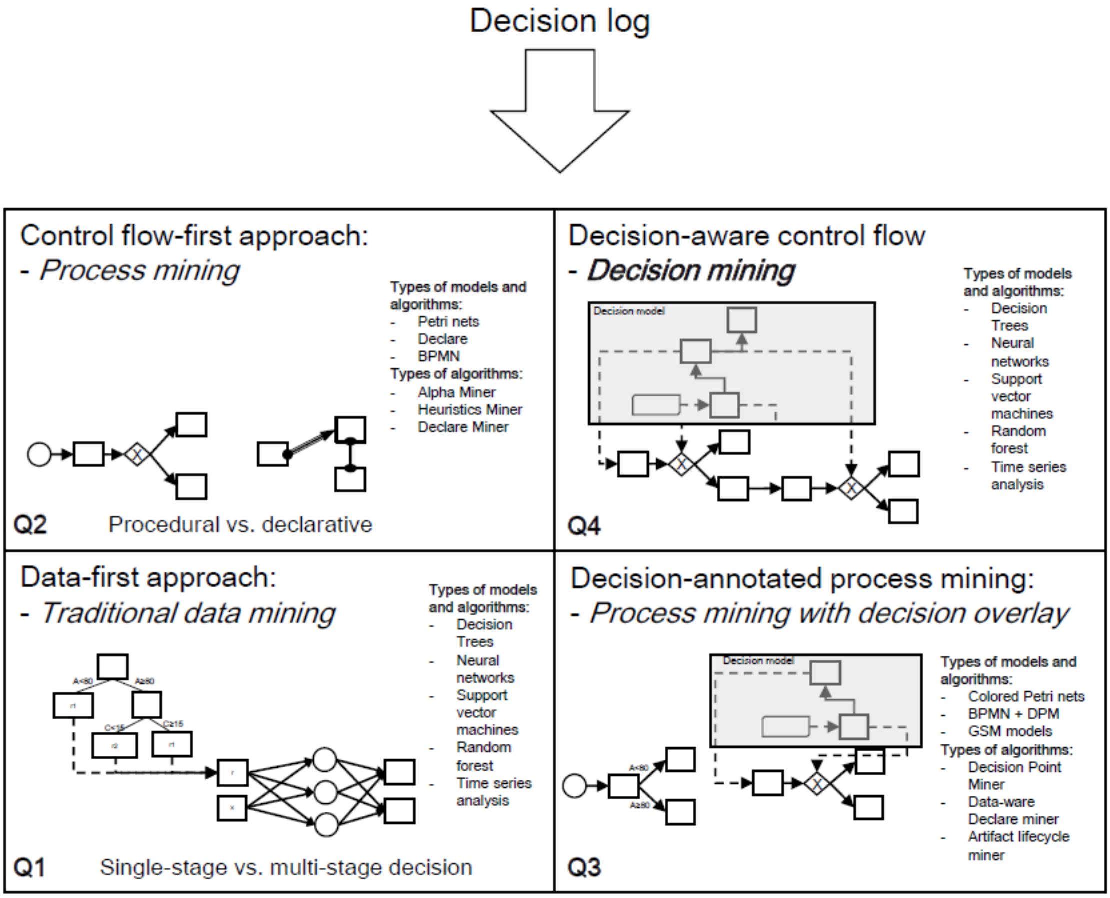


Figure 3 - The Decision Mining Quadrant. Source: De Smedt et al. (2016)

The first perspective refers to traditional data mining, focused exclusively on data from single instances. The second perspective is related to traditional process mining, procedural or declarative, which concerns to the sequential and concurrent aspect of the instances in the flow control. The first and second perspectives allow expressing mostly the decision rather than the logic of decision making. The third perspective refers to the mining of processes with overlapping decisions, i.e., the decision model is not connected to an entire workflow, but instead, it is present only in some parts. The fourth perspective addresses real decision mining since the decision model is used in a global and integrated way to the process model. They concluded that few studies cover this perspective that explains how the decisions discovered are related to the own decision-making process.

2.4 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 interchangeable between entities through an XML representation. 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" (OMG, 2016).

According to OMG (2016 p. 15), 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. The relationships between these three aspects of modeling are shown in Figure 4.

While business process models define tasks in which decision making occurs, Decision Requirements Diagrams define the decisions to be made, their interrelations, and their requirements for decision logic, which provide enough detail to allow validation and / or automation.

Figure 4 presents a simple Decision Requirements Diagram (DRD), consisting of three basic elements: an input, a decision and business knowledge models. The notation of all components of a DRD is summarized in Table 2. The concept of decision adopted by the DMN refers to the "act of determining an output value (the chosen option) from input values" (OMG, 2016, p 20). This decision logic may include one or more models of business knowledge.

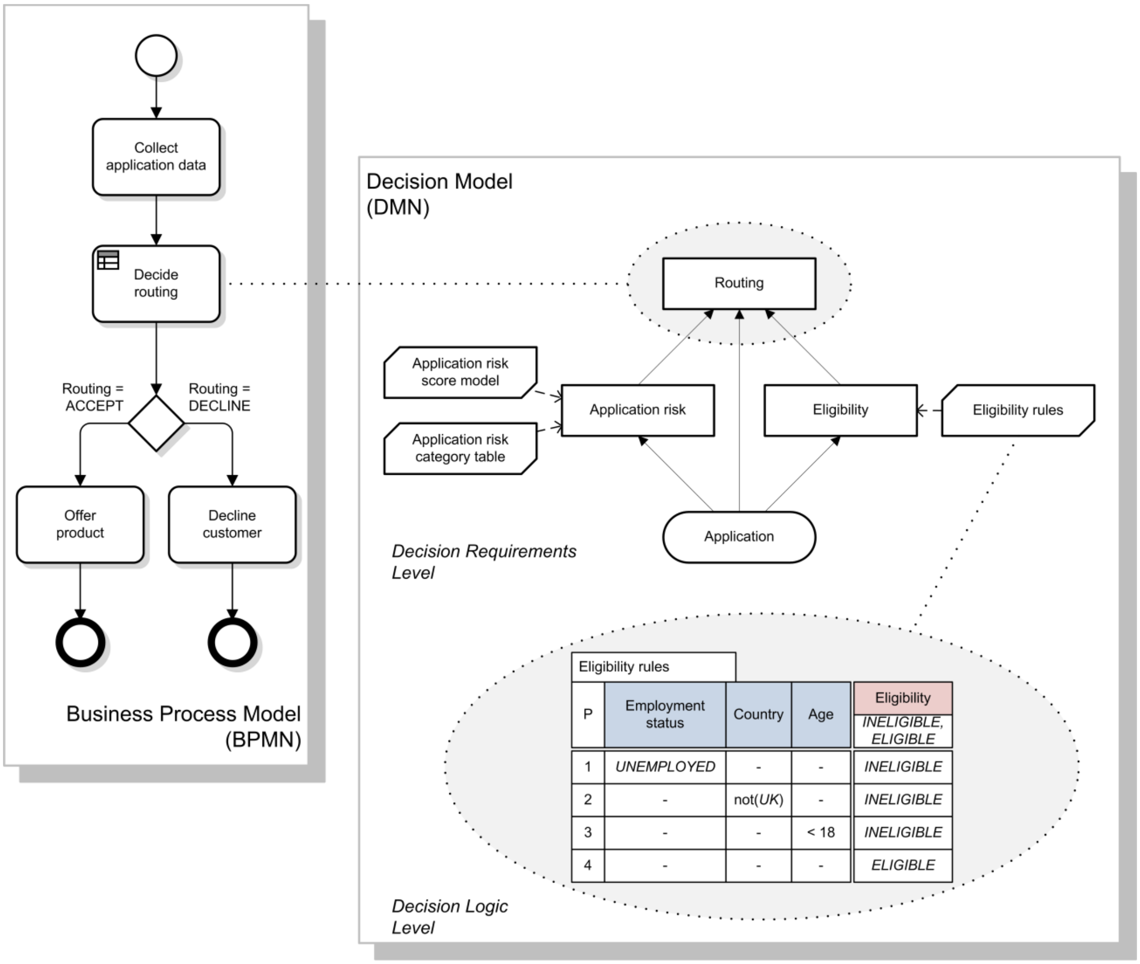


Figure 4 - Relationship between BPMN, DRD and decision logic. Source: OMG (2016)

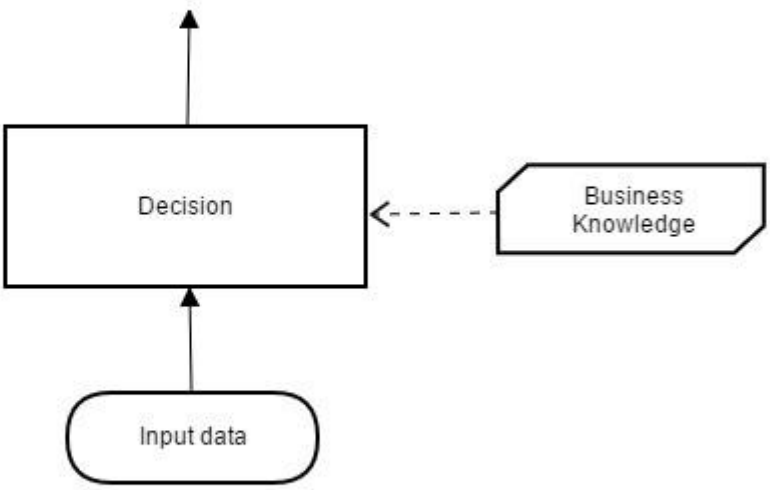
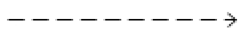
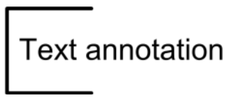


Figure 5 - Basic elements of a decision model. Source: OMG (2016)

Table 2 - DRD components. Source: OMG (2016)

Component		**Description**	**Notation**
Elements	Decision	A decision denotes the act of determining an output from a number of inputs, using decision logic which may reference one or more business knowledge models.	
	Business Knowledge Model	A business knowledge model denotes a function encapsulating business knowledge, e.g., as business rules, a decision table, or an analytic model.	
	Input Data	An input data element denotes information used as an input by one or more decisions. When enclosed within a knowledge model, it denotes the parameters to the knowledge model.	
	Knowledge Source	A knowledge source denotes an authority for a business knowledge model or decision.	
Requirements	Information Requirement	An information requirement denotes input data or a decision output being used as one of the inputs of a decision.	
	Knowledge Requirement	A knowledge requirement denotes the invocation of a business knowledge model.	
	Authority Requirement	An authority requirement denotes the dependence of a DRD element on another DRD element that acts as a source of guidance or knowledge.	
Artifacts	Text Annotation	A Text Annotation consists of a square bracket followed by modeler-entered explanatory text or comment.	
	Association	An Association connector links a Text Annotation to the DRG Element it explains or comments on.	

In a DRD, authorities can be defined for decisions or models of business knowledge. Such authorities are called sources of knowledge and may be, for example, domain experts responsible for defining or maintaining them, or documents from which business knowledge models are derived, or sets of test cases with which decisions must be consistent.

Figure 6 shows two representations of these authorities, one acting on one decision and the other on a business knowledge model. In the same figure, it may also be noted that the entries of a decision (Decision 1) can be entered into or exited from other decisions (Decision 2).

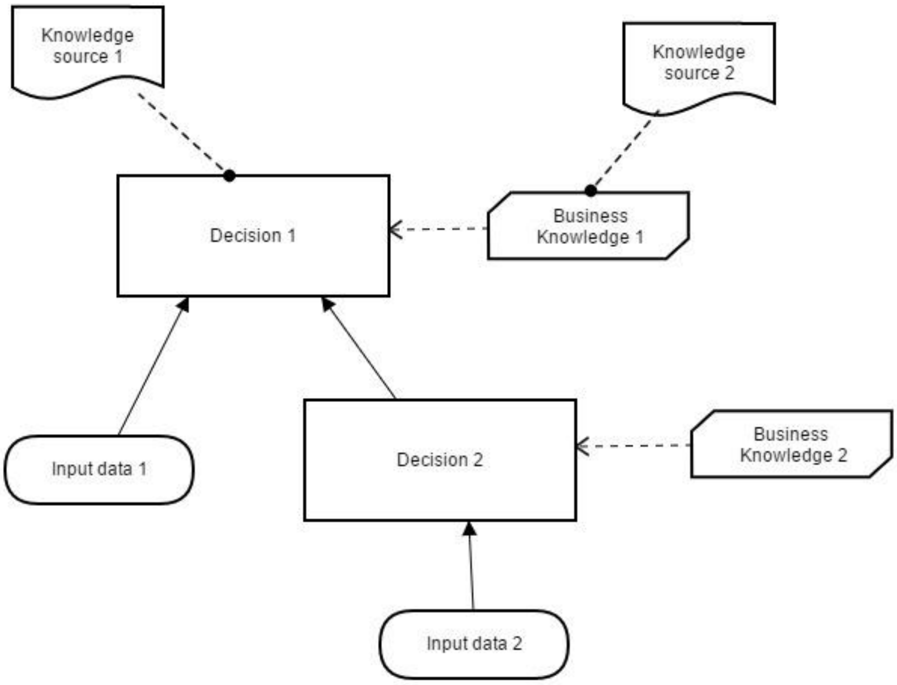


Figure 6 - A simple Decision Requirements Diagram (DRD). Source: OMG (2016)

The OMG (2016, p. 18) 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 provide 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 two 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

3.2.1 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 undergraduate dissertation is on decisions, we did not choose techniques commonly used to discover unstructured models (such as declarative ones) since it is difficult to find decision mining techniques available in ProM. The MPE plugin was chosen to support this task because the technique fits into the third perspective of the framework proposed by De Smedt et al (2016). 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 Figure 7. 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.

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

Figure 7 - 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 "AddNote" 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 as show the Figure 8.

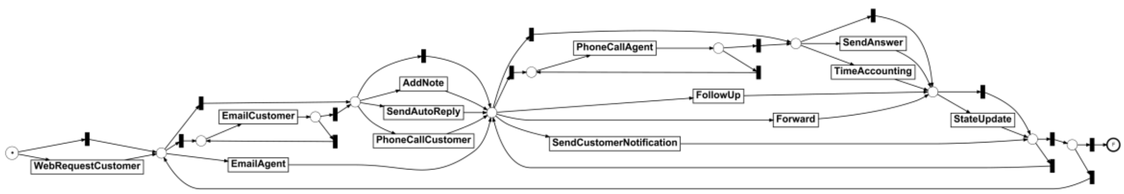


Figure 8 - Base model generated by MPE

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 Petri net 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.

3.2.2 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 at the position named "sink 6" for the "AddNote" activity. The “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 3.

Table 3 - 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" || ticketState == "closed successful") || ticketState == "closed with workaround") || ticketState == "merged") || ticketState == "new") || ticketState == "open") || ticketState == "pending auto close+") || ticketState == "pending auto close-") || ticketState == "pending reminder")
Correct Events	12570	12570	12570
Wrong events (data)	0	0	0
Missing Events	0	0	0

In Table 3, 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 Figure 9.

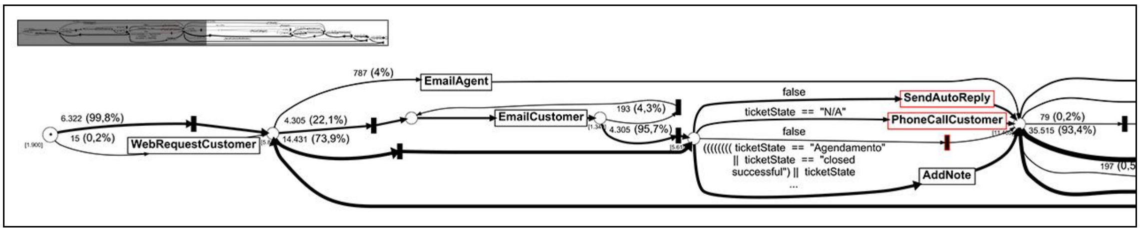


Figure 9 - Rule discovered by decision tree algorithm related to “AddNote” at “sink 6”

In Figure 9, it is possible to visualize the large and complex discovery rule in the "Discover Data Perspective" mode of MPE found in the position called "sink 6" in the Petri net model. This rule, shown in Table 3, refers to the thicker arc that takes the transition (activity) "AddNote".

In Figure 9, we notice that some activities are highlighted (involved by a red rectangle). 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 10 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.

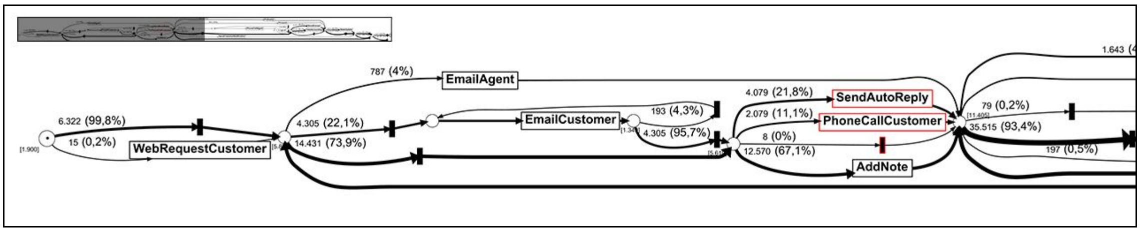


Figure 10 - Control flow with the percentage of events that executed each activity

It is also possible to evaluate the accuracy of the base model generated after the computational alignment. Table 4 shows a comparison of the accuracy of the base model with "sink 6", from which the rules were found. The accuracy of the base model 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. According to Buijs, Van Dongen and Van Der Aalst (2012), replay fitness, simplicity, precision and generalization are four dimensions that must be considered to evaluate the quality of the discovery of a process model.

Table 4 - 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

3.3.1 Method

The fourth scenario of the experiment explored the unstructured part of the event log, which records e-mail messages, using text mining techniques. As these are non-categorized data, this information can't be used by conventional process mining techniques. The goal was to seek decision rules in natural language 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 (e-mails) exchanged between the employees and the clients.

In this scenario, we sought to identify the existence of decision rules implicitly or explicitly declared in natural language, a complementary approach to the previous one that could bring more understandable results to the company's managers. Because they are incident resolution records, the mining of these data can bring a better understanding of the decisions made by tacit employees, whether or not they are governed by rules, providing information that can enrich the structured data log. At first, the discovery of such rules was not made fully automated, as this would require the use of natural language processing techniques, which is not the main focus of this experiment.

These conversations were extracted from the original event log and copied to a new text file. After loading the text in free software R, a cleaning procedure was performed that removed all duplicate messages. These messages are sometimes repeated because their content is used to perform other activities throughout the execution of the process. The original text file had 63,424 messages. After cleaning, the text file now has 28,353 messages (44.7%), a significant reduction of 55.3% of the total.

Afterwards, the function *grep* was used to filter the messages in which the following terms are found (in singular): “incidente”, “regra”, “procedimento” and “decisão”. In the case of the term “decisão”, the radicals “decis” e “decid” were used. The first one is related to the noun decisão and the second to the verb decidir, which may appear in different types of forms (mode, time and voice). This function returned 114 messages.

Then, a new search was performed on the 1144 messages to find incidents that were resolved. The function was again run to filter messages where the term "solução" appears. In this case, the radical "soluc" was used, which refers to both the noun “solução” and the verb “solucionar”. After executing the function, only 55 messages were filtered, corresponding to 4.8% of the total. At the end, the results were saved to text files. The script used to perform the text mining in the log is shown in Figure 11.

```
# Script
# Documentation of the GREP function
# https://stat.ethz.ch/R-manual/R-devel/library/base/html/grep.html

# 1 - Load the text file

messages <- readLines ("messages.txt", encoding = "UTF-8")
messages [1:10]
length (messages)
```

```
# 2 - Remove duplicate messages

messages <- unique (messages)
messages [1:10]
length (messages)

# 3 - Create line and phrase filtering functions

filterLines <- function (vector, variable) {
      result <- grep(paste(vector, collapse="|"), variable, ignore.case = TRUE, perl = TRUE, value
= TRUE)
      return(result)
}

filterPhrases <- function (string, variable) {
     result <- grep(string, unlist(strsplit(variable, '(?<=\\.)\\s+', perl = TRUE)), value = TRUE)
     return(result)
}

# 4 - Filter all messages that have the terms "incidente", "regra", "procedimento" and "decisão"
(use "decis" and "decid" radicals)

terms <- c ("incidente", "regra", "procedimento", "decis", "decid")
filteredMessages <- filterLines (terms, messages)
filteredMessages [1:10]
length (filteredMessages)

# 5 - Filter all messages, from the previous result, that have the term "solução" (use radical
"soluc")

term <- c ("soluc")
solution <- filterLines (term, filteredMessages)
solution [1:10]
length (solution)

# 6 - Filter only the sentences in which the term "solution" appears

solutionPhrases <- filterPhrases ("soluc", solution)
head (solutionPhrases)
length (solutionPhrases)

# 7 - Export results to .txt file

write.table(solution, file = "solutions.txt", quote = FALSE, sep = "\t", na = "NA", row.names =
FALSE, col.names = FALSE, fileEncoding = "UTF-8")
write.table(solutionPhrases, file = "solution_phrases.txt", quote = FALSE, sep = "\t", na = "NA",
row.names = FALSE, col.names = FALSE, fileEncoding = "UTF-8")
```

Figure 11 - Script used to mine texts in the log

3.3.2 Results

Among the 144 messages obtained, we selected 4 that were significatively related to incident solution, as shown in Table 5. To keep the privacy of clients and employees, some names and parts of the message text were removed.

Table 5 - 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 12 shows a decision model of the treatment of that incident using the DMN notation.

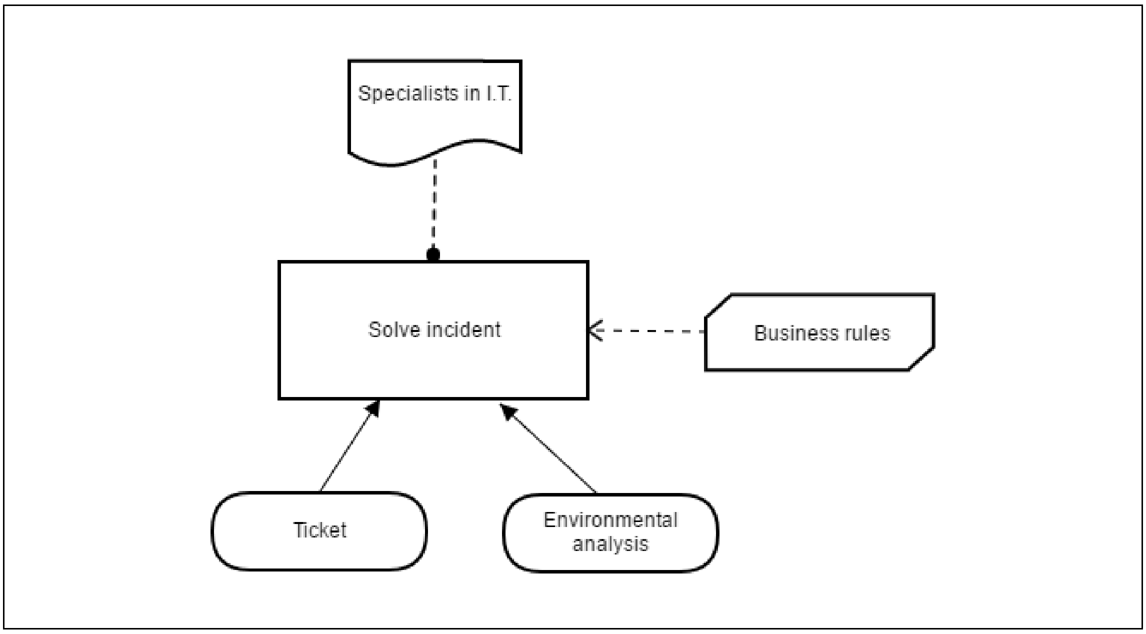


Figure 12 - Decision Requirement Diagram of the 1st and 2nd incident (Table 5)

In Figure 12, 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. A business rule that can be derived from the resolution of this incident is: "A local environment analysis must be performed to perform a proper correction procedure." For this incident, the same DRD of Figure 12 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 Figure 13, this incident is modeled in a DRD.

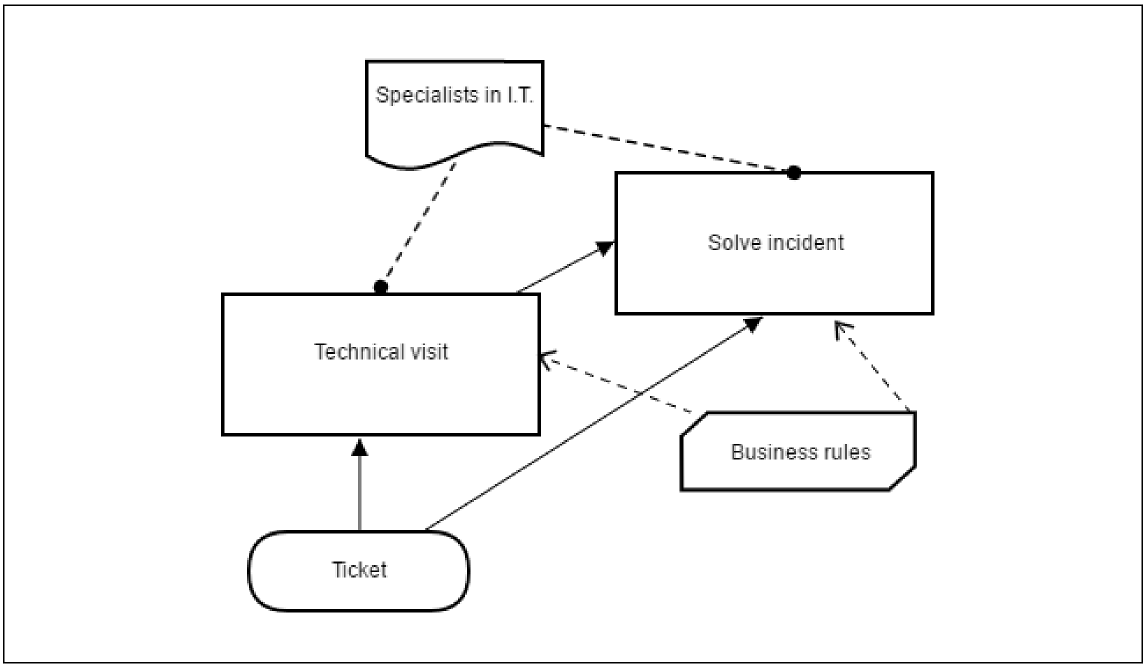


Figure 13 - Decision Requirement Diagram of the 3rd incident (Table 5)

We notice that the decision about how to solve an incident depends on another decision, i.e. assessing the need for a technical visit. According 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 (Figure 14). 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.

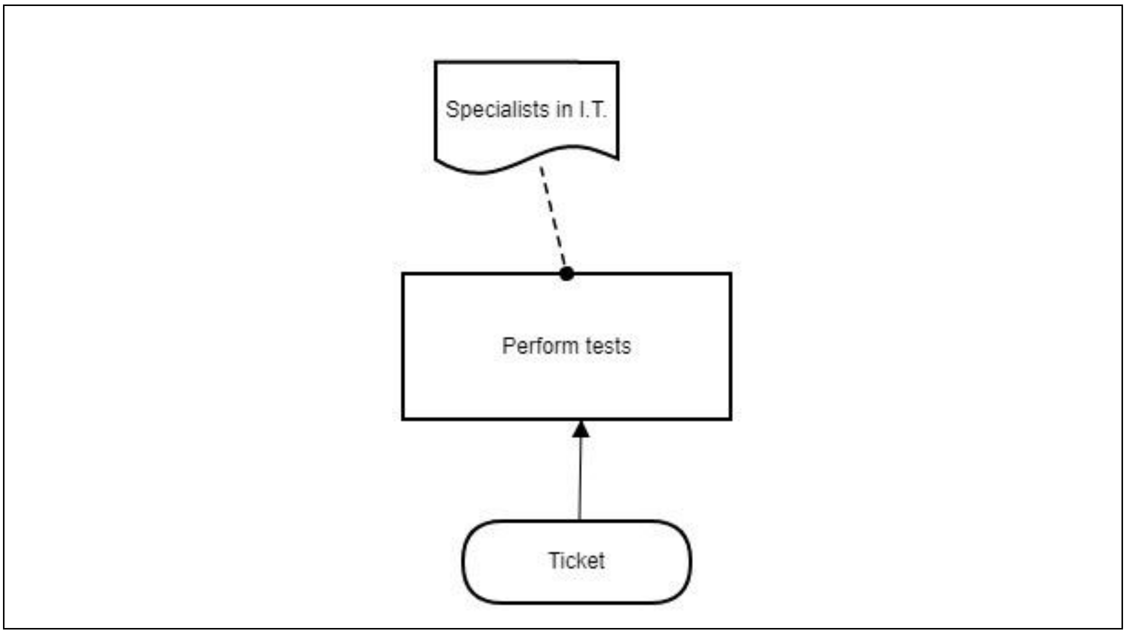


Figure 14 - Decision Requirement Diagram of the 4th incident (Table 5)

The analysis of the four scenarios under the KiPO perspective, we are able to identify that the employee technical staff as the agents mostly responsible for decision making during the incident treatment process. Some of them are Impact Agents since they are responsible for executing the KiP; others are Innovation Agents, since they incorporate new ways of executing knowledge-intensive activities and propose alternatives to be consider during decision-making. These alternatives are discussed during a Socialization, which in the context of the company is performed by the message exchange among KiP agents. In the examples of this case study, it is possible to note how constraints (in this case, decision rules that may be converted into business rules) impact on decision making.

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 (MPE) may have discovered few rules due to the low number of event attributes in the company log, although the authors of this technique were able to discover rules in a data log that had only two data attributes (DE LEONI; VAN DER AALST, 2013, p. 1461). The data log used in this experiment has five data attributes, but if there were more attributes, the decision mining technique could discover more rules.

In addition, the decision rules discovered by the MPE 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.

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 Janssens et al (2016, p. 124), for whom in a Kip scenario, a decision model is required, and it is more relevant than the BPMN model.

Table 6 shows the rules discovered in the structured and unstructured log. There are differences in the nature of these rules. The rules obtained from the structured log have global coverage, that is, they refer to a set of instances of the model generated by the decision mining technique (MPE). It is also important to note that these rules were discovered by manually setting two parameters of the Discover Data Perspective mode of the MPE: *min instances and min fitness*. Therefore, there are several possible combinations of values that can be manually made by the user to define whether he wants more restrictive or more comprehensive rules.

However, the rules obtained from the unstructured log have local coverage, that is, refer to instances of the process mined. Furthermore, these rules were inferred from a non-automated textual analysis that verified the existence of rules, procedures or standards in decision-making regarding problem solving. Although the discovered rules have distinct scopes, it is noted that in this study the understanding of them is greater at the instance level, which reinforces the importance of further exploring unstructured data log with other techniques not supported by mining of conventional processes.

Table 6 - Rules discovered in structured and unstructured log

Rules Discovered in Structured Log	Rules Discovered in Unstructured Log
article_id > 605709.0	The first alternative to be tested in a case of internet interruption must be the pre-existent connection.
article_id > 622246.0	A local environment analysis must be performed to perform a proper correction procedure.
((((((((ticketState == "Agendamento" || ticketState == "closed successful") || ticketState == "closed with workaround") || ticketState == "merged") || ticketState == "new") || ticketState == "open") || ticketState == "pending auto close+") || ticketState == "pending auto close-") || ticketState == "pending reminder")	A technical visit must be requested for highly critical incidents.

A few limitations of the experiment are observed. In the first step, we tested only one technique for mining decisions within KiPs, proposed by Manhardt, De Leoni and Reijers (2015). We also tested the decision mining technique proposed by Bazhenova, Buelow, and Weske (2016), which automatically models the discovered decisions using the DMN notation; however, the results obtained were not satisfactory using the same dataset. We also could not test other algorithms mentioned in De Smedt et al. (2016), such as declare miner, due to the difficulty of finding decision mining techniques available in ProM.

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. Although the base model generated by MPE achieved a high fitness rate of 100%, its precision was low (37.6%), which significantly affected its quality, since it allows the existence of behavior that is not observed in the log.

The scenario with text mining enabled the discovery and inference of more precise decision rules through the decision mining technique that was applied. Some of these rules, identified in the procedures executed by the company employees, may not have been institutionalized by the company. Thus, identifying these rules may help the company to optimize the process of incident resolution and help the technical staff to make the best decisions in risk situations, in order to increase the quality of the delivered service. This scenario also 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 use other approaches and techniques to compare the results.

References

BAZHENOVA, Ekaterina; BUELOW, Susanne; WESKE, Mathias. Discovering decision models from event logs. In: **International Conference on Business Information Systems**. Springer, Cham, 2016. p. 237-251.

BUIJS, Joos CAM; VAN DONGEN, Boudewijn F.; VAN DER AALST, Wil MP. On the role of fitness, precision, generalization and simplicity in process discovery. In: **OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"**. Springer, Berlin, Heidelberg, 2012. p. 305-322.

CAMPOS, Júlio et al. Discovering business rules in knowledge-intensive processes through decision mining: an experimental study. In: **International Conference on Business Process Management**. Springer, Cham, 2017. p. 556-567.

DE LEONI, Massimiliano; VAN DER AALST, Wil MP. Data-aware process mining: discovering decisions in processes using alignments. In: **Proceedings of the 28th annual ACM symposium on applied computing**. ACM, 2013. p. 1454-1461.

DE SMEDT, Johannes et al. Decision Mining in a Broader Context: An Overview of the Current Landscape and Future Directions. In: **International Conference on Business Process Management**. Springer, Cham, 2016. p. 197-207.

DI CICCIO, Claudio; MARRELLA, Andrea; RUSSO, Alessandro. Knowledge-intensive processes: characteristics, requirements and analysis of contemporary approaches. **Journal on Data Semantics**, v. 4, n. 1, p. 29-57, 2015.

DUMAS, Marlon et al. **Fundamentals of business process management**. Heidelberg: Springer, 2013.

FRANÇA, J. **Uma ontologia para definição de processos intensivos em conhecimento**. 2012. Dissertação de Mestrado (Mestrado em Informática) – Departamento de Informática Aplicada (DIA), Universidade Federal do Estado do Rio de Janeiro, Rio de Janeiro.

HAY, David; HEALY, Keri Anderson. **Defining business rules ~ what are they really?** Final Report, revision 1.3, the Business Rules Group, formerly the GUIDE Business Rules Project, 2000.

JANSSENS, Laurent et al. Consistent integration of decision (DMN) and process (BPMN) models. In: **Proceedings of the CAiSE'16 Forum, at the 28th International Conference on Advanced Information Systems Engineering (CAiSE 2016)**. CEUR-WS. org, 2016. p. 121-128.

MANNHARDT, Felix; DE LEONI, Massimiliano; REIJERS, Hajo A. The Multi-perspective Process Explorer. **BPM (Demos)**, v. 1418, p. 130-134, 2015.

MANNHARDT, Felix et al. Decision mining revisited-discovering overlapping rules. In: **International Conference on Advanced Information Systems Engineering**. Springer, Cham, 2016. p. 377-392.

OBJECT MANAGEMENT GROUP (OMG). **Decision model and notation (DMN)**. Version 1.1, formal/2016-06-01. 2016

RICHTER-VON HAGEN, Cornelia; RATZ, Dietmar; POVALEJ, Roman. Towards self-organizing knowledge intensive processes. **Journal of Universal Knowledge Management**, v. 2, p. 148-169, 2005.

ROZINAT, Anne; VAN DER AALST, Wil MP. Decision mining in ProM. In: **International Conference on Business Process Management**. Springer, Berlin, Heidelberg, 2006. p. 420-425.