MASTER'S THESIS

Human decision point mining in semi-automated workflows

Houwer, W (Wietse)

Award date:
2020

Link to publication

Human decision point mining in semi-automated workflows

Degree program:	Open University of the Netherlands, Faculty of Management, Science & Technology Business Process Management & IT master's program
Course:	IM0602 BPMIT Graduation Assignment Preparation IM9806 Business Process Management and IT Graduation Assignment
Student:	Wietse Houwer
Identification number:	
Date:	21 June 2020
Thesis supervisor	L. Rutledge
Second reader	S. M. M. Joosten
Version number:	1.0
Status:	final version

Abstract

Within the domain of process mining, automated business rule discovery can greatly advance the discovery and conformance checking of business rules. Nevertheless, the human applied business rules and decision points in a business process remain subjective and can differ greatly between employees, between departments, and between automated rules. This can leave a gap in the overall quality of business rule understanding and governance. Therefore, we present a mining approach to reveal the *human decision points* within a set of mined decision points. This approach could support process owners and management in revealing the hard-to-find business logic from human actors. Our research is based on a behavior detection technique from online gaming (Choi, 2016), and in 5 consecutive process mining steps, we aim to provide a multi-perspective model that high lights the human decision points. We applied our proposition onto an event log from an unstructured semi-automated workflow process in the process mining tool ProM (ProM, 2019). The results show progress in the individual five mining steps but errors in intermediate model compatibility and comparison. Although individual working ProM implementations deliver sound intermediate models, we did not succeed to create a stable mining experiment to demonstrate our proposal. Further research is needed to define better compatible models and explore other supporting tooling, more tailored to data mining, to create a better behavior model. A richer data set will increase the chance of having more human behavior distinguishing attributes.

Key terms

process mining, business rules mining, human decision point mining, human behavior detection

This thesis explains a process mining approach that focusses on revealing human decision points in an unstructured semi-automated workflow. A better understanding of human decision points is important, as they are more subjective and more difficult to capture, compared decision points that are programmed. Also, as per design, the human decision points are often high-value expert tasks that can only be done by humans, making them very relevant for monitoring and governance. Human decision point tent to generate more process variance and can be an interesting source of information for process owners and management. We proposed a mining approach that takes a sample event log from a workflow system and tries to mine sequentially for process discovery, decision point discovery, and human decision point discovery. Our approach is based on a game bot detection method from the field of online gaming research, that uses a bot behavior model to detect small differences player behavior from game logs. Our philosophy is that if a behavior model can successfully detect the bots amongst the human players, we could also use this approach to detect the human decision points amongst the total set of decision points. In five practical mining steps, we aim to come to this result. This includes the manual creation of a human behavior model to differentiate human behavior on decision points from other decision points. The result would be displayed in a multi-perspective process model, where the human decision point would be highlighted, e.g. in a colored Petri Net or a manually created multi-perspective model.

The results show that the initial process discovery steps are successful, issues arise in the behavior model creation and model comparison. Although individual working ProM plugins appear to deliver sound intermediate models, we did not succeed to create a consistent mining process to demonstrate our proposal with a working experiment. Further research is needed to improve model comparison techniques and tool support. Also, our data set did not contain sufficient process attributes to detect minor or major differences in behavior.

Contents

1. Introduction

This research paper describes a mining approach to reveal human decision points in an unstructured semi-automated workflow process.

1.1. Exploration of the topic

The research field of rule mining focusses on the discovery of business rules from application event log data. Various approaches and techniques have been proposed (Aalst, 2016)(De Leoni, 2016)(Garcia, 2019) to mine large and complex business processes for business rules in an automated way, which otherwise would be challenging to retrieve through manually business rule elicitation. Business rule mining (Rozinat, 2006)(Bolt, 2018)(Crerie, 2009) allows a bottom-up research approach, to discover actual business rules of business events in applications supporting a business process. This allows for valuable insights into process details and process efficiency, especially in process areas where the business rules are less clear or even subjective because of human process expert interventions. This human factor brings human business logic to the process, which is not captured in or part of an automated system, like workflow systems. It is application-external business logic from a human actor that is being applied to the process, at process execution time and this can make the process instances *vary* and or even *subjective*, as different human actors may cause small or big process variation, based on their understanding and interpretation of the organization's business rules. Also, workflows with human expert tasks tend to be more *unstructured*, as the workflow needs to be flexible to support the decision direction and steer on routing from human users. Little literature (Rijers, 2005)(Khodabandelou, 2013) has been found about how human tasks in an unstructured semi-automated workflow system influence the overall governance of business rules. There are however interesting benefits to gain from when an organization can collect and govern these human business rules. Therefore, this research focusses on the discovery of human decision points in semi-automated workflow applications. Through combining different mining approaches, we aim to demonstrate how to detect human decision points which would otherwise be difficult to detect.

1.2. Background

This research is executed in the domains of data science, process mining, and business rule mining. The broad research field of data science has developed various areas of data mining techniques, one of those focusses on process mining. From application event logs, process mining techniques allow process analysis, aid to measure process compliance (Thullner, 2011), process deviations (Bolt, 2018), or even process discovery (Aalst, 2016). Within the field of process mining, business rule mining is an area where the research focusses on *business rule discovery* to bring valuable insights about a process of its business rules (Crerie, 2009). Business rules are considered to be a very valuable part of an organization's inner workings and intellectual assets but are often poorly documented, poorly maintained, or even well-known inside an organization (Crerie, 2019). On top of that, human business rules, as in human decision points in a process, maybe even more difficult to discover, as the human decision points are often even less documented (Crerie, 2019) and may slightly or substantially differ between human workers or not explicitly or consistently recorded in any system. This gives challenges when investigating processes with these informal, subjective human decision point, for processes where human judgment is steering process output. Therefore, we would like to extend the field of Business rule mining with an approach to solve this challenge, by focusing on extracting human decision points out of event data that prove statistically existing business rules in a workflow process.

1.3. Problem statement

The problem with processes in semi-automated workflow systems that contain a degree of human tasks, is the lack of formal business rules for those human tasks in the system (Sonntag, 2010). Some or large parts of a workflow process will get its steer from human actors applying their interpretation of business rules in their workflow tasks; rules that are not part or captured in the algorithms of the workflow system itself. For organizations, this can give a challenge: despite having an automated or semi-automated workflow system to standardize and unify business processes, these human interventions still can give considerable output variance, troubling organizations that try to manage and measure their business rules for continuous improvement. This particularly becomes visible in workflow processes with a high degree of human tasks, where it is more likely that consecutive human tasks will accumulate process variance and a higher level of output variance, either in output quality or speed (Fernandes, 2006). Workflow systems supporting such processes, typically do not have much formal business rules. Instead, these applications allow the process flows to be unstructured to let the steer done by the human actors, but this gives challenges to the process owners and management to capture and improve the business rules, as there is often a mix of automated rules from the workflow system and human decision points. We propose an approach to discover and differentiate the human decision points from the automated decision points. Therefore, we have taken a sample workflow process with a mix of automated and human decision points from the IT Service Management (ITSM) domain (ITIL v4, Axelos, 2019), where human actors are tasked to solve complex support tickets with the unstructured rules of the incident resolution process, according to Incident Management (IM) as specified by ITIL (ITIL v4, Axelos, 2019). Our sample process has an intrinsic risk of process instances to deviated or linger about in the system, due to human decisions being made. This is a concrete concern for process owners and Service Delivery management as these instances can result in tickets being sent back and forth between teams which lead to time wasted and breached IT service levels. Understanding and continuously improving this workflow process and the human decision points are crucial to the performance of the IT support function. To support such process analysis, we aim to demonstrate how to detect these human decision points which would otherwise be difficult to extract.

1.4. Research objective and questions

In this study, we aim to provide a business rule mining implementation on the event log of a workflow system, to detect the human decision points that drive process variance. With this we try to demonstrate how business rule mining can also be used to retrieve less explicit business rules, to still allow for in-depth process analysis on workflow processes with a high degree of human tasks.

Research question:

- How and to what extent can process mining be utilized to retrieve human decision points applied in a semi-automated workflow process?

1.5. Main lines of approach

The mainline of the research approach is to describe the current challenges around capturing human decision points and the potential risk they bring to process conformance and performance in a workflow system. We will do so by explaining a general example of a semi-automated workflow process from the field of ITSM Incident Management (IM). From this process domain, we want to look to the research domains of process mining, business rule mining, and workflows with human actors, to see what mining approaches and techniques are available for mining the less obvious human decision points in a process. Subsequently, we want to prove that process mining can also be utilized to retrieve human decision points in an unstructured workflow process. We will try to demonstrate our proposal by running an experiment with combining several mining techniques. To conclude we will reflect on our research results through a discussion, conclusion, and recommendation for further research.

2. Theoretical framework

This section provides the theoretical framework.

The theoretical framework builds upon the research domain of *process mining*. By applying process mining tailored to mining decision points and process variance that human actors generate in handling semi-automated workflow tasks, we aim to discover those human decision points, separately from other decision points.

The theoretical framework for this research relates to three areas:

- Process mining
- Decision mining
- Human behavior

The overlap of these three areas will be our *area of research*.
Within this area of research, we ask ourselves how and to what extent can process mining be utilized to retrieve human decision points applied in a semi-automated workflow process.

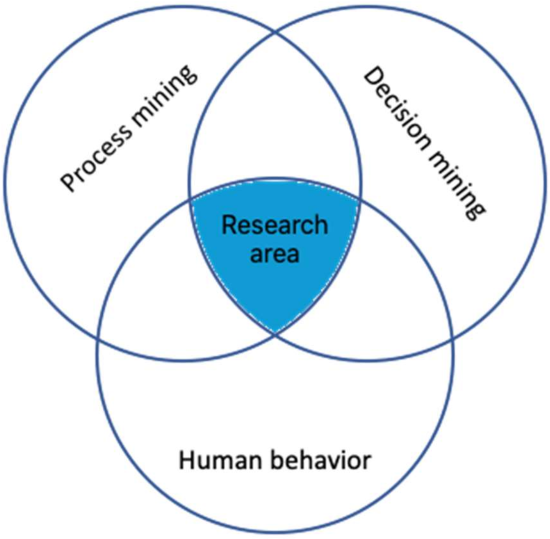

Figure 1: research domains and focus area

2.1. Research approach

In this section, we set out a high-level theoretical framework. In chapter 3 we specify in more detail how we will technically approach our mining experiment and results follow in chapter 4. First, we provide an up to date view from the literature on the three domains of decision mining, process variance mining, and business rule mining. This provides a theoretical background from where we identify existing mining techniques from previous research and we highlight where we see opportunities for re-use and where we see gaps. Subsequently, this will lead to our proposal for testing an actual mining implementation on a sample workflow event data set. The results of this test will be explained and generalized for a broader application. In the last part of our research, we will discuss our conclusions, limitations of this research, and recommendations for further work.

2.2. Implementation

For the literature study, we used our university's digital library service and Google Scholar. We found over +40 relevant articles that helped to answer our questions and also, from our OU university's *research circle on Business Rule Mining*, we gained insights from parallel business rule mining research.

2.3. Results and conclusions

This section sets out the theoretical framework that we developed, At the end of each section, we will list the conclusions of the theoretical framework and the implications of these conclusions for the remainder of the research as objectives for the follow-up research we propose. We structure this section in the three domains, starting with the underlying research domain of process mining and process variance through which we aim to discover processes, subsequently the research domain of decision mining and lastly details on typical challenges with human behavior detection.

2.3.1. Process mining and process variance analysis

Process mining principles and techniques are the foundation upon which we discover our workflow processes and process variances, based on event data from an IM ticket system. The literature on process mining is plentiful, but we have limited our literature research to 'process mining', 'process variance analysis', 'workflow', and 'ProM', our process mining research tool.

Process Mining:

Van der Aalst gives in *Process mining: data science in action* (Aalst, 2016) a thorough introduction into process mining, both theory as practice and from various perspectives. Process mining techniques sit between data mining and machine learning (Aalst, 2016). Process mining uses event logs that are generated through workflow engines and produce Petri nets to express process models, using different *mining algorithms*. The main types of process mining approaches are process discovery, process conformance checking, and process enhancement. Different types of process mining algorithms are developed over the years like heuristic miner algorithms, inductive miners, alpha miners, ILP miners, evolutionary miners, and many more (Aalst, 2016). The majority of these algorithms require an event log that contains an execution trace for the process instance and most of them, it is necessary to identify the process instance identifier. Often the algorithms work with threshold parameters to set sensitivity or generalization. In process mining tools like ProM, algorithms, and utility programs are available to investigate data, event logs, and process models. Augusto provides with *Automated Discovery of Process Models from Event Logs* (Augusto, 2019) a systematic literature review and comparative evaluation of automated process discovery methods. Over 50% of the 35 studies provide implementation as a plug-in for the ProM framework, showing the importance of ProM to the research community. Mannhardt et al. with *Guided Process Discovery - A pattern-based approach* (Mannhardt, 2017) shares a view on the importance of *comprehensibly* of event abstraction and demonstrates to what extent the domain knowledge can be discovered.

Process variance analysis:
To investigate the variance in a process, we also wanted to understand the correlating and clustering of dynamic behavior for process variance analysis. De Leoni et al. with *A general process mining framework for correlating, predicting, and clustering dynamic behavior based on event logs* (De Leoni, 2016) present in this interesting research a combination of variance analysis and relate it to a more abstract layer of correlating business process characteristics. From their implementation, we learn how to split event logs into clusters of traces with similar behavior, which gives different *perspectives* on the same process (control-flows, data-flows, time-perspectives, etc.). Bolt, et al. present in *Process variant comparison: using event logs to detect differences in behavior and business rules* (Bolt, 2018) two important domains (process variant analysis and detection of business rules), which are very relevant for our research objectives. We will use an approach similar to Bolt, to apply it to our case data set and find an alternative use for the approach to detect human decision points.

2.3.2 Business rules, decision points, and decision point mining

We first establish the common definitions for *business rules* and *decision points*, and subsequently, we look at the research field of decision point mining.

Defining business rules:
A business rule is a guideline to influence or guide the conduct of business, a sentence that defines or qualifies any aspect of the business, representing the knowledge of experts (Ross, 2003).
We work with the Business Rules Group's definition (BRG Group, 23) of a business rule which states that:

- *a business rule is a statement that defines or constrains some aspect of the business*

Business rules accumulate all the knowledge of the business, built over time by the organization, making the rules a major structural and intellectual asset for the organization (Crerie, 2009). There are several initiatives to organize and classify business rules, but there are no universal standards (BRG Group, 23) (OMG, 24, 49) (SBVR, 24) and Decision Model Notation (DMN, 49). Crerie (Crerie, 2009) states that business rules can be considered as decision points and vice versa because they exhibit activities that are restricted by rules. Our topic of human decision points is scarcely documented. In literature, we found work (De Nicola, 2011) that addressed the implicit or ambiguous nature of business rules. In most contexts, it was addressed that there is a clear need for early formalization of business rules, to do automated analysis on them. So, interestingly, most literature we found acknowledges the ambiguity of human business rules but give little suggestions on how to retrieve these for formalization, hence the relevance of our proposal for this. In section 2.3.3 we come back to human behavior detection.

Business rule mining:
If we look at related work for business rule mining, we find interesting sources with different approaches to rule discovery. In many circumstances, manual elicitation and business rule collection are just not feasible or realistic (Crerie, 2009)(De Nicola, 2011). In larger organizations, business processes and their business rules can be long, complex, spread over different company silo's, or have localizations. One would rather look at the actual process execution in the underlying business process systems like a companywide ERP system. This is where automated business rule mining comes into play. An approach from Rozinat/Aalst (Rozinat, 2008) is based on process attributes and decision trees, one approach from Crerie (Crerie, 2009) is based on the assessment on the context attributes of activities and another approach from Rozinat/Aalst's Decision Mining in ProM (Rozinat, 2006) provide relevant work with their approach on retrieving process decision points in ProM. In their paper, they use a well-known concept of decision tree mining to carry out a decision point analysis, to find out which properties of a case might lead to taking certain paths in the process. Crerie (Crerie, 2009) focusses on an explanation of the actual mining approaches, by setting out required data attributes, classification logic, and data preparation. The article also describes how the mining approach deviates from other rule

mining approaches like proposed by other research from Rozinat/Aalst in Decision mining in ProM (Rozinat, 2006). Whilst the Rozinat/Aalst analysis is made only on the process activities that have more than one successor activity, Crerie's proposal analysis the context attributes of each activity isolated, regardless of its relationship with other activities of the flow. We have taken these two different approaches to our research design. Bolt et al. (Bolt, 2018) present in this work a useful approach on how to compare business rules based on the process context. As stated, we will use an approach similar to Bolt et al, to apply it on two models for human decision point detection.

2.3.3. Semi-automated workflows, human actors and selection of a case process

On workflows and automated workflow systems is an abundance of literature available in various research disciplines like management science, computer science, engineering, or medicine. For our research, we want to limit the literature analysis and definition of workflows with just a high-level definition: *a digital and executable representation of any process can be seen as a workflow* (Barthelmess, 1995). Workflows can be fully manual, semi-automated, or fully automated.

Human actors in workflow processes have always been an interesting area for research. Also, here an abundance of literature can be found on workflow automation, the interaction between human actors and IT systems, and workflow optimization with human actors (e.g. Vanderfeesten, 2005). We want to limit our scope of analysis to focus on the semi-automated workflows with an expert role for the human actor. Automated workflows are generally considered as static prescribed sequences of steps; however, particularly semi-automated workflow processes are subject to human expert's choices intervening in the process course (Khodabandelou, 2013). Hence, this makes workflows with human tasks, subjective to variable process output.

We found different approaches to identify human behavior in systems. We evaluated the concept of *desired lines* in process models (Aalst, 2016): a process path that emerges through 'erosion' caused by the footprint of human actors in a system. The paths in the process model traveled most can be highlighted by using brighter colors or thicker lines. ProM's Fuzzy Miner could deliver such a model. However, we think desired lines will not bring sufficient proof of human decision points for a workflow process as our case process does not allow human actors to choose the shortest route. Another approach we evaluated came from case fraud detection which suggests creating a *social network* based on the event log with deviating cases. Aalst (Aalst, 2016) uses classification techniques to investigate non-conformance of deviating cases, where a resulting decision tree attempts to explain conformance in terms of characteristics of a case. This way one could find out that cases from certain customers handled by employee X tend to deviate. Similar to our ITSM workflow process, we could find out why certain ticket types are handled differently by different people. This would bring us closer to identifying human decision points, however, we think the ITSM process instances, that represent tickets, are too different to compare for deviating cases.

A third most interesting approach to behavioral detection comes from the research domain of *Online gaming*. Instead of identifying human actors in online gaming, the challenge is to detect automated *bot-players* from human players. Bots are autonomous computer programs, impersonating human players in an online game, that act in human-like behavior, to collect game credits illegally (Choi, 2016). Several studies (Choi, 2016)) provide proposals for log file data mining for *bot detection*, including methods of game character behavior, user observation, or network traffic analysis. Choi et al. (Choi, 2016) suggested an approach based on log file mining and behavior model comparison. We want to test if Choi's bot detection approach can be used to detect human decision points in a decision model from a semi-automated workflow.

Case process selection:
In search of an example process where a substantial amount of business rules is applied by human actors, we came across examples with a repeating pattern in which a human expert role is required to assess and judge the course of a process instance. We found an applicable example in an ITSM support workflow process, where support groups, ranging from 1st line low-expert dispatch groups to 2nd and 3rd line high-expert groups, all work on process instances (IT support tickets) that need to be resolved as soon as possible. Some parts of the workflow are automated, for example, automated email upon ticket creation or ticket closure, but most parts are steered by human tasks. The human aspect in this example process lays within the dependency of personal knowledge, skills, and judgment of individuals working on complex IT tasks. Retrieving the human decision points of such a process might allow for better process analysis and improvements, which can have a direct effect on the process KPIs, overall business process alignment, and in our sample case less business impact caused by IT issues.
In the next section, we describe the case process domain.

Background of ITSM Incident Management:
IT support organizations use ticket systems, which are workflow systems to streamline the swift support activities around IT issues and outages. A workflow supports the IT end-user in creating a ticket in an online portal and this ticket (process instance) will be routed and dealt with by various IT specialists. A typical lifecycle of an incident ticket in a ticket system is ticket creation, ticket assignment, ticket investigation, ticket management, ticket reassignment (optional), ticket resolution, and ticket validation (Salah, 2016). Different levels of IT specialists will review the ticket and try to solve the issue. International best practices like ITIL provide incident management processes and IM WFMS are often configured to support such workflows. Inefficient ticket handling due to human decisions can result in bigger IT issues and SLA compliance issues. The selection of this case process has a lot of human decision points that together will determine the quality of the process output (swift, accurate and sustainable repairs of IT services). Our research on human decision points focusses on an event log from such an IM WFMS.

Results and conclusion from the literature
Based on our research question, we found little literature on how to discover human decisions in a larger set of decision points, from a semi-automated workflow. To propose an automated mining approach for this question, we have found various approaches in the field of process mining. Aalst (Aalst, 2016) provided us a generic 5-step research approach for process mining research. From the research from Leemans (Leemans, 2014)(Leemans, 2017) we derived an inductive mining approach, and from Maggi (Maggi 2012)(Maggi, 2013), we derived a declarative mining approach. To focus on decision mining, we used Rozinat’s decision mining approach (Rozinat, 2008). For identifying human features in a model, we used Choi's bot player detection approach (Choi, 2016), and lastly, Bolt provided us with a model comparison technique (Bolt, 2018). All research steps (chapters 4) will be performed in the tool ProM. This gives us sufficient background to approach our research problem: how to detect human decision points in a semi-automated workflow.

2.4. Objective of the follow-up research

The objective of our follow-up research is to follow a 5-step approach, where we apply process mining techniques on an event data set to retrieve human decision points in a workflow, in an automated way.

3. Methodology

We approached this research with experimental research of process mining techniques, to apply it on a single case study to discover human decision points in a semi-automated process. With this research, we want to explore a broader applicable example of discovering the more *subjective* human decision points in a process and to provide a multi-perspective model of a workflow process, which highlights the human actor's decision points.

3.1. Conceptual design: selected research methods

The conceptual design of our research builds upon a model comparison to identify human behavior at decision points. On one hand, we have the process mining and decision mining techniques, on the other hand, we have a game bot behavior detection technique from a near research field of data mining in game logs. From both streams, we can create models that are comparable and should allow for model comparison on human behavior in the decision models. To structure our research steps, we will follow Aalst's (Aalst, 2016) *5-step mining approach* outlined below. In step 4 we come to the model comparison. The case process has been selected in section 2.3.3.

A brief overview of our 5 mining steps: first step, we start with data collecting, cleansing, and formatting of our case process event data. The second step is mining the workflow process and third step modeling are resulting in two process models (a procedural Petri net model and a declarative process model). In the fourth step, we focus on two model extensions: via decision point mining we generate an overall *decision model* of our case process and a second extension is manually created *behavior model*, based on a proposed technique from game bot detection in online gaming (Choi, 2016), which we will use to mine for unique human behavior at decision points. Finally, in the fifth step, we combine the results of step four with the process models of step 3, to create an integrated multi perspective model that will highlight human decision points within the process. The next sections are structured accordingly:

1. Step 1 – Data collection, cleansing, and formatting
2. Step 2 – Process mining
3. Step 3 – Creating process models
4. Step 4 – Model extensions: a decision model and a behavior model
5. Step 5 – Model comparison and final integrated multi perspective model

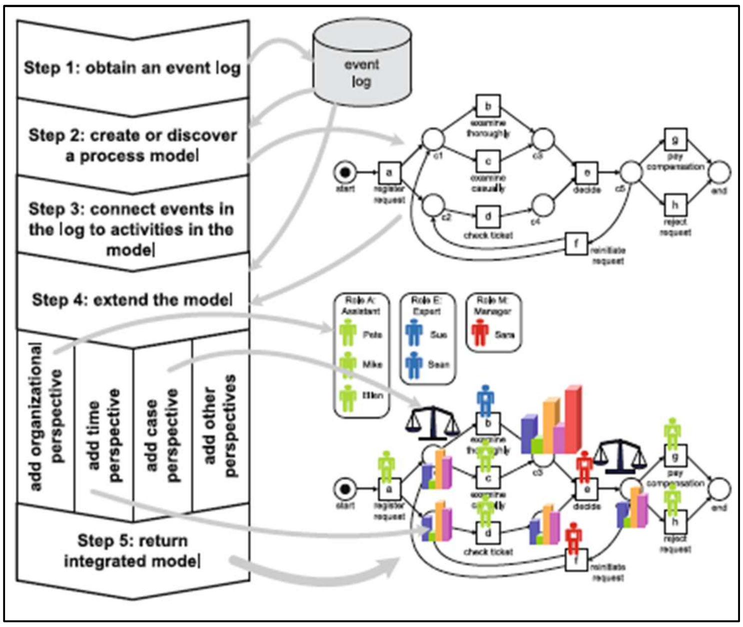

Figure 2: a practical 5-step mining approach according to Aalst (Source: Aalst, 2016)

3.2.Technical design: elaboration of the methods

Step 1 - Data collection, cleansing, and formatting
The dataset is retrieved from a cloud-based ITSM workflow system named ServiceNow (www.servicenow.com). Via a custom query, we retrieved an initial data set with 6 months (May to Nov 2019) of IT ticket data. The query was set to only retrieve a limited set of ticket attributes (unique ID, status field, timestamp, business service, etc). Flat text interpretation of ticket content was not part of the mining scope, we omitted the ticket subject and ticket content conversations. We retrieved ticket status changes and lead time per ticket state.

sys_created_on	definition	id	value	start	end	duration	calculation_(	business_service	application_ci
16/05/2019 11:15	Incident State Duration	Incident: INC0142537	Resolved	16/05/2019 11:15	19/05/2019 12:00	261889	TRUE	End User Computing Services	Desktop
08/05/2019 11:27	Incident State Duration	Incident: INC0141337	New	08/05/2019 11:27	08/05/2019 11:27	19	TRUE	End User Computing Services	Mobile
27/05/2019 11:00	Incident State Duration	Incident: INC0136146	Closed	27/05/2019 11:00		0	FALSE		
08/05/2019 16:24	Incident State Duration	Incident: INC0141433	New	08/05/2019 16:24	08/05/2019 16:25	69	TRUE	Supply Chain Services	SAP Source to Pay (SRM)
06/05/2019 15:19	Incident State Duration	Incident: INC0140825	Resolved	06/05/2019 15:19	09/05/2019 16:00	261622	TRUE	Collaboration Services	Exchange
30/05/2019 12:33	Incident State Duration	Incident: INC0144690	Resolved	30/05/2019 12:33	02/06/2019 13:00	260812	TRUE	Network Services	WAN - AU
17/05/2019 09:58	Incident State Duration	Incident: INC0143009	New	17/05/2019 09:58	17/05/2019 09:59	29	TRUE	End User Computing Services	Desktop
21/05/2019 09:35	Incident State Duration	Incident: INC0143531	Awaiting User I	21/05/2019 09:35	21/05/2019 14:09	16469	TRUE	End User Computing Services	Desktop
03/05/2019 11:49	Incident State Duration	Incident: INC0140234	Active	03/05/2019 11:49	06/05/2019 11:45	258992	TRUE	CRM Services	Fontem SFCC

Figure 3: a snapshot of the data structure of the CSV (Excel)

Data cleansing:
Data cleansing was performed to remove traces of half-finished and half-started cases, which we exclude as we want to investigate only cases with a full lifecycle, from ticket creation to ticket closure. Also, other incomplete or faulty cases (e.g. missing fields) are removed from the set.

Data formatting:
The resulting log will be formatted into a XES event log (eXtensible Event Stream), which is a widely accepted standard (XES, 16) for event logs and often required as an input format for mining algorithms in ProM (Aalst, 2016).

Step 2 - Process mining
Prom 6.9 (ProM Framework, 2019) is used as a process mining tool. Our selection of appropriate process mining algorithms needs to be substantiated: for this Step 2- process mining, we evaluated several types[1] of discovery mining algorithms, recently benchmarked by Augusto et al. (Augusto, 2019). To give context, most of the evaluated process miners are *procedural* algorithms, as opposed to *declarative* or *hybrid* algorithms, and most miners produce Petri net *process models* (as opposed to *BPMN models*, *process trees*, *causal nets* or other model formats). Petri net is still the predominant process modeling language (Augusto, 2019). Process mining algorithms are extensively compared in (Aalst, 2016)(Augusto, 2019) and (Garcia 2019) and Inductive Miners (IM) and Heuristic Miners (HM) are most commonly used (Augusto, 2019, Garcia, 2019). Many other mining algorithms specialize or to solving specific limitations of IM and HR, or focus on special mining goals. For our initial process mining step, we will use the Inductive Visual Miner, which is an efficient miner with good *model fitness*, and *precision*. The result of Step 2 is a procedural Petri net model. As a side activity, for comparison and fault checking, we also generate a Petri net from a heuristic miner from Mannhardt's *interactive Data-aware Heuristic Miner* (iDHM)(Mannhardt, 2017)(Mannhardt, 2018). Nonetheless, we will use the IVM Petri net in the following steps.

[1] Alpha miner (2002), Inductive Miner (IM)(2013), Evolutionary Tree Miner (ETM)(2014), Fodina (FO)(2017), Structured Heuristic Miner (S-HM)(2003), Hybrid ILP miner (H-ILP)(2009) and Split Miner (SM)(2017).

Step 3 - Creating process models
Previous step 2 has generated the basic procedural description of the discovered process, in the form of an IVM Petri net model. Alternative models of the same process will lead to better insight (Aalst, 2016). We are interested in the unstructured nature of our human ITSM workflow process, and we bear in mind that the procedural IM miner is less suitable for unstructured processes and may produce unreadable spaghetti-like diagrams. As we are looking for process rules and decision point, we are interested in any rule or constraints on the process. Therefore we apply a *declarative* process mining algorithm, which is better suitable for unstructured processes and which will give us an alternative process perspective based on process constraints (Maggi, 2013). For a declarative miner, any unstructured process is allowed as long as it does not violate a rule. In this way, a declarative model defines a *control-flow* (Maggi, 2013) of given constraints and the mining result is a declarative model that specifies a set of constraints/rules that have been found during process execution. Various declarative approaches (Augusto, 2019) are available, but we highlight an approach from (Maggi, 2013), for a *data-aware* declarative approach where the resulting model gives more insights into the role of data in the execution of rules of a process. The importance of data in business processes, particularly dynamic/unstructured ones, is that often data drives the decisions that participants make. The result of this step 3 is a Declare process model, created with Maggi's Declare Miner in ProM (Maggi, 2013).

Step 4 - Model extensions: creating a decision model and a behavior model
In step 4 we focus on the creation of two specific models for model comparison.
First, we want to mine a *decision model* from our event log and secondly, we want to create a *behavior* model. The first model, the decision model, will give us a set of *all* decision points in the process, both automated decision points as human decision points. To generate this decision model, we will use a decision miner algorithm from Rozinat (Rozinat, 2006)(Rozinat, 2008), briefly introduced in the next paragraph. A second model, the behavior model, is needed to capture unique human behavior attributes from process attributes and data attributes, and compare it with the mined decision points. This behavior model we create similarly as to Choi’s bot detection approach (Choi, 2006), see next paragraph Step 4.2. Choi’s approach can generate a behavior model from several behavior feature-sets and identifying attributes based on extensive earlier research (Bartle Test of Gamer Psychology). For our research goal, we did not find a comparable extensive source of human decision point discriminators that we could use to build a behavior model, therefore we will create the behavior model manually in the next section. Below we briefly describe the two model creations in step 4.1 and step 4.2:

4.1 Creating a Decision model with the Rozinat miner in ProM:
We use the Rozinat miner in ProM (Rozinat, 2008) to generate the decision model. Rozinat’s decision miner works on the principle of converting every decision point into a classification problem, where the classes are the different decisions that can be made. The Decision Miner formulates the learning problem and the analysis is carried out by a decision tree classifier algorithm (J48, Rozinat, 2008). Some background on decision mining from Rozinat explains that for any process transition that has *more than one* outgoing transition, a *decision* needs to be made: only one of the outgoing transitions can be executed. Such states are called as *decision points* (Bolt, 2018). For any decision point, the context information contained in the event log, i.e., data attributes, can be used to discover these decision points. *Decision trees* can be created to discover data conditions at the decision points (Aalst, 2016)(Bolt 2018). Also, additional *predictor* attributes can help to make the decision miner algorithms more data-aware. Aalst (Aalst, 2016) and others (Crerie, 2009) refer to Rozinat for a decision mining approach we also will use. Rozinat (Rozinat, 2008) approach suggests converting every decision point into a classification problem, where the classes are the different decisions that can be made. The approach is called data-aware as it assumes that all attributes that have been written before the considered choice, are relevant for the routing of a case at that point. To solve such a classification there are various decision tree algorithms available such as J48, which were popular inductive inference algorithms at the time of Rozinat’s publication (2008), as they include methods to avoid overfitting data, to prevent that

the decision tree is over-tailored towards the training examples. Also, more recent work on decision mining from Bolt (Bolt, 2018) uses decision trees to discover data conditions at decision points. Bolt extended the approach to compare different decision trees for business rule comparison, something similar to what we will do in step 5, for which we will use Bolt's comparison techniques. The result of this first part of step 4 is a decision model created in ProM with the Rozinat Decision miner. The miner input will be again our XES event log and will result in an *Evaluation Result output* in ProM.

4.2 Creating a behavior model based on Choi:
The second part of step 4 is the creation of a *behavior* model. A behavior model describes unique discriminating features and must be able to use in a comparison technique. The challenge is to define characteristics for human decision points and/or characteristics for non-human decision points, to differentiate but also to find a comparable format (model) to use it in model comparison techniques. Once we have defined a set of detection criteria via the behavior model, we can work through model comparison techniques like from Bolt (Bolt, 2018) in step 5, to highlight differences between two models of the same process, potentially revealing human decision points.

In essence, our adoption of Choi's works is as follows: discriminating behavior of automated processes, e.g. game bots, needs to be identified first, to create a *profile* or *behavior model*. For example, discriminating behavior can be that game bots have much *higher frequencies of particular actions* than those of known human players. This way a *behavioral model* can be created, based on these and other discriminating attributes that successfully detected game bots via behavioral patterns.
Other discriminating behaviors in Choi's bot detection are 'battle', 'collect' or 'explore' behavior, for which we will use behavior attributes of actors in our semi-automated workflow processes. We extract attributes that characterize the human actor's behavioral features by analyzing the behavior. These attributes are combined to be *feature sets* for user groups. Groups are created via classification and Choi used an a priori algorithm for association analysis, which results in a model (frequency table).

Attributes	Hunting	Attack	Hit	Defense	Avoidance	X	Y	DirX	DirY	HP
Party	44.6	47.5	47.52	58.44	55.09	48.93	48.89	48.89	48.86	35.18
Money	62.74	67.84	67.91	81.69	78.14	72.07	71.98	72.06	71.97	59.12
Exp	62.61	67.64	67.69	81.51	78	70.74	70.65	70.73	70.65	58.57
MP	58.61	62.98	63.07	76.52	72.99	67.84	67.75	67.83	67.75	56.22
HP	44.68	47.45	47.54	56.3	54.96	52.1	52.02	52.13	52.05	–
DirY	56.92	61.04	61.12	68.96	67.03	71.63	73.11	71.77	–	–
DirX	56.97	61.08	61.16	69.03	67.1	73.19	71.63	–	–	–
Y	56.91	61.04	61.12	68.95	67.02	71.81	–	–	–	–
X	56.96	61.08	61.16	69.03	67.09	–	–	–	–	–
Recovery	45.51	48.41	48.48	58	55.71	–	–	–	–	–
Avoidance	60.7	64.82	64.88	77.52	–	–	–	–	–	–
Defense	62.23	67	67.12	–	–	–	–	–	–	–
Hit	55.49	67.6	–	–	–	–	–	–	–	–
Attack	55.44	–	–	–	–	–	–	–	–	–

Figure 4: Example of Choi's frequency table frequent with items from the status and behavioral data

Like in Choi's approach, we will cluster all actors (automated and human) into groups according to behavior style. Human detection is then performed on each *style* group. *Actor* group classification is performed using K-means clustering. Choi's bot detection was using a support vector machine (SVM) with each clustered player style group.

Now, important to mention is that the identification of discriminating attributes can be both manually and semi-automated. For bot detection, Choi's used proven discriminators from earlier research (Bartle Test of Gamer Psychology, 2004), but also other discriminators were added manually by Choi into a behavior model. In total 50 data attributes were available for Choi's analysis. As we did not find a useful set of proven discriminators for human behavior in semi-automated work flow s, we decide to create our behavior model manually. To create a behavioral model, we will extract process attributes and data attributes that characterize human behavior and collect these attributes in groups of *features* (Choi, 2016). These features are combined to form *feature sets* for human user group classification and used for automated process group detection. The result of this second part of step 4 is a manually created behavior model in the form of a decision tree, that can be used for model comparison in step 5.

Step 5 – Model comparison into one integrated multi perspective model

In this final 5th step we want to compare two models and create one integrated multi perspective model (Aalst, 2016). For model comparison, we select a similar approach as proposed by Bolt et al. in *Process variant comparison* (Bolt, 2018). Bolt et al. have demonstrated a working implementation of a process comparator in ProM, with which process variants can be compared to each other to find similarities and differences. Bolt's method uses two decision trees to compare. To use this approach to compare our models, we need to introduce the measurements used for comparison. With the *Process Comparator* algorithm, measurements are collected from the event log and annotated on a transition system that represents the behavior of both variants. The annotations of each variant are compared using statistical significance tests to detect relevant differences. This technique should be able to detect if one variant is significantly different from the other, even if they have the same control-flow, according to Bolt. The output will be the Process Comparator panel result, similar to the below sample picture.

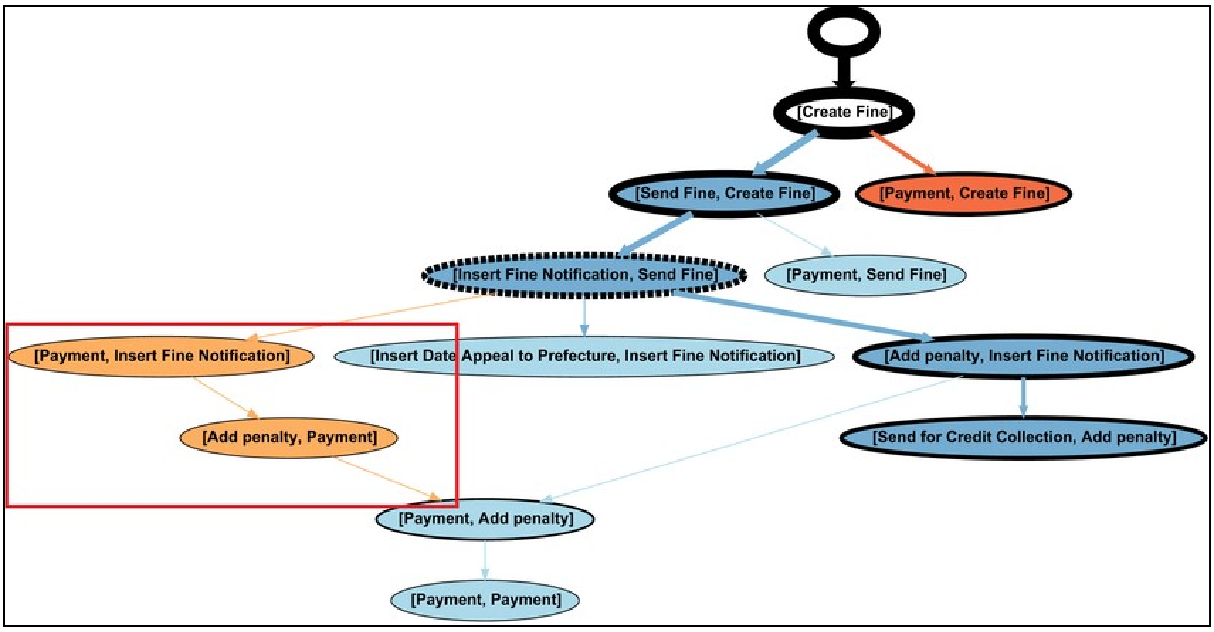

Figure 5: Illustrative example of Bolt's process comparator result, highlighting process comparison differences (Source: ResearchGate)

Lastly, we combine the results from step 3, step 4, and step 5 into multi perspective models.

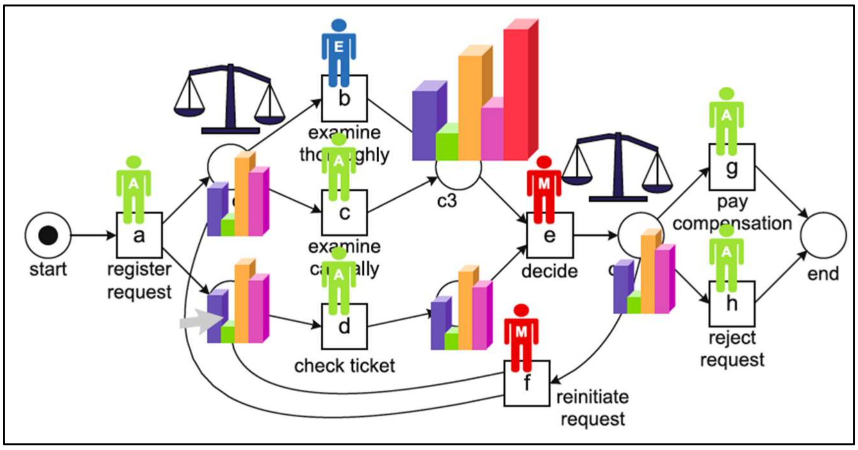

Figure 6: Illustrative example of multi perspective process model
(Source: Aalst, 2016])

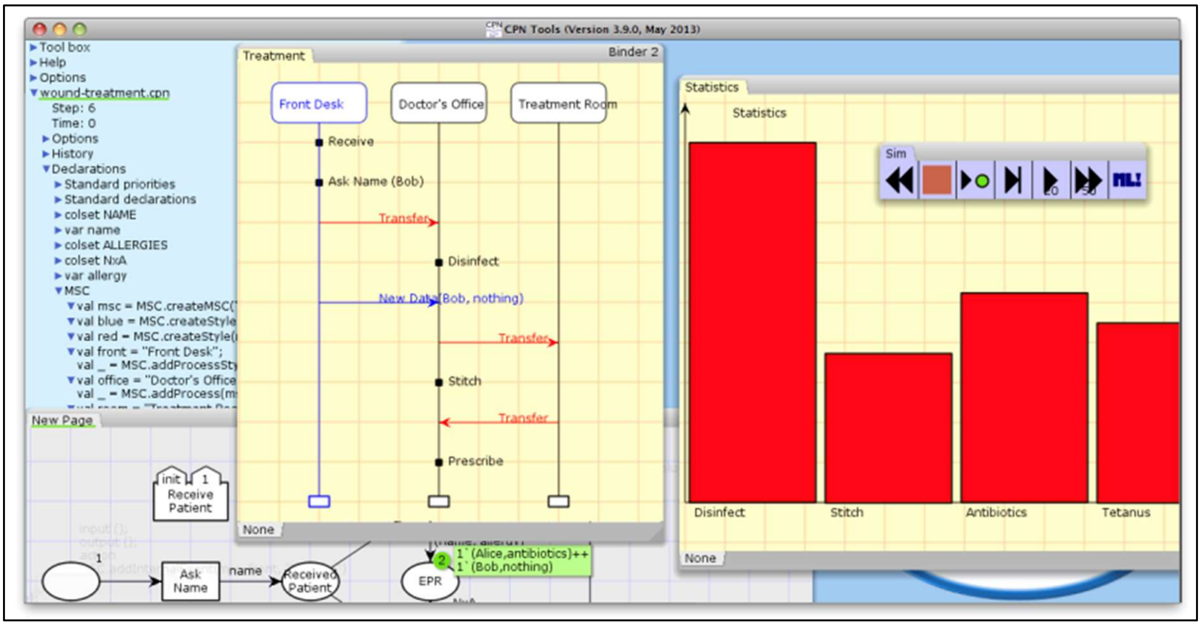

Figure 7: Illustrative example of a Colored Petri net created with CPN Tools (Westergaard, 2013)
(Source: www.cpntools.org)

3.3.Data analysis

The next section is structured according to the 5 steps.

1. Step 1 – Data collection, cleansing, and formatting
   The initial data set is retrieved from a cloud-based source system via a no-SQL query to a raw CSV output file. For data cleansing, we used R-studio to remove incomplete or faulty cases.
   We formatted the CSV into a XES event log (Ref) via CSV-to-XES plugin in ProM.
   The output is a XES event log file.

2. Step 2 – Process mining
   The XES event log file is used as input for the Inductive Visual Miner in ProM.
   This inductive miner will the heuristic miner will create both a Peri net model in PNML file format.
   The IVM Petri net model will be further used.

3. Step 3 – Creating process models
The same XES event log file is used as input for the Declarative miner in ProM.
This declarative miner will create a Declare model in the output of a Declare file.

4. Step 4 – Model extensions: a decision model and a behavior model
The same XES event log file is used as input for the Rozinat Decision miner.
The output of the decision miner is a decision model (file format) and we need to convert the model into a re-engineered event log via a ProM plugin as the output of this decision model needs to be again in Xlog format event log, in order to be able to use it in step 5's process comparator.
The Behavior model is manually created. We use the ProM plugin 'Text based Petri Net' to first create a Petri net that has typical human behavior and then converts the model back into a re-engineered event log again. The output of this behavior model also needs to be in a Xlog format event log, in order to be able to use it in step 5's Bolt process comparator.

5. Step 5 – Model comparison and final integrated multi perspective model
The two re-engineered Xlog event logs from step 4 are used an input for the process comparator in ProM. The output will be presented in the Process Comparator panel and we select a Petri net diagram as most descriptive.
Lastly, we will combine the two Petri nets from step 2 and step 5, which should present a new integrated model with a certain degree of overlap/match and high light the human decision points in the process. The Declare model is used as additional multi perspective information but will not be further integrated, as per intrinsically different model format.

Note that the data is being analyzed, mined, and represented in various forms. Also, we use reengineering steps to translate models back to event logs to use in next steps. This gives a risk to data quality and validity. We address these as much as possible in next chapter 4. Results.

3.4. Reflection with regards to validity, reliability and ethical aspects

A general remark on validity and reliability can be made about process discovery and rule modeling. As Aalst (Aalst, 2016) states, any representation of a process captured in a process model notation, will suffer from *representational bias*, as the expressiveness of the discovered model is limited by the abstraction of the process model notation. Next to that, our method tries to pull different mining techniques from process mining into the field of decision mining; this makes this research highly experimental and the overall impact and generalization of this research need to be considered within the boundaries of its scope; it is a single case study on sample data from one workflow process. The result describes statistically existing human behavior but we need to be aware that there can be lots of other human factors that contribute to human decision point detection (false positives). Our data set can be considered reliable as the data is from a substantial (6 months) period, containing sufficient live examples. The used ProM algorithms can be considered reliable as they all have their proven theoretical backing in literature studies and practical implementations as ProM plugins. However, the consecutive use of result sets from one algorithm as input for another introduces a risk of interference and magnifying noise. Further limitations of our methods will be addressed in chapter 5. From an ethical standpoint, we did not identify major ethical risks. We altered the unique ticket IDs from the source system, to not be able to relate our data to real-life tickets and any employees in the actual source system.

4. Results

This chapter describes the activities and results of the proposed mining steps. First, we briefly describe the data set characteristics data cleansing and data preparations. Then we will describe the main mining steps of this research:

1. Step 1 – Data collection, cleansing, and formatting
2. Step 2 – Process mining
3. Step 3 – Creating process models
4. Step 4 – Model extensions: a decision model and a behavioral model
5. Step 5 – Model comparison and final integrated multi perspective model

4.1. Step 1: Data collection, cleansing, and formatting

Data collection

Via a custom query, we retrieved an initial data set with 6 months (May to Nov 2019) of IT ticket data. The query was set to only retrieve a limited set of ticket attributes:

1. Unique ticket ID (id)
2. Date/time created (sys_created_on)
3. Business service (business_service)
4. Business application (application_ci)
5. Ticket status (value)
6. Ticket status start date/time
7. Ticket status end date/time
8. Ticket status duration in seconds

Data collection from the sample process was done via a custom extraction query (no-SQL) on the IM ServiceNow application database (cloud-based), creating a representation of an event log by retrieving all states and timestamps of IM tickets during its lifecycles. The dataset contains 6 months of ticket data in 136.185 event records, containing 39.499 events (ticket status changes) 19039 unique tickets IDs.

This resulted in a CSV data set (20 MB) with 136.185 event records. The data set contained 19039 unique ticket IDs.

Data cleansing:

The dataset was investigated and cleansed using R-Studio. The dataset contained 136.185 event records, 17 records had no unique ticket id so it could not be matched to any case (ticket) and were removed. 19039 unique ticket id were found. Subsequently, looking for cases with a complete lifecycle, so tickets with both status 'New' to 'Closed', only 11746 unique cases (of total 19039 unique cases) had a start event 'New' and 'Closed' event. This can be explained by the design of the query, where the remaining 7293 cases are event records of traces of tickets that were already halfway their life cycle on the moment of querying. To create an initial dataset with only fully completed cases, we choose to omit these 7239 incomplete traces. From the 11746 unique fully completed cases, we derived a total of 45131 event records. This subset of 45131 event records has been the initial data set for process analysis. Furthermore, two irrelevant columns (‘definition’ and ‘calculation’) were removed, the aforementioned eight columns remain.

Data formatting to XES conversion:
The cleansed CSV data set of 45131 full process events has been loaded into ProM 6.9 for CSV to XES conversion (XES Standard, 16). For XES conversion, the case attribute 'id' is identified manually and the event attribute 'value' with various ticket process states, is identified manually. After conversion, the same number of cases (117373) and events (45131) were available in the XES (ProM's XES format is Xlog), so no loss due to conversion.

4.2. Step 2: Process mining

As per our proposal in section 3.2, we use the Inductive Visual Miner from Leemans (Leemans, 2014) for the initial process discovery and a heuristic miner for comparison. We selected the Inductive Visual Miner (IVM) (Leemans, 2017) as plugin in ProM 6.9. According to Leemans, the IVM is an inductive miner with a strong focus on performance for *soundness* and *fitness* and with an acceptable processing time (Leemans, 2014), meaning that it can produce sound models in acceptable computer processing time; we used a Windows 7 PC with a 2GHz i7 processor and 8 GB RAM for all research steps. The ProM IVM plugin supports different IM algorithms: IM infrequent (IMF), IM directly-follows (IMD), IM infrequent—directly-follows (IMFD). Also, to avoid overfitting (Aalst, 2010), resulting in very complex and spaghetti-like process models, IVM process discovery algorithms can use clustering and variant control techniques like noise filtering, activity filtering, and many more. This does, however, affect potentially the theoretical 100% fitness of the models.

We mined our XES event log with default IVM settings (default IM-infrequent miner)(activity filtering fraction=1; means no filtering)(default noise filter factor = 0.8 which corresponds to 1-0,8 = 0,2 noise filtering). The resulting Petri net model looked as per our expectation, typically the model reflects that new instances (new tickets) are *created*, *resolved*, and *closed*. Some process deviations are detected, as expected, where a small number of process instances got in exception states like *Awaiting 3rd party support'* or *Awaiting problem analysis*. The result of a process discovery algorithm can be evaluated considering quality dimensions like soundness, fitness, simplicity, precision (Leemans, 2017), and generalization (Aalst, 2016) and many more according to Rozinat (Roziniat, 2008). Due to the relative simplicity of our event log, our model scores well on fitness, precision, and simplicity, so overall soundness was good. We exported the Petri net model as an EPMNL file for re-use (step 2).

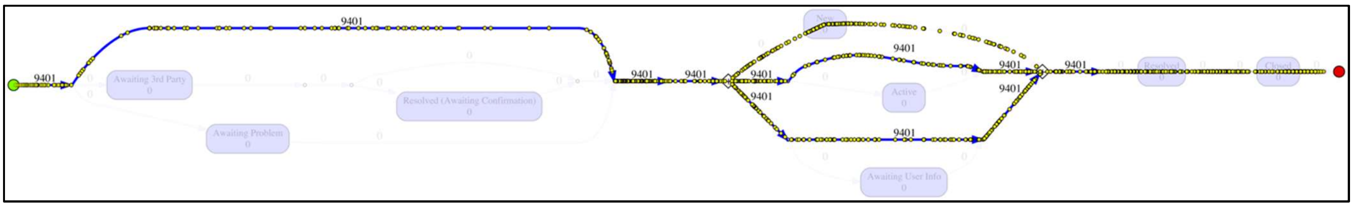

Figure 8: Step 2 mining result from default IM miner process model

Heuristic miner:
Heuristic mining algorithms take frequencies of events and sequences into account when constructing a process model e.g. infrequent paths can be suppressed in the model. Both the representational bias provided by causal nets and the usage of frequencies makes the heuristic approach much more robust than most other approaches (Aalst, 2016). So-called 'noise' from infrequent paths can be filtered out. However, Leemans (Leemans, 2014) states in that frequency-based techniques such as the heuristic miner do not guarantee the full soundness nor fitness as Leemans's IVM, as the heuristic approach can filter out infrequent paths that affect a 100% compliant HR model.

Nonetheless, we selected Mannhardt's *interactive Data-aware Heuristic Miner* (iDHM) as HR miner in ProM 6.9 (Mannhardt, 2017) as the iDHM miner is a recent heuristic miner that tries to reduce the noise filtering problem. The data-aware iDHM miner reduces the risk of filtering out infrequent but *relevant* process paths. Similar to the previous visual-oriented IVM miner, iDHM also gives an interactive exploration of (heuristic) process discovery. Also, data-flow and control-flow of the process are discovered together as some infrequent behavior may be characterized by very deterministic rules, and thus be of great interest to our topic of rule detection later (step 2).

We mined the same XES event log with iDHM miner with default settings (dependency miner=flexible HR miner; dependency factor=0.9)(conditional miner=C4.5 F1-score; conditional factor=0.5)(binding miner=nearest activity FHM; binding factor=0.1)(frequency factor=0.01)(all tasks connected=yes). We display the result in a Directly-follows model from iDHR, as it displays instance numbers which gives a notion of high-frequency paths (process highways) and low-frequency paths, for better comparison with the IVM result figure 9.

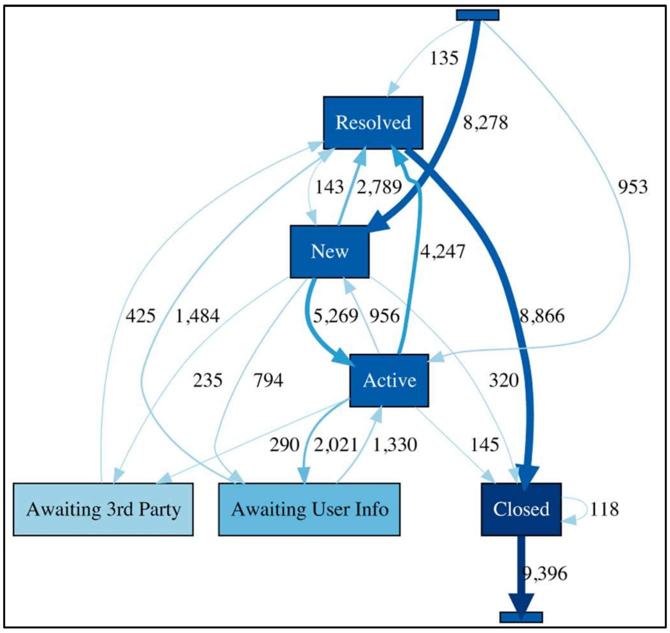

Figure 9: Step 2 mining result from default iDHR miner Petri net model

Also for the result of the heuristic process discovery algorithm is in line with real-life expectations. Also, for the iDHR miner, similar quality dimensions like soundness, fitness, simplicity, and precision (Leemans, 2017) apply. As per the design, filtering out infrequent paths would lower the theoretical fitness compared with IVM. Again, using a mitigating variant like the iDHR and again the relative simplicity of our event log, also this iDHR generated model scored well on fitness, precision, and simplicity, so overall soundness was good.

4.3.Step 3: Creating process models

Our focus is now on rule mining which is about finding and substantiating decision points, however, despite the result of our previous process discovery step, there are limitations in process models for supplementary discoveries related to identifying and generalizing *decision points* (Garcia, 2019). De Smedt (De Smedt, 2016)(De Smedt, 2019) confirms the limitations of just using mined process models as input for rule mining and suggest to ensure a good *fit* between the input data, the desired result format and the decision mining technique used. De Smedt et al. present a decision mining quadrant and chart to support on the best approach.

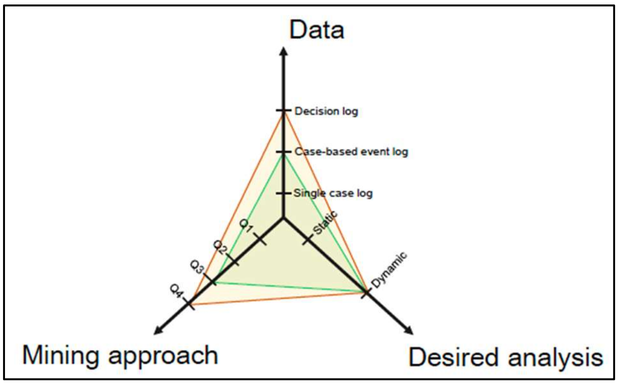

Figure 10: De Smedt et al. 2016, Positioning data, mining approach, and desired analysis.
(Source: De Smedt, 2016)

A couple of new terms need to be introduced for business rule mining: constrains, control-flows, decision points, DMN, decision trees, and decision mining.

For decision mining, we will be focusing on process states, transitions, constraints, and attributes that steer this decision making. Transition systems are the most basic process modeling notations and a transition model consists of *states* and *transitions* (Aalst, 2016) which are often translated to higher-level languages such as Petri nets, BPMN, and UML activity diagrams. Any split (XOR) in a transition model reveals a decision point, where some constraint must be met. (Aalst, 2016). Process constraints take process attributes as parameters and can be used to determine the process behavior of states and transitions (Bolt, 2018). The process constrains will control the process, which is just one perspective, control flow, of looking to a process.

Control-flows, control-flow perspective, and other perspectives:
With decision mining, we want to approach the process model (the result of step 1) with a certain *perspective*, namely with a focus on how the process constraints, decision points, and rules form together with the process business rules. Most processes mined Petri net and BPMN models usually describe a *control-flow perspective*, describing the ordering of activities and possible paths. But there are other perspectives too, like a *data-*, *functions-*, *resources-* or *organization perspectives*(Bolt, 2018). Aalst (Aalst, 2016) lists four perspectives: *control-flow-*, *organizational-*, *case-*, and *time-perspective*. More advanced models can also present a multi-perspective view on a process (step 3). For this step 2 on business rule mining, we will focus on the control-flow perspective of our case process.

Decision Mining in Prom:
Rozinat (Rozinat, 2006) presented a data-aware decision mining approach with the aims at the detection of data dependencies that affect the routing of a case. Besides data attributes, resource information, and timestamps, even more, general quantitative (e.g., key performance indicators like waiting time derived from the log) and qualitative (i.e., desirable or undesirable properties) information could be included in the analysis if available. Rozinat has provided a Decision Miner for the ProM framework. The Decision Miner plug-in determines the decision points contained in a Petri net model and specifies the possible decisions for the log while being able to deal with invisible and duplicate activities in the way described in Rozinat, 2006.

4.4. Step 4: Model extensions: creating a decision model and a behavior model

The decision model is created using Rozinat's decision miner plugin in ProM
('Rozinat Decision miner Log on Petri net', created by Seppe van den Broucke, KU Leuven, BE). The miner takes as input the XES log, the Petri net PNML from Step 2, and a marking file (created with the 'create

initial marking' plug-in to add an initial marking). We mined the model with default miner settings: classifier = XES 'event name' which relates to the ticket status changes, algorithm J48 for an unpruned C4.5 decision tree, and other settings disabled. Although figure 8 shows our initial mining result in the Model view of the Decision Miner, which provides a visualization of each decision point of the Petri net, the miner was unable to generate a decision model, due to the lack of sufficient process attributes in the data set to learn the decision tree.

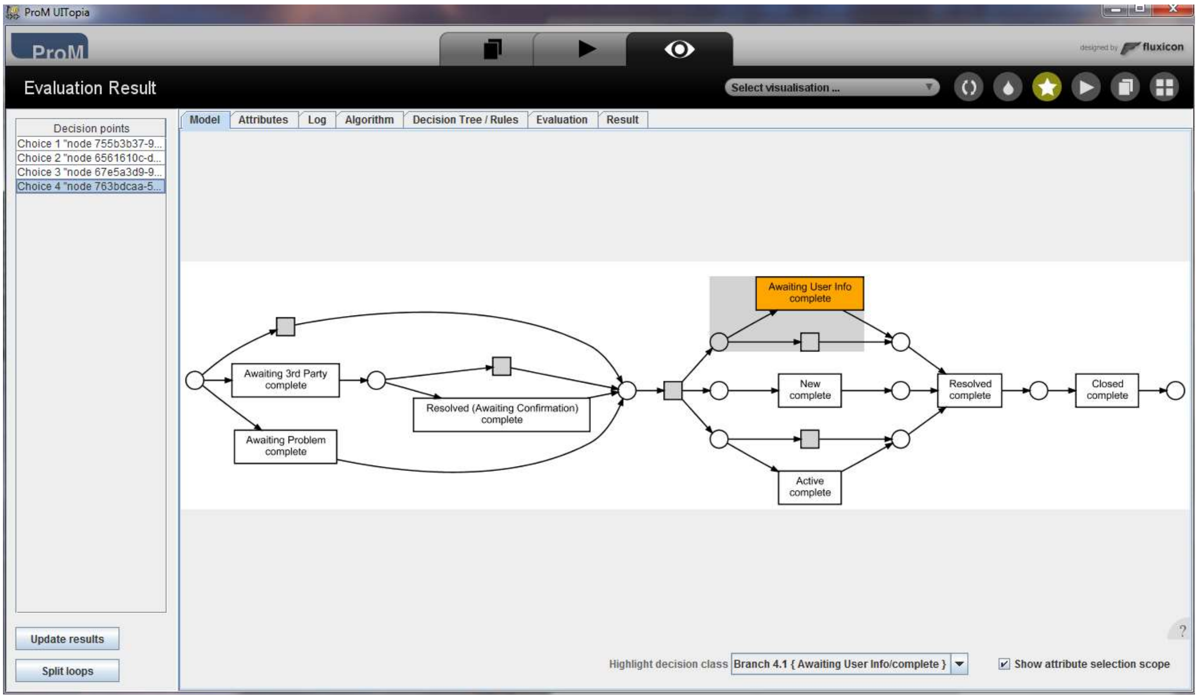

Figure 11: Step 4 result from Rozinat decision miner

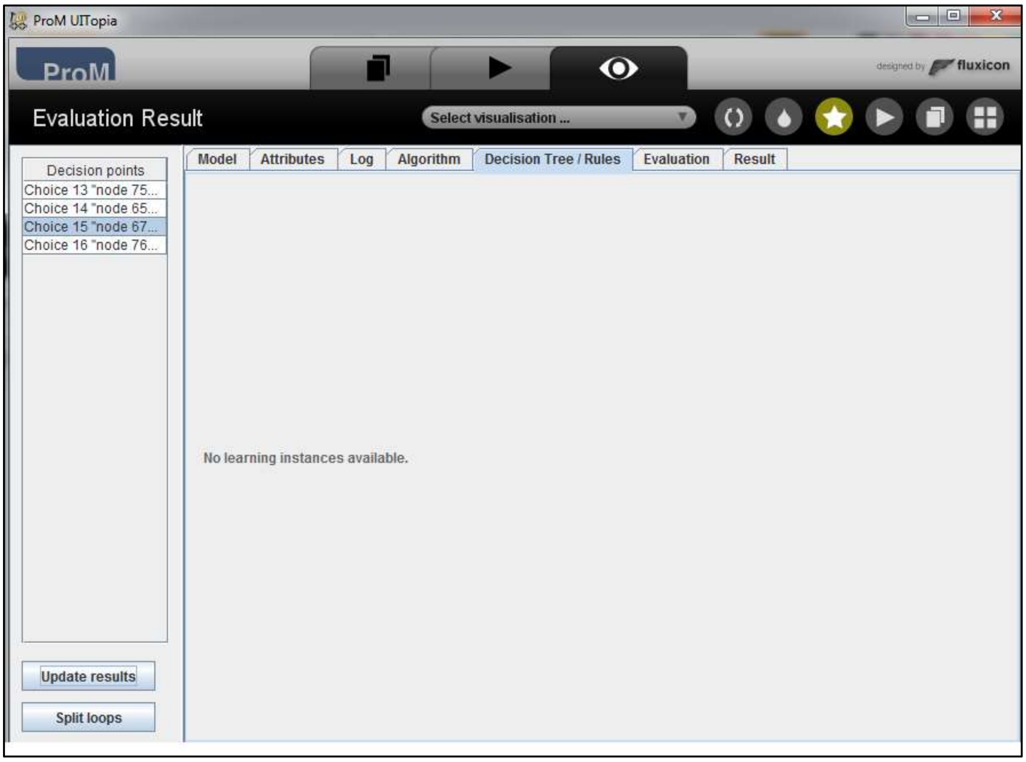

Figure 12: Step 4 result from Rozinat decision miner result (unsuccessful)

Creating a Behavioral Model:
Choi (Choi, 2016) created a behavior model in a semi-automated way by using proven discriminating features to identify human vs. bot behavior, based on available literature. We needed to create the behavior model manually, as we did not find a useful set of behavior identifiers for human decision points from previous research. To still test our conceptual design in this step 4, we created a behavior model manually. Using Choi's approach, we simulate frequent items from the status and behavioral transaction data: the features are selected to represent categories of behaviors commonly observed: e.g. behavior 1, behavior 2, and behavior 3. We established a new feature set based on the selected features and these features characterize workflow process behavior styles 1, 2, 3. This way, all decision points are clustered into groups according to workflow process behavior styles 1, 2, 3. Human actors are classified from a *reference set*, as described in Choi's. For Bot detection that would be the reference set of the Bartle Test of Gamer Psychology (Choi, 2016). Bot detection is then performed on each style group (Choi's approach uses a player group classification performed using K-means clustering. Bot detection was done using a local support vector machine (SVM) with each clustered player style group).

4.5. Step 5: Model comparison and integrate result in a multi perspective model

For model comparison, we use Bolt's Process Comparator plugin in ProM (Bolt, 2018).
Aim to detect relevant differences and the plugin uses transition systems to model behavior and to highlight differences. Like Bolt, we use two decision trees, produced in step 4, to discover data conditions at the decision points. For this, the process comparator creates observation instances, training, and testing the decision trees to obtain rules and compare decision trees to identify differences. Input for this step will be an event log array, one array containing event log traces from two processes: one for the original ITSM process and one trace that is artificially created, highlighting the human characteristics. We have no results from this step (see for illustrative purposes Figure 4 in section 3.2, Step 5).

The final multi perspective model:
The process model of step 2 is extended with the additional perspectives created in step 3 and step 4. The new multi-perspective model gives a holistic view (Aalst, 2016) and the new perspectives give a better understanding, potential new ideas for process improvement, and can be used as input for next ProM steps, for example, to generate a colored Petri net that can display various combinations of control-flow, data flow, decisions, routing probabilities, etc. thus capturing all aspects relevant (Aalst, 2016).
In this final result, we show a Colored Petri net, created in CPN Tools (www.CPNTools.org) (Westergaard, 2013). We have no results from this step (see for illustrative purposes Figure 5 and Figure 6 in section 3.2, Step 5)

5. Discussion, conclusions, and recommendations

5.1. Discussion

Initially, we explored the role of human actors in an unstructured semi-automated workflow process and the relevance and importance of having visibility and understanding where human actors apply decisions during the process. Human decision points can be hard to grasp and sometimes be subjective as different human actors may steer the process and its content in slightly different ways. For process quality and process governance, this human-generated process variance is a valuable key performance indicator for process owners and management. However, detecting and filtering out human decision points can be challenging. A good behavior model needs to be created, tailored to human behavior of the process domain, and the process itself. The small differences for separating human behavior from automated behavior, need to be detectable and solid enough to stand out in automated statistical analysis. So, we asked ourselves which mining approach would allow us, to some extent, retrieve human decision points applied in a semi-automated workflow process?

We approached this question by exploring a pattern detection technique from the field of online gaming research where there are some interesting log mining solutions available, which can detect minor differences in gameplay behavior, revealing automated game bots amongst thousands of human players. Detecting differences in game style behavior amongst human player's game style behavior shows lots of similarities with detecting human behavior in an unstructured workflow process. Based on process attributes and data attributes, we can create classes of patterns that reveal behavior and this is a starting point to apply those patterns to event log data and decision models, to see where in the process steps and decision points, different patterns are seen. These differences can be further investigated for which attributes steer the patterns and ignore attributes that seem irrelevant for those patterns at that decision point. This way, we have a broad set of statistical methods at hand to gradually narrow down the relationships of process attributes, data attributes, and behavior patterns seen at decision points. However, a lot of data needs to be evaluated from different perspectives at various points in the process, which can be challenging if not done structured and well-orchestrated. Choi approached this by creating a behavior model first and then apply it to process data.
Creating a behavior model that is matching the specifics of the process under investigation is crucial. In Choi's game bot detection approach, there has been extensive research in clearly distinguished game player styles, which was used as input for the behavior model for bot detection. In a business process, the 'player styles' of human actors may be less distinctive as most workflow processes to not have the level of freedom as a user would have in an online role-playing game. Human behavior will also vary a lot per workflow implementation and business domain. However, we are able to identify the right process-specific behavior attributes that hint towards human pattern detection, we are therefor also able to apply Choi's behavior model in a new context. As progressing through our research, we found good support from literature that mining human decision points based on a behavior model is possible.

5.2. Conclusions

Most prominent element in our proposal is the concept of using a human behavior model that could be used to detect statistically matching decision points and use the result for decision model comparison. Our approach was experimental, as we took different components from earlier research and tried to combine these in a 5-step mining process, which has not been proven to work before. The technical execution of the 5 steps showed to be more complex as the intended formats for sequential mining steps and the behavior model could not be made compatible within time and with the available ProM plugins. Although individual working ProM plugins appear to deliver sound intermediate models, the models were less compatible for our purpose than we expected, both with respect to their inner workings as the technical compatibility within the tooling.

Also, we learned that our data set was *too lean*, not containing enough distinguishing attributes to properly detect big or small variances in behavior or other classifications. In Choi's research, over 50 potentially relevant attributes were available, through which new patterns could be revealed. Our data set contained 8 attributes which merely describe process instance *status changes* and *timestamps*, but do not describe enough 'context' to detect behavioral patterns. When mining for new, unknown attributes as potential identifiers for behavior, we learned the more context data the better. For future practice, a much broader, richer data set will be needed, as every attribute available upon a decision point can be a potential behavioral feature (Bolt, 2018).

5.3. Recommendations for practice

Recommendations for future practice would focus on creating a more consistent technical process for multi-step model comparison. Future research should focus on better technical compatibility of the experiment steps, to allow results from one step to be valid and sound as the input of the next step. Our tool of choice, ProM, did not allow us to create a behavior model in a format able to serve as input for model comparison, so other tooling more tailored to data mining and data manipulation can be explored.

Also, a richer data set with more process attributes is required, to lower the risk of ending up with none classifying attributes. to be able to detect small or big behavior variances. Also, the mining platform of choice, ProM and its plugins could be documentation more extensively, as most plugins found did not have technical documentation, except for their scientific publications.

Another recommendation regarding data would be extending the literature research for human behavior classifications. Choi's work on bot behavior detection builds largely on existing literature of game behavior styles, which already provides the contextual setting in which the bot behavior detections will operate. Something similar is needed for human behavior in workflows as well, although human behavior patterns may be very process domain-specific.

5.4. Recommendations for further research

Our research revealed promising reuse of a successful behavior detection from another research field into the field of process mining and rule detection. The advances of data mining solutions in online game logs are very scalable and well suited for large and complex data sets. Also, the data mining applications appear to have more data processing and pre-processing steps before actual behavior detection can be done. These are useful practices for future research.
Future research on human decision point mining should focus on creating a compatible behavior model format, that can be used in process mining and decision mining comparison. Different schools of tooling can support here. ProM seems to be the right choice for process mining, decision mining, and model comparison, but creating manual models as input could be better done via other tooling (e.g. RapidMiner, Disco, or a custom program). With regards to workflow data sets, we recommend to use a workflow event log with sufficient process instances but equally important to have a 'rich' attribute set with attributes that cover or touch a broad range of process context. This may need to be tailored to the specific business domain.

We also reflect on how our conclusions of this research affect our view on current literature.
Despite an abundance of literature on human behavior and human interaction in workflows, the specific topic of detecting human behavior seems less documented. Also, it seems that the log file mining techniques from gaming research are well-tailored to the fast speed of gameplay, high user numbers, and underlying logging and data mining technology. This makes the techniques quite applicable for practical experiments, hence the importance of keeping a broad outlook to the entire field of data science to encourage cross-functional exchanges of theory and practice.

References

A list of all references used:

1. Rozinat, A., Aalst, W. Van der.: Decision Mining in ProM. In. Dustdar et al (Eds.), Int Conf Business Process Management (BPM 2006), 4102. Springer-Verlag. pp. 420-425 (2006)
2. Aalst, W. v. d. (2016). Process mining: data science in action. In. Retrieved from http://dx.doi.org/10.1007/978-3-662-49851-4
3. Augusto, A. (2019). Automated Discovery of Process Models from Event Logs: Review and Benchmark. IEEE Transactions on Knowledge and Data Engineering, 31(2019-04), pg:686 -705.
4. Bolt, A., de Leoni, M., Van der Aalst, W.M.P. (2018). Process variant comparison: using event logs to detect differences in behavior and business rules. In (Vol. 74, pp. 53-66). Information Systems.
5. Chris J. Turner, A. T. e. a. (2012). Process mining: from theory to practice. Business Process Management Journal, 18(03), 493-512.
6. Crerie R., B. F. A., Santoro F.M. (2009). Discovering Business Rules through Process Mining.
7. de Leoni, M., W.M.P. van der Aalst, and M. Dees. (2016). A general process mining framework for correlating, predicting and clustering dynamic behavior based on event logs. Information Systems, 56, 235-257.
8. Garcia, C. d. S., Meincheim, A., Faria Junior, E. R., Dallagassa, M. R., Sato, D. M. V., Carvalho, D. R., Scalabrin, E. E. (2019). Process mining techniques and applications – A systematic mapping study. Expert Systems with Applications(133), 260-295.
9. Mannhardt, F., de Leoni, M., Reijers, H. A., Aalst, W. M. P. v. d., & Toussaint, P. J. (2018). Guided Process Discovery - A pattern-based approach. Information Systems, 76, 1-18.
10. ProM. (2019). ProM Framework.
11. Rapidminer.com (2019): RapidMiner data science software platform
12. Robert Thullner, S. R. (2011). Proactive Business Process Compliance Monitoring with Event-Based Systems. In (Vol. IEEE 15th International Enterprise Distributed Object Computing Conference Workshops, pp. 429 -437).
13. Salah, S., Maciá-Fernández, G., Díaz-Verdejo, J.E. et al. (2016). A Model for Incident Tickets Correlation in Network Management. Network System Management, 24.
14. Saunders, M., et al. (2015). Research Methods for Business Students. Pearson Education.
15. Smit, K. a. Z., Martijn. (2017). A Business Rules Management Reference Process for the Dutch Government. PACIS Proceedings, Summer 2017.
16. XES event log standard, IEEE 1849-2016: The XES Standard, https://standards.ieee.org/standard/1849-2016.html
17. Ta-Hsin, L. (2014). Incident Ticket Analytics for IT Application Management Services. IEEE International Conference on Services Computing, 568 -574.
18. ITIL v4 Best Practice for IT Service Management, Axelos, 2019
19. FITSM Part 0: Overview and vocabulary, Edition 2016 – Version 2.4

20. van der Aalst, W. M. P. (2018). Process discovery from event data: Relating models and logs through abstractions. WIREs: Data Mining & Knowledge Discovery, 8(3), 1-1.

21. W.M.P. van der Aalst, V. R., H.M.W. Verbeek, B.F. van Dongen, E. Kindler, C.W. Günther, (2008). Process mining: a two-step approach to balance between underfitting and overfitting. Softw. Syst. Model, 87.

22. ASL and BSL frameworks, ASL BiSL Foundation, 2002. https://aslbislfoundation.org/

23. BRG (Business Rules Group), What Defining Business Rules - What Are They Really? http://www.businessrulesgroup.org/

24. SBVR, Semantics of Business Vocabulary and Rules. http://www.omg.org/spec/SBVR

25. Ross, R. G.: Principles of the Business Rule Approach. Addison-Wesley Information Technology Series. ISBN-13: 978-0201788938 (2003)

26. Buhler, P., O'Callaghan, R., Aubry, S., DeJoy, D., Kuo, E., Shoup, N., ... & McBride, N. S. (2014). Service desk and incident impact patterns following ITIL change implementation. BPI Challenge

27. Obwegeser, N., T. Nielsen, D., & M. Spandet, N. (2019). Continual Process Improvement for ITIL Service Operations: A Lean Perspective. Information Systems Management, 36(2), 141-167.

28. Marrone, M., Kolbe, L.M. Uncovering ITIL claims IT executives' perception on benefits and Business-IT alignment. Inf Syst E-Bus Manage 9, 363–380 (2011)

29. Detecting Implicit Dependencies Between Tasks from Event Logs by Wen, Lijie; Wang, Jianmin; Sun, Jiaguang Frontiers of WWW Research and Development - APWeb 2006, 2006 Book

30. M. Chabrol, B. Dalmas, S. Norre and S. Rodier, "A process tree-based algorithm for the detection of implicit dependencies," 2016 IEEE Tenth International Conference on Research Challenges in Information Science (RCIS), Grenoble, 2016, pp. 1-11.

31. Kalynychenko, S. Chalyi, Y. Bodyanskiy, V. Golian and N. Golian, "Implementation of search mechanism for implicit dependencies in process mining," 2013 IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS), Berlin, 2013, pp. 138-142.

32. De Nicola, Missikoff and Smith, "Towards a method for business process and informal business rules compliance", Journal of Software, Wiley, 2011

33. M. Sonntag, D. Karastoyanova and E. Deelman, "Bridging the Gap between Business and Scientific Workflows: Humans in the Loop of Scientific Workflows," in E-Science 2010. 6th IEEE International Conference on E-Science (E-Science 2010), Brisbane, QLD, 2010 pp. 206-213.

34. Vanderfeesten, I., & Reijers, H. A. (2005, September). A human-oriented tuning of workflow management systems. In International Conference on Business Process Management (pp. 80-95). Springer, Berlin, Heidelberg.

35. G. Khodabandelou, "Contextual recommendations using intention mining on process traces: Doctoral consortium paper," IEEE 7th International Conference on Research Challenges in Information Science (RCIS), Paris, 2013

36. Paulo Barthelmess and Jacques Wainer. 1995. "WorkFlow systems: a few definitions and a few suggestions". In Proceedings of conference on Organizational computing systems (COCS '95).

37. Bolt, A., de Leoni, M. & van der Aalst, W.M.P. Scientific workflows for process mining: building blocks, scenarios, and implementation. Int J Softw Tools Technol Transfer 18, 607–628 (2016).

38. Leemans, S. J., Fahland, D., & Van Der Aalst, W. M. (2014). Process and Deviation Exploration with Inductive Visual Miner. BPM (Demos), 1295(46), 8.

39. Leemans, S. J. (2017). Inductive visual Miner - Manual

40. Rozinat A., de Medeiros A.K.A., Günther C.W., Weijters A.J.M.M., van der Aalst W.M.P. (2008) The Need for a Process Mining Evaluation Framework in Research and Practice. In: ter Hofstede A., Benatallah B., Paik HY. (eds) Business Process Management Workshops. BPM 2007. Lecture Notes in Computer Science, vol 4928. Springer, Berlin, Heidelberg

41. Seppe K.L.M. vanden Broucke, Jochen De Weerdt, Fodina: A robust and flexible heuristic process discovery technique, Decision Support Systems, Volume 100, 2017

42. Mannhardt, F., de Leoni, M., Reijers, H. A., & van der Aalst, W. M. (2017, June). Data-driven process discovery-revealing conditional infrequent behavior from event logs. In International Conference on Advanced Information Systems Engineering (pp. 545-560). Springer, Cham.

43. Maggi, F. M., Bose, R. J. C., & van der Aalst, W. M. (2012, June). Efficient discovery of understandable declarative process models from event logs. In International Conference on Advanced Information Systems Engineering (pp. 270-285). Springer, Berlin, Heidelberg.

44. Maggi, Dumas, García-Bañuelos, Montali (2013, Aug). Discovering Data-Aware Declarative Process Models from Event Logs. In International Conference on BPM 2013 (pages 81-96), Springer, Berlin, Heidelberg.

45. van der Aalst, W.M.P., Rubin, V., Verbeek, H.M.W. et al. Process mining: a two-step approach to balance between underfitting and overfitting. Softw Syst Model 9, 87 (2010)

46. Choi, Y., Chang, S., Kim, Y. et al. Detecting and monitoring game bots based on large-scale user-behavior log data analysis in multiplayer online games. J Supercomput 72, 3572–3587 (2016)

47. De Smedt, J., Hasić, F., van den Broucke, S. K. L. M., & Vanthienen, J. (2019). Holistic discovery of decision models from process execution data. Knowledge-Based Systems, 183, 104866.

48. De Smedt, J., vanden Broucke, S. K., Obregon, J., Kim, A., Jung, J. Y., & Vanthienen, J. (2016, September). Decision mining in a broader context: an overview of the current landscape and future directions. In International Conference on Business Process Management (pp. 197-207). Springer, Cham.

49. DMN, Decision Model Notation, OMG, https://www.omg.org/dmn

50. Westergaard, M., & Slaats, T. (2013). CPN Tools 4: A Process Modeling Tool Combining Declarative and Imperative Paradigms. In BPM (Demos).

51. A. Fernandes, K. K. Breitman, T. A. S. C. Vieira, M. A. Casanova and A. L. Furtado, "Semi-automated Workflow Synthesis," Second International Symposium on Leveraging Applications of Formal Methods, Verification and Validation (isola 2006), Paphos, 2006, pp. 332-334, doi: 10.1109/ISoLA.2006.32.