INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Business Process Analysis in Healthcare Environments

Álvaro José da Silva Rebuge

Dissertation for the degree of Master of Science in

Information Systems and Computer Engineering

Jury

President:	Prof. Dr. Nuno João Neves Mamede
Supervisor:	Prof. Dr. Diogo Manuel Ribeiro Ferreira
Member:	Profª.Drª Cláudia Martins Antunes

May 2012

Acknowledgements

To Prof. Diogo R. Ferreira for introducing me to the field I am passionate about. I am very grateful for the excellent guidance, availability to help, support, dedication, and knowledge sharing. He inspired me to found my own start-up where the knowledge acquired during this journey plays a key role.

To my research colleagues Carlos Libório, Cláudia Alves, and Gil Aires for the exchange of ideas, optimistic support and partnership. To the process mining community for the inspiration and inestimable knowledge they share.

To Hospital de São Sebastião for providing me the opportunity to develop the case study, and particularly to the people who took their time to help and provide valuable feedback. To the IT team for the great work environment, and a special thanks to Diogo Reis, for his guidance, vision, inspiration, and for believing from beginning in this project.

To my family and close friends for the encouragement and comfort. To my father for all the support, friendship, and knowledge he passes to me. I would be glad one day to be half the father he is. To my girlfriend Marta for all the love and for being there no matter what.

Abstract

Business process improvement is a critical success factor for healthcare organizations worldwide. Business Process Analysis (BPA) is a key step for process improvement initiatives, but traditional approaches are difficult to apply in practice due the highly dynamic, complex, ad-hoc, and multi-disciplinary nature of healthcare processes. Traditional BPA is time-demanding, costly, and mainly based on subjective descriptions made by people, which may not be aligned with reality. This work explores process mining as means to provide objective, efficient healthcare BPA without incurring with prohibitive costs. Process mining aims at an automatic extraction of process knowledge from event logs recorded by information systems, however, most techniques do not perform well in capturing the complex and ad-hoc nature of healthcare processes, and there is also the need to distinguish and understand process variants and exceptional behavior. We introduce a methodology to address these issues, where sequence clustering plays a key role. The approach is demonstrated in a case study conducted at a portuguese hospital, involving the analysis of processes running at the emergency service. A special purpose tool was developed to integrate the main stages of the methodology on top of the data acquired from the hospital information system.

Keywords: Business Process Analysis, Healthcare Processes, Process Mining, Sequence Clustering

Resumo

A melhoria dos processos de negócio é um factor crítico de sucesso para as organizações de saúde. Um dos pontos chave para iniciativas de melhoria passa por conhecer e analisar a situação actual dos processos, mas os métodos tradicionais revelam-se difíceis de aplicar na prática dada a natureza dinâmica, complexa, pouco estruturada, e multidisciplinar dos processos de saúde. Uma análise tradicional é geralmente demorada, incorre custos avultados, e depende essencialmente de descrições subjectivas que pessoas fazem da realidade. Este trabalho explora a extração automática de processos (*process mining*) executada a partir de dados registados por sistemas de informação. O conceito permite analisar processos na saúde de uma forma objectiva, eficiente, e sem custos proibitivos. Contudo, a grande maioria das técnicas de extração de processos apresenta dificuldades em capturar a natureza complexa, não estruturada dos processos de saúde. É importante também analisar as várias variantes dos processos e exceções. É proposta uma metodologia que dá resposta aos requisitos identificados e onde a clusterização de sequências (*sequence clustering*) assume um papel importante. A solução é demonstrada através de um caso de estudo realizado num hospital português e envolve a análise de processos do serviço de urgência. Foi desenvolvida uma ferramenta que integra os passos principais da metodologia sobre os dados adquiridos do sistema de informação do hospital.

Keywords: Análise de Processos de Negócio, Processos de Saúde, Extração Automática de Processos, Clusterização de Sequências

Contents

List of Figures

1 Introduction

Business process improvement is a topic of increasing concern and a critical success factor for healthcare organizations worldwide. The purpose is to increase organizational performance by process or information system redesign, covering the fundamental needs for today's healthcare organizations [3, 4, 5, 6]. These organizations are constantly pushed to improve the quality of care services in an unfavorable economical scenario and under financial pressure by governments [3, 6]. Improving process efficiency is therefore of utmost importance. On the other hand, it has been reported that faulty healthcare processes are one of the main causes leading practitioners to make technical mistakes [7]. These can severely compromise patient safety and even cost lives [8]. Moreover, recent trends such as patient-centric services and integrated care pathways have been introduced to improve the quality of care services, which are also requiring healthcare organizations to redesign and adapt their processes [3, 5]. They break functional boundaries and offer an explicit process-oriented view of healthcare where the efficient collaboration and coordination of physicians becomes a critical issue [3, 4, 5]. In this scenario, healthcare information systems should be designed to directly support clinical and administrative processes, integrating and coordinating the work of physicians [3, 5, 6]. Unfortunately, in general these systems need to be rethought since they lack maturity and interoperability with other systems, and offer a weak support to healthcare processes [7, 9, 10]. In some specific cases, the proportion of these problems can be worrisome [9].

Business Process Analysis (BPA) becomes extremely important to enable the successful improvement of business processes [11, 12]. BPA defines a set of methodologies, techniques and tools

aimed at understanding and reasoning, from a process perspective, how an organization works, in order to detect flaws or potentials of improvement. The knowledge obtained from BPA establishes an important starting point to redesign processes and systems in alignment with strategical objectives. [13, 14, 15, 16]. The motivation for this work is to provide healthcare organizations with innovative means to perform BPA, in particular with the aid of process mining. The essence of process mining is the automatic extraction of process knowledge from event logs recorded by information systems. The scope is limited to the operational business processes of healthcare, commonly known as healthcare processes.

The remainder of this chapter is as follows. Section 1.1 introduces notions related to healthcare processes and explores their characteristics. The latter is important to understand the problem definition, which is described in Section 1.2. Section 1.3 presents the concept of process mining by explaining its role within the context of our problem. Section 1.4 introduces the research goals. Finally, Section 1.5 develops the rationale of the research design.

1.1 Healthcare Processes

Healthcare processes are seen as operational business processes of healthcare organizations. In a broader context, business processes establishes a modern perception on the operations of an organization. Rather than depicting the organization as being fragmented into separated functions or departments, as in the traditional view, business processes focus on how the organization jointly works on producing an outcome; thus the cross functional/organizational characteristics identified in literature. There is no universally accepted definition of what a business process actually is, see [12, 17, 18]. A rather liberal definition is given in [18]: a business process is "a set of activities that, taken together, produce a result of value to a customer".

Similarly, there is no coherent definition of what healthcare processes are. Notwithstanding, healthcare processes can be classified as *medical treatment processes* or *generic organizational processes* [5]. Medical treatment processes, also known as clinical processes, are directed linked to the patient and are executed according to a diagnostic-therapeutic cycle, comprising observation, reasoning and action. The diagnostic-therapeutic cycle heavily depends on medical knowledge to deal with *case-specific decisions* that are made by interpreting patient-specific information. On the other hand, organizational or administrative processes are *generic process patterns* that support medical treatment processes in general. They are not tailored for a specific condition but aim to coordinate medical treatment among different people and organizational units. Patient scheduling is an example of an organizational processes.

None of these processes are trivial. They are executed under an environment that is continually changing and that is commonly accepted to be one of the most complex when compared to other environments [3]. The healthcare environment and its underlying processes have peculiar characteristics with respect to their degree of dynamism, complexity and multi-disciplinary nature. In general, healthcare processes are recognized to have the following characteristics:

- Healthcare processes are *highly dynamic* [3, 6, 10, 19]. Process changes occur due to a variety of reasons including the introduction of new administrative procedures, technological developments, or the discovery of new drugs. Moreover, medical knowledge has a strong academic background that is continually evolving, meaning that new treatment and diagnostic procedures are constantly being discovered that may invalidate current treatment pathways or require corrective measures. Also new diseases are constantly being discovered that may require healthcare organizations to change their processes.

- Healthcare processes are *highly complex* [3, 5, 6, 20, 21]. Complexity arises from many factors such as a complex medical decision process, large amounts of data to be exchanged, and the unpredictability of patients and treatments. The medical decision process is made by interpreting patient-specific data according to medical knowledge. This decision process is the basis of clinical processes and it is difficult to capture, as medical knowledge includes several kinds of medical guidelines, as well as the individual experience of physicians. Also, the amount of data that supports medical decisions is large and of various types; consider for example the different types of reports or the different types of exams that are possible and are exchanged between physicians. Moreover, the patient's body may react differently to drugs and complications may arise during treatment, meaning that new medical decisions need to be made accordingly. Medical decisions and treatment outcomes may therefore be unpredictable, also meaning that as clinical processes are instantiated their behavior may also be unpredictable.

- Healthcare processes are *increasingly multi-disciplinary* [10, 19]. Healthcare organizations are characterized by an increasing level of specialized departments and medical disciplines, and care services are increasingly delivered across organizations within healthcare networks. Healthcare processes, therefore, are increasingly executed according to a wide range of distributed activities, performed by the collaborative effort of professionals with different skills, knowledge and organizational culture.

- Healthcare processes are *ad-hoc* [19, 20, 21]. Healthcare highly depends on distributed human collaboration, and participants have the expertise and autonomy to decide their own working procedures. As physicians have the power to act according their knowledge and experience, and need to deviate from defined guidelines to deal with specific patient situations, the result is that there are processes with high degree of variability, non-repetitive character, and whose order of execution is non-deterministic to a large extent.

These characteristics are problematic for traditional BPA efforts in healthcare environments.

1.2 Problem Definition

Traditional BPA assumes that people must be capable of explaining what is happening within the organization and describing it in terms of processes, such that each description is valid, unambiguous, and is a useful abstraction of reality [13]. These descriptions can be represented in terms of process models [22, 23]. Traditionally, process models result from the collaborative effort between key stakeholders and process analysts, as on one side we have the inside knowledge of how the organization works, and on the other side the expertise to represent that knowledge in formal languages associated with process modeling [3]. This traditional approach has two main problems.

The first is that traditional BPA is time-consuming [1, 24], as it implies lengthy discussions with workers, extensive document analysis, careful observation of participants, etc. The second problem is that, typically, there are discrepancies between the actual business processes and the way they are perceived or described by people [24]. Several reasons for this can be pointed out, including the inherent difficulty in understanding complex and non-deterministic phenomena [25, 26]. The more complex and ad-hoc the processes are, the more difficult it will be for people to describe them. Also, when processes involve distributed activities, it is difficult for workers to have a shared and common perspective of the global process [27], especially if the process is continually changing.

In summary, we find that BPA is extremely important for healthcare organizations; however, traditional approaches are time-consuming, and they may not provide an accurate picture of business processes, which are highly dynamic, highly complex, multi-disciplinary and ad-hoc.

1.3 The Role of Process Mining

Process mining offers an interesting approach to solve or mitigate the above problem [28]. The idea is depicted in Figure 1.1 and can be explained as follows. As organizations depend on information systems to support their work, these systems can record a wide range of valuable data, such as which tasks were performed, who performed the task, and when. For example, as patients enter an emergency department a system records the triage, the nurse who performed it, the time it occurred, and for which patient, i.e. the work case. These event data can be organized in such a way that they contain a history of what happened during process execution − the so called event logs. From these event logs one can extract process knowledge using process mining techniques.

The extraction of process knowledge from systems, which can be made automatically to a large extent, can reduce the time required for process analysis. The main benefit, however, is that the models acquired from process mining are based on real executions of the processes; therefore, one gains insight about what is *actually* happening, and ultimately the knowledge provided from process mining can be used for effective improvement of those processes and of their supporting systems.

In healthcare environments, it may be possible to extract event data from several kinds of specialized systems [21], such as those based on Electronic Patient Records (EPR). As another example, radiology information systems (RIS) can record the workflow of patient examinations,

from the exam request to the exam report. Also, emergency information systems can record the careflow of patients, and billing information systems usually combine data from other hospital systems about the activities performed on a patient.

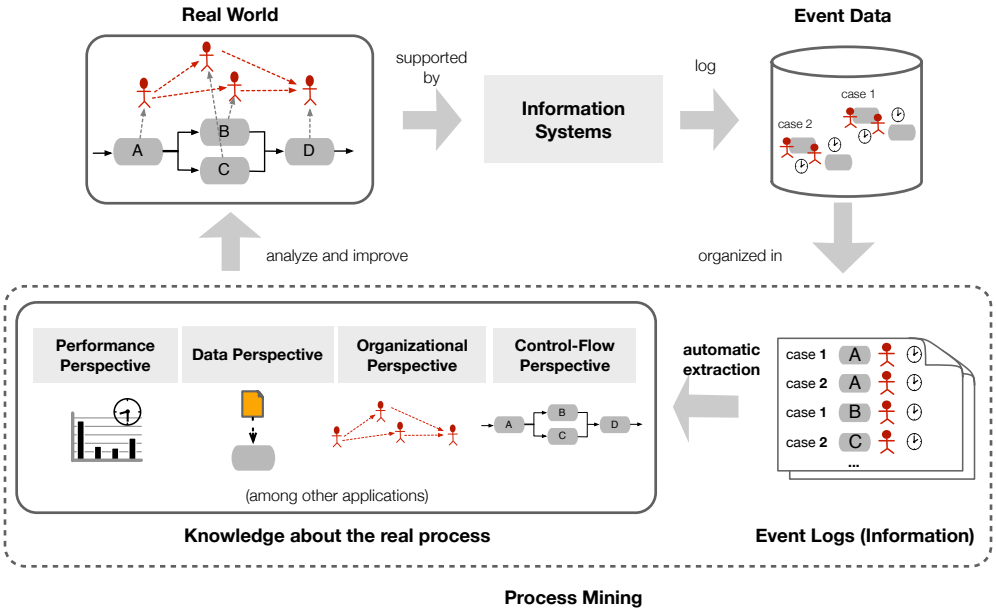

Figure 1.1: The concept of process mining.

1.4 Research Goals

The present research proposes to achieve the following goals:

Goal 1: *Explore how process mining can be used to improve traditional BPA approaches in healthcare.* Despite the potential benefits of process mining for analyzing healthcare processes, the concept is a recent field of research and applications in real life organizations are still scarce. The present work intents to contribute for a better understanding on how process mining helps in improving the analysis of healthcare processes; so as to understand advantages, limitations, and difficulties of application, not only from a theoretical but also from a practical point of view.

Goal 2: *Devise a methodology based on process mining in order to support BPA in healthcare.* There is still a lack of established methodologies for BPA initiatives based on process mining. Moreover, such approaches disregard the specificities of healthcare processes. In order to promote reusability of our approach, we devise a methodology based on process mining that supports the analysis of different perspectives of healthcare processes; without disregarding the characteristics of these processes, and by including process mining techniques that are especially useful in healthcare. The methodology extents the work of previous studies in order to tackle specificities of healthcare.

Goal 3: *Validate the potential of the methodology in a real life healthcare organization.* It is the aim of this work to test the theoretical development of the methodology in a real life healthcare organization, so that one evaluates the potentially of the methodology in the real world (as well

as the applicability of process mining within the context of our problem). More specifically, it is conducted a case study at the Hospital of São Sebastião (HSS), located in the north of Portugal.

Additional Goal: *Develop a tool that is capable to support the methodology within the domain of the case study.* This additional goal appears because it was found of interest to make all the steps of the methodology available in a single application that could be used by the staff of HSS. The tool appears as a means to support the proposed methodology within the domain of the case study.

1.5 Research Methodology

The research goals were achieved following a methodology of research comprising literature study and case study research.Literature study is known to be fundamental for an investigation of the concepts and theories developed in the specific field, as well as for exploring the findings of other researchers and reviewing the employed research methods and achieved results [29]. It conforms with our intents to: (1) explore from a theoretical point of view the applicability of process mining in the context of our problem (part of goal 1); and (2) define the methodology that goal 2 proposes to achieve, since it develops from the theoretical understanding of previous studies and other related theories.

The case study is among the most employed research strategies in the field of information systems [30]. It allows to capture reality in great detail and can be used to build, test and extend theories within organizational environments [31]. The case study is a feasible approach in situations where [32]: (1) there is a need to conduct the research in its natural setting; (2) the problems reside within a rapidly changing environment; (3) there is an emphasis on the why and how questions; and (4) there is a lack of previous studies and elaborated theoretical understanding with regard to the problem under investigation. In our particular situation, the case study is used to: (1) explore, from a practical point of view, how process mining can be used to improve traditional BPA in healthcare (part of goal 1); and (2) test the theoretical development of the methodology in a real life healthcare organization, in order to achieve goal 3.

1.6 Document Outline

This introductory chapter presented the foundations of the research, namely: (1) the problem definition; (2) the role of process mining as means to mitigate the problem; (3) the research goals; and (4) the research methodology. The document is further divided in two parts for clear separation of theory and practice, reflecting the research methodology followed (see Figure 1.2). **Part I** regards the theoretical development of the research and comprises the following chapters.

Chapter 2 develops the theoretical background of the research based on literature study of available publications on process mining. The chapter starts by introducing general concepts of process mining and by exploring existing techniques and tools. Relevant studies related to the

applications of process mining in healthcare settings are then introduced, which let us better understand advantages, limitations and challenges to process mining in this particular domain. The chapter concludes with an analysis and reflection on the theoretical findings, resulting on the identification of research opportunities and requirements for the definition of the proposed methodology.

Chapter 3 explains the rationale of the proposed methodology. The methodology results from an extension to the work of Bozkaya [2] by incorporating a sub-methodology where sequence clustering plays a key role in order to address the specificities of healthcare. The sub-methodology is named Sequence Clustering Analysis and is explained in detail given its importance. The result of the chapter is a methodology for analyzing different perspectives of healthcare processes.

Part II regards the practical development of the research (essentially the case study) and comprises the following chapters.

Chapter 4 introduces the case study conducted at HSS by explaining: (1) the organizational context; (2) the importance of process mining to the hospital; (3) the scope of the case study and how the data was gathered; and (4) the rationale of the tool developed in order to support the analysis of the data in alignment with the methodology. The database developed for the case study contains event data from the emergency operations of the hospital, including triage, treatments prescribed, diagnosis made, medical exams requested, and patient discharges/transfers. The tool developed implements the methodology and integrates the case study databased in the same environment.

Chapters 5, 6 and 7 develop the analysis of a set of selected processes, in accordance with the methodology and using the event data gathered. **Chapter 5** analyzes the emergency radiology process, an organizational process that highly impacts the operations of the emergency service. **Chapter 6** analyzes how the emergency department treats patients with acute stroke (which regard a clinical process and which performance is impacted by the emergency radiology process). These later two chapters mainly analyze the behavioral and performance perspective of the process. The organizational perspective is left to **Chapter 7** in order to study a very specific and sensitive problem experienced at HSS regarding the transfer of patients among emergency professionals. The results obtained are discussed in a concluding section of each chapter of the case study, incorporating the feedback and validation of domain experts.

Chapter 8 concludes the study with a reflection on the research findings, as well as an identification of main contributions and future directions.

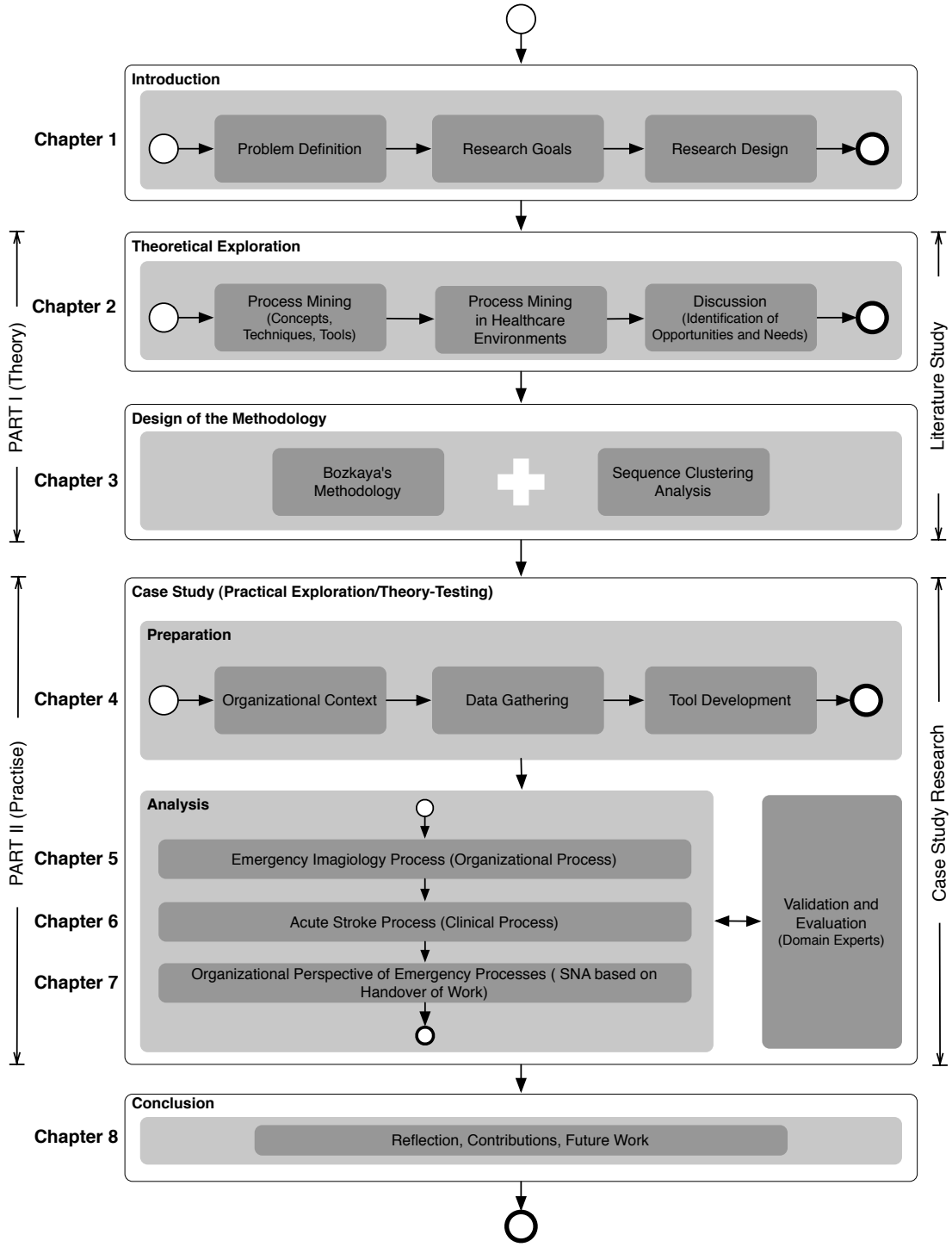

Figure 1.2: Overview of the research project.

Part I

Theoretical Development

2

Related Work

In the introductory chapter, process mining is identified as an approach with potential to reduce the time required for process analysis, as well as to provide more accurate results than traditional approaches for process analysis. The potential applicability of this concept in healthcare established a research opportunity worth to explore and defined the course of the research. This chapter develops theoretical understanding related to process mining. Section 2.1 introduces some notions and explores existent techniques and tools. Section 2.6 studies previous work on the specific application of process mining in healthcare settings. Finally, section 7.3 concludes the chapter with a summary and discussion on the theoretical findings in order to understand advantages, limitations, and challenges of process mining with regard to the analysis of healthcare processes; as well as to identify opportunities and requirements for the methodology proposed in next chapter.

2.1 Concepts of Process Mining

Process mining can be seen in context of Business Intelligence and Business Activity Monitoring; however, it is usually referred in the broader context of Business Process Management (BPM) [11, 23]. BPM aims at an iterative improvement of business processes by using a set of methods, techniques, and technology, to support the business process life-cycle. This life-cycle comprises: (1) the design; (2) the configuration; (3) the enactment; and (4) the diagnosis of business processes. Process mining emerged to support the design and diagnosis phase of BPM while also changing

the traditional BPM life-cycle approach. Instead of design some business process model at first, we assume that, during the enactment phase of the process, systems can record event data regarding what tasks have been executed, the order of execution, and for which process instance[1]. The event data is then used to discover process knowledge (such as process models) which is suitable for the analysis of processes, and also serves as input to the design phase.

Process mining currently has three possible applications [28]: (1) the discovery of process knowledge; (2) conformance checking, i.e. detect, locate, explain and measure deviations between an *a priori* model (an already defined model) and the real process model (the discovered model); and (3) extension of a process model with event data, e.g. enrich a process model with performance information.

case_id	task_id	originator	timestamp	data
1	A	Phil	7/9/09 - 9:30	(...)
1	H	Phil	7/9/09 - 9:40	(...)
1	I	Peter	7/9/09 - 9:41	(...)
2	A	Steve	7/9/09 - 9:43	(...)
4	A	Phil	7/9/09 - 9:50	(...)
2	B	Jeffery	7/9/09 - 9:51	(...)
2	C	Thomas	7/9/09 - 9:53	(...)
4	B	Jefferson	7/9/09 - 9:55	(...)
2	F	Jeffery	7/9/09 - 10:01	(...)
4	D	Jefferson	7/9/09 - 10:04	(...)
2	G	Jeffery	7/9/09 - 10:16	(...)
4	G	Jefferson	7/9/09 - 10:18	(...)
2	I	Jeffery	7/9/09 - 10:21	(...)
4	I	Jefferson	7/9/09 - 10:22	(...)

Figure 2.1: Example of an event log.

The event log is the starting point for any process mining application. Figure 2.1 depicts one example. The event log combines event data in an ordered sequence of audit trail entries. Each audit trail entry can be denoted as a tuple $< case\_id, task\_id >$, where $case\_id$ is a unique identifier of a process instance, and $task\_id$ an unique identifier of a process task. This is the minimum form required. If systems can record other kinds of event data, such as the task originator, i.e. who performed the task, a timestamp for the event, or other additional data (e.g. relevant data objects attributes that changes during a process execution), then audit trail entries can be augmented and have a form such as $< case\_id, task\_id, originator, timestamp, data >$. An event log can be classified according to [22]: (1) the completeness of the information present in the log; and (2) the degree of noise. The event log is said to be complete if it captures all the possible process behavior. Noise refers to parts of the event log that may be incorrect, missing, or refer to exceptions (e.g. someone skips an activity, an event is incorrectly logged, or rare process events are executed).

[1] A process instance, or case, refers to one specific execution of a process.

The event data contained in the log determines which process perspectives are possible to analyze [28]. Currently, process mining considers four perspectives: (1) the control-flow perspective (how); (2) the organizational perspective (who); (3) the data perspective (what); and (4) the performance perspective (when). The control-flow perspective is concerned with the process behavior, namely the activities comprised in the process and their relations. The organizational perspective focus on who performed the activities; for this we need the originator field in the event log. The performance perspective aims to model bottlenecks or derive performance indicators, such as throughput times, or sojourn times; the timestamp field is required. The data perspective is related with the data objects that serve as inputs to or outputs for the activities of a case; the data field must be present.

Researchers have been proposing a large variety of techniques in order to support the different applications of process mining. In next sections we discuss the most relevant.

2.2 Process Mining Techniques

Mining techniques are aimed at discovering different kinds of models for different perspectives of the process, namely: (1) the control-flow perspective; (2) organizational perspective; and (3) the data perspective. The format of the output model will depend on the technique used.

2.2.1 Control-Flow Perspective

The main focus of process mining is to discover the control-flow perspective as it is the starting point to discover other perspectives as well. One of the first developments in this area was the *alpha-algorithm* [33], proposed by van der Aalst et al., and which discovers a Petri Net [34] that models the process (see Figure 2.2). Assuming that the event log is complete, it establishes a set of relations between the tasks in the event log, such as causal relations. One task A is considered the be the cause of a task B only if B follows A but A never follows B. The algorithm just checks if the relation holds or not and does not take into account the frequency of a relation. Therefore, if noise is present in the event log that make, for instance, A appear after B the algorithm do not provide valid results. Also, it can not mine some workflow constructs [1], such as short loops and duplicate tasks. To address this last problem, Wen et al.[35] proposed two extensions to the alpha-algorithm, the *Tsinghua alpha-algorithm* and the *alpha++-algorithm*.

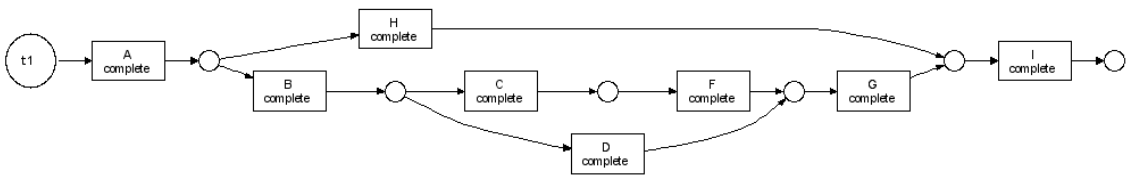

Figure 2.2: The resulting Petri Net by applying the alpha-algorithm to the event log in Fig.2.1.

[1] Workflow constructs are behavioral patterns of processes. For more insight the reader is referred to [24].

Van Dongen et al. latter introduced the *Multi-Phase Miner* [36], which discovers an Event-driven Process Chain instead of a Petri Net. This approach shares the same limitations with the alpha-algorithm with respect to noise, as both consider the same relations between tasks.

Weijters et al. approached the problem of noise with the *Heuristic Miner plug-in* [37]. The relations between tasks are established as the alpha-algorithm, however, it also considers some heuristics to compute the frequency of the relations. Therefore, it is possible to disregard the relations with a frequency below a given threshold, which is one of the input parameters of the algorithm. The output model is an heuristic net. Unfortunately, in presence of complex and ad-hoc processes, the event log contains a large variety of traces, and the Heuristic Miner will discover a large set of relations. The result is a very confusing model that is hardly understandable at human eye. These kind of results are commonly referred as spaghetti-like models, see Figure 2.3.

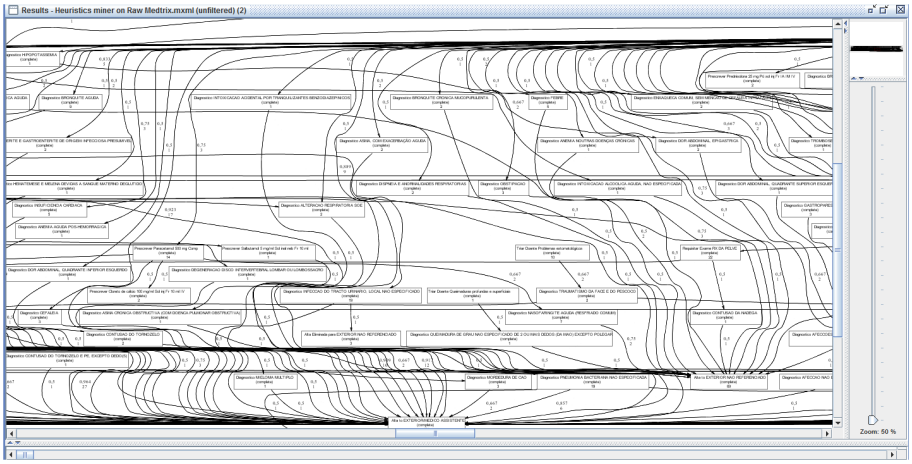

Figure 2.3: Example of a spaghetti-like model.

To prevent spaghetti-like models, researchers have been paying a particular attention to clustering techniques. These are described ahead in more detail. At this point we can refer the approach proposed by Günther, the *Fuzzy Miner* [38]. This technique is very robust to noise, and handles the problem of spaghetti models with a mixture of abstraction and aggregation. It provides a high-level view of the process and the details are aggregated, i.e. the details are abstracted from regular behavior. The most significant tasks and relations are emphasized. To achieve these results two metrics are used [38]: (1) significance, that measures the level of relevance of tasks; and (2) correlation, that determines how closely related two following tasks are, and such that highly related tasks can be aggregated. Additionally, the Fuzzy Miner offers a dynamic view of the process by replaying the log in the model and showing the cases flowing though the model.

Most of the mining techniques are mainly based on searching local information in the log, i.e. the dependency relations between tasks are captured considering the tasks that directly precede or directly follow each other. For this reason these techniques have difficulties in discovering workflow patterns that do not suppose a direct succession or precedence of tasks, such as non-local dependencies. To address these limitations, de Medeiros et al. introduced the *Genetic Algorithm*

[39]. It exploits the global search capabilities of genetic algorithms and tries to capture process models containing non-local relations between tasks, while also handling noise.

2.2.2 Organizational Perspective

To discover the organizational perspective of the process the main references are the *Social Network Miner* [40] and the *Organizational Model Miner* [41], both developed by Song and van der Aalst. The *Social Network Miner* focuses on showing the relation between individual performers, or groups of them. This technique aims at discovering a sociogram (Figure 2.4 (a)) from the event log by considering some socio-metric, such as: the handover of work; people working on joint cases, i.e. "working together"; and people working on joint activities. After generating the social network, several techniques can be applied to measure centrality, density, cohesion, etc. which are standard Social Network Analysis (SNA) metrics.

Instead of showing the relations between individual performers, the *Organizational Miner* shows the relation between the performers, or groups, and the process. It attempts to discover an organizational model that structures the organization by classifying people in terms of roles or organizational units. To classify the organization in terms of roles it uses a simple approach that derives a role for each task and assigns it to the performers who executed the task. To classify the organization in terms of organizational units (Figure 2.4 (b)) this technique attempts to obtain different clusters from a social network. The idea is that each cluster correspond to a possible organizational unit, which is then assigned to the corresponding tasks and originators.

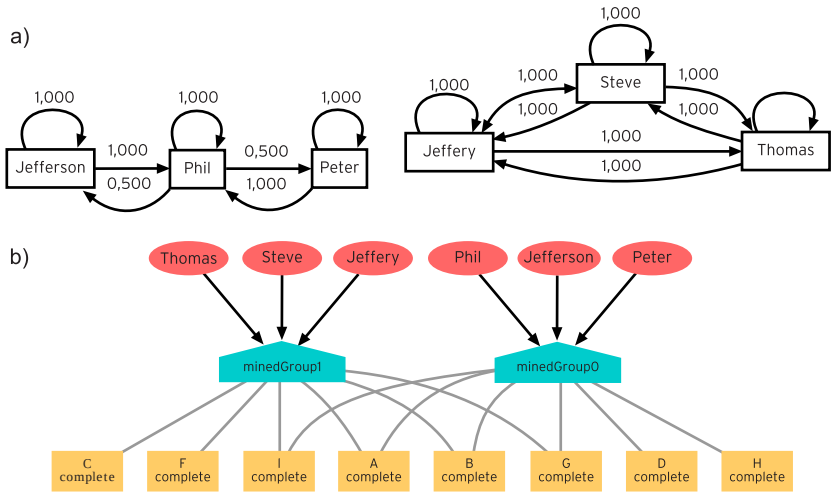

Figure 2.4: Example of the organizational perspective: a) sociogram from the event log shown in Figure 2.1; b) organizational model from the same event log.

2.2.3 Data Perspective

The main advances to discover the data perspective are included in the *Decision-Miner* [42] implemented by Rozinat et al. It aims at discovering choices in a business process by taking into account the data values contained in the event log and their dependencies. Using this information this technique discovers decision rules for taking alternative paths in the process.

2.3 Process Analysis Techniques

Whereas mining techniques focus the discovery of models in order to describe the process at different perspectives, analysis techniques provide means of inspection, reason, and evaluation of models or event logs. Two main categories are: (1) performance analysis; and (2) conformance checking.

2.3.1 Performance Analysis

To analyze the performance of the process one can start by referring the Performance Analysis with Petri Net[43]. This technique assumes the presence of a Petri Net model (mined for example with the alpha-algorithm). It replays the log traces of the event log to retrieve the values of various performance indicators, such as arrival rates, throughput times, i.e. the total flow time of the process, waiting times, etc. The waiting times are projected onto the Petri Net model. The *Basic Performance Analysis* focuses on the process instances to compute the performance indicators, disregarding the structure of the process, or the presence of a process model. The results are presented in different kinds of charts, such as bar charts, Gantt charts, meter charts, etc. Other approach is the *Dotted Chart Analysis* [44], see Figure 2.5. This technique displays a chart where the events are represented as dots, and the time is measured along the horizontal axis. The vertical axis can represent the process cases, tasks, originators, etc. It also provides performance metrics such as the time of the first and of the last event, time between events, case durations, etc.

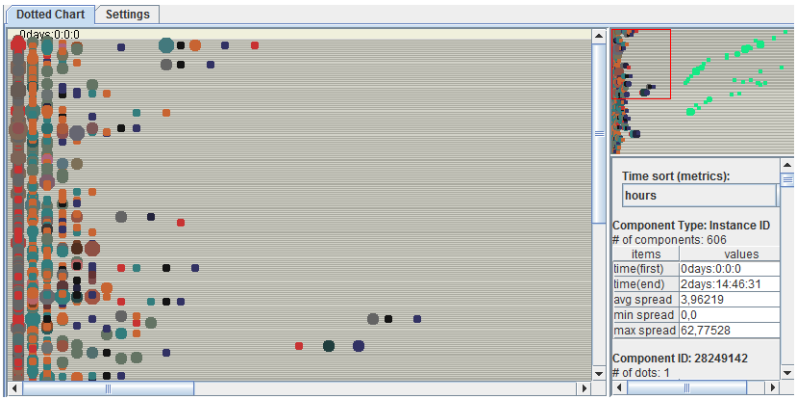

Figure 2.5: Example of the Dotted Chart.

2.3.2 Conformance Checking

The *Conformance Checker* aims at answering if business processes are being executed as planned [45, 46]. This technique analyzes the gap between an a-priori model and the observed executions of the process, depicting and quantifying detected discrepancies. To measure the conformance the concepts of fitness and appropriateness are considered. The fitness metric checks if the event log complies with the control flow specified for the process. The appropriateness metric checks if the model describes the behavior present in the event log. These metrics also serve as foundation to assess the quality of a mined model. Sometimes, however, there is no a-priori model at hand, or the model is not complete. In this case, the *LTL Checker* [47] can be used. The user defines a set of business requirements, such as business rules, and translate those requirements to Linear Temporal Logic formulae. The event log is then checked in order to find whether it satisfies the formulae. An example is to define a formula stating that task A has to be executed after B. LTL Checker run over the event log and detect the cases where A is not executed after B.

2.4 Clustering Techniques

Clustering techniques can be used as a preprocessing step, and their purpose is to handle event logs that contain large amounts of data and high variability in the recorded behavior [48]. Rather than running control-flow mining techniques directly on large event logs, which would generate very confusing models, by using clustering techniques it is possible to divide traces into clusters, such that similar types of behavior are grouped in the same cluster. One can then discover simpler process models for each cluster.

Several clustering techniques can be used for this purpose, such as the *Disjunctive Workflow Schema (DWS) plug-in* [49] which can be seen as an extension of the Heuristic Miner plug-in. It uses the Heuristic Miner to construct the initial process model; however, this model is iteratively refined and clustered with a k-means algorithm. The final result is a tree with the models created, where the parent nodes generalize their child nodes. At each level of the tree, a set of discriminant rules characterize each cluster. These discriminant rules identify structural patterns that are found in the process model but not registered in the log.

The *Trace Clustering plug-in* [50] offers a set of distance-based clustering techniques based on the features of each trace, such as how frequently the tasks occur in the trace, or how many events were created by the originators in that trace. Different clustering methods, such as k-means and Self-Organizing Maps, can then be used to group closely-related traces in the same cluster. This technique does not provide a model for visualization; i.e. to visualize the clusters one needs to use some other mining technique.

The *Sequence Clustering plug-in* [51] was motivated by previous works outside the ProM framework [52, 53, 54]. Sequence clustering takes the sequences of tasks that describe each trace and groups similar sequences into the same cluster. This technique differs from Trace Clustering in

several ways. Instead of extracting features from traces, sequence clustering focuses on the sequential behavior of traces. Also, each cluster is based on a probabilistic model, namely a first-order Markov chain. The probabilistic nature of Markov chains makes this technique quite robust to noise, and it provides a means to visualize each cluster. In the work of [51], sequence clustering was shown to generate simpler models than trace clustering techniques.

2.5 The ProM Framework

The techniques discussed so far are readily available in the ProM framework[1] (the figures accompanying the description of the techniques were obtained from the same tool). ProM [55] is an open-source framework developed by the Eindhoven University of Technology that incorporates different process mining techniques under an extensible plug-in architecture. Since process mining has been an active field of research, the community has been highly encouraged to implement their techniques as ProM plug-ins, and contribute to the framework development. ProM is nowadays considered the *de facto* tool for process mining.

The event logs that serve as input to ProM are structured according to the Mining XML (MXML) format [56]. The MXML format started as an initiative to share among different process mining tools a common structure for event logs[2]. In other words, process mining plug-ins can understand what is contained in an event log regardless the source system and independently of other possible representations.

ProM incorporates three main kinds of plug-ins: (1) mining plug-ins; (2) analysis plug-ins; and (3) conversion plug-ins. Conversion plug-ins aim at converting a process model to other different languages, such as the conversion from Petri Net models to Event-driven Process Chains (EPC) and vice-versa. Mining and analysis plug-ins implement the mining and analysis techniques already discussed. ProM does not clearly distinguishes clustering techniques. They are classified as analysis techniques instead.

To note that ProM itself only interprets the event log and does not include the extraction of event logs from systems. The gap can be fulfilled with the ProM Import Framework [57] [3]. It provides plug-ins to extract MXML event logs from popular log-producing systems (e.g. WebSphere, Staffware, Apache, Concurrent Version System), as well as a library that can used to Java code custom plug-ins. Although requiring manual effort, the extracted event logs can then be imported to ProM for analysis.

[1] ProM can be obtained at http://prom.win.tue.nl/tools/prom/

[2] The schema for the MXML format is available at http://www.processmining.org/WorkflowLog.xsd

[3] The ProM Import Framework can be obtained at http://prom.win.tue.nl/tools/promimport/.

2.6 Process Mining in Healthcare Environments

The application of process mining for BPA in healthcare is a relatively unexplored field, although it has already been attempted by some authors. For example, [20] applied process mining to discover how stroke patients are treated in different hospitals. First there was a need for intensive preprocessing of clinical events to build the event logs. Then the ProM framework was used along with the Heuristic Miner to gain insights about the control-flow perspective of the process. Different practices that are used to treat similar patients were discovered, and unexpected behavior as well. The discovered process model was converted to a Petri net. The performance of the process was then analyzed by projecting performance indicators onto the Petri net. It was concluded that process mining can be successfully applied to understand the different clinical pathways adopted by different hospitals and different groups of patients.

In further work, [21] conducted a case study in the AMC hospital in Amesterdam. Process mining was used to analyze the careflow of gynecological oncology patients. An intensive preprocessing of data was also needed to build the event log. The control-flow, the organizational, and the performance perspectives of the process were analyzed. To discover the control-flow perspective, the Heuristic Miner was used first, which resulted in a spaghetti model that was not useful for analysis. The authors explain this difficulty based on the complex and unstructured nature of healthcare processes. Trace Clustering and the Fuzzy Miner were then used to separate the regular behavior from the infrequent one, and understandable process models were discovered for each cluster. To study the organizational perspective, the Social Network Analysis plug-in was used to understand the transfer of work between hospital departments. To analyze the performance perspective, the Dotted Chart and the Basic Performance Analysis plug-in were used, both giving useful performance indicators about the careflow. The discovered models were confirmed by the staff of the AMC hospital, and also compared with an *a priori* flowchart of the process, with good results. It should be noted that this flowchart was created with a lot of effort from the AMC staff. The authors concluded that process mining is an exceptional tool for the analysis of complex hospital processes.

Another study [1] focused on the problems of traditional BPA in the Erlangen University Clinic, in Germany. In order to support the analysis of the radiology workflows at the clinic, the authors developed a data warehouse for process mining. During the study several control-flow mining techniques were evaluated, and the authors found that none of the techniques alone was able to meet all the major challenges of healthcare processes, such as noise, incompleteness, multiple occurrence of activities, and the richness of process variants (see Figure 2.6). Deterministic approaches, such as the α-algorithm and also the Multi-Phase algorithm, are severely affected by the incompleteness and noise present in clinical logs, so they were not able to produce valid process models. The Heuristic Miner, the DWS Algorithm, and the Genetic Miner produced the best results in presence of the noisy data. The detection of process variants was only possible with the DWS Algorithm, since it was the only one that used clustering techniques. The α-algorithm was the only one to (at

least partially) handle activities that occurred multiple times in the process without being part of a loop. Despite the limitations, the authors concluded that process mining has a great potential to facilitate the understanding of medical processes and their variants.

The author of [19] evaluated the capabilities of the Heuristic Miner and also the DWS Algorithm to analyze the processes of an Intensive Care Unit of the Catharina Hospital, in Eindhoven. The Heuristic Miner produced inaccurate and confusing models, and it is unable to distinguish process variants. The clustering approach of the DWS Algorithm was able to discover some behavioral patterns; however, the discriminants rules were hard to understand. None of them was considered to be useful to gain insight about exceptional medical cases (that can be translated into infrequent behavior), and about variants of processes. To handle this problem, the author introduced the Association Rule Miner (ARM) plug-in, which aims to discover association rules and frequent itemsets in the event log. The technique has proved to be useful to obtain behavioral patterns in the event log and to group similar patients. To improve the capabilities of the algorithm in discovering exceptional medical cases, and also to obtain simpler process models, the ARM includes a clustering technique that divides the log into clusters with similar association rules and frequent itemsets.

	Alpha-algorithm	Alpha++-algorithm	Heuristic miner	DWS-algorithm	Genetic algorithm	Multiphase miner	Region miner
Truth to reality	-	+	+	+	+/-	-	-
Noise and incompleteness	-	+/-	+	+	+	-	-
Sequences, forks, and concurrency	-	+	+	+	+	+	-
Loops	-	+	+	+	+	-	-
Repetitive activities	-	-	-	-	+/-	-	-
Fuzzy entry and endpoints	-	+/-	+	+	+	-	-
Process types and variants	-	-	-	+	-	-	-

Figure 2.6: Lang's evaluation of control-flow mining techniques in healthcare (from [1])

2.7 Summary

In this chapter we have introduced the theoretical background related to our work. Process mining concepts, existent techniques and tools were explored at extent. Previous work exploring the application of process mining in healthcare organizations were also studied. Theoretical findings were discussed focusing the topic of this research.

Related work by several authors suggests that process mining can be successfully applied to the analysis of healthcare processes [1, 19, 20, 21]. However, most of the techniques are not

useful to handle the complex and ad-hoc nature of these processes [1, 20]. The main challenges these techniques need to handle are: (1) the incompleteness and noise found in medical event logs [1, 19, 20, 21]; (2) the richness of process variants, that need to be clearly distinguished [1, 19, 21]; and (3) the presence of infrequent behavior of interest (e.g. exceptional clinical cases, medical errors, deviations from guidelines), that need to be captured and not disregarded. [1, 20, 21].

With respect to noise and the incompleteness of event logs, it is commonly accepted that deterministic approaches such as the alpha-algorithm and the Multi-Phase do not provide useful results. The Heuristic Miner is a good approach to handle noise [20] but the richness of process variants results in confusing models, making it impossible to distinguish exceptional medical cases [1, 19, 21]. Clustering techniques are highly recommended for these situations. The DWS Algorithm can provide some results [1], but they are not the most useful [19]; both Trace Clustering and the ARM techniques seem to perform better than the DWS Algorithm [19, 21]. The Fuzzy Miner has also been successfully used [21].

The usefulness of Sequence Clustering has not been demonstrated in real-life healthcare settings, but it seems to be an interesting approach, as it has already been successfully applied in other complex and ad-hoc environments [51, 52, 53, 54, 58] and it may perform better than the Trace Clustering [51]. Also, the result of Sequence Clustering is a mixture of Markov chains, which provides visual models for the different behavioral patterns of the process. In ARM, the behavioral patterns are in form of association rules, therefore they are presented in the form of statements rather than models. These can be harder to analyze. Therefore, it becomes necessary to study the application of Sequence Clustering to healthcare processes.

The organizational and performance perspectives of the process are less explored in healthcare applications. With respect to the organizational perspective, the Social Network Miner has been successfully applied in [20, 21]. The fact is that it becomes very important to understand and quantify the working relationships between physicians; this is very difficult, if not impossible, to achieve with traditional BPA. The usefulness of the Organizational Miner is still to be shown in real-life healthcare processes. With respect to the performance perspective, both the Basic Performance Analysis and the Dotted Chart seem to be useful to measure the performance of healthcare processes [20, 21].

Previous authors do not describe nor formalize a methodology for BPA in healthcare based on process mining. Rather, they tend to focus on a specific technique or a specific perspective of the process. We argue that it is important to develop such methodology, aiming at providing the community with an approach for a holistic analysis of healthcare processes, capable to address the three main challenges identified. Such methodology is devised in next chapter.

3

Methodology

In this chapter it is devised a methodology based on process mining for analyzing healthcare processes. Section 3.1 introduces the proposed methodology as an extension to the work of [2]. The extension is done by means of Sequence Clustering Analysis – a sub-methodology that can cope the large amounts of noise in medical logs and sort different behaviors (regular behavior, process variants and infrequent behavior). The rationale of the Sequence Clustering Analysis is explained in detail in Section 3.2. Section 3.3 relates the Sequence Clustering Analysis with the remaining steps of the methodology, namely the control-flow, performance, and organizational analysis. Section 4.4 concludes the chapter.

3.1 Proposed Methodology

The methodology we propose for BPA in healthcare is an extension to the work of [2]. In [2] the authors aims for a quick analysis of generic processes by using process mining techniques. This methodology, depicted in Figure 3.1, comprises: (1) the preparation of an event log; (2) log inspection; (3) control-flow analysis; (4) performance analysis; (5) organizational analysis; (6) transfer of results.

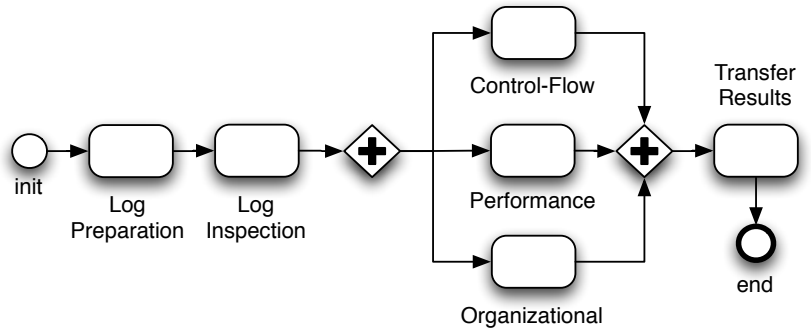

Figure 3.1: Bozkaya's methodology for BPA based on process mining (adapted from [2]).

Log preparation builds the event log by preprocessing event data gathered from information systems. Log inspection provides a first impression about the event log; it includes the analysis of statistical information such as the number of cases, the total number of events, the distribution of number of cases per number of events, the number of different sequences, the number of originators, etc. Then follows the analysis of the control-flow, performance, and organizational perspectives of the process, using the techniques described in chapter 2. The final step in the methodology is the transfer of results, where the knowledge acquired during the previous analysis steps is presented to the organization for validation.

For the purpose of our work, we are most interested in techniques that can handle the characteristics of healthcare processes, as described in chapter 1. Given those characteristics, we find that it becomes extremely important to study infrequent behavior and process variants. To do so, we extend the work of [2] with a new step after log inspection, see Figure **??**. In fact, this new step is a sub-methodology that includes a set of techniques to cluster the log and pre-analyze the process. The goal is not only to produce simpler models for the next steps, but also to systematize the analysis of process variants and infrequent behavior, as described ahead. The scope is limited to a set of techniques based on sequence clustering, which we use for two main reasons: (1) because we have seen that it is a good approach for the analysis of complex and ad-hoc processes; and (2) because we are interested in demonstrating the usefulness of this technique for the analysis of real-life healthcare processes.

3.2 Sequence Clustering Analysis

A detailed view of the Sequence Clustering Analysis step is presented in Figure **??**. It comprises: (1) running the sequence clustering algorithm; (2) building a diagram for cluster analysis; (3) understanding the regular behavior of the process; (4) understanding the process variants and infrequent behavior; and (5) performing hierarchical sequence clustering if needed. The next subsections describe each of these steps in more detail.

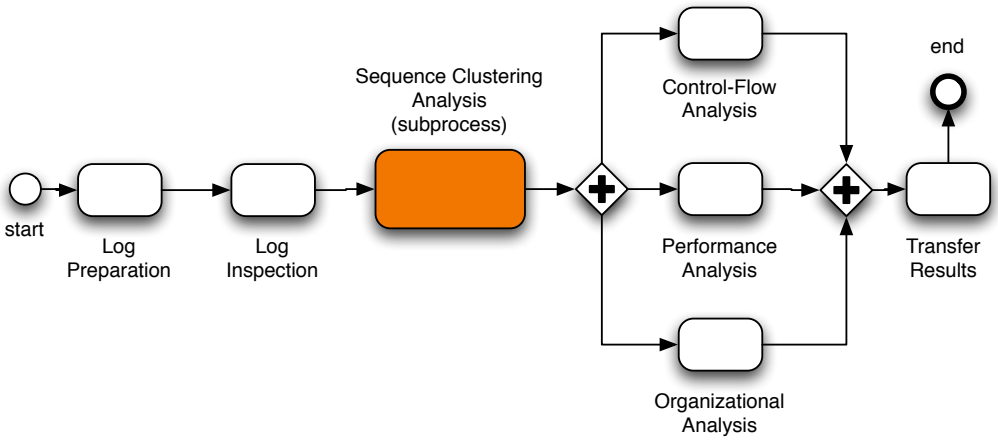

Figure 3.2: The proposed methodology for BPA in healthcare.

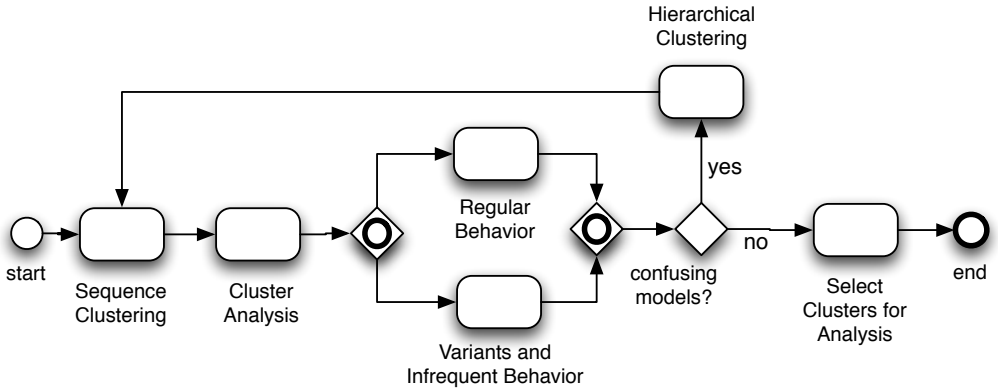

Figure 3.3: The Sequence Clustering Analysis sub-methodology.

3.2.1 Running the Sequence Clustering Algorithm

The first step is to run the sequence clustering algorithm as described in [51, 59] in order to discover the behavioral patterns contained in the event log. The resulting clusters will provide insight into the regular behavior, the infrequent behavior, and the process variants. To explain the concepts behind sequence clustering and how it works, at this stage we will consider a simple event log with n process instances and three types of sequences: AAAABC, AAAABBBBC, and CCCCBA. Figure 3.4, depicts one possible outcome of applying sequence clustering to such event log. There are three clusters, each cluster contains a set of sequences and is represented by a first-order Markov chain extracted from the behavior of those sequences.

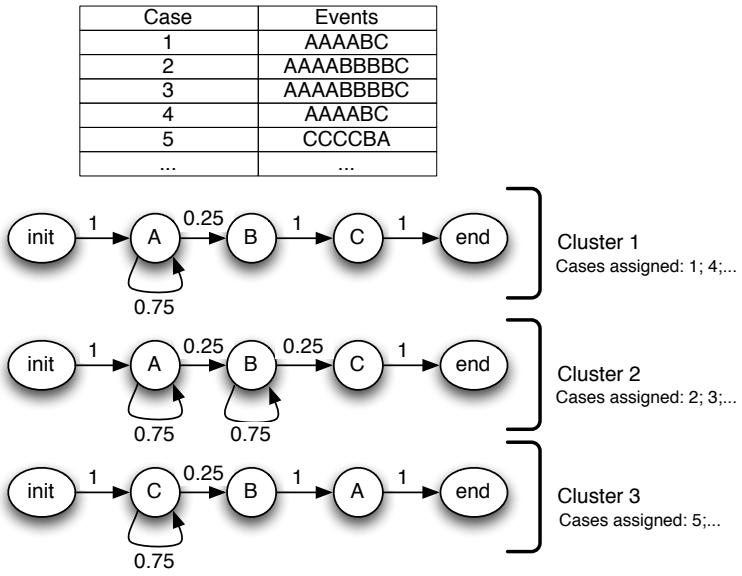

Figure 3.4: Example of the output of Sequence Clustering.

A Markov chain is defined by a finite set of N allowed states $S = \{S_1, \ldots, S_N\}$ and the Markov property, which means that the state at time t, denoted by $s(t)$, depends only on the previous state $s(t-1)$ and not on past states such as $s(t-2)$, $s(t-3)$, etc. This property is expressed by means of transition probabilities. In our context, the state-space S is given by the different tasks recorded in the log, augmented with two auxiliary states – the *init* and the *end* state – which are used to calculate the probability of a given task being the first or the last in the process. In the above example, $S = \{init, A, B, C, end\}$. Given some present task, we know what tasks can be executed next, and their probability. For cluster 1, from task A there is a probability of 75% to execute the same task A, and a probability of 25% to execute task B.

Mathematically, a Markov chain is represented as a $N \times N$ transition matrix where each element $M_{ij} = P(s(t+1) = S_j \mid s(t) = S_i)$ is the probability of the next state being S_j given that the present state is S_i. In addition, the following conditions hold: $\forall_{1 \leq i,j \leq N} : 0 \leq M_{ij} \leq 1$, and $\forall_{1 \leq i \leq N-1} : \sum_{j=1}^{N} M_{ij} = 1$. For example, the transition matrix for the Markov chain of cluster 1 is given by:

$$
\begin{array}{c} \\ init \\ a \\ b \\ c \\ end \end{array}
\begin{array}{c}
\begin{array}{ccccc} init & a & b & c & end \end{array} \\
\left(\begin{array}{ccccc}
0 & 1 & 0 & 0 & 0 \\
0 & 0.75 & 0.25 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0
\end{array} \right)
\end{array}
$$

The purpose of sequence clustering is both to discover the transition matrices and to assign each sequence in the event log to one of the available clusters. Given that clusters are represented by Markov chains, each sequence should be assigned to the cluster that can produce it with higher probability. For a sequence $x = \{x_1, x_2, \ldots, x_L\}$ of length L, the probability that this sequence is produced by the Markov chain associated with cluster c_k can be computed as:

$$P(x \mid c_k) = P(x_1 \mid init; c_k) \cdot \left[\prod_{i=2}^{L} P(x_i \mid x_{i-1}; c_k) \right] \cdot P(end \mid x_L; c_k) \tag{3.1}$$

where $P(x_i \mid x_{i-1}; c_k)$ is the transition probability of state x_{i-1} to state x_i in the transition matrix of cluster c_k. The quantities $P(x_1 \mid init; c_k)$ and $P(end \mid x_L; c_k)$ refer to the transition probabilities from the start state and to the end state, respectively. For the sequences AAAABC in the above example, the Markov chain of cluster 1 can produce this type of sequence with a probability of $P(A \mid init; c_1) \cdot P(A \mid A; c_1) \cdot P(A \mid A; c_1) \cdot P(A \mid A; c_1) \cdot P(B \mid A; c_1) \cdot P(C \mid B; c_1) \cdot P(end \mid C; c_1) = 1 \times 0.75 \times 0.75 \times 0.75 \times 0.25 \times 1 \times 1 \approx 0.1$, cluster 2 with a probability of approximately 0.03, and cluster 3 with a probability of zero (since $p(a \mid init; c_3) = 0$). Therefore, every trace in the form AAAABC is assigned to cluster 1.

Since the Markov chains are unknown at the beginning, the sequence clustering algorithm uses an iterative Expectation-Maximization procedure. The algorithm takes as input parameter the number K of clusters, and assumes an initial set of clusters $C = \{c_1, c_2, \ldots, c_K\}$. Each cluster c_k is a Markov chain on its own, and is represented by a transition matrix. The algorithm can be described as follows:

1. Initialize randomly the transition matrix of each cluster;
2. (Expectation step) - Assign each sequence in the event log to the cluster that is able to produce it with highest probability (using Equation 3.1);
3. (Maximization step) - Recompute the transition matrices of all clusters by considering the set of sequences assigned to each cluster in the previous step;
4. Repeat steps 2 and 3 iteratively until the transition matrices do not change; at this point the assignment of sequences to clusters also does not change anymore.

As a result of step 2 it may happen that some clusters do not get any sequences assigned to them. In this case, the final number of clusters will be less than the initially specified K.

3.2.2 Building a Diagram for Cluster Analysis

With the behavioral patterns distinguished in the previous step, the next step is to understand which clusters represent regular behavior, which ones contain variants, where is the infrequent behavior, and how much clusters differ from each other. For this purpose we extent the traditional approach of sequence clustering, by providing sequence clustering with means to perform a cluster analysis. The output of this step is a diagram that depicts the support of each cluster, i.e. how many sequences are assigned to each cluster, and the similarity between clusters, i.e. how much the Markov Chains differ from each other. Figure 3.5 presents the diagram for cluster analysis from our running example. The nodes represent the three clusters of the example and are labeled with the corresponding support. The edges represent the similarity between clusters, and are weighed with the corresponding similarity value. The higher the support the darker the nodes, and the higher the similarity the thicker the edges.

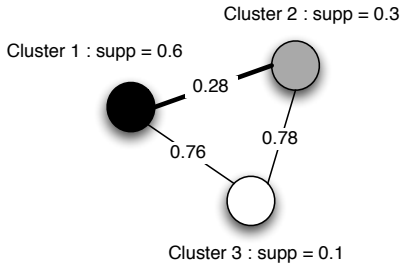

Figure 3.5: Cluster diagram for the running example

Formally, we define the support of cluster c_k as:

$$support(c_k) = \frac{\#sequences(c_k)}{\sum_l \#sequences(c_l)} \tag{3.2}$$

where $\#sequences(c_k)$ is the number of sequences contained in c_k, and $\sum_l \#sequences(c_l)$ is the total number of sequences assigned to all clusters, i.e. the total number of sequences in the event log.

One way to look at cluster similarity is to consider the "distance" between their Markov chains, i.e. the distance between two probability distributions. We use a metric proposed in [60] that also takes into account the distribution of the state-space of each cluster model. Let S_i be a state in the state-space of cluster c_k, and let $\mathbf{x}_r$ of length $|\mathbf{x}_r|$ be a sequence that is assigned to cluster c_k. Then the marginal probability of S_i in c_k is given by:

$$P(S_i)^{c_k} = \frac{\sum_r \#S_i(\mathbf{x}_r)}{\sum_r |\mathbf{x}_r|} \tag{3.3}$$

where the summation on r is over all sequences that belong to c_k, and $\#S_i(\mathbf{x}_r)$ counts the occurrences of S_i in each sequence $\mathbf{x}_r$.

The distance between two clusters c_k and c_l is defined as [60]:

$$D(c_k \parallel c_l) = \frac{1}{2} \sum_i \sum_j |P(S_i)^{c_k} \cdot M_{ij}^{(c_k)} - P(S_i)^{c_l} \cdot M_{ij}^{(c_l)}| \tag{3.4}$$

where $M_{ij}^{(c_k)}$ is an element in the transition matrix of cluster c_k. By computing the distance between all pairs of clusters in the cluster diagram, one can derive a $K \times K$ distance matrix $D_{kl} = D(c_k \parallel c_l)$, where $\forall_{1 \leq k,l \leq K} : 0 \leq D_{kl} \leq 1$.

Given the cluster support and the distance matrix, we build a diagram for cluster analysis. This diagram is an undirected weighted graph were the set of nodes is given by the set of K clusters, and each node is labeled with the corresponding cluster support. The edges are given by the distance matrix entries, and are weighted with the corresponding value, as in Figure 3.5.

Assuming that the event log contains 100 process instances with 60 sequences of type AAAABC, 30 of type AAAABBBBC, and 10 of type CCCCBA, and that these traces are assigned to cluster 1, cluster 2, and cluster 3 respectively, we have: $support(c1) = 0.6$; $support(c2) = 0.3$; $support(c3) = 0.1$.

To compute the cluster distances, we first calculate the state-space distribution of each cluster. In cluster 1, which contains only one type of sequence (AAAABC), we have $\#A(\mathbf{x}_r) = 4$, $\#B(\mathbf{x}_r) = 1$ and $\#C(\mathbf{x}_r) = 1$ for all sequences $\mathbf{x}_r$. Therefore, $P(A)^{c_1} = \frac{4 \times 60}{6 \times 60} = \frac{2}{3}$, and similarly $P(B)^{c_1} = p(C)^{c_1} = \frac{1}{6}$. Following the same procedure for other clusters, and calculating the distance matrix D_{kl} according to Equation (3.4), we have:

$$\begin{array}{c} \\ c_1 \\ c_2 \\ c_3 \end{array} \begin{array}{c} \begin{array}{ccc} c_1 & c_2 & c_3 \end{array} \\ \begin{pmatrix} 0 & 0.28 & 0.76 \\ 0.28 & 0 & 0.78 \\ 0.76 & 0.78 & 0 \end{pmatrix} \end{array}$$

With the support for each cluster and the distance between clusters, we draw the cluster diagram of Figure 3.5. This kind of diagram will play a key role in the next steps of the methodology.

3.2.3 Understanding Regular Behavior

The next step in the analysis is to understand the regular behavior of the process. This is given by the cluster with highest support. Therefore, one looks at the cluster diagram and inspects the Markov chain associated with that cluster. For example, in Figure 3.5 we see that cluster 1 is the one with highest support. By inspecting the Markov chain assigned to this cluster, we can describe the regular behavior of the process as follows: task A is always the first to be executed. When task A is executed, there is a probability of 0.75 that task A is executed again, and a 0.25 probability to execute task B. When B is executed it is certain that task C is the next to be executed. After the execution of task C the process ends.

In practical applications, there may be more than one cluster containing typical behavior, i.e. there could be several typical behaviors. The decision of identifying clusters as typical behavior depends to some extent on the application context and it may require domain-specific knowledge as well. In any case, the typical behavior will be contained in the clusters with highest support.

3.2.4 Understanding Process Variants and Infrequent Behavior

Process variants are alternative paths in the process that deviate from the regular behavior, or from some original model [61, 62, 63]. Once the clusters with highest support have been identified, which represent regular behavior, it is possible to consider the remaining cluster models as variants of the process. In the running example, clusters 2 and 3 are the variants of the process.

To gain a better understanding about the variants we propose the analyst to follow a stepwise approach. Basically, one should inspect all the clusters models by comparing each model with the closest ones. The closer a cluster is, the more similar is the corresponding Markov chain, and therefore it will be easier to identify how the process varies. At the same time, we are minimizing the total cost of analysis. For example, cluster 1 and cluster 2 in Figure 3.5 are the most similar. Comparing the Markov chains of both clusters (Figure 3.4) we see that they have only a small difference (in cluster 2 task B has a self-loop). On the other hand, the Markov chains from cluster 1 and cluster 3 are very different. If more clusters were present, it would be easier to understand the differences (and therefore how the process varies) by comparing cluster 1 with cluster 2, and cluster 3 with some other similar cluster.

This approach is particularly useful for large cluster diagrams and it is equivalent to finding the Minimum Spanning Tree (MST) [64] in the diagram. The MST of a connected, undirected, and weighted graph is a subgraph which: (1) is a tree, i.e. any two nodes are connected by one single path; (2) connects all nodes; and (3) has the lowest total cost, i.e. weight, between connections. Well-know algorithms to find MSTs are Prim's algorithm [65] and Kruskal's algorithm [66]. In Figure 3.6 we see an example of a undirected weighted graph with its corresponding MST highlighted. This graph is another useful diagram for cluster analysis. By looking at the MST we clearly distinguish the most similar cluster models, i.e. the Markov chains that have smaller differences. At the same time, one can efficiently visit every cluster, i.e. one can iteratively compare every Markov chain knowing that for each iteration the differences between them are minimized, and therefore easier to understand.

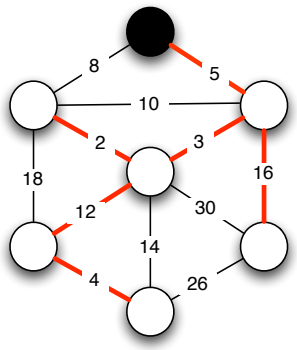

Figure 3.6: Example of a Minimum Spanning Tree.

Infrequent behaviors can be seen as special cases of process variants and are given by the clusters with lowest support. Eventually, these clusters will give the analyst insight about very

specific clinical cases, medical errors, deviations from clinical guidelines, or systems flaws, and they should be inspected carefully. In our running example, cluster 3 is considered as infrequent behavior, as it has a support of 0.1 which is relatively low. In this case, and attending to its distance to cluster 1, we can expect a very different behavior from the regular one. With further inspection, one finds that the Markov chain associated to cluster 3 models the opposite behavior of the regular one. In practice, this could represent for example some clinical exception.

3.2.5 Hierarchical Sequence Clustering Analysis

It may happen that some cluster models might still be hard to understand, and therefore not suitable for analysis [51]. To mitigate this problem one can apply hierarchical sequence clustering, i.e. re-applying sequence clustering to the less understandable clusters. The idea is that by hierarchical refinement we decrease the diversity of the sequences within a cluster, which eventually leads to the discovery of behavioral patterns. To do so, one must consider only the cases from the event log that have been assigned to the cluster under study, and then re-apply sequence clustering to that subset of cases.

3.3 Process Analysis

With sequence clustering analysis one has already addressed the following problems: (1) the incompleteness and noise of clinical event logs; (2) how to distinguish process variants; and (3) how to distinguish infrequent behavior (exceptional clinical cases, medical errors, etc). In essence, these are the major challenges that healthcare environments pose to process mining. For each cluster of interest, one can then proceed with the remaining analysis at the different perspectives of the process using the techniques already discussed in chapter 2. Of course, at this point we already have insight about the control-flow of the process, with Markov chains capturing some of the behavior. However, sequence clustering cannot discover some special workflow constructs, such as parallelisms or synchronizations in the process. The idea is therefore that one can explore the strengths of other control-flow mining techniques. Since each cluster contains traces that are similar in their sequential behavior, this should help other techniques produce more understandable and meaningful models as well.

3.4 Summary

In this chapter, it was defined a methodology based on process mining for analyzing healthcare processes. It was proposed an extension to the work of [2] in order to cope the main challenges of healthcare environments (as identified in chapter 2). It was defined a sub-methodology where sequence clustering plays a key role in identifying regular behavior, process variants, and infrequent behavior as well. This was achieved by means of a cluster diagram and a minimum spanning tree, which provide a systematic way to analyze the results. It remains the validation of the methodology in a real life scenario, which is addressed in the next part of this work by conducting a case study at Hospital of São Sebastião.

Part II

Practical Development (Case Study)

4

Case Study Preliminaries

This chapter introduces the case study conducted at HSS. Section 4.1 describes the organizational context, including the difficulties that the hospital lives with regard to the time consuming and costly nature of traditional process analysis approaches. Section 4.2 defines the scope of the case study, explains how the data was collected and presents the structure of the database considered. Section 4.3 describes the tool developed to support the different analysis conducted. Finally, section 4.4 concludes the chapter with a summary of the key points.

4.1 Organizational Context

HSS is a public hospital with approximately 310 beds, located in Santa Maria da Feira, Portugal. Since its creation, in 1999, it provides tertiary healthcare services for the 367.000 citizens of the north side of Aveiro's district. To do so, the hospital counts with 1017 professionals, including 164 doctors and 286 nurses. As with other hospitals, the main strategic goals of HSS include: (1) to deliver patient-centric services, in articulation with other hospitals of National Health Service, as well as with other primary and continuing care units; (2) improve the efficiency and quality of care services; and (3) reduce expenses of activity.

HSS makes use of an information system to support most of its activity. The system is known as Medtrix and has been developed in-house[1]. It provides an integrated view of all clinical information

[1]Microsoft has released international product and service brochures featuring HSS as a successful case study of an EPR system developed using Microsoft .NET Framework and related technologies.

of patients across the different departments. Medtrix is widely used in the hospital and it is regarded as a critical enabler for collaborative work among professionals.

The analysis of clinical and administrative processes is of great concern for HSS, especially when focused on detecting deviations from the main clinical guidelines, or flaws in the processes that Medtrix provides support to. Unfortunately, BPA has not been a common practice in HSS. Key stakeholders find BPA time-consuming and expensive, and they currently lack resources. A significant part of process improvement initiatives are delegated to the Department of Information Systems, as Medtrix is a leverage point to improve HSS processes. The department is run by eleven people from which only four are assigned to implement and maintain the Medtrix system, and also responsible for process analysis. The space of maneuver for process analysis is very tight. The main concern of executives and physicians is to see quick results from Medtrix, and do not conceive nor fully understand the idea of spending time and resources on a thorough BPA initiative. Moreover, the department runs under strong financial constraints and cannot afford external process analysts. The analysis of processes in HSS mainly results from discussions between representatives of specialized departments and the Medtrix team which, on their turn, are pressured to focus on system maintenance and development, rather than on analyzing processes. The process knowledge resulting from these discussions is usually not documented, and formal languages are not used. In conclusion, knowledge about the processes in HSS is tacit and unstructured, and given the characteristics of healthcare processes, the hospital needs more sophisticated tools and methodology to perform process analysis.

The potential knowledge that process mining can offer is of utmost importance for HSS, and the Medtrix system can provide valuable data for this purpose. The HSS staff embraced this possibility with great enthusiasm.

4.2 Setting the Scope and Gathering Data

After discussing with key stakeholders, it was decided to limit our scope to the operations of the emergency service, and activity comprising triage, treatments, diagnosis, medical exams, and transfer/discharge of patients. The main reasons for this decision were: (1) the perceived quality of HSS services is mainly based on the patients' opinion about the emergency service, therefore, it is a priority to understand the different processes related to the emergency service; (2) emergency processes, and the required interactions between physicians, are one of the most complex to actually understand; (3) the main concern of emergency services is performance and every analysis in this direction is welcome; (4) since Medtrix began by supporting the emergency processes, by integrating the emergency and radiology information systems, it represents a good opportunity to gather rather mature and useful data to build an event log. The next concern was how to get the event data.

In a discussion with the Medtrix team coordinator it was decided that the best approach would be to explore the data recorded in the system database. However, this was not as easy as it seemed. The Medtrix database currently contains more than 400 tables that are not documented.

The solution was to move the relevant part of this database to a new one that reflects the domain of the case study, as depicted in Figure 4.1. The new database contains event data of emergency processes, recorded from January to July, 2009.

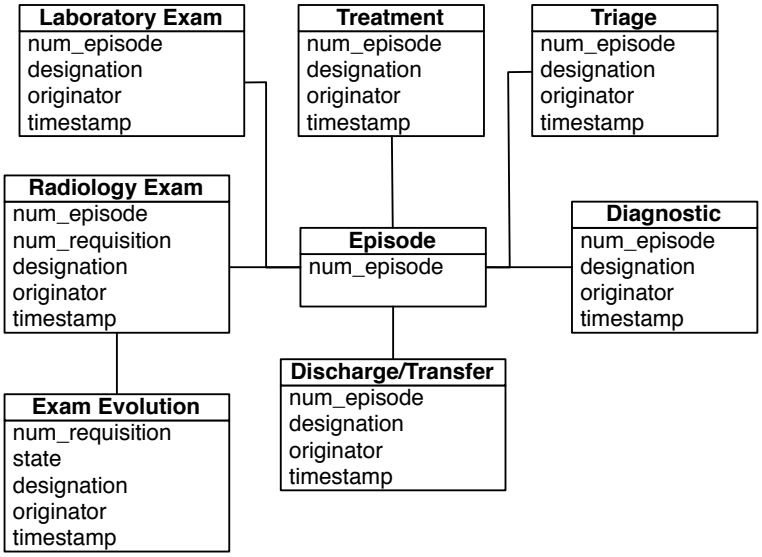

Figure 4.1: Scope of the case study.

The Emergency Episode table contains the emergency patients, i.e. the case identifiers, and the remaining tables represent possible activities performed on each patient. The Exam Evolution table is slightly different, since the activities in this table are given by the set of the possible states of the exam, such as exam scheduled, exam canceled, exam to be reported, etc. In each table we have information about the originator and the timestamp of events, which allows us explore the organizational and performance perspective of a process. The designation field makes it possible to explore the behavior of a process at different levels of abstraction and gives more flexibility to the analysis. For example, if we want to understand the process at a high level, we can simply use the activity name as the name of the corresponding table where the event has been recorded; one would then have activities such as "Triage", "Diagnostic", "Treatment", "Laboratory Exam", etc. On the other hand, if we want to study the clinical activities in more detail, one can use also the designation field, so that the activity names become more specific, such as "Diagnostic Thrombosis", "Radiology Exam X-Ray", etc. Besides enabling the generation of different event logs from the same data, this also makes it easier to analyze how patients with specific health conditions are handled.

4.3 Medtrix Process Mining Studio

To support the analysis of the emergency processes at HSS we developed a process mining tool called Medtrix Process Mining Studio (MPMS). This tool was designed to support the methodology described in Section 3. Also, to allow the use of ProM, we have developed a component to export

the event logs in MXML (Mining XML) format. The reason for developing a new tool, rather than simply using ProM, was that there was the need to make all the steps of the methodology available in a single application that could be readily used by the staff of HSS and that could be customized to this specific environment. Since HSS has a strong culture in developing systems in-house, MPMS increased the hospital adherence to this research project and opened doors to future projects.

The MPMS tool provides capabilities to inspect the log, perform sequence clustering analysis, inspect the cluster models and the cluster diagram, calculate the Minimum Spanning Tree, and facilitate the study of process variants by displaying the differences between cluster models. It also has some capabilities for creating, displaying and analyzing social networks extracted from event data. It should be noted that the purpose of MPMS was not to replicate the functionalities of ProM, so the user can still resort to ProM for more advanced control-flow, performance and organizational analysis.

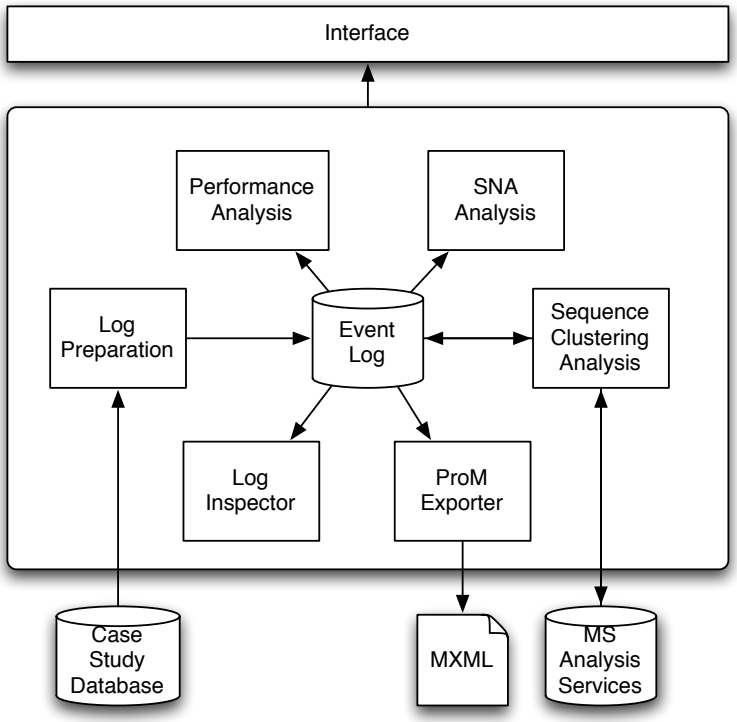

Figure 4.2: Architecture of the Medtrix Process Mining Studio.

Aligned with HSS' technological culture, MPMS is based on Microsoft's technologies. It is a Windows Presentation Foundation[1] application developed in C# that uses Microsoft SQL Server 2008 as database system. MPMS has five main components, each one supporting a different step of the methodology (see Figure 4.2): (1) log preparation; (2) log inspector; (3) sequence clustering analysis; (4) performance analysis; and (5) social network analysis. In the center we have the

[1]Documentation about Windows Presentation Foundation can be obtained at http://msdn.microsoft.com/en-us/library/ms754130.aspx

event log, which is internally represented according the MXML standard and shared among all components. The ProM Exporter exports the event logs to a MXML file, as already referred. The remaining of this section explains the main components of MPMS in more detail. Screenshots of each component are in Appendix A for the interested reader.

Log Preparation Component: This component is responsible for building the event logs, which is done via SQL queries over the database. Figure A.1, Appendix A, depicts the interface presented to the user. On the right side we have the information contained in the event log built as an example. An event log is built according the parameters given to the Event Log Builder Options (on the left side), from which the user chooses the data to include and the detail level of event names. There are also filters and other pre-processing options. An important feature is the possibility to filter patient cases according a specific diagnostic present in the database, i.e. it is possible to build an event log containing information of an *homogeneously diagnosed group* and direct the analysis to a specific clinical process. An example is the acute stroke process analyzed in chapter 6, which results from building an event log of patients homogeneously diagnosed with acute stroke. In summary, the log preparation component provides means for a quick and flexible extraction of event logs from the database, which can be immediately analyzed with the remaining components.

Log Inspector Component: After preparing the event log the log inspector component can be used in order to obtain statistical information about the log, such as the distribution of events, the distribution of cases per number of events, and several information about the sequences of events (such as the distribution of sequences, or the percentage of unique sequences). The latter provides insight about the complexity of the event log, from which one infer whether sequence clustering analysis is justified or not. The component mostly uses charts to present the information. The visualization of charts were implemented using Visifire controls[1]. Figure A.2, Appendix A, exhibits an example of one of these charts, depicting the distribution of events observed in the event log previously built.

Sequence Clustering Analysis Component: The sub-methodology described earlier in chapter 3 is implemented in this component. The component uses the Microsoft Sequence Clustering algorithm, implemented in Microsoft SQL Server 2008 Analysis Services [67]. The algorithm is invoked programmatically via a Data Mining API to extract the information about each cluster: the Markov Chains, the state-space distribution, and support. The cases assigned to each cluster are obtained by means of a Prediction DMX query[2]. Hierarchical Sequence Clustering can then be performed by filtering the event log and keeping only the cases that belong to a selected cluster. The distance metric from Equation (3.4) was implemented in order to build the cluster diagram. To find the Minimum Spanning Tree, Kruskal's algorithm was implemented as well. For the vi-

[1] Visifire is available at: http://visifire.com/

[2] The DMX (Data Mining Extensions) language is used to create and work with data mining models in SQL Server Analysis Services.

sualization of the Markov Chains, and the other graphs, NodeXL [68] was used[1]. Finally, it was implemented a technique that depicts the differences between two Markov Chains (Figure A.3 in Appendix A), which is useful to visualize and understand the variants of the process.

Performance Analysis Component: This component computes and show statistical information about the throughput time of the process, as well as the transition times observed for each cluster. The latter is computed attending the Markov property and the transitions of the Markov Chains are colored in order to detect bottlenecks within the cluster. For the sake of clarity, Figure A.4, Appendix A, exhibits an example of this feature. By selecting a cluster for performance analysis the component presents the Markov Chain of the cluster with transitions colored according the mean time between transitions and relatively to the maximum and minimum mean transition time. Transitions in green, yellow, and red show low, medium and high transition times, respectively. Looking at the colored Markov Chain one easily detects potential bottlenecks in the process.

Social Network Analysis Component: In order to analyze the organizational perspective of the process, the user can use this component to discover a social network based on the following metrics: handover of work, working together; or joint activities. These are same metrics implemented in the Social Network Miner of ProM, described in detail in [40]. The visualization and analysis of the discovered network is supported by NodeXL, which computes standard graph metrics for social network analysis, namely: degree, betweenness centrality, closeness centrality, eigenvector centrality, clustering coefficient, and graph density. These features were implemented in the component in addition to the graph clustering algorithm provided by NodeXL, described in [69]. A screenshot of the component is shown in Figure A.5 in Appendix A.

4.4 Summary

An introduction to the case study conducted at HSS was given in this chapter. The organizational context was described, alerting to the fact that for the people in the hospital it is crucial to have a good understanding of the clinical and administrative processes. However, there is no room to perform a traditional, manual process analysis via interviews (it would have been too time-consuming and the 11 people in the IT department are mainly concerned with the maintenance and development of the in-house system, which is highly demanding). Hiring external process consultants is also out of the question because of financial constraints.

After justifying the value of process mining to HSS, the scope of the case study was defined with key stakeholders and limited to the emergency service; more specifically, to activity regarding triage of patients, treatments prescribed, diagnosis made, medical exams performed, and transfer/ discharge of patients. The Medtrix system, developed in-house, presented a good opportunity to extract the data given the maturity of the system and the fact that it integrates several hospital services, namely the emergency service and the radiology service (which is also responsible for

[1] NodeXL is an open-source graph visualization and social network analysis tool; distributed as a Microsoft Excel 2007/2010 add-in and .Net libraries as well. NodeXL is available at: http://www.codeplex.com/NodeXL.

emergency radiology). However, extracting the data was not easy because Medtrix database currently owns more than 400 non documented tables. The case study database resulted from a time demanding joint effort between the researcher and the Medtrix team coordinator.

Finally, it was presented the tool developed in order to support the analysis of the processes contained in case study database. The main components and features of the tool were explained, including the capability to: generate several and distinct event logs from the data; inspect the log; perform sequence clustering analysis; inspect the cluster models and the cluster diagram; calculate the Minimum Spanning Tree; facilitate the study of process variants by displaying the differences between cluster models; compute the throughput time of the process; facilitate the detection of bottlenecks by coloring the transitions of the Markov Chains according the median transition times observed; create, display and analyze social networks from data according to some metrics like handover of work, working together and similar tasks. The tool also export the logs to the MXML format in order to integrate the analysis with ProM.

With the foundations of the case study settled it remains the analysis of the data gathered. In the next chapter we start with the analysis of the process of emergency radiology.

5

Analysis of the Emergency Radiology Process

The radiology service plays a central role on the daily operations of the emergency department. Whenever emergency physicians need medical imaging to diagnose a patient (x-rays, computed tomography scans, ultrasounds, etc) they send their request to the radiology service. When receiving a request, the latter performs a set of interrelated activities in order to provide accurate and timely results to physicians. We will name this particular set of interrelated activities as the emergency radiology process.

The emergency radiology process supports and coordinates different clinical processes that, at some point, run at the emergency service. Accordingly, it belongs to the class of organizational healthcare processes, which reflects one of the most important properties of this process: it highly impacts the global performance of the emergency service. Because of that it made sense and was of HSS interest to start by studying this process. It proved crucial to understand the performance issues discovered when analyzing the clinical process presented in chapter 6.

Using the data previously collected and following the methodology devised in chapter 3, the emergency radiology process is analyzed in this chapter. In section 5.1 we start by preparing and inspecting the event log. The sequence clustering analysis is then performed in section 5.2. The Markov Chains obtained will suffice to understand the behavior of interest and insight about control-flow, therefore, we do not explore other control-flow mining techniques. In section 5.3, we conduct the performance analysis step, focusing regular behavior and the specific path followed by computed tomography scans. Section 7.3 concludes the chapter with a summary and discussion

on the main findings, including the feedback of the radiology coordinator, advantages that process mining brings to the hospital, and points of improvement to make in future.

5.1 Log Preparation and Inspection

Following the methodology, we start by building an event log and further inspect it on a second step. To build the event log the MPMS Log Preparation component was used to get the information contained in the *Radiology Exam* and the *Exam Evolution* tables. The type of exam associated with each event was respectively included in the data field of the event log. It proved useful to relate the discovered behavioral patterns with the different types of exams. The resulting log comprised 27930 process instances[1], a total number of 179354 events, and 12 different types of events (for simplicity, let us name them as tasks or activities). Figure 5.1 shows the activities found and respective absolute frequencies. One sees the exam request and eleven possible states from which an exam can evolve.

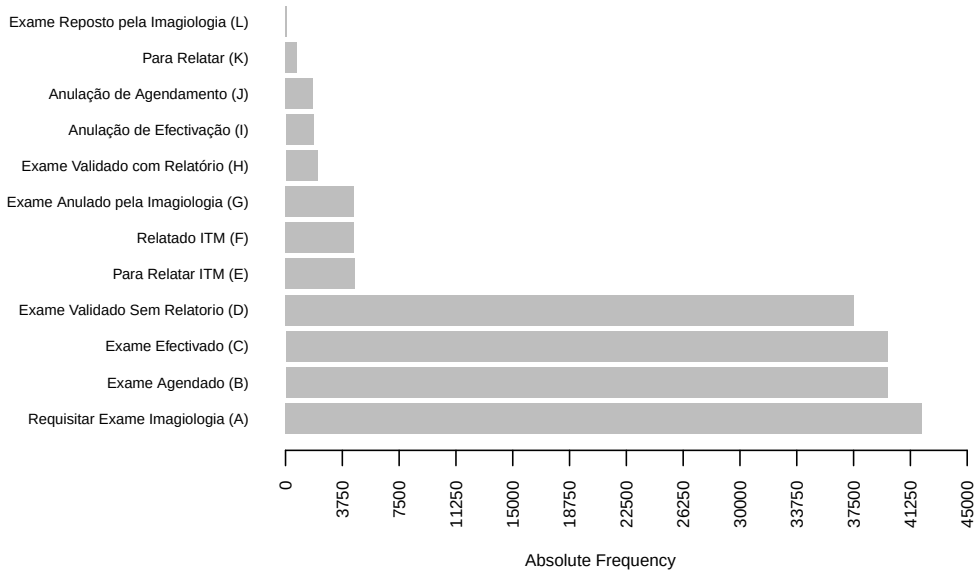

Figure 5.1: Emergency radiology Process - Tasks observed in the log and respective absolute frequencies.

[1]The Medtrix system assumes that a request can comprise more than one exam; therefore, a process instance may include one or more exams with the respective states interleaved in the sequence.

Inspecting the log, it was identified 2296 different sequences (from the total 27930). The top twelve sequences are represented in Figure 5.2 [1] attending their relative frequency. One sees a dominant sequence representing approximately 50 % of the process behavior and the remaining with a relative frequency below 0.06; 1820 sequences occurred only once. Since we are in presence of an organizational process, it was not surprising to find a dominant pattern in the run-time behavior. Notwithstanding, the nature of this process still creates a large degree of noise and diversity in the event log. As a result one can expect a confusing process model.

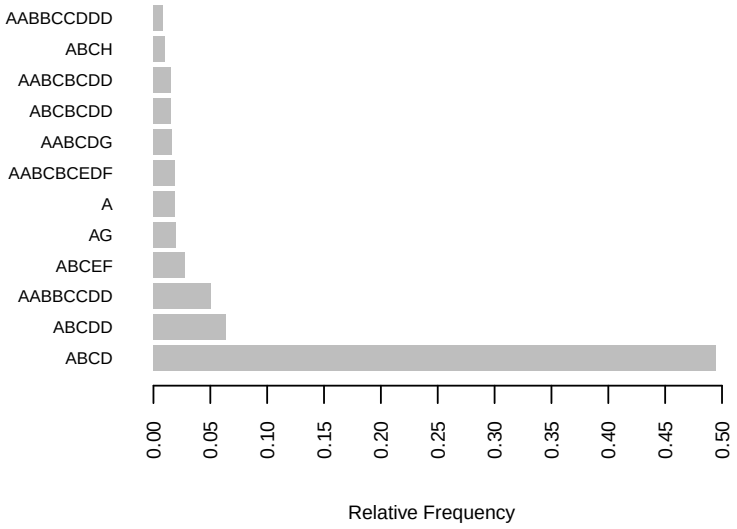

Figure 5.2: Emergency radiology Process - Top twelve sequences and respective absolute frequencies.

Figure A.7, Appendix B, shows a Petri Net modeling the global behavior of the process [2]. In spite of the low number of activities (represented by white rectangles), it was discovered a large number of transitions, specifically, the presence of so many silent transitions (i.e. different choices) after each step suggests that there are actually several variants of this process. A sequence clustering analysis is performed in order to discover these variants.

[1] Each letter in a sequence correspond to the activity with the same letter in Figure5.1

[2] The model results from converting a dependency graph, obtained with the Heuristic Miner, to a Petri-Net. Given the large diversity of sequences, it was not possible to mine a valid process model by directly applying the alpha-algorithm

5.2 Sequence Clustering Analysis

The results obtained from each step of the sequence clustering analysis are exhibited bellow.

Running the Sequence Clustering Algorithm: As a first approach it was explored a feature of Microsoft Sequence Clustering that suggests a number of clusters, and it resulted in an initial set of 22 clusters. However, this relatively high number of clusters actually made the analysis rather difficult. After some experimentation, we found that separating the data into 8 clusters provided more intuitive results, which were easier to analyze. In general, a too high number of clusters makes it difficult to identify typical behavior, since even small variations may cause similar variants to be scattered across several clusters with relatively low support. On the other hand, using a very low number of clusters will aggregate different behaviors in each cluster, producing cluster models that are too complex to interpret and analyze.

The Cluster Diagram: From the eight cluster models, it was obtained the cluster diagram depicted in Figure 5.3. One identifies cluster 2 as the dominant cluster with a support of approximately 0.5. As we will see, it refers to the dominant sequence found during log inspection. The remaining clusters refer to variants. Given the rather low support of cluster 1, 4, 6 and 8 they are labeled as infrequent behavior. With the diagram one also has a perception of the most similar clusters. It is possible to apprehend, for instance, that the process model of cluster 2 is relatively similar to the one of cluster 5 and 7. This is particularly useful to understand process variants, but before let us understand regular behavior.

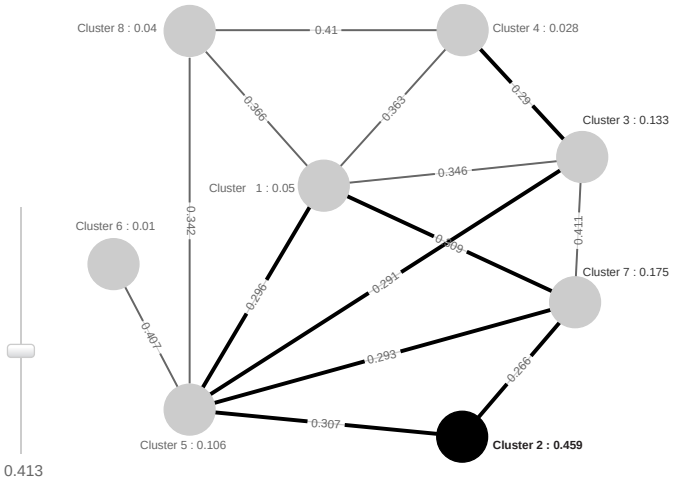

Figure 5.3: Emergency radiology Process - Cluster diagram obtained with eight clusters as input parameter (edges with a distance higher than 0.413 are hidden for better readability).

Understanding Regular Behavior: To acquire knowledge regarding regular behavior one inspects the Markov Chain associated with cluster 2 (the dominant cluster). Figure 5.4 presents the respective model. Whereas it was difficult to distinguish this kind of behavior by analyzing the global model we are now in presence of a simple sequential model that conforms with the dominant sequence previously found. Regular behavior is therefore explained as follows: (1) physicians start

by requesting an exam ("Requisitar Exame Imagiologia"); (2) the exam is then scheduled ("Exame Agendado"); (3) the exam is performed ("Exame Efetivado"); (4) the exam is validated without report ("Validar Exame Sem Relatório") and the process "ends".

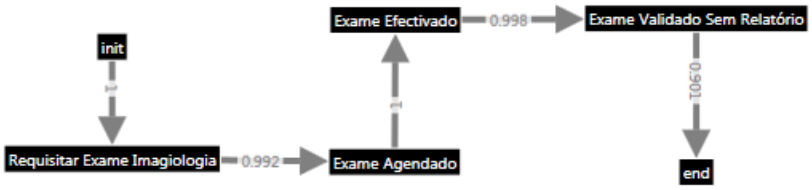

Figure 5.4: Emergency radiology Process - Process model of cluster 2 depicting regular behavior.

Further inspecting the types of exams associated with this cluster it was only found x-rays. Other interesting observation is that this cluster only comprises a single exam requests (otherwise one would detect a self-loop at the exam request task). In other words, the emergency radiology service handles a single x-ray request in 50% of cases, approximately. When presenting the results to the service coordinator the model was confirmed. It was surprising however the significance of this behavior. Despite knowing that x-rays are the most requested exams, there was the perception that single requests represented at most 20 to 30% of cases.

Understanding Process Variants and Infrequent Behavior: Process variants and infrequent behavior were analyzed with the aid of the Minimum Spanning Tree of the clusters diagram, see Figure 5.5. As we see, differences between Markov Chains are minimized by comparing: (1) cluster 2 with cluster 7; (2) cluster 7 with cluster 5; (3) cluster 5 with clusters 1, 3, 8, 6; and (4) cluster 3 with cluster 4. This logic was followed to understand how the process varies.

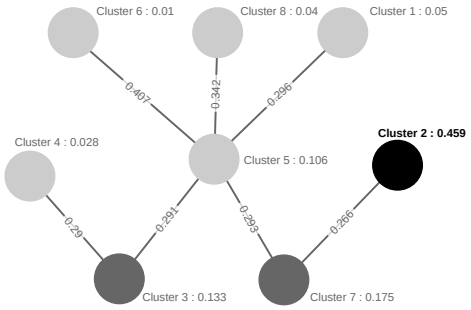

Figure 5.5: Emergency radiology Process - Minimum spanning tree of the cluster diagram.

Let us start by comparing cluster 7 with cluster 2. Figure 5.6 exhibits the Markov Chain of cluster 7. Tasks and transitions not observed in cluster 2 are marked in red. Thresholds are hiding

tasks and transitions with low probability, such that one can focus on the main differences.

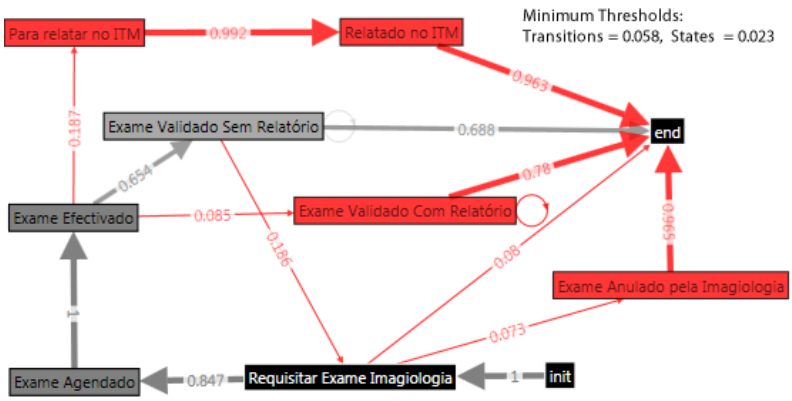

Figure 5.6: Emergency radiology Process - Process model of cluster 7 with differences from cluster 2 marked in red.

It is possible to see some paths in the model deviating from regular behavior, such as: (1) after the request of an exam the radiology service can cancel the exam (*Requisitar Exame Imagiologia => Exame Anulado Imagiologia*); (2) after the request of an exam, the process can end with no further developments on the exam state (*Requisitar Exame Imagiologia => end*); (3) after being performed an exam can be validated *with* report (*Exame Efetivado => Exame Validado Com Relatório*); and (4) after being performed, an exam can be set to report at the Institute of Telemedicine (ITM) and the process ends after ITM reports the exam (*Exame Efetivado => Para Relatar ITM => Relatado no ITM*).

Three paths to note:

1. When the process ends after the exam request and no tasks are observed in between. Since the end state is a dummy state, it means that an exam can "indefinitely remain" in a request state. It was explained that these cases refer to patients leaving the emergency department without warning, a common problem observed at the emergency service and of difficult control;

2. When the exam is sent to report at ITM. There are two important points: (1) only CT scans follow this path; and (2) ITM is an external entity that delivers radiology services. That is, even thought HSS performs CT exams internally these are reported by an external entity, i.e. HSS outsources part of the emergency radiology process;

3. When the exam is validated with report. This path refers to other specific exams (such as ultrasound exams) and was a surprising result. According HSS, the validation of an exam with report must always be preceded by the task *"Para Relatar"*, i.e. an exam must be set to report before validation of the report, meaning it should have been observed the path *Exame Efetivado => Para Relatar => Exame Validado Com Relatório* in all cases of exams validated with report. However, contrary to the perception of professionals, it is observed the

transition *Exame Efetivado => Exame Validado Com Relatório* without *Para Relatar* in the middle [1]; also note that, attending the observed absolute frequencies, as seen in Figure 5.1, the value of *Para Relatar / Exame Validado Com Relatório* is approximately 0.37, meaning that the expected behavior is not observed in 63% of the cases.

As already referred, there is other behavior in cluster 7 hidden in Figure 5.6 due low occurrence. More ahead we will use hierarchical sequence clustering to unveil and better understand this behavior.

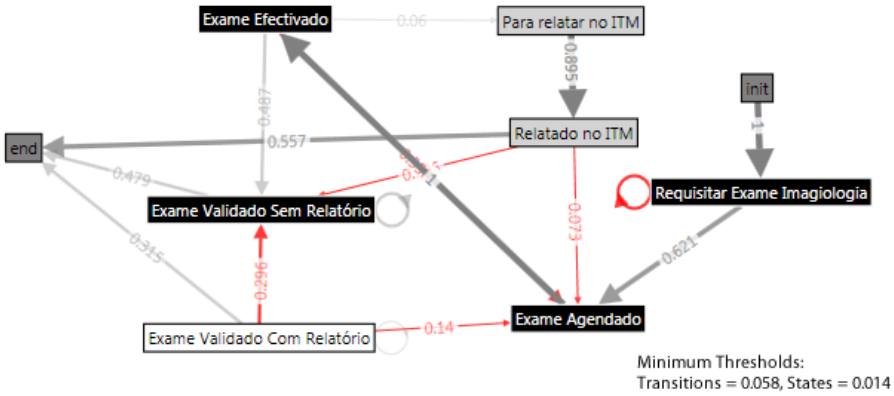

Figure 5.7: Emergency radiology Process - Process model of cluster 5 with differences from cluster 7 marked as red.

Now we compare cluster 7 with cluster 5. It revealed little significant differences, which was an expected result given the similarity value for these two clusters. These differences regard some transitions in cluster 5 not previously observed in cluster 7, see Figure 5.7. They refer to interleaved exams, such as: (1) when an exam is requested some other exam can be requested at next; (2) when an exam is performed some other can be scheduled at next; (3) after an exam validated with report, some other can be validated without report; (4) when an exam is reported by ITM, some other can be scheduled next, or validated without report.

The remaining comparisons were similar. They revealed interleaved behavior or stressed previous observations. For instance, the comparison between cluster 5 and cluster 1 only revealed a difference in one transition: after the validation of an exam without report, some other exam can be canceled. Also, comparing cluster 5 with cluster 3, it was easy to detect the absence of the task "Para Relatar" in cluster 3, whereas the task "Exame Validado Com Relatório" was present in both.

Let us however exhibit cluster 6. It unveils an infrequent yet very interesting behavioral pattern, see Figure 5.8. Transitions in red highlight a path where an exam starts by being scheduled. Note the request is observed after the exam performed. Because the process is supposed to start with

[1] The path *Exame Efetivado => Para Relatar => Exame Validado Com Relatório* is observed in some cases of the cluster but *Para Relatar* has such a relative low occurrence that it is hidden in Figure 5.6 due the minimum thresholds applied

the exam request, HSS people was caught in surprise with this result. This behavior was not being perceived till presentation and was of difficult detection given the low occurrence (it happened 131 times, representing approximately 0.5% of global behavior).

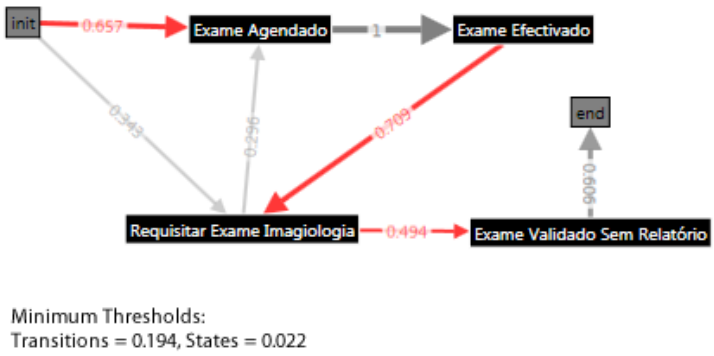

Figure 5.8: Emergency radiology Process - Process model of cluster 6 (example of an infrequent pattern unveiling deviation from protocols).

With main process variants and infrequent behavior presented, let us explore the concept of hierarchical sequence clustering.

Hierarchical Sequence Clustering: To not loose focus we will apply the concept to cluster 7. It contained other behavior that we hid with thresholds due low occurrence. Now, Figure 5.9 shows the model of cluster 7 without thresholds. It unveils tasks with low frequency (represented by white rectangles) and respective in/out transitions. Other low frequent transitions previously hid are also revealed.

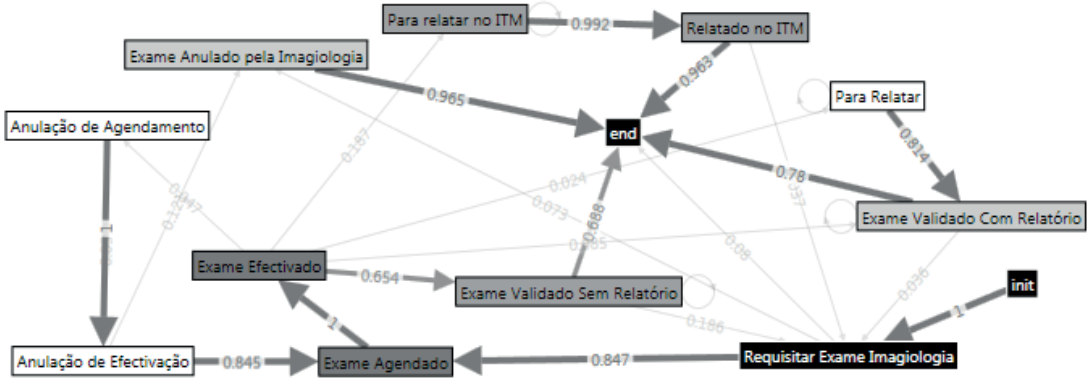

Figure 5.9: Emergency radiology Process - Process model of cluster 7 without thresholds.

Hierarchical sequence clustering has a two fold application: (1) it may further reduce the complexity of a certain cluster; (2) it may distinguish other specific or less frequent behavior of possible interest. This was confirmed when applying the concept to cluster 7. To do so, the sequence clustering algorithm was run over cluster 7 cases with four clusters as input parameter.

Figure 5.10 presents the resulting models that gave better insight about this particular cluster and the process. In essence:

- Cluster 7.1 better depicts the cases of exams validated with report (*"Exame Validado Com Relatório"*). It also stresses the problem when the task *"Para Relatar"* is skipped.
- Cluster 7.2 clearly distinguishes exams reported by ITM, as already discussed;
- Cluster 7.3 stresses the situations when an exam is requested and there is no further developments on its state. However, it is now clear that the radiology sometimes cancel the exam (*"Exame Anulado pela Imagiologia"*);
- Cluster 7.4 depicts infrequent cases in which the scheduling of the exam is canceled (*"Anulação Agendamento"*). The interesting aspect is that it happens after the execution of *"Exame Efetivado"*, i.e. after the exam already performed. Hence, after *"Anulação Agendamento"*, the exam already performed is canceled as well (*"Anulação de Efetivação"*). Then, the exam is rescheduled, i.e. is set again to *"Exame Agendado"*.

Summary: Sequence clustering was able to deal with the noise and large diversity of traces present in the event log. It was possible to discover and understand regular behavior, variants, and infrequent behavior. It was detected specific examination cases (such as the exams reported by ITM) and behavior deviating from what it was perceived.

From each cluster it was obtained understandable process models that, when compared with the global model, gave better insight about behavior. Given the simplicity of the most relevant clusters we will not explore other control-flow mining techniques with stronger semantic expressiveness. This topic will become relevant and will be better understood during the analysis of the stroke process introduced in chapter 6. For the interested reader, in Appendix B are the examples of Petri Nets obtained from cluster 2 and cluster 7.2. Both are similar with the corresponding first-order Markov Chains but are of interest for a direct comparison with the global model presented in Figure A.7. Having said that, we proceed with the analysis of the performance perspective of the emergency radiology process.

5.3 Performance Analysis

Given the results obtained with the sequence clustering analysis, the radiology coordinator showed interest in knowing: (1) throughput time and bottlenecks of regular behavior and CT exams cases; and (2) whether CT reports are available within one hour, as they are supposed to. In order to provide these answers the cases associated with cluster 2 (regular behavior) and cluster 7.2 (CT exam cases) are focused.

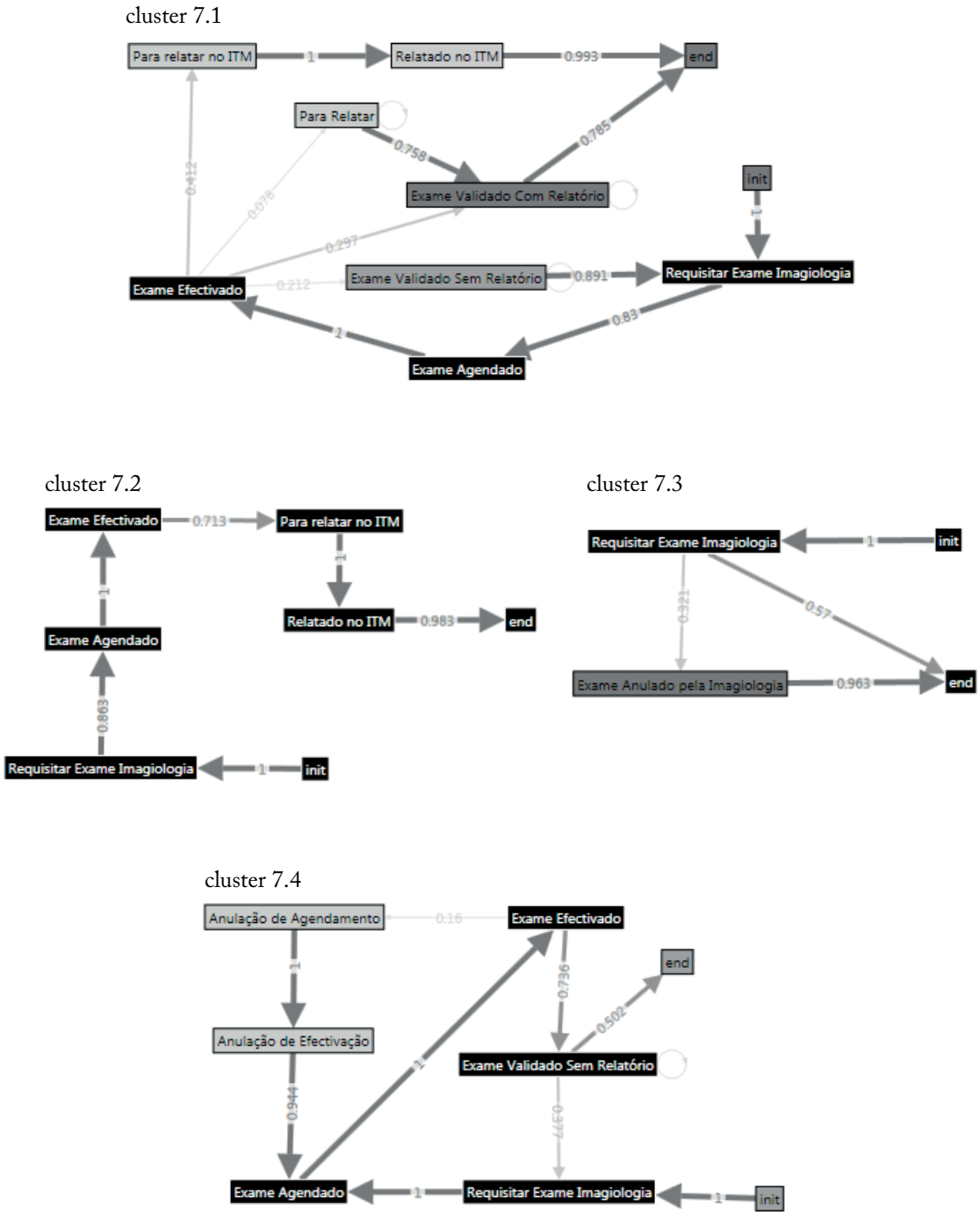

Figure 5.10: Emergency radiology Process - Result of applying hierarchical sequence clustering to cluster 7 (four clusters as input parameter).

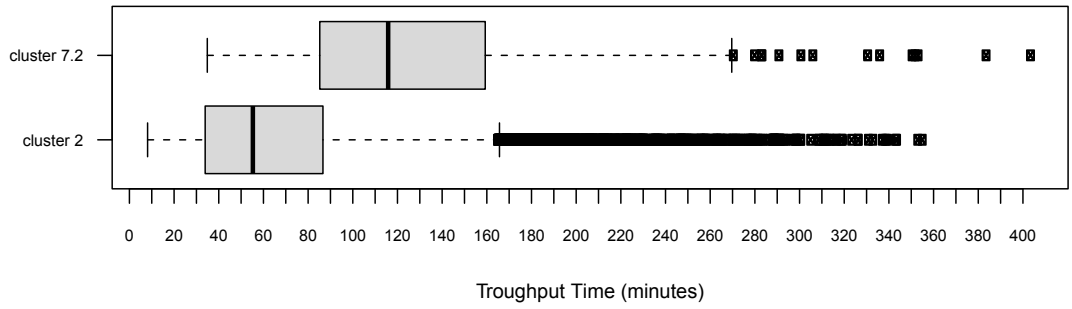

Figure 5.11: Emergency radiology Process - Boxplots depicting throughput time observed for cluster 2 (regular behavior) and cluster 7.2 (CT exams).

		Throughput (*minutes*)					
cluster c_k	n	*min*	$Q1$	*median*	*mean*	$Q3$	*max*
2	13805	8.13	33.92	55.27	66.92	86.64	355.90
7.2	762	34.87	85.22	115.80	127.00	159.30	404.40

Figure 5.12: Emergency radiology Process - Descriptive statistics of throughput time observed for cluster 2 (regular behavior) and cluster 7.2 (CT exam cases).

Starting with throughput: both Figures 5.11 and 5.12 are elucidative of the throughput time of cluster 2 and 7.1 [1]. In spite of the high spread in observations, and the long upper tails of both distributions (showing preoccupying throughput times), it is clear that CT exams (cluster 7.2) have a significantly higher throughput than regular behavior (cluster 2). According the median, it takes approximately 55 minutes to handle typical emergency radiology cases, i.e. single x-ray exam requests, and approximately 116 minutes to handle CT requests, i.e. one extra hour.

Bottlenecks were detected by coloring each transition of the Markov Chains attending the respective mean transition time, as described in section 4.3 of chapter 4. Results are exhibited in Figure 5.13. Green, yellow, and red transitions regard low, medium and high mean transition time, respectively. Figure 5.14 depicts the boxplots [2] of yellow and red transitions, i.e. the potential bottlenecks.

Concerning typical cases (cluster 2), and averagely speaking, the transition taking the longest time is: *"Exam Request => Exam Scheduled"* (*"Requisitar Exame => Exame Agendado"*); with an average transition time of approximately 28 minutes. The distribution is right-skewed and long-tailed; the upper quartile ranges up to 56 minutes, the upper whisker up to 119 minutes, and it was

[1] The data was obtained from the MPMS performance analysis component and analyzed with R statistical suite. R can be obtained at `http://www.r-project.org/`, where it is also found extensive documentation and scientific publications.

[2] The outliers detected at upper values were removed in order to improve readability, however, the maximum values are referred on the side.

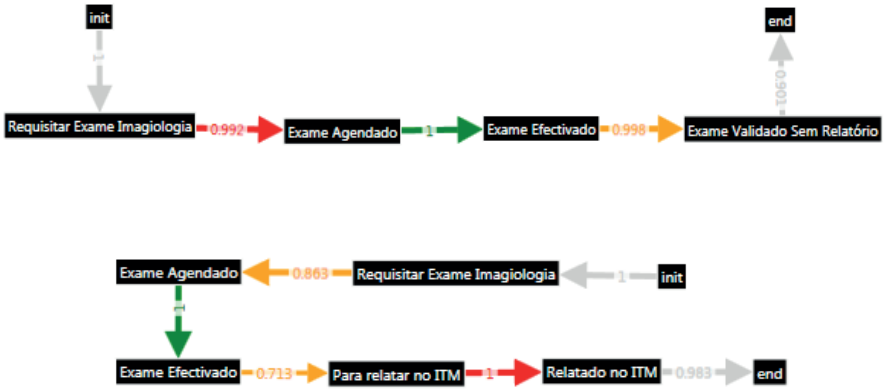

Figure 5.13: Emergency radiology Process - Transitions of cluster 2 (top) and cluster 7.2 (bottom) colored according the respective mean transition time.

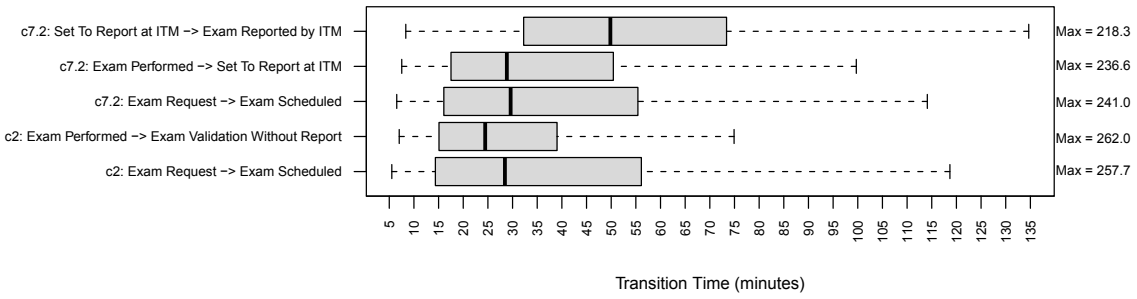

Figure 5.14: Emergency radiology Process - Boxplots depicting bottlenecks detected at cluster 2 (regular behavior) and cluster 7.2 (CT exams).

observed a maximum value of 258 minutes, approximately. The transition *"Exam Performed => Exam Validation Without Report"* (*"Exame Efetivado => Exame Validado Sem Relatório"*) shows similar average time, but with lower upper quartile and upper whisker range; the first ranges up to 39 minutes and the second up to 75 minutes, approximately.

What regards CT exams (cluster 7.2), let us focus the transition *"Set To Report at ITM => Exam Reported by ITM"* (*"Para Relatar no ITM => Relatado no ITM"*), referring to the outsourced part of the process. As Figure 5.14 shows, this transition has the highest median transition time (approximately 50 minutes). Moreover, the first quartile (approximately 32 minutes) is above the median values of the remaining transitions. The transition also presents the highest upper quartile and upper whisker values; 73 and 134 minutes, respectively. In other words, the time taken to report CT exams explains the one hour difference.

Since *"Set To Report at ITM => Exam Reported by ITM"* let us understand whether CT reports are available within one hour (as supposed), this transition was of particular interest for the service coordinator. Despite we already saw the one hour maximum is not always respected, both Figures 5.15 and 5.16 describe the distribution in more detail.

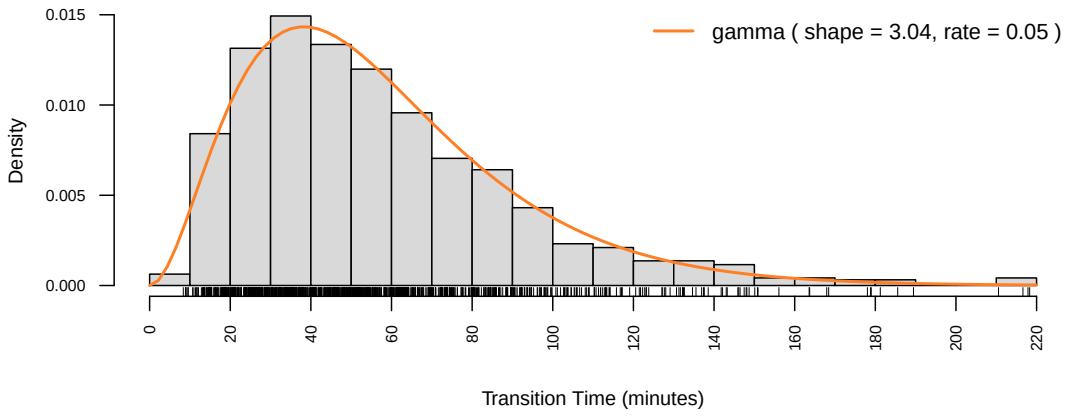

Figure 5.15: Emergency radiology Process - Histogram and theoretical density function regarding the transition times observed for: *Set To Report at ITM => Exam Reported by ITM.*

In Figure 5.15 one sees the histogram and a theoretical density function describing the transition times observed for *Set To Report at ITM => Exam Reported by ITM*. It is reasonable to assume the distribution follows a gamma law with $shape \approx 3.04(-0.35, +0.27)$ and $rate \approx 0.05(-0.002, +0.009)$ minutes [1]. Figure 5.16 depicts the empirical and theoretical cumulative density function (CDF). As the latter shows, the transition time exceeds the expected 60 minutes in

[1] Several distribution families were tested, the gamma distribution presented the lowest Anderson-Darling statistic (≈ 1.12). Parameters were estimated by Maximum Likelihood Estimation. Values in parenthesis regard the confidence interval at 95% level, relatively to the parameter. Tests and computations were performed with *fitdistrplus* R-package [70], implemented according [71] and [72].

40% of cases, exceeds the 80 minutes in 20% of cases, and exceeds the 100 minutes in 10% of cases, approximately.

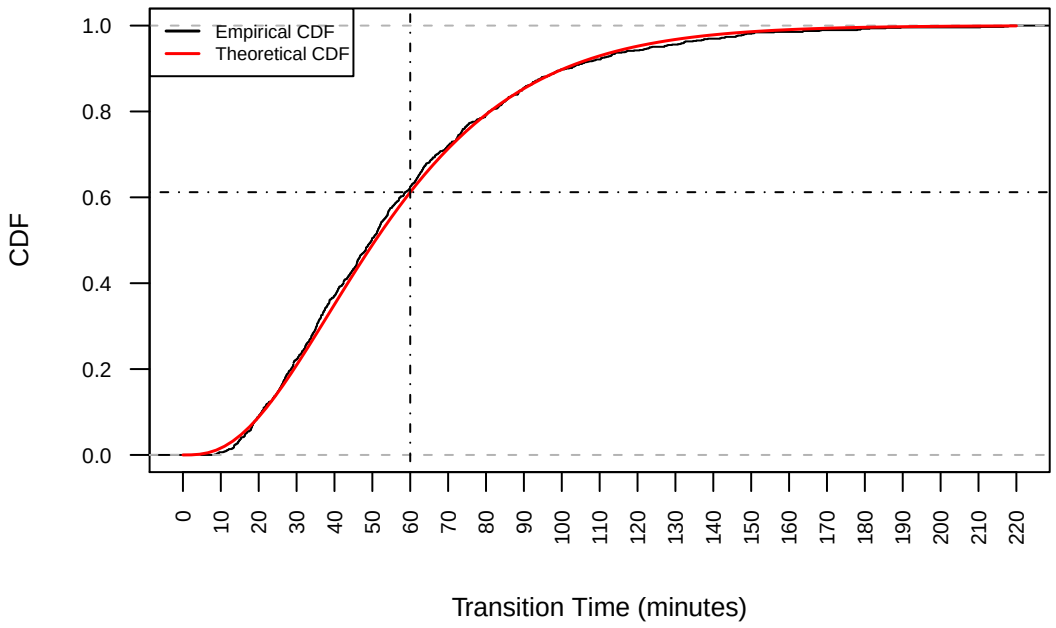

Figure 5.16: Emergency radiology Process - Empirical and theoretical cumulative density function regarding the transition times observed for: *Set To Report at ITM => Exam Reported by ITM.*

5.4 Summary

In this chapter, process mining and the proposed methodology were successfully applied to understand the control-flow and performance perspectives of the emergency radiology process as observed by the Medtrix system. Sequence clustering proved useful to handle noise and large diversity of log traces. It was possible to discover and understand regular behavior, variants, and infrequent behavior. It was detected specific examination cases and behavior deviating from what it is perceived by people. From each cluster it was obtained understandable process models that, when compared with the global model, gave better insight about behavior.

The next sub-chapters summarize, discuss main findings of the analysis, advantages that process mining brought to the department, as well as points of improvement to make in future.

5.4.1 Main Findings

What regards behavior, main findings can be summarized as follows:

- It was found a dominant behavioral pattern accounting for approximately 50% of the observed behavior: *"init => Request Exam => Schedule Exam => Perform Exam => Validate Exam Without Report => end"*. It was not surprising to find a dominant pattern since we are dealing with an organizational healthcare process, which implies the presence of generic, more structured process patterns. Neither it was surprising for the service coordinator when the model of the pattern was presented to him.

- Further inspection revealed that the above pattern concerns the workflow of single x-rays requests. Despite knowing that x-rays are the most requested exams, the service coordinator had the perception that single requests represented at most 20 to 30% of cases and not 50% as discovered. This is an interesting example showing that there can be deviations from what is perceived by professionals and what actually happens, even when on the subject of typical behavior.

- Although it was discovered a regular pattern with significant support, the nature of the emergency radiology process still creates a large degree of noise and diversity in the event log (resulting on an over complicated global process model). This problem was tackled with the methodology proposed in this work, from which it was possible to discover and analyze variants of regular behavior and infrequent behavior of particular interest.

- It was found process variants illustrating: paths followed by specific exams, such as CT scans and ultrasound exams; cancellations of exam requests/scheduling; and patient leaves without warning. The process model illustrating the path followed by CT exams showed that HSS outsources the reporting of these specific exams no an external entity (ITM). This behavior was confirmed and it was further told that ITM is expected to deliver the report within one hour. The path followed by exams validated with report (such as ultrasounds) revealed that the task "Para Relatar" is not being observed for all cases (as it is was supposed to), but only for 37% of cases.

- What concerns infrequent behavior, it was discovered a pattern where the exam starts by being scheduled, which is then performed and only then the request is observed: *"init => Schedule Exam => Perform Exam => Request Exam => Validate Exam Without Report => end"*. The coordinator was caught in surprise with this result since the workflow of an exam is expected to start at *"Request Exam"* state. This behavior was not being perceived till presentation of results and was of difficult detection given the relative low occurrence observed.

The performance analysis focused on computing and studying throughput and bottlenecks of regular behavior and CT cases, as well as in discovering whether CT reports are delivered within the expected hour. It was found that:

- The observed median throughput to handle typical exam requests is of approximately 55 minutes, and it takes one extra hour to handle CT requests (according to the median). Both regular behavior and CT exams cases presented high spread in observations and long upper tails on the throughput distribution, showing preoccupying times.

- For typical cases, there is a potential bottleneck at *"Request Exam => Schedule Exam"*, the transition taking the longest time with a median value of approximately 28 minutes. For CT cases, the bottleneck is at *"Set To Report at ITM => Exam Reported by ITM"*, with a median value of 50 minutes and which refers to the outsourced part of the process.

- The service level agreed with the external entity regarding CT report delivery within one hour was respected in 60% of the observed cases.

- The performance results with regard to throughput and bottlenecks were not surprising for the service coordinator, arguing they reflect the problem that the radiology service has in providing response to the current demand. The hospital was already expanding the infrastructure of the radiology service in order to improve performance, even though the coordinator referred that exams requests could be better rationalized as means to decrease demand and improve overall efficiency. The observed amount of CT reports that are not delivered within the expected hour revealed a preoccupying issue of which the coordinator was unaware; but this can be controlled more carefully from now on.

5.4.2 Advantages

The process mining setting developed at HSS and the results obtained in this chapter proved of extreme value for the organization. Despite the limited resources and effort HSS allocates to perform process analysis, the hospital has now means to be always in control of the radiology process based on objective data. Most importantly, the cost of analysis was reduced to an extent that it is now feasible for the organization to perform it whenever is needed. For example, the radiology department has now means to quickly analyze the process after the investment on infrastructure expansion is completed and compare it with the before; being able to benchmark without incurring with prohibitive costs.

Other important advantage regards the objectiveness of process mining. As this chapter showed, there are deviations from what people think it is happening and what really happens. In these cases, a traditional analysis based on people beliefs would not be accurate. In this context, process mining also contributes to objective organizational learning in a sense that it changes the mental models of people when they are presented with objective results.

Finally, as pointed out by the radiology coordinator, results are obtained and communicated attending process oriented perspectives of the organization, which is important to aware and educate professionals about the importance of process oriented thinking, which most of them miss in practice.

5.4.3 Points of Improvement

In spite of the advantages referred, there is space for improvement and challenges to overcome. One of the limitations concerns the usability of process mining to non-experts. The service coordinator showed great interest in using the process mining setting implemented, but the concept was found very technical and results difficult to achieve by non-experts. This clearly yields a barrier for effective internal use. Usability has to be improved to a minimum point at which professionals can be trained without needing deep technical understanding, otherwise the hospital is forced to invest in specialists, which is unlikely to happen given the economical panorama and organizational culture.

Other issue regards the understandability of results. The process formalisms devised, such as Petri Nets and first-order Markov Chains are not easily interpreted by non-experts. Heuristic graphs are better interpreted but, once more, to achieve those results is difficult for non-experts. It is important to explore process models more natural to health professionals while capable of modeling uncertainty and dealing with the complex medical logs with diverse features and high degree of heterogeneity in log traces.

One pertinent question that was asked is if it would be possible to incorporate cost analysis and control, which would be valuable for a fine grained understanding of how much the department actually spends, and benchmark different practices and process flows. Process mining currently lacks of techniques to support discovery and analysis of costs, but Activity Based Costing concepts is a natural fit for the problem. A feasible solution would be to create a pool of activity costs. Given an observable set of tasks of interest, each task would be mapped to a precomputed average cost. Human effort would be needed to create and maintain this pool of costs, and results would be based on averages, therefore, impacting accuracy. Nevertheless, it would be an important, feasible first step.

A more advanced solution would be to automatically devise costs based on resources spent by each task. There is however the need to study the feasibility of this solution in practice, mainly because of costs based on task duration. With the current data acquired, and logging capabilities of Medtrix, it is not possible to compute the duration of tasks, therefore, one can not automatically compute costs based on task duration. It is not possible, for example, to automatically compute how much cost it represents the time required by a professional to perform a task.

From a technical viewpoint, and regarding the methodology used in this study, there is also space for improvement; particularly, with the initial selection of clusters. In general, a too high number of clusters makes it difficult to identify typical behavior, since even small variations may cause similar variants to be scattered across several clusters with relative low support. On the other hand, using a very low number of clusters will aggregate different behaviors in each cluster, producing cluster models that are complex to interpret. Achieving this trade-off manually is time consuming and requires some domain expertise. On the other hand, the off-the-shell Microsoft's algorithm to automatically suggest the number of clusters did not provide satisfactory results.

There is therefore the need to develop proper heuristics to provide an adequate indication for the number of clusters.

In next chapter follows the analysis of the acute stroke process, a clinical process highly impacted by the performance of the emergency radiology process here analyzed.

6

Analysis of the Acute Stroke Process

Acute stroke is a medical emergency related with the loss of brain functions caused by interruption of blood supply to any part of the brain. It can be classified according two major and very distinct categories: (1) ischemic, when due blood blockage; or (2) hemorrhagic, when due blood leakage. Stroke remains one of the leading causes of mortality, hospitalization, and chronic disability worldwide. Due these reasons, the efficient management of stroke victims is usually a concern for receiving hospitals.

HSS has the following goals for stroke cases: (1) immediate room triage, clinical, laboratory and imaging evaluation; and (2) accurate diagnosis, therapeutic decision and appropriate patient transfer/discharge. Underlying these goals there is a *clinical process* driven by difficult and sensitive medical decisions that must be made as quickly as possible. Let us call this process the acute stroke process. The aim of this chapter is to analyze the acute stroke process at HSS using the data collected and following the methodology proposed. The results obtained are compared with guidelines [73, 74, 75], as well as validated with clinical knowledge of a HSS specialist.

The remainder of this chapter is as follows. In Section 6.1 the event log is prepared and inspected. Section 6.2 proceeds with sequence clustering analysis in order to understand different behavioral patterns. It reveals: most frequent behavior; different variants in the process (observed due different stroke complications); and infrequent behavior (hiding rare cases and deviations from guidelines). Given the importance of time in stroke management, section 6.3 analyzes the performance perspective of the process. Section 7.3 concludes the chapter with a summary and

discussion on the main findings, including the feedback of a specialist, advantages, and points of improvement to make in future.

6.1 Log Preparation and Inspection

The event log was built by exploring the MPMS Log Preparation feature that filters the case study database according a specific diagnostic. Attending our purpose, acute stroke was selected. Because handling stroke cases can be seen as a clinical process, we are essentially interested in medical decisions/actions taken by professionals. Given our context, we stress: emergency triage; requests of radiology and laboratory exams; treatment prescriptions; and patient discharges/transfers. To do so, it was included all the data contained in the following tables: *Triage*; *Diagnosis*; *Laboratory Exam*; *radiology Exam*; *Treatment*; and *Transfer/Discharge*.

Event names were selected as detailed. With this we will analyze the process behavior at a fine-grained level; useful to unveil medical errors, detect unnecessary examinations, distinguish patients with specific complications (which reflects different medical decisions), etc.

The resulting log contained 83 instances, a total number of 639 events and 48 distinct types of events (let us name them as activities for simplicity). Figure 6.1 presents the top twenty activities according their absolute frequency. It gives the reader an idea about the main kinds of triages, exams requests, drugs prescriptions and discharges/transfers found in the event log.

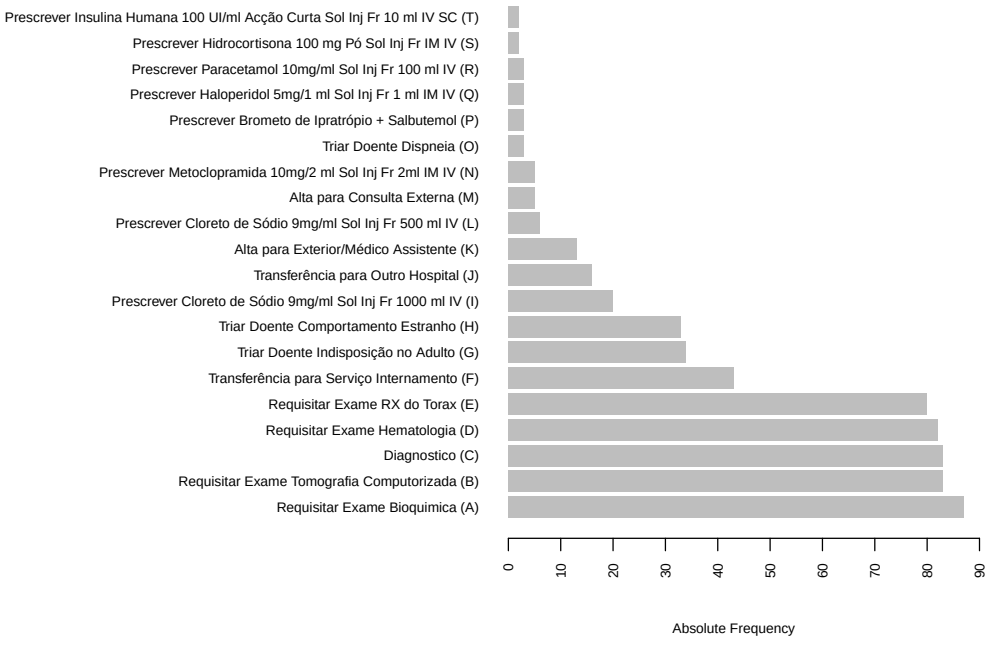

Figure 6.1: Top twenty events present in the stroke log and respective absolute frequencies.

To insight the reader about the log complexity, from the 83 sequences of activities observed

there are 55 different sequences, with 46 occurring only once. It corresponds to a percentage of unique behavior of approximately 55%, which indicates a high degree of ad-hoc behavior. Since we are dealing with a clinical process, this was expected.

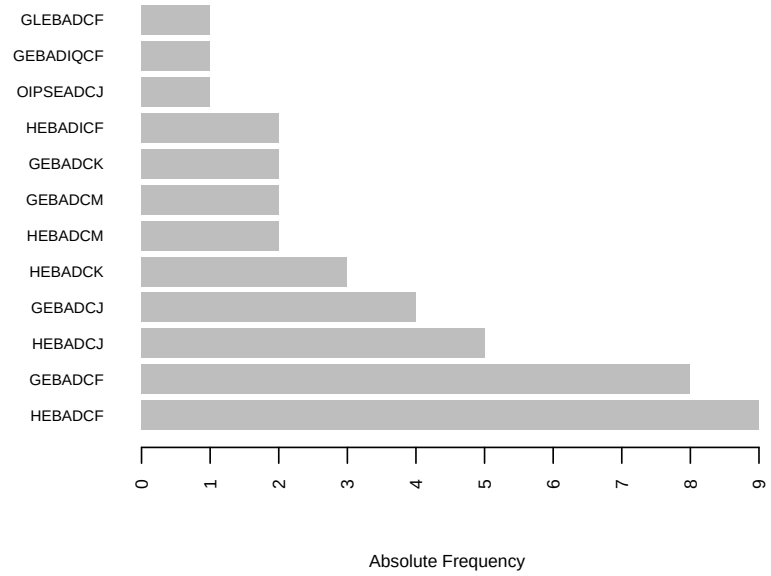

Figure 6.2: Top twelve sequences present in the stroke log and respective absolute frequencies.

However, one can notice an interesting aspect by analyzing Figure 6.2 showing the top twelve sequences[1]. The top five sequences vary very little; basically, it only varies how a patient is triaged or discharged. It means that there is a behavioral pattern that confers some structure to the process despite the diversity of traces found in the event log. Eventually, this kind of behavior is translated to regular behavior during sequence clustering analysis.

Figure A.8, Appendix C, depicts a dependency graph that models the control-flow of the global process. It was obtained by applying the Heuristic Miner with parameters as default to the non-clustered event log. As the reader may have noticed, this model is not clear enough for analysis, since one can not easily detect behavioral patterns, nor easily distinguish what parts correspond to regular behavior or main variants. To clarify these issues, we proceed with sequence clustering analysis.

6.2 Sequence Clustering Analysis

The results obtained from each step of the sequence clustering analysis are revealed next.

Running the Sequence Clustering Algorithm: As a first attempt it was explored the MS sequence clustering feature that tries to automatically output the optimum number of clusters.

[1]Each letter in a sequence correspond to the activity with the same letter in Figure6.1

Surprisingly, the result was only two clusters. Considering the high degree of ad-hoc behavior identified it was expected a larger set of clusters. With further analysis it was clear that two clusters was not an optimum number because both contained too much behavior, i.e. both did not clearly separate different behavioral patterns. Our concern was to aggregate in the same cluster the most frequent sequences previously identified, while trading-off between a small number of clusters and an acceptable degree of complexity at each cluster model. After some experimentation, based on trial and error, we found the best results by setting eight clusters as input parameter.

The Cluster Diagram: From the eight cluster models it was obtained the cluster diagram depicted in Figure 6.3. Analyzing the diagram one sees cluster 5 with the highest support, having a value of 0.34, approximately. Following our assumptions, the Markov Chain assigned to this cluster will let us understand how stroke victims are typically treated at HSS emergency department.

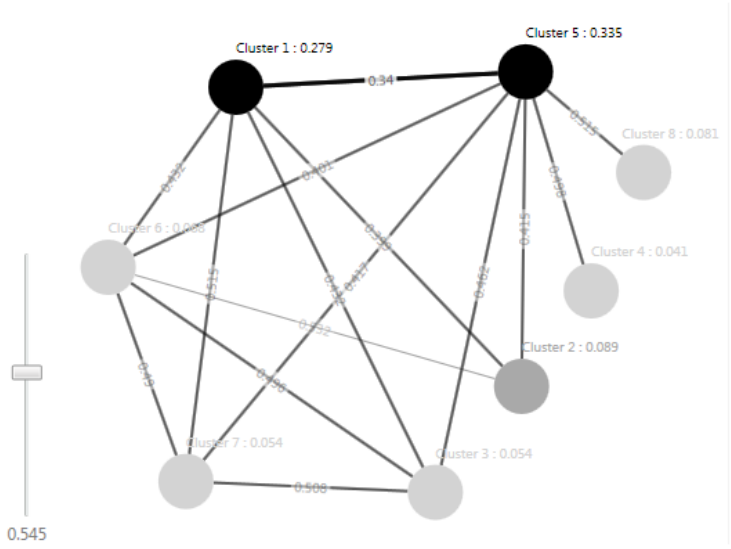

Figure 6.3: Acute Stroke Process - Cluster diagram obvtained with eight clusters as input parameter.

Although cluster 1 has significant support (0.28), it will be considered a process variant as the remaining clusters. Cluster 3 and 4, for example, show a relative low support. Therefore, they are good candidates to understand infrequent behavior. By looking at the diagram one also has a perception about the similarity between cluster models. One sees, for instance, that cluster 5 and 1 are the most similar in the set with a similarity value of 0.34. It means that the process models given by cluster 5 and 1 present the less structural and behavioral variance among all other cluster models.

Understanding Regular Behavior: As already referred, the Markov Chain associated with cluster 5 will insight us about how HSS typically treats emergency patients with stroke. The respective model is depicted in Figure 6.4. As the model shows, emergency professionals start by triaging a stroke victim. More specifically, nurses select one of the following triage fluxograms: patient presenting strange behavior on set ("Triar Doente Comportamento Estranho"); or adult in unwell condition ("Triar Doente Indisposição Adulto"). After triage, it is requested a set of radiology and laboratory exams. They clearly follow a pattern. It is requested a chest x-ray,

following a CT scan, a biochemistry exam, and a hematology exam. After examinations the patient's diagnosis is made (activity "Diagnostico"). Finally, the patient follows one of these paths: (1) is hospitalized ("Transferência Serviço Internamento"), which has the highest probability, 42%; (2) is transferred to an external hospital ("Transferência Outro Hospital"); or (3) is discharged to an external supporting physician ("Transferência Exterior/Médico Assistente").

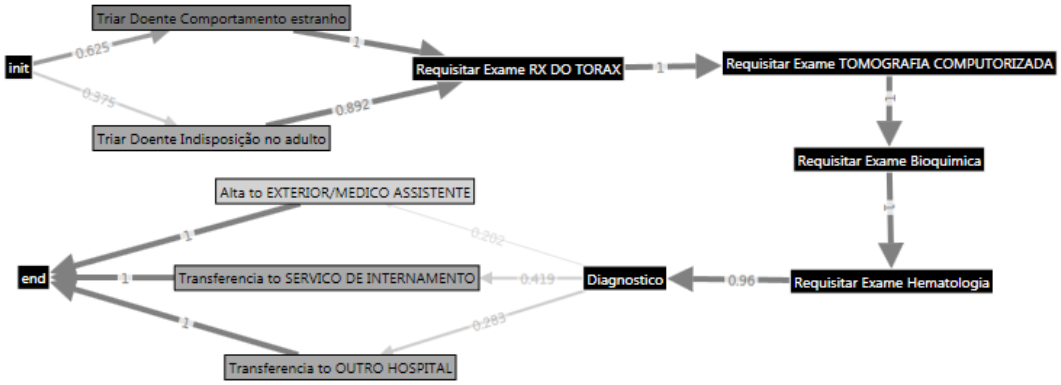

Figure 6.4: Acute Stroke Process - Process model depicting regular behavior (thresholds are applied in order to improve readability).

The behavior discovered with this model conforms with the top five sequences shown in Figure 6.2. Routine requests of biochemistry and blood test are normal and the request of a brain imaging test (a CT scan is this case) is mandatory in order to distinguish ischemic from hemorrhagic stroke. Making this distinction is of extreme importance because ischemic and hemorrhagic stroke have incompatible treatments (e.g. an anticoagulant drug proper to manage ischemic stroke would be dangerous for a patient with hemorrhagic stroke since it would increase the bleeding). Of particular concern is the request of chest x-ray exams observed in the vast majority of cases (the request of chest x-rays is not only observed in all cluster 5 cases but also in the vast majority of cases in the event log). According guidelines [75], chest x-rays were found of little use in absence of an appropriate clinical indication, therefore, this kind of examination is only recommended for selected patients (such as those presenting respiratory complications) and not for the majority of cases as observed at HSS. There is therefore evidence that the hospital could reduce utilization of chest x-ray resources by better selecting stroke patients for this kind of examination.

Understanding Process Variants and Infrequent Behavior: According to the MST of the cluster diagram (see Figure 6.5), the differences between clusters are minimized by comparing: cluster 5 with cluster 1, 4, 6, 7 and 8; and cluster 1 with cluster 2 and 3. We follow the MST to better understand the process variants.

Comparing cluster 5 with cluster 1 (Figure 6.6) one notices little significant variations. This was expected since cluster 5 and cluster 1 are the most similar in the set. The model of cluster 1, for example, shows a direct transition from the *init* state to the request of a chest x-ray, meaning

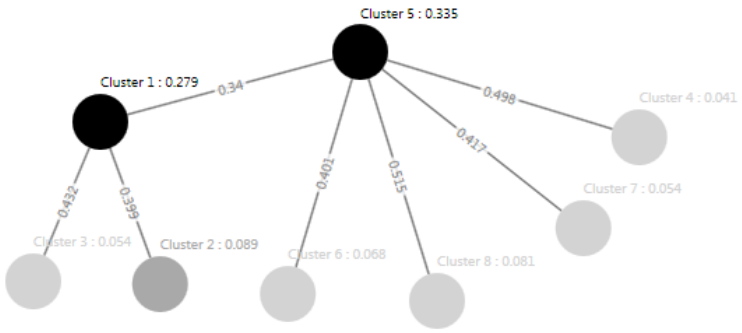

Figure 6.5: Acute Stroke Process - Minimum Spanning Tree of the cluster diagram

the triage is sometimes skipped. These cases regard patients that give entrance at the emergency department in a life threatening condition, therefore, physicians proceed with immediate care. Other difference seen in cluster 1 is the rather frequent prescription of an intravenous normal saline solution after examinations ("Prescrever Cloreto de Sódio 9 mg Sol Inj Fr 1000 ml IV"). This is observed because some stroke patients need fluid replacement on admission to hospital. The use of normal saline solution is a common practice and is recommended for this purpose. In Figure 6.6, the thresholds applied to the transitions and states of the Markov chains of each cluster are hiding less frequent behavior, but hierarchical sequence clustering is used ahead to better understand this remaining behavior.

Since the other clusters are less similar, it is expected more structural and behavioral variance when proceeding with the remaining comparisons. One example is the comparison of cluster 5 with cluster 4 (Figure 6.7). The red activities in cluster 5 refers to those not observed for the cases assigned to cluster 4, and the red activities and transitions in cluster 4 refer to to those not observed in cluster 5. Cluster 4 unveils a specific pattern. First, it only regards patients triaged as presenting strange behavior on set ("Triar Doente Comportamento Estranho"), as well as patients transferred to an external hospital ("Transferência Outro Hospital"). Second, it shows that a patient can be prescribed with a normal saline solution right after the triage is made. Third, and more interesting, is the prescription of intravenous Haloperidol, often used for treating patients with acute psychosis and delirium. The presence of Haloperidol in this cluster, as well as triage by strange behavior, suggests specific cases of stroke patients presenting delirium or similar complications onset, which has been associated with worse functional outcomes, higher mortality and longer hospital stay [76]. HSS opted to transfer these specific cases to other external hospital rather than hospitalize them.

The same logic is followed to understand the remaining variants of the stroke process, therefore, Table A.9, Appendix C, summarizes main findings regarding each variant. For the interested reader, in Appendix C is also depicted the process models of each comparison made. Table A.9 shows that it is observed different procedures for handling stroke victims with specific symptoms or complications. It also shows that there is some clusters (such as cluster 3, 6 and 8) that are

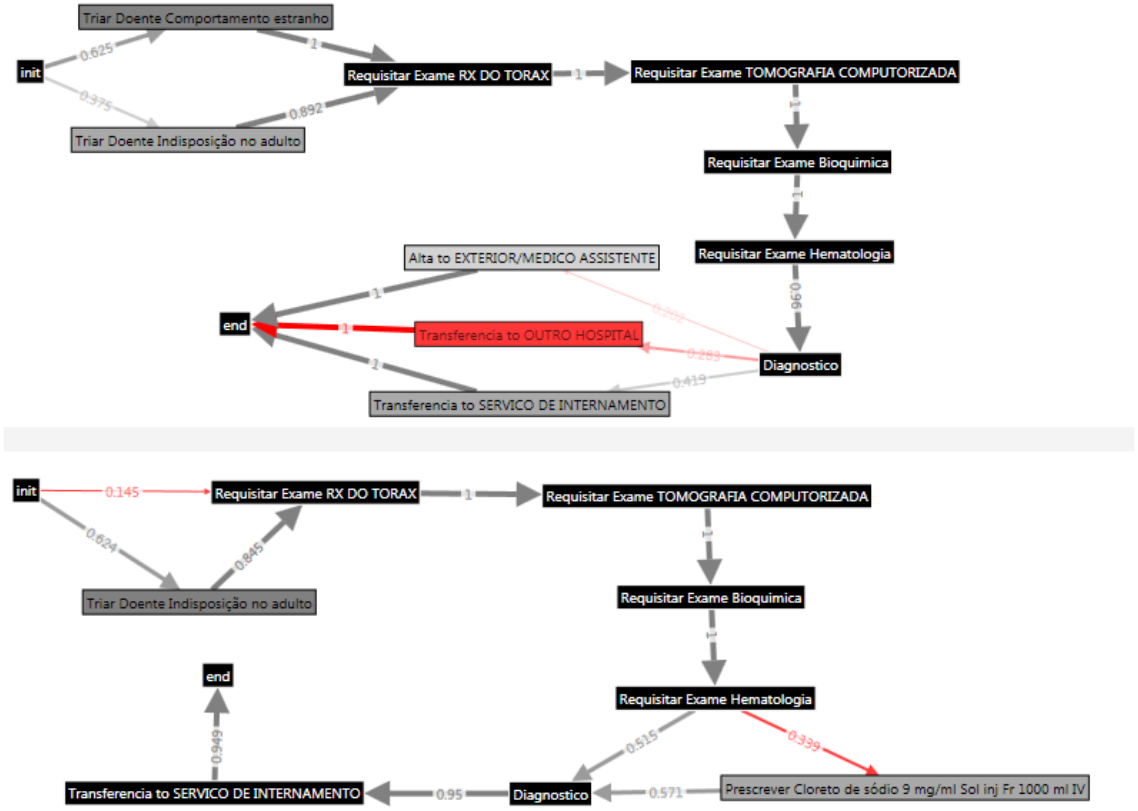

Figure 6.6: Acute Stroke Process - Differences between cluster 5 (on top) and cluster 1 (on bottom).

good candidates for hierarchical sequence clustering, mainly because these models remain complex to understand.

Hierarchical Sequence Clustering: Here we apply hierarchical sequence clustering to cluster 1 as previously indicated. When thresholds were applied in Figure 6.6 for visualizing cluster 1 one aimed at understanding the most significant structural and behavioral features of this process variant. However, thresholds are hiding less frequent behavior that may contain other behavior of interest. Figure 6.8 depicts the model of cluster 1 without thresholds where less frequent behavior is represented by activities in white rectangles (meaning they have low occurrence) and also by transitions with low probability (the thinner the arrow the lower the probability). Since the complexity of the model increases it is not that clear to understand the process structure and flow in presence of less frequent behavior. Henceforth, hierarchical sequence clustering is performed in order to obtain less complex sub-models that are easier to visualize. To do so, the sequence clustering algorithm was run over the log traces associated with cluster 1. It was selected five clusters as input parameter.

From the five clusters, cluster 1.3 had the lowest support and revealed as the most interesting. The respective Markov Chain is depicted in Figure 6.9. It is observed that enoxaparine sodium (an anticoagulant drug) is prescribed and the diagnosis of the patient is made without previous brain image study (Triar Doente Indisposição Adulto => Prescrever Enoxaparina Sódica =>

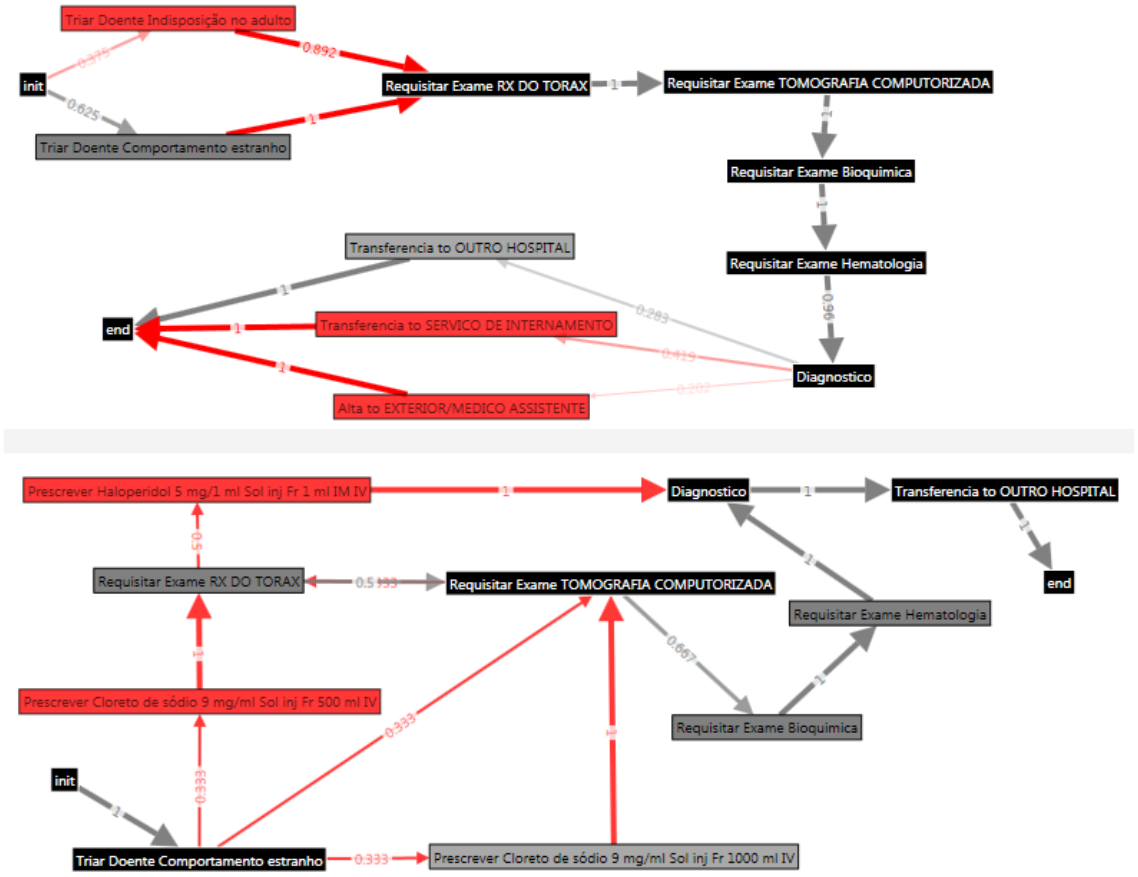

Figure 6.7: Acute Stroke Process - Differences between cluster 5 (on top) and cluster 4 (on bottom)

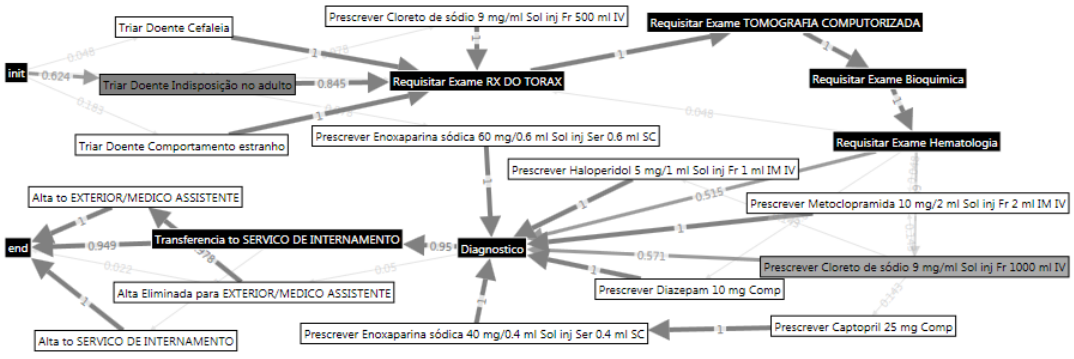

Figure 6.8: Acute Stroke Process - Process model of cluster 1 without thresholds.

Diagnóstico). Recall that brain imaging is mandatory to distinguish ischemic from hemorragic stroke. In case of a miss-diagnosed hemorrhage, treating a patient with enoxaparine sodium could be harmful because the anticoagulant properties of this drug would increase bleeding. Therefore, cluster 1.3 reveals cases of malpractice and deviation from guideline recommendations.

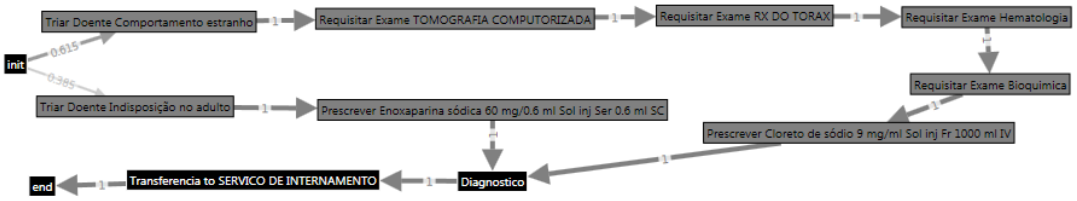

Figure 6.9: Acute Stroke Process - Process model of cluster 1.3 without thresholds (revealing a deviation from guidelines).

Summary: With the sequence clustering analysis we were able to separate and distinguish: how a stroke patient is typically treated; different variants, referring to specific patient cases and complications; and infrequent behavior, referring to rare cases and deviations from guidelines. From each cluster we obtained less complex models, with first order Markov Chains insighting about control-flow. At this point, however, we like to stress two limitations regarding the control-flow semantic expressiveness of first-order Markov Chains.

The first limitation is in expressing activities performed in parallel. The exams requests are an example. Despite that regular behavior showed a specific sequence (Chest X-Ray => CT => Biochemistry => Blood), other clusters showed that this order varies. In reality, there is no specific order because doctors usually request these exams in parallel. Since first-order Markov Chains are not able to express AND joins/splits semantics, this kind of behavior is not evident in the models.

The second limitation is in expressing behavior with non-local dependencies. Since first-order Markov Chains only infer local dependencies of an S_n state (namely the transitions from S_{n-1}), non-local behavior (i.e. transitions from $S_{n-k}, k > 1$) remains undiscovered. See for instance the model of cluster 8 in Figure A.13, Appendix C. The model shows that after prescribing normal saline solution it can be prescribed ipatropium bromide + salbutemol or it can be requested a chest x-ray, independently of triaging a patient with dyspnea or convulsions (recall the Markov property). However, according the log traces, ipatropium bromide + salbutemol is only prescribed when patients are triaged with dyspnea (a non-local preceding activity affecting the choice of the drug), and not when a patient is triaged with convulsions. This behavior is clearly not expressed in the model of Figure A.13, Appendix C.

In order to overcome the limitations above, one needs to run other control-flow mining techniques on top of the sequence clustered traces. In essence, exploit the strengths of other mining techniques with richer semantic expressiveness given that the log traces are already clustered and less complex models can be mined for each cluster. In Appendix D, the Heuristic Miner is applied on cluster 8 traces in order to illustrate how the stated limitations can be overcome.

The control-flow perspective of the stroke process has been focused till this point, but a question of great concern remains unanswered: how is HSS performing regarding the time taken to manage acute stroke cases? Next section analyzes the time dimension of the acute stroke process at HSS.

6.3 Performance Analysis

Time is among the most critical variables in what regards acute stroke management. The first minutes to hours after the onset of acute stroke victims frequently hold the only opportunity to prevent death or permanent brain injury. Consensus guidelines propose the following goals for hospital emergency services [77]: (1) triage of the stroke patient to medical attention within 10 minutes; (2) history, physical examination, and blood tests as soon as possible; (3) brain imaging within 30 minutes; and (4) treatment decisions within 60 minutes of presentation. In short, the first hour is critical. Ideally, a stroke victim should be evaluated and provided with a treatment plan within this time interval.

HSS could not provide a precise, readily available answer to analyze whether the hospital deviates from the recommendations above. Moreover, it is not clear for the organization where bottlenecks are. Fortunately, with the event data gathered and with the aid of process mining one can gain insights about this topic. To do so, we start by analyzing the throughput time of the stroke process. It will let us understand how long does the emergency service take to manage stroke victims, from their triage to their discharge/transfer. With this one can understand if patients are handled within the first recommended hour. We will then analyze the time spent at specific parts of the process. It will insight us about bottlenecks, and provide answers to questions such as: Does the patient receives medical attention within 10 minutes after triage? Can we obtain brain imaging results within 30 minutes?

One possible approximation for analyzing throughput time is the Dotted Chart available at ProM. Because this technique is not dependent of a process model (such as the performance analysis with Petri Nets), it is appropriate to comprehend the global performance of a process regardless the degree of complexity of the latter. Figure 6.10 depicts the Dotted Chart obtained from the global stroke process cases, with parameters set to measure throughput time. At the y-axis we have each stroke case. The colored dots refer to the different activities observed for each case and span across the x-axis with respect to their relative time of occurrence (measured in hours). The throughput time of each case is given by the corresponding last dot (colored in gold or orange and referring to patient discharges or transfers). Finally, cases are ordered according their throughput time in order to insight about the distribution.

Analyzing the chart, one observes a small percentage of stroke victims handled within the first hour. Most part is distributed along the interval ranging from three to four hours. Moreover, there is a significant number of cases handled within five to nine hours, which is particularly preoccupying. It is also identified larger timespans from the exams requests (purple dots) to the patient diagnosis (green dots), indicating a potential bottleneck.

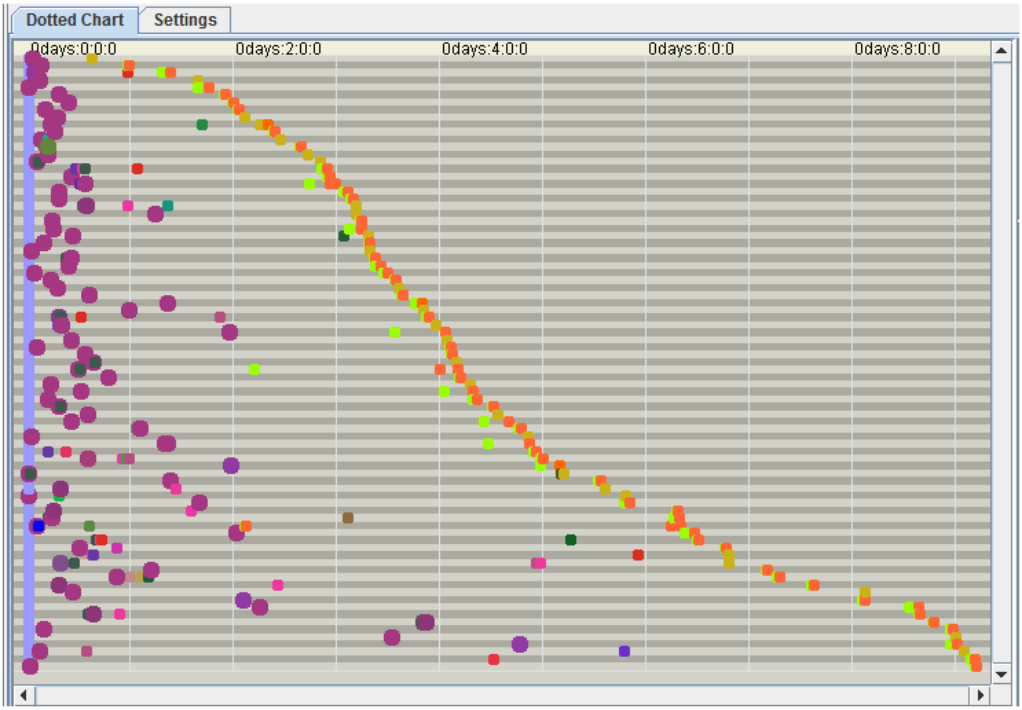

Figure 6.10: Acute Stroke Process - Dotted chart analysis.

To provide more precise answers, one can explore the results obtained during the sequence clustering analysis. First, one can detect possible variations on throughput time depending on specific patient complications or clinical decisions; hence, become aware of possible performance issues observed at specific cases. Second, one can better explore techniques capable to project the time dimension on process models and detect bottlenecks more easily; in a sense that we already reduced the complexity of the global model.

Table 6.11 shows descriptive statistics about the throughput time of each cluster previously discovered. As reference, it also shows the same information for global cases.

		Throughput (*minutes*)			
cluster c_k	$support_{c_k}$	*mean*	*min*	*max*	*stdv*
1	0.279	270.6	99.7	552.2	130.3
2	0.089	354.5	194.5	519.0	111.3
3	0.054	349.0	233.1	545.8	147.6
4	0.041	386.0	280.0	487.7	103.9
5	0.335	224.6	37.7	540.3	115.0
6	0.068	290.4	144.5	542.0	159.8
7	0.054	211.3	83.1	408.4	138.6
8	0.081	361.3	191.5	527.1	119.2
Global	1	282.0	37.7	552.2	133.8

Figure 6.11: Acute Stroke Process - Throughput time of observed for each cluster.

Let us focus our efforts on analyzing cluster 5 cases. With this cluster we will understand

the root cause of the performance problems identified, which is affecting all kinds of stroke cases. Figure 6.12 depicts the Markov Chain of cluster 5 with each transition colored according the mean transition time observed. Transitions marked as red, yellow and green are showing high, medium and low values, respectively. Transitions in grey refer to values equaling zero.

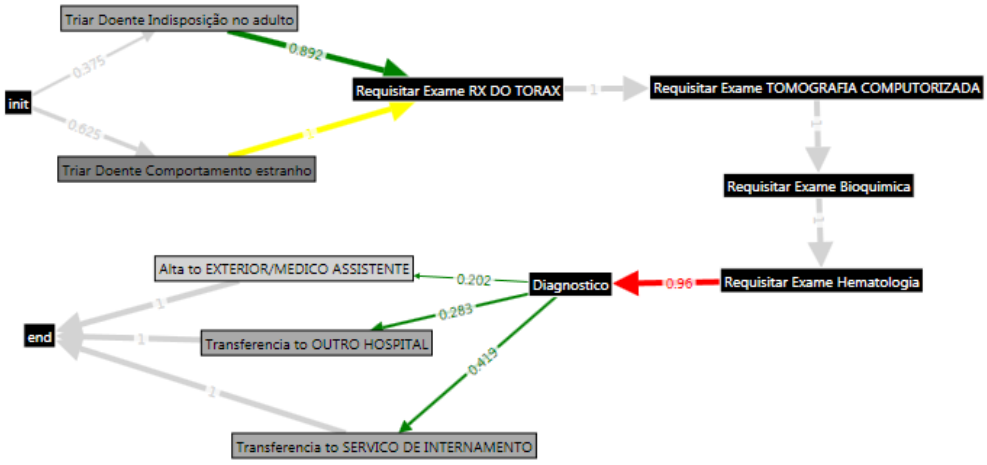

Figure 6.12: Acute Stroke Process - Markov Chain of cluster 5 colored according transition times.

Let us abstract from the different triages without compromising the analysis. We can also abstract from the different exam requests, in a sense that exams are requested in parallel for cluster 5 cases (note the transition times between the sequence of exams equaling zero). Looking at Figure 6.12, one identifies two points of concern: (1) the transition time from triaging a patient to the exams requests, let us say T1 ; and (2) the transition time from exam requests to diagnosis, let us say T2[1].

We start by analyzing T1. One can answer from this if stroke patients are actually receiving medical attention within 10 minutes after triage as recommended (for cluster 5 cases, the exams requests are that medical attention). Figure 6.13 plots the histogram and density function of T1. It also shows an equivalent density function for global cases [2]. We assume reasonable to describe both distributions according a log-normal law [3]. For cluster 5 cases, it was estimated a geometric mean $\bar{x}^* \approx 20.88\ (-5.11, +6.44)$ and a multiplicative standard deviation $\sigma^* \approx 3.03(-0.65, +2.24)$ minutes. For global cases, $\bar{x}^* \approx 24.73(-5.21, +6.6)$ and $\sigma^* \approx 2.86(-0.37, +0.64)$ minutes [4].

Considering cluster 5 cases and averagely speaking, stroke patients receive medical attention after triage in 24.73 (-5.11, +6.44) minutes. It corresponds to an average deviation from guidelines of 14.73 (-5.11, +6.44) minutes. As the histogram shows, only 30 % of cases in cluster 5 were

[1] T1 is given by the transition times observed from both triages to the chest x-ray request. T2 is given by the transition times observed from the hematology exam request to the patient diagnosis.

[2] In our case, all activities observed right after triage are 'medical attention', therefore, considering these transition times, one obtains a density function with an equivalent semantic meaning than the one of cluster 5.

[3] Goodness of fit for global cases: $D = 0.086$, $p\text{-}value = 0.1582$. Goodness of fit for cluster 5: $D = 0.1287$, $p\text{-}value = 0.5201$. Values according Lilliefors's test, from which we accept null hypothesis

[4] Parameters estimated by Maximum Likelihood Estimation. Values in parenthesis show the confidence interval at 95% level, relatively to parameter

treated within 10 minutes, but more concerning is the long tail of the density curves, showing the occurrence of preoccupying transition times.

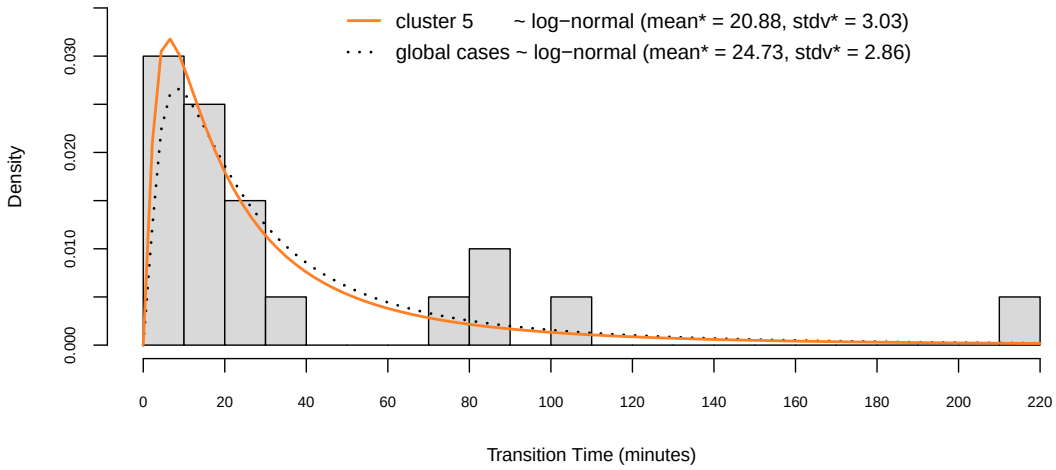

Figure 6.13: Acute Stroke Process - Distribution of the transition time from triage to medical attention (cluster 5 and global cases).

Let us now analyze T2, i.e. the transition time from exam requests to diagnosis (observed for cluster 5 cases). In Figure 6.14 we have the histogram and the density function considered. The latter is expressed according a gamma distribution with $shape \approx 3.62$ $(-0.921, +1.23)$ and $scale \approx 60.28$ $(-16.26, +22.27)$ [1]. By definition, the mean transition time $\bar{x} = shape * scale \approx$ 218.21 $(-14.98, +22.39)$ minutes. In other words, it takes an average of 218.21 (-14.98, +22.39) minutes from exam requests to diagnosis, which indicates that this transition time is the main responsible for the observed throughput values of the stroke process. It also indicates the root cause of the bottleneck is found at the laboratory or radiology process level (i.e. at organizational processes).

Since the radiology process was already analyzed in the previous chapter, one can provide a more precise answer to the above. Recall that the analysis of the radiology process revealed that CT exams cases, which are mandatory for all stroke cases, had significant higher throughput values than typical radiology exams (in average). Also recall that the outsourcing of the reporting of CT exams has significant impact on throughput time, and the external entity responsible for the reporting also exceeded the expected 1 hour for report delivery in approximately 40% cases. For the sake of clarity, Figure A.17 in Appendix E provides the reader a global picture of the performance issues of the stroke process and their interconnection with the performance of the radiology process.

[1] Goodness of fit: $D = 0.0698$, $p\text{-}value = 0.793$. Values according the Kolmorov-Smirnov test, from which we accept null hypothesis. Parameters estimated by Maximum Likelihood Estimation. Values in parenthesis show the confidence interval at 95% level, relatively to parameter.

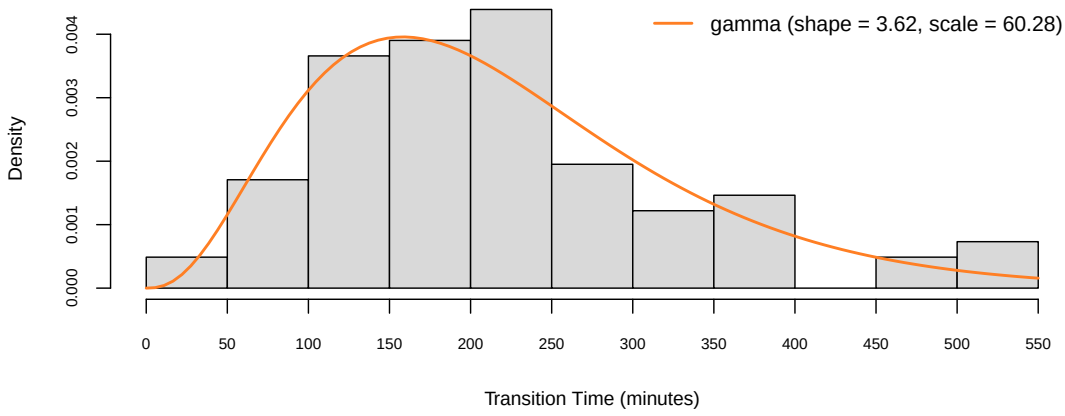

Figure 6.14: Acute Stroke Process - Distribution of the transition time from exam requests to patient diagnosis (cluster 5 cases).

6.4 Summary

In this chapter, the acute stroke process was analyzed in order to gain insight about the control-flow and performance perspectives of the process. With the sequence clustering analysis it was possible to discover typical behavior in patient treatment, different treatment variants (due different stroke complications), and infrequent behavior, which revealed cases of malpractice. When compared with guidelines, the analysis also revealed that HSS could better rationalize chest x-rays.

The process performance was analyzed using the Dotted Chart Analysis (to gain insight about throughput), and by projecting transition times on the first-order Markov Chain models discovered. From there it was possible to study transitions of interest in more detail, by means of statistical analysis. More specifically, it was studied the time from triage to medical attention, as well as the time from exam requests to patient diagnosis. The latter revealed that the performance of the radiology process, studied in the previous chapter, highly impacts the throughput of the acute stroke process.

The next sub-chapters summarize, discuss main findings in more detail, advantages of process mining to analyze clinical processes, as well as points of improvement to make in future.

6.4.1 Main findings

Regarding control-flow analysis:

- Despite the large diversity of traces found in the event log, there is a behavioral pattern which occurs for the vast majority of cases, and which confers some structure to the process. This pattern refers to the request of a chest x-ray, a CT scan, a biochemistry exam, and a hematology exam. The request of chest x-rays as a common practice is of particular concern because guidelines tell that chest x-rays were found of little use in absence of an appropriate

clinical indication (such as those presenting pulmonary or cardiac pathology); therefore, only recommended for a selected set of patients, and not as common practice. The HSS specialist confirmed that requesting chest x-rays is indeed a common internal practice. It was further explained that given the fast paced environment of the emergency it is a way to assure that the patient is not suffering from pneumonia or heart failure, which can be a cause of stroke. Practicing defensive medicine in fast-paced environments such as emergency departments is usual. It serves as means to decrease the risk of miss diagnosis; but at same time it increases the risk of resource waste. It would be interesting for HSS to assess and control the costs of chest x-rays that are of little use; basically counting the number of acute stroke patients who did not had pulmonary or cardiac pathology but had chest x-ray performed, and multiplying that value for the median cost to perform a chest x-ray. If justified, a mechanism should be thought to mitigate waste.

- Several variants in treatment were found due different patient complications. Cluster 1, for example, illustrated cases in which the triage is skipped (referring to patients that give entrance in life threatening conditions), as well as prescription of intravenous normal saline solution for fluid replacement, which some patients need. Cluster 4 revealed evidence of patients presenting delirium or similar complications onset, which has been associated with worse functional outcomes, higher mortality and longer hospital stay. The cases associated with this cluster were not hospitalized but rather transfered to an external hospital. Main findings of the remaining clusters were already summarized in Table A.9, showing different procedures to handle stroke victims with specific problems.

- Hierarchical Sequence Clustering revealed an interesting infrequent pattern indicating cases of malpractice. It was observed that an anticoagulant drug (enoxaparine) is prescribed and the diagnosis of the patient is made without previous brain image study. According guidelines, a brain imaging study is mandatory to distinguish ischemic stroke from hemorrhagic since both conditions have incompatible treatment (a proper anticoagulant drug to treat ischemic stroke would be dangerous for a patient with hemorrhagic stroke because it would increase the bleeding).

What regards performance:

- Dotted chart gave a first glance of the global performance. It was observed an average throughput of 4.7 hours, with a small percentage of stroke victims handled within the first hour onset. Most part is distributed along an interval ranging from three to four hours, and there is a significant number of cases handled within five to nine hours. The average throughput was considered normal attending national average, but there is still the need to reduce variation and decrease the number of cases surpassing 5 hours.

- The transition time from triage to medical attention can be expressed as a log-normal law with an estimated geometric mean $\bar{x}^* \approx 24.73(-5.21, +6.6)$ minutes, and multiplicative standard

deviation $\sigma^* \approx 2.86(-0.37, +0.64)$ minutes. In average, stroke patients receive medical attention after triage in 24.73 (-5.11, +6.44) minutes, whereas guidelines recommend emergency triage in less than 10 minutes. HSS therefore presents an average deviation from guidelines of 14.73 (-5.11, +6.44) minutes, with approximately 30% of cases treated within 10 minutes. The long tail of the density curves showed the occurrence of preoccupying transition times.

- The transition time from exam requests to diagnosis can be expressed as a gamma distribution with *shape* ≈ 3.62 $(-0.921, +1.23)$ and *scale* ≈ 60.28 $(-16.26, +22.27)$. It means that it takes in average 218.21 (-14.98, +22.39) minutes from exam requests to diagnosis. This transition time is the main responsible for the high values of throughput time observed, and showed that the root cause of the problem may be due the performance of laboratory or radiology processes.

- Attending the analysis made in the previous chapter, the root cause of the performance problems is found at the emergency radiology process; specifically due the bottleneck found at the computed tomography (CT) workflow, which is mandatory for all stroke victims. The outsourcing of CT reports has significant impact on throughput, and the external entity responsible for the reporting exceeded the expected 1 hour of report delivery in approximately 40% of cases. For a more comprehensive view on how everything is interconnected, the reader is referred to Figure A.17, Appendix E

6.4.2 Advantages

The results obtained in this chapter let us re-iterate the advantages already discussed in the previous chapter. HSS has now the ability to be in always in control of the acute stroke process based on objective data.

A major advantage to stress here is the scalability of the solution. In this chapter, the acute stroke process was focused but the system can be used to analyze any other emergency process attending patient diagnosis, meaning HSS can be in control of any observable clinical process of the emergency department. Achieving this transparency would be unfeasible with traditional process analysis due the prohibitive associated costs. Process mining therefore proves valuable to manage clinical complexity.

Process mining results can be compared with clinical guidelines in order to detect deviations from high quality evidence recommendations. In this sense, process mining is useful to support the practice of evidence based medicine (EBM) [78]. Particularly, it has the potential to support scalable, more accurate EBM auditing with less incurred costs. Moreover, one can easily attribute responsibility in case of deviations, because one can easily map a task, i.e. an action, to the professional who perform it.

Finally, process mining offers a comprehensive way to understand how different units interact, and how they affect each other. In this work we objectively interconnected the performance of the

acute stroke process with the performance of the emergency radiology process, and the performance of the latter with the performance of the external entity responsible to report CT scans.

6.4.3 Points of improvement

Although the current setting provides valuable insights about clinical processes, it would be interesting to include data about the patient context (such as demographics, problems, medications, or history) as well as the patient outcomes. That would offer new possibilities of analysis from a clinical viewpoint. The patient context would provide capability to build event logs targeting specific groups of patients, which would then be analyzed with the methodology proposed in this work. The patient outcomes, in addition with the patient context, would let one measure the risk associated with the different pathflows. There is however two main challenges to acquire the patient context and outcomes. First, most of these data is still found in admission, discharge/transfer reports recorded in text free form that is non-trivial to process and structure. Second, it requires patient sensitive information which access is seen with reluctance due privacy concerns.

Despite the quick results delivered by process mining, comparing those results with implicit knowledge contained in text based, conventional medical guidelines is a laborious task prone to ambiguity. In this context, computer-interpretable guidelines formalisms (CIG) (e.g. Asbru, EON, GLIF, GUIDE, PRODIGY, PROforma, Arden Syntax)[79, 80] assumes an important role. They represent guidelines objectively, and it would be possible to automate, or partially automate, the detection of deviations and conformance measurement. The problem is that CIG require a level of technical expertise that most clinicians do not have. Attending that guidelines are developed by clinicians, it is natural that they favor the text free form over CIG. This gap can be bridged with a conjunct effort between IT people and clinicians. For the particular case of HSS, it would be valuable to team up the IT department with a group clinicians in order to formalize the most relevant guidelines for the hospital as means to leverage the advantages process mining brings.

What regards the methodology proposed, the initial selection of the numbers posed the same problems discussed in the concluding section of the previous chapter. Let us therefore re-iterate that one needs to develop proper heuristics to provide an adequate indication for the number of clusters.

The limitations of first-over Markov Chains to express non-local dependencies and activities performed in parallel became evident in this chapter. These limitations are however easily overcome by running control-flow mining techniques with stronger semantic expressiveness on top of the traces associated with each cluster. It justifies in future to directly devise process models such as heuristic graphs from the traces associated with each cluster (instead of first-order Markov Chains).

7

Analysis of the Organizational Perspective of Emergency Processes

In last two chapters all steps of the methodology were approached with exception of the organizational analysis. This chapter is dedicated to that. Using organizational mining techniques it will be analyzed a sensitive and controversial problem perceived by HSS people but of difficult proof. According HSS protocols, when an emergency physician calls a patient he/she is responsible for executing all clinical activities till patient transfer/discharge; with exception of two situations: (1) when work shift ends; or (2) exceptional cases where the patient actually needs attention of other physician. Essentially, and disregarding the exceptions referred, emergency doctors are not expected to handover patients to other fellow doctors after calling a patient.

HSS has been suspecting that may exist physicians handing over patients more than needed, but the organization lacks of objective means to measure and prove such deviations. Traditional BPA techniques are not a feasible approach to analyze such situations because clinicians deviating from the hospital policy would not risk their professional careers by describing reality accurately. Process mining provided an opportunity to address the stated issue.

The remaining of this chapter develops as follows. Section 7.1 describes how the log was prepared. Section 7.2 presents the organizational mining technique used, the metric defined to measure deviations, and the results obtained. Section 7.3 concludes the chapter with a reflection and summary of the analysis.

7.1 Log Preparation and Inspection

In order to prepare a suitable event log for the problem, the following aspects must be considered: (1) one needs to measure the total amount of patients called by a doctor, i.e. the total amount of patients "handed" by triage nurses to that doctor; and (2) one needs to measure the amount of patients transferred among doctors. By establishing a relation between these two measures one can identify potential deviations (details are discussed in next section). What regards our dataset, one focus the following activities: *Triage*, *Prescribe Treatment*, *Request Laboratory Exam*, *Request Radiology Exam*, *Diagnosis*, and *Transfer/Discharge* (respectively included in *Triage*, *Treatment*, *Laboratory Exam*, *Radiology Exam*, *Diagnosis*, and *Transfer/Discharge* tables). This high level of abstraction is justifiable because we are not particularly concerned in studying how patients are managed but rather in studying the originators of triage and clinical events, as well as the amount and flow of patient transfers among originators. The resulting event log contained a total number of 78623 cases and 262047 events originated by 475 distinct professionals.

7.2 Organizational Analysis (Social Network based on Handover of Work)

From all organizational mining techniques and metrics proposed by Song and Aalst [40][41], building a social network based on handover of work is a natural fit to analyze our problem. Given a process instance, there is a handover of work from originator o_i to originator o_j if there are two subsequent activities where the first is performed by o_i and the second by o_j. Note that the metric considers a causal relation between originators that is needed for our case. It let us detect and quantify: (1) transfer of work from a triage nurse to the doctor performing the first clinical activity, from which we obtain the number of patients called by doctors; and (2) possible patient transfers from a calling doctor to other fellow that performs a subsequent clinical activity within the same case, i.e. a potential deviation.

By applying the handover metric to the event log it was obtained sociogram depicted in Figure A.18, Appendix F [1]. It is a directed and weighed graph: green nodes refer to triage nurses and the remaining to doctors (node colors are discussed ahead); edges show the direction and absolute frequency (the edge weight) of patients handovers between emergency professionals.

Looking at Figure A.18, Appendix F, one gets an idea of how complex "social work interactions" are among triage nurses and emergency physicians. The ill-defined work schedule of these professionals explains the high density of the network. The schedule of emergency nurses, for example, usually changes on weekly basis, varying work days, shifts, or number of shifts per day. Therefore, a triage nurse eventually ends handing over a patient to every other emergency physician. It is

[1] The sociogram develops from a $N \times N$ matrix $M = \{a_{ij}\}$ with N given by our 475 originators and a_{ij} given by the absolute frequency of handovers from originator i to originator j observed for all 78623 cases, $0 \leq i, j \leq N-1$, $a_{ij} \geq 0$. It was disregarded "self handovers" ($\forall_{i=j}, a_{ij} = 0$). The sociogram is also hiding nodes and respective edges associated with doctors and nurses that handled less than 20 cases.

observed, however, stronger work interactions between some nurses and physicians (note the darker edges).

In spite of complexity, the network provides useful information for analytical detection of potential deviations. The inflow and outflow of each node was of particular interest. Let:

- o_n be a node from the set $O = \{o_0, o_1, ..., o_{N-1}\}$ of our sociogram N nodes;
- $IN\text{-}EW^{o_n} = \{w(in\text{-}e_0), w(in\text{-}e_1), ..., w(in\text{-}e_{J-1})\}$ be a vector of weights of the J incoming edges $in\text{-}e_0, in\text{-}e_1, ..., in\text{-}e_{J-1}$ of node o_n;
- $OUT\text{-}EW^{o_n} = \{w(out\text{-}e_0), w(out\text{-}e_1), ..., w(out\text{-}e_{K-1})\}$ be a vector of weights of the K outgoing edges $out\text{-}e_0, out\text{-}e_1, ..., out\text{-}e_{K-1}$ of node o_n.

We define $INFLOW(o_n) = \sum_{i=0}^{J-1} IN\text{-}EW_i^{o_n}$ and $OUTFLOW(o_n) = \sum_{i=0}^{K-1} OUT\text{-}EW_i^{o_n}$. In essence, we are measuring the total amount of patients an emergency professional *receives from* colleagues and the total amount *transfered to* colleagues, respectively. In our case, if o_n refers to a triage nurse we expect $INFLOW(o_n) = 0$ because they are the process initiators. If o_n refers to a physician, the more $OUTFLOW(o_n)$ approximates $INFLOW(o_n)$ more likely o_n is deviating from the hospital policy.

To properly express the relation between $INFLOW(o_n)$ and $OUTFLOW(o_n)$ of an originator o_n it was defined:

$$\Psi_{o_n} = \frac{OUTFLOW(o_n) - INFLOW(o_n)}{OUTFLOW(o_n) + INFLOW(o_n)}$$

It develops from the Bray-Curtis similarity measure which properties suited our needs[1]. Note that Ψ_{o_n} is defined at $[-1, 1]$ interval. It provides interesting features to classify and distinguish originators:

- $\Psi_{o_n} = 1 \Rightarrow INFLOW(o_n) = 0$, let us say o_n is a pure process initiator;
- $\Psi_{o_n} = -1 \Rightarrow OUTFLOW(o_n) = 0$, let us say o_n is a pure process finisher;
- $\Psi_{o_n} = 0 \Rightarrow INFLOW(o_n) = OUTFLOW(o_n)$, let us say o_n transfers the exact same amount of work it receives;
- $\Psi_{o_n} \to 0$, let us say it diminishes the tendency of o_n as process initiator (if $\Psi_{o_n} \to 0^+$) or as process finisher (if $\Psi_{o_n} \to 0^-$).

[1] In its generalized form the Bray-Curtis similarity measure is expressed as:

$$BC_{ij} = \sum_{k=1}^{n} |x_{ik} - x_{jk}| \,/\, \sum_{k=1}^{n} (x_{ik} + x_{jk})$$

where i and j can be seen as two objects, each defined by a set of n attributes x_{ik} and x_{jk}. BC_{ij} has the property that if all n attribute values of i and j are positive, its value is defined between zero and one, i.e. a normalized value. Removing the absolute sign, BC_{ij} is defined at $[-1, 1]$ interval. Either way, $BC_{ij} = 0$ represent exact similar objects. If i and j are in zero coordinates, BC_{ij} is undefined. The latter is not our case, otherwise one would be dealing with isolated professionals, not receiving and not transferring patients.

Having said all that, Figure 7.1 plots the Ψ_{o_n} value of each emergency professional o_n previously shown in the sociogram. Triage nurses are clearly distinguished at $\Psi = 1$ since they are pure process initiators. Emergency physicians span along the $[-1, 0)$ interval. With more or less tendency they are process finishers, as they are supposed to. However, as this tendency diminishes more likely they are transferring patients more than needed.

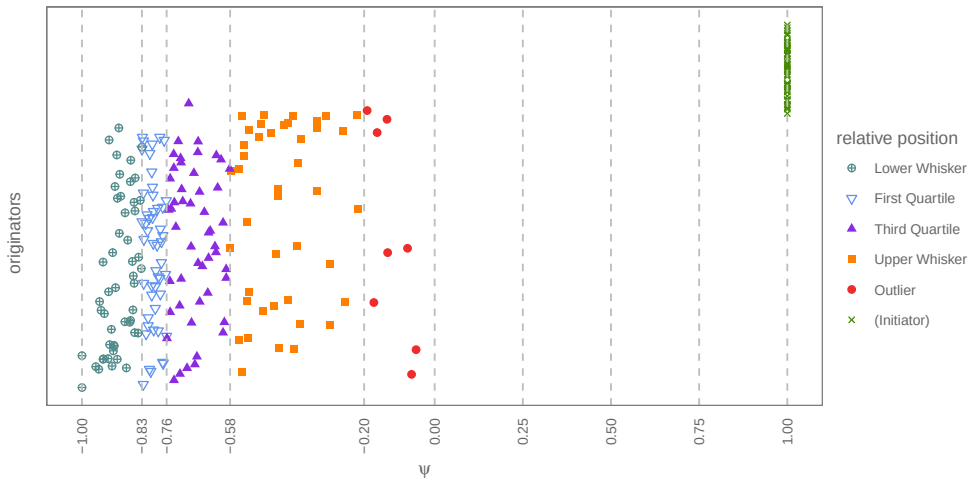

Figure 7.1: Distribution of emergency professionals according their Ψ value.

To clearly distinguish potential deviants from the remaining, each doctor was positioned relatively to the central tendency of doctors' Ψ values (it is assumed $-1 \leq \Psi \leq 0$). The different colors and shapes of emergency doctors in Figure 7.1 reflects that. It follows the same logic of a boxplot [81]. The median is at $\Psi = -0.76$, the first quartile $Q1$ at $\Psi = -0.83$, and third quartile $Q3$ at $\Psi = -0.58$. As accepted in literature [82], the maximum value of the upper whisker is defined at $Q3 + 1.5 * IQR$, where $IQR = Q3 - Q1$ is the interquartile range. Every doctor with Ψ value above the upper whisker maximum is considered an outlier, i.e. most likely a deviant. However, doctors lying at upper whisker, specially near the maximum value of the interval, are also of interest. This information was used to filter originators and work relations of interest in order to reduce the graph complexity presented in Figure A.18, Appendix F, and better understand its structure. Figure A.19, Appendix F, exhibits an example on how the sociogram presented in Figure A.18, Appendix F, can be filtered in order to reveal significant outliers and upper whiskers, as well as the colleagues they often transfer patients to. Table A.20, Appendix F, shows the inflow, outflow and Ψ values of relevant outliers and upper whiskers.

7.3 Summary

This chapter focused the analysis of the organizational perspective of the emergency department. It was devised a social network based on transfer of work in order to quantify the amount of patients that doctors transfer to each other. This was performed to analyze if there were professionals transferring patients more than needed, which is a problem HSS suspects to exist but was lacking of objective means of proof.

A special purpose metric Ψ was developed to measure the relation between the observed amount of patients a doctor receives and the observed amount of patients the same doctor transfers. Following the same logic of a box-plot, each doctor was positioned relatively to the central tendency of the observed Ψ values, letting us to objectively detect outliers referring to potential deviants of internal hospital policies.

Traditional process analysis techniques would not be a feasible approach to analyze this problem because clinicians deviating from the hospital policy would not risk their professional careers by telling the truth. Process mining and the metric developed proved useful to objectively analyze the problem.

8 Conclusion

Healthcare organizations constantly struggle to control and optimize their care delivery processes, as means to improve quality and efficiency while reducing costs; as well as to keep up with legislations, regulation acts, and the extensive amount of clinical guidelines issued. Business Process Analysis (BPA) becomes extremely important to enable effective control and improvement of healthcare processes. BPA defines a set of methodologies, techniques and tools which let one understand and reason about how an organization works, so that one can detect points of improvement.

However, traditional BPA approaches are time-demanding, costly, and depend on subjective descriptions made by people. These issues yields strong barriers for effective use of BPA in healthcare, where processes are in general highly dynamic, complex, multi-disciplinary, and ad-hoc in nature. In practice, the value of BPA remains unrealized for the most part.

8.1 Main Contributions

In this work we have presented a methodology based on process mining for the analysis of healthcare processes. Healthcare information systems that record clinical activities as they take place can be a valuable source of data for analyzing these processes and studying them according to several perspectives. Process mining offers agile means to automatically devise, monitor, analyze, and understand different views of actual (not assumed) processes by extracting knowledge from event

logs recorded by systems.

In environments with process characteristics such as healthcare, the mining techniques that become most useful are those that can cope with large amounts of noise and that can sort different behaviors so that one can study them separately. We have therefore devised a methodology where sequence clustering plays a key role in identifying regular behavior, process variants, and infrequent behavior as well. This is done by means of a cluster diagram and a minimum spanning tree, which provide a systematic way to analyze the results.

The proposed methodology was applied in practice by conducting a case study at the emergency service of Hospital de São Sebastião, a portuguese hospital with approximately 400 beds that has its own electronic patient record system, developed in-house. Event data collected from this system was analyzed with a special purpose tool as well as with plug-ins available in the ProM framework. This work focused the analysis of the emergency radiology process, the acute stroke process, and the organizational perspective of the emergency department.

The analysis showed that process mining can provide insight into the flow of healthcare processes, their performance, and their adherence to institutional or clinical guidelines. Sequence clustering analysis proved particularly useful to handle the large diversity of log traces, as well as to discern and analyze different behavioral patterns, including regular/infrequent behavior and their variants. It was possible to discover clinical malpractice, potential waste of resources, performance bottlenecks, violation of service level agreements, and potential violation of internal practices. How professionals perceive reality differed in several occasions from the objective results provided by process mining. For a summary and discussion on main findings the reader is referred to the summary section of chapter 5, chapter 6, and chapter 7.

The process mining setting developed proved of extreme value for the hospital. Despite the limited resources and effort HSS allocates to perform process analysis, the hospital has now means to be in control of the emergency processes by relying on objective data. Most importantly, the cost of analysis was reduced to an extent that it is now feasible for the organization to perform it whenever is needed. The scalability of the solution is also worth to note. Although this work focused two important emergency processes and the organizational perspective of the department, in reality, any emergency process can be analyzed attending the patient diagnosis. This would be unfeasible to achieve or maintain with traditional process analysis. Moreover, to scale the solution to other departments is now just a matter of acquiring additional data from the hospital system.

8.2 Future Work

There are still points to improve and challenges to overcome in future work. A critical point is to improve the usability of process mining and the understandability of results to non-experts. IT and non-IT professionals showed interest in using process mining, but they find the concept very technical. On one hand, achieving results still implies clear understanding of the techniques involved. On the other hand, the models devised, such as Petri Nets or the sequence clustering models, can

be confusing to interpret without proper training. Heuristic graphs are better interpreted but, once more, it is difficult for a non-expert to use the technique. These issues clearly yields a barrier for effective internal use. Usability has to be improved to a minimum point at which professionals can be trained without needing deep technical understanding, otherwise the hospital is forced to invest in specialists, which is unlikely to happen given the economical panorama and organizational culture. For the sake of understandability, it is important to explore and find out which process models are of natural comprehension to health professionals.

It would also be valuable to include other kinds of data and process perspectives. From a management viewpoint, it is pertinent to extent the methodology and incorporate cost analysis in alignment with Activity Based Costing concepts, which seems a natural fit for the problem. Process mining current lacks of techniques to support the discovery and analysis of costs, and the development of such techniques would certainly be valuable to provide a fine grained understanding of how much is actually spent in a process, or to benchmark different practices and process flows. For a discussion of possible challenges and solutions please refer to the summary section of chapter 5. From a clinical viewpoint, it would be interesting to include data about the patient context (such as demographics, problems, medications, or history) as well as the patient outcomes. We believe that it would offer new possibilities of analysis. The patient context would offer the capability to filter or discover specific groups of patients, which could then be analyzed with the proposed methodology. The patient outcomes would offer the capability to measure the risk associated with different pathflows, which is important to support clinical decisions. For the particular case of HSS, it is challenging to acquire this additional data, because most of it is found in admission, discharge/transfer reports recorded in text free form, which is non-trivial to process and structure. Moreover, it deals with patient sensitive information which access is usually seen with reluctance due privacy concerns.

Despite the quick results delivered by process mining, comparing those results with implicit knowledge contained in text based, conventional medical guidelines is a laborious task prone to ambiguity. HSS could team-up the IT department with a group clinicians in order to formalize the most relevant guidelines, or at least some guideline rules. That would contribute to automate, or partial automate, both the detection of deviations and conformance measurement.

From a more technical side, and regarding the methodology proposed, the initial selection of clusters can be improved. Particularly, one needs to develop a proper technique to automatically indicate the initial number of clusters. In general, a too high number of clusters makes it difficult to identify typical behavior, since even small variations may cause similar variants to be scattered across several clusters with relative low support. On the other hand, using a very low number of clusters will aggregate different behaviors in each cluster, which produce cluster models that are complex to interpret. An automated method to handle this trade-off is important because the manual counterpart is time consuming and requires domain expertise.

The limitations of first-over Markov Chains to express non-local dependencies and activities performed in parallel became evident in chapter 6. In a sense that one easily overcomes the

problem by running control-flow mining techniques with stronger semantic expressiveness on top of the traces associated with each cluster, it justifies to apply these mining techniques directly on each cluster, so that one can directly obtain richer process models, such as heuristic graphs.

8.3 List of Publications

- Álvaro Rebuge, Diogo R. Ferreira, *Business process analysis in healthcare environments: A methodology based on process mining*, Information Systems, vol. 37, no. 2, pp. 99-116, April 2012

References

[1] Martin Lang, Thomas Bürkle, Susanne Laumann, and Hans-Ulrich Prokosch. Process Mining for Clinical Workflows: Challenges and Current Limitations. In *Proceedings of MIE2008, The XXIst International Congress of the European Federation for Medical Informatics*, pages 229–234. IOS Press, 2008. xi, 4, 19, 20, 21

[2] M. Bozkaya, J. Gabriels, and J.M. van der Werf. Process Diagnostics: A Method Based on Process Mining. In *International Conference on Information, Process, and Knowledge Management (eKNOW '09)*, pages 22–27, 2009. xi, 7, 23, 24, 32

[3] M. Poulymenopoulou, F. Malamateniou, and G. Vassilacopoulos. Specifying workflow process requirements for an emergency medical service. *Journal of medical systems*, 27(4):325–335, 2003. 1, 2, 3, 4

[4] P. Dadam, M. Reichert, and K. Kuhn. Clinical workflows-the killer application for process-oriented information systems. In *Proc. 4th Int. Conf. on Business Information Systems*, pages 36–59, 2000. 1

[5] R. Lenz and M. Reichert. IT support for healthcare processes–premises, challenges, perspectives. *Data & Knowledge Engineering*, 61(1):39–58, 2007. 1, 2, 3

[6] K. Anyanwu, A. Sheth, J. Cardoso, J. Miller, and K. Kochut. Healthcare enterprise process development and integration. *Journal of Research and Practice in Information Technology*, 35(2):83–98, 2003. 1, 3

[7] Institute of Medicine. *Crossing the quality chasm: a new health system for the 21st century.* National Academy Press, Washington, DC, 2001. 1

[8] Linda T. Kohn, Janet M. Corrigan, and Molla S. Donaldson. *To Err Is Human: Building a Safer Health System.* National Academy Press, Washington, DC, 2000. 1

[9] Luís Velez Lapão. Survey on the status of the hospital information systems in Portugal. *Methods of Information in Medicine*, 46(4):493–499, 2007. 1

[10] R. Lenz and K. A. Kuhn. Towards a continuous evolution and adaptation of information systems in healthcare. *International Journal of Medical Informatics*, 73(1):75 – 89, 2004. 1, 3

[11] W.M.P. van der Aalst, A.H.M. Hofstede, and M. Weske. Business Process Management: A Survey. *Lecture Notes in Computer Science*, 2678:1–12, 2003. 1, 11

[12] M. Weske, W. M. P. van der Aalst, and H. M. W. Verbeek. Advances in business process management. *Data & Knowledge Engineering*, 50(1):1–8, 2004. 1, 2

[13] G. Darnton and M Darton. *Business Process Analysis*. International Thompson Business Press, 1997. 2, 4

[14] C. Gramlich. Business Process Analysis. *Process-oriented Information Systems*. 2

[15] S. Biazzo. Approaches to business process analysis: a review. *Business Process Management Journal*, 6(2):99–112, 2000. 2

[16] S. Junginger and E. Kabel. Business Process Analysis. *eBusiness in Healthcare*, pages 57–77, 2008. 2

[17] T.H. Davenport. *Process innovation: reengineering work through information technology*. Harvard Business Press, 1993. 2

[18] M. Hammer and J. Champy. Reengineering the corporation, 1993. 2

[19] Shaifali Gupta. Workflow and process mining in healthcare. Master's thesis, Technische Universiteit Eindhoven, 2007. 3, 20, 21

[20] R. Mans, H. Schonenberg, G. Leonardi, S. Panzarasa, A. Cavallini, S. Quaglini, and W. van der Aalst. Process mining techniques: an application to stroke care. *Stud Health Technol Inform*, 136:573–8, 2008. 3, 19, 20, 21

[21] R.S. Mans, M.H. Schonenberg, M. Song, W.M.P. van der Aalst, and P.J.M. Bakker. Application of Process Mining in Healthcare – A Case Study in a Dutch Hospital. In *Biomedical Engineering Systems and Technologies*, number 25 in Communications in Computer and Information Science, pages 425–438. Springer, 2009. 3, 4, 19, 20, 21

[22] B.F. van Dongen. *Process Mining and Verification*. PhD thesis, Technische Universiteit Eindhoven, 2007. 4, 12

[23] M. Weske. *Business Process Management: Concepts, Languages, Architectures*. Springer, 2007. 4, 11

[24] WMP van Der Aalst, AHM Ter Hofstede, B. Kiepuszewski, and AP Barros. Workflow patterns. *Distributed and parallel databases*, 14(1):5–51, 2003. 4, 13

[25] J.D. Sterman. System dynamics modeling for project management. Technical report, MIT Sloan School of Management, 1992. 4

[26] J.D. Sterman. Modeling managerial behavior: misperceptions of feedback in a dynamic decision making experiment. *Management science*, 35(3):321–339, 1989. 4

[27] A. Vasconcelos, R. Mendes, and J. Tribolet. Using organizational modeling to evaluate health care IS/IT projects. In *Proceedings of 37th Annual Hawaii International Conference On System Sciences (HICCS37), Hawaii, USA*, 2004. 4

[28] WMP van Der Aalst, HA Reijers, A. Weijters, BF Van Dongen, AK Alves de Medeiros, M. Song, and HMW Verbeek. Business process mining: An industrial application. *Information Systems*, 32(5):713–732, 2007. 4, 12, 13

[29] M.B. Miles. Qualitative data as an attractive nuisance: The problem of analysis. *Administrative Science Quarterly*, 24(4):590–601, 1979. 6

[30] G.G. Gable. Integrating case study and survey research methods: an example in information systems. *European Journal of Information Systems*, 3(2):112–126, 2010. 6

[31] R. Galliers. *Information Systems Research: Issues, Methods and Practical Guidelines*. Blackwell Scientific Publications, 1994. 6

[32] R.K. Yin. *Applications of case study research*. Sage Publications, Inc, 2003. 6

[33] W. van der Aalst, T. Weijters, and L. Maruster. Workflow mining: Discovering process models from event logs. *IEEE Transactions on Knowledge and Data Engineering*, 16(9):1128–1142, 2004. 13

[34] T. Murata. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4):541–580, 1989. 13

[35] L. Wen, J. Wang, and J. Sun. Detecting implicit dependencies between tasks from event logs. *Lecture Notes in Computer Science*, 3841:591–603, 2006. 13

[36] B.F. van Dongen and W.M.P. van der Aalst. Multi-phase process mining: Building instance graphs. *Lecture notes in computer science*, 3288:362–376, 2004. 14

[37] A.J.M.M. Weijters, W.M.P. van der Aalst, and A.K. Alves de Medeiros. Process mining with the heuristics miner algorithm. BETA Working Paper Series WP 166, Eindhoven University of Technology, 2006. 14

[38] C.W. Günther and W.M.P. van der Aalst. Fuzzy Mining – Adaptive Process Simplification Based on Multi-perspective Metrics. *Lecture Notes in Computer Science*, 4714:328–343, 2007. 14

[39] A. K. Alves De Medeiros and A. J. M. M. Weijters. Genetic Process Mining. *Lecture Notes in Computer Science*, 3536:48–69, 2005. 15

[40] W.M.P. van der Aalst and M. Song. Mining Social Networks: Uncovering Interaction Patterns in Business Processes. *Lecture Notes in Computer Science*, 3080:244–260, 2004. 15, 40, 80

[41] M. Song and W.M.P. van der Aalst. Towards comprehensive support for organizational mining. *Decision Support Systems*, 46(1):300–317, 2008. 15, 80

[42] A. Rozinat and W.M.P. van der Aalst. Decision mining in ProM. *Lecture Notes in Computer Science*, 4102:420–425, 2006. 16

[43] Peter T.G. Hornix. Performance analysis of business processes through process mining. Master's thesis, Eindhoven University of Technology, 2007. 16

[44] M. Song and W. van der Aalst. Supporting process mining by showing events at a glance. In *Proceedings of the Seventeenth Annual Workshop on Information Technologies and Systems*, pages 139–145, 2007. 16

[45] A. Rozinat and WMP van der Aalst. Conformance checking of processes based on monitoring real behavior. *Information Systems*, 33(1):64–95, 2008. 17

[46] A. Rozinat and W.M.P. van der Aalst. Conformance testing: Measuring the fit and appropriateness of event logs and process models. *Lecture Notes in Computer Science*, 3812:163–176, 2006. 17

[47] W.M.P. van der Aalst, H.T. de Beer, and B.F. van Dongen. Process mining and verification of properties: An approach based on temporal logic. *Lecture notes in computer science*, 3760:130–147, 2005. 17

[48] Gabriel M. Veiga and Diogo R. Ferreira. Understanding spaghetti models with sequence clustering for ProM. In *Business Process Intelligence (BPI 2009): Workshop Proceedings*, Ulm, Germany, September 2009. 17

[49] Ana Karla Alves de Medeiros, Antonella Guzzo, Gianluigi Greco, Wil M. P. van der Aalst, A. J. M. M. Weijters, Boudewijn F. van Dongen, and Domenico Saccà. Process Mining Based on Clustering: A Quest for Precision . *Lecture Notes in Computer Science*, 4928:17–29, 2008. 17

[50] M. Song, C.W. Günther, and W.M.P. van der Aalst. Trace Clustering in Process Mining. *Lecture Notes in Business Information Processing*, 17:109–120, 2008. 17

[51] Gabriel Veiga. Developing Process Mining Tools, An Implementation of Sequence Clustering for ProM. Master's thesis, IST – Technical University of Lisbon, 2009. 17, 18, 21, 26, 31

[52] Diogo R. Ferreira, Marielba Zacarias, Miguel Malheiros, and Pedro Ferreira. Approaching Process Mining with Sequence Clustering: Experiments and Findings. *Lecture Notes in Computer Science*, 4714:360–374, 2007. 17, 21

[53] D.R. Ferreira and M. Mira da Silva. Using process mining for ITIL assessment: a case study with incident management. In *Proceedings of the 13th Annual UKAIS Conference, Bournemouth University*, 2008. 17, 21

[54] Diogo R. Ferreira. Applied sequence clustering techniques for process mining. In Jorge Cardoso and Wil van der Aalst, editors, *Handbook of Research on Business Process Modeling*, Information Science Reference, pages 492–513. IGI Global, 2009. 17, 21

[55] B.F. van Dongen, A.K.A. de Medeiros, HMW Verbeek, A. Weijters, and W.M.P. van der Aalst. The ProM framework: A new era in process mining tool support. *Lecture Notes in Computer Science*, 3536:444–454, 2005. 18

[56] B.F. van Dongen and W.M.P. van der Aalst. A Meta Model for Process Mining Data. In *Proceedings of the CAiSE'05 Workshops (EMOI-INTEROP Workshop)*, pages 309–320, 2005. 18

[57] C. Gunther and W. van der Aalst. A generic import framework for process event logs. In *Business Process Management Workshops*, pages 81–92. Springer, 2006. 18

[58] Diogo R. Ferreira and Daniel Gillblad. Discovering process models from unlabelled event logs. *Lecture Notes in Computer Science*, 5701:143–158, 2009. 21

[59] I. Cadez, D. Heckerman, C. Meek, P. Smyth, and S. White. Visualization of navigation patterns on a web site using model-based clustering. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 280–284. ACM New York, NY, USA, 2000. 26

[60] Daniel Gillblad, Diogo R. Ferreira, and Rebecca Steinert. Estimating the parameters of randomly interleaved markov models. In *The 1st Workshop on Large-scale Data Mining: Theory and Applications, in conjunction with ICDM 2009, December 6-9, Miami, FL, USA*, 2009. 28

[61] C. Li, M. Reichert, and A. Wombacher. Mining process variants: Goals and issues. In *IEEE International Conference on Services Computing*, pages 573–576, 2008. 30

[62] C. Li, M. Reichert, and A. Wombacher. Discovering reference process models by mining process variants. In *Proceedings of the 2008 IEEE International Conference on Web Services*, pages 45–53, 2008. 30

[63] Alena Hallerbach, Thomas Bauer, and Manfred Reichert. Managing Process Variants in the Process Lifecycle. In *10th Int'l Conf. on Enterprise Information Systems (ICEIS'08)*, pages 154–161, 2008. 30

[64] R.L. Graham and P. Hell. On the history of the minimum spanning tree problem. *Annals of the History of Computing*, 7(1):43–57, 1985. 30

[65] EW Dijkstra. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271, 1959. 30

[66] Joseph B. Kruskal. On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem. *Proceedings of the American Mathematical society*, 7(1):48–50, 1956. 30

[67] Zhaohui Tang and Jamie MacLennan. *Data mining with SQL Server 2005*. Wiley, 2005. 39

[68] M.A. Smith, B. Shneiderman, N. Milic-Frayling, E. Mendes Rodrigues, V. Barash, C. Dunne, T. Capone, A. Perer, and E. Gleave. Analyzing (social media) networks with NodeXL. In *Proceedings of the fourth international conference on Communities and technologies*, pages 255–264. ACM, 2009. 40

[69] K. Wakita and T. Tsurumi. Finding community structure in mega-scale social networks. In *Proceedings of the 16th international conference on World Wide Web*, pages 1275–1276. ACM, 2007. 40

[70] Marie Laure Delignette-Muller, Regis Pouillot, Jean-Baptiste Denis, and Christophe Dutang. *fitdistrplus: help to fit of a parametric distribution to non-censored or censored data*, 2010. R package version 0.1-3. 55

[71] R.B. D'Agostino and M.A. Stephens. *Goodness-of-fit techniques*. CRC, 1986. 55

[72] A.C. Cullen and H.C. Frey. *Probabilistic techniques in exposure assessment: a handbook for dealing with variability and uncertainty in models and inputs*. Springer, 1999. 55

[73] S. Swain, C. Turner, P. Tyrrell, and A. Rudd. Diagnosis and initial management of acute stroke and transient ischaemic attack: summary of nice guidance. *BMJ*, 337, 2008. 61

[74] C.J.M. Klijn and G.J. Hankey. Management of acute ischaemic stroke: new guidelines from the american stroke association and european stroke initiative. *The Lancet Neurology*, 2(11):698–701, 2003. 61

[75] H.P. Adams, G. Del Zoppo, M.J. Alberts, D.L. Bhatt, L. Brass, A. Furlan, R.L. Grubb, R.T. Higashida, E.C. Jauch, C. Kidwell, et al. Guidelines for the early management of adults with ischemic stroke. *Circulation*, 115(20):e478–e534, 2007. 61, 65

[76] Q. Shi, R. Presutti, D. Selchen, and G. Saposnik. Delirium in acute stroke. *Stroke*, 43(3):645–649, 2012. 66

[77] R.G. González. *Acute ischemic stroke: imaging and intervention*. Springer Verlag, 2010. 70

[78] D.L. Sackett, W. Rosenberg, JA Gray, R.B. Haynes, and W.S. Richardson. Evidence based medicine: what it is and what it isn't. *Bmj*, 312(7023):71, 1996. 76

[79] M. Peleg, S. Tu, J. Bury, P. Ciccarese, J. Fox, R.A. Greenes, R. Hall, P.D. Johnson, N. Jones, A. Kumar, et al. Comparing computer-interpretable guideline models: a case-study approach. *Journal of the American Medical Informatics Association*, 10(1):52, 2003. 77

[80] P.A. de Clercq, J.A. Blom, H.H.M. Korsten, and A. Hasman. Approaches for creating computer-interpretable guidelines that facilitate decision support. *Artificial Intelligence in Medicine*, 31(1):1–27, 2004. 77

[81] R. McGill, J.W. Tukey, and W.A. Larsen. Variations of box plots. *American Statistician*, 32(1):12–16, 1978. 82

[82] M. Frigge, D.C. Hoaglin, and B. Iglewicz. Some implementations of the boxplot. *American Statistician*, pages 50–54, 1989. 82

Appendices

Appendix A

Medtrix Process Mining Studio Screenshots

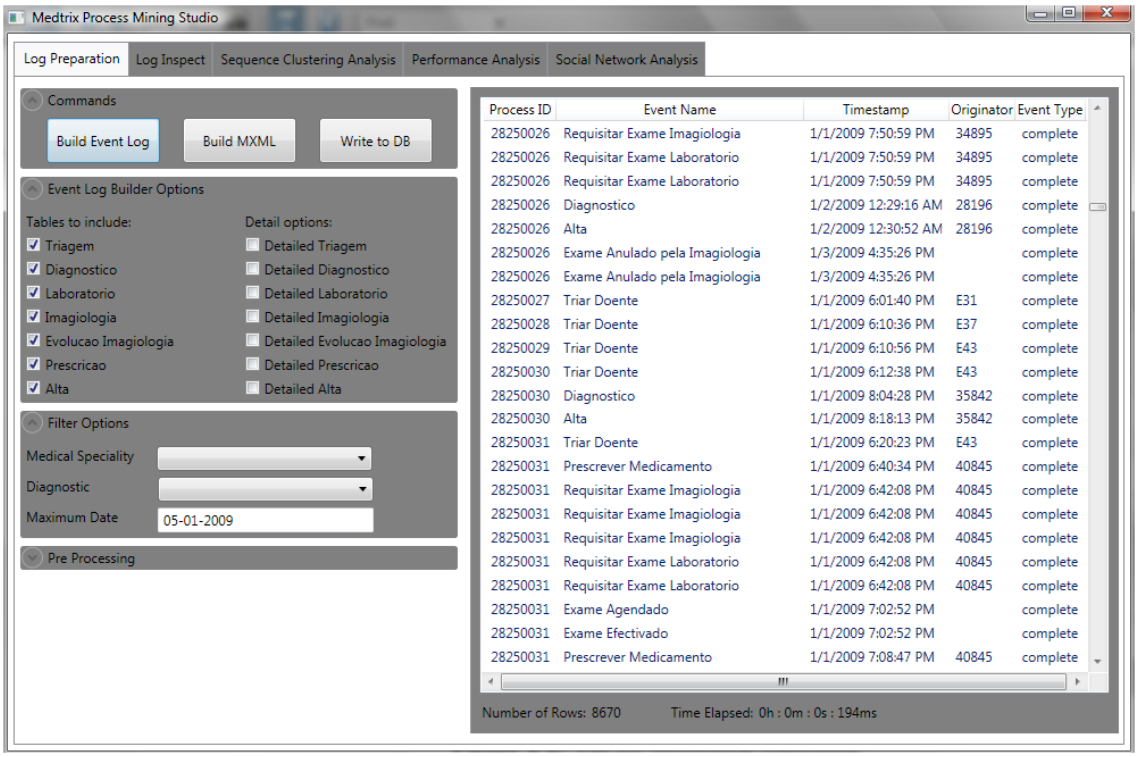

Figure A.1: Medtrix Process Mining Studio - Screenshot of the log preparation component.

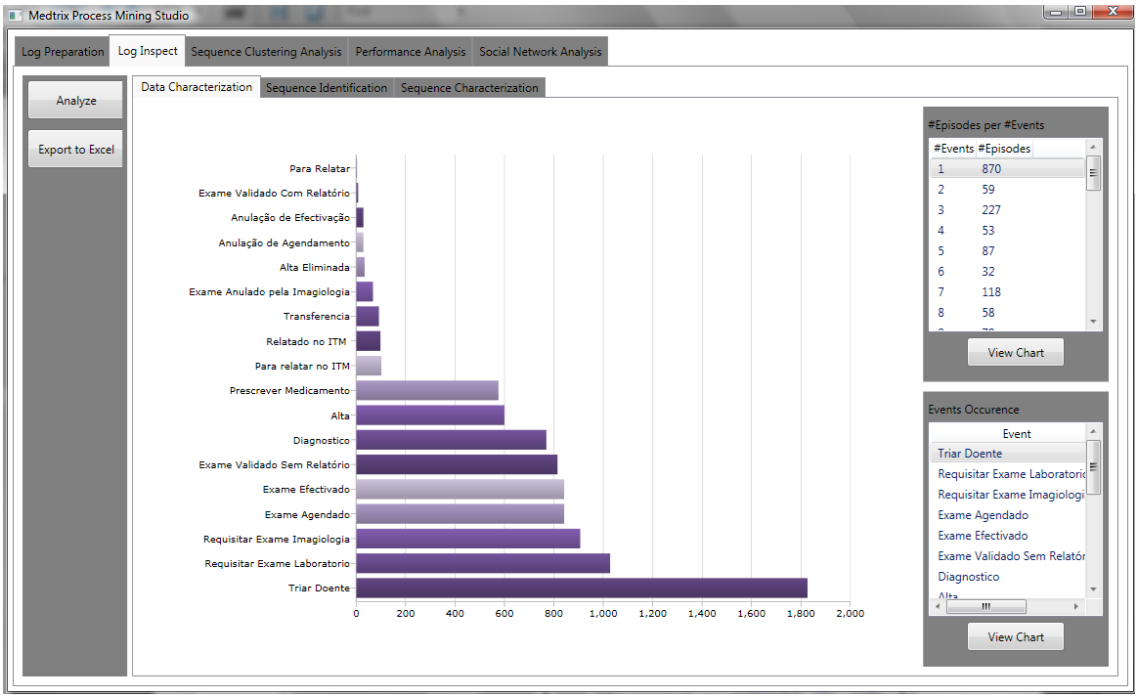

Figure A.2: Medtrix Process Mining Studio - Screenshot of the log inspector component.

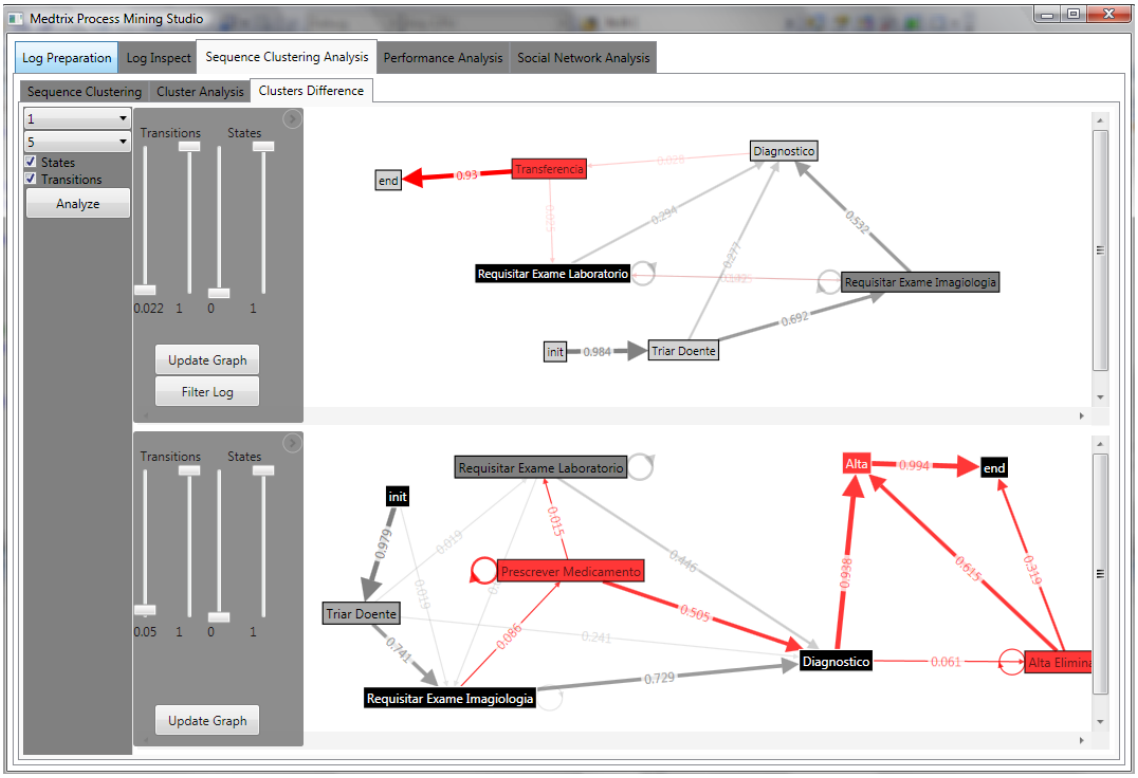

Figure A.3: Medtrix Process Mining Studio - Screenshot of the cluster analysis component (clusters differences).

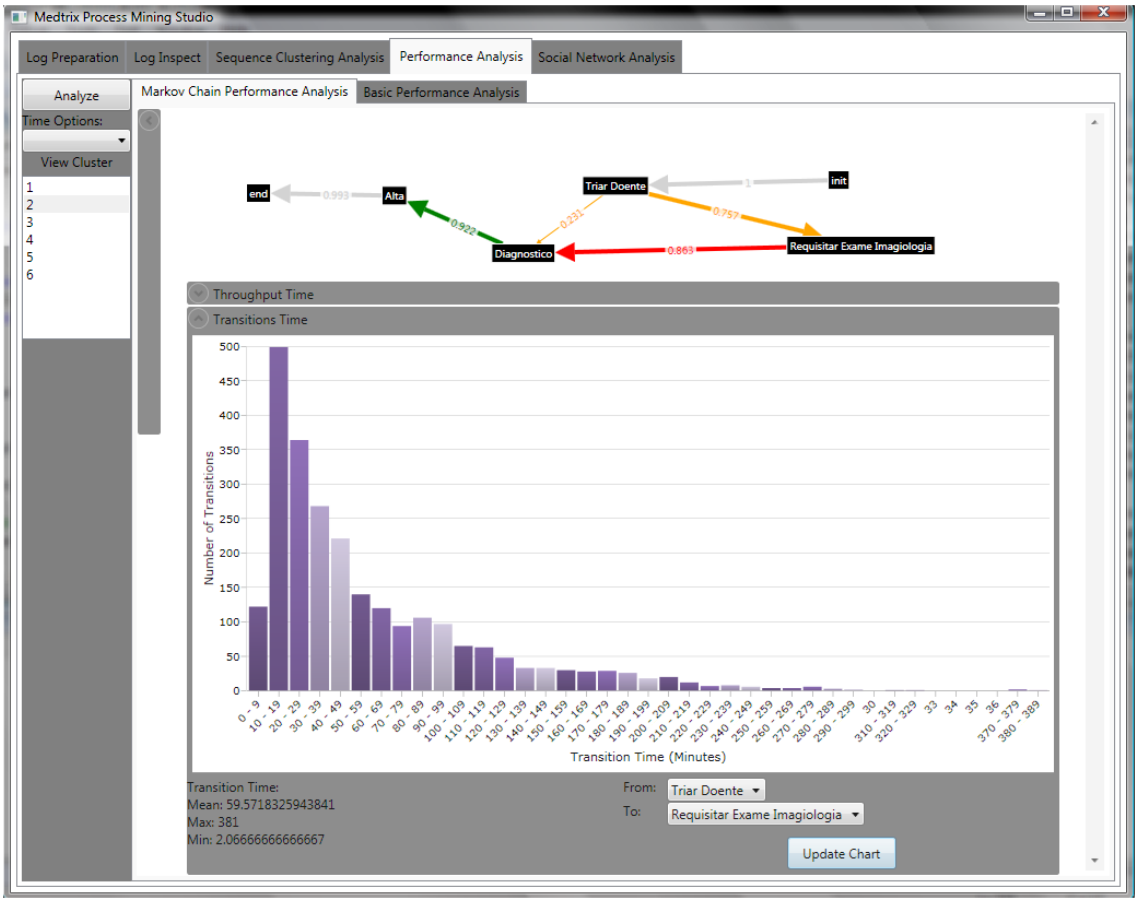

Figure A.4: Medtrix Process Mining Studio - Screenshot of the performance analysis component.

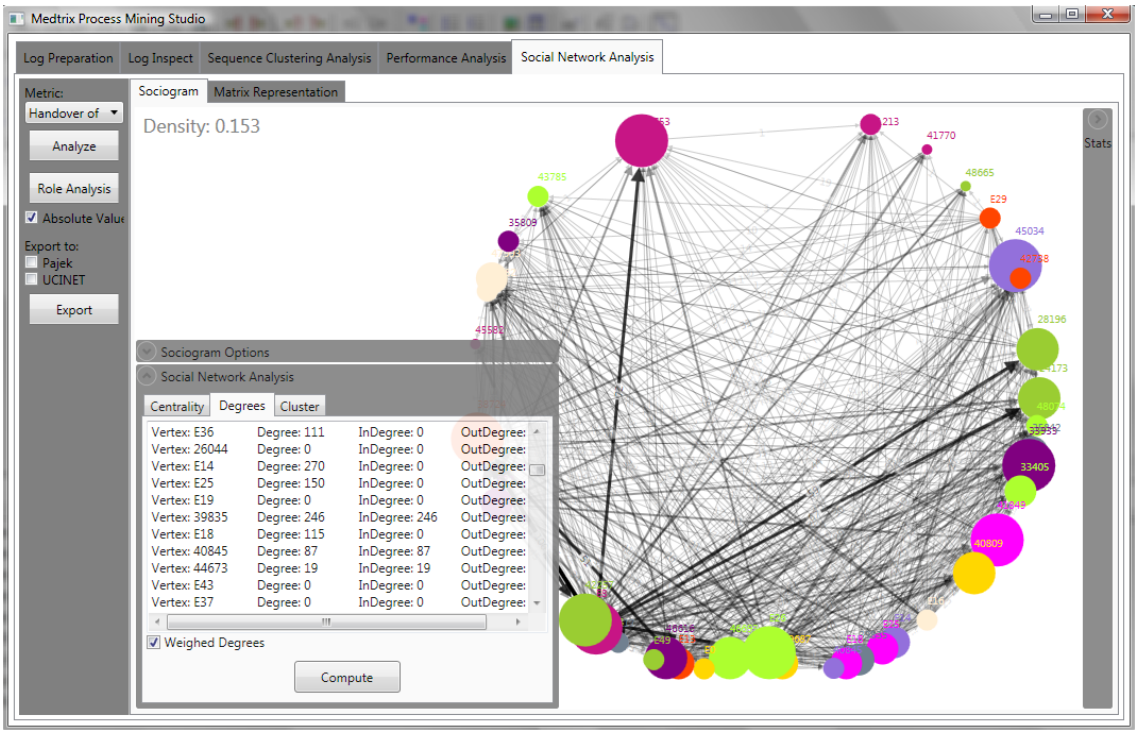

Figure A.5: Medtrix Process Mining Studio - Screenshot of the social network analysis component.

Appendix B

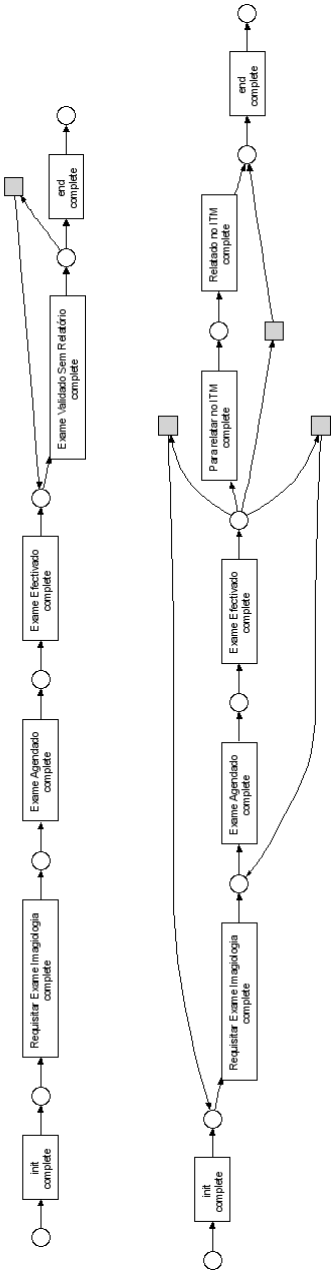

Figure A.6: Emergency Radiology Process - Petri Nets obtained from cluster 2 (left) and from cluster 7.2 (right).

Figure A.7: Emergency Radiology Process - Petri Net modeling the global process.

Appendix C

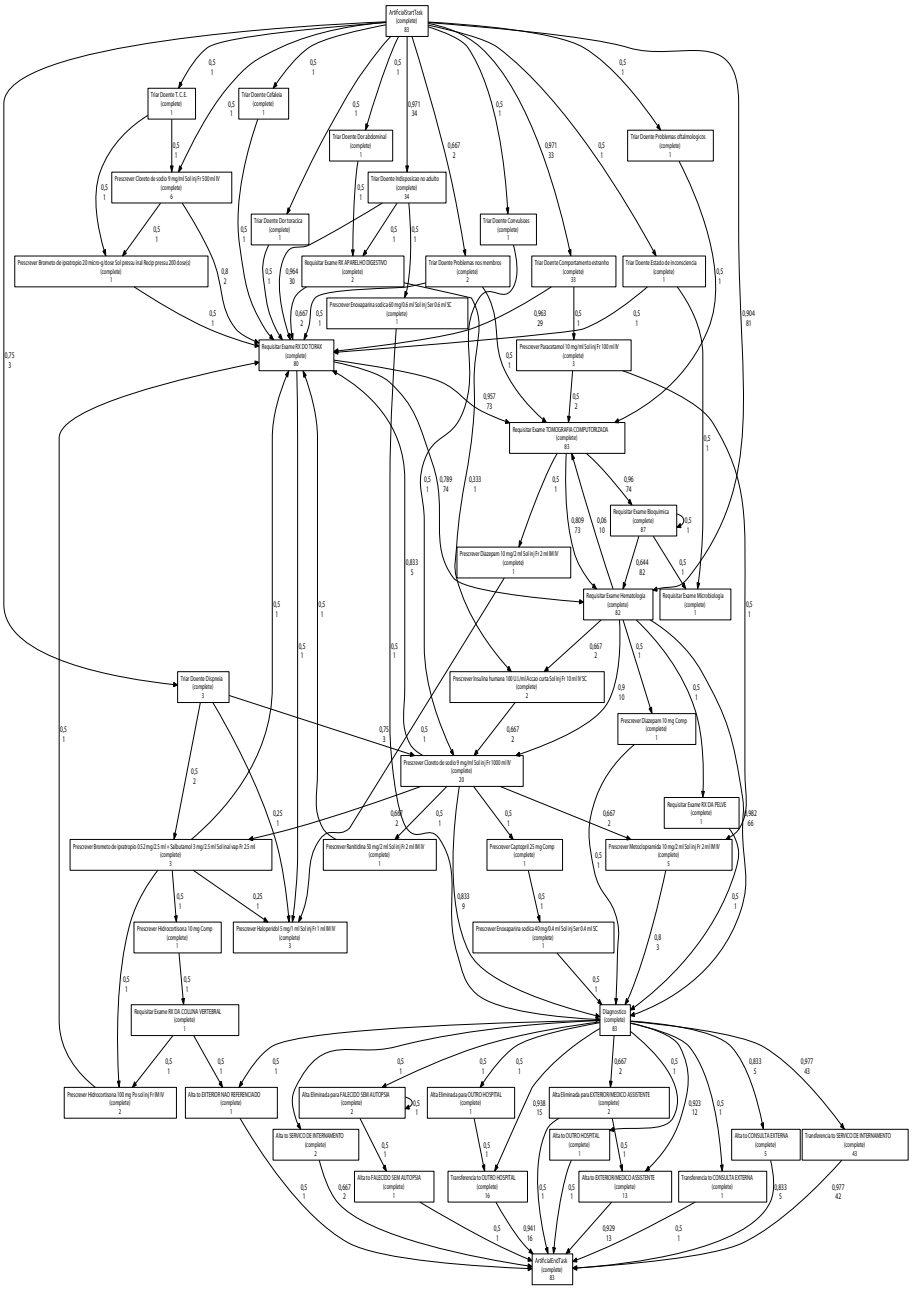

Figure A.8: Acute Stroke Process - Heuristic graph depicting global behavior.

Cluster	Support	Triage	Treatment	Exams	Discharge/Transfer	Observations
c1	0.279	Adult in unwell condition; or Skipped	Normal Saline Solution	Chest X-Ray => CT => Biochemistry => Blood	Hospitalization	The use of normal saline solution indicates patients needing fluid replacement. Triage skipped sometimes, meaning very urgent cases.
c2	0.089	Adult in unwell condition	None	Chest X-Ray => CT => Biochemistry => Blood	External Appointment	Specific cases of stroke patients discharged with a note to an external appointment.
c3	0.054	Brain Trauma; or Unconscious State; or Adult in Unwell Condition; or Strange Behavior	Human Insulin => Normal Saline Solution; Normal Saline Solution => Ipratropium Bromide	Chest X-Ray; CT; Biochemistry; Blood; Microbiology (no particular order)	External Hospital or Hospitalization	Indication of specific cases: with brain trauma, potentially referring to hemorrhagic stroke; in unconscious state; with hyperglycemia and respiratory complications, thus the prescription of human insulin and ipratropium bromide, respectively; needing a microbiology test. Good candidate for hierarchical sequence clustering.
c4	0.041	Strange Behavior	Normal Saline Solution; Haloperidol	Chest X-Ray; CT; Biochemistry; Blood (no particular order)	External Hospital	The prescription of Haloperidol and strange behavior triage suggests specific cases of patients presenting delirium onset. HSS opted to transfer these patients to an external hospital.
c6	0.068	Adult in Unwell Condition; or Ophthalmology Disorders; or Arms/Legs Disorders	Human Insulin => Normal Saline Solution => Metoclopramide	Chest X-Ray; CT; Biochemistry; Blood; Digestive System X-Ray (no particular order)	Hospitalization; or External Supporting Doctor	Indication of specific cases of patients with hyperglycemia, nausea and vomiting complications; thus the prescription of human insulin (for hyperglycemia management), the prescription of metoclopamide (for nausea and vomiting), and the x-ray of the digestive system; There is also indication of specific cases of ophthalmological and leg/arms related complications; thus triaging according oftalmological or arms/legs problems; Good candidate for hierarchical sequence clustering.
c7	0.054	Adult in Unwell Condition; or Strange Behavior	Paracetamol => Normal Saline Solution; Normal Saline Solution	Chest X-Ray; CT; Biochemistry; Blood (no particular order)	Hospitalization; or External Hospital; or External Supporting Doctor	The prescription of Paracetamol suggests specific cases suffering from pyrexia (fever).
c8	0.081	Convulsions; or Dyspnea (diminished breathing); or Adult in Unwell Condition; or Strange Behavior	Normal Saline Solution; Ipratropium Bromide + Salbutemol => Hydrocortisone; Diazepam => Haloperidol	Chest X-Ray; CT; Biochemestry; Blood (no particular order)	Hospitalization; or External Hospital; or External Supporting Doctor	Indication of specific cases of stroke patients with pulmonary disorders; thus the dyspnea triaging and prescription of ipratropium bromide + salbutemol and hydrocortisone. Indication of stroke patients with convulsions; thus the convulsions triaging and prescription of diazepam and haloperidol. Good candidate for hierarchical sequence clustering.

Figure A.9: Acute Stroke Process - Summary of main findings.

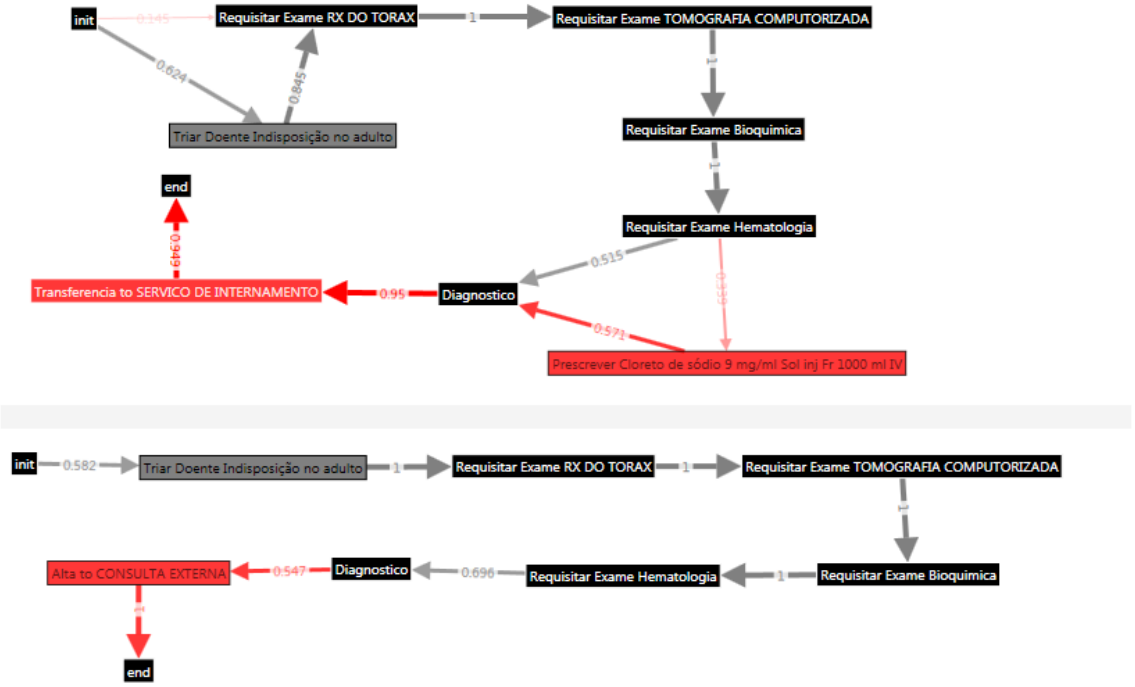

Figure A.10: Acute Stroke Process - Differences between cluster 1 (on top) and cluster 2 (on bottom)

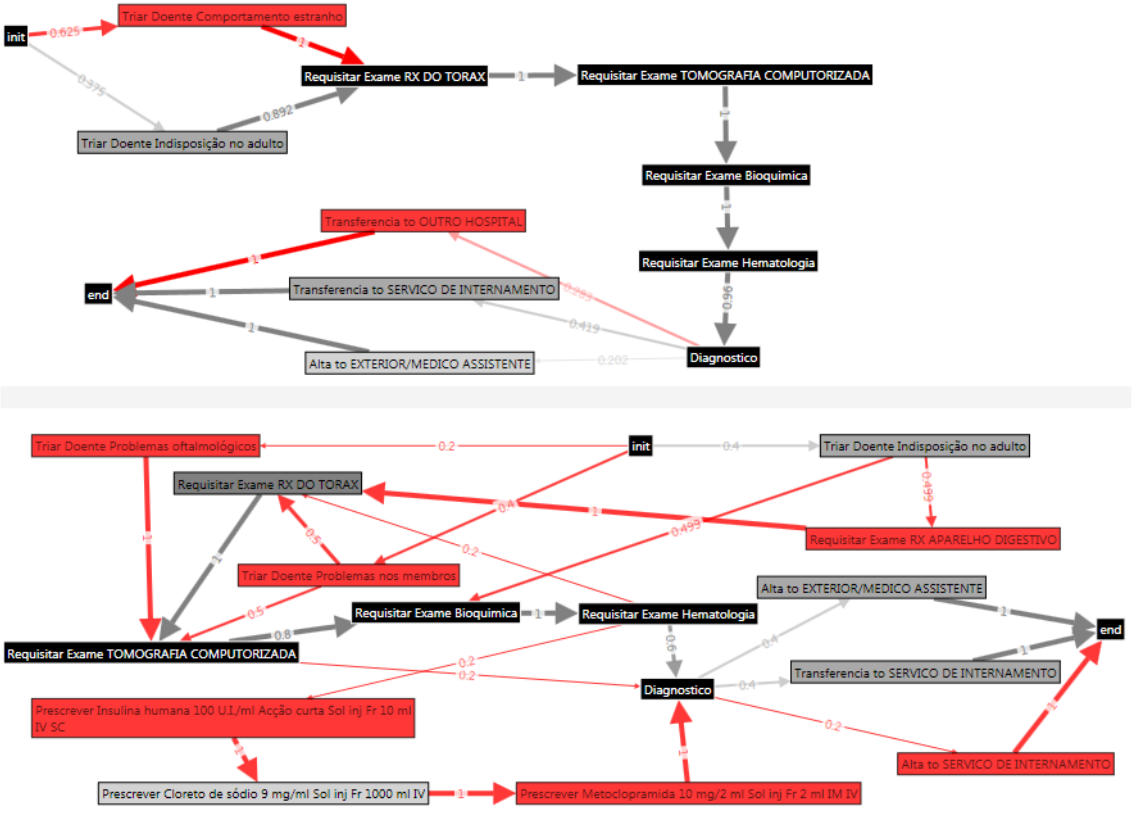

Figure A.11: Acute Stroke Process - Differences between cluster 5 (on top) and cluster 6 (on bottom)

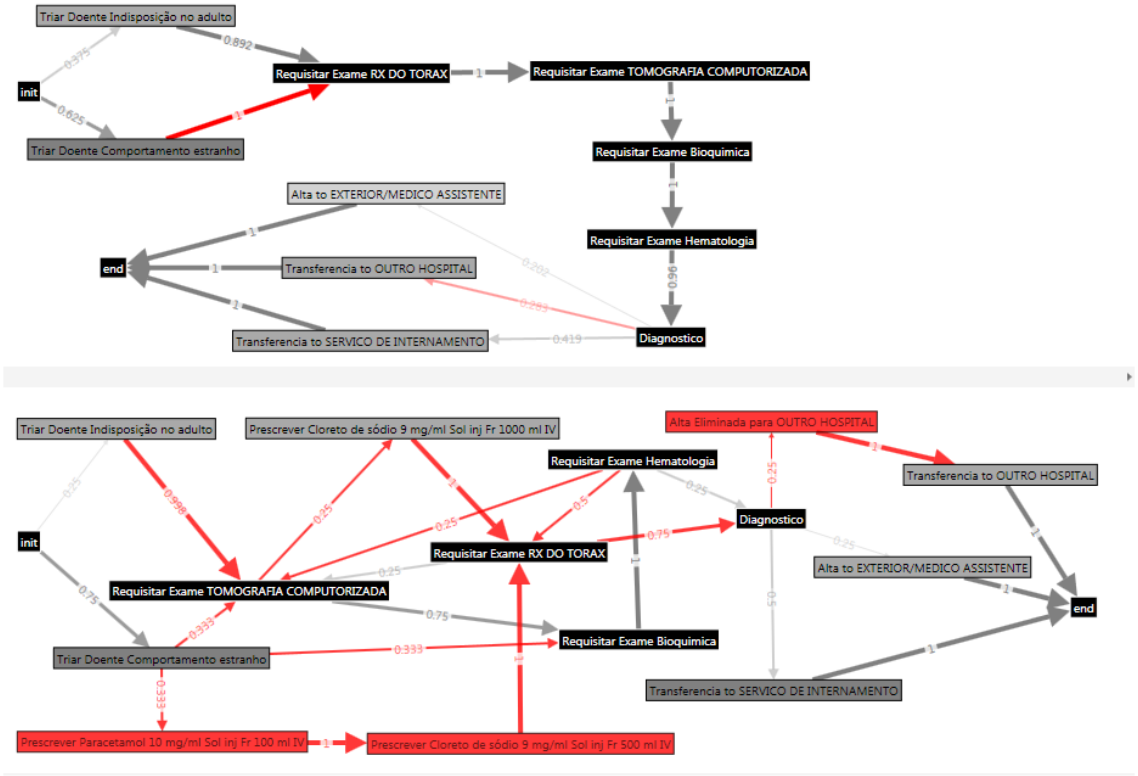

Figure A.12: Acute Stroke Process - Differences between cluster 5 (on top) and cluster 7 (on bottom)

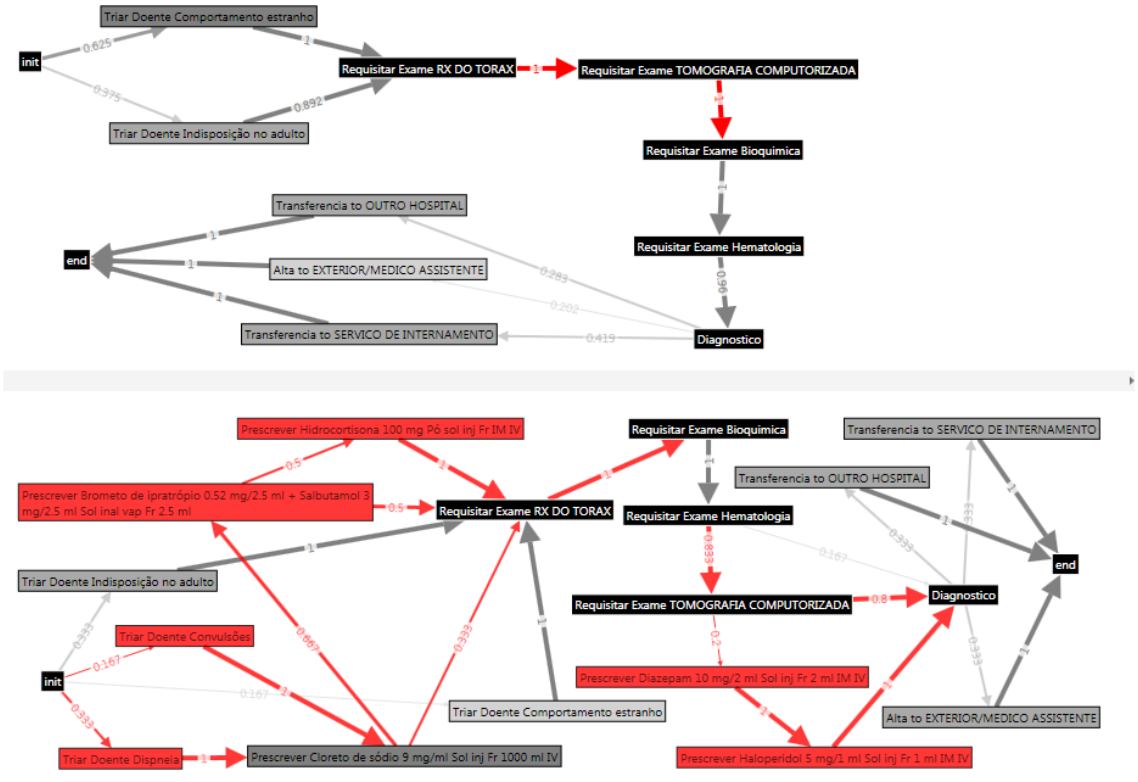

Figure A.13: Acute Stroke Process - Differences between cluster 5 (on top) and cluster 8 (on bottom)

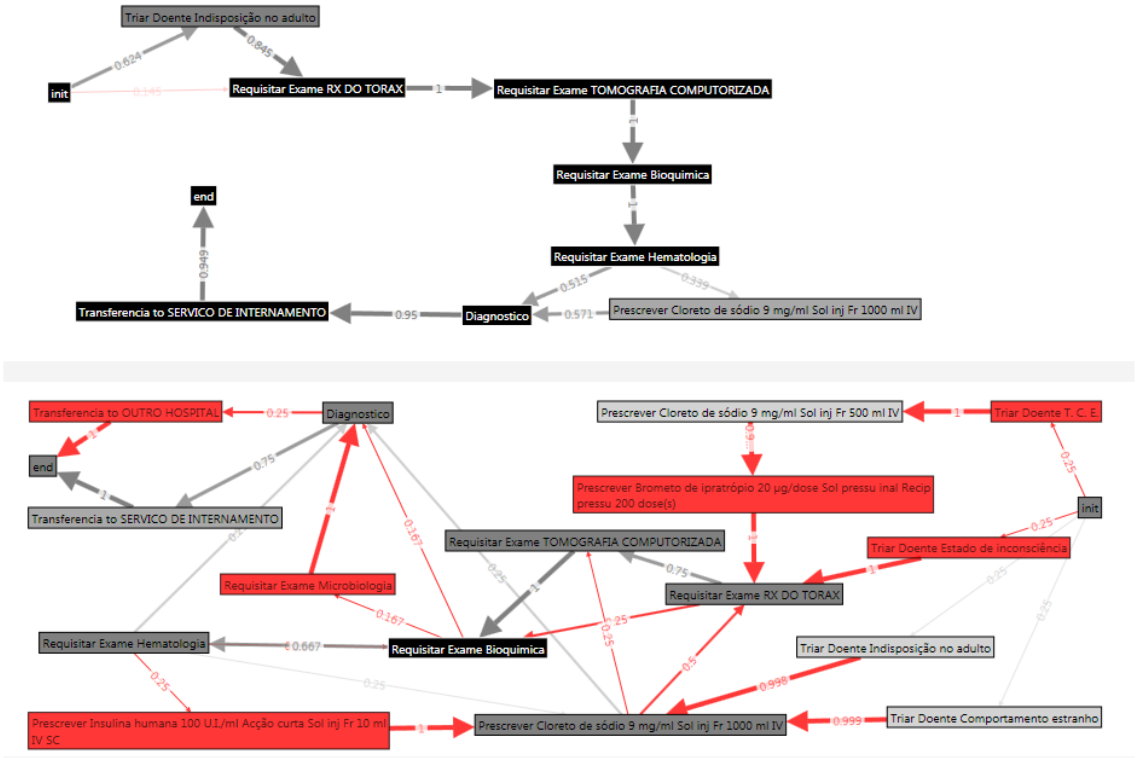

Figure A.14: Acute Stroke Process - Differences between cluster 1 (on top) and cluster 3 (on bottom)

Appendix D

In order to overcome semantic limitations of first-order Markov Chains, as discussed in Section 6.2 during sequence clustering analysis of the acute stroke process, one needs to apply control-flow mining techniques capable to discover parallelisms or non-local dependencies. In essence, techniques with richer semantic expressiveness. Since one obtains less complex models with clustering, now one can better exploit the strengths of other mining techniques, which would generate confusing models otherwise. The Heuristic Miner is an example. It is capable to infer parallelisms and non-local dependencies, although it generates confusing models in presence of log traces that significantly vary from each other.

To proof the concept, let us show the results obtained when applying the Heuristic Miner to cases associated with cluster 8 (Figure A.13). This cluster is a good example because it combines parallelisms and non-local dependencies. Figure A.15 depicts part of the resulting process model (a heuristic graph). It is interesting to note the non-local behavior detected. Figure A.15 shows that ipatropium bromide + salbutemol (and hydro-cortisone as well) was prescribed only when patients were triaged with dyspnea. Other example is the request of an x-ray of the digestive system, appearing after the request of a chest-xray but affected by the triage of a patient with abdominal pain ("Triar Doente Dor Abdominal").

To depict the latter example more precisely, Figure A.16 displays the split/join semantics of the model. With this, one can understand points of synchronism and parallelisms. Note the activity "Triar Doente Dor Abdominal" (abdominal pain triage). The "XOR and XOR" semantics is indicating an AND split. That is, the activities referred by the outgoing XOR arcs are performed in parallel: "Requisitar Exame RX Torax" (chest x-ray request) and "Requisitar Exame RX Aparelho Digestivo" (digestive system x-ray request). On the other hand, the "XOR and XOR" semantics at activity "Requisitar Exame RX Aparelho Digestivo" is indicating an AND join (a point of synchronism). It is showing that, in order to request a x-ray of the digestive system, a patient has to be previously triaged with abdominal pain. A chest x-ray also needs to be requested, but this is observed independently of the triage made. In other words, "when at" chest x-ray request the choice for a x-ray of the digestive system will depend on whether a patient was previously triaged with abdominal pain or not. This is an example of a non-free choice, a pattern that is not trivial to discover with process mining.

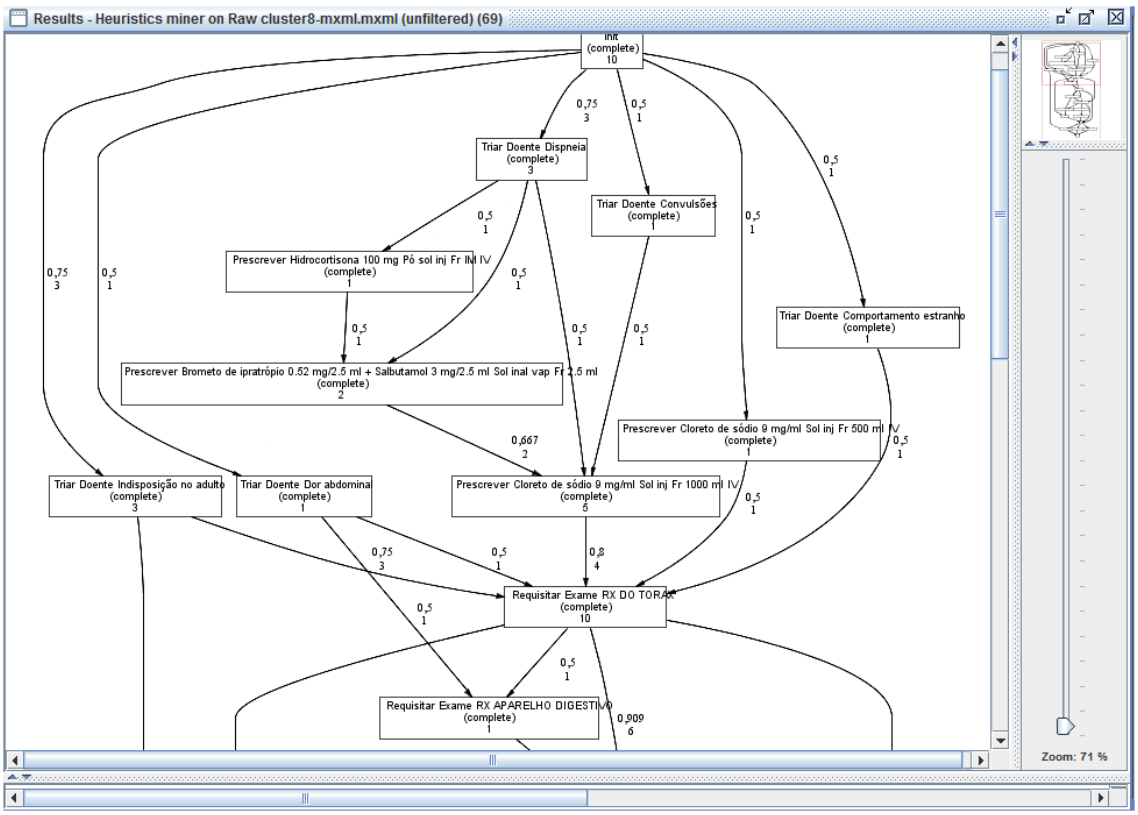

Figure A.15: Acute Stroke Process - Part of the dependency graph obtained by applying Heuristic Miner to cluster 8.

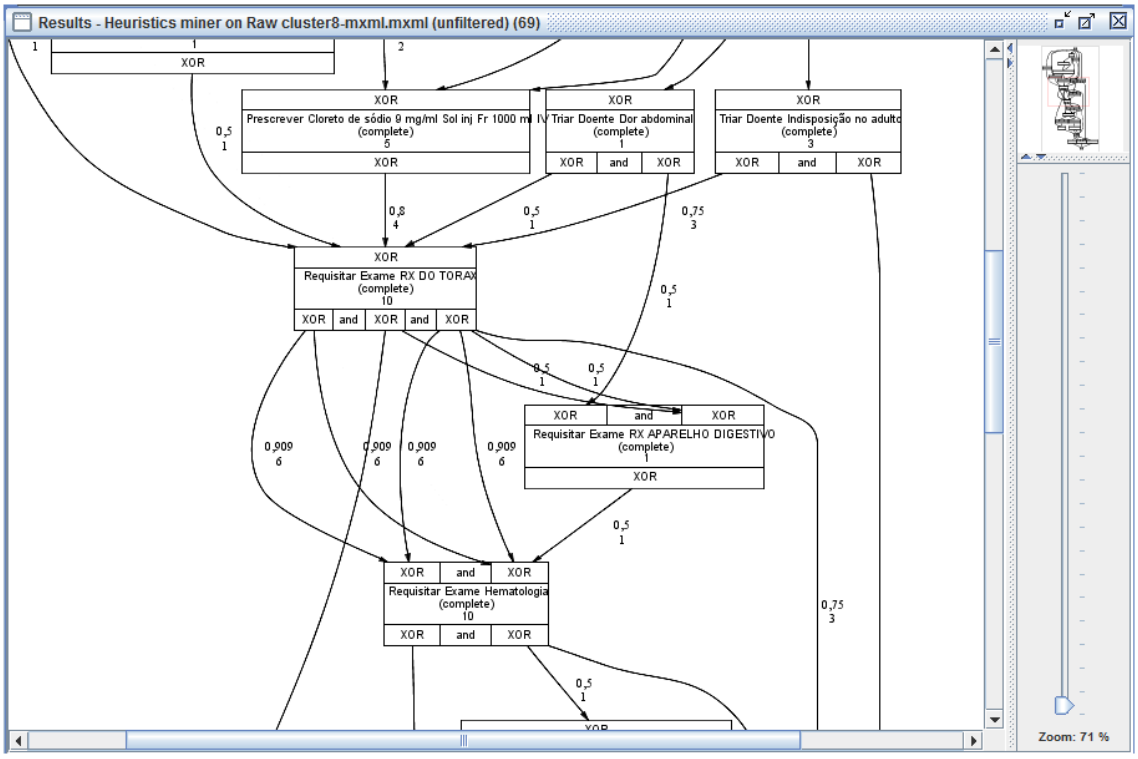

Figure A.16: Acute Stroke Process - Part of the heuristic net obtained by applying Heuristic Miner to cluster 8 (split/join semantics displayed).

Appendix E

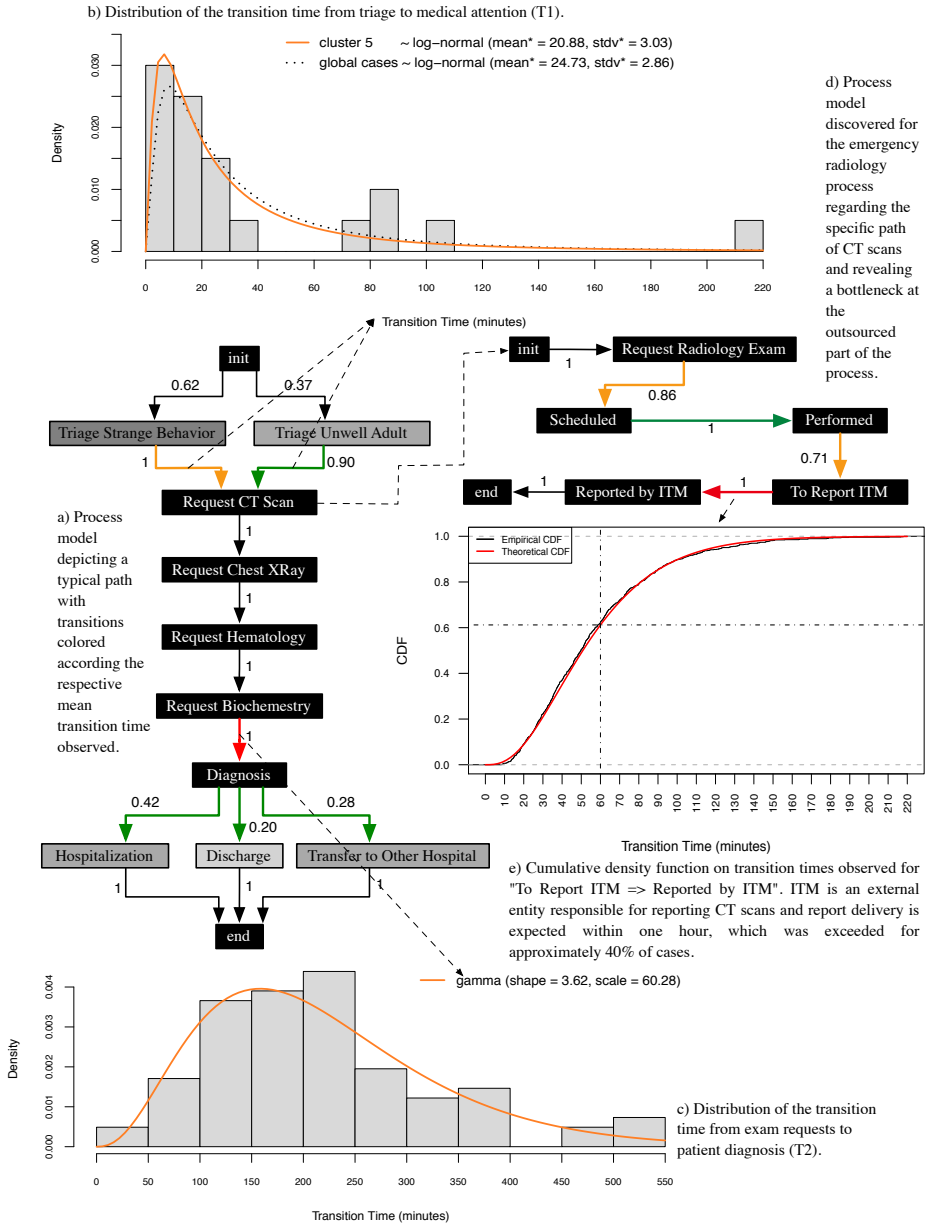

Figure A.17: Control-flow and performance models discovered for the acute stroke process and their interconnection with the workflow of CT scans observed at the emergency radiology process.

Appendix F

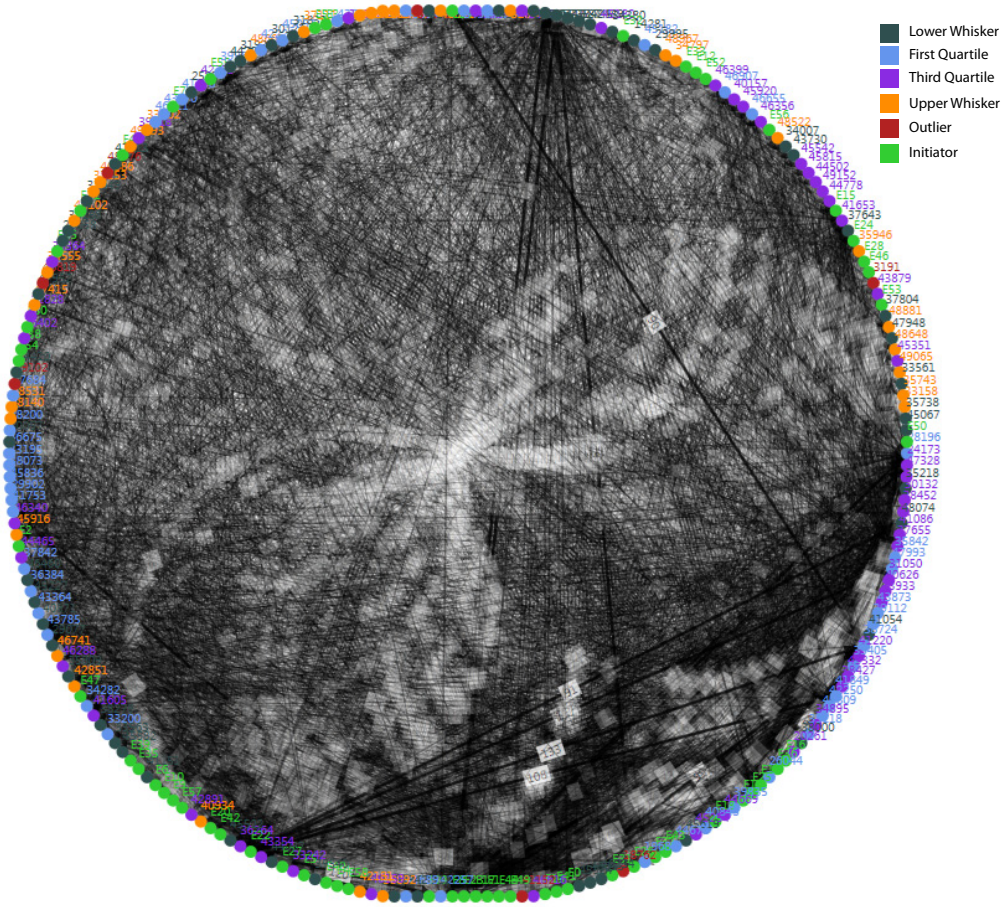

Figure A.18: Sociogram based on handover of work obtained from global emergency cases.

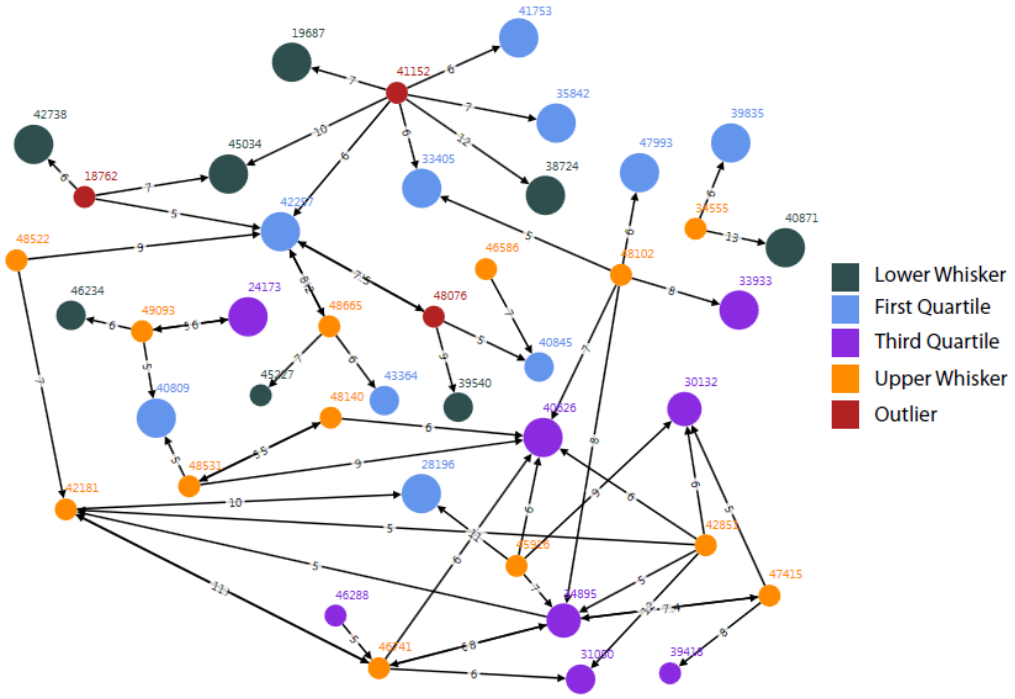

Figure A.19: Social network focusing emergency clinicians at upper whisker and outliers (according their Ψ value) and the colleagues they more often transfered patients to. Clinicians who handled less than 80 cases are not depicted, neither is depicted edges with weight, i.e. patient transfers, below 5.

Originator o_n	$INFLOW(o_n)$	$OUTFLOW(o_n)$	Ψ_{o_n}	Relative Position
18762	122	107	-0.065	Outlier
41152	154	132	-0.077	Outlier
48076	82	59	-0.163	Outlier
48665	91	58	-0.221	Upper Whisker
48102	90	53	-0.259	Upper Whisker
48531	100	40	-0.428	Upper Whisker
34555	133	46	-0.486	Upper Whisker
48522	129	44	-0.491	Upper Whisker
47415	158	53	-0.498	Upper Whisker
48140	106	33	-0.525	Upper Whisker
42851	180	55	-0.532	Upper Whisker
46586	97	29	-0.540	Upper Whisker
46741	164	49	-0.540	Upper Whisker
49093	157	46	-0.547	Upper Whisker
45916	192	55	-0.555	Upper Whisker
42181	210	56	-0.579	Upper Whisker

Figure A.20: Inflow, outflow, Ψ value and relative position of each outlier and upper whisker depicted in Figure A.19.