UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

DIEGO TORALLES AVILA

Relying on Heterogeneous Data Sources to Detect Business Process Change in Process Models

Thesis presented in partial fulfillment of the requirements for the degree of Doctor of Computer Science

Advisor: Prof. Dr. Lucineia Heloisa Thom

Porto Alegre
August 2023

CIP — CATALOGING-IN-PUBLICATION

Avila, Diego Toralles

Relying on Heterogeneous Data Sources to Detect Business Process Change in Process Models / Diego Toralles Avila. – Porto Alegre: PPGC da UFRGS, 2023.

137 f.: il.

Thesis (Ph.D.) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR–RS, 2023. Advisor: Lucineia Heloisa Thom.

1. BPM. 2. Business process change. 3. Organizational change. 4. Machine learning. I. Thom, Lucineia Heloisa. II. Título.

ACKNOWLEDGMENTS

I would like to express my deepest gratitude and appreciation to the following individuals who have played a significant role in the completion of my PhD thesis:

First and foremost, I would like to extend my heartfelt thanks to my advisor, Lucineia Heloisa Thom, for their guidance, mentorship, and unwavering support throughout this journey. Their expertise, patience, and encouragement were invaluable in shaping my research and shaping me into a better researcher.

I would also like to acknowledge all my colleagues who have been students of my advisor over these past few years. Our frequent meetings, presentations, and discussions have enriched my understanding and provided a stimulating academic environment. I am grateful for the intellectual exchange and the shared experiences we have had together. Special thanks go to Vitor Camargo de Moura, a dedicated student who collaborated with me on the development of tools and the classification of process elements. Your commitment was instrumental in our progress, and I am grateful for the fruitful collaboration we shared. Special thanks also go to Rachele Bianchi Sganderla, a colleague who generously provided us with process models to analyze. Your contribution significantly enhanced the breadth and depth of our research, and I am thankful for your willingness to share your valuable resources.

To the administrative staff of the university whom I interviewed, I express my gratitude for their availability and patience in conducting the analysis of their department's processes.

This thesis was financially supported by CAPES, which I thank for granting me the opportunity to pursue my studies.

Last but certainly not least, I want to express my heartfelt gratitude to my family for their unending love, support, and understanding. Their unwavering belief in my abilities and their encouragement have been the driving force behind my achievements. I am profoundly grateful for their sacrifices, patience, and the countless ways they have cheered me on throughout this demanding academic endeavor.

ABSTRACT

Due to changing customer needs, regulations, protocols, and technologies, an organization's business processes must regularly change and improve. The Business Process Management (BPM) discipline guides organizations to perform these changes through the BPM life-cycle, in which business processes are modeled, analyzed, redesigned, and implemented. However, sometimes these changes bypass the BPM life-cycle, happening directly at the implementations' operational level. Consequently, the respective process models need to be updated. Business process event logs can be analyzed to identify which models need updates, but not all implementations generate event logs.

One possible approach to help detect business process changes is monitoring external systems, participants, documents, and other items used or produced by a business process. These items are observable entities, which are components required for a business process execution. Monitoring change in these entities turns them into heterogeneous data sources, named as such because their data cannot easily be merged with event logs. We show that these entities can be used to create a framework for assisting in updating outdated process models, though it demands a method for identifying these entities. It also requires the mapping between entities and process models, allowing process analysts to quickly identify outdated models when the linked entities have suffered changes.

In this thesis, we assess the feasibility of creating this framework. We evaluated and compared different frameworks of organizational change, business process analysis, and redesign with an investigation of the changes required to update 25 real process models. This comparison guided us to define a taxonomy of observable entities related to business process change, which we applied to manually classify 1329 process elements originating from 88 process models. The classification frequency of the process models was 57% on average. The classification was also used to train automated classifiers using machine learning. The best automated classifiers achieved F1-scores of up to 95.4%.

Our method of semi-automated manual classification of process elements with process analysts is the primary method for identifying observable entities as required by our suggested framework. In addition, we defined a set of recommendations to help build the mapping between entities and process models and ensure it stays consistent, as well as instructions on how to use the framework to identify outdated process models.

Keywords: BPM. business process change. organizational change. machine learning.

LIST OF ABBREVIATIONS AND ACRONYMS

IT	Information Technology
BPM	Business Process Management
BPMN	Business Process Model and Notation
BPMS	Business Process Management System
OMG	Object Management Group
7PMG	Seven Process Modeling Guidelines
CDD	Concept-Drift Detection
GDPR	General Data Protection Regulation
PAIS	Process-Aware Information System
CompNB	Complement Naive-Bayes
MultNB	Multinomial Naive-Bayes
RF	Random Forest
SVC	Support Vector Classifier
BoW	Bag of Words
TF	Term-Frequency
IDF	Inverse Document Frequency
TP	True Positive
FP	False Positive
TN	True Negative
FN	False Negative
TPR	True Positive Rate
FPR	True Negative Rate
ROC	Receiver Operating Characteristic
AUC	Area Under the Curve

WST	Work System Theory
PP	Process Participants
STT	Systems, Tools, and Technologies
PDI	Processed Documents and Information
NLP	Natural Language Processing

LIST OF FIGURES

LIST OF TABLES

CONTENTS

1 INTRODUCTION

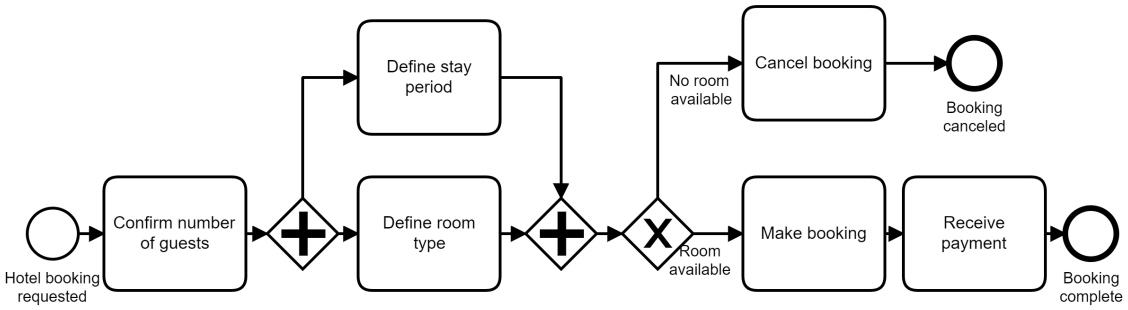
Any organization contains numerous activities that are the core of their work, called *business processes*. These business processes (also sometimes referred to as *workflows*) are a series of related tasks that produce a product or a service that fulfills a certain goal for a particular process participant or set of participants (KROGSTIE, 2016). Typical examples of business processes include selling a product to a client in person or through online purchases, paying employees, approving credit, and many others.

Given how fundamental business processes are to organizations, there is an increasing interest in their management. Business process management (BPM) is a discipline dedicated to ensuring business processes have consistent results and taking advantage of improvement opportunities, such as reducing costs, shortening execution times, or improving the overall quality of the product or service (AALST, 2013). To do this, BPM provides much-needed assistance through a selection of principles, methods, and tools to model, administrate, configure, execute, and analyze business processes (DUMAS et al., 2018).

Adopting BPM usually involves studying an organization's business processes and drawing them graphically as process models. A process model describes at some level of abstraction the domain of a business process, including the activities, the decision-making, and the events that happen during its execution (KROGSTIE, 2016). Figure 1.1 shows an example of a process model. Process models can be used as graphical representations of the business process that may be analyzed and improved until it is put into practice at the organization. In this latter step, organizations adopting BPM usually implement their business processes using process-aware information systems (PAIS) (REICHERT; WEBER, 2012), that not only can execute a business process as depicted in its process model but also provide mechanisms to help control and monitor the activities of the business processes and the actors performing them (DUMAS; AALST; HOFSTEDE, 2005). A PAIS also allows for the generation of data that records each business process execution in event logs (Van Der Aalst, 2009)

The use of BPM in an organization is a continuous life-cycle that constantly takes the information observed while monitoring the implemented business processes to recreate its process model and discover again new improvement opportunities (DUMAS et al., 2018). It is during this cycle that business processes change. Change is a process, independent from BPM, in which it is possible to empirically observe that the form, quality,

Figure 1.1 – Example of a process model for booking a hotel room.



Source: The author.

or state of something differs over time (VEN; POOLE, 1995). In the context of business processes, this difference can usually be noticed through modifications in their activities or in the sequence of their execution (Van Der Aalst, 2016). Still, a business process operation is linked to other components that can affect or be affected by changes, such as process participants, customers, information, and technologies (ALTER, 2013).

Business process change can be an improvement in the performance of some variables, such as cost or time (DUMAS et al., 2018). This way, change is deliberately sought by an organization. However, change can also be involuntary. For example, recently many organizations had to change their data privacy policies due to the General Data Protection Regulation (GDPR) that was established in the European Union (EUROPEAN COMMISSION, 2018). In order to become GDPR compliant, these organizations had to rethink and change their business processes to achieve the privacy constraints of their users' personal data (AGOSTINELLI et al., 2019).

1.1 Motivation

In practice, the continuous life-cycle of BPM sometimes suffers from poor management. For example, a clear division can be observed between the analytical phases of this cycle, when business processes are identified and studied and process models are created and improved, and the practical phases, when process models are implemented and their execution is monitored. Realizing the practical phases may require a significantly greater investment than the analytical phases, partially due to the cost of using existing process automation tools (TrustRadius Inc., 2020) and the complexity of implementing processes in a PAIS (Van Der Aalst; HOFSTEDE; WESKE, 2003; DUMAS et al., 2018). As such, the life-cycle and the business processes can be stuck in one of the analytical

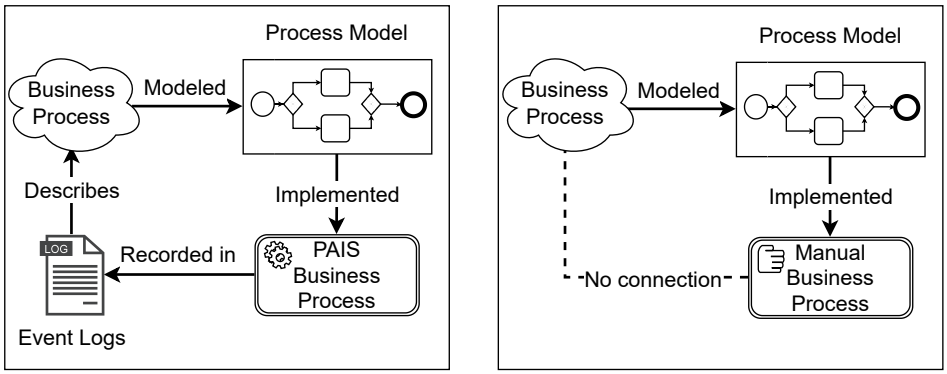
phases (SADIQ et al., 2007; CONFORT, 2010).

Even when organizations have the capacity to implement their process models, problems can still occur. The analytical phases of the BPM life-cycle can take a long time to be completed. During this time, it is unlikely that the business processes themselves and the organization's management would stop changing since they need to meet changing customers, regulations, and technology needs to remain competitive (DUMAS et al., 2018). As such, by the time the process models are ready to be implemented, there exists the possibility they may no longer be valid due to changes that happened to the business processes in execution. Essentially, there may be a discrepancy between a process model and the practical execution of its business process. Attempting to implement the process models regardless of these discrepancies may not be feasible since the organization may no longer have the infrastructure resources necessary for executing the business process as depicted in the process model (IHDE et al., 2019; BIAZUS et al., 2019).

The difficulty of realizing the implementation of process models thus creates a disconnect between the conceptual knowledge of a business process (represented by the process model) and the actual execution of this business process in reality. This disconnect makes maintaining process models updated laborious. Figure 1.2 allows us to better understand this disconnect. Figure 1.2a shows the ideal circumstances, in which a process model is implemented in an organization using a PAIS. The PAIS then records the execution of this business process in event logs, which can be analyzed through process mining (Van Der Aalst, 2016) to measure the performance of the business process and detect deviations from the expected execution. With the data from these logs, it is possible to update our understanding of the business process, which then allows us to transfer any unexpected changes to the process model, ensuring it is up-to-date.

However, the difficulties of implementing process models may lead organizations to not use PAIS to execute their business processes. Figure 1.2b shows an example in which the business process is implemented manually, that is, when there is no information system assisting in the management and execution of the business process' activities. As a consequence, event logs are not generated, thus creating the disconnect between the business process in abstract, i.e. how one thinks the business process is being executed, and its actual execution. Without a less laborious way of updating our understanding of the business processes, the analytical phases of the BPM life-cycle have to rely on more onerous techniques to discover how they work every time the process models need to be updated.

Figure 1.2 – Differences of business process implementation.
(a) Business process implemented through PAIS. (b) Business process implemented manually.



Source: The author.

Regardless of circumstances, it is indispensable that business process changes must be reflected in the process models created during the life-cycle's analytical phases. Otherwise, these process models will no longer be valid and may cause problems if used for analysis, learning, and implementation (IHDE et al., 2019; BIAZUS et al., 2019). However, the lack of data that would be generated by a PAIS implementation of the business processes makes it necessary to look for alternative methods to detect when their execution has changed. Particularly, we believe these detection methods require identifying possible alternative data sources related to the business process execution unsupported by a PAIS, because through monitoring them it is possible to observe when a change occurs. These data sources may not register their data the same way a PAIS would, making it difficult to merge their data into an event log that can be mined with traditional process mining techniques, such as conformance checking (AALST; ADRIANSYAH; DONGEN, 2012; ADRIANSYAH; DONGEN; AALST, 2011b; POLYVYANYY et al., 2016; GARCIA-BANUELOS et al., 2018) or concept drift detection (BOSE, 2012; OSTOVAR et al., 2016; ZHENG; WEN; WANG, 2017; TAVARES et al., 2019). Possible examples of these sources are the other components that are required for, used by, or produced by a business process execution, such as process participants, customers, information, and technologies (ALTER, 2013). When the data produced by these components are being monitored, we call them *heterogeneous data sources*, due to the data produced one type of component may have a dissimilar structure to event logs and to other component types.

To identify heterogeneous data sources, we consider it important to understand how and when business process change happens and how this affects their respective business process models, since this understanding may provide clues of what are the most

probable points of change in a business process and how they may be linked to different data sources. To achieve this understanding, research is required to observe how business process change happens in practice, to discover how existing theories describe the processes of change in an organization, and to relate these existing theories to the business process changes observed in practice. With this relation between practical and theoretical perspectives on business process change, we believe we can define a framework that supports the detection of business process changes and the identification of which process models are affected by these changes. This definition requires us to classify heterogeneous data sources and utilize this classification to determine how they can be monitored and how to link this monitoring to the corresponding process models.

1.2 Hypotheses and Objectives

Based on our motivations, we established four hypotheses:

H_1 It is possible to analyze how business processes change by updating outdated process models and comparing the old and new versions.

H_2 It is possible to evaluate business process change through the perspective of frameworks on business process redesign and theories on organizational change.

H_3 An analysis of a process model can identify possible heterogeneous data sources that may help in observing changes in the business process behavior, such as external services, documents, participants, and others.

H_4 It is possible to identify which process models are affected by changes happening to a heterogeneous data source.

We have established seven objectives to help us verify these hypotheses. We first (1) performed a survey with BPM practitioners to verify how process models are used in their organizations and how they perceive change. This way, we would empirically show the existing problem of the conformance deviation between process models and the business processes in execution. Following that, we (2) analyzed how existing real-world process models have diverged from their business processes through an interview of their participants. Objective (2) concerns hypothesis H_1.

The results of these analyses were used to (3) discover categories for how the analyzed process models have changed and (4) compare the discovered changes with existing theories of organizational change. Objectives (3) and (4) are related to hypothesis

H_2. We used the knowledge acquired from these analyses to guide us on how to develop the framework to detect business process change in process models.

The development of this framework refers to hypotheses H_3 and H_4. Concerning H_3, we (5) defined a taxonomy of heterogeneous data sources that are linked to a business process execution and (6) demonstrated a method to identify heterogeneous data sources from analyzing process models. In applying the method created by objective (6) in real process models, we show not only the viability of the identification of the data sources, but also how much of any process model we can reasonably expect to monitor using these data sources.

Finally, for hypothesis H_4 we (7) defined a method for creating a mapping between heterogeneous data sources and process models. The mapping is essential for a process analyst to monitor changes in heterogeneous data sources and subsequently locate which process models require updates. The definition of methods for monitoring the heterogeneous data sources is not within the scope of this work since these methods rely on the organizations' context and require knowing the type and structure of the specific data source.

By completing these objectives, we will make three main contributions: firstly, the definition of the taxonomy of heterogeneous data sources, created based on our investigations on how business process change in practice. For every taxonomy class, we provide clear examples of how they are associated with business process change; secondly, the method of identifying heterogeneous data sources in process models, which we created to apply and validate the taxonomy with real process models. We provide multiple open-source tools to help classify process elements based on the taxonomy classes, including an automated classifier using machine learning; thirdly, the definition of the mapping between heterogeneous data sources and process models. This mapping is one of the fundamental components of building a framework to detect business process change by monitoring heterogeneous data sources.

1.3 Text Organization

This thesis is organized as follows: Chapter 2 shows a background about business process management and machine learning; Chapter 3 discusses related works on business process change, focusing mostly on process mining techniques; Chapter 4 presents our survey on the current state of process model maintenance in the industry, which is re-

lated to objective (1), as well as research on frameworks related to organizational change and business process redesign; Chapter 5 exhibits our investigation of business process change, which is related to objectives (2) to (4); Chapter 6 we accomplish objectives (5) and (6) by defining a taxonomy of observable entities that will serve as data sources for updating process models. This chapter also presents the definition of a method of identifying observable entities in process models, and the creation of a dataset of classified process models elements according to the taxonomy; In Chapter 7 we present our use of the created dataset to train machine learning algorithms to automatically classify process model elements according to the defined taxonomy; Chapter 8 describes how to create the mapping between observable entities and process models, in order to build the framework necessary for updating process models when changes are detected through monitoring the entities, thus fulfilling objective (7); Chapter 9 presents a final summary of this thesis.

2 FUNDAMENTALS OF BUSINESS PROCESS MANAGEMENT AND MACHINE LEARNING

This chapter presents fundamental concepts about BPM that are relevant in the context of this work, particularly to understand how BPM practitioners usually work to analyze and improve business processes. As such, this chapter focuses on showing the basics of the BPM life-cycle and the Business Process Model and Notation (BPMN) (OMG, 2011), as well as some of the common methods and tools used by practitioners that are relevant in the context of business process change. We also present concepts regarding machine learning, including supervised machine learning algorithms, feature extraction methods, and text processing techniques.

2.1 Business Process Management

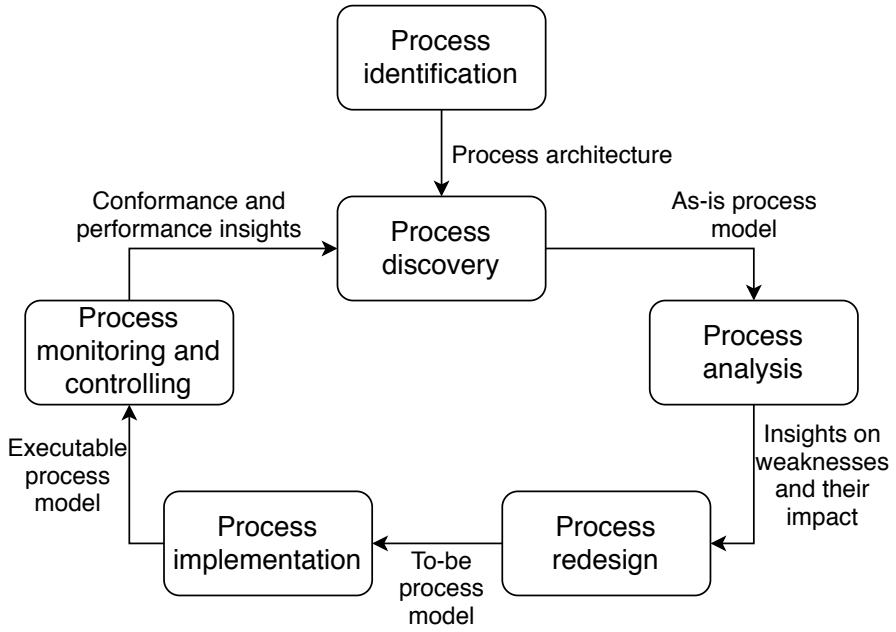
BPM is a discipline that oversees the work performed in an organization to ensure consistent outcomes and to take advantage of improvement opportunities (DUMAS et al., 2018). It contains a selection of principles, methods and tools that can be used to turn business processes more effective, more efficient, and more adaptable, which, in turn, improves productivity and reduces costs (AALST, 2013).

To explain how an organization normally manages a business process using BPM, including when this business process goes through changes, we first show in this section the BPM life-cycle, which explains the different stages through which a business process is managed. We then present which business process stakeholders exist, that is, the roles within an organization that are involved with the management and execution of business processes (DUMAS et al., 2018). Following that, we introduce a few relevant topics to business process change, such as the BPMN (OMG, 2011), the practical implementations of business processes using PAIS, and the mining of process event data with Process Mining (Van Der Aalst, 2016).

2.1.1 BPM life-cycle

The BPM life-cycle is a series of phases organized in a cyclical structure, with logical dependencies between the end of one phase and the beginning of the next. In the

Figure 2.1 – BPM Life-cycle



Source: Dumas et al. (2018))

literature, there are multiple definitions of the BPM life-cycle, though the definition we reference in this work is the one proposed by Dumas et al. (2018), which stands out for being one of the most detailed definitions. It contains six phases, as seen in Figure 2.1.

Process Identification is the first phase, in which an organization's business processes are identified and an architecture of processes is created, detailing an overall view of the existing business processes and how they relate to each other.

Process Discovery is the second phase, when a business process is studied to understand it in detail, discover its process elements and create an *as-is* process model. An as-is process model reflects the current state of the business process, before any changes are made. To create this process model, it is typically discovered what activities and events occur during the execution of the business process. It is also discovered how these elements relate to each other, that is, in which order they occur, if they occur in parallel, if they are exclusive to one another and why, or if they happen repeatedly within a loop. This discovery of process elements can be done through interviews with the business process participants, through the analysis of evidence such as descriptive documents, or through an observation of the business process execution. These elements are depicted in the process model using a modeling language, such as the BPMN (OMG, 2011). We describe the features of this language in more detail in section 2.1.3.

Regarding the analysis descriptive documents, there is a considerable amount of work in the literature dedicated to using NLP (JURAFSKY; MARTIN, 2009) to identify

which sentences of these documents contain important process information (ROSA et al., 2022; FERREIRA et al., 2018; SILVA et al., 2018), to improve them (SILVA et al., 2019), and to consolidate process information spread across multiple documents (BOHNENBERGER; SCHMITT; THOM, 2021). It is also possible to automatically model processes by mining process descriptions (FRIEDRICH; MENDLING; PUHLMANN, 2011; CAPORALE, 2016). However, the results are usually only a rough approximation of the actual business process (AA et al., 2018), meaning the other process discovery methods are still necessary.

Process Analysis is the third phase, when the business process and its process model are analyzed both quantitatively and qualitatively. In a quantitative analysis, performance measures, such as cost, waiting time, or total cycle time, are used to identify possible inefficiencies or obstacles in a business process execution and estimate how much better this execution would be if these problems were solved. Qualitative analysis evaluates non-numerical data to identify the problems of a business process.

Process Redesign is the fourth phase, when the *as-is* process model is changed to accomplish some goal, such as solving the problems identified in the process analysis phase. Multiple redesigns may be proposed in this stage, thus an analysis of these proposals is required to identify and choose those that are the most viable. There is a variety of methods and techniques for the elaboration of redesigns. The final product of a redesign phase is a *to-be* process model that describes the future business process that is expected to accomplish the goals established in this phase.

Process Implementation is the fifth phase, when the changes defined in the *to-be* process model are put into practice. How this implementation is performed depends on what has been changed in the *to-be* process model, but generally there are two main aspects: the first aspect is the change in the organization's structure, that is, the participants that are involved in the execution of the business process. Each process participant shown in the *to-be* process model may have new or different responsibilities that they must be capable of performing in this new implementation; the second aspect involves changing the execution of the business process. This includes changing existing systems and implementing new ones to support the *to-be* process model.

Process Monitoring and Control is the sixth and final phase that happens after a business process is implemented. During the execution of the business process data must be collected to measure its performance. With time, new issues may be found in the execution, such as errors or deviations from the intended process behavior. It is with the

data collected that the BPM life-cycle restarts from the process discovery phase, allowing for the *as-is* process model to be updated to correct these possible deviations.

2.1.2 Stakeholders

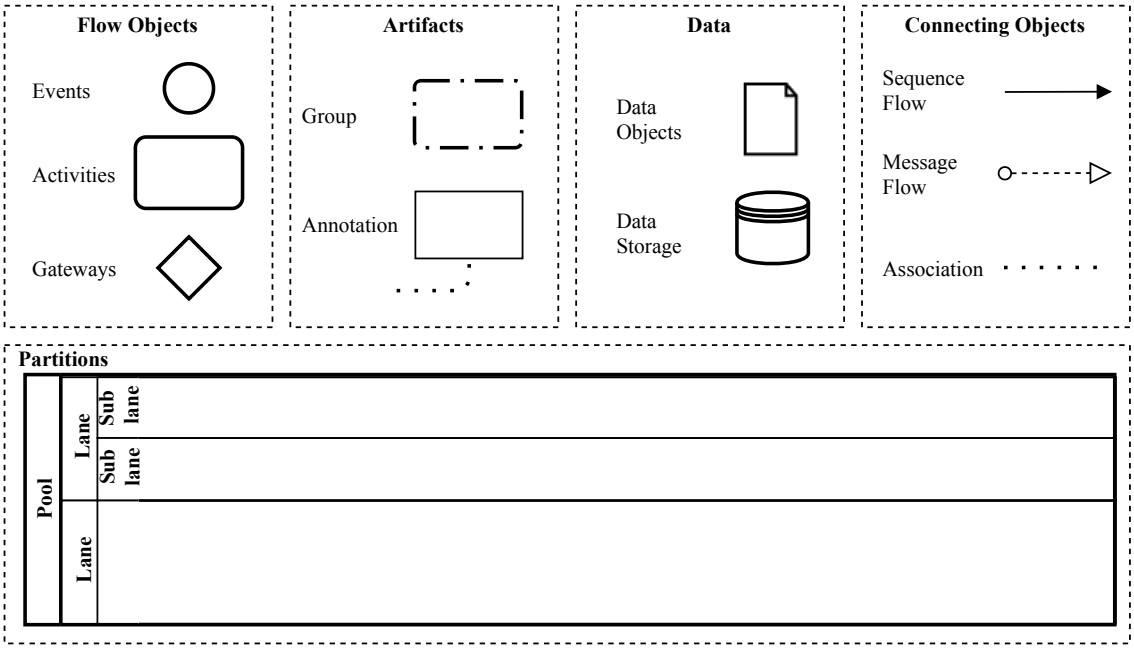
There are different *stakeholders* involved in a business process during its life-cycle (DUMAS et al., 2018). Knowing who they are is important to understand their responsibilities and how they can influence the work of a business process. Among the stakeholders, the following groups and individuals can be identified:

- The *Management Team* is responsible for supervising the processes, starting the process redesign initiatives, and providing resources and guidance to all *stakeholders* involved in all phases of the BPM life-cycle.
- The *Process Owners* are responsible for the efficient and effective operation of a given process. That is, they are responsible for planning, organizing, monitoring, and controlling the execution of the process.
- The *Process Participants* are the human actors who perform the activities of a process.
- The *Process Analysts* conduct identification, discovery, analysis, and process redesign activities.
- The *System Engineers* are responsible for capturing the requirements defined by the process analysts and implementing, testing and deploying a system that meets these requirements.
- *BPM Group* is responsible for preserving and maintaining the knowledge and documentation of completed BPM projects.

2.1.3 Business Process Model and Notation

BPMN is a graphical representation for modeling business processes that is maintained by the *Object Management Group* (OMG), with its 2.0 version being released in 2011. Since then, BPMN has been rising in popularity, with several modeling tools sup-

Figure 2.2 – The basic elements of BPMN



Source: OMG (2011)

porting it, such as the Signavio [1], Bizagi[2] and Camunda [3]. In 2013, BPMN was defined as an ISO standard (ISO, 2013). The main objective of BPMN is to provide a user-friendly notation for all stakeholders, including the process analysts who create the initial drafts of the processes, the technical developers who are responsible for implementing the technology that will execute these processes, and the people who will administer and monitor the processes (OMG, 2011). There are five main categories of process elements in BPMN, as seen in Figure 2.2: Flow Objects, Data Objects, Connecting Objects, Partitions, and Artifacts.

Flow objects are the main elements of any BPMN process model. They define what the business process does through three basic element types: *Events* represent something that happens instantaneously within a business process, such as the moment a business process starts when a new client request is received, or when this same business process ends because the service or product requested is complete and delivered; *Activities* are elements that describe the actions performed during the execution of the business process. They can be atomic (referred to as *tasks* or composite (also known as *sub-processes*; *Gateways* control the splitting and merging of the execution flow of activities and events of a business process. They can, for example, make specific execution flows happen in a loop, in parallel or when certain conditions are met.

[1] www.signavio.com
[2] www.bizagi.com
[3] www.camunda.com

Both activities and events have multiple sub-types that further specialize the semantics of these elements. Of particular importance to this thesis are the sub-types that represent some form of interaction between different organizations, resources or services, such as message events, message activities, user tasks, or service activities, because there may be a potential heterogeneous data source in this interaction. Gateways also have types that define how the execution flow is split and merged. The main gateway types are:

- *Exclusive (XOR) gateways* define the beginning or end of a split in the process flow. For example, a XOR-gateway may split into multiple outputs. These outputs are mutually exclusive, that is, only one path can be taken. Therefore, each output branch must have a condition to define which branch is taken. On the other hand, a XOR-gateway may also join multiple inputs. In this case, it is only necessary for the flow of only one input branch to end to activate the gateway's output.
- *Parallel (AND) gateways* fork and merge the process flow between all connected inputs and outputs, allowing for the process flow to be executed in parallel.
- *Inclusive (OR) gateways*, similarly to exclusive (XOR) gateways, split the process flow, but in this case, the outputs and inputs are not mutually exclusive. Because of this, an OR-split may cause multiple output flows to become active, while an OR-join requires that all currently active input flows end before the output is activated.

Connecting objects link flow objects, data objects, and artifacts together. There are three types of connecting objects: *Sequence flows* link flow objects, defining the order of the execution flow of a business process; *Message flows* represent the trading of messages between different organizations through a link between message events or activities. They can be e-mails, fax messages, phone calls, or even the delivery of letters or packages; *Associations* link flow objects to data objects or artifacts. In the case of data objects, the association shows that the linked flow objects uses the data object as an important resource for its execution.

Data objects are process elements that show a perspective of the data of a business process. Compared to flow objects, which have a functional perspective of a business process, data objects show the flow of information between activities and events. They commonly represent documents or files that are read andor written by flow objects and, as such, are required for their execution. Data objects may be stored in *data repositories*, so that they continue to exist after a specific instance of a business process. As such, data repositories are important resources that preserve evidence of the execution of a business

process.

Partitions are process elements that show another perspective on the resources of a business process. In this case, partitions show the process participants and the organizations in which they belong. Thus, there are two types of partitions: *pools*, representing the organizations; and *lanes*, which subdivide pools to represent the participants of an organization. Communication between pools of a process model has to be shown explicitly and performed through the use of message flows connecting message events or activities. On the other hand, communication between lanes of a pool cannot be done through message flows. As such, this communication is often implicit between the information flow defined by data objects or managed by whatever system that implements the execution flow of a business process.

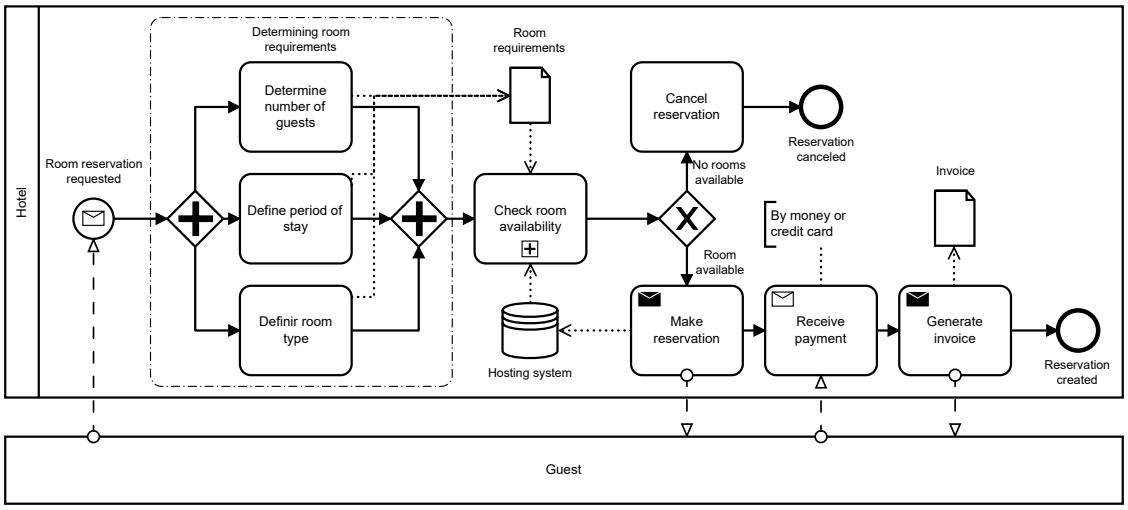
Artifacts are the last type of process element. They do not represent any sort of object or behavior of the business process. Instead, they present additional information regarding some element of the process model through the use of *annotations*, which usually contain informative texts, and *groups*, which organize different activities together but do not affect their execution.

Figure 2.3 shows a process model example that contains process elements of all these categories and shows how they can be used. This example presents a business process for *room reservation in a hotel*. The business process in this process model starts when a *guest* communicates with the *hotel*, requesting the reservation of a room. Once this request is made, the hotel has to determine the *room requirements*, including the number of guests, the period of stay, and the room type. These requirements can be determined in any order. Afterwards, the hotel checks room availabity in the *hosting system* based on the determined requirements. If no rooms are available, the reservation is canceled and the process ends. If there is a room available, the hotel makes the reservation in the hosting system and informs the guest. The hotel awaits payments from the guest by either money or credit card. After payment, they generate an *invoice*, which is sent to the guest. The process then ends with the reservation created.

The example in Figure 2.3 shows how a potential *guest* communicates with the *hotel*, as well as what activities are realized once a room is requested. It also shows how these activities interact with other relevant objects, such as the *hosting system*, the *room requirements*, and the *invoice*.

While process models are composed of many individual process elements, it is sometimes more useful to examine arrangements of interconnected process elements. In

Figure 2.3 – Example process model with some of the basic elements of BPMN



Source: The author.

this work, we call these arrangements *process model fragments*. Many of our analyses on business process change affect fragments since change can happen both to process elements and the connections between them.

2.1.4 Process Modeling Guidelines

Process modeling is a fundamental aspect of the process discovery phase of the BPM Life-cycle. This task is widely accepted to be challenging (MENDLING; REIJERS; AALST, 2010), given that it involves understanding the modeling notation, its many different elements, and their respective semantics (LEOPOLD; MENDLING; GÜNTHER, 2016). Therefore, the quality of the resulting process model often depends on the expertise of the process modeler (FIGL, 2017; NELSON et al., 2012). In spite of the efforts of even experienced process modelers, process models frequently have modeling issues, such as control flow errors, badly designed structures, layouts, and incorrect labeling (MENDLING; STREMBECK, 2008; LEOPOLD; MENDLING; GÜNTHER, 2016). These issues significantly impair the quality of process models, especially their comprehensibility, which causes them to become less useful for the other phases of the BPM life-cycle (WESENBERG, 2011). Thus, it is very important that the process modeling task results in high-quality process models (REIJERS; MENDLING; RECKER, 2015).

Many solutions have been proposed in the literature to ensure the quality of process models. Process modeling guidelines (LEOPOLD; MENDLING; GÜNTHER, 2016; GSCHWIND et al., 2014; MENDLING, 2013; SÁNCHEZ-GONZÁLEZ et al., 2017;

Table 2.1 – The Seven Process Modeling Guidelines (7PMG)

	Guideline
G1	Use as few elements in the model as possible
G2	Minimize the routing paths per element
G3	Use one start and one end event
G4	Model as structured as possible
G5	Avoid OR routing elements
G6	Use verb-object activity labels
G7	Decompose a model with more than 50 elements

Source: Mendling, Reijers and Aalst (2010)

KOSCHMIDER; FIGL; SCHOKNECHT, 2016) are one of these solutions. These guidelines define simple rules for process analysts to follow. Their goal is to increase the comprehensibility and comparability of process models in order to facilitate efficient model analysis (MENDLING; REIJERS; AALST, 2010). Beginner process modelers strongly benefit from these guidelines, as they do not yet have the teaching or the experience necessary to create high-quality process models. However, more experienced modelers can also use the guidelines both proactively, to enhance the process modeling task, and retroactively, to find if the process model has any modeling issues. Furthermore, using process modeling guidelines within a group context, with multiple process modelers, can help ensure the consistency and integrity of process models (DUMAS et al., 2018).

The "Seven Process Modeling Guidelines (7PMG)" by Mendling, Reijers and Aalst (2010) are a widely cited example in the literature. These guidelines, as listed in table 2.1, were some of the first that were proposed based on a strong empirical foundation, while also trying to keep the instructions simple and related to concrete actions that process modelers execute during the process modeling task. However, there are many more guidelines dispersed across the literature with different types of recommendations regarding a process model's size, topology, decomposition, layout, and nomenclature. A systematic literature review by Avila et al. (2020) has compiled many of these guidelines and evaluated their empirical foundation.

2.1.5 Process Analysis and Redesign

After a business process is modeled, an analysis of the process model and the available data is usually done to identify problems and improvement opportunities. Process analysis in BPM can be done both in a qualitative and a quantitative way Dumas et al. (2018). Qualitative analysis uses techniques to identify why a particular fragment of

the process model is not performing satisfactorily. For example, “Value-added” analysis evaluates which tasks of a business process contribute to its goals or its execution, thus adding value, while “Root Cause” analysis techniques help identify the root cause of a specific problem. A quantitative analysis makes use of performance measures, such as time, cost, quality, or flexibility, to find and estimate where and how much a business process can be improved.

Process redesign takes what was found during process analysis and uses it to guide the creation of a new and improved process model. This version is usually called the to-be process model, that is, the version that is to be implemented and executed (DUMAS et al., 2018). Since the aim of process redesign is to make the business process better, it is the primary mode through which practitioners of BPM can make changes to process models so that they can be implemented to their business processes. One notable type of change is process automation, in which a business process is automated through a series of process model changes. These changes transform a *as-is* process model, which may have been generally composed of manual activities, into a executable process model that an information system can perform.

It is primarily through these two phases that business processes are changed in the context of the BPM life-cycle. However, as we have discussed in Section 1.1 and as we will show in Chapter 4 and Section 5.3, there are situations when business processes change in ways that differ from the methodology of process redesign. Changes that are applied directly at a business process implementation need the update of the respective process models, but these changes are difficult to track when these implementations are not supported by a PAIS.

2.2 Process-Aware Information Systems

Business processes of an organization may be implemented and executed in a variety of ways. Before BPM is introduced, a business process may be executed manually by a company’s employees, or use one of its pre-existing systems that gives support to some difficult or demanding task. After BPM is introduced, and the business processes are discovered and analyzed according to the BPM life-cycle, one of the best ways to take advantage of the methods and techniques of BPM is to implement the process models created in the BPM life-cycle using a PAIS. A PAIS is an information system that supports process automation Reichert and Weber (2012). It manages and executes business

processes based on its process models Dumas, Aalst and Hofstede (2005). A PAIS differentiates itself from other information systems by being process-aware, that is, by using the explicit representation of a business process present in a process model to dictate its execution. This can lead to better communication, management, and performance of the business processes Van Der Aalst (2009).

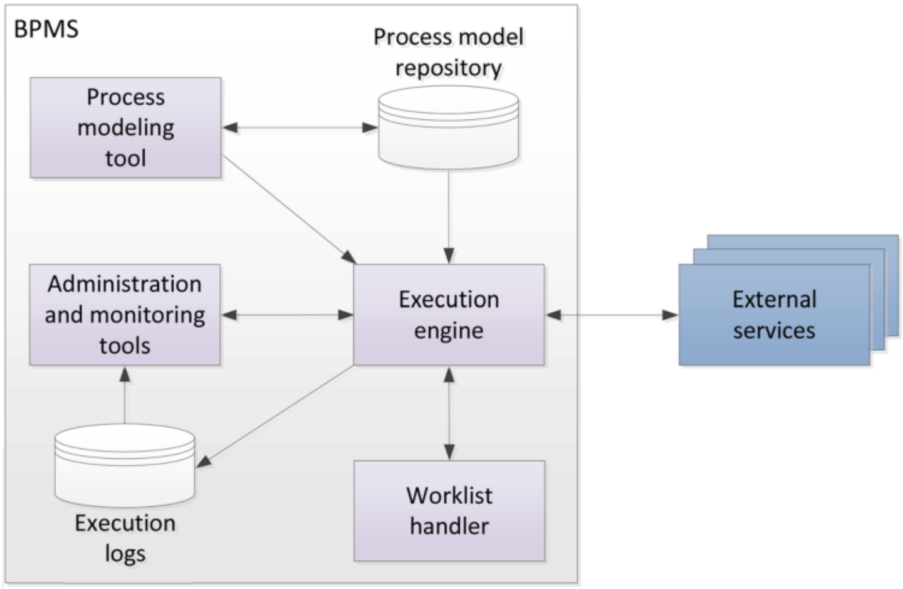
One particularly important type of PAIS is a Business Process Management System (BPMS). A BPMS is a domain-agnostic PAIS, i.e., a system that allows for the automation of business processes of any domain (DUMAS et al., 2018). Vendors offer many different BPMSs that support the design, analysis, execution, and monitoring of business processes based on their process models. The automation of these process models ensures that their activities are realized in the correct order and by the right resource or process participant. As such, a BPMS provides the same advantages a PAIS provides, namely, better performance, management, and communication, in addition to the transparency of execution and the flexibility of integration with other systems.

The architecture of a common BPMS includes the following components, as seen in Figure 2.4: the *process modeling tool*, which supports the design and creation of process models within the BPMS; the *process model repository*, which stores the process models created by the process modeling tool; the *execution engine*, which has the abilities to create instances of the process model stored in the repository and to distribute work to proper process participants based on the model's structure; the *worklist handler*, which displays the work that is pending to the process participants; the *execution log*, which stores a record of what has happened in each process instance; the *administration and monitoring tools*, which can manage the operation of the BPMS; and the *external services*, which provide important services not available to the BPMS such as access to databases or e-mails.

2.3 Concepts of Process Mining

Executing business processes in a PAIS allows organizations to generate execution logs of these business processes. These execution logs, also known as *event logs*, contain recordings of events that happened in the PAIS. These events describe, for example, what activities, events, and decisions occurred, step by step, from beginning to end of each instance of a business process (Van Der Aalst, 2016). To complement each event, other information is also stored, such as the process data at that time, who was responsible for

Figure 2.4 – The architecture of a BPMS



Source: Dumas et al. (2018)

executing it and when it was executed. One sequence of events produced by one execution instance of a business process is called a *trace* (GARCIA-BANUELOS et al., 2018). It is possible to use these event logs to enhance the management of business processes with process mining.

Process mining is a method where the awareness of how business processes perform, as presented by BPM, is used to interpret the data provided by event logs to extract knowledge about specific business processes (Van Der Aalst, 2016). For example, one of the techniques possible through process mining is the automatic discovery of business processes (FAHLAND; Van Der Aalst, 2015). This technique uses the traces of an event log to identify the timeline of events for each business process instance. By analyzing a variety of these traces, an algorithm tries to determine the best process model that, when executed, could replicate the majority of the timelines. The criteria that define the quality of this process model can be, for example, the average of how many events are replicated by it, its precision, how simple it is for someone to understand, or how over- or under-fitted it is when compared to the event log (Van Der Aalst, 2016).

In addition to automatic discovery, process mining also has two other techniques, performance analysis, and conformance checking (Van Der Aalst et al., 2012). Performance analysis uses two inputs, a process model and an event log, with the goal of assisting in evaluating the performance of certain aspects of the business process, such as time, resources, or quality (MILANI; MAGGI, 2018). Performance analysis with event logs allows for business processes to be evaluated in great detail given the small granularity of

the data provided by the event logs.

Conformance checking also uses a process model and an event log to verify if the former conforms to the latter and vice versa. Conformance checking may help process analysts identify discrepancies from expected behavior in the execution of a business process. These discrepancies may be caused by errors that have to be corrected. Alternatively, they may be caused by changes to the current behavior of the business process that are not yet present in the process model. The two types of discrepancies conformance checking identifies are *unfitting log behavior*, when the events present in the event log are not allowed by the process model, and *additional model behavior*, when there are possible behaviors that are allowed by the process model but are never observed in the event log (GARCIA-BANUELOS et al., 2018). Of the existing process mining techniques to perform conformance checking, the simplest are *replay techniques* (ROZINAT; AALST, 2008; BROUCKE et al., 2014a; MUNOZ-GAMA; CARMONA; AALST, 2014), which attempt to parse the traces of an event log through a process model, to determine the traces, and more specifically the events, that cannot be replayed. Other conformance checking techniques include *trace alignment* (ADRIANSYAH; DONGEN; AALST, 2011b; MANNHARDT et al., 2015), *behavioral alignment* (GARCIA-BANUELOS et al., 2018), *negative events* (WEERDT et al., 2011; BROUCKE et al., 2014b), and *prefix automata* (MUÑOZ-GAMA; CARMONA, 2010).

Conformance checking is noteworthy in this work because it can detect business process changes in cases where event logs are available. It can also identify in which process model fragments these changes happened. Our proposed means of detecting business process change has similarities in concept to conformance checking, since it substitutes the data source that is event logs with heterogeneous data sources from external components linked to the business process. The monitoring of heterogeneous data sources should be aware of the process model's structure to be able to verify and compare it to the business process' behavior. The challenge is identifying what heterogeneous data sources exist in a process model and determining how this verification would occur.

2.4 Machine Learning

In this section, we introduce concepts of machine learning and a few relevant techniques and algorithms. Everything presented in this section was used in the development and evaluation of the automated classifiers described in Chapter 7.

Machine learning is a subfield of artificial intelligence that is widely applied by many modern technologies and industries, such as image and speech recognition, natural language processing, recommendation systems, financial analysis, healthcare, manufacturing, and others (FACELI et al., 2021). This field is motivated by the increasing amounts of data generated ever faster with the rise of the internet and other digital technologies and the increasing complexity of the problems that could be solved by analyzing this data. With machine learning, we are able to create "learning" methods (MITCHELL et al., 1997) capable of processing large amounts of sample data to acquire “knowledge” (FACELI et al., 2021; ALPAYDIN, 2010). With this sample data, usually referred to as a *training data*, these methods can identify patterns, tendencies, or groupings to create a model (NELLI, 2015). This model is then used to analyze new data entries to make decisions or predictions based on the “knowledge” it learned.

Generally, the training data for machine learning is composed of a table, in which every row is an instance or an item, and every column is an attribute. When training a machine learning model, these attributes are divided into two types based on the desired function of the trained model (LENZ et al., 2020). *Predictive attributes* define characteristics of the items in the training data, containing information considered input data for the training process. On the other hand, *target attributes* represents the desired output of the trained model, which frequently is composed of a class or a numeric value.

There are two classically defined types of machine learning. *Supervised learning* algorithms are applied to perform *predictive tasks*, in which the goal is to predict items' target attributes based on their predictive attributes. The algorithms are trained with labeled data containing every item's predictive and target attributes. *Unsupervised learning* algorithms perform *descriptive tasks*, which explore the training data to identify patterns. Unsupervised learning is called such because the training data does not contain target attributes. Example applications of unsupervised learning are the grouping of items based on the similarity of their attributes or the discovery of interesting relationships between attributes.

2.4.1 Supervised Learning Algorithms

One of the most basic applications of supervised learning algorithms is determining whether an item belongs to a specific class. This type of application is known as a binary classification, in which the algorithms learn from a training dataset containing

positive examples (items that are within the class) and *negative examples* (ALPAYDIN, 2010). After training, the resulting machine learning model attempts to predict if a new item is a positive or negative example of that class. One example of a supervised learning algorithm is the Naive-Bayes algorithm (FACELI et al., 2021), a probabilistic classifier based on the Bayes theorem:

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \tag{2.1}$$

where:

- $P(A)$ and $P(B)$ are the prior probability of events A and B, independent of each other.
- $P(A|B)$ is the posterior probability of event A given that event B has occurred.
- $P(B|A)$ is the likelihood of event B given that event A has occurred.

The Naive-Bayes algorithm is commonly used in text classification tasks, such as spam detection, sentiment analysis, and topic classification. The algorithm works by first calculating the prior probabilities ($P(A)$ and $P(B)$) of class A and event B based on the training data, e.g., how many items are in A for every possible label of A, and the same with B. Then, it calculates the likelihood of event B given the class ($P(B|A)$), e.g., the probability of observing a specific attribute of B in documents that are positive examples of class A. Finally, it combines the prior probabilities and the likelihood to calculate the posterior probability of class A, given the observed event B in a document ($P(A|B)$).

In chapter 7, we performed a binary classification task using four supervised learning algorithms. They are:

- *Multinomial Naive-Bayes* (MultNB) is a variant of the Naive-Bayes algorithm for multinomially distributed data. It is a variant suitable for text classification tasks where the features (i.e., words) have a count-based representation. Using a multinomial distribution, it models the likelihood of each word occurring in each class (PEDREGOSA et al., 2011).
- The *Complement Naive-Bayes* (CompNB) is another variant of the Naive-Bayes algorithm suitable for textual data. It uses the complement of each class's probability when calculating the likelihoods. It is well-suited for imbalanced datasets and it often outperforms MultNB on text classification tasks (PEDREGOSA et al., 2011).
- *Random forest* (RF) is an ensemble learning method that generates a multitude of

decision trees and combines their outputs to make a final prediction. Each tree is constructed using a random subset of the training data and a random subset of the features (FACELI et al., 2021; PEDREGOSA et al., 2011).

- *Support vector classification* (SVC) is a linear classification algorithm that finds the hyperplane or set of hyperplanes that best separates the classes. It works by maximizing the margin between the hyperplane and the closest data points of each class. It can also be extended to handle non-linear decision boundaries (FACELI et al., 2021; NETTO, 2021; PEDREGOSA et al., 2011).

One particular worry of these algorithms is the balance of how many items the target class has for each of its possible labels. An imbalanced training data can cause the algorithms to underperform. The most common solutions to this problem are oversampling or undersampling the items until the training data is balanced. In oversampling, the objective is to increase the number of items in the minority class. One method for doing so is duplicating existing items. Undersampling is the reverse, in which items in the majority class are removed.

2.4.2 Feature Extraction and Text Processing Techniques

When applying machine learning algorithms to textual data, a few considerations have to be made regarding the quality of the training data and how the texts are going to be analyzed. For the sake of clarity, the textual data is usually referred to in the literature as *documents*, while the set of all documents is referred to as the *corpus* (JURAFSKY; MARTIN, 2009). Generally, one of the first steps in the analysis of the corpus is tokenization, in which documents are segmented into tokens representing a single word each. It is frequently recommended that these tokens be preprocessed to prepare them for training algorithms (FACELI et al., 2021). The classifications tasks in Chapter 7 utilize these preprocessing techniques:

- *Converting all tokens to lower-case*: This reduces the dimensionality of the data, allowing all identical words to be considered the same independently of their capitalization.
- *Removal of stop-words*: Stop-words are common words frequently used in language, but do not carry much meaning for text analysis, such as prepositions, conjunctions, and articles.

Table 2.2 – Example of a *Bag of Words* matrix

Document	apple	banana	orange	pear	pineapple
Document1	1	0	2	1	0
Document2	0	1	1	0	1
Document3	2	0	0	1	0

Source: The author.

- *Lemmatization*: This transforms all words into their base or dictionary form (their lemma), reducing the dimensionality of the data by removing inflections and variations such as prefixes, suffixes, and tenses.

Once preprocessing is done, feature extraction methods determine how documents are represented computationally to the machine learning algorithms (FACELI et al., 2021). One simple method is the *Bag of Words* (BoW), in which a dictionary is created of all words in the corpus. Then, a co-occurrence matrix is made, combining every document with every word in the dictionary. An example of this matrix is shown in Table 2.2. The values of this matrix are filled by counting the number of times each word appears in each specific document. This count is called the Term-frequency (TF). Alternatively, a slightly more detailed feature extraction method is the TF-IDF, in which the values of the matrix are filled with the TF multiplied by the *Inverse Document Frequency* (IDF). IDF measures how frequently a term appears in all documents of the corpus. TF-IDF is calculated as follows:

$$\text{tf-idf}(t, d, D) = \text{tf}(t, d) \times \text{idf}(t, D) \tag{2.2}$$

where:

- t is the term being considered.
- d is the document in which the term appears.
- D is the set of all documents in the corpus.
- $\text{tf}(t, d)$ is the term frequency of t in d, which measures how frequently t appears in d.
- $\text{idf}(t, D)$ is the inverse document frequency of t in D, which measures how important t is across all the documents in the corpus.

The IDF component is calculated as:

$$\text{idf}(t, D) = \log \frac{N}{|\{d \in D : t \in d\}| + 1} \tag{2.3}$$

where:

- N is the total number of documents in the corpus.
- $|d \in D : t \in d|$ is the number of documents in which the term t appears.

One characteristic of both BoW and TF-IDF is that they remove the relationship between the order of words in the documents. This happens because, in the co-occurrence matrix, every column represents only a single term. To add a representation of the order of words to this matrix, it is possible to use *n-grams* as columns (JURAFSKY; MARTIN, 2009). N-grams are contiguous sequences of words in a document. For example, a 2-gram represents sequences of 2 neighboring words. In the sentence *"The customer service representative resolved the issue."*, both *"customer service"* and *"service representative"* are 2-grams. N-grams capture the local context of words within the document, which can provide more information for the classification task of the supervised learning algorithms.

2.4.3 Cross-Validation and Evaluation Metrics

The validation of supervised learning algorithms is an important step in estimating the performance of trained models on unseen data. A model that performs well on training data might not do the same when classifying new data. As such, the validation of supervised learning algorithms is commonly performed using cross-validation. Cross-validation separates the training dataset into multiple subsets or folds. The model is then trained using a subset of these folds, while the performance is evaluated on the remaining folds. This process is repeated multiple times, and the folds used for training and evaluation are different each time. The final estimate of the performance of the model is the average of all evaluation results. A specific type of cross-validation is *k-fold*, in which the training dataset is divided into k equal-sized folds. The model is then trained k times on k-1 folds and evaluated on the remaining fold.

The result of the validation process is a confusion matrix. This matrix is typically a 2x2 table that summarizes the predicted and actual classifications within for cells: *True positives* (TP), *False Positives* (FP), *True Negatives* (TN), and *False Negatives* (FN). An example is shown in Table 2.3. TP and TN represent the number of items correctly classified by the machine learning models. On the other hand, FP and FN represent errors, with FP being items classified incorrectly as positive and FN items classified incorrectly as negative. The confusion matrix values are used to calculate multiple evaluation metrics,

Table 2.3 – Example of a confusion matrix.

		Predicted Class	
		Positive	Negative
Actual Class	Positive	TP	FN
	Negative	FP	TN

Source: The author.

such as precision, recall, F1 score, and accuracy. The equations for these metrics are as follows.

$$Precision = \frac{TP}{TP + FP} \tag{2.4}$$

$$Recall = \frac{TP}{TP + FN} \tag{2.5}$$

$$F1Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{2.6}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.7}$$

Precision evaluates how many predicted positives are true positives. It measures the model's ability to precisely identify positive items, without false positives. Recall evaluates how many true positives were predicted positives. It measures the model's ability to identify positive items without missing any. F1 score balances the results of the precision and recall measures, providing an overall score for the performance of the model based on the positive items. Accuracy evaluates how many accurate predictions were made out of all predictions. It is the only metric in this set of four that evaluates the prediction of both true positives and true negatives.

Another interesting evaluation is the *Receiver Operating Characteristic* (ROC) curve. This curve visualizes the trade-off between the True Positive Rate (TPR) and True Negative Rate (FPR) at varying classification thresholds. Each point of this curve represents a different threshold, with the optimal threshold being located at the top-left corner of the curve, where FPR is low and TPR is high. Calculating the area under the curve (AUC-ROC) is a common metric to evaluate the performance of the trained model. The equation for this metric is as follows:

$$AUC - ROC = \int_{-\infty}^{\infty} TPR(FPR^{-1}(t))dt \tag{2.8}$$

2.5 Chapter Summary

In this chapter, we presented fundamental concepts of BPM such as Dumas et al. (2018) BPM life-cycle and BPMN (OMG, 2011). The life-cycle displays the phases through which a business process is modeled, analyzed, redesign, and implemented. BPMN process models are the main type of artifact process analysts use when working with business processes. We dive deeper into the discovery, analysis, and redesign phases by exhibiting concepts related to the analysis and improvement of process models, which includes process modeling guidelines that aim at ensuring process models are easy to understand and have no errors. We defined PAIS and how they are related to the generation of event logs. These event logs are a key input for process mining techniques, which we introduce as one of the traditional methods of checking if business process executions have diverged from their process models.

We also presented concepts regarding training machine learning algorithms, which we use in Chapter 7 to develop automated classifiers. We show what are supervised learning algorithms, including the four algorithms used in this thesis. Based on the importance of using high-quality training data, we show techniques to balance datasets, preprocess, and perform feature extraction on textual data. We finalize by showing how to validate the results of trained machine learning models, such as k-fold cross validation and the common performance metrics used to evaluate the relationship between the real data and the predicted results.

3 RELATED WORKS

In this Chapter, we discuss related works to this thesis. We start by reviewing important studies in the development of process mining techniques, focusing primarily on conformance checking and concept drift detection. These techniques share the goal of our thesis of detection when process models need updates. While their dependency on event logs makes them inapplicable when those logs are unavailable, they have been widely studied and are easily available, making them very useful techniques.

Afterward, we review studies that have similarities with individual aspects of our thesis. These include studies on analyzing process models to identify non-explicit resources required for their implementation. The problem elucidated in these studies is similar to the problem of evaluating by hypothesis H_3, in which we want to identify explicit and non-explicit heterogeneous data sources in process models. It also includes studies evaluating reasons for business process non-compliance and deviation from process models. Understanding how business processes change is fundamental for creating methods for detecting those changes. As such, for hypotheses H_1 and H_2, we have investigated in-depth how change happens from practical and theoretical perspectives.

3.1 Studies on the Development of Process Mining Techniques

We have established in chapter 2.3 that process mining is a valuable tool for process discovery and conformance checking, i.e., detecting differences between a process model and events contained within an event log of a business process. There have been numerous studies on the use of process mining for conformance checking (see Table 3.1). They propose approaches and techniques to detect the differences and to update the process models.

One factor that varies between studies on conformance checking is the different measures of quality they choose to focus on. For example, the fitness of the updated process model is a measure of how well it replays the events present in the event log. The approaches may emphasize this fitness towards generalization, to prevent overfitting, or precision, to prevent underfitting. Simplicity may also be desired, that is, the process model should not have unnecessary complexity as defined by metrics such as *number of nodes* or *structuredness* (AALST; ADRIANSYAH; DONGEN, 2012). Besides these quality measures, some approaches focus only on certain aspects of the event log, such

Table 3.1 – Process mining techniques references.

Technique	References
Conformance checking	(AALST; ADRIANSYAH; DONGEN, 2012) (ADRIANSYAH; DONGEN; AALST, 2011b) (ADRIANSYAH; DONGEN; AALST, 2011a) (ADRIANSYAH; SIDOROVA; DONGEN, 2011) (BROUCKE et al., 2014a) (BROUCKE et al., 2014b) (CALDERS et al., 2009) (COOK; WOLF, 1999) (FAHLAND; Van Der Aalst, 2015) (GARCIA-BANUELOS et al., 2018) (GOEDERTIER et al., 2009) (MUÑOZ-GAMA; CARMONA, 2010) (MUNOZ-GAMA; CARMONA, 2011) (MUNOZ-GAMA; CARMONA; AALST, 2014) (POLYVYANYY et al., 2016) (REISSNER et al., 2017) (ROZINAT; AALST, 2008) (WEERDT et al., 2011)
Concept drift detection	(BOSE, 2012) (FIROUZIAN; ZAHEDI; HASSANPOUR, 2019) (HOMPES et al., 2017) (MAARADJI et al., 2017) (MAISENBACHER; WEIDLICH, 2017) (MARTJUSHEV; R.P.; AALST, 2015) (LI et al., 2017) (LIU; HUANG; CUI, 2018) (OSTOVAR et al., 2016) (PRATHAMA et al., 2019) (RICHTER; SEIDL, 2019) (STERTZ; RINDERLE-MA, 2018) (YESHCHENKO et al., 2019)
Online conformance checking	(STERTZ; MANGLER; RINDERLE-MA, 2020b) (STERTZ; MANGLER; RINDERLE-MA, 2020a) (TAVARES et al., 2019) (WEBER; TIÑO; BORDBAR, 2012)
Online concept drift detection	(CARMONA; GAVALDÀ, 2012) (CERAVOLO et al., 2020) (OSTOVAR et al., 2016) (SOUSA; PERES, 2020) (STERTZ; RINDERLE-MA, 2019) (STERTZ; RINDERLE-MA; MANGLER, 2020) (TAVARES et al., 2019)

Source: The author.

as Fahland and Van Der Aalst (2015) approach, which updates process models by sub-processes containing the new elements, but does not consider the organization or the resource perspectives of the business process.

The use of conformance checking has been proposed for more specialized cases. For example, due to the ubiquity of database management systems in most organizations (such as MySQL, PostgreSQL, Oracle, and others), Aalst (2015) analyzed how the record of database operations could be extracted and used to check the execution of the process activities. This work shows that to detect discrepancies in a process model, it is unnecessary to have a complete event log, that is, a log that shows records of all possible events when they are executed. Instead, by analyzing only the events of a specific resource, in this case a database, it is possible to perform conformance checking on the process model elements related to that resource. Additionally, the author argues that too much time is spent finding, selecting, converting, and filtering the data extracted from various systems for process mining. Thus he establishes twelve *guidelines for logging* that would help improve this data for process mining.

In addition to conformance checking, many other process mining studies present methods and analyses on *concept drift detection* (CDD) in event logs. A concept drift happens when a business process has changed while being recorded in an event log. Detecting these drifts involves challenges such as detecting when a change has happened, identifying the region of the changes, and characterizing what has changed (control-flow, data, or resources) and how (sudden, gradual, recurring or incremental changes) Bose (2012). The main difference between detecting business process changes through conformance checking and CDD is that the latter removes the need to compare the event log to an existing process model.

Two popular academic process mining software solutions, the Process Mining Workbench (ProM[1]) and the Advanced Process Analytics Platform (Apromore[2]) have techniques for CDD (OMORI et al., 2020). The first technique is the one proposed by Bose (2012) that is implemented in ProM. The second technique, implemented in Apromore, is proposed by Ostovar et al. (2016). These two techniques have been more thoroughly applied and tested in practice by being implemented in these tools. On the other hand, many other techniques have been proposed in the literature (see Table 3.1). Omori et al. (2020) performed an in-depth analysis of some of these techniques (BOSE, 2012; OSTOVAR et al., 2016; ZHENG; WEN; WANG, 2017; TAVARES et al., 2019), com-

[1] www.promtools.org
[2] apromore.org

paring their results to identify the trade-offs between usability, access, and testing of the implementations. Elkhawaga et al. (2020) also analyze concept drift studies through a systematic literature review. Based on their results, they established a framework through which users can evaluate the maturity of other concept drift studies. Yeshchenko et al. (2021) create visualizations of CDD to support process analysts in better understanding the drifts of a business process event logs.

Some of the literature focuses on the online process mining of *process event streams* for both conformance checking and CDD (see references in Table 3.1). Process event streams differ from event logs because they produce the events generated by the execution of business processes in real-time (RUTKOWSKI; JAWORSKI; DUDA, 2019). Thus, online process mining techniques have to read these events one-by-one, which incurs additional challenges since it can not be assumed that a business process's sequence of events is complete. Still, online process mining techniques also allow for a faster response time to any changes in the business process's execution.

To evaluate the requirements and performances of different online CDD techniques, Ceravolo et al. (2020) defined a set of goals commonly used to evaluate other online process mining techniques, such as process discovery and conformance checking. These goals include minimizing the memory consumption, the response latency, the frequency of runs of the technique, and the optimization of its accuracy. Based on these goals, they analyzed some of the CDDs techniques proposed in the literature (BOSE et al., 2014; OSTOVAR et al., 2016; TAVARES et al., 2019; YESHCHENKO et al., 2019; ZHENG; WEN; WANG, 2017).

As shown, online process mining techniques have the motivation to maintain up-to-date knowledge of business processes, which is similar to our motivation to maintain process models updated. However, our work looks for alternative data sources to analyze in the absence of event logs. Nevertheless, both approaches would allow for the faster detection and prevention of mistakes in the execution of business processes (CERAVOLO et al., 2020). They would also empower organizations to make better decisions with the up-to-date information acquired through both approaches.

3.2 Studies on Business Process Resources and Change Reasons

In addition to studies related to the detection of business process changes, another important field to review are the studies that have similarities to our proposal of identify-

ing alternate data sources for detecting the changes in the absence of event logs. These data sources can be resources required for the business process to function and that are part of the organization's infrastructure, such as software, hardware, or people. One of the challenges of BPM is that organizations do not always have these resources ready to implement their process models (CONFORT, 2010). As such, approaches that help identify which resources are necessary for implementation are useful for designing process models. Biazus et al. (2019) present a semi-automatic approach for this problem, in which they recommend possible software resources based on analyzing the textual labels present in a process model. This recommendation aims to inform what resources would be necessary to implement the process model.

The alternate data sources can also be external data storage and services utilized by business processes. BPMN offers ways to display these in the process model, but they are not always utilized. It may be necessary to infer their existence based on the semantics of the BPMN elements present in the process model. Balbinot, Thom and Fantinato (2017) address this problem, presenting definitions of recurring types of process data utilized by business processes. These definitions identify the data passed between activities and participants of a business process and the use of external services and databases that can contain these process data. With these definitions, the authors relate how BPMN elements in the process models can represent the process data and external services being utilized.

Other authors have tried to conceptualize and define what can be a business process change. Alter (2014) defines *workarounds* as goal-driven changes made to overcome or minimize the impact of an obstacle that prevents a work system (e.g., a business process) or its participants from achieving a desired level of efficiency. Workarounds are a response to problems regarding the inefficiencies in how work is done or the misalignment of goals and interests between different participants and management layers. While they can be created as a "temporary quick fix", they can also be long-lasting, in which case they should be properly integrated into the process models, mainly because these fixes can also create inefficiencies and hazards of their own.

Andrade et al. (2016) performed a case study to evaluate the effects of business process non-compliance. In this study, non-compliance refers to instances in which the execution of the business process does not match what is intended by the management team (e.g., in a process model). The authors discovered nine business processes of a German IT company and identified all instances of non-compliant behavior. They detected five factors that triggered non-compliant behavior and organized them into two classes:

Intended non-compliance, containing factors *desire to improve process outcome*, *desire to prevent future mishaps*, *desire to avoid tedious tasks*; and *Unntended non-compliance*, containing factors *lack of knowledge* and *carelessness*. The authors concluded that well-intended non-compliance mainly had positive effects on the business process outcomes, though most instances of non-compliance had negative effects.

Finally, König, Linhart and Röglinger (2019) create a set of 33 reasons for why business processes deviate from their intended behavior. The authors classify three concepts as being related to process deviation: *exceptions*, which are sudden unexpected interruptions of process executions, *workarounds*, which are intentional adaptations or improvisations in single tasks or sub-processes, and *non-compliant processes*, which occur when the business process executes differently from predefined specifications, such as their process models. These reasons were compiled through a Delphi study that interviewed 30 specialists to propose and consolidate the set of reasons, which include, for example, problems with the business process tasks, problems with its documentation, the lack of reviews to update process models, the unavailability of resources, the process participants changing the business process by themselves, and others. As such, the authors show that there are many reasons for the execution of business processes to deviate from the intended behavior and that these reasons can cause intentional change to business processes. However, the authors do not address what to do when these changes are identified. In our thesis, we have proposed using the reasons as a starting point to understand how business processes change and discover the means through which we can monitor those changes.

3.3 Chapter Summary

In this Chapter, we reviewed related works regarding business process change. We explored these works first by detailing studies on the development of conformance checking techniques using process mining. Like all process mining techniques, conformance checking relies on event logs to discover when the execution of a business process is different from the elements in the process model. We also explored concept drift detection, which detects business process changes by comparing the traces of the same event log at different points in time.

We presented two popular academic process mining software that implements conformance checking and concept drift detection algorithms. We also explored studies

that evolve these algorithms to perform online process mining on event streams, allowing the analysis of business process executions in real time. The common factor of all these approaches is that they still rely on event data as a trustworthy source of information on the operation and behavior of a business process in practice.

Finally, we discussed a few studies regarding the resources for implementing process models and the possible reasons for business process change. These studies and our thesis have similar objectives, particularly in our search to find alternate data sources for detecting business process change. The lack of resources is a good example reason for business process change, and attempting to prevent it requires an in-depth analysis of the process model to find all required resources for its execution.

4 REVIEWING THE STATE OF PROCESS MODEL MAINTENANCE

In this thesis, we study the feasibility of a framework to identify outdated process models whenever there are changes to the components linked to its execution. We aim to understand how business processes change to identify relevant components that could be observed and monitored, becoming heterogeneous data sources for the updating of outdated process models. In Figure 4.1, we display an overview of the methodology employed in this thesis, in which we organize every topic of our study based on the hypothesis we defined in Section 1.2.

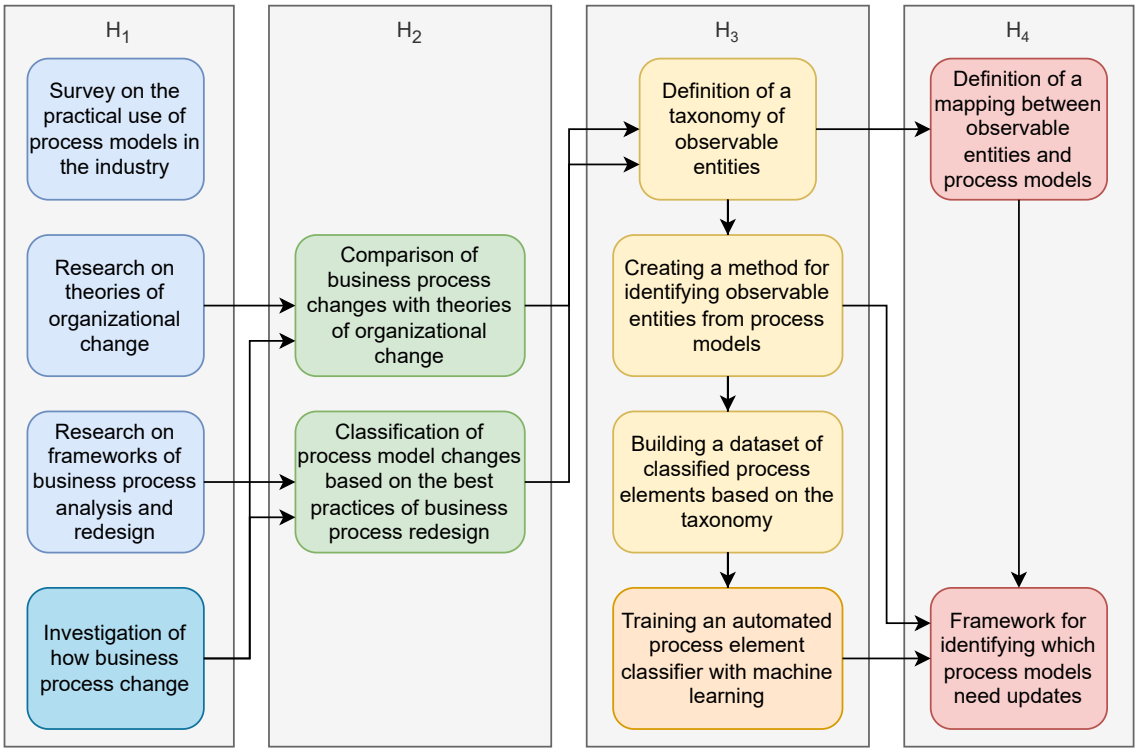
In the remainder of this Chapter, we present the first three topics of our methodology. We start with our research to discover what is the state of process model maintenance in practice. As we have elucidated in Section 1.1, sometimes organizations face challenges when trying to move on from the analytical phases of BPM to the implementation of their process models, which usually leads to these process models becoming outdated.

However, we did not have details regarding how organizations use their process models and what difficulties they meet when trying to apply their BPM knowledge. So, we deployed a public survey to investigate how process models are currently being used in organizations. This way, we would empirically show how frequently organizations have outdated process models and what circumstances may cause this problem to occur. The results of this survey can be seen in Section 4.1.

After the survey, we sought to understand business process change from a theoretical perspective. We studied the literature on organizational change to find the different theoretical perspectives on the topic of change. From these perspectives, we aimed to discover a potential form of change that could be compared to how changes happen in BPM. Thus, Section 4.2 shows existing organizational change theories. This Section also examines the similarities between basic concepts of organizational change theories and how change can be viewed in BPM.

Finally, in Section 4.3 we explore the frameworks of Alter (2013) and Reijers and Mansar (2005) in the context of business process redesign. We aimed to understand what components of process models have been studied and related to the improvement of business processes. We have found these frameworks useful for defining potential heterogeneous data sources.

Figure 4.1 – Methodology employed in this thesis.



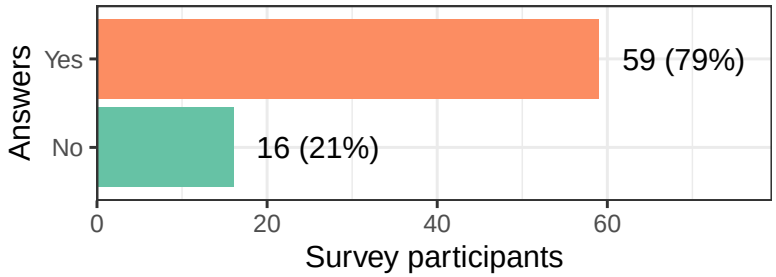
Source: The author.

4.1 Surveying the Use of Process Models in the Industry

As shown in Section 2, it is expected that organizations applying BPM use the entirety of its life-cycle, since by doing so, they can manage and improve their business processes by discovering and analyzing up-to-date process models. However, many organizations may try to use a limited set of methods and techniques from BPM due to them being unable to complete all BPM life-cycle' phases. As such, these organizations may face challenges in identifying when their process models have to be reviewed, which may cause problems and errors to appear when someone fails to understand and communicate with others about a business process through its, possibly outdated, process model.

To evidence that organizations face these challenges in practice, we employed a survey to find how are the state of process models and the business process life-cycle in organizations. We sought to discover the reasons why organizations change their business processes and how outdated process models are identified and updated. The survey form can be seen in Appendix B at the end of this document. This survey was broadcast to students of IT Management and Software Engineering of a federal university, the "Latin-

Figure 4.2 – Answers to the question: Is there a process modeling initiative in your department or organization?



Source: The author.

Table 4.1 – Distribution of responses on the use of process models.

How are the process models used?	Answers
They are used as documentation.	36 (61,02%)
They are used as a teaching tool.	12 (20,34%)
They are implemented ...	41 (69,49%)
... manually.	12 (20,34%)
... using a system of the organization.	18 (30,51%)
... using a commercial system (e.g. a BPMS).	20 (33,90%)
Other.	03 (5,08%)

Source: The author.

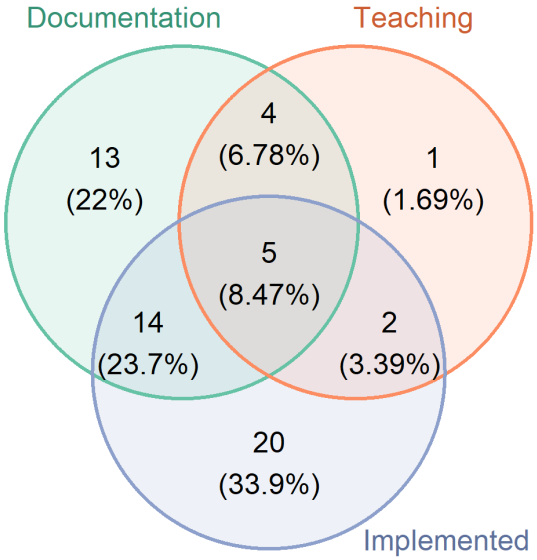
American Community of BPM" (BPM-LAC)[1] and to personal contacts on LinkedIn[2]. As can be seen, the survey targeted primarily Latin-American organizations that likely use BPM, since outdated process models are a problem that frequently occurs in Latin-American countries, especially in Brazil (CONFORT, 2010).

The survey collected a total of 75 participants. The survey first asked them if their organizations had a process modeling initiative active, with 59 participants giving an affirmative answer (Figure 4.2). In addition, most participants showed that their organizations have a strong interest in using process models to improve productivity, choosing mainly to implement their processes in an automated system (see table 4.1). However, the low use of these process models as a teaching tool is worrying, since most organizations also use them as documentation. Additionally, only 19 participants reported that their organizations use process models for both documentation and implementation (see Figure 4.3), which accounts for roughly a third of all answers in either of these options.

The participants were also asked if the business processes of their organizations have undergone evolution or changes and for what reasons. Of the 75 participants, 59

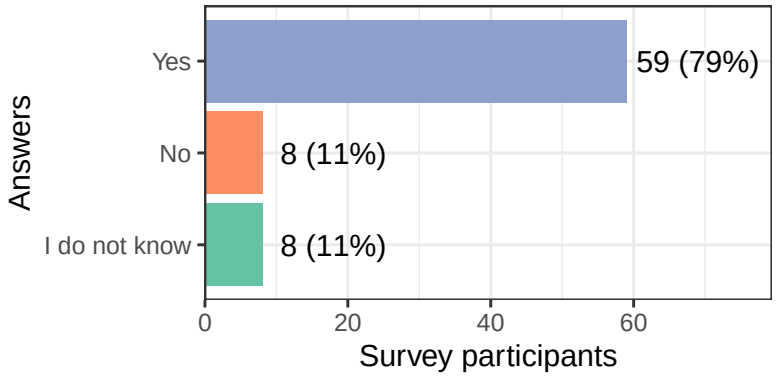
[1] https://www.facebook.com/BPMLAC/
[2] https://www.linkedin.com/

Figure 4.3 – Venn diagram of the responses on the use of process models.



Source: The author.

Figure 4.4 – Answers to the question: Have your organization's processes ever evolved or changed for any reason?



Source: The author.

of them (unrelated to the 59 in the first question) said yes (see Figure 4.4), mainly for reasons of process performance or because of new requirements (see Table 4.2). Together, we asked the participants to describe, optionally and in free text, how these changes were made in their organizations. Based on 19 responses, it was possible to see that there is a dependence on meetings and projects to make a change. It was also noticed that, in some cases, the need for this change is detected only when a problem appears. There is an immaturity in the management of the processes of these organizations because, in addition to multiple complaints about the lack of disclosure of the changes, only two responses mentioned changes in the process models, which should be vital both for the change management and for the documentation of the business processes.

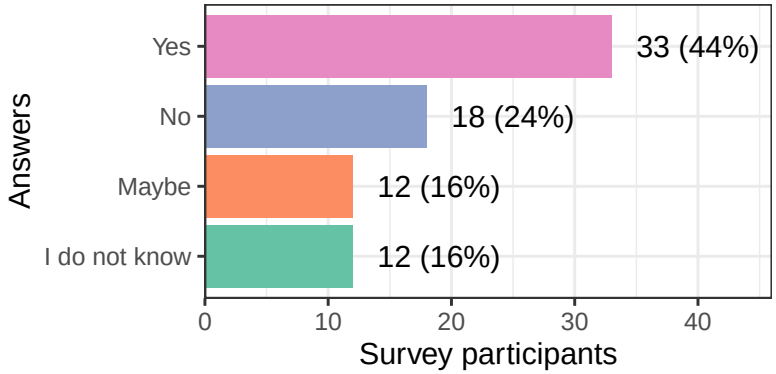
Finally, we asked participants about updating process models in their organiza-

Table 4.2 – Distribution of answers on the reasons for the change/evolution of processes in an organization.

Reason for the change/evolution of business processes	Answers
To increase performance.	37 (62,71%)
To reduce costs.	25 (42,37%)
To adapt it to changes in employees responsibilities.	16 (27,12%)
To adapt to personnel changes (e.g. someone left the organization).	20 (33,90%)
To meet new requirements (e.g. new regulations).	35 (59,32%)
Other.	5 (8,47%)

Source: The author.

Figure 4.5 – Answers to the question: In your organization, is there an effort being made to maintain process models up-to-date?

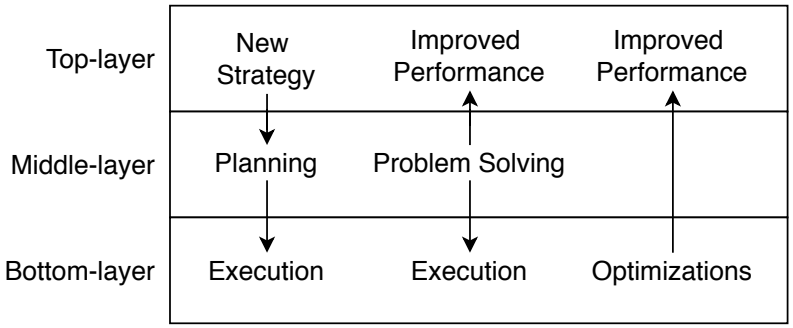


Source: The author.

tions, as well as identifying outdated models. 60% (45) of the participants stated or at least suspected that their organizations try to keep their process models up to date (see Figure 4.5). However, eight out of a total of twenty participants described that the identification of outdated process models only occurs when the process begins to fail during execution, compared to three participants who indicated the existence of a periodic review of the process models (the other nine responses did not present relevant information or were ambiguous). Also, no participants mentioned if process mining was used to detect outdated process models.

The answers to this questionnaire present a context in which an organization's processes are modeled and implemented, but these models often become outdated over time due to their lack of connection with the implementation of the business processes. Also, projects that change business processes do not usually include the task of updating the models. As a result, these process models are often not used for teaching about the business processes because when they are, it is common for errors to occur during the execution of the process. The lack of communication of changes also contributes to the occurrence of errors, which causes the sudden need to update the models.

Figure 4.6 – Example of a typical management division and how change may propagate.



Source: The author.

This context showed us that a framework is necessary for maintaining process models up-to-date for organizations like those discovered by our survey. Due to not implementing and automating these process models in a PAIS, these organizations lack the event logs necessary to perform process mining to detect business process changes. Consequently, they depend on the manual update of the process model, which is sometimes very costly and may need to happen at a critical moment of an organization, such as when the business process operation has errors.

4.2 Theories of Organizational Change

Understanding how and why change happens in an organization is a complex challenge. Change is an integral part of an organization's management to adapt itself against obstacles and be more competitive (BURKE, 2017). Most organizations organize this management in layers that divide the responsibilities and people. A typical management division (see Figure 4.6) includes three layers, where the top-most layer defines goals and strategies for the organization, the middle layer organizes how to fulfill those goals and execute those strategies, and the bottom-most layer operates the work as organized by the middle layer (SADIQ et al., 2007; BURKE, 2017). In this typical management division, a change might start from a new strategy established by the top layer, which is then propagated to planning and then to execution. However, change might also happen naturally from slight optimizations performed by the workers from the bottom layer. It is also possible for the middle layer to update its plans to solve problems as they appear, without needing instructions from the top layer.

What this management division illustrates is that a change in an organization can start at any level of management. Its propagation is not limited to a top-down or a bottom-

up approach. However, all management layers must be knowledgeable of what is being done and decided in the other layers. From the perspective of BPM, it is important that the business processes in execution, i.e., the bottom layer, are known in their entirety by the process analysts and the management teams in the middle and top layers to manage their performance and develop new strategies and goals.

Appropriately, business process change is one of the central features of a BPM life-cycle. There are many techniques and strategies available for smart managers to analyze, redesign ,and improve business processes in a proactive manner, such as redesign heuristics (DUMAS et al., 2018; REIJERS; MANSAR, 2005), process mining (Van Der Aalst, 2009; Van Der Aalst, 2016), and modeling guidelines (MENDLING; REIJERS; AALST, 2010; AVILA et al., 2019; AVILA et al., 2020). However, due to how it is structured, the BPM life-cycle is more suited to execute changes in business processes from a top-down approach since it follows a strict order of events: analysis $\rightarrow$ new goals $\rightarrow$ redesign $\rightarrow$ implementation. When a change happens outside of this order (PLATTFAUT et al., 2011), such as when it happens from the bottom layer of management, the BPM life-cycle relies on its cyclical aspect to catch-up to those changes and to update its conceptual process models.

To further understand how and why change happens in BPM outside of its life-cycle, it may be helpful to investigate existing organizational change theories and compare them to how BPM works. Examining these theories through a business process perspective may give us insight into managing change in the BPM life-cycle better. Many theories have been written and continue to be written every year, but no single theory has been established as the best one for every organization.

Ven and Poole (1995) have explored many of the different ideologies and perspectives that formed 20 developmental theories that tried to explain how organizational change happens. Their work discusses that the interplay of these theories can help draw a clearer picture on how and why organizational change happens since any theoretical perspective alone offers only a partial explanation of a complex phenomenon (VEN; POOLE, 1995). The authors analyzed the 20 developmental theories selected for their work and tried to explain the difference between them. To do so, they first defined change as an event in which a difference in form, quality, or state over time in an organization's entity is empirically observed, that is, its people, its products, its programs, or its jobs. After the definition of change, the authors outlined four basic theories of change and related the 20 developmental theories analyzed to these four types. The four basic theories are:

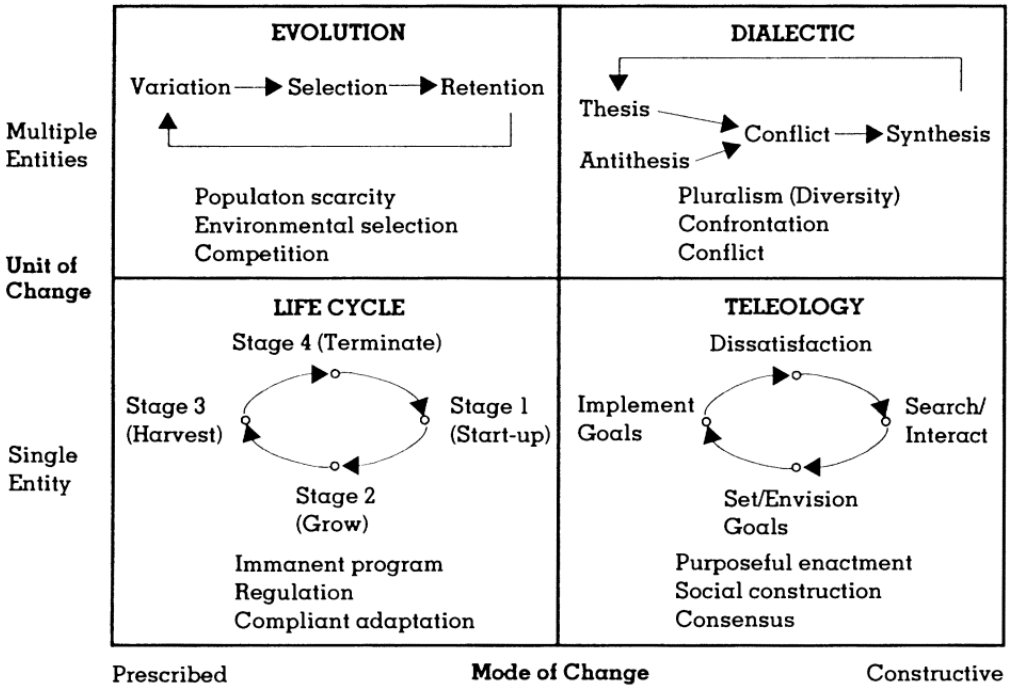
Life-cycle theories express that change is imminent and that it follows a prescribed series of stages. Each stage contributes to the next and the final product results from each of these contributions. As such, there is an underlying model that determines the series of stages, governs the progress through them, and instructs how change must happen within each stage. This change is linear and irreversible. An example of this can be seen in some software development methodologies, such as Scrum which has distinct phases with a sprint to plan, develop and deliver a software in increments Schwaber and Sutherland (2012).

Teleological theories connect change to a purpose or goal that is desired and is the final cause for this change. It is assumed, for example, that an organization has the creativity to define a goal to pursue and that it can adapt itself to achieve it by taking action toward the goal. Thus, these theories view change as a "repetitive sequence of goal formulation, implementation, evaluation, and modification of goals based on what was learned" (VEN; POOLE, 1995). However, there is no prescribed model that defines the necessary steps to achieve these goals.

Dialectical theories assume that there are two or more distinct entities that compete with each other due to contradictory values or priorities. These entities can be internal, belonging to the same organization, or external, in the form of other organizations. The conflict of two entities may be described as one entity subscribing to a thesis (A) that is challenged by an antithesis ($Not - A$) of the opposing entity. These entities maintain a status quo until one of them gains sufficient power to force a change. The result of this change may be a synthesis of all conflicting theses or a complete replacement of the status quo in favor of the winner's thesis. Dialectical change can often be seen in political contexts, where multiple parties have to negotiate and compromise to proceed in their work.

Evolutionary theories focus on change in populations of organizations across communities, industries, or society at large. This change follows a continuous cycle, as in biological evolution: Variation, when novel organizations emerge at random; Selection, when organizations compete for resources and best fitting is selected; and Retention, when some organizations are maintained due to other forces (such as inertia and persistence). Change in these theories is recurrent, cumulative, and probabilistic. Thus, evolutionary change can be exemplified by the natural selection of software that perform a specific function in a market. The best software applications may be chosen and used by more users, which leads to them being further improved by their developers, while the others

Figure 4.7 – Theories of Organization Change



Source: Ven and Poole (1995)

may eventually stagnate and be abandoned due to a lack of interest from both users and developers.

Ven and Poole (1995) established a typology of these theories based on four distinguishing characteristics: a cycle of change events, a "motor" that generates change, a unit of change, and a mode of change. Figure 4.7 shows the four basic theories organized in terms of these four characteristics. Each cell illustrates the cycle and the motor of change of each theory. The unit of change differentiates theories that focus on change that happens to a single entity (Life-cycle and Teleological) or multiple entities (Evolutionary and Dialectical). Mode of change separates theories that say change happens according to a prescribed mode and those that happen in a constructive mode. In the prescribed mode, change is often small and predictable. A larger change in this mode happens over the long term. In the constructive mode, change is unpredictable and may create a significantly different entity.

The use of this typology and its four basic theories has appeared in modern research in computer science. For example, Dong et al. (2013) show that these theories can be used to develop a framework for how information technology can be applied for innovation. Shanks and Johnston (2012) explore the use of these theories in information systems research, suggesting that they are useful for analyzing longitudinal data from case studies. Plattfaut et al. (2011) show that BPM maturity models implicitly rely on

life-cycle theories, but other theories, such as evolutionary theory, may fit better in certain organizations.

One of the benefits of using Van de Ven and Poole's work is that it helps us understand, from a theoretical perspective, the *how* and sometimes the *why* of organizational change (WHETTEN, 1989; BURKE, 2017). Other organizational change theories have a greater focus on defining *what* to change. The content of this *what* can vary. Many works (PORRAS; ROBERTSON, 1992; WEISBORD, 1976; NADLER; TUSHMAN, 1980; LEAVITT, 1965; TICHY, 1983) present models defining different possibilities for the *what* of organizational change, such as an organization's strategy, mission, structure, technology, rewards, leadership, tasks, people, culture, and others. While models presented in these theories are valuable, in this work our perspective on *what* changes is the business process and its conceptual process model, and our focus is on understanding the relationship between the elements of a business process and how they change. Therefore, we believe it is more reasonable to use Ven and Poole (1995) typology to analyze change in business processes and their process models.

Weick and Quinn (1999) also refer to the typology of Ven and Poole (1995). They note that the language of motors is useful in the analysis of change theories, because it draws attention to the process of change instead of the outcome, and that it is important to avoid a mismatch between the prevailing conditions that cause change and the kind of motor that is activated. While Van de Ven and Poole classified the theories through the mode of change and unit of change, Weick and Quinn suggest that the tempo of change is also a meaningful partition. They define that change may be either episodic or continuous. Episodic change groups together organizational changes that are infrequent, discontinuous, and intentional. They are occasional interruptions that tend to be dramatic, short-run, and broad in scope. Comparatively, continuous changes are ongoing, evolving, and cumulative. They happen in the long-run, through recurrent small-scale adaptations.

When comparing Van de Ven and Poole's typology to the BPM life-cycle, it is possible to observe similarities between it and both life-cycle and teleological theories. In life-cycle theories, the similarities are in the prescribed series of stages through which business processes are created. Since business processes are often in constant execution in an organization, the life-cycle stability may be a prized characteristic to manage business processes better and prevent errors. On the other hand, the BPM life-cycle is also similar to teleological theories based on the purposeful enactment of change and the pursuit of goals since it contains specific phases to analyze and improve process models and their

respective business processes.

4.3 Frameworks for Business Process Analysis and Redesign

In addition to organizational change theories, it is useful to understand other concepts related to the work performed by business processes in organizations. Some frameworks proposed in the literature relate the business process and its activities to the components involved in their execution. Examples are the Work System Theory (WST) framework by Alter (2013) and the framework for business process redesign of Reijers and Mansar (2005).

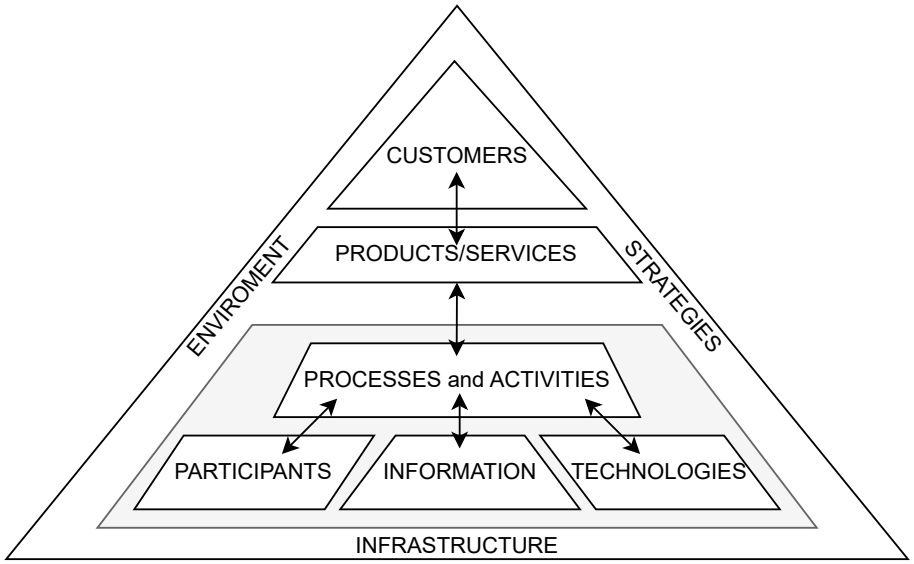
4.3.1 WST Framework

Alter (2013) defines a *work system* as a system in which the work (i.e., business processes and activities) is performed by human participants and/or machines to produce products and services. This system produces these outputs using information, technologies, and other resources for internal and/or external customers. The main proposal of the WST framework is to provide a natural unit of analysis for thinking about an organization's systems, whether those systems are fully automated, have IT systems supporting the work, or perform work unrelated to IT.

A static view of the WST framework is presented in Figure 4.8. As seen in this framework, (business) processes and activities are at the center. Connected to the business process, the other three components, *participants*, *information*, and *technologies*, are considered to be entirely within the work system of an organization. They are defined as:

- *Participants*: The people who perform the work, independent of whether they are IT users or not. This does not include automated agents performing work.
- *Information*: Informational entities that are used, created, captured, transmitted, stored, retrieved, manipulated, updated, displayed, and/or deleted. It includes, for example, orders, invoices, warranties, schedules, income statements, reservations, medical histories, resumes, job descriptions, and job offers. Information may or may not be computerized. Thus, it includes conversations and verbal commitments.
- *Technologies*: The tools used by participants and the automated agents to perform the work, including hardware/software that perform totally automated activities.

Figure 4.8 – Work System Theory (WST) framework.



Source: (ALTER, 2013)

Partially inside and outside the work system are *products/services*, which are produced within the system, and *customers*, which frequently are participants within the work system. They are defined as:

- *Products/services*: Information, physical things, and/or actions produced by a work system for the benefit and use of its customers.
- *Customers*: Recipients of a work system's products/services for purposes other than performing work activities within the work system. External customers are the organizations' customers, whereas internal customers are those employed by the organization, such as customers of a payroll work system. Customers can also be participants (e.g., patients in a medical exam or students in an educational setting).

The other three elements, *infracstructure*, *enviroment*, and *strategies*, exist outside of the work system, yet they may directly affect it. They are defined as:

- *Infrastructure*: Includes relevant human, information, and technical resources used by the work system but managed outside of it and shared with other work systems.
- *Enviroment*: Includes the relevant organizational, cultural, competitive, technical, regulatory, and demographic environment within which the work system operates and that affects the work system's effectiveness and efficiency. Organizational aspects of the environment include stakeholders, policies and procedures, and organizational history and politics, all of which are relevant to the operational efficiency and effectiveness

- *Strategies*: Includes the three levels of strategy, that is, enterprise strategy, department strategy, and work system strategy.

An interesting application of the WST framework is the snapshot, which summarizes the work systems based on the six elements inside or partially inside it (process and activities, participants, information, technologies, customers, and products/services). Table 4.3 presents an example provided by Alter (2013) of this snapshot for a hiring system. This summary is rigorously created by making lists for each of those elements following a few consistency rules:

- Each process and activity listed must be stated as a complete sentence that briefly specifies which participants perform the work and what they do.
- Each participant must be involved in at least one step in the processes and activities. Customers are viewed as participants if they participate in at least one of the steps.
- Information and technology entities listed must be created or used in at least one step in the processes and activities.
- Each product/service must be received and used by at least one customer.
- Each customer must receive and use at least one product/service.

In addition to the WST framework, Alter (2013) also presents a work system life-cycle model, which depicts how work systems change over time through planned and emergent (unplanned) changes. This life-cycle is shown in Figure 4.9, showing four phases: initiation, development, implementation, and operation and maintenance. In each of these four phases, inward-facing arrows represent emergent changes, such as adaptations and workarounds that change the work system without demanding significant resources. Thus, this life-cycle is yet another perspective that shows how business process change can occur independently of the full cycle of the BPM life-cycle.

4.3.2 Framework of Business Process Redesign

As mentioned in Section 2.1.1, the BPM life-cycle has a phase dedicated to business process redesign, in which process models are proactively changed to design an improved version of the business process. Reijers and Mansar (2005) developed a methodology for business process redesign, which included defining a framework and identifying best practices (or heuristics).

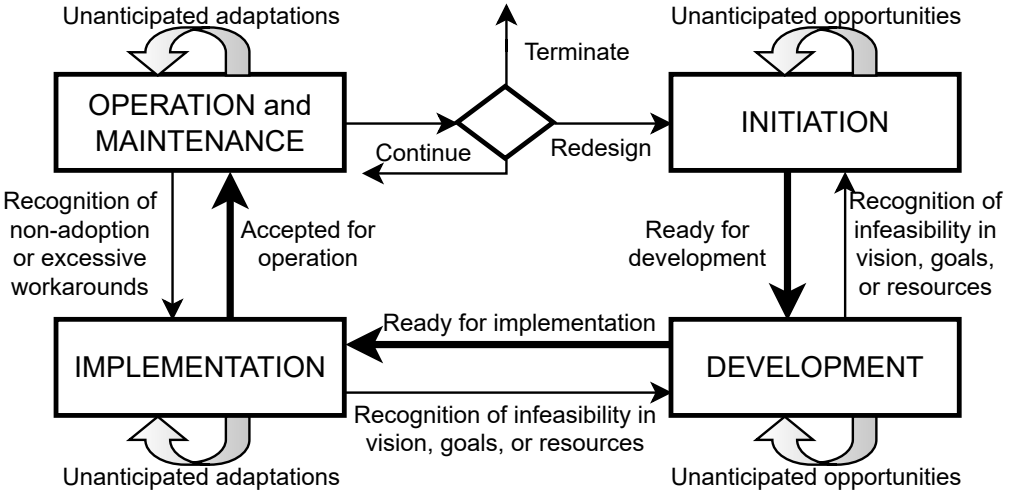
Table 4.3 – An example of a snapshot detailing all components of a work system.

Customers	**Products/services**
Hiring manager Larger organization (which will employ the new hire) HR manager (who will analyze the nature of applications)	Applications (which may be used for subsequent analysis) Job offers Rejection letters Hiring of an applicant
Main activities and processes	
Hiring manager submits request for new hire within existing budget. Staffing coordinator defines the parameters of the new position. Staffing coordinator publicizes the position. Applicants submit job applications. Staffing coordinator selects shortlisted applicants. Hiring manager identifies applicants to interview. Staffing coordinator sets up interviews. Hiring manager and other interviewers perform interviews. Hiring manager and other interviewers provide feedback from the interviews. Hiring manager makes hiring decisions. Staffing assistant sends offer letters or rejections. Successful applicant accepts or rejects job offer or negotiates further.	

Participants	**Information**	**Technologies**
Hiring managers Staffing coordinator Applicants Staffing assistant Other employees who perform interviews	Job requisition Job description Advertisements Job applications Cover letters Applicant resumes Short list of applicants Information and impressions from the interviews Job offers Rejection letters	New HR portal that is being built Word processor Telephones Email

Source: Alter (2013)

Figure 4.9 – Work system life-cycle model

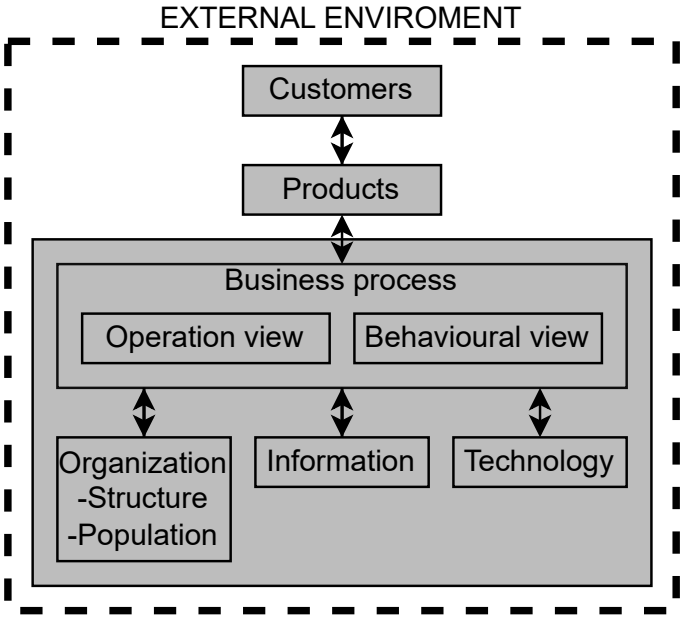


Source: (ALTER, 2013)

Similar to the WST framework of Alter (2013), Reijers and Mansar (2005) defined a framework to help practitioners identify relevant topics that should be considered when redesigning business processes and how these topics are related. As such, the definition of their framework was based on an examination and synthesis of multiple business process analysis frameworks and models proposed in the literature, including the WCA framework (the predecessor of the WST framework (ALTER, 2013)), the MOBILE workflow model (JABLONSKI; BUSSLER, 1996), the CIMOSA enterprise modeling views (BERIO; VERNADAT, 2001), and the process description classes of Seidmann and Sundararajan (1997). Figure 4.10 presents the defined framework containing seven elements, as defined by Reijers and Mansar (2005):

- The *Customers*, both internal and external, of the business process.
- The *Products* (or services) generated by the business process.
- The *business process*, which has two views:
  - The *operation view* defines what the business process does and how (the number, relative size, nature, and degree of customization of the business process tasks).
  - The Behavior view defines when tasks are done (sequencing of tasks, task consolidation, scheduling of jobs, etc.).
- The *participants* considering the organization's:
  - *structure* (elements, roles, users, groups, departments, etc.).
  - *population* (the individuals or agents who receive and execute tasks.

Figure 4.10 – The Business process redesign framework.



Source: (REIJERS; MANSAR, 2005)

- The *Information* used or created by the business process.
- The *Technology* used during the business process.
- The *external enviroment* other than the customers.

The framework of Reijers and Mansar (2005) has similarities with the WST framework, though it focuses more on the business process and the components required to execute it. This focus is aligned with the goals of the BPM redesign phase, considering that it is these elements (the business process operation and behavior, the participants, the information, and the technology) that most commonly are changed and improved. Changing products and external customers are outside the scope of Reijers and Mansar (2005), given that those changes are more related to the strategy level of organizations (see the Top-layer in Figure 4.6).

After defining their framework, Reijers and Mansar (2005) describe and evaluate 29 best practices for business process redesign. These best practices aim to help a redesigner to implement an improved business process design. Reijers and Mansar (2005) also present a classification of these best practices, organizing them according to which framework elements they are oriented towards. This classification contains seven classes:

- *Customers*, which focuses on improving contacts with customers.
- *Business process operation*, which focuses on how to implement the business process.

- *Business process behavior*, which focuses on when the business process activities are executed.
- *Organization*, which considers both the structure of the organization (mostly the allocation of resources) and the resources involved (types and numbers).
- *Information*, which describes best practices related to the information that a business process uses, creates, may use, or may create.
- *Technology*, which describes best practices related to the technology the business processes use or may use.
- *External environment*, which tries to improve upon the collaboration and communication with third parties.

In the context of our research, the best practices goals differ from our goals of investigating and detecting already finished business process changes. As such, our focus in this thesis is primarily on this classification since it cannot only classify business process changes that are improvements, they can easily be adapted to classify any type of business process changes.

4.4 Chapter Summary

In this Chapter, we presented the results of a survey we performed to discover how process models are being used in practice in Latin American organizations. The questions of this survey were mainly aimed at discovering how process models are used in organizations and if these organizations have established methods to detect and update outdated process models. Based on the answers, we were able to confirm that organizations are often able to implement their process models, yet outdated process models are a frequent problem due to the organization's inefficiency in detecting them.

We also presented our study of the literature regarding organizational change theories. We outlined the four main theories of change of the typology of Ven and Poole (1995) and the notion of the tempo of change (WEICK; QUINN, 1999). We compared these concepts with the BPM life-cycle, showing that some theories present mechanisms of change that are not represented in the BPM life-cycle.

Finally, we analyzed existing frameworks of business process analysis and redesign. We highlighted the frameworks of Alter (2013) and Reijers and Mansar (2005), which break down the work systems that execute business processes and their activities.

These frameworks detail the components connected to business processes because they are used or produced by their execution. To analyze these components, we presented how they can be summarized in a work system snapshot. We also showed another life-cycle model presented by Alter (2013), in which changes to a work system can happen at any point of this cycle, further emphasizing the necessity of methods of detecting these types of changes in business processes.

5 INVESTIGATING BUSINESS PROCESS CHANGE IN PRACTICE

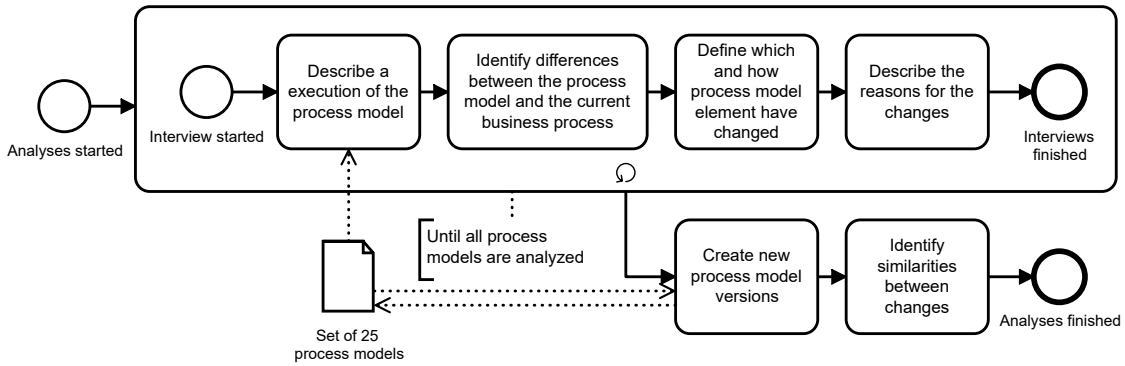
This chapter presents our approach to achieve the objectives related to hypotheses H_1 and H_2. H_1 is related to whether we can analyze business process change by re-evaluating old process models, updating them with the necessary changes, and comparing the two versions. Our main goal in answering this question was to have a set of *before* and *after* process models, which we can analyze to continue our study on identifying heterogeneous data sources to maintain process models updated. In Section 5.1, we describe a set of interviews we performed to analyze old process models with their respective domain experts and acquire a set of updated process models for our analysis.

Following the interviews, we attempted to achieve the objectives related to H_2. We have used frameworks from the literature regarding business process redesign and organizational change to evaluate the changes observed by updating the process models analyzed during the interviews. In Section 5.2, we detail and categorize the observed process model changes by adapting the best practices proposed by (REIJERS; MANSAR, 2005). We show that many observed changes are related to components other than the operation and behavior of the business process. In Section 5.3, we evaluate the changes through the perspective of organizational change theories to conclude what can define a heterogeneous data source for monitoring and updating process models.

5.1 Discovering Business Process Changes Through Interviews With Domain Experts and Examining Their Process Models

We have analyzed a set of 25 business processes from departments of a Brazilian university. These business processes were modeled with BPMN by groups of students learning BPM in a course offered by that institute between the 2013 and 2017 academic years (THOM, 2020). Each group of students, assuming the role of process analysts, was tasked with modeling one business process of one of the institute departments out of four departments. They participated in meetings with a domain expert of that department, i.e., the person who knows how all business processes operate, during which they interviewed the domain expert about the business process. They were also encouraged to ask for and analyze any documentation related to the business process. After the meetings, they created the process model based on the information discovered. For most students, these meetings were their first real experience with discovering a business process and process

Figure 5.1 – How the analyses of the business processes were performed.



Source: The author.

modeling. By the end of the BPM course, the domain experts manually validated the resulting process models, and the student's lecturer verified their correctness. The process models were also documented in a collaborative wiki created for this course, and they are available for consultation on the institute's intranet. Despite their availability, they were not implemented in any PAIS.

When we chose to analyze these business processes, it was assumed that enough time had passed since they were modeled for some of them to have changed. We also understood that the process models' semantics and syntax were already validated, so we would not find many modeling errors that would confuse the domain experts and us when we reviewed these process models. Nevertheless, we reviewed every process model shortly before the interviews to ensure only those without syntactical errors would be analyzed. We also applied process modeling guidelines (AVILA et al., 2020) to guarantee the process models' understandability without altering their semantics.

Therefore, we believed that performing revision interviews with the domain experts of each department would result in a set of updated process models that captured noticeable changes compared to their previous versions. As such, we started our analyses according to the steps seen in Figure 5.1.

We performed four revision interviews in which we interviewed one domain expert about the business processes of their department. Of the four interviewed domain experts, two had participated in the students' process modeling. The other two had replaced the previous domain experts and, thus, had not participated in the previous process modeling task. In these interviews, we focused primarily on collecting data on the operational level of the business process because the domain experts were usually limited to a perspective at that level. They were responsible for executing most of the business

process activities, and they did not often engage with the design and analysis of the process model since they had almost no experience working with BPMN. Nevertheless, they provided information regarding all activities present in the previous process models.

For every interview, we presented a list of known business processes of that department, along with their process models. The interviews were guided by four core questions:

1. Which business processes are still in operation?
2. Since the processes were modeled, has the department undergone an organizational change (e.g., new employees, new responsibilities, changes in structure)? If yes, what has changed?
3. What are the differences between what is represented in the process models and how the business processes are currently being executed?
4. What were the reasons for the changes?

To ensure the understanding of the domain experts in the third question, we displayed to them the previous process models and described their execution step-by-step according to what was shown in them. Any identified differences could have represented a change in how the department works during that process. Alternatively, they could have also represented a lack of compliance with the business process. To establish if that was the case, we have explicitly verified with the domain experts whether the current behavior is intended and thus does represent a change.

After these changes were identified, we created updated process models with those changes using BPMN. Table 5.1 lists the 25 business processes analyzed and the number of process elements and process participants present in the process models. We also list which of these business processes had experienced some change. As shown by the number of process participants, every process was interdepartmental, though most activities belonged to one of the four investigated departments. Incidentally, none of the four departments interacted with each other in these 25 processes. In total, 20 process models were updated.

Due to privacy constraints, we cannot make the process models public. Nevertheless, in Figures 5.2a and 5.2b, we show an anonymized example of a process model before and after it was updated with the changes identified during the interviews. In this example, the domain experts informed us that the activities of one of the lanes were now performed by two different participants (in yellow), and one activity of these lanes was

Table 5.1 – List of all business processes analyzed in our case study.

Deparment	Process	Num. Elements	Num. Participants	Changed?
D1	P1	58	4	Yes
D1	P2	22	3	Yes
D1	P3	30	4	Yes
D1	P4	39	5	Yes
D1	P5	22	3	No
D1	P6	23	3	Yes
D1	P7	7	2	No
D1	P8	13	2	Yes
D1	P9	32	4	Yes
D1	P10	27	3	No
D1	P11	22	3	No
D1	P12	21	4	No
D2	P13	26	5	Yes
D2	P14	15	3	Yes
D2	P15	20	4	Yes
D2	P16	12	3	No
D3	P17	43	3	Yes
D3	P18	47	8	Yes
D3	P19	54	10	Yes
D4	P20	58	8	No
D4	P21	52	5	Yes
D4	P22	30	4	Yes
D4	P23	29	4	No
D4	P24	61	9	No
D4	P25	17	5	Yes

Source: The author.

new (in green). They also informed us that the first activity was now supported by a new information system (in blue) and that there was some restructuring of the control flow (in red).

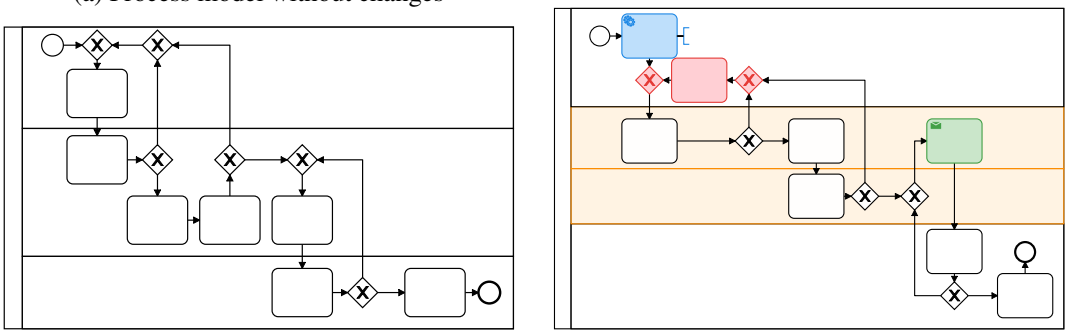
5.2 Analysis of Business Process Changes

We sought to discover and classify the changes that generated our updated process models. As mentioned in Section 3, other studies have analyzed business process changes, particularly within the process redesign phase of the BPM life-cycle. We decided to utilize the framework of Reijers and Mansar (2005) due to its relevance within this area of study, particularly because it focuses more closely on intentional changes to the business processes, instead of momentary differences between process models and

Figure 5.2 – Example of a process model before and after it was updated.

(a) Process model without changes

(b) Process model with changes highlighted.



Source: The author.

business process execution.

However, a slight mismatch exists between their framework's redesign best practices and the changes observed during our investigation. The purpose of redesign best practices is to improve existing business processes by finding opportunities for optimization and advising how to change the business process. In comparison, our analysis seeks to classify past business process changes that may or may not have caused an improvement. These changes may have happened due to a change of requirements, demanding the business process to complete additional or altered tasks. Thus, the changes within our investigation may have worsened the business process in terms of time, cost, quality, or flexibility.

Nevertheless, we still decided to use the seven classes established by the framework of Reijers and Mansar (2005), as their definitions were easily expanded to consider both types of business process changes, i.e., those that improve the business process or those that do not. To do this expansion of definition, we substituted mentions of "improvement" and "best practices" with the mentions of "change". The seven classes of business process changes and their definitions are:

- *Customers*, which focuses on changes regarding contact with customers.
- *Business process operation*, which focuses on how to implement the business process.
- *Business process behavior*, which focuses on when the business process activities are executed.
- *Organization*, which considers the organization's structure and the people and resources involved.

Table 5.2 – Summary of our analysis, showing the classification of changes according to the business processes analyzed.

Change class	Processes
Customers	P03, P09, P16, P18, P22.
Business process operation	P02, P03, P05, P11, P12, P13, P14, P15, P16, P18, P19, P22, P24, P25.
Business process behavior	P08, P09, P11, P13, P14, P16, P18, P19, P24, P25.
Organization	P05, P14, P16, P17, P18, P19, P20, P21, P22, P23, P24, P25.
Information	P02, P03, P08, P11, P13, P14, P15, P16, P18, P19, P24.
Technology	P03, P11, P12, P14, P15, P16, P18, P19.
External environment	P03, P22, P25.

Source: The author.

- *Information*, which describes changes related to the information the business process uses, creates, may use or may create.
- *Technology*, which describes changes related to the technologies the business processes uses or may use.
- *External environment*, which contains changes regarding the collaboration and communication with third parties.

To classify the changes discovered during our revision interviews, we have compared the two versions of each process model analyzed (the outdated and the updated version) and considered the information given by the domain experts regarding those changes. In Table 5.2, we present our classification of changes of each analyzed process model.

We were able to identify a few important similarities between the discovered business process changes. For instance, many processes received significant changes to their process participants. As we mentioned in Section 5, the largest change in this regard was that two domain experts had replaced the previous members of the department's $D3$ and $D4$. As such, some departments' processes received some changes regarding what activities were being performed and by whom. Additionally, both departments were previously the same department within the institute, with both members sharing the same responsibilities and set of processes. After this department was split into $D3$ and $D4$, their processes and process models had to be adjusted to this new structure and the two domain experts' split responsibilities. Most notably, the processes regarding the institute's professors ($P16 - P21$) stayed with the $D3$ department, while processes regarding graduate students ($P22 - P25$) were attributed to the $D4$ department.

Two processes were changed to include new activities for new process participants. $P05$ is a process that weekly controls if the students receiving scholarships are still actively working. Its process model received a new lane representing the coordinator of the post-graduate program being informed of any students that have not justified their absence. Process $P14$, which deals with the institute's creation and approval of new projects, also received a new participant to evaluate the projects.

In addition to internal organization changes, some processes had their communication with customers and external third parties altered. For example, the process model of $P03$, which implements student scholarship requests, was changed to have different activities for each of the two funding agencies providing the scholarship. These activities differentiate the method and the quantity of information that is sent to each agency. With this change, more messages began being exchanged between the university and the student requesting a scholarship. Similarly, $P16$ and $P18$ are processes that control the career progression requests of professors. Both processes were changed to allow the professors to send addendums to their requests at a later date.

In other processes, the communication was reordered or reduced. In the case of processes $P09$, which handles a student's special enrollment, and $P25$, which administers freshman students' enrollment, both had some of their message activities removed or reordered to improve process flow. Similarly, process $P22$, which promotes internship opportunities to the students, changed how they were received and to whom they were publicized. Furthermore, process $P15$, which controls contract processing, heavily relied on multiple messages between parties, which were changed to improve collaboration. Finally, in process $P08$, which controls publishing a report to a federal government agency, the data of this report was batched to be sent all at once, while previously, this data was sent multiple times in parts.

Another common change we observed is how process information was handled by the business processes. Some changes removed the necessity of creating and sending physical documents to different process participants. Occasionally this was done by incorporating a new information system that digitizes those documents and handles their distribution to process participants. This is the case of processes $P03$, $P14$, $P15$, $P16$, $P18$, and $P19$.

In other processes, these documents and the evaluation of their information were removed to simplify the process. One example was process $P14$, which dealt with the institute's creation and approval of new projects and during which many documents are

created and distributed. Digitizing those documents was not possible, so instead, a change was made to either send fewer documents to process participants or skip sending any documents to them altogether when their evaluation was a low priority. Similarly, process $P13$, which deals with collaborative projects with private companies, was slightly simplified in how document generation and distribution were handled.

A few processes changed which information was required to execute the process and how this information was transmitted. We already mentioned how $P08$ batched the information to be sent in its report all at once. In the cases of $P11$ and $P25$, these processes deal with a student's final work in the graduate or post-graduate courses. Both processes were changed to allow the student to send a digital copy of their work directly to the professors for evaluation. These processes were further changed to allow for more flexibility regarding how information was sent and to require additional information from visiting professors. One last minor information-based change happened in process $P02$, which received changes in the authentication data sent to new students.

The last significant type of changes were those related to technology, specifically the use of information systems to help manage the activities of processes. Previously, the analyzed processes were not fully automated, with most of them being fully manual or having only a few activities that interacted with an existing system. After the changes, we observed that more of these processes' activities had a system to support them. One of these processes was $P03$, which we previously mentioned was changed to include different activities for each type of scholarship. This change also resulted in some activities being executed through a new online portal of its agency.

Similarly, processes $P12$, $P14$, $P15$, $P16$, $P18$, and $P19$ were changed because they were dependent on the frequent transfer of several documents and approvals between process participants. Thus, existing university systems were expanded to support some activities of these processes. $P15$ is notable since it contains many process participants and thus benefited greatly from this change.

Considering that the business processes analyzed were not implemented after they were initially modeled, it would have been difficult to detect that the process models were outdated without the analysis we performed in this study. Additionally, most business processes still have manual activities, which limits the detection of changes with this type of analysis.

As a result of our analysis, we have observed a few recurrent types of changes, which we classified using Reijers and Mansar's framework (REIJERS; MANSAR, 2005).

We also note that most business processes have changes classified as *business process operation* and *behavior*, which directly impact the activities performed during those processes and their order. Detecting those changes without event logs and thus without process mining requires observing the execution of the business process through alternative sources. Fortunately, our analysis has also shown that many of our observed changes have impacted the other classes of changes, i.e., *customers*, *organization*, *information*, *technology*, and *external environment*.

5.3 Analysis of Business Process Change Through the Lens of Organizational Change

In this section, we compare the observations made during the update of the process models with the four basic theories presented by Ven and Poole (1995) and the concepts introduced by Weick and Quinn (1999). Then, we discuss how these theories and concepts might be observed in process models and how this may be useful for monitoring business process change.

5.3.1 Comparing the Analyses of the Business Processes to the Theories on Organizational Change

The changes identified in our analyses share one important characteristic: the business processes were changed unexpectedly. The majority of the changes happened as a reaction to new circumstances. Often, these new circumstances were either new systems that had to be incorporated into the business process or they were the new requisites regarding how communication between participants is handled, or what information is processed. It seems clear that the changes observed in our analyses would be classified as episodic since the adaptation to new systems or requirements was sudden and short-run. In addition, since the resulting business processes after the changes were unpredictable, that is, there were no existing instructions to guide how the business processes should have been adapted, the mode of these changes was likely the constructive mode, in which a new significantly different business process is created that is adapted to the new systems and requirements.

As seen in the typology of Ven and Poole (1995), two basic theories have a constructive mode of change: the dialectical theories and the teleological theories. However,

how these theories compare changes identified in our analyses is unclear. The distinguishing characteristic of these theories, the unit of change, may have multiple interpretations. For example, suppose a business process of our analyses is viewed as a single entity. In that case, any changes that have been made to it must have been done to pursue a goal, according to teleological theories. However, business processes are often performed by a collaboration of different process participants who perform the business process' activities (DUMAS et al., 2018). Sometimes, these process participants belong to different organizations or distant departments of one organization, as seen by the processes with changes classified with *External environment* in Table 5.2. There might be no consensus on how a business process must change in these cases since there might be competing interests between process participants. Consensus is one of the motors that cause change according to teleological theories, so the lack of it would imply that a business process that changes in the constructive mode would do so according to dialectical theories. Thus, from the dialectical perspective, a business process's participants are multiple entities with their own priorities. They may have the authority to change their part of the business process according to their methods and goals. Though, when these changes affect other process participants, there will exist a conflict between each participant's priorities. Overall, when changes to a business process are observed in the constructive mode, what defines if these changes happen according to teleological or dialectical theories is if there is a consensus or a conflict of priorities between the process participants.

While applying this knowledge to our analyses, we observed that the interviewed department members, who are active participants in the business processes, rarely expressed that they were part of a consensus on how the business process would change. To them, change usually happened due to a force outside of their departments. From their perspective, change is usually dialectical since they have shown no autonomy to create their own goals. It would only be possible to identify the true reason for the changes we observed by interviewing the other process participant of the business process and discovering why and how a new system was made or a requisite was changed.

Another interesting aspect of the changes identified in our analyses was the scope of the changes. Both dialectical and teleological theories allow for change to happen in a larger scope. The same can be said for episodic changes, which can be characterized by a dramatic shift in paradigm within some work. However, the new systems introduced had small effects from the perspective of the process model elements. The majority of the change in these cases was the automation that is usually hidden within the internal se-

mantics of the process model elements (i.e., what they do in addition to what is defined by the notation). As such, the evaluation of business process changes must always consider these hidden behaviors, even if the process model's structure remains unaltered by those changes.

5.3.2 Discussing Organizational Change and BPM

The comparison between our analyses and the organizational change theories shows that we need to consider the circumstances through which business processes are changed to better control and manage them and their process models. While there are strong similarities between the life-cycle and teleological theories from Ven and Poole (1995) and the BPM life-cycle, the analyses showed that business processes could change outside of the expected phases of this cycle. These observed changes were unexpected to the interviewed process participants, and they contributed to a significantly different version of the business process and process model. As such, the changes presented attributes associated with episodic and constructive change, which would classify them as a dialectical or a teleological change in Van de Ven and Poole's typology. However, most of the observed changes did not match neatly to one single basic theory. Instead, it seems that multiple motors generated a change in most business processes analyzed. Thus, it may be possible that distinguishing how each motor can change a process model may help determine how to monitor when change happens in the execution of the business processes.

Additionally, the comparison showed that business process change is perceived differently based on how many entities one interprets existing in a business process. Through the teleological theory, change happens to the business process as a single entity. However, we also know that this change has to happen through a consensus. Thus, to monitor change according to this theory, it is necessary to identify the possible entities with authority to manage and change the process model. In BPM, this entity may be called the *process owner* (DUMAS et al., 2018). Most of the time, we can expect an organization to have one or more individuals responsible for managing some grouping of business process activities, such as a department chief being responsible for his department's activities. Thus, we can identify how many process owners are involved with a business process by identifying how many organizations collaborate to execute it. Still, it is also possible that multiple process owners exist within each collaborating organization

due to their internal structure or hierarchy. Nevertheless, identifying the process owners allows us to discover and group the fragments of the business process that they have the authority to manage and change. It is also through them that it may be possible to discover which goals are being pursued to improve the business process, according to teleological theory.

Similar to teleological theory, life-cycle theories may also require the identification of the process owners to monitor business process change. While life-cycle theories do not require consensus, there has to be someone to execute the life-cycle with authority to manage the business processes. In addition, understanding which life-cycle is changing a business process may indicate what may change in the process model and when. For example, the BPM life-cycle creates and implements change only after an analysis of the previous cycle's data, so discovering what is being measured and analyzed would help identify what is most likely to change.

When a business process is analyzed through a multiple-entity perspective, such as through dialectical theories, change happens near the activities of the business process where these entities interact with each other since these activities are where conflicts might happen. If the entities are process participants, as observed by our analyses, then the change likely happens to the activities that are near the areas where communication between participants happens. In BPMN, for example, this communication is most clearly seen where message flow elements are sent and received between pools. This change could also happen to the process participants of one organization when the workflow is passed from one participant to the next.

Finally, change according to evolutionary theories was not observed in our analyses, but since these theories also analyze multiple entities, process model change may also be observed where these entities interact. However, in this case, the interaction is competition, not conflict, which means that the business process changes according to what are the best entities to perform a specific task. For example, a seller's business process may provide different payment options, which may be selected based on which provides the most revenue. Thus, there is competition between payment options and the business process has to be adapted to support how an option is performed when it becomes a compelling option. Thus, by identifying this competition's existence, it may be possible to predict when the process elements related to the competing options may change, particularly in the long term since the evolutionary cycle is usually a slow process of changes.

It is noteworthy that the multiple entity perspective aligns better with the per-

Table 5.3 – How business process change may happen according to each basic theory of organizational change

The entities that cause change are...	The time of change is...	What may change are the process elements...
A process owner and a goal.	After goals are set and consensus is achieved.	The process owner has control over and that are associated with the goal.
A process owner and a life-cycle.	In specific phases of the life-cycle.	The process owner has control over and are defined in the life-cycle.
Process participants, data objects and systems.	After one entity changes or forces a change.	Near the areas where the communication between participants happens or that use a data object or system.
Competing options.	When the best competing option changes.	That deal with using the competing options.

Source: The author.

spective of the WST framework and the framework of Reijers and Mansar (2005). Both frameworks present a business process to be connected to components that are fundamental to its execution. These components may not be able to make active decisions, but they can suffer changes that force the business process and its activities to adapt. Thus, calling these components entities from a dialectical theory perspective is not inaccurate. An important feature of these components when viewed as entities is that their changes happening tend to propagate to the business processes. Due to their tangibility, it is more feasible to observe changes to these components than trying to monitor the causes of change in other theories, such as the goal, the life-cycle, or the competition.

In conclusion, from the perspective of this thesis, the most important aspect of monitoring of business process change is identifying the entities that compose a business process. An analysis through the perspective of every basic theory of Van de Ven and Poole's (VEN; POOLE, 1995) typology shows that a process model may be divided into intersecting groups of process model elements, with each group being associated with an entity that may influence and change the elements of that group. These groups also have different causes and methods of change, determined by what type of entity they are associated with. The speed, frequency, and scope of the changes may also be determined by the different cycles of the four basic theories and the mode of change (i.e., prescribed or constructive). We summarize this analysis in Table 5.3, showing the relationship between the main entities involved in a change in the context of BPM, the time when change happens, and what process elements may be changed. Using these concepts, we propose to use the perspective of multiple entities linked with the framework of Reijers and Mansar (2005) to turn components into monitorable entities.

Table 5.4 – Summary of all process models analyzed.

Summary item	Details
Description of the set of process models	• A set of 25 process models created from business processes of an informatics institute of a Brazilian university. • Originally created by students. • Verified and reviewed in conjunction with process analysts and domain experts. • Four university departments perform the majority of the processes activities.
Example domains	• Career progression, management of student scholarships, student enrollment, contract processing, research projects development and approval, report generation.
Process elements	• Between 7 and 61, with an average of 31.4.
Process participants	• Between 2 and 10, with an average of 4.5.
Process models changed	• 20 out of 25.

Source: The author.

5.4 Chapter Summary

In this Chapter, we presented an investigation and analysis of how business process change in practice. Our objective was to understand how business processes change, why these changes happen, and how they affect the respective process models. As such, we attempted to update a set of 25 existing process models in order to compare the outdated and updated versions. We summarize the features of the 25 process models in Table 5.4. To perform this update, we interviewed four domain experts and together we analyzed the process models of their domain to discover what had changed and what had caused those changes.

While analyzing the identified changes, we classified them based on an adaptation of the business process redesign framework presented by Reijers and Mansar (2005). We discovered that many changes involved components of the framework that exist outside the business process, such as customers, process participants, documents, systems, and external parties. We also analyzed the changes through the perspectives presented by the theories of organizational change (VEN; POOLE, 1995). We concluded that identifying the entities capable of causing change provides a means of monitoring those changes. Also, we proposed that the external components of a business process can be considered entities from the multiple-entity perspective, leading us to propose the monitoring of business process change through these components.

6 TAXONOMY OF OBSERVABLE ENTITIES FOR MONITORING BUSINESS PROCESS CHANGES

This chapter presents our approach to achieving the objectives related to hypothesis H_3. Regarding H_3, we believe that analyzing a process model can allow us to discover heterogeneous data sources that would help monitor the business process execution. However, to ensure that this analysis is performed adequately, we need to understand the different types of data sources that may exist within an organization and how they are related to changes happening in a business process. As such, we would use the understanding we achieved from our analyses of how business processes and their process models change, as seen in Sections 5.2 and 5.3.

We have concluded in Section 5.3 that entities are an adequate aspect through which we can evaluate and monitor business process changes. Additionally, the multiple entities per business process perspective, represented primarily by dialectical change theories, provide the most compatible approach to monitoring process model changes using the framework of Reijers and Mansar (2005). Therefore, we aimed to create a taxonomy of *observable entities* related to the change classes of the framework of Reijers and Mansar (2005). We define *observable entities* as the components linked to the business process execution that can be monitored to detect changes. Monitoring observable entities turns them into heterogeneous data sources for identifying business process changes. The taxonomy will define how each type of observable entity can be connected to a type of business process change. We may use the links between observable entities and business processes to detect when the respective process models must be updated.

To evaluate the usefulness of this taxonomy, we applied it to real process models to determine how the identification of observable entities performs and how much of a process model can be connected to these entities. To do this, we designed an approach to create a dataset of process elements classified by human participants. The resulting dataset was evaluated statistically to find patterns between the semantics of the process elements and each of the taxonomy classes. This dataset will be further explored in chapter 7 for training an automated classifier through machine learning algorithms.

6.1 Defining a Taxonomy of Observable Entity Groups

One of the main conclusions of section 5.2 was that process model changes are rarely caused by changes related only to the *business process operation* and *behavior*. It was often the case that most changes could be connected to classes of change related to *customers*, *organization*, *information*, *technology*, and *external environment*.

These five classes may provide us with data sources to detect business process change without relying on event logs. This is because they represent entities that exist separately from the business process but that are also necessary for the execution of its activities and events. Generally, these entities can be considered part of an organization's infrastructure. For example, the payment machines and systems required to execute a "Process payment" activity exist within the context of the *technology* class, such that changes in those machines and systems may propagate into changes to the business process and its process model.

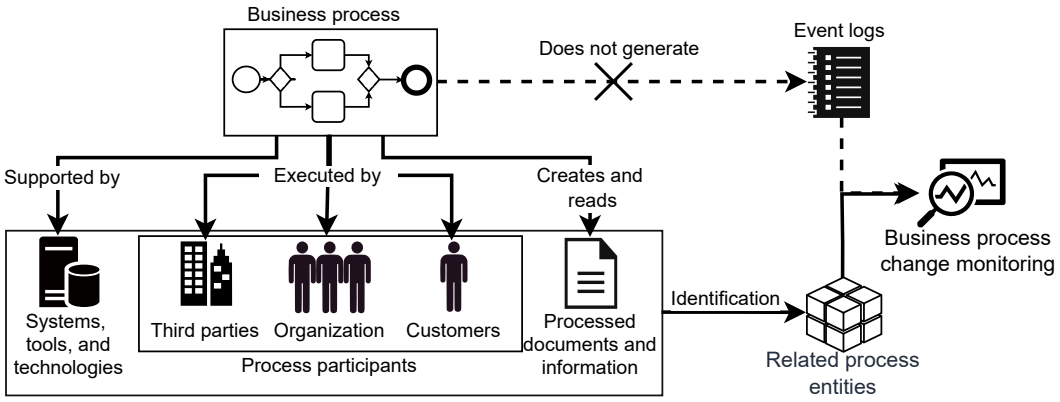
Compared to these classes, changes in the *business process operation* and *behavior* classes cannot be detected without monitoring the business process itself through event logs. These two classes are related to changes to the implementation of business process elements (i.e., activities, events, gateways) and the order dependency between them. The implementation and the order are internal concepts of the business process. As such, changes to them are only observable if external components are linked to those process elements involved in their changes.

Therefore, we propose using the entities related to those five classes to determine the *observable entities* related to business process changes. For example, in our analyzed processes, most changes attributed to the *technology* class came from new information systems that started to support the execution of certain activities. If we establish a way to monitor these systems, we can detect which business processes need to be updated when this system is altered or discontinued. The way to monitor these systems may vary, but determining which business processes and activities require a system allow us to create a link between them that can be traced whenever one of them changes in some aspect.

Hence, we define a taxonomy with three *observable entity* groups based on classes of analyzed changes:

- *Systems, tools, and technologies (STT)*, which groups technologies that support and enhance the execution of process activities;
- *Processed documents and information (PDI)*, which represents the data that is cre-

Figure 6.1 – Overview of how observable entities can be used for monitoring business process changes when event logs are unavailable.



Source: The author.

ated or read during the process execution;

- *Process participants (PP)*, which groups customers, third parties, and the organization executing the process, including the organization's structure and its people.

We chose to link all three classes *customer*, *organization*, and *external environment* to the single entity group *process participant* because all of them similarly involve the people that execute a business process activities. In Figure 6.1, we present an overview of how these entity groups relate to a business process and the monitoring of its changes.

Table 6.1 presents the relation between the three entity groups and how business processes change. We have defined those relationships based on our analysis of the business processes in Section 5.2. Each relationship was observed in at least one of the business processes analyzed. While they exemplify how a change in the entity can change the business process, we note that the presented relationships are likely not complete, so there can be other ways these entities can affect changes in the business process. Additionally, we emphasize that the complexity of monitoring the examples of business process changes presented in this Table varies significantly. For instance, it is difficult to conceive how to monitor a new system being added to a business process since it is unknowable. Comparatively, detecting if an existing system was discontinued is less complex and a more straightforward application of the monitoring of business process changes.

6.2 Evaluating the Taxonomy

To evaluate our proposed taxonomy, we must first determine how to apply it. One of our objectives is to use the relationships between the observable entities and the process

Table 6.1 – Taxonomy of entity groups and examples of how they can be related to a process model change.

Entity groups	How they can be related to business process change?
System, tools, and technologies	• A new system became required to perform new activities. • A system was introduced to support existing activities. • A change in a system alters the semantic of existing activities. • A system ceases being used or was discontinued. • A system is being used differently than before.
Processed documents and information	• Receiving a document starts being required by some activity. • Some activity starts producing new documents. • The contents of the documents read or produced during the process have changed.
Process participants	• The person executing the process has left the organization or was replaced. • The person executing the process has changed roles within the organization. • The structure of the organization's departments and members has changed. • A process participant was added or removed from the process. • Messages between participants have been added, removed, or their contents were changed. • The order of the messages between participants has changed.

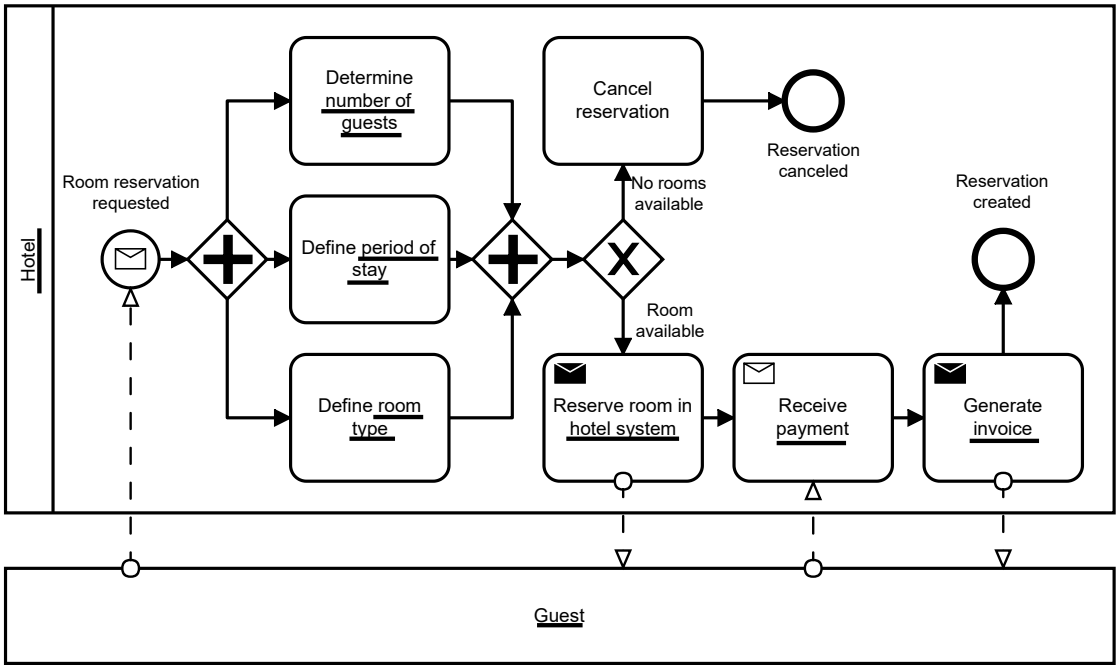
Source: The author.

models to create a new process monitoring approach. One of the steps of this approach is determining a method to identify the observable entities of a business process. Ideally, this identification would be realized during the modeling and implementation phases of the BPM life-cycle. Process analysts and domain experts can achieve a more comprehensive understanding of a business process during those phases since they have more relevant and up-to-date information at those moments. Comparatively, the resulting process models of those phases have a reduced scope of information available.

Nevertheless, we propose to create a method to identify observable entities based on the information provided within process models. In cases in which one needs to evaluate the implementability of old process models, such a method may be a valuable solution. As mentioned in Section 6.1, observable entities are closely related to the infrastructure of an organization. Thus, by identifying these entities and matching them with an organization's infrastructure, it is possible to determine if this organization currently lacks the necessary components to execute the process model and show which process elements need to be updated to match the current infrastructure.

One of the valuable features of BPMN is that it already provides a notation for

Figure 6.2 – Example of an identification of entities from a process model by analyzing the labels of process elements.



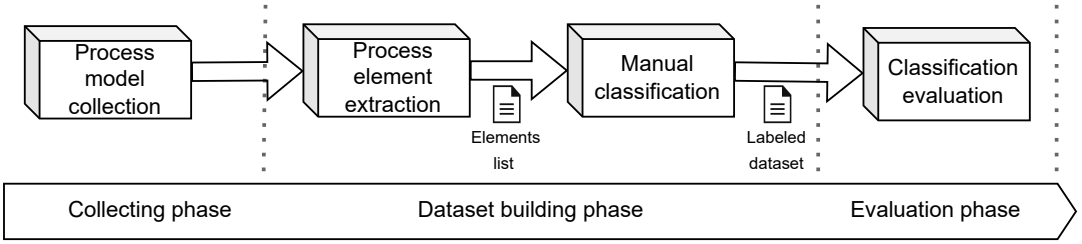
Source: The author.

identifying entities of a business process. Process participants, for example, are an integral part of BPMN collaboration models, represented by pool and lane elements. The data object element also frequently represents the documents and data systems utilized during the business process' execution. Unfortunately, utilizing those elements is not required by the notation, and process modelers can omit those elements if they want. As we have seen in our analysis of BPMN models, process modelers may lean their focus toward the operation and behavior aspects of a business process, thus neglecting to consider the data and resources utilized during its execution.

Even if those aspects are neglected, they may still be implied within the labels of the other process elements in a process model. For example, in Figure 6.2, we present a process model for room reservation in a hotel, in which we underlined the words that could indicate an entity present in this business process. For instance, *number of guests*, *period of stay*, and *room type* can be considered information entities acquired during this process. Similarly, *hotel system* definitely evidences the presence of a system to manage hotel rooms.

Thus, to evaluate the usefulness of our taxonomy, we can use this idea of analyzing individual process elements to identify if they contain information related to any of the three taxonomy classes. To do so, we performed an experiment in which two human users

Figure 6.3 – Our approach to create our dataset of classified process elements



Source: The author.

analyzed a set of process elements from real process models to classify each element in each of the three classes. This classification aimed to explore how many elements of each model could be related to one of the taxonomy classes and verify if both human classifiers could reach some consensus regarding their classifications. The final result of these classifications is a dataset of classified process elements.

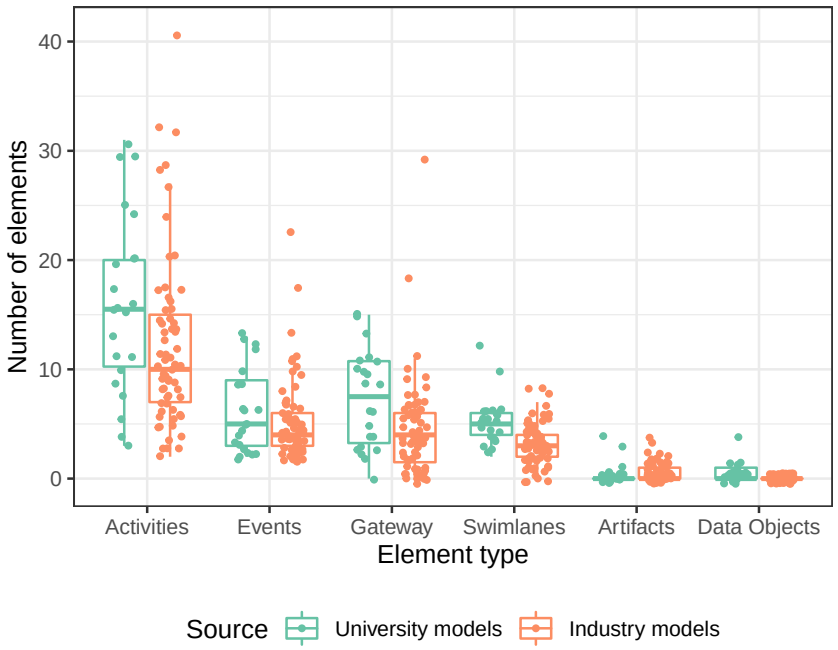
6.2.1 Building a Dataset of Classified Process Model Elements Based on Our Taxonomy

To build a dataset of classified process model elements, we developed an approach with three phases, which can be seen in Figure 6.3. In the *Collecting* phase, we define the sources of the process models used to build our training dataset and the benefits of using different sources. In the *Dataset building* phase, we analyze the .bpmn format, in which BPMN diagrams are saved, and the categories in which we aim to classify the process elements. We also show the development of a method and the tools to create our training dataset, including the extraction of all the process elements, the filtering of this data, and the manual classification of each process element. This classification is analyzed in the *evaluation* phase to remove inconsistent classifications and process elements with errors.

We have selected models from two primary sources to assemble a collection of process models in the *Collecting* phase. One source was the set of 25 process model analyzed in Chapter 5. Specifically, we used the updated versions of those process models, as they had higher quality than the original versions created by students and they were verified with process modeling guidelines (AVILA et al., 2020). Another source was a BPM consulting company which allowed us access to 63 models. Like the previous collection, these process models were also created in Portuguese at first before they were translated into English by an employee of the consulting company.

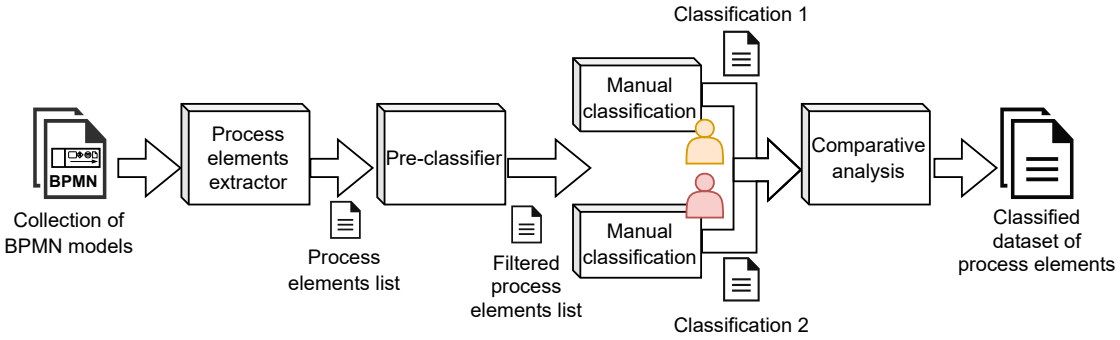
Process models made by different process analysts and from entirely different con-

Figure 6.4 – Distribution of the amount process elements per process element type.



Source: The author.

Figure 6.5 – Our sequence of steps for extracting process elements from BPMN models and creating our classified dataset of process elements.



Source: The author.

texts increase our data's robustness. However, vocabulary variability also increases. Our collection has 88 process models with varying sizes and structures. In Figure 6.4, a box-plot shows the distribution of the number of process elements of the process models based on their type. It is also possible to compare this distribution between the two assembled collections.

After we defined the sources for our collection process models, we needed to determine what were the tools necessary to create our classified dataset of process elements. As seen in Figure 6.5, we first extracted the individual elements from each process model. To perform this extraction, we created a process element extractor tool to generate a list of process elements automatically. Not all elements from this list were fit to be a part of a

classified dataset since many elements had no labels, duplicate labels, or labels with Portuguese words. Thus, we also created an automatic pre-classifier to filter these undesired elements to reduce the workload of the manual classification. To complete this phase of our approach, we created a tool for manual classification that allowed multiple people to evaluate many process elements in multiple sessions swiftly. In this tool, each process element would be presented individually, in a random order, without the context of its process model.

The final result is the dataset of classified process elements. Each process element in this dataset contains information defining from which process model it originated from, as well as all other semantic information of that element such as its label, types, and subtypes. It also contains the values of the classification given by the manual classifiers for each of the three classes: STT, PDI, and PP.

All three tools used to create this dataset can be found on Github[1]. Screenshots of the interfaces of these tools can be seen in Appendix C. The lists of classified process elements will be evaluated in the next phase of the methodology.

To create the process element extractor, we needed to consider the structure of the files containing the process models. Two popular modeling tools were used to create the process models of our collection. The university process models were created with the "Camunda Modeler," while the process models from the industry were created with the "Bizagi Modeler." The choice of process modeling tool can impact the resulting model's file because, even though the BPMN notation is the same, there are different ways for tools to build a process model both graphically and internally in the code. In our case, the output of both tools was a .bpmn file, which has a generally consistent structure with minor variations across different tools.

The .bpmn file is an XML-based file structured in a hierarchical format. When analyzing this hierarchy, our first point of interest was the collaboration tag, which is unique for each file and contains the registry of all the swimlanes and artifacts. At the same hierarchy level, each swimlane has a process tag, in which all the process elements related to the swimlanes are allocated. Within these tags are attributes containing all information related to those elements, including the subtypes and the textual information. It also includes the positioning coordinates, though this information is not relevant to this work.

Using the Beautiful Soup library for Python, we developed an extractor capable of

[1]<https://github.com/diegotavila/SBSI-ExtractAndClassifyProcessElements>

navigating through all the tags in the .bpmn files and fetching every process element and their associated information. The extractor generates a list in which each item represents one element. From the 88 process models in our collection, we have extracted 4606 process elements.

Of these 4606 process elements, not all were suitable for our classifier. We have applied a series of filters through a pre-classifier tool to remove those unsuitable elements. The first filter discarded all connecting objects (sequence flows, message flows, associations) that, while essential for visually understanding the order of execution of the process, have no meaning when appearing out of context.

The second filter removed all elements without textual labels. The main focus of our classification is the information present in these labels, so their absence requires the discarding of the process element.

The third filter removed all duplicate elements, i.e., identical element types and labels. Because we aimed to evaluate each element individually, with no context, identical process elements should theoretically be classified in the same way every time. Thus, discarding duplicate process elements would reduce the total number of process elements to be classified with no significant loss of training data.

The fourth and final filter considered the quality of the textual labels of every process element. For example, some labels had orthography problems or contained non-English words, such as the original untranslated Portuguese. As such, we used the API of the Google translator library to detect the language and filter out all process elements with non-English words. After this filter, the total number of process elements was 1329. Unfortunately, this filter alone was imperfect, so we complemented it during the next step of our methodology by instructing our classifiers to manually verify and flag all process elements with non-English labels or orthography errors.

After extracting and filtering all 1329 process elements, it was of vital importance that the process of manual classification was rigorous and consistent. Unfortunately, the high number of process elements to classify and the subjectivity of the classification made achieving this difficult. As such, a few considerations were made about the classification methodology to ensure good results.

The first consideration was how many individuals would perform the manual classification. Having multiple human classifiers would give us some redundancy within the data, from which we could remove some of the subjectivity of the individuals. However, when we were creating the pilot design for the classification process, we noticed

that a significant amount of time and effort was necessary to classify all 1329 process elements. While the first few classifications could be performed quickly, exhaustion eventually would impair the classifier. We extrapolated that multiple breaks would be necessary before the classification was completed. Considering these circumstances, only two classifiers were able to perform this task. The first classifier had more than five years of experience using BPMN, while the second was a beginner with six months of study. Both classifiers completed their tasks within four classification sessions. Before the first of these sessions, the classifiers discussed with each other about the definitions of each category.

The second consideration was the tool used by the classifiers to evaluate all process elements. Since we knew that classifying all 1329 process elements would be a long-term task, we developed a tool that would allow the classifier to track their progress, save partial results, and resume from where they left off.

The third consideration was how each process element would be classified in our classifier tool. We chose to classify each of the 1329 elements individually, separated from their process models. This method ensures that the individual performing the classification must only consider the information contained in that process element. A benefit of this classifying method is its simplicity for both the human and machine classifiers.

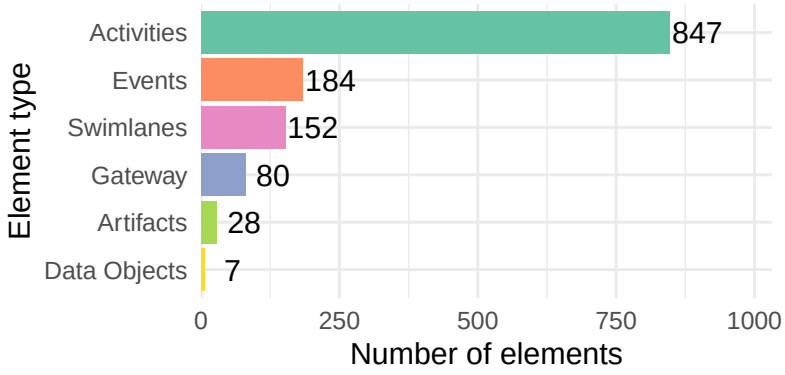
Each process element was presented graphically to the classifiers, using the appropriate BPMN notation, so that all attributes (textual information, element types, etc.) could be easily identified. The classifiers evaluate whether the process element belongs to each of the three classes according to a 5-point Likert scale, from "*1 - Strongly disagree*" to "*5 - Strongly agree*". We used a Likert scale to give us some granularity to evaluate the classifications.

6.2.2 Analysing the Dataset

The final step to complete the creation of our dataset is to analyze and evaluate the results. We began our analysis by examining the process elements that passed our filtering requirements. As we explained in Section 6.2.1, the classification process presented the option to manually flag process elements with non-English words or orthography problems. Of the 1329 classified process elements, 60 were removed in this way.

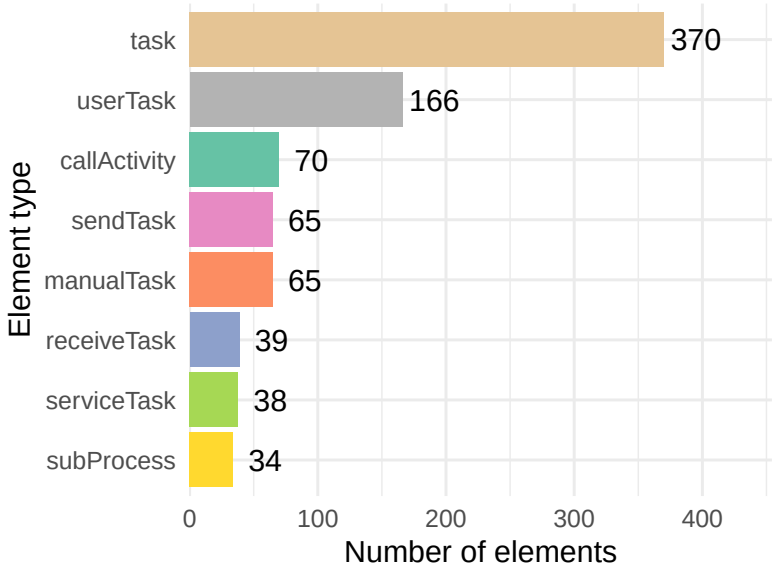
Regarding the process element types, *activities* were the most common, as seen in Figure 6.6, followed by *events* and *swimlanes*. *Artifacts* and *data objects* were rare since

Figure 6.6 – Count of the number of classified elements per type.



Source: The author.

Figure 6.7 – Count of the number of classified activities per type.

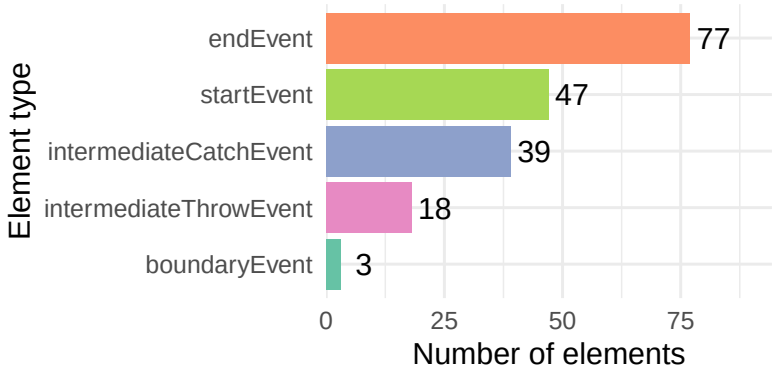


Source: The author.

they were not used frequently in our collection of process models, which is fairly expected since these elements are optional and do not describe the workflow of the business process. *Gateways* were also uncommon in this dataset, being almost entirely composed of *exclusive gateways*. The alternative, *parallel gateways*, are generally not labeled in process models, and thus they were filtered by our pre-classifer step.

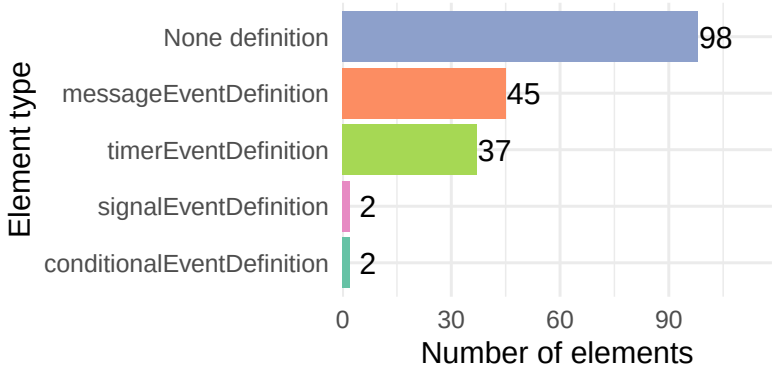
Regarding *activities*, the dataset shows a decent variety of sub-types. As shown in Figure 6.7, untyped tasks are most common, though it is important to highlight that all other types, except for *subprocesses* and *call activities*, may be strongly connected to our three taxonomy classes. Additionally, 18 activities have parallel instance task markers, which indicate that said task creates multiple instances to be executed in parallel. Similarly to *activities*, the dataset contains a decent distribution of *events* types, as depicted

Figure 6.8 – Count of the number of classified events per workflow position.



Source: The author.

Figure 6.9 – Count of the number of classified events per semantic definition.



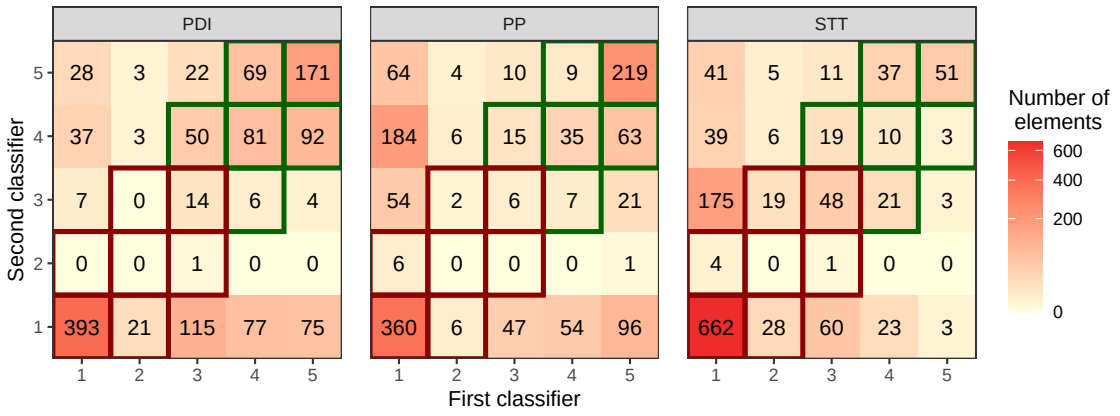
Source: The author.

in Figure 6.8, which shows the workflow position of the events, and Figure 6.9, which shows the semantic definitions.

In the next step of our analysis, we compared the Likert values of each classification list for each category. We illustrate this comparison in Figure 6.10. Each axis represents the classifications of each of the two classifier participants. The tiles are organized according to the 25 possible combinations of Likert scores between the two participants. Each tile shows the number of process elements that received that specific combination of Likert scores. For example, in the PDI category, the tile at position (1,1) shows that 393 process elements were scored 1 by both participants, meaning they both strongly disagree that those elements have entities related to the PDI category.

Based on these tiles, we can see many divergent classifications between the two classifiers. We separated the classified process elements into three sets: *Positive*, *Negative*, and *Inconclusive*. The *Inconclusive* set contains all elements with divergent classifications since they lack a consensus between both classifiers. We defined divergent classifications as those in which the difference between the Likert scores was higher than one. In Figure

Figure 6.10 – Quantitative comparison of the classifications between each classifier for each category



Source: The author.

6.10, this is represented by all tiles without highlighted borders.

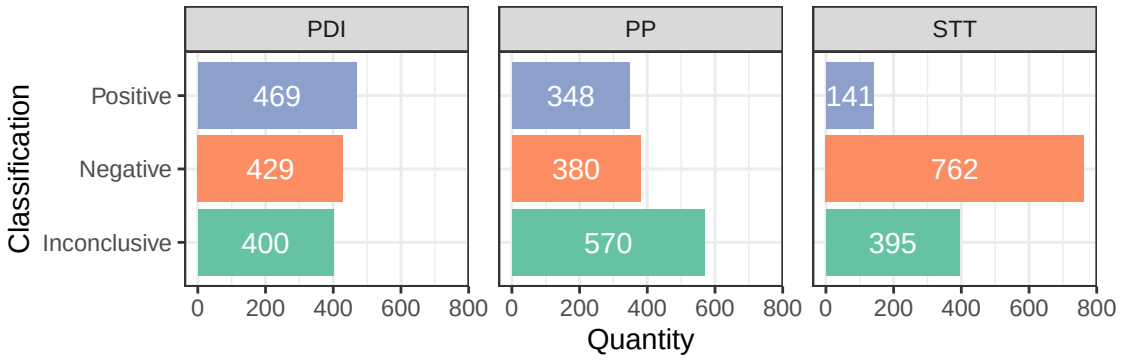
The *Positive* and *Negative* sets contain the non-divergent classifications, in which the difference between the Likert scores was equal to one or lower. They define which process elements would be classified as belonging or not to a category. The process elements which had one of the scores higher than three were placed in the "*Positive*" set, while the rest were placed in the "*Negative*" set. In Figure 6.10, these sets are represented by the tiles highlighted with green and red borders, respectively.

In Figure 6.11, we present the total number of classified elements for each category and each of the three sets: *Positive*, *Negative*, and *Inconclusive*. As can be seen in this Figure, the PDI category had the most balanced classification, with 469 of the 1298 process elements classified into the *Positive* set, 429 into the *Negative* set, and 400 into the *Inconclusive* set. The PP category was closely balanced between the *Positive* and *Negative* sets, containing respectively 348 and 380 process elements. However, it contained the most elements in the *Inconclusive* set, totaling 570 process elements. Finally, the STT category had only 141 elements in the *Positive* set, the fewest of the three categories. This category also has 762 in the *Negative* set, the highest of the three categories. These results show that STT-related observable entities are underrepresented in our dataset.

After the definition of the *Positive* and *Negative*, we were interested in analyzing if there were process element types closely related to one of our taxonomy categories. As Figure 6.12 shows, many of the element types are significantly unbalanced when comparing the *positive* and *negative* classifications. For example, all data objects and swimlanes elements were consistently classified in the PDI and PP categories, respectively.

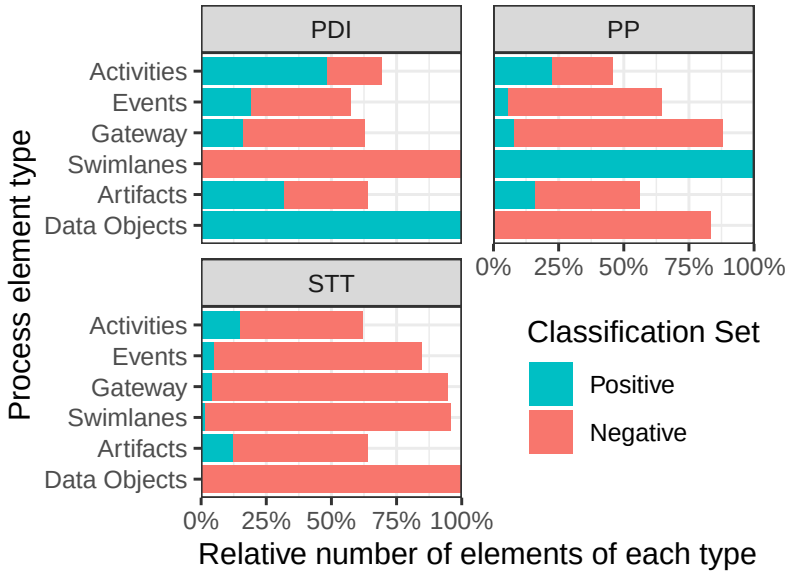
To evaluate the extent to which our method can identify process elements related to

Figure 6.11 – Cout of classified elements for each category.



Source: The author.

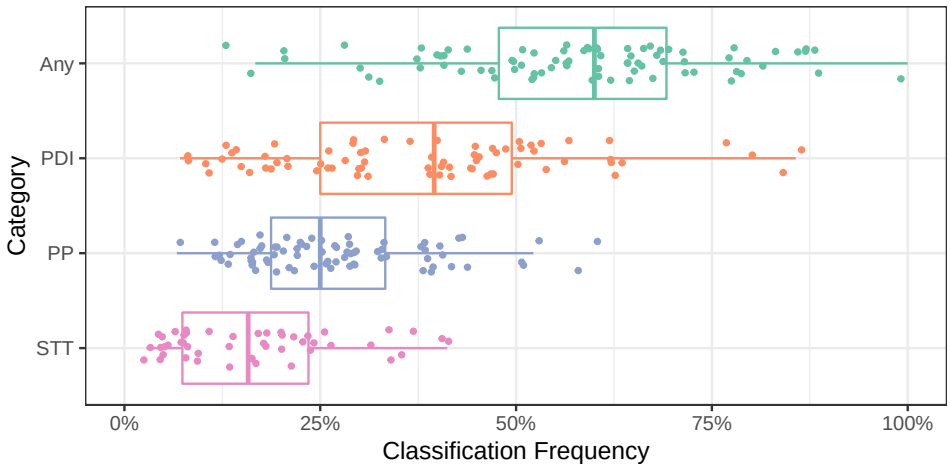
Figure 6.12 – Relative distribution of how many process elements exist within each classification set based on type.



Source: The author.

observable entities within existing process models, we used the results of our classification experiment to measure the classification frequency, i.e., the number of process elements that were classified positively in any of the three taxonomy classes. This measure defines the coverage of our proposed taxonomy, which allows us to evaluate its ability to identify observable entities that can be used for detecting business process changes. To measure these frequencies, we grouped the 1298 process elements of our dataset according to which process models they originated from. Then, for each group, we calculated the ratio between the number of classified elements in the *Positive* set and the total number of process elements of that group. We calculated this measure for each category. We also calculated this measure for process elements classified in any of the three taxonomy categories.

Figure 6.13 – Measure of the process model coverage by the observable entities identified in our classified dataset.



Source: The author.

In Figure 6.13, we present boxplots of the classification frequencies for the process models of our dataset for each category. In this Figure, every dot within one of the rows represents one process model of our collection. As can be seen in the Figure, in all three categories, the third quartile of classification frequency does not reach 50%. As such, for the individual categories, it is generally unlikely for them to be able to cover, and thus monitor, more than half the elements of any process model. When combining the process element classification of the three categories, the process model coverage significantly improves. In this case, the average classification frequency was 57%, though there was a large variance demonstrating some process models with less than 25% and more than 75% classified process elements.

While these frequencies may improve by refining the classification of process elements, such as solving the problem of divergent classifications, there is likely a limit to how many process elements can be classified by our taxonomy for any process model. Many process elements do not contain any semantic information that could identify a related observable entity. This lack of information may be a flaw in the modeling of that business process. It needs to be accepted that some process models will naturally have low classification frequencies. Nevertheless, it is perhaps more critical to monitor positively classified process elements since, in addition to being susceptible to changes regarding the business process operation like any other process element, they are sensitive to changes to their linked observable entity.

6.3 Chapter Summary

In this Chapter, we presented a taxonomy of observable entities. Observable entities are the means through which we propose monitoring business process change without relying on event logs. The taxonomy was defined based on our investigation of business process change in practice and the business process redesign framework of Reijers and Mansar (2005). It contains three classes: *process participants*; *systems, tools, and technologies*; and *processed documents and information*. For each of these classes, we presented examples of how they may be related to different causes of business process change.

To create a framework for detecting business process change, we proposed applying this taxonomy in the identification of observable entities within process models. We argued that process models frequently contain information regarding observable entities based on the semantics of process elements, including the elements types, sub-types, and labels. Thus, classifying process elements became our main task to validate the definition of the taxonomy.

Two human participants realized a manual classification of 1329 process elements originating from 88 industry process models. A set of tools was developed to extract process elements from the process models, filter elements that were irrelevant to our taxonomy or contain errors (duplicate elements, elements without labels, elements with non-English labels), and coordinate the classification process (including the display of process elements and the tracking the classifiers' progress). During the classification process, the participants analyzed each process element separately from its context and in the three taxonomy classes. We assessed the results of the classification for each taxonomy class, dividing the process elements into *positive*, *negative*, and *inconclusive* sets. The inconclusive set contained all elements in which the difference between classification scores was higher than 1 on a 5-point Likert scale. The worst of the three inconclusive sets (one for each taxonomy class) contained 570 process elements. We also evaluated the process element classification frequency of the process models. On average, our set of process models had a classification frequency of 57% in any given class, showing it is reasonable to expect that at least half of any process model can be monitored through observable entities. In Chapter 7, the dataset of classified process elements is used to train an automated classifier using machine learning.

7 TRAINING AN AUTOMATED CLASSIFIER OF PROCESS MODEL ELEMENTS

In the previous Chapter, we created a dataset of classified process elements based on our three taxonomy categories. We presented this dataset as an example of how a manual identification of observable entities can be performed by analyzing the information already present within process models. In order to improve the identification method and provide further evidence towards validating hypotheses H_3, we propose using our dataset to train an automated classifier utilizing machine learning algorithms.

There were two main concerns when training our automated classifiers. The first concern was preprocessing our dataset to make it suitable for machine learning algorithms. We needed to determine which process elements would be used for training and which variables from our dataset would be entered into the algorithms. The second concern was choosing which machine learning algorithms would be used, implementing them, and implementing the proper methods to evaluate the results. Sections 7.1 and 7.2 address these concerns. Sections 7.3 and 7.4 present an analysis and discussion of the results of our automated classifiers.

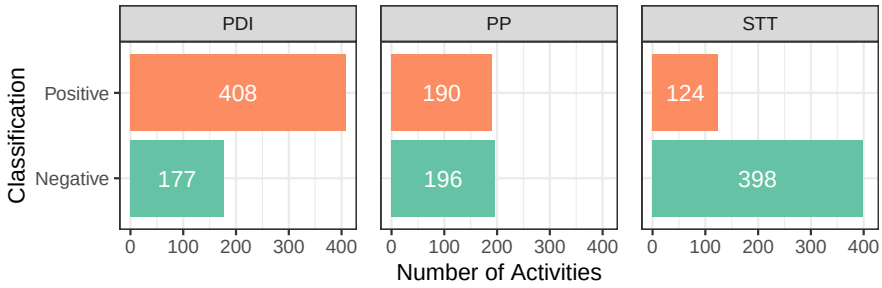
7.1 Preprocessing the Training Dataset

To prepare our dataset of machine learning training, we need to consider its composition. To begin, every item (or row) of our dataset represents one process element and its classification according to one of the taxonomy categories. In addition to a unique identifier, every item contains:

- The label of the process element (e.g., "Generate invoice").
- The element type (e.g., activity, event, gateway, etc.).
- The element sub-types.
- The classification category (PDI, PP, STT).
- The classification result (*Positive*, *Negative*, *Inconclusive*).

Knowing the general structure of our dataset of process elements, we divided our classification problem into three separate single-class classification problems according to the three categories of infrastructure components. Our first concern when evaluating our dataset was choosing which items would be used for training the machine learning algorithms. According to our analysis in Section 6.2.2, some process element types (data

Figure 7.1 – Distribution of process activities present in each taxonomy category.



Source: The author.

objects, artifacts, gateways) had few classified elements (see Figure 6.6). Additionally, when evaluating the classification results according to the element types (see Figure 6.12, we saw that swimlanes and events had an imbalanced distribution between the *Positive* and *Negative* sets. Because of these results, we chose to focus on training our classifier on activities, which are the most numerous in our dataset and it was almost balanced in the classifications. Creating an automated classifier based on activities may also be more useful since it is more prevalent for them to have textual labels compared to other process element types. Thus, our training dataset had 847 process elements, and every element was an activity. In Figure 7.1, we show the number of activities in the *Positive* and *Negative* sets for each category, ignoring those in the *Inconclusive* set.

To further improve the training dataset, we performed another preprocessing step to balance the dataset in the three categories. Although the activities in the PP category were close to the ideal 50:50 balance (as seen in Figure 6.12), the PDI and STT categories presented some dominance in Positive (70.8%), and Negative (76.1%) sets, respectively. To keep the symmetry between our three ML models, we balanced the three datasets by oversampling the minority class through random duplication of elements from the minority group.

The next preprocessing step depended on determining the input variables. Since our training will focus only on activities, we chose to train our algorithms, at first, only on the process elements' textual information. This decision may reduce our overall predictive capability since an element's sub-types could have vital information for learning and predicting its classification. Nevertheless, developing the text-only approach first may reveal if it is feasible to have good classification performance overall since this attribute produces the most variety between process elements. It also lets us analyze how a machine learning model learns from process model texts. One noteworthy factor regarding activity labels is that common convention establishes that they should begin with a verb in

infinitive form followed by the object (MENDLING, 2013; AVILA et al., 2020) and that labels are often short to fit within the boundaries of the process element in the diagram. These characteristics may influence how the machine learning techniques are applied and may affect the classifier's results.

Since we were working with texts, we decided to implement three common text preprocessing techniques to clean and normalize our data: conversion to lower-case, lemmatization, and removal of stop-words. The implementations of these techniques were done using functions available in the spaCy library. On average, for every 1000 words, the lemmatization function modified 111 words, and 134 words were removed as stop-words. We also wanted to test if lemmatization and removal of stop-words had significant effects on the results of the automated classifier, so they were implemented as options in our algorithms. We considered four scenarios when training the automated classifier: original (lower-case) text, lemmatization, stop-words, and lemmatization + stop words.

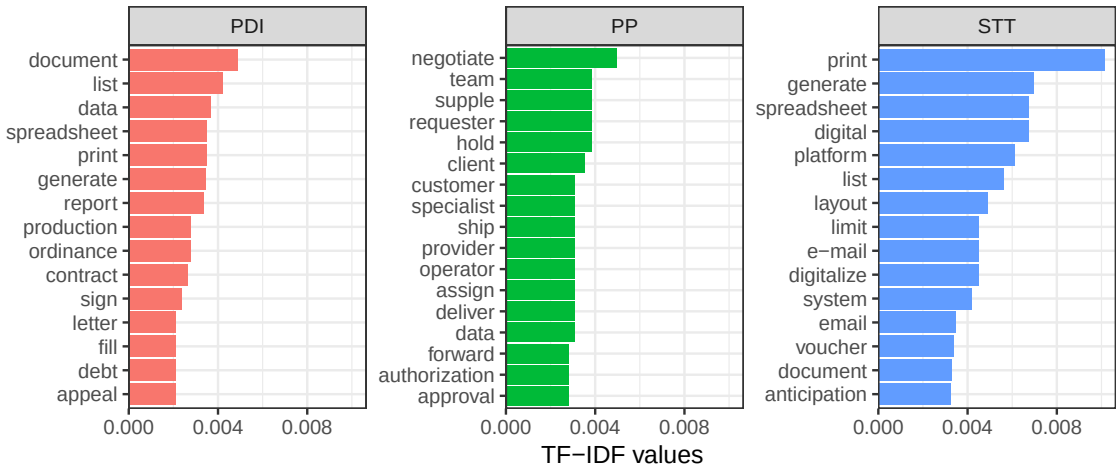
Our final preprocessing step is feature extraction, that is, transforming the raw textual data of our dataset into meaningful data to be used as input for our machine learning algorithms. We tried converting our raw textual data into a numeric representation format by applying two techniques: the Bag of Words (BoW) and the TF-IDF (FACELI et al., 2021; JURAFSKY; MARTIN, 2009). Our applications of these two techniques are very similar since both count the term frequency in each process element. So, we also wanted to evaluate if there was a significant difference between the effects of these techniques on our results. To give context into which words were most meaningful in our dataset, figure 7.2 shows a list of the top 15 words of process elements classified as *Positive* for each category based on the TF-IDF.

7.2 Training the Machine Learning Algorithms

There were three main factors considered in the training of our automated classifier. The first factor was how our text data would be represented using n-grams. The second factor was how we would ensure the robustness of our results using cross-validation. Lastly, our third factor was determining which machine learning algorithms we would use.

N-grams are an approach that would allow the automated classifier to capture the local context of the textual data to better classify the process elements. Using single words, or 1-gram, may lead to a loss of linguistic meaning since we are not capturing the

Figure 7.2 – List of the 15 words with the highest TF-IDF values in all process tasks classified positively in each category



Source: The author.

relation between the words. The order in which the words appear in a sentence can completely change its meaning. To handle this issue, we chose to use combinations between 1-gram, 2-grams, and 3-grams in our training. Therefore, we evaluated six possible n-grams combinations, which we represent as a range $(a-b)$, where a is the lowest n-gram, and b is the highest: $(1-1)$, $(1-2)$, $(1-3)$, $(2-2)$, $(2-3)$, and $(3-3)$. As such, n-grams $(1-3)$ considers all 1-gram, 2-grams, and 3-grams. We chose to stop at 3-grams because our activities dataset has 3.5 words on average.

To ensure our training and testing procedure is not biased, we used the k-fold cross-validation method. We chose to use $k = 5$, so that our algorithms were trained 5 times, each time with a different subset of our dataset containing 80% of all items. Then, the trained algorithms were validated with the remaining 20%. Our cross-validation method was also stratified, to ensure that every subset contained roughly the same number of items classified as *Positive* and *Negative*. By using 5-fold cross-validation, we avoid relying on a one-time result for evaluating the performance of the classifier. We also ensure that every item of our dataset has been used at least once for training and for testing.

Finally, we chose four machine learning algorithms: the *Complement Naive-Bayes* (CompNB), *Multinominal Naive-Bayes* (MultNB), *Random Forest* (RF), and the *Support Vector Machine* in its classification mode (SVC). In general, the choice of these algorithms was done because of the ease of their implementation using the *Sklearn* library of Python. All four algorithms are suitable for text classification tasks, though some might be better or worse given the size of our dataset and the structure of our texts.

Table 7.1 – Summary of the training steps and their options.

Training Steps	Options	Total
Selected Categories	• Processed Documents and Information • Process Participants • Systems, Tools, and Technologies	3
Text preprocessing	• Lemmatization, • Removal of stop-words	4
Feature extraction	• Bag of words (BoW) • Term Frequency - Inverse Document Frequency (TF-IDF)	2
N-gram text representations	• 1-gram • 2-grams • 3-grams	6
Algorithms	• Complement Naive-Bayes (CNB) • Multinominal NB ǀ(MNB) • Random forest (RF) • Support Vector Machine (SVC)	4

Source: The author.

In Table 7.1, we summarise all preprocessing and training options. By implementing all those options, we obtained results from 576 possible combinations. Our implementations of these algorithms and every option discussed in this section can be found on Github[1].

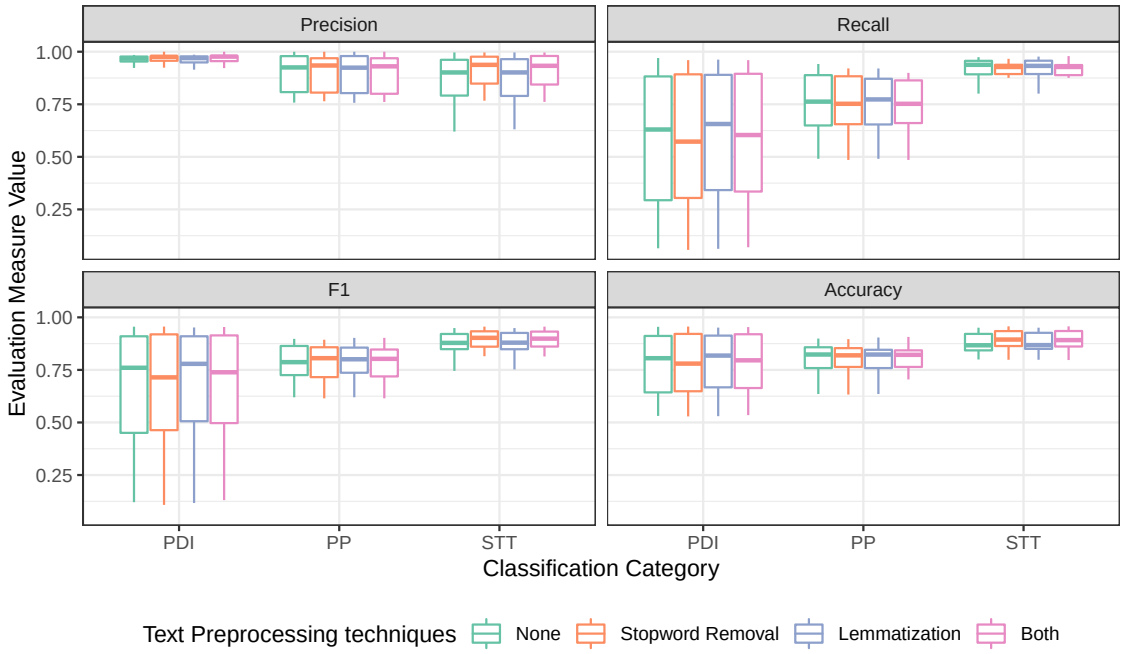
7.3 Analysis of the Training Results

We evaluated the results of our training algorithms using performance metrics, such as precision, recall, accuracy, and the F1-score. We aimed to compare the impact of each of our training options, and the performance of all four machine learning algorithms alongside the six different n-gram ranges.

To assess the impact of text preprocessing techniques on our machine learning algorithms, in Figure 7.3 we compared their performance metrics with and without lemmatization and stop-words removal using a boxplot. We expected these techniques to have a low impact due to the characteristics of the texts inside activity labels. The results in Figure 7.3 also evidence this assessment, making it unclear to determine whether these techniques could improve or hinder the performance of our classifier. Still, considering that these techniques are theoretically helpful, according to the literature on Natural Language Processing (NLP) (JURAFSKY; MARTIN, 2009), and that they simplify the vocabulary evaluated by our feature extraction methods, we opted to continue using them in our evaluations. Thus, all the following analyses from this point were performed based

[1] <https://github.com/diegotavila/SBSI-AutomatedClassification>

Figure 7.3 – Boxplot comparing the impact of the text preprocessing techniques across all possible training algorithms and options



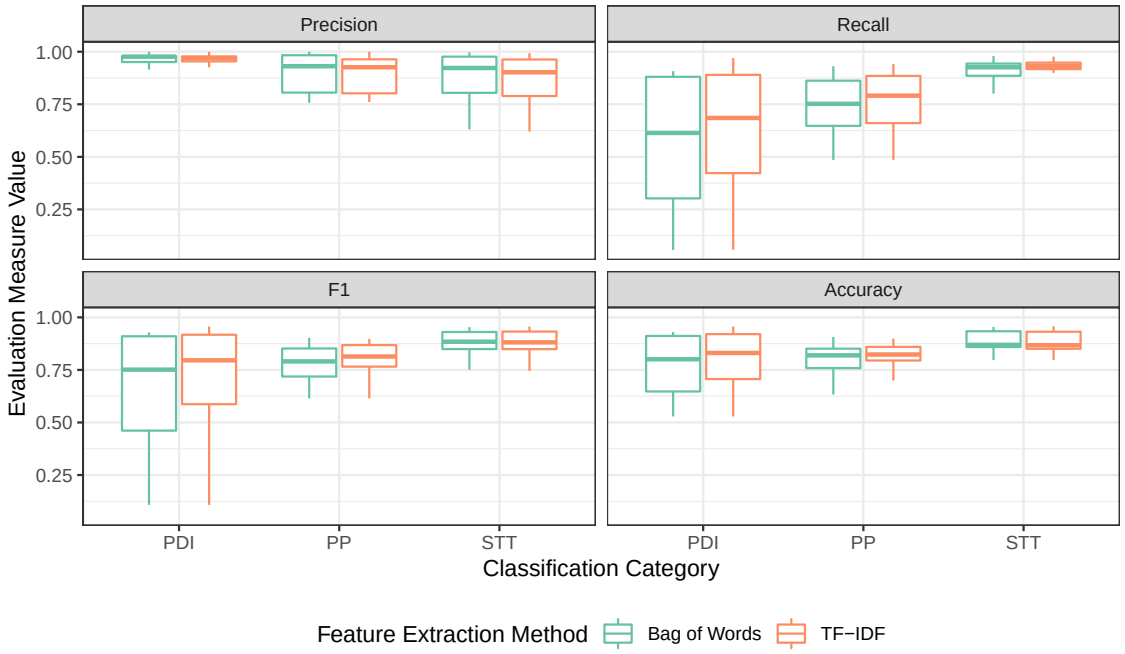
Source: The author.

on the results obtained using lemmatization and stop-words removal.

Regarding the feature extraction methods, we expected an advantage for TF-IDF because it brings more information in a scenario where there are no repeated words in the same text. However, as seen in Figure 7.4, there was little difference in the results according to our evaluation measures. The highest difference was in the recall measure for the PDI category, in which the median for TF-IDF measures roughly 0.07% higher than BoW. Overall, we can assume that TF-IDF method will not perform significantly worse than BoW, thus our subsequent analyses in this section will focus on the training results that use the TF-IDF method.

To compare the results of the combinations between n-gram ranges and machine learning algorithms, we present Figures 7.5, 7.6, 7.7, and 7.8, in which we present the results based on each of our evaluation measures. Starting our analysis with our precision measure in Figure 7.5, we can see that our results favor stricter n-gram ranges, such as $(3-3)$, due to reducing the number of false positives. Regarding the algorithms, it is clear that CompNB and MultNB achieved worse results than the others, though CompNB does achieve better results with 2-grams and 3-grams. Overall, the precision of our classifiers was consistently high, with measures ranging from 76% to 100%. In the context of the identification of observable entities, this high measure ensures that the process analyst

Figure 7.4 – Boxplot comparing the impact of the feature extraction methods across all possible training algorithms and options



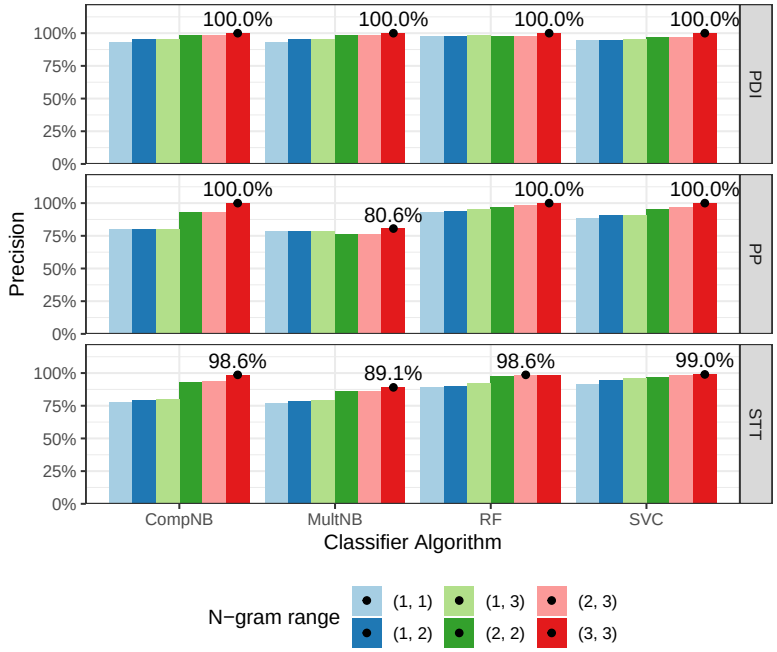
Source: The author.

will not have to review many erroneously classified process elements.

While high precision is valuable, low precision algorithms can be mitigated by reviewing the resulting set of classified process elements. A high recall is perhaps more important since it evaluates how many process elements which have observable entities were not classified by the algorithms. Considering this context, in Figure 7.6 we examine the recall measures, in which we observe that the $(1-1)$, $(1-2)$, and $(1-3)$ ranges are essential to ensuring high recall results. We can see the effect of these ranges more clearly in the PDI category, in which the recall of $(2-2)$ and $(2-3)$ is below 50%, and $(3-3)$ is below 10%. This behavior may be explained by the usual lengths of activity labels, which are not long enough to generate multiple n-grams of bigger sizes.

Regarding the algorithms, it is difficult to highlight one algorithm that has a better recall in all three classification categories. Though if we limit our considerations to the PDI category again, since it has the highest variability, the SVC algorithms have a clear advantage. Comparing the highest results of each algorithm in this category, SVC is the best with 96%, and RF is the worst with 86.0%. If instead, we consider all three categories again, but we ignore the results from $(2-2)$, $(2-3)$, and $(3-3)$ ranges, the lowest recall measure was 80.7% (RF, at PDI with $(1-3)$ range) and the highest 96,9% (tie between MultNB and CompNB, at STT with $(1-3)$ range).

Figure 7.5 – Comparision of the precision measures.



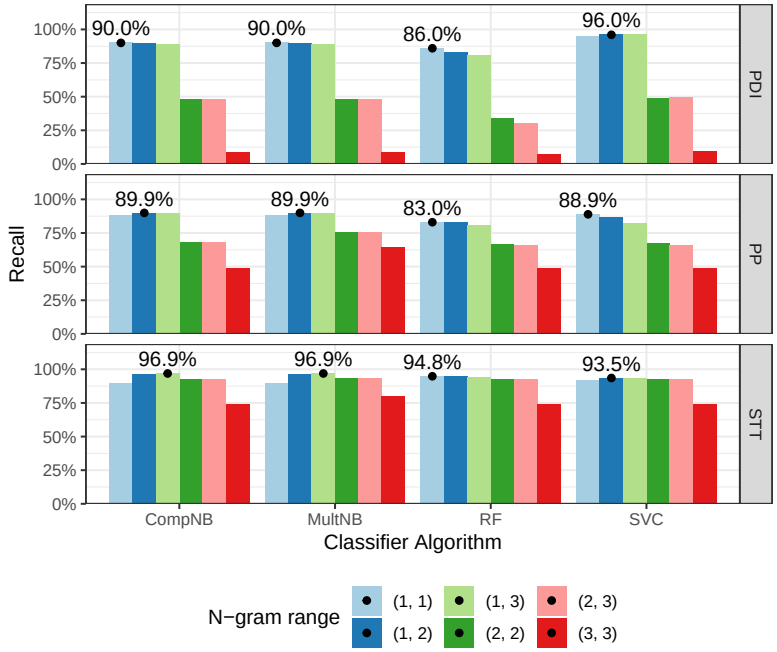
Source: The author.

Given the mostly consistent precision measures achieved by our algorithms, it was expected that the F1-score of the results would more closely resemble the recall measures. As seen in Figure 7.7, this is indeed the case. These scores again highlight the importance of 1-gram in the classification of process elements. Nevertheless, to balance both precision and recall, it may be recommended to use either $(1-2)$ or $(1-3)$ since they never had worse F1-scores than $(1-1)$ and they add more information for the algorithms to evaluate in their classification process. Following this recommendation, the best average performing algorithm across all three categories was SVC, with an F1-score of 92.4%, followed by RF with 89.8%, CompNB with 87.9%, and MultNB with 87.5%.

While analyzing the accuracy, as seen in Figure 7.8, we notice again that the SVC algorithms have achieved good performance overall in all three categories. In the automated classifier, we are generally less interested in evaluating the performance of the algorithms in predicting the true negatives, since the desired output is the list of predicted positives representing the process elements with observable entities. Nevertheless, the accuracy results provide further evidence for our conclusions made using the previous three metrics.

A final evaluation of the results is presented in Figure 7.9, in which we plot the Receiver Operator Characteristic (ROC) curves for all algorithms with $(1-2)$ or $(1-3)$ n-gram ranges. They show that these combinations had good proportions between

Figure 7.6 – Comparision of the recall measures.



Source: The author.

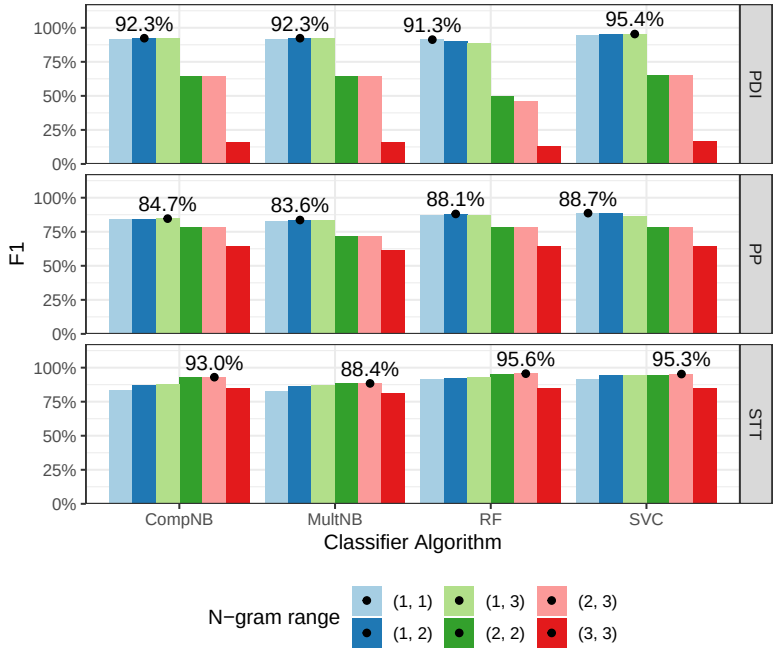
true positive (TP) and false positive (FP) rates. As expected, the algorithms for PDI and STT categories quickly achieve high TP rates while maintaining low FP rates. In the PP category, on the other hand, the TP rates rise slower compared to the FP rates. In the PP and STT categories, both CompNB and MultNB algorithms generally have lower TP rates compared to RF and SVC, which explains their lower F1-scores in Figure 7.7. Nonetheless, the ROC curves for these algorithms were placed above the diagonal y=x, with the area under these curves (AUC) being close to ideal in many of the tested classifiers.

7.4 Discussion of the Results

Based on the presented performance metrics, the automated classifiers achieved promising results overall. Using machine learning algorithms can enhance the analysis of process models by being faster at evaluating large quantities of process models and discovering valuable information. In our use case, the automated classifiers were able to identify most process elements belonging to our three classes. As such, it can assist with the identification of observable entities within process models, though at this moment it is still limited to analyzing only textual labels within activities.

Since the identification of the observable entities requires that a process analyst

Figure 7.7 – Comparision of the F1-score measures.



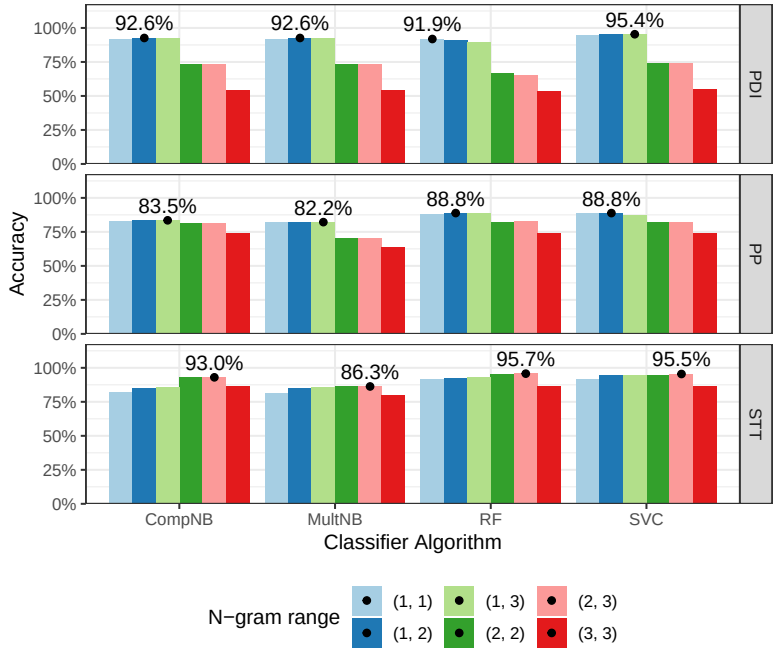
Source: The author.

understands the semantics of the classified process elements and their context within its process models, the automated classifier must have lower false negative rates, emphasizing the importance of a good recall measure. False positive rates are less critical because the process analysts can easily review and discard irrelevant process elements. Knowing this, the most suitable n-gram ranges for machine learning are $(1-2)$ and $(1-3)$, according to our results. They provide relatively high recall and moderately high precision results. Determining the best machine learning algorithms is more difficult, given that they all four tested algorithms achieved at least 80% recall at $(1-2)$ or $(1-3)$ ranges. Even so, it seems that SVC had the most consistently high results in every evaluation compared to the other algorithms.

The differences between the categories have impacted the results of the automated classifiers. Not only the categories had their differences on the number of elements and their balance as seen in Figure 7.1, but their data likely also influenced how well the classifiers identified relevant process elements. For example, we observed that in each category there are different features in the process elements that indicate whether or not it is classified in that category. In the PDI category, the appearance of the words "document," "data," and "spreadsheet" (as shown in Figure 7.2) may have a significant impact in determining the classification of an element.

Another observation was that the PP category generally achieved slightly worse

Figure 7.8 – Comparision of the accuracy measures.

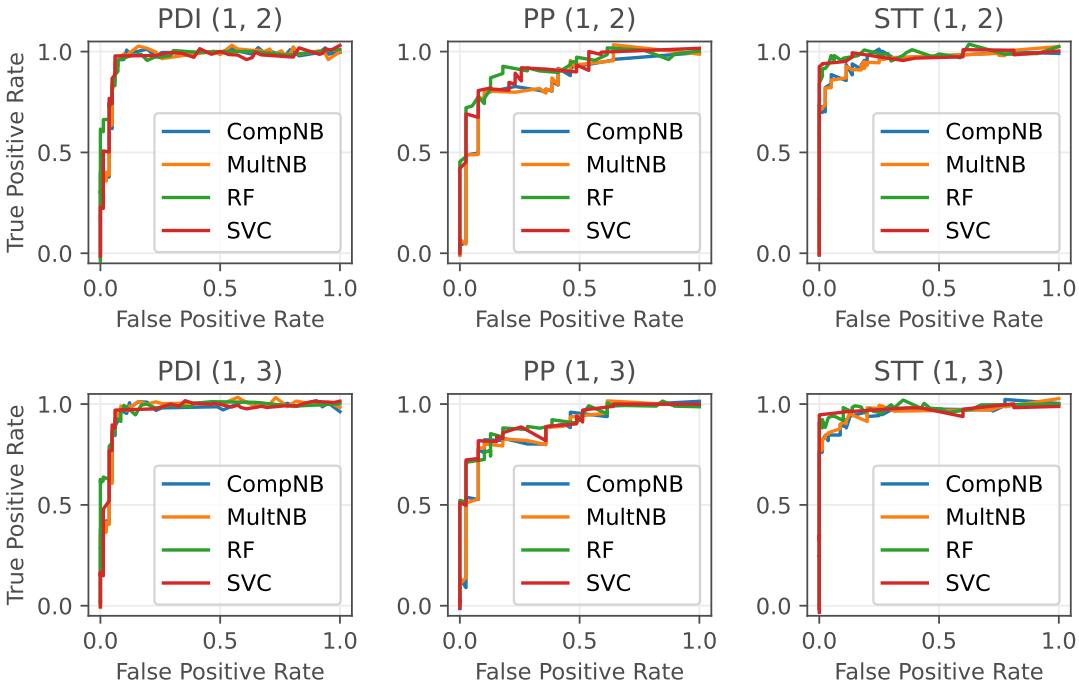


Source: The author.

results than PDI and STT. This effect may have been caused due to our decision not to consider the process elements types and subtypes at this stage. We have shown in Figure 6.12 that the swimlanes of our dataset are correlated with the PP category. Similarly, activities that include indicators for incoming and outgoing message tasks might be particularly relevant for the PP classification, since they imply communication between process participants. Once we start training the dataset with those attributes, we expect an increase in the results in the PP category.

The good results shown in Figures 7.7 and 7.9 demonstrate that our current classifiers consistently detect most infrastructure components related to PDI and STT. Further improvements may help us achieve the same for the PP category. In a real setting, the current trained machine learning models may help process analysts identify if their organization has the necessary infrastructure components for implementing their business processes. The time it takes for a process model to reach the implementation phase can be large, and there is no guarantee that the infrastructure components will remain the same until then (BIAZUS et al., 2019; BALBINOT; THOM; FANTINATO, 2017). Alternatively, monitoring these infrastructure components may provide another path to identify process models which are outdated and, thus, do not correctly represent the behavior of the business process in execution.

Figure 7.9 – ROC curve plot for the 12 best results.



Source: The author.

7.5 Chapter Summary

In this Chapter, we presented the training of an automated classifier of process model elements using machine learning. Our goal was to provide an automated assistant for the task of identifying observable entities within process models. The target classes of this classifier were the three taxonomy classes defined in Section 6.1. The training data was a subset of the dataset created manually in Section 6.2.1 containing only process activities. With this data, we trained four supervised learning algorithms (*Complement Naive-Bayes*, *Multinominal Naive-Bayes*, *Random Forest*, and *Support Vector Machine* in its classification mode) on the textual data of process elements (i.e., the labels). We also tested applying text preprocessing and feature extract techniques, including lemmatization, removal of stop-words, bag of words, TF-IDF, and 1, 2, and 3-grams.

Based on the combinations of all options, 576 automated classifiers were trained. The results generated were validated using 5-fold cross-validation, and the performance was measured through 5 metrics: precision, recall, accuracy, F1-score, and AUC-ROC. We concluded that text preprocessing techniques such as lemmatization and removing stop-words had little effect on the results. However, using them was better since they generally increased training data quality. Similarly, the difference between bag of words

and TF-IDF was slight, though we opted to continue using TF-IDF since it adds more granularity to the data.

According to the performance metrics, all algorithms performed generally well in classifying process elements, with the Support Vector Machine algorithm having slightly better and more consistent performance. Regarding the use of n-grams, the structure of process elements labels seems to favor classifiers that use the first two or all three n-grams (i.e., ranges $(1-2)$ and $(1-3)$). Considering the three taxonomy classes, *process documents and information* and *systems, tools, and technologies* achieved better performance than *process participants*;. We speculated that predicting process elements in the *process participants* class depends more on the evaluation of process element types and sub-types, such as incoming and outgoing message tasks. Overall, we conclude that machine learning classifiers are suitable assistants for identifying observable entities within process models.

8 FRAMEWORK FOR DETECTING BUSINESS PROCESS CHANGE THROUGH THE USE OF HETEROGENEOUS DATA SOURCES

In this chapter, we start defining the framework that achieves the objectives related to hypothesis H_4. The main goal of this framework is to provide a mapping that connects the observable entities detected in Chapters 6 and 7 to their respective process models. A process analyst can use this mapping to discover which process models are connected to each entity. By monitoring changes to these entities, they become the heterogeneous data sources that analysts can use to detect business process change. Then, they can use the mapping to quickly identify and update the process models affected by those changes.

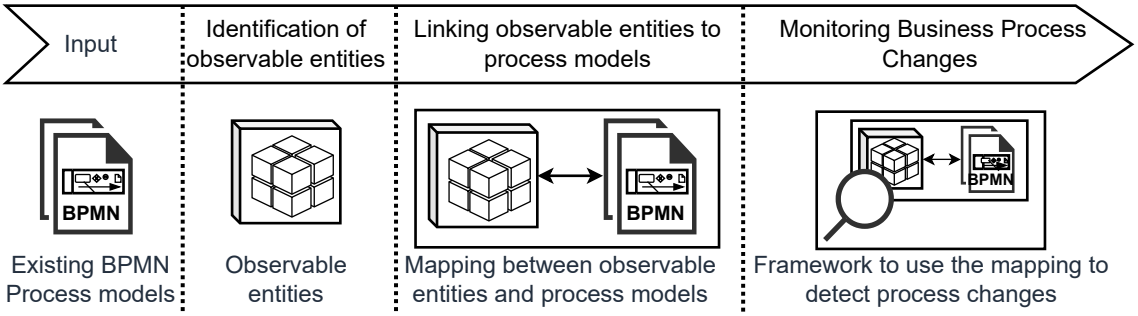
The overall steps for creating this mapping and monitoring approaches are presented in Figure 8.1. In this Figure, Chapters 6 and 7 solve the *identification of observable entities*. As such, there are two main steps left to complete: create the mapping between entities and process models and determine how to use this mapping when detecting changes to observable entities. In the remainder of this Chapter, we define the structure of the mapping of entities and process models and how to create it. We also define what must be done when changes are detected.

Defining how to turn the observable entities into heterogeneous data sources and how to monitor changes in them is outside the scope of this thesis, since the method for performing this procedure relies on knowing the structure of the entities in the practical context of their organization. For example. we cannot provide a method for monitoring systems, tools, and technologies without knowing certain details such as who developed them, how they are stored, and how they are updated. Similarly, we cannot provide a method for monitoring process participants without knowing how an organization represents both internal and external participants of their business process. Finally, we cannot provide a method for monitoring documents and information without knowing the data structure of the information or possible templates of the documents.

8.1 Creating the Mapping

The main idea of the mapping between observable entities and process models is inspired by the work system snapshot defined by Alter (2013), which we replicate in Table 4.3. In the snapshot, a list is created of all processes and activities within a work system,

Figure 8.1 – Overview on creating the framework for monitoring business process changes.



Source: The author.

as well as all components. For our mapping, we want to create these similar lists for all process models of an organization and all identified entities according to our taxonomy classes. The mapping also requires the creation of links connecting entities to process models whenever the former is used by the latter.

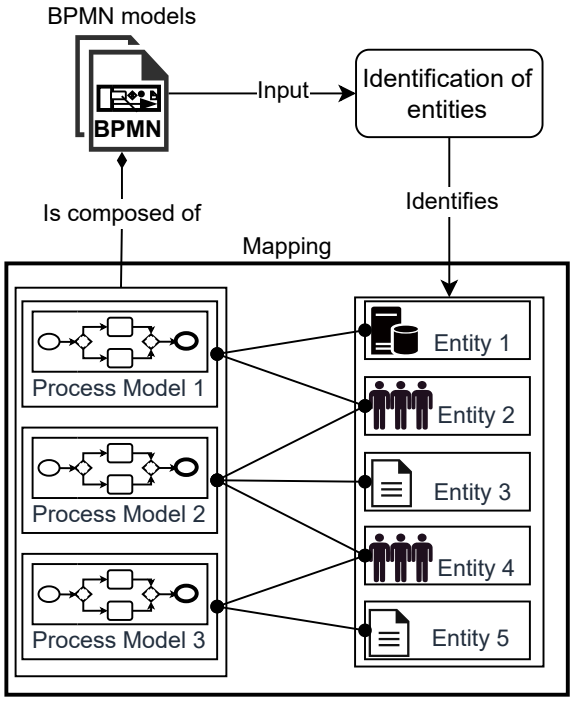
A conceptual representation of this mapping is displayed in Figure 8.2. One concern when creating this mapping is to ensure the quality of both the list of process models and the list of entities. Regarding the list of process models, it is important to ensure that they have no errors and mistakes, as well as being easy to understand. For these purposes, process analysts can apply the process modeling guidelines (MENDLING; REIJERS; AALST, 2010; AVILA et al., 2020), as detailed in Section 2.1.4. For the list of entities, on the other hand, no method of ensuring their quality exists yet.

One important problem regarding the quality of the list of entities is ensuring that none of its items are redundant, i,e., that two or more items cannot refer to the same person, system, document, etc. This problem may occur because every process model produces its own sets of identified entities, and these sets must be merged when creating the mapping. It is likely that intersections exist between those sets, e.g., two or more process models sharing the same process participants or systems. If the identified entities have the same name, merging them might be simple. However, the identified entities might be synonyms, and these synonyms might also be specific to the vocabulary of the organization.

Another problem that possibly affects this merging is homonyms, in which case the same word or name might be used to refer to two different entities. As such, to ensure the quality of the list of entities, someone with domain knowledge about the business processes must review the identified entities either manually or semi-automatically, finding and dealing with the similarities, differences, and redundancies in the list.

Another concern regarding the use of the mapping structure is that it must be com-

Figure 8.2 – Example of mapping between observable entities and process models.



Source: The author.

patible with the notion that its process models and entities will eventually change. Thus, there needs to be a way for multiple versions of the same process models or entities to exist within the mapping while also clearly defining which versions are *inactive*, i.e, that the organization stopped using it in practice and thus are no longer relevant to the business processes that are currently in execution. Maintaining previous versions of the process models and entities guarantees the preservation of their history, which can prove valuable in case the changes need to be reversed. Thus, we propose that the mapping structure utilizes a simple versioning system to manage every new update to process models and entities.

Knowing how essential the mapping is for our framework to detect business process change, maintaining its consistency rigorously is very important. A set of requirements must be followed in order to ensure this consistency, especially according to the quality concerns we presented in this Section. Therefore, we make the following set of requirement recommendations:

- Requirement recommendations regarding the entities' relevance:

  R1 All entities within the framework must be connected to a process model (even if it is an old version).

  R2 All process participant entities must be internal or external participants of a

business process (even if it is an old version).

R3 All processed documents and information entities must be created and/or used by a business process (even if it is an old version).

R4 All systems, tools, and technologies entities must be used by a business process (even if it is an old version).

- Requirement recommendations regarding the entities' uniqueness:

  U1 Each entity in the framework must have a unique name, and two entities cannot have different names if they represent the same entity in reality. An effort must be made to prevent synonyms when identifying entities.

  U2 All entities in the framework must have clear names, and one entity cannot represent two different entities in reality. An effort must be made to prevent homonyms when identifying entities.

- Requirement recommendations regarding the versioning of entities and process models:

  V1 All active process models in the framework must either be up-to-date or undergoing updates.

  V2 Outdated process models must be inactive, and they are stored only for versioning history purposes.

  V3 All active entities must be connected to an active process model.

  V4 All entities connected to an active process model must be active entities.

  V5 All active entities must be monitored.

8.2 Using the framework to update process models

During the monitoring of the observable entities within the framework, any kind of change that is detected (such as those outlined in Table 6.1) is a possible alert that the business processes related to those entities have changed, even if only slightly. Process analysts should receive this alert and promptly analyze if this change is significant enough that it requires updating the process models and the entities within the framework.

As such, every time an alert is raised by the framework, a series of instructions must be performed to ensure that all framework elements are up-to-date, and that the recommendations defined in Section 8.1 continue being followed:

1. Create an updated copy of the entity (if necessary).
2. Identify the connected process models.
3. For each process model, identify in which process elements the entity is used.
4. For each process model, identify any necessary changes.
5. For each process model, create updated versions of the process models with the necessary changes applied.
6. Turn inactive the outdated versions of the process models.
7. Perform the identification of entities in the updated process models.
8. Connect the updated process models to any identified entities that already exist in the framework.
9. Create any newly identified entities.

To explain these steps, we consider what actually happens after the alert is raised by the framework. Step (1) starts on the entity that raised the alert, signifying some change occurred to it. Since the representation of this entity within the framework does not correctly mirror the changed entity in practice, it must be deactivated from monitoring status. An updated copy of the entity is created in place of the old representation, but only if this updated entity will continue being used by (and connected to) at least one process model. As we demonstrate in Table 6.1, there are cases in which entities are just discontinued from use, or they are replaced by a different kind of entity (e.g., a process participant is replaced by an automated system).

Step (2) utilizes the mapping structure to easily trace all process models stored in the framework that might need updates to their elements. In steps (3), (4), and (5), a process analyst must analyze all identified process models to their understanding of the business processes and what the entity does within their context. It is possible for small changes to an entity to affect only a subset of the process models connected to it (e.g., the development of internal systems adds new functions, at the demand for improvements to the automation of some arduous business processes). The modeling of the updated process model versions may require discovering if significant changes have occurred to the business process operation and behavior relative to the process elements associated with the changed entity. Once this updated process model is created, it is added to the framework and the outdated version is deactivated, according to step (6).

Given that a new process model was just added to the framework, in steps (7), (8), and (9) the mapping structure must be updated to account for the entities of this new

process model. The identification of entities in step (7) can use the assistance of our proposed automated classifier, though given that a process analyst just created the new process model, they likely have a highly comprehensive and up-to-date understanding of the business process, hence they may identify entities the AI classifier would be unable to. Step (8) is performed to reconnect the new process model with already existing entities, which are still being monitored, ensuring consistency according to the recommendations defined in Section 8.1. Step (9) adds new entities to the framework, from which the monitoring methods can be created.

8.3 Chapter Summary

In this Chapter, we defined the creation of the framework to detect business process changes by monitoring changes to observable entities. We established three main steps to create this framework, which are the method of identifying observable entities, the mapping between observable entities and process models, and the use of this mapping when detecting observable entities changes.

The method for identifying observable entities was already defined in the previous Chapters. Thus, in this Chapter we presented the structure of the mapping between observable entities and process models. We emphasized the important attributes that must be considered when creating this mapping. These attributes include having process models that are easy to understand and with no errors, having a list of entities without redundancies, synonyms, and homonyms, and being capable of storing past versions of entities and process models. Based on these attributes, we established a set of requirement recommendations to ensure the rigorous consistency of the mapping.

To use the mapping, we established a series of instructions that must be executed when changes are detecting during the monitoring of the observable entities. These instructions carefully consider how to update the entities and process models of the mapping while following the recommendations established for this mapping, as well as storing the outdated versions of the entities and process models, in case the changes need to be reversed.

9 CONCLUSIONS

Throughout this thesis, we have established that maintaining process models up-to-date is both highly important and challenging. The management of business processes follows the BPM life-cycle which connects the beginnings and ends of analytical phases (process discovery, analysis, and redesign) and practical phases (process implementation, monitoring, and control). This ensures that information discovered from practice can inform the decisions made during the design and vice-versa. In this cycle, business process change occurs primarily for the improvement of the business process during the redesign phase. However, as seen through our survey on the state of business processes and process models in practice (Section 4.1), this connection between the two sides of BPM management does not always exist in organizations, possibly due to an absence of process-aware implementations of their process models. This absence would imply that no data is generated about how the business processes are performing, and thus it would not be possible to use existing methods to update process models, such as through process mining. Additionally, the works of (VEN; POOLE, 1995) and Alter (2013) provide additional theoretical perspectives contrasting with the BPM life-cycle, presenting change as something that may start happening in organizations and work systems at multiple points in time, for multiple reasons, and at multiple management levels.

Understanding that business processes are volatile and that we cannot always rely on event logs and process mining to ensure process model maintenance, we sought to start the establishment of a method to assist in the detection of outdated process models of an organization. We argued that business processes are generally linked to a variety of components that are used or created by their activities and that without these components the business processes cannot be executed. Based on this dependency, we proposed an evaluation of how feasible it would be to turn these components, which we now call observable entities, into monitorable data sources from which we could detect when a business process has changed. Our main goal was to check if it was possible to define a framework in which a mapping exists between process models and observable entities, through which process analysts can quickly find all related process models whenever an entity endures changes.

9.1 Challenges and Contributions

There were four main challenges in order to accomplish our goal: understanding how business processes change in practice; defining a classification of our concept of observable entities in relation to business process change; developing a method to identify observable entities based primarily on analyzing process models and their elements; defining how to map observable entities to process models and how to use this map to identify and update outdated models. We begun preparing to solve these challenges by researching in the literature important studies regarding change, including Ven and Poole (1995), Weick and Quinn (1999), Alter (2013), and Reijers and Mansar (2005) (as seen in Sections 4.2 and 4.3). These studies significantly influenced our understanding of change and our decisions on how to evaluate and build the methods for our framework to detect business process changes.

To solve our first challenge, we performed a series of interviews with four domain experts to analyze and update 25 existing process models. Throughout these interviews (as seen in Chapter 5), we discovered which process models were outdated, why their business processes have changed, and which were the main components that caused the identified changes. After the interviews, we have analyzed the similarities between the identified changes, and we used an adaptation of the seven classes established in the framework of Reijers and Mansar (2005) to classify those changes. We concluded that most process model changes of our study were related to at least one type of component that could be monitored without relying on event logs. We also analyzed the changes through the theories presented by Ven and Poole (1995), which led us to determine that a multiple-entity perspective was more appropriate for detecting business process changes outside the BPM life-cycle. This perspective was more compatible with the frameworks of Alter (2013) and Reijers and Mansar (2005), guiding us to use these concepts to define observable entities as a way to monitor business process change.

Regarding the second challenge, we created our taxonomy of observable entities containing three classes: process participants (PP); systems, tools, and technologies (STT); and processed documents and information (PDI). As shown in Chapter 6, these classes were synthesized mainly from the framework of Reijers and Mansar (2005), excluding changes that would be difficult to monitor through entities (e.g., business process operation and behavior changes) and aggregating all “people” entities into a single class. To evaluate this taxonomy, we attempted to apply it in building a dataset of process ele-

ments classified according to the three classes. The building of this dataset involves our third challenge, in which we defined that the identification of observable entities within process models could be done by analyzing semantical information present in process elements, such as their type, sub-types, and textual labels. We created a series of tools to extract process elements from 88 process models, filter irrelevant and problematic elements from the resulting set, and classify each process element individually, without their process model context. Two participants classified a total of 1329 process elements and were able to achieve consensus in at least 56% of the process elements in any given class. The results showed that the process models of our dataset had an average classification frequency of 57% when combining all three classes. This average indicates the degree to which our framework is expected to be capable of monitoring any given process model.

We decided to improve our identification of observable entities by training machine learning classifiers using the created datasets as training data. In Chapter 7, we have explored multiple options in terms of text preprocessing techniques, feature extraction, and machine learning algorithms. The algorithms attempted were selected based on their capacity to solve a binary classification problem with textual data as the only predictive variable. The results were evaluated by multiple performance measures, most notably the F1-score which reached an average of 92.4% when using the SVC algorithm and $(1-2)$ and $(1-3)$ n-grams. Overall, the many variations of automated classifiers showed good results at two of our taxonomy classes (PDI and STT), and slightly worse results for the PP class, though clear reasons explain why PP classifiers underperformed and how we may be able to improve their performance in the future.

In chapter 8, we detailed the components of the framework to detect business process changes by monitoring heterogeneous data sources. Specifically, we address our fourth and final challenge by demonstrating the structure of the mapping that connects process models and observable entities. We also defined a set of requirement recommendations to ensure that this mapping always presents a clear and consistent snapshot of all active elements and that the versioning history of outdated elements is maintained in case any updates have to be reversed. Finally, we detailed a series of instructions on what process analysts must do when changes are detected within observable entities, ensuring all elements connected to this entity are updated and that the mapping requirements continue being followed.

The most important contribution of this thesis is the definition of our taxonomy classes, which is clearly linked to frameworks of analysis and redesign of business pro-

cesses and to which we provided clear examples of how they can be related to business process change. The interviews and the analysis of the updated process models from those interviews show how important it is to consider the entities of these classes when analyzing business processes. Even when this taxonomy is not applied for detecting business process change, the analysis of these entities can help assess the viability of implementing process models in organizations sooner, avoiding the sudden obstacles that these implementations may face during the BPM life-cycle.

Another contribution was the definition of the mapping between entities and process models provides the means through which process analysis can start building alternative monitoring methods to detect business process changes. The mapping structure helps analysts keep track of all the entities within an organization, providing similar benefits to the work system snapshot shown by Alter (2013). It also allows for the evaluation of how dependent the business processes are to certain entities, and thus are more likely to suffer changes by virtue of this dependency.

Finally, we made available the tools to create a dataset of classified process elements and to use this dataset to train automated classifiers. These tools mainly support the classification according to our three taxonomy classes, and they help ensure the consistency of the manual classification. Since their source codes are publicly available, their application can easily be expanded to consider other types of process element classifications.

9.2 Publications

Regarding the publications made during our research, we have participated in the development of five articles and one book chapter. These publications are:

- **A Systematic Literature Review of Process Modeling Guidelines and their Empirical Support**
  Authors: Diego Toralles Avila, Rubens Ideron dos Santos, Jan Mendling, Lucineia Heloisa Thom.
  Journal: Business Process Management Journal - 2020.
  Qualis: A1.
- **An Experiment to Analyze the Use of Process Modeling Guidelines to Create High-Quality Process Models**

Authors: Diego Toralles Avila, Raphael Piegas Cigana, Marcelo Fantinato, Hajo A. Reijers, Jan Mendling, Lucineia Heloisa Thom.
Conference: International Conference on Database and Expert Systems Applications - DEXA 2019.
Qualis: A4.

- **A Service-Oriented Architecture for Generating Sound Process Descriptions**
Authors: Thanner Soares Silva, Diego Toralles Avila, Jean Ampos Flesch, Sarajane Marques Peres, Jan Mendling, Lucineia Heloisa Thom.
Conference: IEEE International Enterprise Distributed Object Computing Conference - EDOC 2019.
Qualis: A4.
- **A Practical User Feedback Classifier for Software Quality Characteristics**
Authors: Rubens dos Santos, Karina Villela, Diego Toralles Avila, Lucineia Heloisa Thom.
Conference: International Conference on Software Engineering & Knowledge Engineering - SEKE 2021.
Qualis: A4.
- **Introdução à Modelagem de Processos de Negócio em BPMN 2.0 e à Automação em BPMS**
Authors: Lucineia Heloisa Thom, Diego Toralles Avila
Book: 39º Jornadas de Atualização em Informática - 2020.
Qualis: N/A.
- **Using Machine Learning to Classify Process Model Elements for Process Infrastructure Analysis**
Authors: Diego Toralles Avila, Vitor Camargo de Moura, Lucineia Heloisa Thom.
Conference: Simpósio Brasileiro de Sistemas de Informação - SBSI 2023.
Qualis: A4.

9.3 Limitations and Future Research

While our definition of the taxonomy of observable entities was based on well-established theoretical background, it suffers from the limitation that our prior analysis involved only 25 process models that were primarily administrative and in an academic

context. In future work, by updating and analyzing process models of different contexts, we may discover different reasons for business process changes that may improve the accuracy of our definitions, up to including new taxonomy classes not contemplated by Aalst (2013) and Reijers and Mansar (2005). Our current definitions might be just a starting point for the identification of observable entities, and it could be possible to discover other types of observable entities that can be monitored to detect business process change.

The dataset of classified process elements helped evaluate the definition of our taxonomy, showing that identifying observable entities from process models is possible. However, a drawback of our approach is that we could not consider the context of the process models, primarily because our classifiers did not have knowledge about the business processes and their respective domains. This forced our classification process to be simplified, allowing for process elements to be classified quickly, but there was a sizeable amount of divergent classifications between the two participants. It is also possible that some observable entities would be missed regardless since there is no way to ensure that conceptual process models contain semantical information about all important components for its execution. For future work, performing the classification with more participants could help solve classification ambiguities and create a better-quality dataset. Adding more process model elements to be classified could also be useful, given that we used oversampling to balance the datasets, though we are unsure if it would significantly improve our automated classifiers' best performances.

Regarding the automated classifiers, we have mentioned that the current versions only evaluate process activities and only extract textual information for its training. This choice simplified the development of the classifiers and their evaluation, but, as the results showed, it certainly impacted the performance of the classification, particularly in the prediction of process elements in the PP class, which depends more heavily on certain process element types and subtypes. Fortunately, the algorithms we tested are capable of using more variables as training input, so in future work, we can improve the classification results with better-quality datasets and better implementations of the preprocessing techniques and machine learning algorithms.

Finally, in the framework for detecting business process change, we have established a method to identify which process models are outdated when changes to connected observable entities are detected. However, we have not yet established how to monitor the heterogeneous data of those entities. The type and structure of the entities determine how these entities can be monitored, and we currently have no data to predict these attributes.

Further studies are necessary to analyze these entities in practical settings and establish patterns for methods of monitoring them. These studies may also help the evaluation of the framework by applying them with their established monitoring methods.

REFERENCES

AA, H. van der et al. Challenges and opportunities of applying natural language processing in business process management. In: *Proceedings of the 27th International Conference on Computational Linguistics*. Santa Fe, New Mexico, USA: Association for Computational Linguistics, 2018. p. 2791–2801.

AALST, W. M. P. van der. Extracting Event Data from Databases to Unleash Process Mining. **BPM - Driving Innovation in a Digital World SE - 8**, Springer, Cham, p. 105–128, 2015.

AALST, W. M. Van der. Business process management: a comprehensive survey. **International Scholarly Research Notices**, Hindawi, v. 2013, 2013.

AALST, W. van der; ADRIANSYAH, A.; DONGEN, B. van. Replaying history on process models for conformance checking and performance analysis. **WIREs Data Mining and Knowledge Discovery**, Wiley, v. 2, n. 2, p. 182–192, jan 2012.

ADRIANSYAH, A.; DONGEN, B. F. van; AALST, W. M. P. van der. Towards robust conformance checking. In: **Business Process Management Workshops**. [S.l.]: Springer Berlin Heidelberg, 2011. p. 122–133.

ADRIANSYAH, A.; DONGEN, B. van; AALST, W. van der. Conformance checking using cost-based fitness analysis. **2011 IEEE 15th International Enterprise Distributed Object Computing Conference**, IEEE, p. 55–64, aug 2011. ISSN 1541-7719.

ADRIANSYAH, A.; SIDOROVA, N.; DONGEN, B. van. Cost-based fitness in conformance checking. In: **2011 Eleventh International Conference on Application of Concurrency to System Design**. [S.l.]: IEEE, 2011.

AGOSTINELLI, S. et al. Achieving gdpr compliance of bpmn process models. In: CAPPIELLO, C.; RUIZ, M. (Ed.). **Information Systems Engineering in Responsible Information Systems**. Cham: Springer International Publishing, 2019. p. 10–22. ISBN 978-3-030-21297-1.

ALPAYDIN, E. **Introduction to Machine Learning**. Cambridge, MA, USA: MIT Press, 2010. Available from Internet: <https://ieeexplore.ieee.org/book/6267367>.

ALTER, S. Work system theory: Overview of core concepts, extensions, and challenges for the future. **Journal of the Association for Information Systems**, Association for Information Systems, v. 14, n. 2, p. 72–121, feb 2013.

ALTER, S. Theory of workarounds. **Communications of the Association for Information Systems**, v. 34, 01 2014.

ANDRADE, E. et al. Factors leading to business process noncompliance and its positive and negative effects: Empirical insights from a case study. In: . [S.l.: s.n.], 2016.

AVILA, D. T. et al. An experiment to analyze the use of process modeling guidelines to create high-quality process models. In: HARTMANN, S. et al. (Ed.). **Database and Expert Systems Applications**. Cham: Springer International Publishing, 2019. p. 129–139. ISBN 978-3-030-27618-8.

AVILA, D. T. et al. A systematic literature review of process modeling guidelines and their empirical support. **Business Process Management Journal**, Emerald Publishing Limited, Nov 2020.

BALBINOT, M.; THOM, L.; FANTINATO, M. Identificando Fontes de Dados em Modelos de Processos de Negócio com base em Elementos de BPMN. **SBC**, SBC, p. 444–451, May 2017. ISSN 0000-0000.

BERIO, G.; VERNADAT, F. Enterprise modelling with cimosa: Functional and organizational aspects. **Production Planning and Control**, Informa UK Limited, v. 12, n. 2, p. 128–136, 03 2001.

BIAZUS, M. et al. Software resource recommendation for process execution based on the organization's profile. In: **Database and Expert Systems Applications**. [S.l.: s.n.], 2019. p. 118–128. ISBN 978-3-030-27617-1.

BOHNENBERGER, N. M. de M.; SCHMITT, A. C.; THOM, L. H. Discovering healthcare processes from natural language documents: a case study on COVID-19. In: VOGEL, D. et al. (Ed.). **25th Pacific Asia Conference on Information Systems, PACIS 2021, Virtual Event / Dubai, UAE, July 12-14, 2021**. [s.n.], 2021. p. 175. Available from Internet: <https://aisel.aisnet.org/pacis2021/175>.

BOSE, R. J. **Process mining in the large : preprocessing, discovery, and diagnostics**. Thesis (PhD) — Department of Mathematics and Computer Science, 2012. Proefschrift.

BOSE, R. P. J. C. et al. Dealing With Concept Drifts in Process Mining. **IEEE Transactions on Neural Networks and Learning Systems**, v. 25, n. 1, p. 154–171, jan. 2014. ISSN 2162-2388.

BROUCKE, S. K. L. M. vanden et al. Event-based real-time decomposed conformance analysis. In: **On the Move to Meaningful Internet Systems: OTM 2014 Conferences**. Berlin, Germany: Springer Berlin Heidelberg, 2014. p. 345–363. ISBN 978-3-662-45562-3.

BROUCKE, S. K. L. M. vanden et al. Determining process model precision and generalization with weighted artificial negative events. **IEEE Transactions on Knowledge and Data Engineering**, Institute of Electrical and Electronics Engineers (IEEE), v. 26, n. 8, p. 1877–1889, aug 2014. ISSN 1558-2191.

BURKE, W. W. **Organization change: Theory and practice**. [S.l.]: Sage publications, 2017.

CALDERS, T. et al. Using minimum description length for process mining. In: **Proceedings of the 2009 ACM symposium on Applied Computing - SAC '09**. [S.l.]: ACM Press, 2009.

CAPORALE, T. A tool for natural language oriented business process modeling. In: HOCHREINER, C.; SCHULTE, S. (Ed.). **Proceedings of the 8th ZEUS Workshop, Vienna, Austria, January 27-28, 2016**. CEUR-WS.org, 2016. (CEUR Workshop Proceedings, v. 1562), p. 49–52. Available from Internet: <http://ceur-ws.org/Vol-1562/paper7.pdf>.

CARMONA, J.; GAVALDÀ, R. Online techniques for dealing with concept drift in process mining. In: **Advances in Intelligent Data Analysis XI**. [S.l.]: Springer Berlin Heidelberg, 2012. p. 90–102.

CERAVOLO, P. et al. Evaluation goals for online process mining: a concept drift perspective. **IEEE Transactions on Services Computing**, Institute of Electrical and Electronics Engineers (IEEE), p. 1–1, 2020.

CONFORT, V. **THE BPM ISSUES IN BRAZILIAN PERSPECTIVE**. Thesis (PhD) — Universidade Federal do Estado do Rio de Janeiro, Brasil, 2010.

COOK, J. E.; WOLF, A. L. Software process validation. **ACM Transactions on Software Engineering and Methodology**, Association for Computing Machinery (ACM), v. 8, n. 2, p. 147–176, apr 1999.

DONG, J. Q. et al. Information technology in innovation activity of the firm: Theory and synthesis. In: **ECIS**. [S.l.: s.n.], 2013.

DUMAS, M.; AALST, W. M. P. van der; HOFSTEDE, A. H. M. ter (Ed.). **Process-Aware Information Systems**. [S.l.]: John Wiley & Sons, Inc., 2005.

DUMAS, M. et al. **Fundamentals of Business Process Management, Second Edition**. [S.l.]: Springer Berlin Heidelberg, 2018. ISBN 978-3-662-56508-7.

ELKHAWAGA, G. et al. CONDA-PM – A Systematic Review and Framework for Concept Drift Analysis in Process Mining. **arXiv**, Sep 2020.

EUROPEAN COMMISSION. **2018 reform of EU data protection rules**. 2018. Available from Internet: <https://ec.europa.eu/commission/sites/beta-political/files/data-protection-factsheet-changes_en.pdf>.

FACELI, K. et al. **Inteligência artificial : uma abordagem de aprendizado de máquina.** LTC, 2021. ISBN 9788521637349. Available from Internet: <https://search.ebscohost.com/login.aspx?direct=true&AuthType=shib&db=edsmib&AN=edsmib.000021044&lang=pt-br&scope=site&authtype=guest,shib&custid=s5837110&groupid=main&profile=eds>.

FAHLAND, D.; Van Der Aalst, W. M. Model repair - Aligning process models to reality. **Information Systems**, Pergamon, v. 47, p. 220–243, jan 2015. ISSN 03064379. Available from Internet: <http://www.sciencedirect.com/science/article/pii/S0306437913001725>.

FERREIRA, R. C. B. et al. Recognition of Business Process Elements in Natural Language Texts. In: **Enterprise Information Systems**. Cham, Switzerland: Springer International Publishing, 2018. p. 591–610.

FIGL, K. Comprehension of procedural visual business process models: A literature review. **Business and Information Systems Engineering**, Springer Fachmedien Wiesbaden, v. 59, n. 1, p. 41–67, 2017. ISSN 18670202.

FIROUZIAN, I.; ZAHEDI, M.; HASSANPOUR, H. Investigation of the Effect of Concept Drift on Data-Aware Remaining Time Prediction of Business Processes. **International Journal of Nonlinear Analysis and Applications**, Semnan University, v. 10, n. 2, p. 153–166, dec. 2019. ISSN 2008-6822. Available from Internet: <https://ijnaa.semnan.ac.ir/article_4182.html>.

FRIEDRICH, F.; MENDLING, J.; PUHLMANN, F. Process model generation from natural language text. In: MOURATIDIS, H.; ROLLAND, C. (Ed.). **Advanced Information Systems Engineering**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. p. 482–496. ISBN 978-3-642-21640-4.

GARCIA-BANUELOS, L. et al. Complete and interpretable conformance checking of business processes. **IEEE Transactions on Software Engineering**, Institute of Electrical and Electronics Engineers (IEEE), v. 44, n. 3, p. 262–290, mar 2018.

GOEDERTIER, S. et al. Robust process discovery with artificial negative events. **J. Mach. Learn. Res.**, JMLR.org, v. 10, p. 1305–1340, jun. 2009. ISSN 1532-4435.

GSCHWIND, T. et al. A linear time layout algorithm for business process models. **Journal of Visual Languages & Computing**, v. 25, n. 2, p. 117–132, apr 2014. ISSN 1045926X.

HOMPES, B. F. A. et al. Detecting Changes in Process Behavior Using Comparative Case Clustering. In: CERAVOLO, P.; RINDERLE-MA, S. (Ed.). **Data-Driven Process Discovery and Analysis**. Cham: Springer International Publishing, 2017. (Lecture Notes in Business Information Processing), p. 54–75. ISBN 9783319534350.

IHDE, S. et al. Optimized resource allocations in business process models. In: ____. [S.l.]: Springer International Publishing, 2019. p. 55–71. ISBN 978-3-030-26642-4.

ISO, I. O. for S. **ISO/IEC 19510:2013: Information technology – Object Management Group Business Process Model and Notation, V2.0.2**. Geneva, 2013. 1–507 p. [Online; accessed 16. Mar. 2021]. Available from Internet: <https://www.iso.org/standard/62652.html>.

JABLONSKI, S.; BUSSLER, C. **Workflow management: modeling concepts, architecture and implementation**. [S.l.]: International Thomson Computer Press, London, UK, 1996.

JURAFSKY, D.; MARTIN, J. H. **Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition**. [S.l.]: Prentice Hall, Pearson Education International, 2009. 1–1024 p.

KÖNIG, U. M.; LINHART, A.; RÖGLINGER, M. Why do business processes deviate? Results from a Delphi study. **Bus. Res.**, Springer International Publishing, v. 12, n. 2, p. 425–453, Dec 2019. ISSN 2198-2627.

KOSCHMIDER, A.; FIGL, K.; SCHOKNECHT, A. A comprehensive overview of visual design of process model element labels. In: REICHERT, M.; REIJERS, H. A. (Ed.). **Business Process Management Workshops**. Cham: Springer International Publishing, 2016. v. 256, p. 571–582. ISBN 978-3-319-42886-4 978-3-319-42887-1.

KROGSTIE, J. Quality of business process models. In: ____. **Quality in Business Process Modeling**. Cham: Springer International Publishing, 2016. p. 53–102. ISBN 978-3-319-42512-2.

LEAVITT, H. J. Applied organizational change in industry, structural, technological and humanistic approaches. **Handbook of organizations**, Rand McNally & Company, v. 264, p. 1144–1170, 1965.

LENZ, M. L. et al. **Fundamentos de aprendizagem de máquina.** SAGAH, 2020. (Inteligência artificial). ISBN 9786556900902. Available from Internet: <https://search.ebscohost.com/login.aspx?direct=true&AuthType=shib&db=edsmib&AN=edsmib.000020116&lang=pt-br&scope=site&authtype=guest,shib&custid=s5837110&groupid=main&profile=eds>.

LEOPOLD, H.; MENDLING, J.; GÜNTHER, O. Learning from quality issues of bpmn models from industry. **CEUR Workshop Proceedings**, v. 1701, p. 36–39, 2016. ISSN 16130073.

LI, T. et al. Unraveling Process Evolution by Handling Concept Drifts in Process Mining. In: **2017 IEEE International Conference on Services Computing (SCC)**. [S.l.: s.n.], 2017. p. 442–449. ISSN 2474-2473. ISSN: 2474-2473.

LIU, N.; HUANG, J.; CUI, L. A Framework for Online Process Concept Drift Detection from Event Streams. In: **2018 IEEE International Conference on Services Computing (SCC)**. [S.l.: s.n.], 2018. p. 105–112. ISSN 2474-2473. ISSN: 2474-2473.

MAARADJI, A. et al. Detecting Sudden and Gradual Drifts in Business Processes from Execution Traces. **IEEE Transactions on Knowledge and Data Engineering**, v. 29, n. 10, p. 2140–2154, oct. 2017. ISSN 1558-2191.

MAISENBACHER, M.; WEIDLICH, M. Handling Concept Drift in Predictive Process Monitoring. In: **2017 IEEE International Conference on Services Computing (SCC)**. [S.l.: s.n.], 2017. p. 1–8. ISSN 2474-2473. ISSN: 2474-2473.

MANNHARDT, F. et al. Balanced multi-perspective checking of process conformance. **Computing**, Springer Science and Business Media LLC, v. 98, n. 4, p. 407–437, feb 2015. ISSN 1436-5057.

MARTJUSHEV, J.; R.P., J. C. B.; AALST, W. Change Point Detection and Dealing with Gradual and Multi-order Dynamics in Process Mining. In: . [S.l.: s.n.], 2015. p. 161–178. ISBN 9783319219141.

MENDLING, J. Managing structural and textual quality of business process models. In: **Lecture Notes in Business Information Processing**. [S.l.]: Springer, Berlin, Heidelberg, 2013. v. 162, p. 100–111. ISBN 9783642409189.

MENDLING, J.; REIJERS, H. A.; AALST, W. M. P. van der. Seven process modeling guidelines (7pmg). **Information and Software Technology**, v. 52, n. 2, p. 127–136, feb 2010. ISSN 09505849.

MENDLING, J.; STREMBECK, M. Influence factors of understanding business process models. **Lecture Notes in Business Information Processing**, v. 7 LNBIP, p. 142–153, 2008. ISSN 18651348.

MILANI, F.; MAGGI, F. M. A comparative evaluation of log-based process performance analysis techniques. In: ABRAMOWICZ, W.; PASCHKE, A. (Ed.). **Business Information Systems**. Cham: Springer International Publishing, 2018. p. 371–383.

MITCHELL, T. M. et al. **Machine learning**. McGraw-hill New York, 1997. Available from Internet: <http://www.cs.cmu.edu/~tom/mlbook.html>.

MUÑOZ-GAMA, J.; CARMONA, J. A fresh look at precision in process conformance. In: **Lecture Notes in Computer Science**. Berlin, Germany: Springer Berlin Heidelberg, 2010. p. 211–226. ISBN 978-3-642-15617-5.

MUNOZ-GAMA, J.; CARMONA, J. Enhancing precision in process conformance: Stability, confidence and severity. In: **2011 IEEE Symposium on Computational Intelligence and Data Mining (CIDM)**. [S.l.]: IEEE, 2011.

MUNOZ-GAMA, J.; CARMONA, J.; AALST, W. M. van der. Single-entry single-exit decomposed conformance checking. **Information Systems**, Elsevier BV, v. 46, p. 102–122, dec 2014. ISSN 0306-4379.

NADLER, D. A.; TUSHMAN, M. L. A model for diagnosing organizational behavior. **Organizational Dynamics**, v. 9, n. 2, p. 35 – 51, 1980. ISSN 0090-2616. Available from Internet: <http://www.sciencedirect.com/science/article/pii/009026168090039X>.

NELLI, F. Python data analytics. Apress, 2015.

NELSON, H. J. et al. A conceptual modeling quality framework. **Software Quality Journal**, Springer US, v. 20, n. 1, p. 201–228, mar 2012. ISSN 0963-9314.

NETTO, A. **Python para data science e machine learning descomplicado.** Alta Books, 2021. ISBN 9786555203370. Available from Internet: <https://search.ebscohost.com/login.aspx?direct=true&AuthType=shib&db=edsmib&AN=edsmib.000024782&lang=pt-br&scope=site&authtype=guest,shib&custid=s5837110&groupid=main&profile=eds>.

OMG, O. M. G. **Business Process Model and Notation (BPMN) Version 2.0**. [S.l.], 2011.

OMORI, N. J. et al. Comparing concept drift detection with process mining software. **iSys - Brazilian Journal of Information Systems**, Sociedade Brasileira de Computacao - SB, v. 13, n. 4, p. 101–125, jul 2020.

OSTOVAR, A. et al. Detecting Drift from Event Streams of Unpredictable Business Processes. In: COMYN-WATTIAU, I. et al. (Ed.). **Conceptual Modeling**. Cham: Springer International Publishing, 2016. (Lecture Notes in Computer Science), p. 330–346. ISBN 9783319463971.

PEDREGOSA, F. et al. Scikit-learn: Machine learning in Python. **Journal of Machine Learning Research**, v. 12, p. 2825–2830, 2011.

PLATTFAUT, R. et al. Development of bpm capabilities - is maturity the right path? In: **ECIS**. [S.l.: s.n.], 2011.

POLYVYANYY, A. et al. Impact-Driven Process Model Repair. **ACM Transactions on Software Engineering and Methodology**, ACM, v. 25, n. 4, p. 1–60, oct 2016. ISSN 1049331X. Available from Internet: <http://dl.acm.org/citation.cfm?doid=3007747.2980764>.

PORRAS, J. I.; ROBERTSON, P. J. **Organizational development: Theory, practice, and research.** [S.l.]: Consulting Psychologists Press, 1992.

PRATHAMA, F. et al. Trace Clustering Exploration for Detecting Sudden Drift: A Case Study in Logistic Process. **Procedia Computer Science**, v. 161, p. 1122–1130, jan. 2019. ISSN 1877-0509. Available from Internet: <http://www.sciencedirect.com/science/article/pii/S1877050919319350>.

REICHERT, M.; WEBER, B. **Enabling Flexibility in Process-Aware Information Systems**. Berlin, Germany: Springer, 2012.

REIJERS, H.; MANSAR, S. L. Best practices in business process redesign: an overview and qualitative evaluation of successful redesign heuristics. **Omega**, v. 33, n. 4, p. 283 – 306, 2005. ISSN 0305-0483.

REIJERS, H. A.; MENDLING, J.; RECKER, J. Business process quality management. In: **Handbook on Business Process Management 1**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015. p. 167–185. ISBN 9783642019814.

REISSNER, D. et al. Scalable Conformance Checking of Business Processes. In: **On the Move to Meaningful Internet Systems. OTM 2017 Conferences**. Cham, Switzerland: Springer, 2017. p. 607–627. ISBN 978-3-319-69461-0.

RICHTER, F.; SEIDL, T. Looking into the TESSERACT: Time-drifts in event streams using series of evolving rolling averages of completion times. **Information Systems**, v. 84, p. 265–282, sep. 2019. ISSN 0306-4379. Available from Internet: <http://www.sciencedirect.com/science/article/pii/S030643791830019X>.

ROSA, L. S. et al. A visual approach for identification and annotation of business process elements in process descriptions. **Computer Standards & Interfaces**, Elsevier BV, v. 81, p. 103601, apr 2022. ISSN 0920-5489.

ROZINAT, A.; AALST, W. van der. Conformance checking of processes based on monitoring real behavior. **Information Systems**, Elsevier BV, v. 33, n. 1, p. 64–95, mar 2008. ISSN 0306-4379.

RUTKOWSKI, L.; JAWORSKI, M.; DUDA, P. Basic concepts of data stream mining. In: **Studies in Big Data**. [S.l.]: Springer International Publishing, 2019. p. 13–33.

SADIQ, S. et al. Major issues in business process management: A vendor perspective. In: THE UNIVERSITY OF AUCKLAND, SCHOOL OF BUSINESS. **Managing Diversity in Digital Enterprises: Proceedings of the 11th Pacific Asia Conference on Information Systems**. [S.l.], 2007. p. 40–47.

SÁNCHEZ-GONZÁLEZ, L. et al. A case study about the improvement of business process models driven by indicators. **Software & Systems Modeling**, Springer Berlin Heidelberg, v. 16, n. 3, p. 759–788, 2017. ISSN 1619-1366.

SCHWABER, K.; SUTHERLAND, J. The Scrum Guide. In: **Software in 30 Days**. Chichester, England, UK: John Wiley & Sons, Ltd, 2012. p. 133–152. ISBN 978-1-11920327-8.

SEIDMANN, A.; SUNDARARAJAN, A. The effects of task and information asymmetry on business process redesign. **International Journal of Production Economics**, v. 50, n. 2, p. 117–128, 1997. ISSN 0925-5273. Business Process Reengineering.

SHANKS, G.; JOHNSTON, R. Exploring process theory in information systems research. In: GREGOR, S.; HART, D. (Ed.). **Proceedings of The Information Systems Foundation Workshop (ISF 2012 Workshop)**. Australia: ANU E Press, 2012. Information Systems Foundations Workshop ; Conference date: 01-01-2011.

SILVA, T. S. et al. A Service-Oriented Architecture for Generating Sound Process Descriptions. In: **2019 IEEE 23rd International Enterprise Distributed Object Computing Conference (EDOC)**. Paris, France: IEEE, 2019. p. 1–10. ISSN 2325-6362.

SILVA, T. S. et al. Empirical Analysis of Sentence Templates and Ambiguity Issues for Business Process Descriptions. In: **Lecture Notes in Computer Science**. Cham, Switzerland: Springer International Publishing, 2018. p. 279–297.

SOUSA, R. G. D.; PERES, S. M. Online concept drift detection, localization andcharacterization using trace clustering. **SBC**, Sociedade Brasileira de Computação (SBC), p. 35–39, nov 2020. ISSN 0000-0000.

STERTZ, F.; MANGLER, J.; RINDERLE-MA, S. Data-driven Improvement of Online Conformance Checking. **2020 IEEE 24th International Enterprise Distributed Object Computing Conference (EDOC)**, IEEE, p. 187–196, Oct 2020. ISSN 2325-6362.

STERTZ, F.; MANGLER, J.; RINDERLE-MA, S. Temporal Conformance Checking at Runtime based on Time-infused Process Models. **arXiv**, Aug 2020. Available from Internet: <https://arxiv.org/abs/2008.07262v1>.

STERTZ, F.; RINDERLE-MA, S. Process Histories - Detecting and Representing Concept Drifts Based on Event Streams. In: PANETTO, H. et al. (Ed.). **On the Move to Meaningful Internet Systems. OTM 2018 Conferences**. Cham: Springer International Publishing, 2018. (Lecture Notes in Computer Science), p. 318–335. ISBN 9783030026103.

STERTZ, F.; RINDERLE-MA, S. Detecting and Identifying Data Drifts in Process Event Streams Based on Process Histories. In: **Information Systems Engineering in Responsible Information Systems**. Cham, Switzerland: Springer, 2019. p. 240–252. ISBN 978-3-030-21296-4.

STERTZ, F.; RINDERLE-MA, S.; MANGLER, J. Analyzing Process Concept Drifts Based on Sensor Event Streams During Runtime. In: **Business Process Management**. Cham, Switzerland: Springer, 2020. p. 202–219. ISBN 978-3-030-58665-2.

TAVARES, G. M. et al. Overlapping analytic stages in online process mining. In: **2019 IEEE International Conference on Services Computing (SCC)**. [S.l.]: IEEE, 2019.

THOM, L. **A Collaborative and Practical Method for Teaching Business Process Design**. 2020. [Online; accessed 7. Apr. 2021]. Available from Internet: <https://www.researchgate.net/publication/344355033_A_Collaborative_and_Practical_Method_for_Teaching_Business_Process_Design>.

TICHY, N. M. **Managing strategic change: Technical, political, and cultural dynamics**. [S.l.]: John Wiley & Sons, 1983.

TrustRadius Inc. **List of Top Business Process Management (BPM) Tools 2020**. 2020. [Online; accessed 29. Nov. 2020]. Available from Internet: <https://www.trustradius.com/business-process-management-bpm>.

Van Der Aalst, W. et al. Process mining manifesto. In: **Lecture Notes in Business Information Processing**. [S.l.]: Springer, Berlin, Heidelberg, 2012. v. 99 LNBIP, n. PART 1, p. 169–194. ISBN 9783642281075. ISSN 18651348.

Van Der Aalst, W. M. Process-aware information systems: Lessons to be learned from process mining. In: ____. **Transactions on Petri Nets and Other Models of Concurrency II: Special Issue on Concurrency in Process-Aware Information Systems**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. p. 1–26.

Van Der Aalst, W. M. **Process Mining**. [S.l.: s.n.], 2016. 301–317 p. ISSN 0170-6012. ISBN 9783642193453.

Van Der Aalst, W. M. P.; HOFSTEDE, A. H. M. ter; WESKE, M. Business process management: A survey. In: **Lecture Notes in Computer Science**. [S.l.]: Springer Berlin Heidelberg, 2003. p. 1–12.

VEN, A. H. Van de; POOLE, M. S. Explaining development and change in organizations. **Academy of management review**, Academy of Management Briarcliff Manor, NY 10510, v. 20, n. 3, p. 510–540, 1995.

WEBER, P.; TIÑO, P.; BORDBAR, B. Process mining in non-stationary environments. In: **ESANN**. [S.l.: s.n.], 2012.

WEERDT, J. D. et al. A robust f-measure for evaluating discovered process models. **2011 IEEE Symposium on Computational Intelligence and Data Mining (CIDM)**, IEEE, p. 148–155, apr 2011.

WEICK, K. E.; QUINN, R. E. Organizational change and development. **Annual review of psychology**, Annual Reviews 4139 El Camino Way, PO Box 10139, Palo Alto, CA 94303-0139, USA, v. 50, n. 1, p. 361–386, 1999.

WEISBORD, M. R. Organizational diagnosis: Six places to look for trouble with or without a theory. **Group & Organization Studies**, Sage Publications Sage CA: Thousand Oaks, CA, v. 1, n. 4, p. 430–447, 1976.

WESENBERG, H. Enterprise modeling in an agile world. In: **Lecture Notes in Business Information Processing**. [S.l.]: Springer Berlin Heidelberg, 2011. p. 126–130.

WHETTEN, D. What constitutes a theoretical contribution? **Academy of Management Review**, v. 14, p. 490–495, 10 1989.

YESHCHENKO, A. et al. Comprehensive Process Drift Detection with Visual Analytics. In: LAENDER, A. H. F. et al. (Ed.). **Conceptual Modeling**. Cham: Springer International Publishing, 2019. (Lecture Notes in Computer Science), p. 119–135. ISBN 9783030332235.

YESHCHENKO, A. et al. Visual Drift Detection for Sequence Data Analysis of Business Processes. **IEEE Trans. Visual. Comput. Graphics**, IEEE, p. 1, Jan 2021. ISSN 1941-0506.

ZHENG, C.; WEN, L.; WANG, J. Detecting Process Concept Drifts from Event Logs. In: PANETTO, H. et al. (Ed.). **On the Move to Meaningful Internet Systems. OTM 2017 Conferences**. Cham: Springer International Publishing, 2017. (Lecture Notes in Computer Science), p. 524–542. ISBN 9783319694627.

APPENDIX A — <RESUMO EXPANDIDO>

Devido às mudanças nos regulamentos, protocolos, tecnologias, e necessidades dos clientes, os processos de negócio de uma organização devem ser alterados e melhorados regularmente. A disciplina de Gerenciamento de Processos de Negócio (*Business Process Management* - BPM) orienta as organizações a realizar estas mudanças através do ciclo de vida do BPM, no qual os processos de negócio são modelados, analisados, reprojetados e implementados. Pragmaticamente, as melhores implementações de processos de negócio são feitas através de "*Process-Aware Information Systems*" (PAIS), que utiliza a estrutura dos modelos de processo para direcionar o funcionamento da implementação.

Embora seja mais adequado que as mudanças aconteçam primeiro no nível conceitual dos modelos de processo, ocasionalmente estas mudanças ocorrem diretamente no nível operacional das implementações. Consequentemente, os respectivos modelos de processo precisam ser atualizados com as mesmas mudanças realizadas nas implementações. Uma implementação em um PAIS pode auxiliar esta atualização, pois ele gera dados de execução valiosos que podem ser usados na mineração de processos para identificar as diferenças entre a implementação e o modelo. No entanto, nem todas as organizações têm implementações em PAIS, considerando que estas requerem um investimento significativo de recursos e esforços no desenvolvimento de software.

Assim, para garantir que os modelos de processo destas organizações estejam atualizados de forma contínua, é necessário um método que ajude a identificar quando ocorre uma mudança nos processos de negócio e quais modelos de processo precisam ser atualizados. Uma possível abordagem é utilizar fontes de dados heterogêneas relacionadas à execução de um processo de negócio, que podem ser usadas para monitorar quando essa execução não condiz com o comportamento esperado. Exemplos de fontes de dados heterogêneas incluem sistemas externos, recursos, documentos e outros itens utilizados ou produzidos pelos processos de negócio. Chamamos estes itens de entidades observáveis porque os dados deles não podem ser facilmente juntados para que possam ser analisados através da mineração de processos. Nós sugerimos que estas entidades pode ser utilizadas para criar uma abordagem para ajudar na identificação de modelos de processo desatualizados. Esta abordagem precisa um método para identificar estas entidades em modelos de processo e também a criação de um mapeamento entre estes, permitindo que analistas de processo rapidamente identifiquem modelos desatualizados quando as entidades conectadas sofrem mudanças.

Nesta tese, avaliamos a viabilidade de criar esta abordagem. Comparamos diferentes abordagens teóricas de mudança organizacional, de análise, e de redesenho de processo de negócio com uma investigação das mudanças realizadas para atualizar 25 modelos de processos reais. Esta comparação nos guiou para definir uma taxonomia de entidades observáveis que são relacionadas a mudanças em processos de negócio. Esta taxonomia possui três categorias: participantes de processo; sistemas, ferramentas, e tecnologias; e documentos e informações processados. Aplicamos a taxonomia na classificação de 1329 elementos de processo originados de 88 modelos de processo da indústria, assim construindo um conjunto de dados que após foi utilizado para treinar algoritmos de aprendizado de máquina para criar classificadores automáticos de elementos de processo. Com estes classificadores, foi possível analisar quais algoritmos e atributos possuem melhores performances. Destacou-se principalmente o Algoritmo de Classificador de Vetores de Suporte, que atingiu a pontuação-F1 média de 92,4%.

Este método de classificação foi incorporado á nossa abordagem de identificação de modelos de processo desatualizados como meio de identificar as entidades observáveis, assim permitindo a criação do mapeamento entre entidades e modelos. Para nossa abordagem, nós também definimos um conjunto de recomendações que ajudam a manter este mapeamento consistente e a lidar com as atualizações de seus elementos, sendo necessário manter um histórico de versões anteriores e de um conjunto de instruções que definimos para guiar analistas no uso do mapeamento para atualizar modelos de processo quando mudanças são detectadas durante o monitoramento das entidades observáveis.

As principais contribuições desta tese foram o método de classificação de modelos de processo, o conjunto de dados criado por este método, e o classificador automático treinado nestes dados. Os códigos das ferramentas criadas para estes contribuições estão disponíveis publicamente, permitindo que outras aplicações possam ser construídas partindo da metodologia presente nestas ferramentas. Outras importantes contribuições foram os resultados de uma enquete pública realizada para entender como modelos de processos são utilizados na prática e porque eles nem sempre são mantidos atualizados, e também a análise e classificação das mudanças descobertas durante a nossa investigação e atualização dos 25 modelos de processo. Esta classificação, após ser comparada com as perspetivas teóricas, levou a nossa definição de uma taxonomia de entidades observáveis, a qual foi apresentado exemplos claros de como estas entidades estão ligadas a mudanças de processo de negócio.

APPENDIX B — <SURVEY FORM>

Questionário sobre a adoção e uso de modelagem de processos em organizações.

* Required

Universidade Federal do Rio Grande do Sul
Instituto de Informática

Diego Toralles Avila (mestrando)
Prof. Dr. Lucineia Heloisa Thom (orientadora)

Tempo estimado para completar: 10 minutos.

1. Existe alguma iniciativa de modelagem de processos no seu departamento ou na sua organização? *

- ◯ Sim.
- ◯ Não.

2. Como os modelos de processo são utilizados?

- [ ] Eles são usados como documentação.
- [ ] Eles são usados como ferramentas de ensino.
- [ ] Eles são implementados manualmente.
- [ ] Eles são implementados utilizando um sistema próprio da organização.
- [ ] Eles são implementados utilizando um sistema comercial (por exemplo, um BPMS).
- [ ] Other:

3. Os processos da sua organização já foram uma vez evoluídos/mudados por alguma razão? *

- ◯ Sim.
- ◯ Não.
- ◯ Não sei.

4. Se sim, por quais razões?

- [ ] Para aumentar performance.
- [ ] Para reduzir custo.
- [ ] Para se adaptar a mudanças de pessoal (por exemplo, alguem saiu da organização).
- [ ] Para se adaptar a mudanças de responsabilidade.
- [ ] Para atender a novos requisitos (por exemplo, uma nova regulamentação).
- [ ] Other:

5. Como as mudanças são feitas? (Se você não sabe, deixe em branco.)

Your answer

6. Na sua organização, existe algum esforço sendo feito para manter os modelos de processo atualizados? *

- ◯ Sim.
- ◯ Não.
- ◯ Talvez.
- ◯ Não sei.

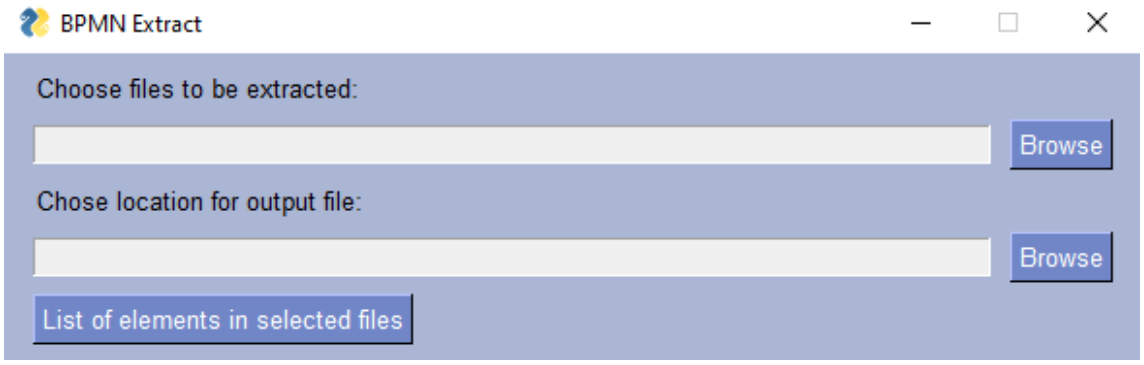
7. Como os modelos desatualizados são identificados? (Se você não sabe, deixe em branco.)

Your answer

Submit

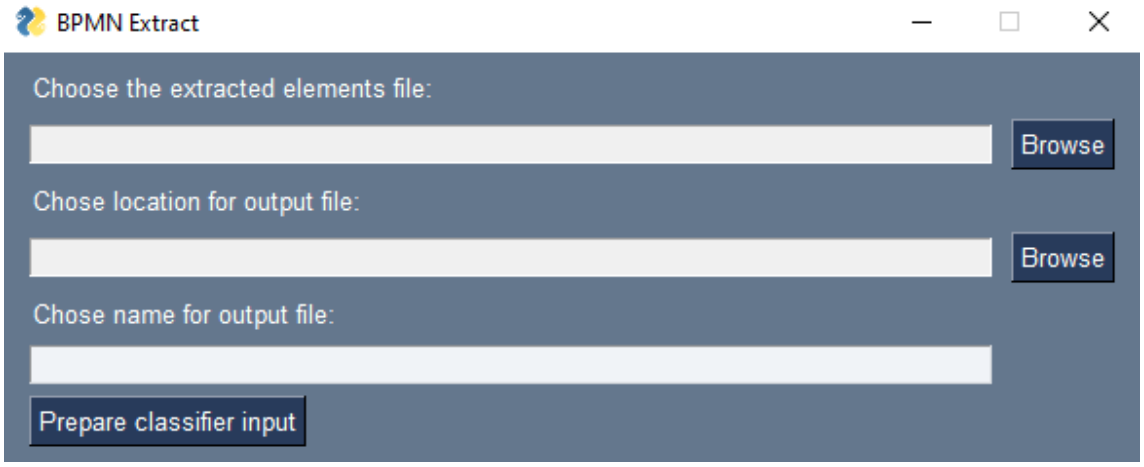
APPENDIX C — <PROCESS ELEMENT CLASSIFIER INTERFACES>

Figure C.1 – Process Element Extractor Interface. It can receive multiple .bpmn process models and it outputs a .csv file listing all process elements.



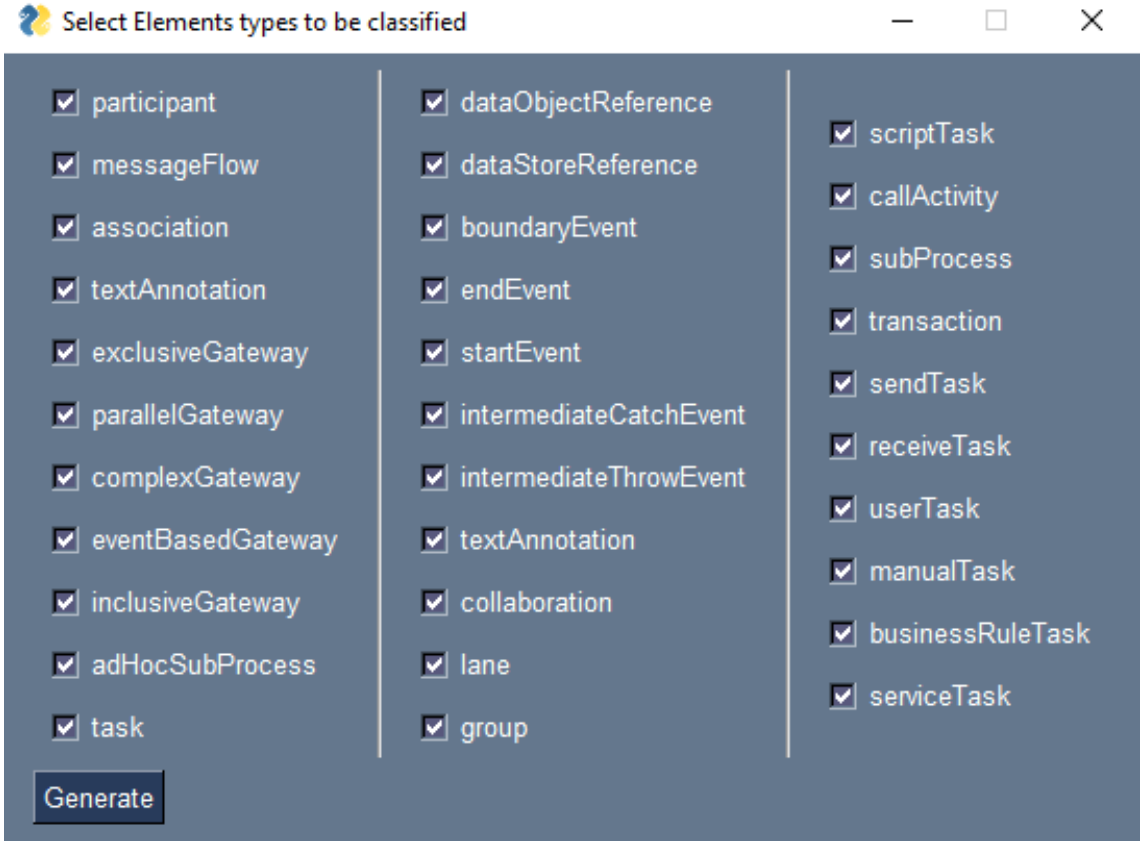
Source: The authors

Figure C.2 – Process Element Pre-Classifier Interface. It receives a .csv from the process element extractor and it outputs another .csv file. The "prepare classifier input" button leads to the interface seen in Figure C.3



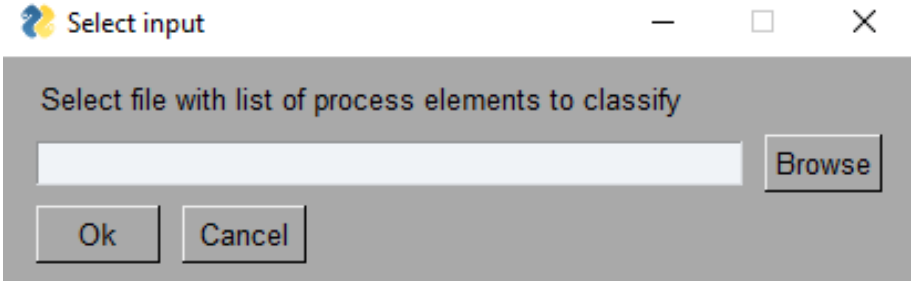
Source: The authors

Figure C.3 – Pre-Classifier Interface for selecting which process elements to filter. It follows the last interface and generates the output when the respective button is pressed.



Source: The authors

Figure C.4 – Process Element Classifier Input Interface. The classifier receives as input the .csv from the Pre-classifer. It can also receive as input a .csv with a process element classification already in progress. Once the input is selected, the next interface is shown in Figure C.5

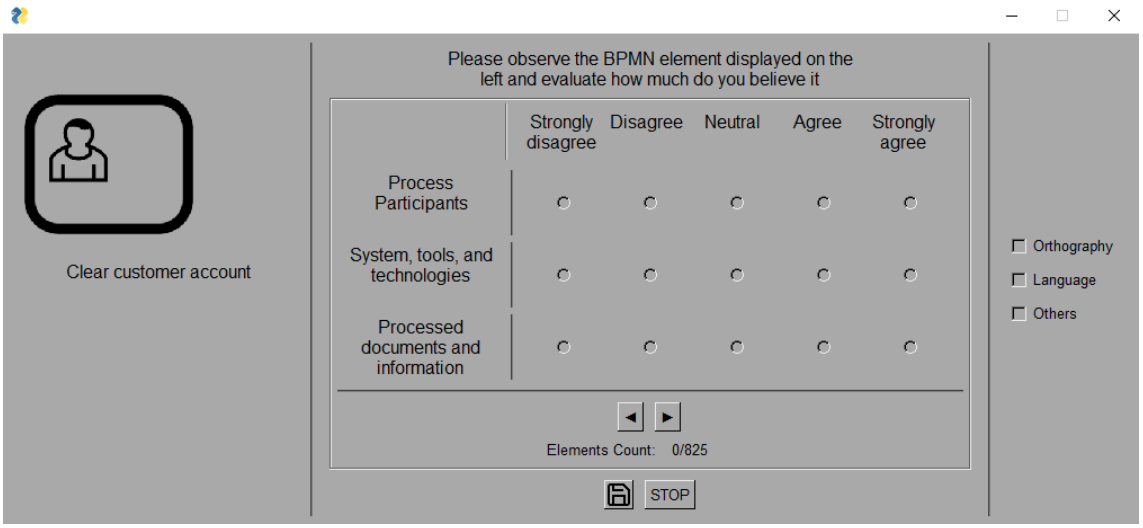


Source: The authors

Figure C.5 – Process Element Classifier Output Interface. This interface only determines the name of the output, which must be a .csv. Once the output is selected, the next interface is shown in Figure C.6

Source: The authors

Figure C.6 – Process Element Classifier Interface. This interface manages the process element classification and its progress for the participants. On the left side, the current process element being classified is displayed. In the center, it is shown the Likert scales for classifying the process elements for the three taxonomy classes. On the right side, there are checkboxes for the classifiers to check whenever they see a process element with problems, such as Orthography and Language. At the bottom, there are the controls for navigating to the previous and the next element, as well as saving and stopping the classification.



Source: The authors