UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

ANA CLÁUDIA DE ALMEIDA BORDIGNON

A systematic literature review on Natural Language Processing in Business Process Identification and Modeling

Work presented in partial fulfillment
of the requirements for the degree of
Bachelor in Computer Science

Advisor: Prof. Dr. Lucinéia Heloisa Thom
Coadvisor: BSc. Renato César Borges Ferreira

Porto Alegre
December 2016

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
Reitor: Prof. Rui Vicente Oppermann
Vice-Reitora: Prof^{a}. Jane Fraga Tutikian
Pró-Reitor de Graduação: Prof. Vladimir Pinheiro do Nascimento
Diretora do Instituto de Informática: Prof^{a}. Carla Maria Dal Sasso Freitas
Coordenador do Curso de Ciência de Computação: Prof. Carlos Arthur Lang Lisbôa
Bibliotecária-chefe do Instituto de Informática: Beatriz Regina Bastos Haro

ABSTRACT

Business Process Management (BPM) has received increasing attention in the recent years. Many organizations have been adapting their business to a process-centered view and fomenting the continuous improvement philosophy, since they began to notice the potential to reduce costs, improve productivity and achieve higher levels of quality. Such organizations have discovered that the implementation of BPM is potentially highly time consuming and expensive and may lead to process models that do not comply with the reality of business. In this context, automation of process identification and discovery and reliable methods to measure and achieve process quality are highly desired. The application of Natural Language Processing techniques to generate process models from unstructured text emerged as an alternative to achieve the expectations. NLP tools are able to decrease the time consumed by process designers and analysts in the process elicitation phase, hence reducing the costs of the process for the company. In this study, a systematic literature review in preparation and processing of natural language text aiming the extraction of business processes and process quality assurance will be provided. The study presents techniques applied to the BPM life-cycle phases of Process Identification, Process Discovery and Process Analysis and tools to support Process Discovery. The results of the present study would be valuable to support research in extraction of business process models from natural language text.

Keywords: Business Process Management. Natural Language Processing. Process discovery. Process analysis. Systematic Literature Review.

Uma revisão sistemática em processamento de linguagens naturais aplicada a identificação e modelagem de processos de negócio

RESUMO

Gerenciamento de processos de negócio (BPM) tem tido um aumento de popularidade nos últimos anos. Muitas organizações têm adaptado sua visão de negócio para uma visão centrada em processos e fomentado a filosofia de melhoria contínua, uma vez que a redução de custos, aumento de produtividade e o aumento do patamar de qualidade se torna notável. Estas organizações têm notado que a implementação do BPM é potencialmente demorada e de alto custo e pode gerar modelos de processos que estão em desacordo com a realidade experenciada pela companhia. Neste panorama, a automatização da identificação e descoberta de processos, da mesma forma que a disponibilidade de métodos confiáveis para mensurar e alcançar qualidade de processos, são extremamente desejáveis. A aplicação de técnicas de processamento de linguagens naturais para geração de modelos de processo usando textos não estruturados como entrada surgiu como uma alternativa para alcançar esses objetivos. Ferramentas de PLN são capazes de diminuir o tempo gasto pelos projetistas na elicitação de modelos de processos, e consequentemente diminuir os custos do processo para a companhia. Neste trabalho é apresentado o estado da arte da preparação e tratamento de texto em linguagem natural visando a extração de modelos de processos e garantia de qualidade de processos. O trabalho apresenta técnicas aplicadas nas fases de Identificação de Processos, Descoberta de Processos e Análise de Processos do ciclo de vida de BPM, assim como ferramentas de apoio à Descoberta de Processos. Os resultados do estudo em questão podem ser de grande valia no apoio a projetos de pesquisa em extração de modelos de processos de negócio a partir de textos em linguagens naturais.

Palavras-chave: Gerenciamento de processos de negócio, BPM, Processamento de Linguages Naturais, NLP, Descoberta de processos, Análise de processos, Revisão literária.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

BPM	Business Process Management
SBVR	Semantics of Business Vocabulary and Rules
NLP	Natural Language Processing
IEEE	Institute of Electrical and Electronics Engineers
BPMN	Business Process Model and Notation
KPI	Key performance indicators
POS	Part-of-speech
UML	Unified Modeling Language
ER	Entity-relationship
SQL	Structured Query Language
ACM	Association for Computing Machinery
TF/IDF	Term Frequency/Inverse Document Frequency
GATE	General Architecture for Text Engineering

CONTENTS

1 INTRODUCTION

Business processes are the basis of organizations. They are the aggregation of all the tasks and activities performed by companies in order to create and provide services. Given the importance of business processes, efforts in process improvement can have a large impact in the performance of the organization, increasing productivity and creating revenue generation opportunities (FARRELL, 2013).

Business Process Management (BPM) is a discipline that focuses on improving processes, in order to reduce cycle time, increase revenue, decrease costs, improve service quality, and so on. In order to implement the BPM philosophy, the company shall recognize its processes and involved resources.

The use of natural language makes it easy for managers and other process participants to define the processes, instead of using modeling languages, since they are probably not familiar with designing techniques. As a result, organizations have a large amount of natural language documents. BLUMBERG and ATRE (2003) states that 85% of the information in companies is stored in an unstructured way, especially as text documents.

1.1 Problem

Modeling of processes is a very time consuming and error prone task. It can be performed by document analysis, logs handling or interviews of process participants (DUMAS et al., 2013). While logs handling can be automated, they generate usually incomprehensible models and depend on specific software to be generated (DUMAS et al., 2013). The other approaches are mainly manual and rely on business analysts.

The documents of an organization and the possible reports of interviews are both natural language texts, hence they require high effort to extract a process model.

In addition to the difficulties in model extraction, the analysis of process models—specially in terms of process model conformity, both with business reality and business policies—needs to deal with the gap between process modeling and process' participants expertise.

Natural Language Processing appears as a solution to both problems: automatically (supervised or unsupervised) extracting process models from natural language text and creating more comprehensible analysis for process' participants.

1.2 Goals and Hypothesis

The goal of this work is to present a systematic literature review on NLP techniques and tools to prepare and process natural language text aiming at the extraction of business processes and process quality assurance.

The systematic literature review would be able to point out the techniques that have been applied to Process Identification, Discovery and Analysis, as well as present tools used during Process Discovery.

1.3 Approach

A Systematic Literature Review protocol was created according to the guidelines in Kitchenham et al. (2009) containing two research questions:

1. *Where is Natural Language Processing being applied in the Business Process Management phases of process identification, discovery and analysis?*
2. *What are the Natural Language Processing tools being used during process discovery?*

The protocol also consists of inclusion and exclusion criteria to help narrowing the results of the research to guarantee the relevance of the selected papers.

The papers selected by the review were analyzed to extract the relevant techniques and their applicability, as well as the used tools. An overview of the results is presented, as well as the applicability context within the studies.

1.4 Related Work

Systematic literature review begun as an evidence-based research approach for medical context, however its popularity increased in other domains over the years. Kitchenham et al. (2009) presents a systematic literature review on tertiary studies, which are other systematic literature reviews, to evaluate the benefits of this review approach in the domain of Software Engineering. This work applies guidelines proposed in Kitchenham (2007), adapted from medical protocols to the domain of Software Engineering.

The systematic literature review performed in the present work is based on the protocol followed by Kitchenham et al. (2009), applied to the area of BPM. In the context of BPM, an important work is VAN DER AALST (2012). Even though the study does not provide a rigorously defined research protocol, it presents a valuable overview of the BPM Conferences throughout almost a decade (between 2003 and 2011). VAN DER AALST (2012) shows statistical data of the researches in many use cases of BPM, such as model design, discover model from event data, check model conformance and so on. Those statistics supported the paper's distribution over the three first BPM life-cycle's phases.

Recker and Mendling (2016) also provided an overview of the BPM Conferences. The study considers conferences from 2003 to 2014 and categorize the results by BPM life-cycle phase. The approach of BPM life-cycle used in the analysis is the same as the adopted in the present work. The results of Recker and Mendling (2016) also support the imbalance in the distribution of the papers within the life-cycle phases.

In the context of NLP, an important study is Yalla and Sharma (2015). It proposes a systematic literature review regarding the combination of NLP and the domain of Software Engineering. Even though the context of the study is broader than the present work, the results presented in Yalla and Sharma (2015) demonstrate that the main uses of NLP in the context of Software Engineering are the generation UML diagrams and process models from natural language text.

The results of Yalla and Sharma (2015) do not refer to the application of NLP techniques to specific steps of the process model generation, or even consider other steps of BPM. Likewise, the other discussed related works do not combine NLP and BPM life-cycle phases either.

The table 1.1 presents an overview of the related works discussed in this section.

Table 1.1: Overview of related works.

Author	*Title*	*Year*	*Overview*
KITCHENHAM et al.	Systematic literature reviews in software engineering – A systematic literature review	2009	Tertiary study of evidence-based software engineering systematic literature reviews.
VAN DER AALST	A Decade of Business Process Management Conferences: Personal Reflections on a Developing Discipline	2009	Overview of the BPM Conferences between 2003 and 2011.
YALLA; SHARMA	Integrating Natural Language Processing and Software Engineering	2015	Systematic Literature Review on the combination of NLP and the domain of Software Engineering.
RECKER; MENDLING	The State of the Art of Business Process Management Research as Published in the BPM Conference	2016	Overview of the BPM Conferences between 2003 until 2014.

Source: the authors

1.5 Organization

This section presents the organization of the sections of the study and its content. The Section 1 introduces the study, presenting the problem it intends to solve, the goals and the approach used. Section 1 also presents an overview of related work. The Section 2 presents the necessary background for the study. It consists of an overview of BPM, the division of BPM within life-cycles and explanation about the three life-cycles that this study approaches: Identification, Discovery and Analysis. In the same section, an overview of NLP is presented, discussing the three steps of text processing: morphological, syntactic and semantic analysis. At the end of the section, it presents a brief exposition of the application of NLP to the domain of BPM. In the Section 3, the Systematic Literature Review protocol is introduced. It contains the research questions, the selection criteria and the research procedures. The results of the Systematic Literature Review are presented in Section 4. This section presents an overview and metrics of the results and thereafter an analysis to answer the proposed research questions. The last section of the study presents the conclusion.

2 BACKGROUND

In this section, the necessary background to understand this work is provided. First an overview of BPM and BPM life-cycle, focusing on the first three phases of the life-cycle—Process Identification, Process Discovery and Process Analysis—, followed by an overview of Natural Language Processing and some techniques and tools.

2.1 Business Process Management

In 1970, the third Industrial Revolution started. Unlike the second one, the third is represented by services as products, instead of physical appliances. Likewise, the method to manage companies and production chain changed to adapt to the market new moment (VAN DER AALST; LA ROSA; SANTORO, 2016). Formerly, the production chain was entirely performed by one single agent. This agent had the view of the entire process. Considering the example of coffee manufacturing, we can consider that the same agent was responsible for planting the coffee tree, reaping the beans and manufacturing them, the same way, this agent was also the consumer.

In the course of time, specialized jobs started to emerge. In this new phase, different agents were responsible for diverse phases of coffee production. Following the previous example, and the final product could be consumed by an external agent. The production process can easily be represented as a workflow, considering a feedstock as input, production steps, and a product as output.

As production techniques improved, agents started to have even more specific tasks within the process, having only the view of a small part of the full chain. They had knowledge only about their own tasks and not about the whole process. The development of an environment composed of very specialized agents raised the necessity to create functional and agile communication between the parts, which can be a challenge as agents have diverse backgrounds.

BPM emerged to cope with the communication difficulties, as well as to formalize and control processes in a company in the most reliable and optimized way possible. While the automation appeared as an approach to increase productivity in physical process chains, BPM emerged as an alternative to improve the administrative processes involved in these production chains. As BPM started, many different authors and researchers from diverse backgrounds (namely Business, Information Technology and Industrial Engineering) studied and wrote about it, for this reason, many definitions and clarifications may be found (DUMAS et al., 2013). One of the definitions, upon which The Workflow Management Coalition (COALITION, 2015), BPM.com (PALMER, 2014) and The Complete Business Process Handbook (ROSING; SCHEEL; SCHEER, 2014) have agreed, that BPM is “a discipline involving any combination of modeling, automation, execution, control, measurement and optimization of business activity flows, in support of enterprise goals, spanning systems, employees, customers and partners within and beyond the enterprise boundaries” (PALMER, 2015). In summary, DUMAS et al. defines BPM as “the art and science of overseeing how work is performed in an organization to ensure consistent outcomes and to take advantage of improvement opportunities”.

The motivation to adopt BPM in an organization relies on the process improvement benefits. A better process reduces costs, increases quality, motivates employees and produces higher revenues (RUDDEN, 2007). A use case of BPM in the area of telecommunication service claimed to reduce in 10% the costs per quarter with the help of the BPM deployment to identify duplicate issues, research disputes more completely, and enforce more consistent payout policies (RUDDEN, 2007).

As an improvement discipline, BPM inherits many of its characteristics from other disciplines, such as continuous improvement from Total Quality Management, principles and techniques of operations management from Lean and Six Sigma, and applies the infrastructure and administration software systems that information technology provides (DUMAS et al., 2013).

2.1.1 BPM Life-cycle

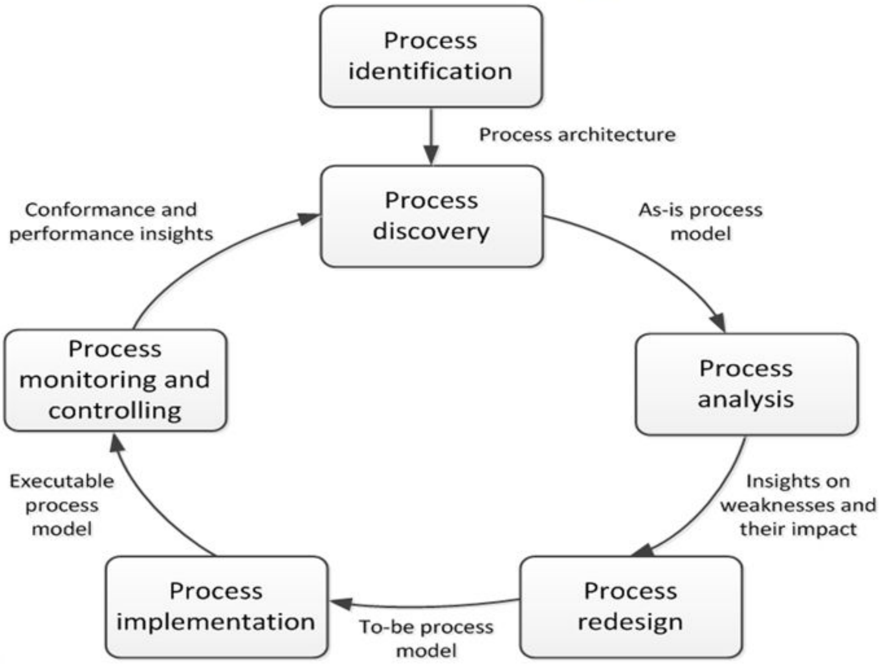
Some of the most recognized approaches in the area of BPM were proposed by Mathias Weske (WESKE, 2007), Wil van der Aalst (AALST, 2011) and Marlon Dumas (DUMAS et al., 2013). Their approaches differ mainly in the division of the process life-cycle. In this work, the approach considered is that from Marlon Dumas (DUMAS et al., 2013), the life-cycle of which is follows:

- Identification: the actions, actors and events participant of the process are identified;
- Discovery: the result of the first phase is better explained, adding details;
- Analysis: the process is analyzed and the issues are pointed out;
- Redesign: the process is remodeled according to the issues found in the analysis

and the proposed solutions are incorporated;

- Implementation: the reviewed process is actually implemented;
- Monitoring and controlling: the process' results are evaluated and compared to the expectations.

Figure 2.1: BPM Life-cycle according to Dumas.



Source: (DUMAS et al., 2013)

We selected Dumas' approach in our work because it is based on Weske's and Van Der Aalst's approaches, however splitting the life-cycle in a more didactic view (DUMAS et al., 2013). This approach is also the newest one, which is pertinent in a research field as mutable as BPM.

This project will focus on the first three phases of the BPM life-cycle. They will be fully explained in the following sections.

Process Identification

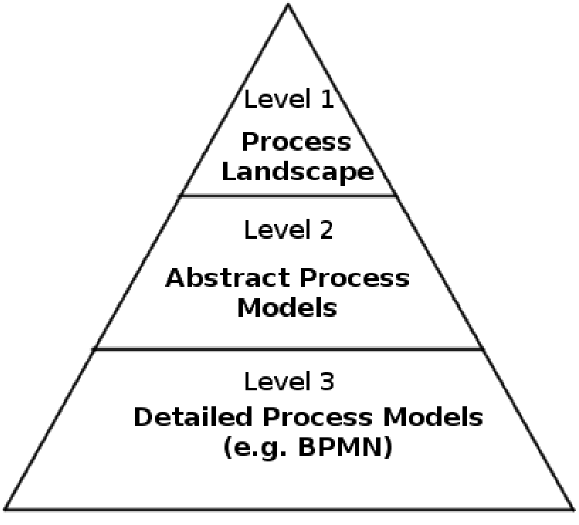
BPM implementation involves many costs in software, people and hardware (RUDDEN, 2007). Before starting efforts to improve processes in a company it is necessary to prioritize the involved processes by considering which ones will result in a greater improvement according to the company goals. The first step is to recognize all the processes performed in an organization (DUMAS et al., 2013). Organizations that already implemented BPM before probably recognized their activities previously, nevertheless the processes can change over time due to variations in the way the company works or market transitions, so the set of processes should be updated before starting a new cycle of BPM.

After recognizing all the proceedings involved in a business, the next step is to prioritize them. There are no general guidelines for choosing focuses of improvement, it depends on the organization's goals, budget and deadlines. It is not a simple task to identify the parts of the whole process where an improvement is necessary or where it is possible to achieve better results with less effort—considering effort as time and money spend, employees' engagement and changes performed. For this reason, many criteria were proposed to guide the evaluation, were the most commonly used are:

- Importance: how important the present process is to the achievement of the organization's goals;
- Dysfunction: how problematic is the process;
- Feasibility: how susceptible the process is to process management initiatives.

The output of the identification phase is a process architecture, which consists of a conceptual model that includes the processes of a company and the relationships between them. These models are composed by three levels. The first level is composed by the main process presented very abstractly, each element of the first level points to a more concrete process in level two, with finer granular representation. Likewise, each element of level two points to a process model in level three, when they are represented in details. Figure 2.2 represents the organization of the different levels of the process architecture.

Figure 2.2: Process architecture according to Dumas.



Source: (DUMAS et al., 2013)

Process Discovery

More than modeling a process, the discovery phase comprehends the gathering and organizing of information about a process, generating an as-is process model. These goals are achieved by performing the following steps:

1. Assembling a team to work on the process;
2. Acquiring information about the process;
3. Organizing the acquired information aiming at the process model creation;
4. Assessing the resulting model (DUMAS et al., 2013).

The first step is to define the setting of the processes, which includes assembling one or a few process analysts and sufficient domain experts to aggregate knowledge to approach the relevant perspectives of the process. The process owner also should be involved in the discovery phase and guarantee the commitment and involvement of the assembled team.

The motivation for having both process analysts and domain experts is that analysts have the modeling skills and the knowledge of tools to functionally construct the model, but they do not have knowledge about the operability of the process. On the other hand, domain experts have profound knowledge about the business process, but usually are not familiar with modeling techniques and languages (like BPMN).

The main challenges of the discovery phase are that each domain expert has knowledge about some parts of the process, but not all of it, and tend to think about the process in terms of specific use cases instead of generally. As well as, the already mentioned lack of familiarity with modeling tools is a great challenge.

Process discovery demands many iterations to gather all the information needed. The knowledge of the experts might be divergent due to personal view and the analysts may need to talk with them several times to resolve conflicts within the specification of the relevant tasks. Also, since domain experts tend to see the process in terms of specific cases, it is necessary to generalize the conclusions to all cases. They may have difficulties answering general questions, so it is a duty of process analysts to organize and abstract the necessary information from the reports (DUMAS et al., 2013).

The gathering of information can be performed using different techniques that can be organized in three classes (DUMAS et al., 2013):

- Evidence-based: based on documents, reports and observations;
- Interview-based: based on personal interviews, one expert at a time;
- Workshop-based: based on interviews that include all experts at once.

Evidence-based approaches can provide valuable information about the process, however they depend on the quality of the documents. Many business documentations are not process-centered, they may be even outdated or do not reflect reality accurately. When a company owns rich documentation, this method is very objective, providing information without ambiguity. An improved approach in terms of time consumed is the use of automatic techniques (DUMAS et al., 2013). It is based on reports generated by information systems with a defined format and generate a model in short time, but not always comprehensible (DUMAS et al., 2013).

Another alternative is to employ observation, which tends to be more reliable than documentation and more comprehensible than automatic generation. The analyst can perform as an active costumer or a passive observer, and, depending on the choice, perform the process (such as a book sale) as a costumer or be inside the company and observe the process from within the process. The passive observer role is more appropriate to follow the entire process, however it demands permissions from the managers of the organization.

Both interview-based and workflow-based approaches may provide rich and detailed pictures of the process. However, interviews may generate biased views, since domain experts see the process from their point of view only. Workshops can avoid that biased view, notwithstanding they may be very time consuming since it is difficult to schedule meetings including all the participants.

Every technique has its strengths and weaknesses, as shown in Table 2.1, so the usual approach is to combine them to fit the process goals as well as possible.

Table 2.1: Strengths and limitations of process discovery methods.

Aspect	*Evidence*	*Interview*	*Workshop*
Objectivity	high	medium-high	medium-high
Richness	medium	high	high
Time Consumption	low-medium	medium	medium
Immediacy of Feedback	low	high	high

Source: (DUMAS et al., 2013)

Once the necessary information is gathered, it is possible to start modeling the process. First of all, the boundaries of the process should be identified. They were already partially detected during the creation of process architecture, however it is necessary to recognize the events that trigger the process and the ones that can lead to outcomes. After recognizing the boundary events, the rest of the events and activities which are part of the process and the person who is responsible for them should be recognized as well. Having this knowledge, it is possible to define the pools and lanes of the process and determine the handover points. The definition of the handover points creates an initial structure to the process sequence. To generate a complete process flow, it is still necessary to determine the order dependencies, decision points, concurrency and loops. Whereas the model is essentially done, the last step is to add possibly useful additional information, such as exception handlers and data stores (DUMAS et al., 2013).

When all the steps are accomplished, the model is created and may be verified according to the syntax of the chosen process modeling language and the real-world domain it is applied to.

Process Analysis

After modeling the process, it should be analyzed both qualitatively as quantitatively. Qualitative analysis has no rules, but principles, since it has more than one single way to produce a good analysis (DUMAS et al., 2013). It aims at analyzing the impact of issues in order to prioritize them for redesign purposes. Quantitative analysis uses metrics to evaluate the compliance of the business process model with the company goals, such as cycle time, waiting time and cost. Qualitative analysis is based on:

- Value-added analysis: it decomposes tasks into steps and analyzes each one of them to check if they have positive outcomes for the client (considered as value adding), if they are necessary for the business but do not provide positive outcomes to the client (business value-adding) or otherwise (non-value adding). The objective is to eliminate the non-value adding, ponder the necessity of the business value-adding considering the efforts they demand and keep the value-adding steps;
- Root cause analysis: investigate root causes of undesired events or issues;
- Issue documentation and impact assessment: systematically document issues, causes and unexpected events in order to prioritize them and aggregate higher improvements to the process as possible (DUMAS et al., 2013).

Quantitative analysis can be measured by key performance indicators (KPI), which are unambiguously determined measured quantities that reflect the state of a business process (WESKE, 2007). The measure's dimensions of interest are:

- Time: usually related to cycle time, it includes all the time measurements related to a business process, such as processing time and waiting time;
- Cost: often related to reducing costs, but can also refer to yield and revenue;
- Quality: can be measured from an external point of view (client's satisfaction) or internal (process participant's point of view);
- Flexibility: based on the ability to react to changes, even internally to the process or externally.

The analysis of the metrics determines whether the process is within the expectations of the company, while the qualitative analysis points to improvements that are feasible and desired (DUMAS et al., 2013).

2.2 Natural Language Processing

A language developed naturally without planning and modeling is considered a Natural Language. A well known example is the English language. A Natural Language could be considered as the counterpart of a computing code (LYONS, 1993).

Fifty years ago, Alan Turing presented the Turing Test, which consists of a text conversation between a human evaluator and two others interlocutors, where one is human and the other is a machine, and define a machine as intelligent if the evaluator can not tell the machine from the human (TURING, 1950). The approach to achieve this intelligence is Natural Language Processing, a sub-area of Artificial Intelligence. In conclusion, Natural Language Processing is responsible for, automatically, creating and processing human natural language.

Jurafsky and Martin (2009) define the objective of Natural language processing as "to get computers to perform useful tasks involving human language, tasks like enabling human-machine communication, improving human-human communication, or simply doing useful processing of text or speech"

Given the importance of the subject, the complexity of natural languages—ambiguous nature and inter-domain use—and the large amount of problems it may cover, many techniques were developed to handle this complexity over the years, such as statistical models and rule-based semantics (ROTH, 1998). Different methods are commonly applied to specific kinds of applications, such as text summarization using weighting schemes, machine translation applying statistical machine translation (Google Translate is based on it) and BPM's process standardize using syntactically driven parsing.

The analysis of a natural language text may have three levels:

- Morphological analysis is focused in the words' structure (JURAFSKY; MARTIN, 2009);
- Syntactic analysis deals with the relationships between words in a sentence, deciding which classification group the word belongs to, according to a grammar (INDURKHYA; DAMERAU, 2010);
- Semantic analysis is built upon the results of the other two levels. It aims at defining words and sentences meaning based on the knowledge of their structure, relationships and role (INDURKHYA; DAMERAU, 2010).

During these levels, many techniques and tools can be applied by specialists. Those techniques and tools are going to be introduced according to the level they are applied to in the next subsections.

2.2.1 Morphological Analysis

As the first level applied to a raw natural language text, this step directly influences the results of the following levels. Hence, the good choice of a method to perform morphological analysis is highly important.

As stated before, morphological analysis aims at handling words' structure. It considers morphemes—the smallest meaning-bearing units in a language—and the combination of them to recognize plurals, inflections, genders and other variations (JURAFSKY; MARTIN, 2009).

During the analysis of the papers by this systematic review, only two techniques were found: lemmatization (JURAFSKY; MARTIN, 2009) and word stemming (JURAFSKY; MARTIN, 2009). Both techniques have the same purpose—to reduce the amount of variations of terms that represent the same meaning, and potentially the same entity—. The difference between the two techniques is the approach, while Word stemming only extracts the word root by handling suffixes, Lemmatization uses a vocabulary (MANNING; RAGHAVAN; SCHÜTZE, 2008).

A tool called RSLP (*Removedor de Sufixos da Lingua Portuguesa*) Stemmer was used in one of the papers to perform word stemming for Portuguese language. Since stemmers are completely language dependent, they cannot be used with English language. The RSLP Stemmer was presented in the paper *A Stemming Algorithm for the Portuguese Language*, by Viviane Moreira Orengo and Christian Huyck. It is based on rules according to the grammar of the Portuguese language (ORENGO; HUYCK, 2001).

The other papers that perform Word Stemming do not mentioned any tool. Even considering that they apply Stanford Parser, for example, and the parser is able to perform Word Stemming, it is not mentioned whether it is used also with this purpose or only to perform other techniques. The same way, Stanford Parser (GROUP, 2015b) can perform Lemmatization, however only one of the papers analyzed in this study mentioned the use of this functionality, while one explicitly uses WordNet Lemmatizer (UNIVERSITY, 2016) and the rest does not clarify which tools they apply with this purpose.

Stanford Parser is a probabilistic parser. It uses hand-parsed sentences to acquire knowledge and then parse new sentences automatically. The tool was created by Stanford University collaborators, first released in 2002, and includes not only the function of lemmatization, but also tokenization, dependencies parsing, text annotation, part-of-speech tagging, semantic tagging, lexiconization and so on (GROUP, 2015b). Since the parser includes many functions, it is used in all the analysis levels.

The other applied tool is WordNet Lemmatizer (UNIVERSITY, 2016), which is part of NLTK Parser that also provides tokenization, word stemming, lemmatization, part-of-speech tagging and semantic parsing. The NLTK Parser was developed within the Department of Computer and Information Science at the University of Pennsylvania in 2001 (COMPUTER; PENNSYLVANIA, 2015).

2.2.2 Syntactic Analysis

The second level of analysis focuses on analyzing the structure of a string of words, instead of one single word, according to a grammar. Syntactic analysis aims at generating a hierarchical structure of tagged words that represents the relationships between the words in a sentence/fragment and is suitable for further semantic processing.

Three main techniques were found during the analysis of the papers selected by the systematic review: Tokenization, Part-of-speech tagging and Shallow Parsing.

Tokenization divides a text into tokens, which are fragments selected as useful units for semantic parsing. In the case of natural language text applied to our context, it is convenient to consider words of the text as tokens (MANNING; RAGHAVAN; SCHÜTZE, 2008). Tokenization is an introductory step to later categorize the words of sentences. To perform sentence segmentation, the set of tools found in the papers analyzed in this study include Stanford Parser, Stanford Tagger, NLTK Tagger, PunktWordTokenizer, SUNDACE, GATE and WordPunctTokenizer.

Similarly to the Stanford Parser, already introduced in the section Morphological Analysis, SUNDANCE and GATE are a suite of many natural language processing functions and provide solutions to all the relevant problems in this section. SUNDANCE was developed by the University of Utah firstly as a Shallow parser, but more functions were added along the development (RILOFF; PHILLIPS, 2004). GATE was developed by the University of Sheffield as a solution for both academic and industrial sectors (SHEFFIELD, 2016).

PunktWordTokenizer is one of the solutions within the NLTK Parser. It creates a list of sentences from a text, applying rules learned from the set of texts manually annotated in the target language. WordPunctTokenizer is also part of the NLTK Parser, but differs from the PunktWordTokenizer by segmenting the text into tokens separated by spaces, without the necessity of a training section (COMPUTER; PENNSYLVANIA, 2015).

After performing sentence segmentation, the approaches present in the studies perform Part-of-speech tagging (POS). POS annotates each of the tokens with information about its grammatical class, such as noun, adjective, verb and so on (MANNING; RAGHAVAN; SCHÜTZE, 2008). Two concepts of POS application were utilized on the papers, some of them used a tagger already trained with a general Corpus, others decided to train their own taggers.

The first concept of POS application considers mainly the already mentioned parsers, such as Stanford Parser and NLTK Parser, including Stanford Tagger and NLTK Tagger which for the purpose being address represent the same processing features (GROUP, 2015a). In addiction to the parsers, a general training Corpora is used, such as WordNet, FrameNet, BabelNet, and VerbNet.

The second concept considers the application of statistical taggers in combination with lexicons. The approach uses n-gram taggers, which are statistical learning techniques, as well as the presented parsers, but they have specific granularity. For example, a Trigram Tagger considers sets of three words in the evaluation of POS, Bigram considers two words sets and Unigram, or bag-of-words, considers only one. All those methods apply training Corpora as well, but in this case, the annotated texts are provided according to the application domain of the study. The authors of the considered papers in this study used a method called lexiconization to create the training Corpora, which consists of creating a list containing all the words in the language (considering language as the domain language) (JURAFSKY; MARTIN, 2009).

The last technique found during papers' analysis is Shallow Parsing. It can be considered as a step between a Part-of-speech tagging and a parsing. Shallow Parsers provide more information about sentence structure than POS, however do not provide a whole parse tree, such as NLP parsers (RAMSHAW; MARCUS, 1995).

2.2.3 Semantic Analysis

The last level of Natural Language text analysis is Semantic Analysis. It may be confused by lexical analysis, however they differ mainly in the presence of undefined context. Lexical analysis deal with meaning of words and predefined word combinations, while semantic analysis aims at handling an undefined number of combinations of words possible within a grammar. In this study, the adopted approach is a growing tendency that considers lexical analysis and semantic analysis as part of the same group, since their field of study overlaps significantly (FRIED; ÖSTMAN, 2004).

The techniques found during analysis of papers to perform Semantic related analysis were word-sense disambiguation, semantic accommodation and Anaphora resolution. The first one consists of examining words in context and deciding in which possible sense of the word is being used (JURAFSKY; MARTIN, 2009). Many of the authors of the analyzed papers did not point out the techniques or tools they used, only one technique was mentioned: enumerative approach. Proposed by Navigli (2009), the method uses as an input a vector containing all the occurrences of each of the senses a word has in the context (it uses BabelNet to list the senses) and processes the input constructing a graph which will be clustered by XMeans and used to identify similar clusters. Where XMeans is a tool to automatically identify clusters based on BIC Scores (PELLEG; MOORE, 2000).

The second technique combines the semantic structures of the fragments of a sentence into a unified structure, defining the sentence meaning. The semantic structure of the fragments is extracted using a lexicon (SELWAY; MAYER; STUMPTNER, 2013). The method applies patterns and rules, and is sometimes called Rule Based Semantic

Analysis (AHMAD; RIYAZ, 2013).

A great challenge in the processing of natural language text is to determine the relationship between an entity and the referring terms used afterwards. The referencing to a previously declared entity is called Anaphora (JURAFSKY; MARTIN, 2009). To deal with this challenge, the third technique is introduced.

The literature identifies three types of anaphora: pronominal, definite noun phrase and one-anaphora, but only the pronominal anaphora is being addresses in this study, since the others are highly more complex (MITKOV, 1999). The approaches to perform Anaphora Resolution consider constraints. Those constrains can refer to gender, number, semantic role, and so on.

2.2.4 Auxiliary techniques

In addiction to the techniques already discussed, there are methods that can be applied to many tasks. For the purpose of this study, they are going to be called Auxiliary Techniques. The techniques can be used to support the other techniques mentioned under the sections Morphological Analysis, Syntactic Analysis and Semantic Analysis.

A very useful and commonly applied technique is Pattern Matching. It consists of interpreting the whole input as a unique structure, instead of constructing the interpretation by analyzing the structure and meaning of the constituents (HAYES; CARBONELL, 1983). Pattern Matching is frequently implemented as Regular Expressions, which are formulas defined according to a specific language that aim at specifying classes of strings (JURAFSKY; MARTIN, 2009).

Another frequent implementation of Pattern Matching is in the format of Templates. They are a set of rules and patterns that aim at creating a piece in the final language by filling the empty spaces within the rules with the fragments of the input (DEEMTER; KRAHMER; THEUNE, 2005). Templates are one of the concepts adopted in the implementation of Statistical Learning methods (ROTH, 1998).

The two techniques presented in the previous paragraph support natural language text processing in the way of filtering and representing the input, however they do not deal with the large amount of data a text processing may generate. To address this issue, a valuable supporting technique is Word Clustering. The technique focuses on creating groups of unlabeled data in a way that they are similar within the groups and dissimilar otherwise (INDURKHYA; DAMERAU, 2010). Word Clustering supports many other mentioned techniques, such as Part-of-speech tagging, lexiconization, and so on (JURAFSKY; MARTIN, 2009). Word Clustering is used to help identifying ambiguities and to map entities of the language.

2.3 Natural Language Processing in the context of Business Process Management

NLP techniques are, generally, used in the context of BPM with two main purposes: to extract useful information from external texts to increment or infer a process model, and to extract information from the process model to facilitate analysis (LEOPOLD, 2013).

The application of NLP to external texts mainly aims at automatically generating conceptual models, conceptual dependency diagrams, entity-relationship models and UML diagrams. Many other studies—such as Richards, Fure and Aguilera (2003), Lahtinen and Peltonen (2005) and Bolloju, Schneider and Sugumaran (2012)—aim at supporting the process model designer providing tools to expose multiple viewpoints of the process, proposing business rules or indicating inconsistencies.

The application of NLP to process models, on the other hand, aims at improving quality by forcing naming convention and terminology, checking for semantic error and creating human readable texts to help the process of compliance check. The use of conceptual models as inputs presents different challenges than the use of natural language text, since they have different (and usually incomplete) grammatical form.

Leopold (2013) presented an overview of the application of NLP techniques in BPM. Table 2.2 summarizes the results shown in the mentioned study. The left-size column of the table represents the input format to the applications, while the applications are grouped by categories (in bold) in the right-side column.

Table 2.2: Applications of NLP in BPM.

Input	*Application*
Natural language text	**Construction of Models**
	BPMN Model from Text
	Dependency Diagram from Text
	ER-Model from Text
	UML Model from Text
	Conceptual Model from Requirements
	Designer Support
	Visualization of Use Case Descriptions
	Consistency of Object Models
	Speech Recognition for UML
	Construction of Formal Specification
	SQL Statements
	Relational Algebra
Business Process Model	**Quality Assurance**
	Term Inconsistency Detection
	User Support
	Linguistic Consistency Checking
	Naming Convention Enforcement
	Reducing Linguistic Variations
	Semantic Annotation
	Detection of Semantic Errors
	Generation of text
	Generation from UML Diagrams
	Generation from Process Models
	Information Elicitation
	Service Discovery
	Detection of Process Patterns Similarity Measurement
	Process Activity Mappings

Source: adapted from (LEOPOLD, 2013)

3 SYSTEMATIC LITERATURE REVIEW

A systematic literature review is a method of research that aims at answering a defined set of research questions by collecting and summarising empirical evidences (primary studies) (EDINBURGH, 2013). The difference between a systematic literature review and an ad-hoc review is that the first one is rigorously defined in terms of methodology (KITCHENHAM et al., 2009). The definition of the protocol reduces the probability of generating a biased result, since the protocol is defined before the beginning of the collection of paper candidates, however it is not able to avoid publication bias (KITCHENHAM, 2007).

The systematic review performed in this work is based on the protocol proposed in Kitchenham (2007), which is composed by:

- Identifying the necessity of a research in the subject;
- Defining the research questions to lead the review;
- Composing a review protocol containing the research questions, inclusion and exclusion criteria, selection procedure, synthesis of the data;
- Evaluating the review protocol by an expert;
- Selecting primary studies;
- Assessing the primary studies;
- Extracting and synthesizing the data (KITCHENHAM, 2007).

The proposed protocol also includes a research commissioning step, which is not applicable to this work since this step is focused on organizations requiring a third party research.

3.1 Review Protocol

This chapter explains the protocol applied during the systematic literature review. The review aims at analyzing the use of Natural Language Processing techniques and tools during the first three phases of BPM life-cycle, pointing which activities are performed to achieve the objectives in text handling, process elicitation, compliance check, process quality analysis, and so on.

3.2 Research Questions

The research questions aim at narrowing the results to a set of relevant content to represent the state-of-the-art in NLP techniques applied to Process Identification, Discovery and Analysis.

To compose the questions, relevant papers and books about BPM were considered, specially the ones which focus on NLP, and a list of relevant terms was created. Using this list, alternative research queries were created and then tested applying them to the selected databases that would be presented as part of the review protocol.

To test the queries, they were applied in each of the academic databases (presented in a section below) and the results were analyzed using a sampling of 10 studies, selected randomly to check the relevancy of the studies to the research topic and adapted to include results within the defined context. After the tests, the chosen question was selected to guide the systematic literature review:

1. *Where is Natural Language Processing being applied in the Business Process Management phases of process identification, discovery and analysis?*
   This question aims at indicating how extensively is NLP being used within the BPM life-cycle, specifically in the first three phases. This analysis may show tendencies in terms of NLP methods being applied more frequently to support BPM and BPM steps being studied in the last 7 years. As a result, it may be possible to identify aspects which could be further studied.
2. *What are the Natural Language Processing tools being used during process discovery?*
   The second question restricts the results in order to find which NLP tools are being used to achieve the motivational goal described in the introduction. This question intends not only to discover suitable tools to support the proposed issue but to point possible lack of research in the subject.

3.3 Academic Databases

In order to reach as many relevant papers as possible, four well-known academic papers database were used:

- IEEE Explorer;

- ACM Digital Library;
- Springer Link;
- Scopus.

Due to the diverse search engines properties, a particular search query was proposed to each database, as follows:

- IEEE Explorer[1] recommends using few words and short expressions. In fact, it does not work properly with many disjunction terms. Due to that, a concatenation of the results of all possible variations of one term of each column of the Table 3.1 related to the terms of the other columns by and AND operator was considered:

  Table 3.1: Elements of the research query for IEEE Explorer

BPM	NLP	Identification
"Business Process"	"Natural Language Processing"	Elicitation
		Discovery
		Modeling
		Analysis

  Source: the authors

  An example of query extracted from Table 3.1 is: “Business Process” AND “Natural Language Processing” AND Discovery;
- ACM Digital Library uses a different notation for query operators, in which “+” indicates a mandatory term and blank spaces between terms represent an OR operator, considering the fact the query used was: *+(bpm "business process" "conceptual model") +(nlp "natural language") +(identification elicitation discovery modeling analysis)*;
- Springer Link and Scopus use similar query syntax and the query used was: *(bpm OR "business process" OR "conceptual model") AND (nlp OR "natural language") AND (identification OR elicitation OR discovery OR modeling OR analysis).*

The research query aims at reaching the papers that discuss the applicability of NLP techniques and tools in the first three phases of BPM life-cycle. It does not exclude papers that discuss the use of other methods besides NLP, since they are both used in some step of the process.

[1] IEEE search guidelines are presented in http://ieeexplore.ieee.org/Xplorehelp/#/ieee-xplore-training/user-guides.

3.4 Selection of Papers

The selection of the paper candidates was performed by applying inclusion and exclusion criteria. This step in the systematic literature review is important to filter the set of results to obtain only papers relevant to the research context. The inclusion and exclusion criteria are a set of characteristics desirable and undesirable, respectively, of a primary study. The criteria help to guide the selection of studies (KITCHENHAM, 2007).

3.4.1 Inclusion criteria

The inclusion criteria were selected focusing on sorting the candidate papers based on their content and how related they are to the subject of the research. The considered inclusion criteria are the following:

IC-1 the paper approaches NLP in a way directly related to the main scope of the study, instead of being only mentioned;

IC-2 the paper approaches BPM in a way directly related to the main scope of the study, instead of being only mentioned;

IC-3 the paper approaches both NLP and BPM as directly related to each other;

IC-4 the paper presents an application of NLP techniques and/or tools to BPM;

IC-5 the paper approaches the first three phases of BPM: identification, discovery and/or analysis.

The method used to verify the Inclusion Criteria compliance was composed by the following steps, where the results of one step are the input to the next step:

1. The set of studies selected by the queries was analyzed in terms of title and author keywords of each piece. The focus was on expressions identical to the ones presented in the search query or related to them;
2. The abstracts were read to check if the pertinent expressions were related to the main scope of the study;
3. The conclusions were read to ensure the relation of the expressions to the scope of the study.

3.4.2 Exclusion Criteria

The exclusion criteria were designed to exclude undesirable papers in terms of format and publication details, instead of their content. The selected exclusion criteria are:

EC-1 the paper has less than 4 pages, considered as a not full paper;

EC-2 the paper is not written in the English Language;

EC-3 the paper is published before 2009;

EC-4 the work or study is not considered a scientific article;

EC-5 the paper is not a primary study.

The exclusion criteria were applied by filtering the set of results from the application of the search queries. EC-1, EC2 and EC-3 were applied by using the filters that the academic databases provide. The EC-4 and EC-5 were applied by analyzing the publication and format of the paper.

4 RESULTS ANALYSIS

In this chapter the results of the systematic literature review will be presented and analyzed. The systematic literature review was performed in two steps:

1. Search by applying the research query in each of the academic databases;
2. Manual application of inclusion/exclusion criteria to the previous step's resulting in the set of papers.

Table 4.1 shows the set of papers resulting of each step:

Table 4.1: Selection of papers

Step	*Set of Papers*
Query application + Exclusion criteria	518
Title + Author keywords	151
Abstracts	65
Conclusions	49
Final set	49

Source: the authors

The final set of papers is considered as containing only relevant content to the research, although during the reading of the articles, 16 of them were considered as not relevant. The issues found are presented in the table 4.2:

Table 4.2: Not relevant papers

Issue	*Set of Papers*
Manual handling	5
Not covered BPM life-cycle phase	3
Not applying NLP	3
Lack of details	2
Applications overview	2
Support to other papers	1

Source: the authors

As shown in Table 4.2, the category which comprehends the majority of the papers to configured an irrelevance is "Manual handling". This class of issues represents the papers which present, usually, a framework to process extraction or analysis, but the development of the solution is completely manual. Those frameworks define steps to achieve the goals and perform them manually, without applying any tools or techniques.

The second and third categories that comprehend more papers are "Not covered BPM life-cycle phase" and "Not applying NLP". Some of the papers implement NLP techniques to other phases of the BPM life-cycle, specifically, two of them use NLP in the implementation phase—source code generation and service derivation—and one in the monitoring phase—to keep the process in compliance with context rules, such as regulatory rules.

The remaining irrelevant papers present issues such as insufficient level of details, summary of possible applications to NLP in BPM without development and supportive content to other papers without configure as an application by itself.

The distribution of relevant papers per database is 54.5% of papers from Scopus database, 33.34% from IEEE Explorer, 18.18% from ACM Digital Library and 18.18% as well from Springer Link.

Table 4.3: Papers selected in the systematic literature review.

Paper	*Author*	*Title*	*Year*
[1]	A.R. Gonçalves, J.C.; Santoro, F.M.; Baião, F.A.	A case study on designing business processes based on collaborative and mining approaches	2010
[2]	Mishra, A.; Sureka, A.	A graph processing based approach for automatic detection of semantic inconsistency between BPMN process model and SBVR rules	2015
[3]	Li, J.; Wang, H.J.; Zhang, Z.; Zhao, J.L.	A policy-based process mining framework: mining business policy texts for discovering process models	2010
[4]	Caporale, T.	A tool for natural language oriented business process modeling	2016
[5]	Li, J.; Wang, H.J.; Bai, X.	An intelligent approach to data extraction and task identification for process mining	2015
[6]	Koliadis, G.; Desai, N.V.; Narendra, N.C.; Ghose, A.K.	Analyst-Mediated Contextualization of Regulatory Policies	2010
[7]	Akbar, S.; Chaudhri, A.A.; Bajwa, I.S.	Automated analysis of logical connectives in business constraints	2013
[8]	Pittke, F.; Leopold, H.; Mendling, J.	Automatic Detection and Resolution of Lexical Ambiguity in Process Models	2015
[9]	Epure, E.V.; Martín-Rodilla, P.; Hug, C.; Deneckère, R.; Salinesi, C.	Automatic process model discovery from textual methodologies	2015
[10]	Elstermann, M.; Heuser, T.	Automatic Tool Support Possibilities for the Text-Based S-BPM Process Modelling Methodology	2016
[11]	Malik, S.; Bajwa, I.S.	Back to origin: Transformation of business process models to business rules	2013

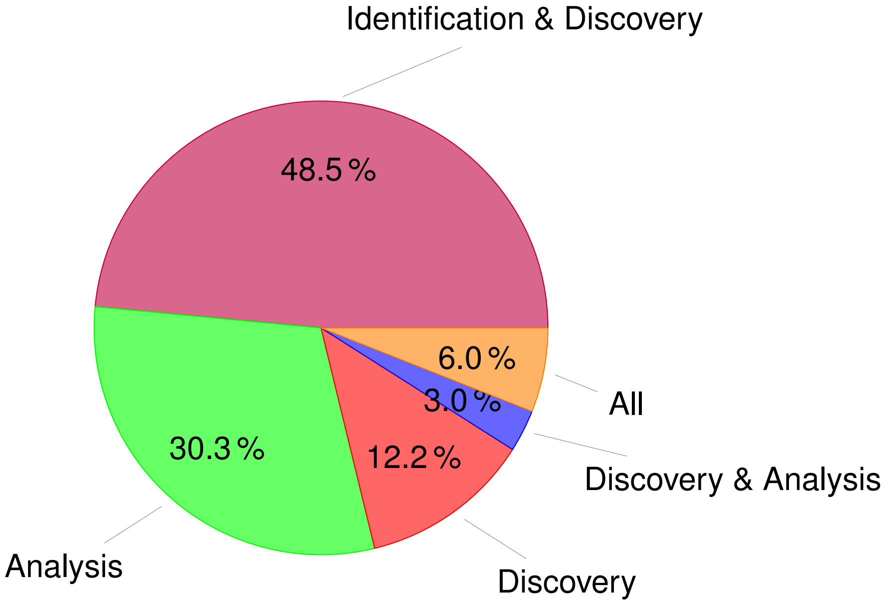
[12]	Gonçalves, J.C.d.A.R.; Santoro, F.M.; Baião, F.A.	Business process mining from group stories	2009
[13]	Gonçalves, J.C.d.A.R.; Santoro, F.M.; Baião, F.A.	Collaborative narratives for business rule elicitation	2011
[14]	van der Aalst, H.; Leopold, H.; Reijers, H.A.	Detecting inconsistencies between process models and textual descriptions	2015
[15]	Ghaisas, S.; Motwani, M.; Anish, P.R.	Detecting system use cases and validations from documents	2013
[16]	Leopold, H.; Eid-Sabbagh, R.-H.; Mendling, J.; Azevedo, L.G.; Baião, F.A.	Detection of naming convention violations in process models for different languages	2013
[17]	Liu, L.; Li, T.; Kou, X.	Eliciting Relations from Natural Language Requirements Documents Based on Linguistic and Statistical Analysis	2014
[18]	Sawant, K.P.;Roy, S.; Parachuri, D.; Plesse, F.; Bhattacharya, P.	Enforcing Structure on Textual Use Cases via Annotation Models	2014
[19]	Schumacher, P.; Minor, M.; Schulte-Zurhausen, E.	Extracting and enriching workflows from text	2013
[20]	Selway, M.; Grossmann, G.; Mayer, W.; Stumptner, M.	Formalising Natural Language Specifications Using a Cognitive Linguistics/Configuration Based Approach	2013
[21]	Njonko, P.B.F.; El Abed, W.	From natural language business requirements to executable models via SBVR	2012
[22]	Leopold, H.; Mendling, J.; Polyvyanyy, A.	Generating natural language texts from business process models	2012
[23]	Gonçalves, J.C.A.R.; Santoro, F.M.; Baião, F.A.	Let me tell you a story - on how to build process models	2011
[24]	Engel, R.; van der Aalst, W.M.P.; Zapletal, M.; Pichler, C.; Werthner, H.	Mining Inter-organizational Business Process Models from EDI Messages: A Case Study from the Automotive Sector	2012
[25]	Annervaz, K.M.; Kaulgud, V.; Sengupta, S.; Savagaonkar, M	Natural language requirements quality analysis based on business domain models	2013
[26]	Leopold, H.; Smirnov, S.; Mendling, J.	On labeling quality in business process models	2009
[27]	Leopold, H.; Smirnov, S.; Mendling, J.	On the refactoring of activity labels in business process models	2012

[28]	Friedrich, F.; Mendling, J.; Puhlmann, F.	Process model generation from natural language text	2011
[29]	Leopold, H.; Smirnov, S.; Mendling, J.	Refactoring of process model activity labels	2010
[30]	Motahari-Nezhad, H.R.; Cappi, J.M.; Nakamurra, T.; Qiao, M.	RFPCog: Linguistic-Based Identification and Mapping of Service Requirements in Request for Proposals (RFPs) to IT Service Solutions	2016
[31]	Bajwa, I.S.; Lee, M.G.; Bordbar, B.	SBVR business rules generation from natural language specification	2011
[32]	Di Francescomarino, C.; Tonella, P.	Supporting ontology-based semantic annotation of business processes with automated suggestions	2009
[33]	Leopold, H.; Mendling, J.; Polyvyanyy, A.	Supporting Process Model Validation through Natural Language Generation	2014

Source: the authors

After removing the non-relevant papers, the resulting set of papers is presented in Table 4.3 and contains 33 papers to be analyzed. They are divided by BPM life-cycle phase applicability in the Chart 4.1. The papers in Table 4.3 are referenced by their number in the column "Paper" along this study.

Figure 4.1: Life-cycle phases in papers.



Source: the authors

Chart 4.1 illustrates that the majority of the considered papers approach both identification and discovery phases, and none of the papers approaches only identification phase. The second most common approach is analysis phase. And then Discovery, Identification in union to Discovery and Analysis, and finally Discovery with Analysis, respectively.

According to Leopold (2013), the main applications of NLP in BPM are generation of models and analysis support and this evaluation supports the findings of the systematic literature review. Both applications have a high impact in the industry since the process designers' work is time consuming and expensive.

The actual generation of a conceptual model involves both process identification and discovery and the threshold between Identification and Discovery phases is very subtle and not easy to differentiate (DUMAS et al., 2013), likewise the two phases can be addressed by the same processing steps, since the information that relate to them is usually presented in the same context of the input text. Those are highly probable reasons why Identification & Discovery is the most commonly addressed application in the studies, and also the Identification phase by itself has not been addressed in any paper.

Different than the relationship between Identification and Discovery phase, the Analysis phase is not similar to the other two in means of input structure. While the first two phases use natural language text as inputs, the third phase uses the proper process model. For this reason, it is not easy to handle them all in the same processing chain.

The techniques applied in each of the phases are discussed in the next sections. The findings of the systematic literature review will be discussed firstly as an overview and then grouped according to the divisions of the Chart 4.2.

4.1 Overview of Findings

The relevant papers selected by the systematic review protocol were read and analyzed individually and classified as Identification, Discovery or Analysis phase, according to the BPM's life-cycle phase they address—considering Identification and Discovery as both part of the process extraction—, as well as the techniques and tools applied during the development of the study were identified. The results of these steps are shown in Table 4.5. This table consists of four columns and uses the classification defined in table 4.4. Its columns are:

- **General information** contains information about the paper (e.g. publication format), not related to its content;
- **Life-cycle applicability** contains a classification of phase of BPM life-cycle that the approach presented in the paper is focused on;
- **Approach** contains the techniques and methods studied in the paper to achieve the research goals;
- **Tools** contains the tools applied to the resolution of the problem.

Table 4.4: Classification used in the Overview table.

Dimension	*Acronym*	*Description*
General information	CNF	Conference
	JN	Journal
	WSP	Workshop
Life-cycle applicability	ID	Identification
	DS	Discovery
	AN	Analysis
Approach	TK	Tokenization
	POS	Part-of-speech
	PM	Pattern matching
	STW	Stemming of word
	SYT	Syntactic Tagging
	SHP	Shallow Parsing
	SET	Semantic Tagging
	LC	Lexicon Construction
	SK	Sequence Kernel
	ABD	Ambiguity Detection
	WC	Word Clustering
	TMP	Templates
	LMM	Lemmatization
	AR	Anaphora Resolution
	JAC	Jaccard Similarity
Tools	SP	Stanford Parser
	ST	Stanford Tagger

	WN	WordNet
	BN	BabelNet
	FM	FrameNet
	UT	Unigram Tagger
	BT	Bigram Tagger
	TT	Trigram Tagger
	BOW	Bag-of-words
	XM	XMeans
	ANP	ANTLR Parser
	RLS	RSLP Stemmer
	SDP	Stanford Dependency Parser
	GAT	General Architecture for Text Engineering
	GAZ	Gazetteer
	VN	VerbNet
	MPS	Multi-Pass Sieve
	SUN	SUNDANCE
	MIN	MINIPAR
	NLT	NLTK Tagger
	PWT	PunktWordTonekizer
	BRT	Brill Tagger

Source: the authors

Table 4.5: Overview of review findings.

Paper Ref	*General information*	*Life-cycle applicability*	*Approach*	*Tools*
[1]	CNF	ID+DS	TK+SHP+ STW+WC	RLS+TT
[2]	JN	AN	POS	SP
[3]	JN	ID+DS	PM+TK+POS+ SK+LC	BOW+TT
[4]	WSP	DS	PM	ANP
[5]	JN	DS+AN	LC+SK+POS	BOW
[6]	CNF	ID+DS+AN	TK+POS	TT

[7]	CNF	DS	TK+STW+POS+LMM	ST+SP
[8]	JN	ID+DS+AN	WC+LC+ABD	BN+XM
[9]	CNF	ID+DS	PM+TK+POS+LMM	ST+NLT+VN+WN
[10]	CNF	ID+DS	TK+POS	SP
[11]	WSP	AN	PM+POS	WN
[12]	CNF	ID+DS	TK+STW+POS+SHP+TMP	—
[13]	CNF	ID+DS	TK+POS+PM	PWT+BRT+TT
[14]	WSP	AN	TK+LMM+POS+AR	SP+SDP+WN
[15]	CNF	ID+DS	PM	—
[16]	JN	AN	POS+PM	—
[17]	CNF	ID+DS	POS	SP
[18]	CNF	ID+DS	POS+WC+LC	GAZ+ST+VN+MPS+GAT
[19]	CNF	ID+DS	TK+POS+PM+AR	SUN
[20]	CNF	ID+DS	TK+TMP+SET+PM	—
[21]	CNF	ID+DS	TK+POS+STW+LMM+PM	—
[22]	WSP	AN	TK+POS+AR	SP+WN
[23]	JN	ID+DS	TK+STW+POS+SHP+TMP	GST+PWT+TT+BT+UT
[24]	CNF	ID+DS	PM	—
[25]	CNF	AN	JAC	—
[26]	WSP	AN	POS	SP+ST+WN
[27]	JN	AN	PM+POS	WN+SP
[28]	WSP	ID+DS	TK+AR+PM	SP+FM+WN
[29]	WSP	AN	PM	WN
[30]	CNF	ID+DS	POS+PM	TT
[31]	CNF	ID+DS	POS+TK+SET+LC+PM	—
[32]	CNF	DS	ABD+PM	MIN+WN

[33]	JN	AN	PM	WN

Source: the authors

The findings presented in the Table 4.5 will be discussed according to the BPM life-cycle phases.

4.2 Research question 1: Natural Language Processing applied to Identification, Discovery and Analysis phases

Table 4.5 shows how the results of the systematic literature review are divided between the three relevant BPM life-cycle phases to this project. It is important to notice that none of the papers taken into account only approaches in the Identification phase.

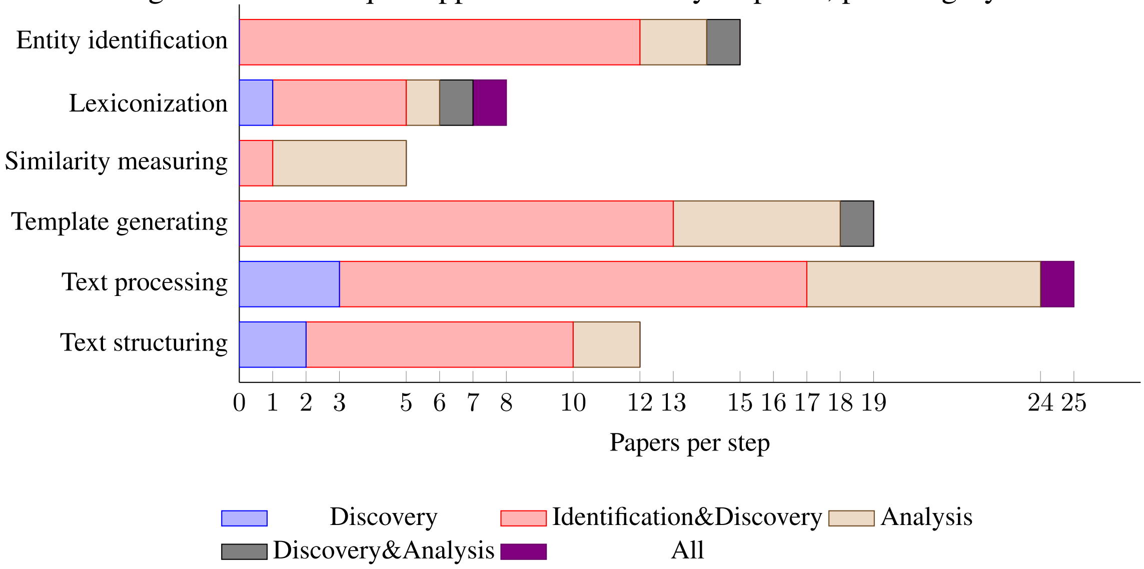
A reasonable explanation for this behaviour is that the simple recognition of the process, without further detailing, is not really relevant. Process modeling is the aim of the studies and it is composed by both Identification and Discovery phases (DUMAS et al., 2013).

Some of the papers approach only the Discovery phase, without considering Identification. This result is relative to the studies what aim only at performing process annotation or that had defined the process being specified beforehand (usually manually). Those studies had the process definition and metrics already provided and focus on aggregate details and descriptions.

The papers in Table 4.5 were analyzed and six categories of techniques were defined, according to the role they perform in the goal of the study:

- Entity identification: recognizes the parts of the business process model, such as actors, activities, events, and so on;
- Lexiconization: generates a lexicon of the context of the business process;
- Similarity measuring: calculates the similarity between entities and rules;
- Template generating: generates a template using rules and patterns to extract data or compose an output;
- Text processing: handles natural language text to extract the relevant data;
- Text structuring: formats the input previously to a desirable form to avoid grammatical issues.

Figure 4.2: Techniques applied in each life-cycle phase, per category



Source: the authors

Chart 4.2 compares the amount of techniques being applied in each step of all three life-cycle phases being covered separated by categories. The categories considered are described in the next sections.

4.2.1 Text structuring

During the analysis of the papers, two predominant patterns of solution were observed. Both patterns used a pipe-and-filter architecture—which incrementally feeds the next component, called filter, with the output of the previous one (ZHU, 2005). However they differ in the preparation phase, which defines the input to the method. The first pattern observed creates a preprocessed text to use as an input to the rest of the solution. The other considers the text as it is, in unstructured natural language.

One of the main challenges that natural language text imposes to automatic processing is ambiguity, which is inherent to unstructured texts (AKBAR; CHAUDHRI; BAJWA, 2013). As an attempt to deal with this issue, some authors proposed a way to partially structure the input text. From the group of 12 papers that applied any kind of previous text structuring, 3 of them did not limit the user's input, one used two different forms of input and the others used controlled inputs.

Within the group of unlimited inputs, two approaches found are completely performed as manual work from process analysts and designers. The third one applied the

TextCleaner solution, which consists of:

- Replacing “-” with “_”;
- Adding a space before and after each punctuation sign;
- Removing “ ’ ” from possessive singular nouns;
- Removing the comments without verbs within parenthesis;
- Removing negative sentences with are not present in the Corpora used (EPURE et al., 2015).

The most common approach to deal with the natural language ambiguities, although, was the use of a controlled input (8 of 12 papers). Most part of the controlled environment used templates, divided into 3 types: use case, Business Rules and patterned text. Except for one of the papers which applies this step to Analysis phase. The latter paper creates a natural language text from a BPM model, to allow stakeholders and domain experts to assure the generated model (MALIK; BAJWA, 2012). So, in summary, the solution gets a model, considered as a controlled input, to extract natural language text.

The remaining paper from this category does not address the efforts in text structuring to deal with natural language ambiguities, but via control flow definition. The solution takes two inputs: a natural language text and a UML model. The text is handled disregarding any previous processing, but the UML model is used to extract all flow information.

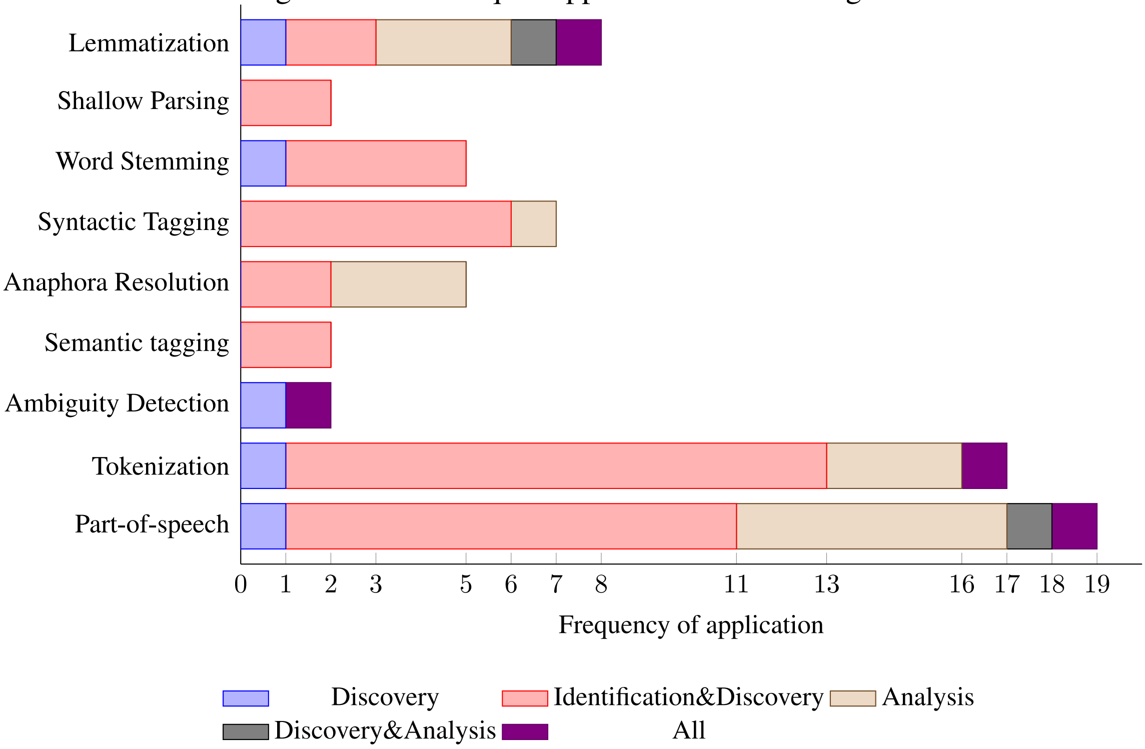
4.2.2 Text processing

The text processing step is the actual handling of natural language text. This phase involves morphological, syntactic and semantic analysis. The totality of the papers organize their solution in a pipeline architecture (called pipes and filters), which uses the output of a function as an input of the next one and applies filters (usually NLP techniques) to them.

The classification shoen in Chart 4.3 reflects the steps detailed in the studies. However, some studies do not mention all the steps, others mention only frameworks or tools in certain parts of the solution. For example, solutions which apply Stanford Parser probably use Tokenization and Part-of-speech tagging, however they are not mentioned.

Considering the absence of details or steps within the analyzed studies, nevertheless the basic approach to handle natural language texts may be defined as a combination of Tokenization and Part-of-speech tagging. Together they configure a Syntactic tagging. Another approach used to achieve approximately the same output is the application of a Shallow parser, which provides a part-of-speech tagging with a little extra information about the context and relation of the words (Semantic tagging) (RAMSHAW; MARCUS, 1995). Other steps are applied to solve some common issues, like ambiguities, passive voice, verb conjugation and so on.

Figure 4.3: Techniques applied in Text handling.



Source: the authors

The first function applied is tokenization. All the studies that applied it used the period mark to chop the input text into phrases, and some of them also used stop words, to separate the phrases into sentences. An issue that can emerge from this approach is to use period mark that symbolizes abbreviation (like in Mr. or Inc.) as a mark of end of phrase. This issue is presented in Friedrich, Mendling and Puhlmann (2011), however it does not propose any possible solution.

After the input had been split, the Part-of-speech tagging (POS) operation is performed and each word is tagged with its correspondent word class (such as noun, verb, adjectives and so on) (SHOPEN, 2007). Some studies used Shallow Parsers with the same goal.

In Chart 4.3, a category called Syntactic tagging is presented, however it does not represent the combination of Tokenization and Part-of-speech tagging. It represents other approaches to achieve a syntactically tagged text, such as Pattern matching. Those techniques are going to be discussed in the sub-section Template generation.

The Chart 4.3 also presents the categories Lemmatization, Word Stemming, Anaphora Resolution and Ambiguity Detection. Those categories only considered papers where a particular algorithm/technique was presented to implement the functionality. Some parsers and suites include options to apply those functionalities in the input text. However, their use is not mentioned to the solutions.

The presence of synonyms and homonyms[1] in the resulting process model is another issue, since they lead to duplicated entities and interfere with the compliance check, in the Analysis phase. To solve this issue, many papers used Lemmatization and Word stemming. Another important contribution of the approach is the possibility to relate phrases that were initially distinctly, but have keywords that derive from the same root and have the same meaning, as shown by [12].

Just like the presence of synonyms and homonyms, the use of anaphoras in the input text is a great issue. Anaphoras can lead to wrong relationship between phrases, since whether they are not recognized and identified as the entities they are referring to, entities that are not present in reality can be considered in the output. In [14], Anaphora Resolution is applied to support the similarity check between an activity in the process model and a sentence in the natural language text. This study uses a dependency tree, created by Stanford Parser, to identify objects in the sentences. If the sentence where the Anaphora occurred has no objects, the anaphoric is replaced by an object present in the previous sentence.

One of the papers ([22]) approached Anaphora Resolution in the opposite way. The paper proposes a solution to generate natural language text from business process models, to support human analysis. One of the steps of the solution introduces some anaphora by replacing recurrent terms by appropriate referring expressions. To find the right expressions, the study infers all the suitable hypernyms of the term using WordNet and obtain a more abstract word with the same semantics to replace the recurrent term.

The least addressed category mentioned in the Chart 4.3 was Ambiguity Detection. This category includes cases of ambiguity that are not only caused by the presence

[1] Jurafsky and Martin (2009) defined synonyms as different lexemes with the same meaning, while homonyms as same lexemes with unrelated meaning, where a lexeme is a pairing of a particular orthographic and phonological form with some form of symbolic meaning representation.

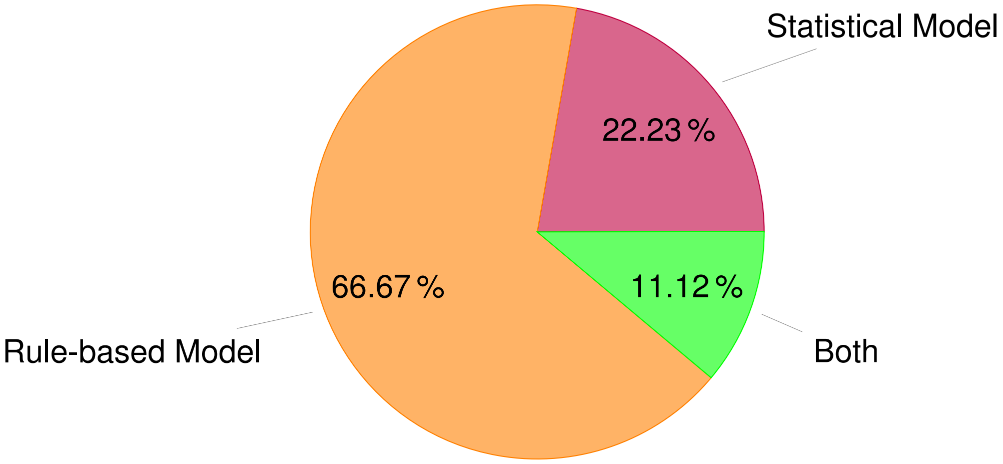
of synonyms and homonyms, that are already handle by lemmatization and word stemming. Many papers actually handle the issue, however they do not specify which is the chosen approach or even mention specifically the issue. Only two papers ([8][32]) consider explicitly a solution to the ambiguities. In [8] an enumerative approach is used, which consists of enumerating the senses that the word has in BabelNet and creates a vector of occurrences to later compare the vectors and find a sense that matches the context. Paper [32] relies on ontology analysis that is out of the scope of this work.

Those issues are usually handled during this step. However, not all the papers address all the mentioned issues, or, at least, do not detail the chosen approach. Most issues—and in some cases all of them—are not present in solutions which apply the text structuring step before text processing, since they already format the text to fit a suitable pattern.

4.2.3 Template generating

An alternative approach to extract useful process information from text is to use a template. Templates are a set of rules to extract the information according to the structure of the text fragments.

Figure 4.4: Template generation method.



Source: the authors

The most common approach used by the analyzed papers was the use of manually created rules. Pattern matching oriented to Semantics of Business Vocabulary and Rules (SBVR) was used by [2][20][31]. [2] aims at extracting natural language text from SBVR model in order to check compliance, so the template was applied considering the structure of SBVR as input. The other two works use SBVR as an intermediate representation, so they apply the rules of SBVR in a natural language text as input. Article [31] also applies rule-based models to syntactic analysis, by providing syntactic patterns, and role labeling to semantic analysis ([21] uses the same approach to perform semantic analysis).

Role labeling is the task of assigning semantic roles to fragments of the sentences. Semantic roles are representations that express the abstract role that an element of a sentence has within the context event (JURAFSKY; MARTIN, 2009). This technique is relevant to the Entity identification step.

The other 13 articles that used rule-based models defined manual sets of patterns based on analyzed group of sentences belonging to the relative domain.

An alternative to the methods that involve human work are the statistical models. Those are created by a machine learning technique which uses a training Corpora or a lexicon to training the agent to categorize the sentences and fragments automatically (MANNING; RAGHAVAN; SCHÜTZE, 2008). A training Corpora is a set of texts manually classified in relation to part-of-speech, usually pertinent to a specific domain. Lexicons are similar to them, however created by parts of the application domain. Lexicons are explained in the subsection Lexiconization.

There are many different statistical learning models, such as bag-of-words and sequence kernel (those are explained as part of the second research question). The works [1][3][5][23] applied statistical models using MAC-MORPHO Corpus, policy domain lexicon, company processes and industry standard lexicon, and company process lexicon, respectively.

The remaining articles ([12][13]) use both approaches. They apply rule-based models to perform sentence segmentation and statistical models to perform syntactic tagging.

4.2.4 Similarity measuring

Similarity measuring is a set of techniques used to check the compliance between the generated entities and the domain context. The methods are applied specially in the Analysis Phase.

There is one exception to the applicability phase within the set of papers considered. Paper [18] uses a Similarity Measuring technique to classify the actions of the model. The study considers a set of action types manually defined and uses Levenshtein distance to classify the model activities extracted by a Shallow Parser. The approach uses VerbNet to check the class of actions in the lexicon and using the calculated distance, places the actions in the closest action type (smallest distance).

The objective of the applicability of Similarity Measuring in [18] is the same as in [27], even though the BPM phase that it is applied to is different. In [27] a combination of semantic similarity and word frequency is used to categorize label styles in the Analysis phase. Semantic similarity is calculated by a pattern matching algorithm that processes WordNet semantics of the words and a frequency list of words extracted from a Corpus. The results of the WordNet application are matched to the list of frequencies of the current words to choose the part-of-speech that it is more likely to be used.

The other papers that applied Similarity Measuring, used it as a manner to verify compliance. Each of the studies selected a diverse approach to perform the checking. The paper [2] used sub-graph isomorphism VF2 Algorithm to check the compliance within the BPMN diagram and the SBVR rules. The approach generated all the possible sub-graphs of the BPMN diagram and the SBVR rules and compare them to check whether they are isomorphic.

Papers [14] and [25], on the other hand, used similarity calculations to accurately measure similarity. In [14], the measure proposed by Mihalcea, Corley and Strapparava (2006) is applied and considers both word semantic similarity and word specificity—a score that represents the inverted document frequency of the word—to avoid common and irrelevant terms. The approach in [25] differs from [14] since it does not consider word specificity and considers even common words, such as auxiliary verbs.

4.2.5 Lexiconization

Lexicons are a valuable support for training techniques that depend on machine learning, such as Statistical Taggers, Anaphora resolution, Word-sense disambiguation and so on.

The majority of the papers that applied lexiconization utilize manual created lexicons by analyzing a percentage of the applicability domain. Works [3][5][15][21] and [25] use this approach.

Three studies applied different alternatives:

- Heuristics were used in [20] to generate a lexicon;

- TF/IDF techniques were applied to the target domain in [23] to create a lexicon. TF/IDF stands for frequency-inverse document frequency and consists of evaluating the importance of a word in a domain according to the frequency that it is used in the domain and may use a weight to avoid common words, such as connectives, articles, and so on (MANNING; RAGHAVAN; SCHÜTZE, 2008);
- Ontology was used in [32]. Ontologies are a formalization of a set of concepts within a domain and the relationships between them (MAN, 2013). They are not in the scope of the present work.

Many studies do not generate domain lexicons, but use general ones—such as English language—to support tasks like ambiguity detection, lemmatization and stemming, and entity identification. However, the use of a domain sensible lexicon has the advantage of considering the word sense that is more suitable to the application, independently of whether that is the most common sense of the word.

4.2.6 Entity identification

Business Process Models are composed by entities, such as events, activities, decision points, actors and resources (DUMAS et al., 2013). From a natural language point of view, those entities are correlated to different grammatical patterns and roles.

Entity identification is an indispensable step in the elicitation of a model. Also, it represents an important step in some papers during Analysis phase. In [14] and [25], Entity identification is performed in order to check the compliance between the generated model and the input text. They differ in the approach: [14] uses Stanford Dependency Parser and bag-of-words to disambiguate conditional clauses and clauses that represent actions and extract the actions; [25] uses a Shallow Parser and a set of patterns to extract the entities.

The majority of the papers that approach Entity identification in Discovery phase use Role Labeling by Pattern Matching. The approach of [1][21][23][25][28] and [31] is similar, and it is composed of a dependency tree, generated with a NLP parser, and by the application of a set of grammatical rules created to the present study.

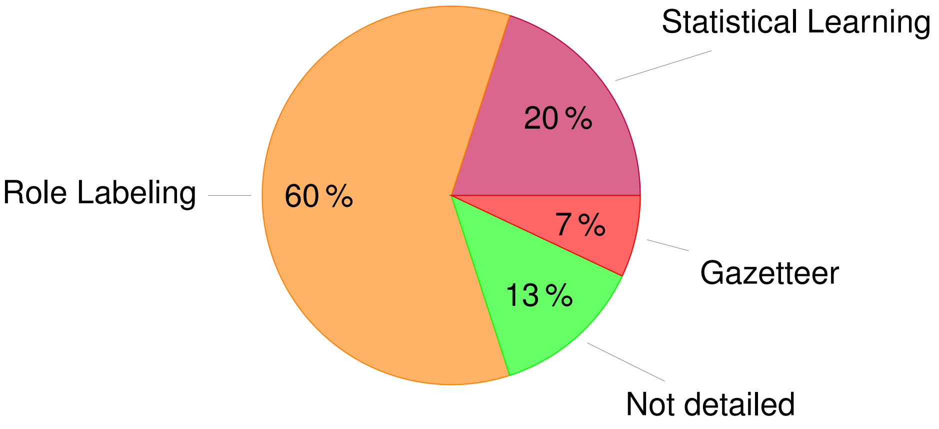
The grammatical rules represent the relationship between the entities and the natural language text elements, the papers considered that verb phrases shall be mapped to actions, and verb objects (direct or indirect) to their objects.

Another approach that also uses the same grammatical rules is Statistical Learning. The difference between this technique and Pattern Matching is that the model is trained using a Corpus created manually by annotating part of the domain text, and after that the model extracts the entities automatically, without the definition of explicit rules.

The last approach is only applicable for use cases, not unstructured text. [18] uses a Gazetteer to extract entities from the text description of use cases. A Gazetteer is a list of entity names that is created using the description of the use cases. The applicability of the Gazetteer is based on comparisons.

Chart 4.5 presents the distribution of techniques within the 15 papers that apply Entity Identification step.

Figure 4.5: Entity Identification approach.



Source: the authors

In addition to the previously discussed approaches, in Chart 4.5 a class called "Not detailed" is presented. This class comprehends the papers that mention the existence of an Entity Identification operation, however do not explain it.

4.3 Research Question 2: Natural Language Processing tools used in Process Discovery

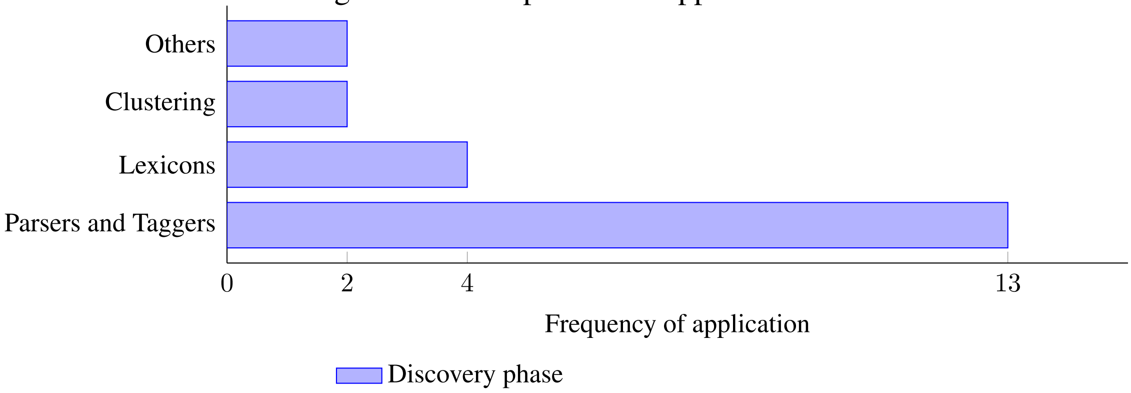
In order to answer the second Research Question, it is useful to consider the last column of Table 4.5. Since the question only considers the Discovery phase, the papers [2][11][14][16][22][25][26][27][29] and [33] are not taken into consideration in this analysis. Also, the papers that approach more than one BPM life-cycle phase are only partially considered.

The analysis of the tools is divided in the following groups:

- Parsers and Taggers;
- Lexicons;
- Clustering;
- Others.

Figure 4.6 shows the most applied tools within the groups. In the next sub-sections each of the groups is going to be discussed.

Figure 4.6: Groups of tools application.



Source: the authors

4.3.1 Parsers and Taggers

Parsers and Taggers comprehend the tools used to implement the majority of the techniques involved in the text processing. They are applied to Sentence Segmentation, Tokenization, Part-of-speech tagging, Shallow Parsing, Dependency parsing, Word Stemming, Lemmatization and Role Labeling.

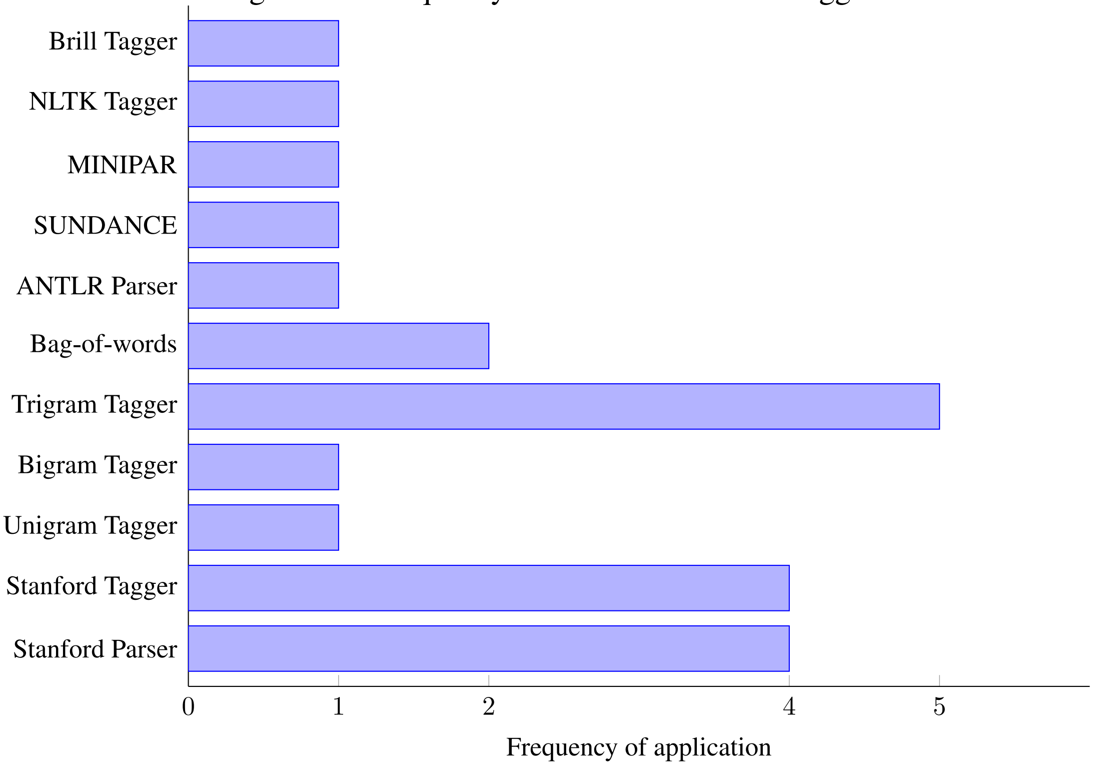
Chart 4.7 presents the frequency of use of each of the Parsers and Taggers used in the solutions. Many studies use combinations of more than one parser/tagger or one of each. In [9], a comparison of three different solution to syntactic and semantic annotation was provided to check for accuracy. The solutions were:

1. Stanford Parser with raw text;
2. Stanford Parser with the text previously POS tagged by Stanford Tagger;
3. Stanford Parser with the text previously POS tagged by NLTK Tagger.

A sample of the annotated text handled by each of these solutions was analyzed manually and the second solution showed higher accuracy. This second solution was also used in

[7].

Figure 4.7: Frequency of use of Parsers and Taggers.



Source: the authors

Another combination presented was composed by three taggers, as in [23]. The solution applied Trigram Tagger, Bigram Tagger and Unigram Tagger, respectively. The n-gram Taggers are based on Markov models and they are able to POS tag a word considering the n-1 previous words (0 for unigram, 1 for bigram and 2 for trigram) (JURAFSKY; MARTIN, 2009). The aim of the combination is to try to tag the word using the Trigram Tagger—which potentially provides more accurate tagging—, if the model is not able to find a suitable part-of-speech, a Bigram is used to re-try, and the same is performed using Unigram Tagger.

There are two types of taggers, regarding the approach they use:

- Constrain-based tagger: they are hand-coded and composed by a morphological analysis, a large lexicon and a set of morphological descriptions of ambiguity cases;
- Statistical tagger: they are automatically generated using an annotated Corpus. They can be represented as collocational matrices, Hidden Markov models, local rules and neural networks (SAMUELSSON; VOUTILAINEN, 1997).

Both types aim at tagging words with part-of-speech information. Samuelsson and Voutilainen (1997) claim that Constrain-based taggers (99% of precision) are more precise than Statistical ones (95-97% of precision), however the Statistical Taggers are easier to use.

Among the Taggers considered, only Brill Tagger uses the constraint-based approach. The others are Statistical Taggers— it is important to note that bag-of-words is a Unigram Tagger. In the solution proposed by [13], Brill Tagger is applied, but the study does not provide accuracy information to corroborate the analysis presented in Samuelsson and Voutilainen (1997).

A comparison between bag-of-words and tree kernel, both of which are Statistical Taggers and are based on kernels, is provided by [3]. The difference between bag-of-words and Tree Kernel is that the first one treats the dependency tree as a vector of features without structural information, while the other considers structure (CULOTTA; SORENSEN, 2004). The results of both approaches applied to extracting business policies from documents are shown in Table 4.6.

Table 4.6: Comparison between Tree Kernel and Bag-of-words to extract policies.

Policy	*Methods*	*Accuracy (%)*	*Precision (%)*	*Recall (%)*	*F-measure (%)*
Purchasing	Bag-of-words	77.33	62.75	40.00	46.48
	Tree kernel	78.71	68.67	36.25	45.34
Travel	Bag-of-words	87.69	50.00	45.00	45.33
	Tree kernel	89.23	50.00	30.00	36.67

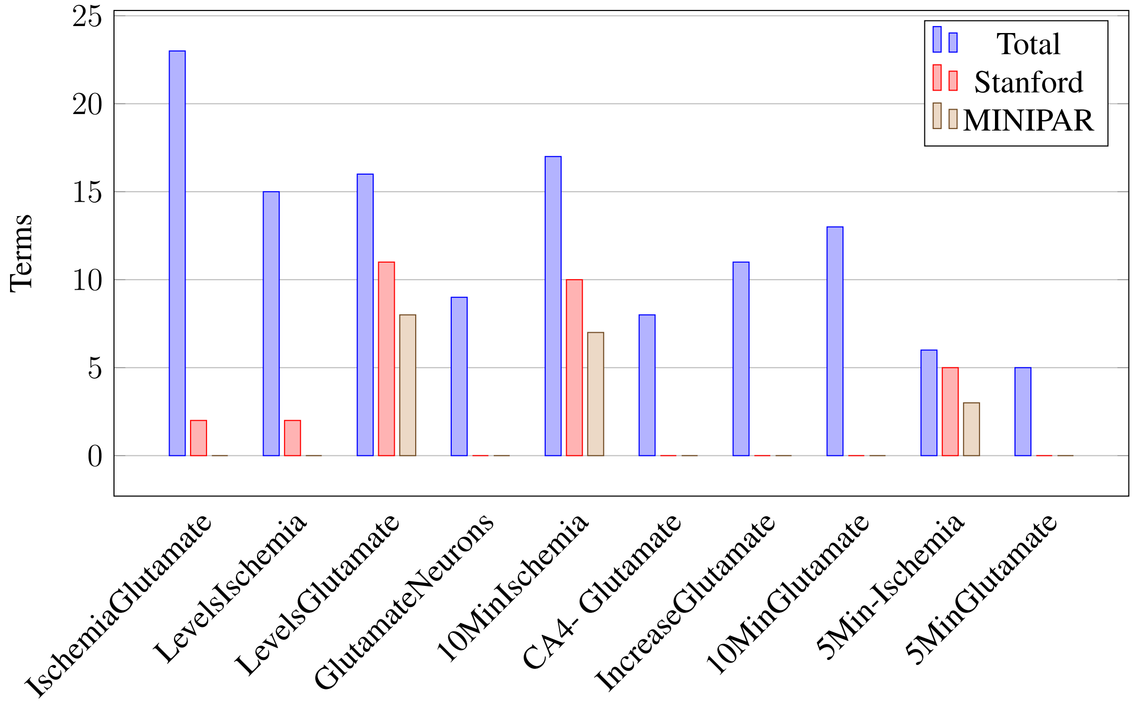
Source: [3]

In both business policies, the Tree kernel presented higher accuracy and in Purchasing, also higher precision. The bag-of-words method has higher recall and F-measure in both policies. This behaviour can be explained by the tendency of tree kernel methods to predict more sentences as negative, even though both methods are conservative. The comparison shows that the amount of information handled by tree kernel (structural information) does not create a more powerful tool than the lexical patterns of bag-of-words.

Figure 4.7 presents MINIPAR and SUNDANCE, along with ANTLR Parser and Stanford Parser. All those are NLP parsers composed by many NLP techniques to perform Morphological, Syntactic and Semantic analysis. They are all statistical parsers. The literature, as far as the author acknowledges, does not provide recall and precision comparisons between those Parsers. A comparison between the application of each of them to the same set of data would be valuable.

In (SHAMS, 2014), a comparison between the Stanford Parser and MINIPAR is presented in the scope of biomedical documents. Figure 4.8 shows the comparison of the number of biomedical terms each of the parsers identified in relation to a concept. The total amount of terms related to each concept is also provide in the chart.

Figure 4.8: Comparison of term recognition between Stanford Parser and MINIPAR.



Source: adapted from Shams (2014)

The differences of performance shown in Figure 4.8 may be caused by the Corpus used to train the model for POS tagging, or by the different approaches for ambiguity detection, anaphora resolution and so on. Since not much information is provided about MINIPAR, it is not possible to assure the cause of the discrepancy, however the distinctness of the Corpus is the most likely for specific domains.

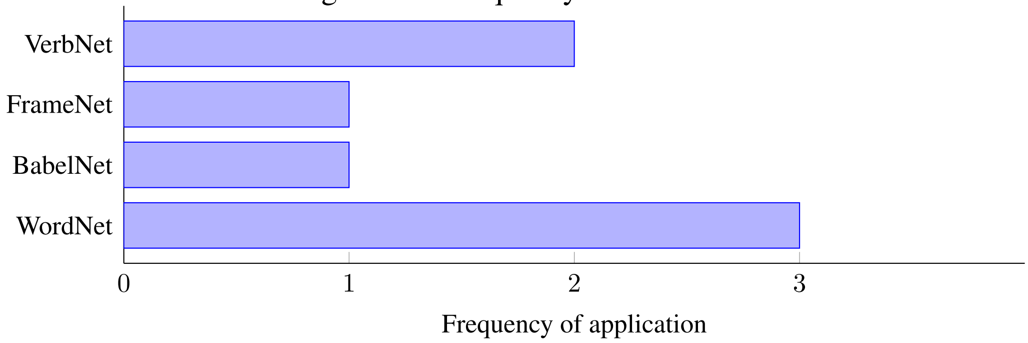
Parallels considering other parsers or within the domain of BPM were not found in the literature.

4.3.2 Lexicons

This section presents general purpose lexicons. These lexicons unlike domain lexicons are composed of words, classifications and definitions (lexemes) of a specific language, but not taking application domain into account.

Since none of the papers used any tool to generate a lexicon by domain documents, only algorithms (that are techniques, not tools), any other class of lexicon is approached in this section. Chart 4.9 shows the lexicons used in the analyzed papers.

Figure 4.9: Frequency of use of Lexicons.



Source: the authors

The presented lexicons differ in the way the Corpus used to generate them, when considering English language application. Variability in the Corpus and the processing of Corpus create lexicons more or less suitable for particular domains and grammatical forms. For example, VerbNet was generated to provide syntactic and semantical information about verbs, considering the Levin's verb classification—verbs in the same syntactic frame share the same syntactic behaviour. On the other hand, FrameNet is based on frames that represent syntactic features and are correlated to semantic roles (SHI; MIHALCEA, 2005).

WordNet is a resource to identify the semantic features of a word. It contains 155,287 words grouped in 117,659 groups of synonyms, called synsets. WordNet contains 206,941 word-senses related to the words (UNIVERSITY, 2016).

BabelNet is focused on translation applications and combines word definitions and senses from WordNet, Wikipedia, OmegaWiki, Wiktionary, VerbNet, and so on. BabelNet contains 1,769,205, 6,667,855 synsets and 17,265,977 word-senses for English language (NAVIGLI, 2016).

The lexicons can be combined to improve the results of particular operations. In [9], WordNet and VerbNet were used together to automatically classify verbs as transitive or intransitive and create a tuple composed by verb and a list of its objects. In the same solution, WordNet is also used to perform lemmatization. Another combination of lexicons is used in [28], in order to analyze synonyms, homonyms and hypernyms and provide semantic information the study combined WordNet and FrameNet.

Laparra and Rigau (2009) provide an analysis of performance of the combination of WordNet and FrameNet for word-sense disambiguation operation. The solution showed a recall, precision and F-measure of 0.69, which are the same results found using SSI-Dijkstra, a Structural Semantics Interconnections algorithm commonly used to handle word-sense disambiguation (LAPARRA; RIGAU, 2009).

4.3.3 Clustering

Clustering tools intend to create groups of similar elements. Only two Clustering tools were used in the solutions analyzed: XMeans and Multi-Pass Sieve.

XMeans was applied in the homonym detection step of [8]. It estimates the number of clusters of word-senses in different models to find the most similar ones and then decide for an alternative to resolve the ambiguity.

On the other hand, Multi-Pass Sieve was applied in [18] to solve co-references. It is executed iteratively to create clusters of business entities and to match the similar ones. It only considers the first mention of an entity in the text, since it contains more accurate information. It is repeated five times to merge all the possible clusters and then find the appropriate entity name to replace the reference. An evaluation of the homonym resolution in [9] including XMeans, but also BabelNet word-sense relations and a word-sense disambiguation algorithm, is presented in Table 4.7. The databases used are a set of process models by SAP, other by TelCo and another by AI.

Table 4.7: Results of the use of Homonym Resolution.

		SAP	*TelCo*	*IA*
SpWA	Before	6.79	8.67	7.29
	After	1.71	1.47	1.86
SpWPA	Before	5.37	6.27	5.70
	After	1.83	1.47	1.54
SpWPBO	Before	7.31	8.96	7.60
	After	1.66	1.76	7.92

Source: [8]

In Table 4.7 the comparison of the average quantity of word-senses per word before and after the use of the solution is presented. Three categories of homonyms are considered: a set of process models (SpWP), a set of actions of process models (SpWPA) and a set of business objects of process models (SpWPBO).

4.3.4 Other tools

This section presents auxiliary tools used in some of the analyzed papers. These tools do not perform an entire operation by themselves, but support others.

Word Stemming is an operation well performed by NLP Parsers, however [1] considers Portuguese words and these are not included in the most popular parsers. For this reason, the study used RSLP Stemmer, a Word Stemming tool for Portuguese words. The performance of RSLP Stemmer was evaluated in (FLORES; MOREIRA; HEUSER, 2010), considering a set of morphological and semantically related 2854 words. Table 4.8 shows a comparison of RSLP Stemmer, a truncation algorithm and the set of non stemmered words.

The Table 4.8 considers four measurements:

- UI: calculates the number of times a suffix is not removed;
- OI: calculates the number of times a part of the stem is mistakenly removed considering it was part of the suffix;
- SW: is the ratio of OI/UI;
- ERRT: determines a truncation line. An adequate stemmer should be on the lower side of this line.

Table 4.8: Results of the use of RSLP Stemmer.

	UI	*OI*	*SW*	*ERRT*
RSLP Stemmer	0.1905226632	0.0002680360	0.0014068458	0.5691374097
Trunc3	0.0304706137	0.0157951554	0.5183733933	1.0000000000
NoStem	1.0000000000	0.0000000000	0.0000000000	1.0000000000

Source: adapted from Flores, Moreira and Heuser (2010)

Considering the reasonably low UI and OI values and the ERRT value, RSLP Stemmer for Portuguese words may be considered as an efficient Word Stemmer.

Similarly to [1], [23] also considers Portuguese words and requires applying more suitable tools for this language. For this reason, the solution does not use any of the mentioned NLP Parser to perform tokenization, but the WordPunctTokenizer, which is part of NLTk library.

The remaining two general tools used are General Architecture for Text Engineering (GATE) and Gazetteer. The Gazetter was already discussed in the Entity Identification section, since it is a very simple tool that implements a specific technique. The GATE is an architectural tool that supports natural language processing offering many operations to handle text. It is not a parser, but an environment to create NLP tools (SHEFFIELD, 2016).

5 CONCLUSION

In this work a systematic literature review regarding the state-of-the-art in NLP techniques applied in the first three BPM phases—Identification, Discovery and Analysis—was provided. It also provided an analysis of NLP tools used for Process Discovery.

Through the papers' analysis, it was concluded that NLP is mainly applied to Identification and Discovery phases in the stage of text processing focused in process model extraction—by performing morphological, syntactic and semantic analysis. In the Analysis phase, it is applied to examine entity naming conventions and to extract natural language text from models to allow a better compliance analysis by process' participants.

The analysis also pointed out that, during the Discovery phase, many tools are used as solution to support process model extraction from natural language text, however the majority of the solutions applied parsers and taggers. In this group, the most used ones were Trigram Tagger and Stanford Tagger and Parser. The use of lexicons was quite frequent to cope with word-sense disambiguation.

In addition to word-sense disambiguation, many other challenges in terms of natural language processing during the extraction of models from natural language text were presented, such as presence of anaphoric references, difficulties in performing part-of-speech tagging, co-reference between sentences and sentence segmentation. Techniques and tools were presented to attempt to solve those issues.

The main contributions of this work are:

- To present useful tools to support the activities of Discovery phase, such as process architecture design, model annotation, model sequence flow definition and so on;
- To point to a deficiency in the literature in terms of comparative analysis of those techniques and tools within the same applications.

The results of this work are useful to support the design of frameworks to extract process models from natural language texts, as well as to compare alternative approaches. The systematic literature review revealed both an important lack of research in comparative analysis of existent approaches and tools, and the shallow exploration of different alternatives to cope with the proposed problem. Both research fields would benefit from the development of more studies.

In the future, this study could be extended by analyzing the remaining three phases of BPM life-cycle: Redesign, Implementation and Monitoring and Controlling.

REFERENCES

AALST, W. M. P. van der. **Process Mining: Discovery, Conformance and Enhancement of Business Processes**. 1st. ed. [S.l.]: Springer Publishing Company, Incorporated, 2011.

AHMAD, M.; RIYAZ, R. Rule based semantic parsing approach for kashmiri language. **International Journal of Advanced Research in Computer Science and Software Engineering**, 2013.

AKBAR, S.; CHAUDHRI, A. A.; BAJWA, I. S. Automated analysis of logical connectives in business constraints. In: **2013 International Conference on Current Trends in Information Technology (CTIT)**. [S.l.: s.n.], 2013. p. 209–213.

BLUMBERG, R.; ATRE, S. **The Problem with Unstructured Data**. 2003. Available from Internet: <http://www.information-management.com/issues/20030201/6287-1.html>.

BOLLOJU, N.; SCHNEIDER, C.; SUGUMARAN, V. A knowledge-based system for improving the consistency between object models and use case narratives. **Expert Systems with Applications**, v. 39, 2012.

COALITION, W. M. **What is BPM?** 2015. Available from Internet: <http://www.wfmc.org/what-is-bpm>.

COMPUTER, D. of; PENNSYLVANIA, I. S. at the University of. **Natural Language Toolkit**. 2015. Available from Internet: <http://www.nltk.org/index.html>.

CULOTTA, A.; SORENSEN, J. Dependency tree kernels for relation extraction. In: **Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics**. Stroudsburg, PA, USA: Association for Computational Linguistics, 2004. (ACL ’04).

DEEMTER, K. V.; KRAHMER, E.; THEUNE, M. Real versus template-based natural language generation: A false opposition? **Comput. Linguist.**, MIT Press, Cambridge, MA, USA, v. 31, n. 1, p. 15–24, mar. 2005.

DUMAS, M. et al. **Fundamentals of Business Process Management**. [S.l.]: Springer Publishing Company, Incorporated, 2013.

EDINBURGH, T. U. O. **Systematic reviews and meta-analyses: a step-by-step guide**. 2013. Available from Internet: <http://www.ccace.ed.ac.uk/research/software-resources/systematic-reviews-and-meta-analyses>.

EPURE, E. et al. Automatic process model discovery from textual methodologies: An archaeology case study. In: **2015 IEEE 9th International Conference on Research Challenges in Information Science (RCIS)**. [S.l.: s.n.], 2015. p. 19–30.

FARRELL, B. **BPM can have a holistic impact on businesses**. 2013. Available from Internet: <http://www.appian.com/blog/bpm/bpm-can-have-a-holistic-impact-on-businesses>.

FLORES, F. N.; MOREIRA, V. P.; HEUSER, C. A. Assessing the impact of stemming accuracy on information retrieval. In: **Proceedings of the 9th International Conference on Computational Processing of the Portuguese Language**. Berlin, Heidelberg: Springer-Verlag, 2010. (PROPOR'10), p. 11–20. ISBN 3-642-12319-8, 978-3-642-12319-1.

FRIED, M.; ÖSTMAN, J. **Construction Grammar in a Cross-language Perspective**. [S.l.]: John Benjamins Pub., 2004. (Constructional approaches to language).

FRIEDRICH, F.; MENDLING, J.; PUHLMANN, F. Process model generation from natural language text. In: **Proceedings of the 23rd International Conference on Advanced Information Systems Engineering**. Berlin, Heidelberg: Springer-Verlag, 2011. (CAiSE'11), p. 482–496.

GROUP, T. S. N. L. P. **Stanford Log-linear Part-Of-Speech Tagger**. 2015. Available from Internet: <http://nlp.stanford.edu/software/tagger.shtml>.

GROUP, T. S. N. L. P. **The Stanford Parser: A statistical parser**. 2015. Available from Internet: <http://nlp.stanford.edu/software/lex-parser.shtml>.

HAYES, P. J.; CARBONELL, J. G. **A tutorial on techniques and applications for natural language processing**. [S.l.]: Carnegie Mellon University, 1983.

INDURKHYA, N.; DAMERAU, F. J. **Handbook of Natural Language Processing**. 2nd. ed. [S.l.]: Chapman & Hall/CRC, 2010.

JURAFSKY, D.; MARTIN, J. H. **Speech and Language Processing**. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2009.

KITCHENHAM, B. Guidelines for performing systematic literature reviews in software software engineering. 2007.

KITCHENHAM, B. et al. Systematic literature reviews in software engineering – a systematic literature review. **Information and Software Technology**, v. 51, n. 1, p. 7—15, 2009.

LAHTINEN, S.; PELTONEN, J. Adding speech recognition support to uml tools. **Journal of Visual Languages and Computing**, v. 16, p. 85—118, 2005.

LAPARRA, E.; RIGAU, G. Integrating wordnet and framenet using a knowledge-based word sense disambiguation algorithm. **Proceedings of Recent Advances in Natural Language Processing (RANLP09)**, p. 1—6, 2009.

LEOPOLD, H. **Natural Language in Business Process Models**: Theoretical foundations, techniques, and applications. [S.l.]: Springer International Publishing, 2013. (Lecture Notes in Business Information Processing, v. 168).

LYONS, J. **Language**, Linguistic Society of America, v. 69, n. 4, p. 825–828, 1993.

MALIK, S.; BAJWA, I. S. Back to origin: Transformation of business process models to business rules. In: **Business Process Management Workshops**. [S.l.: s.n.], 2012.

MAN, D. Ontologies in computer science. **Didactica Mathematica**, v. 31, n. 1, p. 43–46, 2013.

MANNING, C. D.; RAGHAVAN, P.; SCHÜTZE, H. **Introduction to Information Retrieval**. New York, NY, USA: Cambridge University Press, 2008.

MIHALCEA, R.; CORLEY, C.; STRAPPARAVA, C. Corpus-based and knowledge-based measures of text semantic similarity. **AAAI**, v. 6, p. 775–780, 2006.

MITKOV, R. **Anaphora resolution: the state of the art**. [S.l.]: University of Wolverhampton, 1999.

NAVIGLI, R. Word sense disambiguation: A survey. ACM Comput, v. 41, n. 2, p. 1–69, 2009.

NAVIGLI, R. **BabelNet**. 2016. Available from Internet: <http://babelnet.org/>.

ORENGO, V. M.; HUYCK, C. A stemming algorithm for the portuguese language. **Proceedings of the SPIRE Conference**, p. 186–193, 2001.

PALMER, N. **What is BPM?** 2014. Available from Internet: <http://bpm.com/what-is-bpm>.

PALMER, N. **What is BPM?** 2015. Available from Internet: <http://bpm.com/what-is-bpm>.

PELLEG, D.; MOORE, A. X-means: Extending k-means with efficient estimation of the number of clusters. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, p. 727–734, 2000.

RAMSHAW, L. A.; MARCUS, M. P. Text chunking using transformation-based learning. **Proceedings of the Third ACL Workshop on Very Large Corpora**, p. 82–94, 1995.

RECKER, J.; MENDLING, J. The state of the art of business process management research as published in the bpm conference. **Business & Information Systems Engineering**, v. 58, n. 1, p. 55—72, 2016.

RICHARDS, D.; FURE, A.; AGUILERA, O. An approach to visualise and reconcile use case descriptions from multiple viewpoints. In: **Proceedings of the 11th IEEE International Conference on Requirements Engineering**. [S.l.: s.n.], 2003. (RE 2003), p. 373—374.

RILOFF, E.; PHILLIPS, W. An introduction to the sundance and autoslog systems. School of Computing, University of Utah, 2004.

ROSING, M. v.; SCHEEL, H. v.; SCHEER, A.-W. **The Complete Business Process Handbook: Body of Knowledge from Process Modeling to BPM, Volume I**. 1st. ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2014.

ROTH, D. Learning to resolve natural language ambiguities: A unified approach. **AAAI-98 Proceedings**, 1998.

RUDDEN, J. Making the case for bpm: A benefits checklist. **BPTrends**, 2007.

SAMUELSSON, C.; VOUTILAINEN, A. Comparing a linguistic and a stochastic tagger. In: **Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics**. Madrid, Spain: Association for Computational Linguistics, 1997. p. 246–253.

SELWAY, M.; MAYER, W.; STUMPTNER, M. Configuring domain knowledge for natural language understanding. In: **Configuration Workshop**. [S.l.]: CEUR-WS.org, 2013. (CEUR Workshop Proceedings, v. 1128), p. 63–70.

SHAMS, R. Performance of stanford and minipar parser on biomedical texts. **CoRR**, abs/1409.7386, 2014.

SHEFFIELD, U. of. **GATE: a full-lifecycle open source solution for text processing**. 2016. Available from Internet: <https://gate.ac.uk/>.

SHI, L.; MIHALCEA, R. Putting the pieces together: Combining FrameNet, VerbNet, and WordNet for robust semantic parsing. In: **Proceedings of the Sixth International Conference on Intelligent Text Processing and Computational Linguistics**. Mexico: [s.n.], 2005.

SHOPEN, T. **Language Typology and Syntactic Description: Volume 1, Clause Structure**. [S.l.]: Cambridge University Press, 2007. (Language Typology and Syntactic Description).

TURING, A. M. Computing machinery and intelligence. **Mind**, Oxford University Press, Mind Association, v. 59, n. 236, p. 433–460, 1950.

UNIVERSITY, P. **WordNet**: A lexical database for english. 2016. Available from Internet: <https://wordnet.princeton.edu/>.

VAN DER AALST, W. M. P. A decade of business process management conferences: Personal reflections on a developing discipline. In: **Proceedings of the 10th International Conference on Business Process Management**. Berlin, Heidelberg: Springer-Verlag, 2012. (BPM'12), p. 1—16.

VAN DER AALST, W. M. P.; LA ROSA, M.; SANTORO, F. M. Business process management: Donť forget to improve the process! **Business & Information Systems Engineering**, v. 58, n. 1, p. 1–6, 2016.

WESKE, M. **Business Process Management: Concepts, Languages, Architectures**. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2007.

YALLA, P.; SHARMA, N. Integrating natural language processing and software engineering. **International Journal Of Software Engineering and its Applications**, v. 9, n. 11, p. 127—136, 2015.

ZHU, H. : From principles to architectural styles. Oxford, UK: Butterworth-Heinemann, 2005.