



Empirical Analysis of Sentence Templates and Ambiguity Issues for Business Process Descriptions

Thanner Soares Silva¹(✉) , Lucinéia Heloisa Thom¹ , Aline Weber¹ ,
José Palazzo Moreira de Oliveira¹ , and Marcelo Fantinato² 

¹ Department of Informatics, Federal University of Rio Grande do Sul, UFRGS,
Porto Alegre, Brazil

{thanner.silva,lucineia,aline.weber,palazzo}@inf.ufrgs.br

² School of Arts, Sciences and Humanities, University of São Paulo,
São Paulo, Brazil
m.fantinato@usp.br

Abstract. Business process management has become an increasingly present activity in organizations. In this context, approaches that assist in the identification and documentation of business processes are presented as relevant efforts to make organizations more competitive. To achieve these goals, business process descriptions are considered as a useful artifact in both identifying business processes and complementing business process documentation. However, approaches that automatically generate business process descriptions do not explain how the sentence templates that compose the text were selected. This selection influences the quality of the text, as it may produce ambiguous or non-recurring sentences, which could make it difficult to understand the process. In this work, we present an empirical analysis of 64 business process descriptions in order to find recurrent sentence templates and filter them for ambiguity issues. The analysis made it possible to find 101 sentence templates divided into 29 categories. In addition, 13 of the sentence templates were considered to have ambiguity issues based on the adopted criteria. These findings may support other approaches in generating process descriptions more suitable for process analysts and domain experts.

Keywords: Business Process Model and Notation
Natural language processing · Sentence template · Ambiguity

1 Introduction

In order to stay competitive, organizations need to document and manage their business processes. Approaches related to process descriptions and Business Process Model and Notation (BPMN) can help them to achieve these goals. The combination of distinct approaches can improve understanding of the process as

they provide different perspectives on it [21, 23]. Several proposals have been presented with the goal of contributing to the relationship between texts and process models in different scenarios, such as: automatic generation of process descriptions from process models [17, 18, 20], automatic generation of process models from process descriptions [8, 14, 15, 25], process mining from natural language text [5, 6, 10, 19], identification of business process elements in natural language texts [12], integration between texts and process models [16] and verification of conformity between texts and process models [1, 3, 24].

A number of the proposed approaches related to this context use sentence templates to generate or transform process descriptions. However, the corresponding related works do not explain how the sentence templates that compose the process description were selected and this information is important, as it directly interferes with the quality of the text. Sentence templates not carefully selected may produce sentences with ambiguity problems that may not be understood by the stakeholders of the business processes, such as process analysts and domain experts.

Furthermore, this work aims to help in the development of an approach that generates process-oriented texts from natural language texts. In the context of this approach, a process-oriented text is defined as a text that is both structured and capable of maintaining the maximum information related to the business process. In addition, it is expected to verify that the business process described in the text conforms to the BPMN specification. The approach consists of five stages: input data, text reading, BPMN verification, text writing and text output. Firstly, a natural language text is given as input to the input data (i.e., input data stage). Then the approach reads the natural language text and produces an intermediate structure (i.e., text reader stage). Then, the intermediate structure is used to verify the process described by the text (i.e., BPMN verification stage) and to generate the text writer (i.e., text writer stage). Finally, the verification and the structured text are combined for the generation of the process-oriented text (i.e., output text stage). Figure 1 shows the approach with its respective stages. This paper seeks to contribute with the sentence templates repository presented in stage “4. Text writer”.

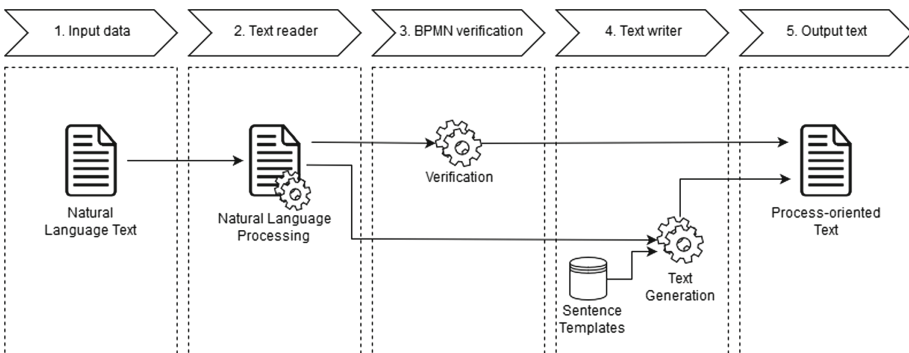


Fig. 1. Process-oriented text generation.

In this context, this work proposes an analysis to identify sentence templates that are common in processes descriptions and that causes less ambiguity problems, so the findings can be useful in writing new process descriptions. For this, an empirical analysis of 64 texts describing business processes was performed. These texts were taken from the book *Fundamentals of Business Process Management* [9] and Friedrich [13]. Moreover, an investigation of ambiguity issues in process descriptions was carried out based on the literature and on the sentence templates identified. This paper presents the results of these analyses.

A total of 101 sentence templates was found, divided into 29 categories based on three criteria, namely: source, target and relationship. Six types of ambiguities were identified and, when compared with the found sentence templates, enabled us to define 13 templates related to ambiguity issues.

The remainder of this paper is organized as follows: Sect. 2 presents BPMN as background of the presented research. Section 3 presents the method used to identify and classify sentence templates as well as the identified ambiguity issues. Section 4 reports the results analysis. Section 5 contemplates the applications of this work, as well as discusses the results obtained. Section 6 presents the related work. Finally, Sect. 7 presents the final conclusions.

2 Background

The BPMN is a standard for process modeling maintained by the Object Management Group (OMG) [22]. BPMN includes five elements categories: flow objects (activities, events and gateways), data (data objects, data inputs, data outputs and data stores), connecting objects (sequence flows, message flows, associations and data associations), swimlanes (pools and lanes) and artifacts (groups and text annotations). In this work, the focus is on flow objects, swimlanes and connecting objects because they are used for the identification and classification of sentence templates.

Regarding flow objects, activities can be defined as a task that a company performs in a process. An activity can be atomic or non-atomic and is represented as rounded boxes. Events are represented as circles and indicate where a particular process starts (start event) or ends (end event). Moreover, there are events that can occur between a start event and an end event (intermediate event) which can affect the flow of the process but cannot start it or end it. Finally, gateways are represented as a diamond shape and are responsible for controlling divergence (split) and convergence (join) of sequence flows in a process. There are six different types of gateways which differ in both the logic that they execute and the representation placed within the gateway diamond. Among them, it can be highlighted: exclusive gateway (XOR, represented with or without a “X” marker), where the decision making leads to the execution of exactly one path; parallel gateway (AND, represented with a “+” marker), where all possible paths must be executed; and inclusive gateway (OR, represented with a “O” marker), where decision making leads to the execution of at least one path.

In relation to swimlane, a pool represents a participant in a business process. A pool is graphically represented as a container that partitions a process from

other participants. If a pool does not contain a process, it is considered as a black box. Lane, on the other hand, are the partitions used to organize and categorize activities within a pool. Lanes are often used for representing internal roles (e.g., Manager, Associate), systems (e.g., an enterprise application), or an internal department (e.g., shipping, finance).

Regarding connecting objects, sequence flows are used to show the order of flow objects in a process. A sequence flow is represented as a solid single line with a solid arrowhead. A message flow represents the flow of messages between two different participants and is represented as a dashed single line with an open circle line start and an open arrowhead line end. In addition, an association is used to link information and artifacts with flow objects. Data associations, on the other hand, are used to relate data objects and activities. Both association and data association are represented as a dotted single line.

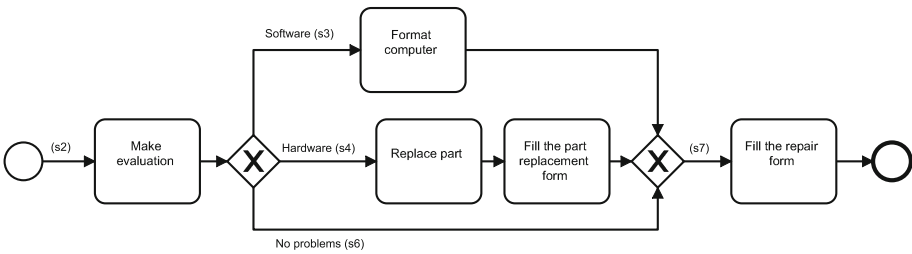


Fig. 2. Example of BPMN model: computer repair.

Figure 2 presents an example of a BPMN process model composed by one start event, five activities, one exclusive decision gateway (XOR-split), one exclusive merge gateway (XOR-join) and one end event. After the process starts, an activity is executed (called “Make evaluation”). Then, there is a decision making in which only one of three possible paths can be followed. After one path is followed, the process returns to the main path, another activity is performed and the process ends. A possible description of the process shown in Fig. 2 can be seen in Fig. 3. The relationships between the text and the model are evidenced through s_x , where x refers to the sentence number in the text.

3 Sentence Templates and Ambiguity Issues

This section presents the method used to identify the sentence templates.

The process descriptions used in the analysis came from two different sources. Only process descriptions in English were considered, as templates are very sensitive to the language. Firstly, 47 process descriptions present in Friedrich [13] were identified. From this first source, 17 process descriptions were disregarded for the following reasons: they were translated from another language through machine translation services (14 texts), were duplicated (2 texts) or had a description

(s_1) The repair department of the company X performs repairs of computers and printers. (s_2) Once a computer with problems arrives, the technician must perform an evaluation. (s_3) In case it is a software problem, the technician must format the computer. (s_4) For the case that it is a hardware problem, the technician must replace the part and fill out the part replacement form. (s_5) This form must contain the part identification code and the technician's signature. (s_6) On the other hand, if no problem is found, no modification should be made to the computer. (s_7) The process finishes after the technician completes the repair form.

Fig. 3. Example of process description: computer repair.

format based on enumeration (1 text). Secondly, 34 process descriptions from the book *Fundamentals of Business Process Management* [9] were identified. The final set of 64 process descriptions, as well as their respective sources and types are presented in Table 1.

Table 1. Data sources.

Source	Amount	Type
HU Berlin	4	Academic
TU Berlin	2	Academic
QUT	8	Academic
TU Eindhoven	1	Academic
Vendor Tutorials	4	Industry
inubit AG	3	Industry
BPM Practitioners	1	Industry
BPMN Prac. Handbook	3	Textbook
BPMN M&R Guide	4	Textbook
Fundamentals of BPM	34	Textbook
Total	64	–

The following subsections present the procedures followed to: (Sect. 3.1) prepare the sentences, (Sect. 3.2) identify and classify the sentence templates and (Sect. 3.3) address the ambiguity issues.

3.1 Preparation of Sentences

In this first stage, the sentences are prepared for the identification and classification of sentence templates. For this, the sentences of a process description are modified to become more generic.

A business process description may contain snippets of text that are directly related to the process context. As an example, the sentences “The manager must sell the product” and “The salesman must sell the product” are identical, except by who carries out the activity of selling the product. This difference can hinder the identification and classification of sentence templates, since these sentences can be considered as different sentence templates. In this sense, a term capable of representing both “manager” and “salesman” could be used in order to make these two sentences equal and, consequently, to define both as the same sentence template. Thereby, the process descriptions were previously analyzed and four different placeholders were created with the goal to replace in the sentences the snippets related to the context by more generic information. The created placeholders are: *role*, *condition*, *number* and *object*.

The placeholder *role* is associated with the role responsible for performing a particular action. In relation to the business process model, a role could be considered as a participant. As an example, in the sentence “The process finishes after the technician completes the repair form.”, once the technician is the one performing the action, the word “technician” can be replaced by the placeholder *role*. Therefore, the sentence after the modification would be written as “The process finishes after the *role* completes the repair form”.

The placeholder *condition* aims to define some condition that needs to be satisfied for a given flow to occur. Normally, the condition appears in a business process model as a label that tracks the output sequence flow of an exclusive or inclusive gateway. For instance, in the sentence “In case it is a software problem, the technician must format the computer.” it is possible to observe that to be done the activity of formatting the computer must exist before the condition “it is a software problem”. Therefore, this condition will be replaced in the text by the placeholder *condition*. Moreover, the technician can also be replaced in this sentence by the placeholder *role*.

The placeholder *number* is used to represent a certain amount of process elements or paths in a process model. As an example, in the sentence “After all five activities are completed, the process ends.”, the amount “five” can be replaced by the placeholder *number*.

Finally, the placeholder *object* can represent the business object to which the sentence refers. For instance, in the sentence “The car can be sold by the manager or the seller”, the business object “car” can be replaced by the placeholder *object*. In addition, the placeholders *role 1* and *role 2* could be created to represent the manager and seller respectively.

After the preparation stage, the modified sentences containing placeholders will be used for the identification and classification of sentence templates.

3.2 Identification and Classification of Sentence Templates

In the context of this work, a sentence template was considered as each pattern present in a sentence that is able to describe one or more process elements. These process elements appear in the template as placeholders to be replaced. For the scope of this paper, a reduced set of elements is taken into account to

find sentence templates, being: activity (A_c), AND-split (G_{+s}), AND-join (G_{+j}), XOR-split (G_{Xs}), XOR-join (G_{Xj}), OR-split (G_{Os}), OR-join (G_{Oj}), start event (E_s), intermediate event (E_i), end event (E_e). In addition, “empty” is used to define paths without elements (e.g., Fig. 2, s_6).

In order to identify a sentence template, it is necessary to identify beforehand the process elements in the text. Although there are works that contribute to the automated identification of process elements in texts, to the best of our knowledge, there is no approach capable of extracting the process elements in a textual description with complete precision [10,12,14]. In addition, automated identification approaches can draw incorrect conclusions about a process by making assumptions about texts that allow for multiple interpretations [2]. Therefore, an automated analysis of sentence templates could be compromised by the selected approach of extracting process elements, so the identification of the sentence templates was carried out manually.

The identification and classification of sentence templates were carried out in parallel. In the context of this paper, each sentence template is considered as composed by the following elements:

- **Target:** the set of process elements described by the sentence template. A target must appear in the sentence, even if implicitly (i.e., without a placeholder to fill with the process element).
- **Relationship:** how the process elements in the sentence are associated to each other: none (R_N), composed by 0 or 1 process element; sequential (R_S), one element follows the other; exclusive (R_X); inclusive (R_O); and parallel (R_+).
- **Source:** the process element that occurs immediately before the analyzed sentence. As in the BPMN specification [22], the source can be understood as the element prior to the currently described element connected by a sequence flow. A source may or may not be evidenced in the sentence.

As an example, for the process description presented in Fig. 3 (corresponding to the BPMN model depicted in Fig. 2), five sentence templates were identified, two of which has target with the sequential relationship (s_4 , s_7) and three has target with the none relationship (s_2 , s_3 , s_6). In the sentence s_2 , it is possible to define the sentence template “Once E_s , the *role* must A_c ”, where E_s is the source evidenced in the sentence template, A_c is a placeholder for an activity described in the target and *role* refers to some participant in the process that performs the activity A_c . The sentences s_3 , s_4 and s_6 have as source a XOR-split gateway not evidenced in the sentence template. In addition, the sentence s_7 has as its source a XOR-join gateway and as target an activity (A_c) and an end event (represented implicitly by “The process finishes after”). Not all sentences in a text are necessarily mapped to a sentence template, since process descriptions can be composed by other information, such as statements that contextualize the process (s_1) and statements that detail an activity or business rule (s_5).

In order to identify the sentence templates, each process description was inserted into a spreadsheet, as illustrated in Table 2. In the spreadsheet, each line represents one sentence and the columns represent the following attributes:

sentence, sentence template ID (i.e., the ID of the sentence template that can be a number or “none”) and sentence template.

Table 2. Example of identification of sentence templates.

Process description: computer repair		
Sentence	Sentence template ID	Sentence template
The repair department of the company X performs repairs of computers and printers	None	
Once a computer with problems arrives, the technician must perform an evaluation	3	Once E_s , the <i>role</i> must A_c
In case it is a software problem, the technician must format the computer	10	In case <i>condition</i> , the <i>role</i> must A_c
For the case that it is a hardware problem, the technician must replace the part and fill out the part replacement form	25	For the case <i>condition</i> , the <i>role</i> must A_c and A_c
This form must contain the part identification code and the technician’s signature.	None	
On the other hand, if no problem is found, no modification should be made to the computer	1	On the other hand, if <i>condition</i> , <i>empty</i>
The process finishes after the technician completes the repair form	8	The process finishes after the <i>role</i> A_c

After all the sentence templates were identified, they were grouped into categories based on source, target and relationship. As a result, each category is composed by sentence templates that can be replaced in a process description and represent the same information. As an example, the sentence s_3 is defined as a sequential relationship between a XOR-split (source) and an activity (target). This sentence can be rewritten by another sentence template that have the same properties, therefore the same category, such as: “Once *condition*, the *role* needs to A_c ”.

The analysis of sentence templates was done in two different manners, namely atomic level analysis and group level analysis. At the atomic level analysis, it is considered that if two sentences have the same text, but represent different process elements in the source, they are defined as two distinct sentence templates. For example, sentences “When a computer with problems arrives, the technician must perform an evaluation.” and “When performing a repair, the technician must perform an evaluation.” are defined as different sentence templates because they have different process elements in the source, being respectively: “When E_s , A_c .” and “When A_c , A_c .”. On the other hand, at the grouped

level it is considered that different possible process elements can be translated as the same sentence template. In this case, the two sentence templates described above can be viewed as a single sentence template (i.e., “When (A_c or E_s), A_c .”), capable to have as source either an activity or a start event.

To facilitate the categorization of sentence templates, a notation was created based on the previously defined criteria. $St_i = R_s(source, target)$ can be interpreted as: there is a sentence template St_i that starts from a *source*, can describe a *target* and is associated through a sequential relationship R_s . In the case of atomic level analysis, a *source* is a process element. On the other hand, in the case of group level analysis a *source* is a set of possible process elements (e.g., $A_c|G_{X_s}|G_{+s}$). A target can be described as $target = R_x(component_1, \dots, component_n)$, i.e., a target is a set of components that relate to each other through a relationship R_x . Finally, a component can be a process element, *empty*, or another target. Thus, two different sentence templates belong to the same category if they share the same notation, which means to start from the same *source* and reach the same *target*.

3.3 Ambiguity in Sentence Templates

After the identification and classification of the sentence templates, they were analyzed in relation to ambiguity issues. A sentence template was considered ambiguous when it allows multiple interpretations of the process. To identify common ambiguity issues in process descriptions, two approaches were carried out: analysis of the literature and analysis of the sentence templates.

As for the analysis of the literature, works related to ambiguity in process descriptions were investigated. Although some works related to this subject were found, only a few of them [1–4] presented cases of ambiguity. This analysis of the literature made it possible to find eight ambiguity problems that were categorized into five different types of ambiguity.

In terms of the analysis of the sentence templates, two independent tasks were conducted. In the first part, an analysis of each sentence template was carried out individually in order to identify ambiguity issues. In the second part, an analysis was carried out involving the combination of sentence templates. For the latter case, sentence templates that share the same description, but do not have the same classification (i.e., source, target or relationship) were considered candidates for ambiguity issues.

Table 3 presents the six different types of ambiguities that were identified in this work, with their respective identifiers ($Ambi_{ID}$), descriptions, examples and source.

4 Results and Analysis

After analyzing the process descriptions in an atomic level, it was possible to obtain a set of 101 sentence templates for 29 categories. Of these, 13 sentence templates were classified as having one of the six ambiguity issues. Tables 4,

Table 3. Identified ambiguity issues.

<i>Ambi_{ID}</i>	Description	Example	Source
<i>Ambi₁</i>	The term “and” can have different meanings, such as: sequence, dependence, parallelism, contrast	<i>The employee must update the document and prepare the product for shipping</i>	- Akbar et al. [4] - van der Aa et al. [2,3]
<i>Ambi₂</i>	The terms “or” and “sometimes” may raise doubts whether it includes or is mutually exclusive to the different alternatives	<i>(1) The document is accepted or denied. (2) The bicycle can be mounted or painted</i>	- Akbar et al. [4] - this paper
<i>Ambi₃</i>	The term “latter” usually does not make clear to what previous activities it refers	<i>In parallel to the latter steps...</i>	- van der Aa et al. [2,3]
<i>Ambi₄</i>	The terms “meanwhile”, “concurrently”, “meantime”, “in the meantime” and “at the same time” make it difficult to specify which sets of activities they refer to	<i>In the meantime, the sales department must prepare the receipt</i>	- van der Aa et al. [3] - this paper
<i>Ambi₅</i>	Repetitions usually not clear what activities should be performed again	<i>The previous steps must be repeated</i>	- van der Aa et al. [1-3]
<i>Ambi₆</i>	The term “while” can mean simultaneity or concession	<i>While it is true that the company has the money, they can't build the houses</i>	- this paper

5 and 6 show the sentence templates for each one of the 29 categories (*C_{ID}*), with their respective category notation. Each sentence template has a specific identifier presented in the “*St_{ID}*” column. In addition, the number of times each sentence template appeared in the process descriptions analyzed is presented in the “N” column. Moreover, the ambiguity issues identified for each sentence template, when identified, is presented in the “*Ambi_{ID}*” column, based on the elements in Table 3.

Of the identified sentence templates, the most recurrent is “If *condition*, *A_c*.” (*St₇₁*), from category *C₁₈*, which appeared 81 times. It is possible to observe that this sentence template is fairly recurrent in process descriptions because the two sentence templates that appear the most after this first (i.e., *St₁* and *St₃₆*) were identified only 15 times. Moreover, the category that presented the largest diversity of sentence templates is *C₁*, with 19 distinct sentence templates, followed by *C₁₈* (with 12) and *C₃* (with 10).

In terms of ambiguity, the type that appeared most in the sentence templates is related to the term “and” (*Ambi₁*), having occurred five times. Moreover, *Ambi₂* appeared three times, followed by both *Ambi₄* and *Ambi₆* (2 times), and *Ambi₅* (1 time). In addition, in the identified sentence templates no case was found related to the ambiguity *Ambi₃*.

Table 4. Atomic sentence templates by category – 1.

C_{ID}	Notation	St_{ID}	Sentence template	N	$Ambi_{ID}$
C_1	$R_S(A_c, R_N(A_c))$	St_1	Once A_c , A_c	15	
		St_2	Then A_c	13	
		St_3	When A_c , A_c	11	
		St_4	After A_c , A_c	10	
		St_5	Next A_c	5	
		St_6	Afterwards, A_c	4	
		St_7	As soon as A_c , A_c	3	
		St_8	Subsequently A_c	3	
		St_9	The <i>role</i> then A_c	3	
		St_{10}	Upon A_c	3	
		St_{11}	After that A_c	2	
		St_{12}	Likewise A_c	2	
		St_{13}	A_c , after which A_c	1	
		St_{14}	Immediately after A_c , A_c	1	
		St_{15}	In addition to A_c , A_c	1	
		St_{16}	In the following A_c	1	
		St_{17}	Moreover, A_c	1	
		St_{18}	Thereafter A_c	1	
		St_{19}	Therefore A_c	1	
C_2	$R_S(A_c, R_N(E_e))$	St_{20}	After A_c , the process ends	1	
		St_{21}	After all <i>number</i> activities are completed the process ends	1	
		St_{22}	For <i>role</i> the process ends then	1	
		St_{23}	The process ends here	1	
		St_{24}	The process finishes only once A_c	1	
C_3	$R_S(A_c, R_N(G_{X_s}))$	St_{25}	After A_c , the <i>object</i> may lead to <i>number</i> possible outcomes: <i>condition</i> or <i>condition</i>	2	$Ambi_2$
		St_{26}	After A_c , it is checked whether <i>condition</i>	1	
		St_{27}	After A_c , the <i>role</i> can either <i>condition</i> , <i>condition</i> or <i>condition</i>	1	
		St_{28}	After A_c , the <i>role</i> investigates whether <i>condition</i> or <i>condition</i>	1	
		St_{29}	After A_c , the <i>role</i> may either <i>condition</i> or <i>condition</i>	1	
		St_{30}	One of the <i>number</i> alternative process paths may be taken	1	
		St_{31}	The <i>role</i> can either <i>condition</i> or <i>condition</i>	1	
		St_{32}	The <i>role</i> can then <i>condition</i> or <i>condition</i>	1	$Ambi_2$
		St_{33}	This procedure is repeated for each <i>condition</i>	1	$Ambi_5$
		St_{34}	When A_c , it is first checked whether <i>condition</i>	1	

Table 5. Atomic sentence templates by category – 2.

C_{ID}	Notation	St_{ID}	Sentence template	N	$Ambi_{ID}$
C_4	$R_S(A_c, R_N(G_{X_j}))$	St_{35}	Then the process continues normally	1	
C_5	$R_S(A_c, R_S(A_c, A_c))$	St_{36}	A_c and A_c	15	$Ambi_1$
		St_{37}	Also A_c and A_c	1	$Ambi_1$
C_6	$R_S(A_c, R_S(A_c, A_c, A_c))$	St_{38}	A_c and A_c and A_c	1	$Ambi_1$
		St_{39}	First A_c , then A_c , and finally A_c	1	
C_7	$R_S(A_c, R_S(A_c, E_e))$	St_{40}	A_c , which ends the process	2	
		St_{41}	Finally, A_c	2	
		St_{42}	The process completes with A_c	2	
		St_{43}	A_c , then the process ends	1	
		St_{44}	After A_c , this process path ends	1	
		St_{45}	Afterwards, A_c and finishes the process instance	1	
		St_{46}	The process finishes when A_c	1	
C_8	$R_S(A_c, R_S(A_c, G_{X_s}))$	St_{47}	The <i>role</i> A_c and may decide to either A_c or A_c	1	$Ambi_1$
C_9	$R_S(A_c, R_+(A_c, A_c))$	St_{48}	While A_c , A_c	2	$Ambi_6$
		St_{49}	Next, A_c while A_c	1	$Ambi_6$
		St_{50}	Once A_c , A_c and meantime A_c	1	
C_{10}	$R_S(A_c, R_X(A_c, A_c))$	St_{51}	After A_c , the <i>role</i> either A_c or A_c	1	
		St_{52}	When A_c , the <i>role</i> may either A_c or A_c	1	
C_{11}	$R_S(A_c, R_X(A_c, A_c, A_c))$	St_{53}	Sometimes A_c , sometimes A_c and sometimes A_c	1	$Ambi_2$
C_{12}	$R_S(A_c, R_O(A_c, A_c))$	St_{54}	<i>object</i> may be A_c from either <i>role 1</i> or <i>role 2</i> or from both	1	
		St_{55}	The <i>role</i> may either A_c or also A_c	1	
C_{13}	$R_S(E_i, R_N(E_e))$	St_{56}	After <i>role</i> E_i , the process flow ends	1	
C_{14}	$R_S(E_s, R_N(A_c))$	St_{57}	The process starts with A_c	7	
		St_{58}	The process starts when A_c	6	
		St_{59}	First, A_c	3	
		St_{60}	The process starts by A_c	2	
		St_{61}	When E_s , A_c	2	
		St_{62}	Whenever E_s , A_c	2	
		St_{63}	After the process starts, A_c	1	
		St_{64}	The process is triggered by A_c	1	
		St_{65}	The process starts once A_c	1	
C_{15}	$R_S(E_s, R_S(A_c, A_c, A_c))$	St_{66}	A_c and A_c and A_c	1	$Ambi_1$
C_{16}	$R_S(G_{+s}, R_N(A_c))$	St_{67}	In the meantime, A_c	2	$Ambi_4$
		St_{68}	At the same time, A_c	1	$Ambi_4$
C_{17}	$R_S(G_{+j}, R_N(A_c))$	St_{69}	Afterwards, A_c	2	
		St_{70}	After each of these activities, A_c	1	

Table 6. Atomic sentence templates by category – 3.

C_{ID}	Notation	St_{ID}	Sentence template	N	$Ambi_{ID}$
C18	$R_S(G_{X_s}, R_N(A_c))$	St_{71}	If <i>condition</i> , A_c	81	
		St_{72}	Otherwise, A_c	10	
		St_{73}	In this case A_c	7	
		St_{74}	In case <i>condition</i> , A_c	3	
		St_{75}	For the case <i>condition</i> , A_c	2	
		St_{76}	<i>condition</i> , in which case A_c	1	
		St_{77}	<i>condition</i> , otherwise A_c	1	
		St_{78}	However, if <i>condition</i> , A_c	1	
		St_{79}	In that case, A_c	1	
		St_{80}	In the latter case, A_c	1	
		St_{81}	On the other hand, if <i>condition</i> A_c	1	
		St_{82}	Sometimes <i>condition</i> , then A_c	1	
C19	$R_S(G_{X_s}, R_N(E_e))$	St_{83}	If <i>condition</i> the process will end	1	
		St_{84}	In the former case, the process instance is finished	1	
C20	$R_S(G_{X_s}, R_N(G_{+s}))$	St_{85}	This action consists of <i>number</i> activities, which are executed in an arbitrary order	1	
C21	$R_S(G_{X_s}, R_N(G_{X_s}))$	St_{86}	If <i>condition</i> , this results in either <i>condition</i> or <i>condition</i>	1	
C22	$R_S(G_{X_s}, R_S(A_c, A_c, A_c))$	St_{87}	If <i>condition</i> , <i>role</i> may need to first A_c , then A_c and finally A_c	1	
C23	$R_S(G_{X_s}, R_+(A_c, A_c))$	St_{88}	Once <i>condition</i> , A_c and meantime A_c	3	
C24	$R_S(G_{X_s}, R_X(A_c, A_c))$	St_{89}	If <i>condition</i> , A_c , otherwise A_c	4	
		St_{90}	In case <i>condition</i> , A_c , otherwise A_c	1	
		St_{91}	<i>role</i> either A_c or A_c	1	
C25	$R_S(G_{X_s}, R_X(A_c, E_e))$	St_{92}	If <i>condition</i> A_c , otherwise the process is finished	1	
C26	$R_S(G_{X_s}, R_X(A_c, empty))$	St_{93}	If <i>condition</i> , A_c , except if <i>condition</i>	1	
		St_{94}	In case <i>condition</i> , A_c otherwise the process continues	1	
C27	$R_S(G_{X_j}, R_N(A_c))$	St_{95}	In any case, A_c	4	
		St_{96}	In either/any case, A_c	1	
		St_{97}	Once one of these <i>number</i> activities is performed, A_c	1	
		St_{98}	The process then continues with A_c	1	
C28	$R_S(G_{X_j}, R_S(A_c, E_e))$	St_{99}	Afterwards, A_c and the process completes	1	
		St_{100}	Finally, A_c	1	
C29	$R_S(G_{X_j}, R_+(A_c, A_c))$	St_{101}	Then, two current activities are triggered, A_c and A_c	1	

Furthermore, it is possible to notice that not all relationships between process elements are explored in Tables 4, 5 and 6. This occurs because some relationships that occur in the model are not explicitly transformed into sentences. Also, there are some relationships that appear less frequently than others in the texts considered.

In the group level analysis, the atomic level sentence templates that share the same target but presents different process elements as sources were grouped. From the data collected in the atomic level analysis, it was possible to identify 8 sentence templates that were transformed into four grouped sentence templates. Table 7 presents the grouped sentence templates. In this table are presented the new notations able to represent the grouped sentence templates, as well as the new sentence templates. In addition, the identifier of the atomic sentence templates used in each grouped sentence template are presented in the “ St_{ID} ” column. Finally, as in atomic level analysis, the number of times each grouped sentence template appeared in the process descriptions analyzed is presented in the “N” column.

Table 7. Grouped sentence templates.

Notation	Sentence template	St_{ID}	N
$R_s(A_c E_s, Rn(A_c))$	When $A_c E_s$, A_c	St_3, St_{61}	13
$R_s(A_c G_{+j}, Rn(A_c))$	Afterwards, A_c	St_6, St_{69}	6
$R_s(A_c GX_j, R_s(A_c, E_e))$	Finally, A_c	St_{41}, St_{100}	3
$R_s(A_c E_s, R_s(A_c, A_c, A_c))$	A_c and A_c and A_c	St_{38}, St_{66}	2

5 Discussion

The identified sentence templates can help approaches for identification of process elements in natural language texts and for automated creation of business process descriptions. For identification of process elements in natural language texts, the approaches can use the identified sentence templates as patterns to be sought in texts. In this case, sentence templates could be searched in the sentences of a process description. By finding a sentence corresponding to a sentence template, the process elements present in the sentence could be identified.

For automated creation of process descriptions, the approaches can choose to use the most recurring sentence templates, or take advantage of the variety of sentence templates in each category to make the text more diversified. As a demonstration, the text described in Fig. 3 could with the use of the identified sentence templates be rewritten as presented by Fig. 4. In this new process description, the sentence s_1 remained the same because it only presents context information. In addition, the sentence s_2 (belonging in the category C_{14}) was modified by the sentence template “The process starts with A_c ” (St_{57}) because it is recurrent in the process descriptions analyzed. Moreover, the sentences s_3

and s_6 (both belonging to category C_{18}) were rewritten using the sentence template “If *condition*, A_c ” (St_{71}) because this sentence template appears recurrent in describing activities starting from an XOR-split. The sentence s_5 also has not been modified because this sentence only details an activity. Furthermore, the sentence s_6 has an ambiguity problem related to “and” ($Ambi_1$). Although the sentence templates collected do not present any candidate capable of removing the problem, the results help indicate that there is a problem in the process description. Among possible solutions, the sentence could separate into two new sentences or seek for a different sentence capable of avoiding the problem of ambiguity, such as: “If it is a hardware problem, the technician must replace the part and then fill out the part replacement form.”. Finally, the sentence s_7 (belonging to category C_7) was modified by the sentence template “Finally, A_c ” (St_{41}) because it appears to be most recurrent in the identified sentence templates.

(s_1) The repair department of the company X performs repairs of computers and printers. (s_2) The process starts with the technician performing an evaluation. (s_3) If it is a software problem, the technician must format the computer. (s_4) If it is a hardware problem, the technician must replace the part and then fill out the part replacement form. (s_5) This form must contain the part identification code and the technician’s signature. (s_6) If no problem is found, no modification should be made to the computer. (s_7) Finally, the technician must complete the repair form.

Fig. 4. Example of rewritten process description: computer repair.

Regarding categorization of sentence templates, although the classification based on source, target and relationship helps in the task of grouping sentence templates that have the same characteristics, in some cases an analysis of the context can help to select a sentence template that more fits the text and, consequently, to produce a text more suitable for process analysts and domain experts. For instance, the sentence templates “If condition, A_c ” (St_{71}) and “Otherwise, A_c ” (St_{72}) are related in the same category (C_{18}) and are both able to represent an activity being performed after an XOR-split. However, the second sentence template could produce a disconnected text by referring to the first possible path of a decision making in a process description.

6 Related Work

The work presented here relates to two different streams of research: generation of process descriptions and ambiguities present in process descriptions.

For the generation of process descriptions context, Leopold *et al.* [18] proposed a technique to generate natural language texts from business process models. The authors used sentence templates to transform business process models

into sentences that compose the text. Furthermore, Aysolmaz *et al.* [7] defined a semi-automated approach to generate natural language requirements documents based on business process models. The authors adopted a template filling technique, in which sentence templates are defined containing gaps that must be filled with information from a requirements model. In addition, Caporale [8] suggested a method that allows generating process models from process descriptions. To achieve this, the author proposed that the process descriptions should be specified with a controlled natural language, based on sentence templates, in order to facilitate the extraction of information necessary to generate the models. Moreover, Ghose *et al.* [15] proposed a framework and prototype tool that can query information resources (e.g. corporate documentation, web-content, code) for construct models to be incrementally adjusted to correctness by an analyst. One of the techniques used by authors to extract information from text is based on template extraction. In this technique, the authors created templates from textual structures that are commonly used in describing processes and used these templates to extract knowledge from text documents.

Regarding the ambiguity present in process descriptions, Ferrari *et al.* [11] conducted a literature review and a set of interviews with different public institutions aiming at improving the process descriptions to be used in public administrations. The authors concluded that ambiguity is one of the macro-areas of research in which computer scientists can contribute towards more quality in business process descriptions. Moreover, van der Aa *et al.* [1] presented an approach to automatically detect inconsistencies between process model and the corresponding textual description. The authors identified that a technique to detect inconsistencies must deal with ambiguity issues present in natural language. In another work, van der Aa *et al.* [2] proposed to deal with ambiguity in textual process descriptions introducing the behavioral space concept. The behavioral space captures all possible behavioral interpretations of a textual process description. Furthermore, van der Aa *et al.* [3] presented an approach to verify the compliance between a process and a process description, considering the ambiguity present in texts. To handle the ambiguity issues, they used the concept of behavioral space previously proposed in [2].

As can be observed, the first stream of research makes use of sentence templates for generation of business process descriptions from process models and extraction of text information for the generation of process models. Moreover, the second stream discusses the problem of ambiguity in process descriptions and presents some cases of ambiguity. This work is distinguished by the fact that it presents an empirical analysis in business processes descriptions, in order to find recurrent sentence templates and to highlight ambiguity issues in these sentence templates.

7 Conclusion

In this work, an empirical analysis of business process descriptions was carried out in order to discover the most recurrent sentences used to describe BPMN

process models. In addition, an analysis was performed in order to find sentences with ambiguous meaning. The analysis consisted of three different steps. First, a set of 64 process descriptions was selected and their sentences were prepared for the identification of sentence templates. Then, an identification and a classification of sentence templates was performed in the prepared sentences. Finally, the sentence templates were marked as having or not ambiguity issues. Among all, 101 sentence templates were found and they were classified into 29 different categories. Of these, 13 sentence templates were considered as having ambiguity issues.

This work aims to contribute to the description of business processes in a way that is closer to a pattern and with less ambiguity issues. It can be useful for creating process descriptions more suitable for process analysts and domain experts. In addition, this analysis can be used by tools that automate the creation of process descriptions. As future work, it is intended to increase this analysis for other elements in BPMN, expand the sample of textual descriptions and construct a technique that uses these sentence templates to produce process descriptions.

Acknowledgments. We are grateful to the national research funding agencies CNPq (National Council for Scientific and Technological Development) and CAPES (CAPES Foundation) for financial support. We also acknowledge the support we have been receiving from the Graduate Program in Computer Science as well as the Institute of Informatics, UFRGS.

References

1. van der Aa, H., Leopold, H., Reijers, H.A.: Detecting inconsistencies between process models and textual descriptions. In: Motahari-Nezhad, H.R., Recker, J., Weidlich, M. (eds.) BPM 2015. LNCS, vol. 9253, pp. 90–105. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23063-4_6
2. van der Aa, H., Leopold, H., Reijers, H.A.: Dealing with behavioral ambiguity in textual process descriptions. In: La Rosa, M., Loos, P., Pastor, O. (eds.) BPM 2016. LNCS, vol. 9850, pp. 271–288. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-45348-4_16
3. van der Aa, H., Leopold, H., Reijers, H.A.: Checking process compliance against natural language specifications using behavioral spaces. *Inf. Syst.* (2018)
4. Akbar, S., Bajwa, I.S., Malik, S.: Scope resolution of logical connectives in NL constraints. In: Eighth International Conference on Digital Information Management, pp. 217–222. IEEE (2013)
5. de AR Gonçalves, J.C., Santoro, F.M., Baiao, F.A.: Business process mining from group stories. In: 13th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2009, pp. 161–166. IEEE (2009)
6. de AR Gonçalves, J.C., Santoro, F.M., Baião, F.A.: Let me tell you a story-on how to build process models. *J. UCS* **17**(2), 276–295 (2011)
7. Aysolmaz, B., Leopold, H., Reijers, H.A., Demirörs, O.: A semi-automated approach for generating natural language requirements documents based on business process models. *Inf. Softw. Technol.* **93**, 14–29 (2018)

8. Caporale, T.: A tool for natural language oriented business process modeling. In: ZEUS, pp. 49–52 (2016)
9. Dumas, M., La Rosa, M., Mendling, J., Reijers, H.A., et al.: Fundamentals of Business Process Management, vol. 1. Springer, Heidelberg (2013). <https://doi.org/10.1007/978-3-642-33143-5>
10. Epure, E.V., Martín-Rodilla, P., Hug, C., Deneckère, R., Salinesi, C.: Automatic process model discovery from textual methodologies. In: 2015 IEEE 9th International Conference on Research Challenges in Information Science (RCIS), pp. 19–30. IEEE (2015)
11. Ferrari, A., Witschel, H.F., Spagnolo, G.O., Gnesi, S.: Improving the quality of business process descriptions of public administrations: resources and research challenges. *Bus. Process Manag. J.* **24**, 49–66 (2017)
12. Ferreira, R.C.B., Thom, L.H., de Oliveira, J.P.M., Avila, D.T., dos Santos, R.I., Fantinato, M.: Assisting process modeling by identifying business process elements in natural language texts. In: de Cesare, S., Frank, U. (eds.) ER 2017. LNCS, vol. 10651, pp. 154–163. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70625-2_15
13. Friedrich, F.: Automated generation of business process models from natural language input. M.Sc., School of Business and Economics. Humboldt-Universität zu Berli (2010)
14. Friedrich, F., Mendling, J., Puhlmann, F.: Process model generation from natural language text. In: Mouratidis, H., Rolland, C. (eds.) CAiSE 2011. LNCS, vol. 6741, pp. 482–496. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21640-4_36
15. Ghose, A., Koliadis, G., Chueng, A.: Process discovery from model and text artefacts. In: 2007 IEEE Congress on Services, pp. 167–174. IEEE (2007)
16. Leopold, H., van der Aa, H., Pittke, F., Raffel, M., Mendling, J., Reijers, H.A.: Integrating textual and model-based process descriptions for comprehensive process search. In: Schmidt, R., Guédria, W., Bider, I., Guerreiro, S. (eds.) BPMDS/EMMSAD -2016. LNBIP, vol. 248, pp. 51–65. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39429-9_4
17. Leopold, H., Mendling, J., Polyvyanyy, A.: Generating natural language texts from business process models. In: Ralyté, J., Franch, X., Brinkkemper, S., Wrycza, S. (eds.) CAiSE 2012. LNCS, vol. 7328, pp. 64–79. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-31095-9_5
18. Leopold, H., Mendling, J., Polyvyanyy, A.: Supporting process model validation through natural language generation. *IEEE Trans. Softw. Eng.* **40**(8), 818–840 (2014)
19. Li, J., Wang, H.J., Zhang, Z., Zhao, J.L.: A policy-based process mining framework: mining business policy texts for discovering process models. *Inf. Syst. E-Bus. Manag.* **8**(2), 169–188 (2010)
20. Meitz, M., Leopold, H., Mendling, J.: An approach to support process model validation based on text generation. In: EMISA Forum, vol. 33, pp. 7–20 (2013)
21. Nawrocki, J.R., Nedza, T., Ochodek, M., Olek, L.: Describing business processes with use cases. In: BIS, pp. 13–27 (2006)
22. OMG: Business process modeling notation (BPMN) version 2.0.2 (2013). <https://www.omg.org/spec/BPMN/>
23. Ottensooser, A., Fekete, A., Reijers, H.A., Mendling, J., Menictas, C.: Making sense of business process descriptions: an experimental comparison of graphical and textual notations. *J. Syst. Softw.* **85**(3), 596–606 (2012)

24. Sánchez-Ferreres, J., Carmona, J., Padró, L.: Aligning textual and graphical descriptions of processes through ILP techniques. In: Dubois, E., Pohl, K. (eds.) CAiSE 2017. LNCS, vol. 10253, pp. 413–427. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-59536-8_26
25. Sinha, A., Paradkar, A.: Use cases to process specifications in business process modeling notation. In: 2010 IEEE International Conference on Web Services (ICWS), pp. 473–480. IEEE (2010)