The Semantic of Business Vocabulary and Business Rules: An Automatic Generation From Textual Statements

ABDELLATIF HAJ, ABDESSAMD JARRAR, YOUSSEF BALOUKI, AND TAOUFIQ GADIR
Laboratory of Mathematics, Computer and Engineering Sciences, Department of Maths, Faculty of Sciences and Techniques, Hassan I University, Settat 26000, Morocco
Corresponding author: Abdellatif Haj (haj.abdel@gmail.com)

ABSTRACT In the early phases of the software development process, specifications are mostly written in a natural language rather than formal models, which is not supported by the Model Driven Architecture (MDA). For this reason, the Semantic of Business Vocabulary and Rules (SBVR) is proposed by the Object Management Group to represent the textual specifications in a language comprehensible by both of humans and machines, to facilitate its integration in the MDA lifecycle. However, businesspeople are usually not familiar with SBVR standard. In this paper we present an approach to automatically transform textual business rules to an SBVR model, to facilitate its integration in nowadays information technology infrastructures. Our approach is distinguished from existing works in that it uses an in-depth Natural Language Processing to extract a more comprehensible SBVR model that includes the semantic formulation of each business rule statement, coupled with a Terminological Dictionary of extracted concepts, to which we have added further specifications such as definitions and synonyms. The evaluation of our approach shows that for three sets of business rules statements taken from different domains, we could generate the correct meaning with an average of F1-score exceeding 87%.

INDEX TERMS Business rules, business vocabulary, SBVR, model transformation, MDA, natural language processing.

I. INTRODUCTION

Adapting software applications to a constantly evolving technologies is a major challenge for organizations. As a solution, the Object Management Group (OMG) [1] has proposed the Model Driven Architecture (MDA) approach, in which the business aspect and the technical aspect of applications are separated and presented using models. Then, a set of models' transformations are used to generate the source code.

MDA defines three levels of application models on which a series of transformations are performed to build a software system:

1. Computational Independent Model (CIM): describes the functional needs of the application, independently of its implementation details.
2. Platform Independent Model (PIM): includes analysis and design models of the application independently of the technical design of the application.
3. Platform Specific Model (PSM): describes the implementation of the application on a specific platform.

Models help to better communicate ideas, but intervenors at organizations are usually not familiar with diagrams notations. For this reason, in the early phases of a software development process, requirements and business specifications are mostly written in a natural language (NL) rather than formal models that are understandable by machines. Thus, its transformation to CIM model is not part of the MDA lifecycle, and tends to be performed manually, which makes it not only error prone and time consuming, but also its validation is one-sided.

One of the most critical resources on which CIM model is based are Business Rules (BR), as they tend to be changed frequently and quickly. Unfortunately, BR are also usually written in a textual form, which in turn is too informal and ambiguous to be directly processed my machines.

Then again, OMG has provided in 2008 the Semantic of Business Vocabulary and Rules (SBVR) standard [2] to capture specifications in a natural language (NL) and
represent them in formal logic understandable by machines. However, the standard requires a prior knowledge of its metamodel and at least one notation that can map to it, such as SBVR-Structured English [2] or Rule-Speak [3].

To overcome these problems, we propose in this paper a method to build a bridge from business to IT, by automatically extracting the business vocabulary as well as business rules from textual statements and present them according to SBVR standard, so we can start the automatic transformation of model at a very early phases of the development process. This paper should be of interest to all IT people who deal with natural language specifications; in particular, those who are interested in transforming textual business rules to formal language. This approach makes the following contributions:

1. Enhance methods dealing with Text to Model transformation.
2. Assist businesspeople to well structure their business vocabulary and describe business rules in a precise and unambiguous manner.
3. Bypass misunderstanding and simplify the communication between business experts and engineers.
4. Start the MDA approach at very early phases.

Generally, there are three main categories of BR statements [4]: i) statements expressing something must or should be true or not, ii) statements verifying condition before enabling other actions, iii) and statements generating new information (e.g. computations executing).

The difference between rules types requires a difference in sentence patterns of rules statements. Regarding the number of templates and patterns that can take a BR statement, we have been focused on statements of the first and second category; namely, statements expressing instantiations and characterizations of concepts, and statements having antecedents and consequence formulation, as they play an important role in improving the comprehensibility of the Terminological Dictionary that we are trying to generate in our approach. Other kinds of BR statements may be easily added by adding their related patterns (and lexicon if necessary), but this will be at the expense of the approach clarity, so these ignored statements will be considered in future work.

Our approach follows three steps to generate the result:

1) **Linguistic Analysis**: The Part-Of-Speech (POS) tagging, and the dependency tree is generated for each statement, before to be dissected to identify atomic clauses.

2) **Business Vocabulary generation**: An SBVR-based terminological dictionary is generated using patterns. The dictionary contains noun concepts, verb concepts, definitions, general concepts, necessary statements, and synonyms of each identified concept.

3) **Business Rules generation**: The semantic formulation used to express the meaning of each statement is identified; namely: atomic formulation, instantiation formulations, and logical formulations such as implications, negation, quantifiers, modality, and conjunctions/disjunctions.

To evaluate our method, experiments were performed on three different cases taken from three different sources [5]–[7]. Values obtained from manual process and automatic process were compared and revealed that our approach can be largely trusted to automatically transform BR from natural language to a more formal language. Our approach is a clear improvement on current methods dealing with textual BR because it generates a more comprehensive meaning of Business vocabulary and rules.

This paper is structured as follows: Section 2 briefly introduces the SBVR standard used in this paper as a target model. Section 3 provides an overview of existing approaches for transforming textual specifications to formal language. Section 4 describes in detail our approach to automatically generate the semantic of business vocabulary and rules from NL BR. Section 5 gives an illustrative example of our approach. Section 6 presents the experimental evaluation as well as obtained results in terms of identification of business vocabulary and semantic formulation of BR and discusses obtained result. Section 7 draws threats to validity of the approach. Finally, section 8 presents our future works and concludes the paper.

II. SEMANTIC OF BUSINESS VOCABULARY AND RULES

The Object Management Group (OMG) has published the Semantic of Business Vocabulary and Rules (SBVR) standard in 2008 [2]. It is a metamodel defined to document the semantic of business vocabulary, and business rules in a natural language and express them in formal logic so they can be understood by both of human and machines. SBVR does not provide a logic language for restating business rules in some other language that businesspeople do not use; rather, SBVR provides a means for describing the structure of the meaning of rules expressed in the natural language that businesspeople use. An XMI schema for the SBVR metamodel is also proposed in the official documentation for the interchange of business vocabularies and business rules between software tools and among organizations.

SBVR standard is created to simplify the communication between business experts and business IT by:

- Assisting the specification of the semantic of business domain.
- Structuring the business vocabulary.
- Describing the business domain in a clear, and unambiguous manner.

Generally, SBVR metamodel defies two main elements:
- **Business Vocabulary**: Terminology used by a community or an organization that contains both of noun concepts (e.g. ‘Client’) and Verb concepts (e.g. ‘Client has account’).
- **Business rules**: two fundamental types are defined: ‘Definitional rules’ (structural) and ‘Behavioral rules’ (operative)

As can be seen in the SBVR metamodel of Figure 1, SBVR separates the semantic formulation (expressions) from
concepts (Meanings). It makes use of logical formulations to represent the meaning of a rule in a set of logical structures, namely:

- Atomic formulation: it is based on exactly one verb concept having two roles ranging over general concepts. The meaning invoked by an atomic formulation puts each referent of each role-binding in its respective verb concept role. The following example ‘Each rental must authorize at most 3 additional drivers’ includes an atomic formulation that is based on the verb concept ‘rental authorize driver’, which includes two roles: the first one binds to the concept ‘rental’ as it is introduced
by the universal quantifier ‘each’, while the second role binds to a variable ranging over the concept ‘driver’ as it is introduced by existential quantification ‘at most 3’.

- **Instantiation formulation**: each logical formulation based on the verb concept that considers exactly one concept and binds to exactly one bindable target as an instance of the concept is called ‘Instantiation Formula’. The following example ‘It is necessary that no city branch is an agency’ is a statement that is formulated by an instantiation formulation based on the verb concept ‘city branch is agency’ that considers the general concept ‘city’ and binds to the bindable target ‘agency’.

- **Modal Logics**: two types are defined namely ‘Alethic’ and ‘Deontic’, to reflect the relationship (true/false) of statements with possible worlds or acceptable worlds. The following example ‘Each rental must authorize at most 3 additional drivers’ contains the modal auxiliary ‘must’ attached to the main verb ‘authorize’. Thus, the atomic formulation ‘rental authorize additional drivers’ is embedded in an obligation formulation as a type of modal formulation.

- **Quantification**: many types of quantifications are supported, and may introduce and constrain a variable namely: ‘Universal’ such as ‘a’ and ‘each’, or ‘Existential’ such as ‘Numeric range’, ‘At least /most n’ and ‘Exactly n’.

- **Binary Logical Operations**: compound statements having at least two clauses are formulated using semantic formulation that includes at least one ‘Binary Logical Operation’. Each one of the latter involves exactly two logical operands. For example: the statement ‘It is necessary that the local area is the organization unit that manages some branches and some service depots’ includes two logical operands of a Binary Logical formulation of type ‘Conjunction’: the atomic formulations ‘organization unit manages branches’ (logical operand 1) and ‘organization unit manages service depots’ (logical operand 2). The ‘Implication’ is another type of binary logical operation used to formulate complex statements having dependent clauses. In other words, it is a binary logical operation that operates on an antecedent and a consequent. For example: the statement ‘It is obligatory that a rental incurs a recovery charge if the rental has a recovery location’ includes a binary logical operation of type ‘Implication’ involving both an antecedent and a consequent: the atomic formulations ‘rental has recovery location’ (antecedent) and ‘rental incurs recovery charge’ (consequent).

SBVR standard defines additional specifications for each concept to create the ‘Terminological Dictionary’, so that a more comprehensive meaning is provided for the business domain. For example, the dictionary includes two forms of ‘Definitions’: i) ‘Extensional definition’ (e.g. ‘rental is an advance rental or walk-in rental’), and ii) ‘Intentional definition’ (e.g. ‘A renter is the driver who is responsible for some recognized rental’). Both two kind of definitions may be an ‘Owned Definition’ that is created by the speech community or ‘Adopted Definition’ that is taken from an external source.

However, the identification of some other specifications is out of the scope of this work as they involve a human intervention to be added such as ‘descriptions’, ‘examples’, and ‘note’. The automatic identification of specifications included in the Terminological Dictionary is one of the features that distinguish our approach from others dealing with text-to-SBVR transformation.

Finally, for each element in our target SBVR metamodel we have defined a set of patterns to be identified using algorithms.

### III. RELATED WORKS

Various approaches have been proposed to solve the issue of transforming Natural Language specifications to a more formal language, which are almost based on NLP techniques [8].

### A. NATURAL LANGUAGE TO (SEMI) FORMAL

This kind of transformation has been widely addressed. Ilieva and Ormandjieva [9] uses heuristics to find candidate classes and their relationships, by automatically extracting Subject, Predicate and Object from textual requirements, before mapping them to object-oriented analysis model. Arora et al. [10] generate the domain model from textual requirements. Harmain and Gaizauskas [11], Sagar and Abirami [12], Sharma et al. [13], and Karaa et al. [14] analysis textual specifications and generate a UML Class Model. A. Al-Hroob et al. [15] and Tiwari et al. [16] automatically extract use case elements from textual specifications. Friedrich et al. [17] propose an automatic approach to generate BPMN models from natural language text based on combination of existing NLP tools.

Other approaches add a corpus for training purpose such as Sadoun et al. [18] who model the domain knowledge through an ontology population process by making use of a training corpus to identify triples in textual specifications; and Kate et al. [19] who present a method for inducing transformation rules that map natural language sentences into a specific formal language by associating patterns with templates. The approach is based on learning the transformation rules from corpora of sentences paired with their formal-language equivalents. They assume that a deterministically parsable grammar for the target formal language is available.

One of the major drawbacks to using these methods is that the corpus for training is not always available, and businesspeople are not always familiar with (semi) formal models which limits their participation in the validation of generated models. For these reasons, we have targeted SBVR standard as an intermediate representation that enable us to write our specifications in a natural language comprehensible by both of humans and machines; thus, businesspeople can participate in the validation of the generated model before any transformation to other formal model.
TABLE 1. Comparative table of approaches generating SBVR from textual specifications.

| Approach                 | Year  | Input                  | Output                                      | transformation |
|--------------------------|-------|------------------------|---------------------------------------------|---------------|
| JL. M-Fernández [27]     | 2008  | free text descriptions | BR statements                               | Manual        |
| M. Selway et al. [21]    | 2015  | Textual specification  | BR statements                               | semi-automatic|
| S. Roychoudhury [22]     | 2017  | legal English text     | SBVR model                                  | semi-automatic|
| I. S. Bajwa [23]         | 2017  | Textual specification  | BR statements, then OCL constraints          | semi-automatic|
| T. Skersys et al. [24]   | 2018  | Use Case Diagram       | General concept, verb concepts, and BR statements | semi-automatic|
| P. Danenas et al. [25]   | 2020  | Textual business rules | Entities, facts, and BR statements          | Automatic     |
| Our approach             |       | Textual business rules | SBVR model: Terminological Dictionary: noun concepts and verb concepts with their specifications such as definitions, synonyms, and generalization relations. BR statements: the semantic formulation such as modal formulation, atomic formulation, instantiation formulation, negation, quantification, implication, and conjunctions. | Automatic     |

B. NATURAL LANGUAGE TO SBVR

Fernández et al. [20] propose an approach to identify candidate BR statements from free Spanish text descriptions; then business analyst decides about the validity of the business rule, before manually writing the valid SBVR expressions. Selway et al. [21] present a method to generate formal SBVR models from NL business specifications by learning the vocabulary from an SBVR SE style glossary, and then by parsing the rules using the acquired vocabulary. Roychoudhury et al. [22] generate SBVR model from natural language. Sentences and seed domain model previously created are used to learn pattern templates before extracting relations. As well, domain experts are involved in series of transformations, which makes it semi-automatic. In the other hand, the editor needs a prior knowledge of the Structured English notation to edit suggestions or to add additional information such as synonyms and definitions. Bajwa and Shahzada [23] presents a method to generate OCL statements from natural language specifications using SBVR standard as an intermediate step. A preexisting UML class model is intended to be consistent with the vocabulary used in the textual constraints. Skersys et al. [24] propose an hybrid approach to extract general concepts, verb concepts, and SBVR statements from formal use case, enhanced later by Danenas et al. [25] using an NLP-based approach instead of purely template-based rules. Chittimalli et al. [26] propose an approach to mine SBVR based business vocabularies and rules from domain-specific business documents. They try to automatically identify entities and relations between them using heuristics, before generating business rules. However, defined heuristics are somehow lexicon-based like ‘If the sentence contains an if’; while many other expressions may be used in textual expressions like ‘when’ or even using only punctuation marks. As well, more SBVR elements should be identified to express a more complete SBVR model, such as modality, quantifications, and logical formulations, as well as specifications about extracted vocabulary (e.g. synonyms).

Despite this interest, research tended to focus on identifying objects and their explicit relations. While implicit information hidden in complex statements are usually neglected. As well, further information should be identified to increase the understanding and usefulness of extracted vocabulary, in particular: definitions and synonyms [28]. However, no one as far as we know has generated a more complete SBVR model from unrestricted BR statements as our approach does.

In this paper, we target both Terminological Dictionary enriched with further specifications for each concept, and Business Rules with their logical formulations using an in-dept NLP. Our target model provides a more comprehensive meaning of each business rule statement and presents them in a MS-Word format understandable by humans coupled with their semantic saved in an XMI file understandable by machines, which is considered as a good alternative to fill the gaps between business analyst and business IT at the early phases of the development processes.

IV. OUR APPROACH

In this section, we present our approach step-by-step to extract the semantic of business rules as well as business vocabulary from textual business rules according to SBVR standard. We detail how we transform Natural Language BR to SBVR based formal logic. Figure 2 is an overview of our method pipeline.

First, we begin with a linguistic analysis of each statement to identify the part of speech (POS) of each word; then, clauses composing each statement are extracted and simplified to obtain their most atomic form. In the second step, explicit verb concepts are generated based on clauses previously extracted, as well as single word and multi word noun concepts; before adding other specifications, such as definitions, synonyms and preferred designation, necessity statements and generalization relation. Next, semantic formulations expressing the meaning of BR statements are identified, namely: atomic formulations, instantiation formulations, restriction formulations, modality, implications, quantifications, conjunctions, and negations. Finally,
identified semantic is saved in an SBVR based XMI file for portability and reusability purposes.

In this paper we are focused on statements that carry information or constrains on other concepts, because they play an important role in enriching our generated terminological dictionary. Other kinds of business rules are skipped in this paper such as those expressing business process (e.g. ‘A passenger may board a flight only after checking flight’) or calculation (e.g. ‘income limit is the EITC income limit plus $1,000’). The rest of this section details each of these steps.

**A. STEP 1: LINGUISTIC ANALYSIS**

1) **ANALYZING GRAMMAR**

After an examination of different structures of NL BR statements discussed in different books specialized in the field [5]–[7], we have created a generic metamodel of figure 3 for NL BR statements.

According to our generic metamodel, statements are expected to include a modality and to be composed of at least one explicit clause (simple statement with independent clause) that may be in relation with other clauses using prepositions such as conjunctions (compound statement with independent clauses) or an implication (complex statement with dependent clauses).

Clauses are expected to be composed of at least one nominal subject with his predicate, which in turn is composed of at most one verb and complements (direct and indirect). Verbs are either expressing the state of the subject (stative verb), or denoting an action done by the subject in relation to the object (transitive verb). In the other hand, nouns may be introduced by a cardinality and could be in a simple form (one word), or compound form (multi-word). As well, nouns may also be restricted either explicitly such in ‘the car that is red’, or implicitly such in ‘red car’ which means the same thing.

As can be seen, our generic metamodel includes the most important elements that should exist in a well-formed textual BR statement. Thus, by identifying components of this generic metamodel, we will have all elements needed to map to our target SBVR metamodel.

For this reason, we have used Stanford CoreNLP [29] to generate the Part-Of-Speech (POS) tags, and dependency tree of each statement (see an example in the next section), before exploiting result to generate an intermediate XML file that respect our generic metamodel of NL statements.

Elements composing this metamodel will be identified using patterns of table 2 to generate the meaning of the business vocabulary and rules.

2) **EXTRACTING CLAUSES FROM STATEMENTS**

Statements are composed of clauses containing both a subject (at least one) and a predicate. To extract a more comprehensive meaning of each statement, we identify not only explicit clauses but also implicit ones as follow:

a: **EXPLICIT CLAUSES**

The POS tags and dependency tree generated from each statement are parsed to look for patterns P1, P2, P3, P4 and P5 expressing explicit clauses.

Pattern number P5 expresses a statement in passive voice which will be converted to active voice before saving it.

b: **IMPLICIT CLAUSES**

To perform an in-depth linguistic analysis of each NL statement, we look for patterns P6, P7, P8, P9, P10, P11, P12, P13 and P14 which express implicit clauses hidden in multi word nouns.
### TABLE 2. Patterns to be identified and mapped to SBVR elements.

| Pattern expressing explicit clauses | Pattern expressing implicit clauses |
|-------------------------------------|-------------------------------------|
| **Pattern**                         | **Example**                         |
| P1 Subject + ‘to be’ + Adjective    | ![Example](image1)                  |
| P2 Subject + ‘to be’ + Noun         | ![Example](image2)                  |
| P3 Subject + Verb + Direct Object   | ![Example](image3)                  |
| P4 Passive Nominal Subject + ‘to be’ + Verb in Past Participle + no agent | ![Example](image4)                  |
| P5 Passive Nominal Subject + ‘to be’ + Verb in Past Participle + agent | ![Example](image5)                  |
| Pattern expressing implicit clauses |                                     |
| P6 Noun (NN) + Verb in past participle (VBN) as a modifier | ![Example](image6)                  |
| Implicit clauses: VBN = NN + “is” + NN & NN + “is” + VBN | ![Example](image7)                  |
| P7 Noun (NN) + Adjective (JJ) as a modifier | ![Example](image8)                  |
| Implicit clauses: JJ + NN + “is” + NN & NN + “is” + JJ | ![Example](image9)                  |
| P8 Noun (NN1) & Noun (NN2) in compound relation | ![Example](image10)                 |
| Implicit clauses: NN1 NN2 + “is” + NN2 | ![Example](image11)                 |
| P9 Noun (NN1) + ‘of’ + noun modifier (NN2) | ![Example](image12)                 |
| Implicit clause: NN1 + ‘is’ + for’ + NN2 & NN2 + has’ + NN1 | ![Example](image13)                 |
| P10 Noun (NN1) + Preposition (‘of’) + noun modifier (NN2) | ![Example](image14)                 |
| Implicit clause: NN1 + ‘is’ + preposition + NN2 | ![Example](image15)                 |
| P11 Passive Nominal Subject + ‘to be’ in present participle + Verb in past participle + no Agent | ![Example](image16)                 |
| Implicit clause: nsubjpass + “is” + VBN | ![Example](image17)                 |
| P12 Passive Nominal Subject + ‘to be’ in present participle + Verb in past participle + Agent | ![Example](image18)                 |
| Implicit clause: Agent + VBN + nsubjpass | ![Example](image19)                 |
| P13 Noun + adjectival clause (verb in past participial + “by” + noun modifier) | ![Example](image20)                 |
| Implicit clause: NN2 + VBN + NN1 | ![Example](image21)                 |
| P14 Noun + adjectival clause (verb in present participle + Direct Object) | ![Example](image22)                 |
| Implicit clause: NN1 + VBG + Dobj | ![Example](image23)                 |
| Pattern expressing definitions of concepts | ![Example](image24)                 |
| P15 Extensional definition: Subject + ‘to be’ + Predicate1 + “Or” + Predicate2 … | ‘Rental is advance rental or walk-in rental’ may be used as an extensional definition for the concept ‘Rental’ |
| P16 Intentional definition: Subject + ‘to be’ + ( NN [~ Adjectival clause] ) | ‘Driver is a person [who is authorized for rental]’ may be used as an intentional definition for the concept ‘Rental’. |
| Pattern reflecting general concepts | ![Example](image25)                 |
| P17 Subject + ‘to be’ + NN1 + “Or” + NN2 … | ‘Rental is open rental or closed rental’ |
| Extracted general concept: Subject is a General Concept of NN1 | ‘Rental’ is General concept of ‘open rental’ |
| Subject is a General Concept of NN2 | ‘Rental’ is General concept of ‘closed rental’ |
| P18 Subject + ‘to be’ + ( NN [~ Adjectival clause] ) | ![Example](image26)                 |
| Extracted General Concept: NN is a General Concept of ‘Subject’ | ‘Gold customer is a customer [that has 5 reservations]’. |
| ‘Customer’ is General concept of ‘Gold customer’ | ![Example](image27)                 |
| Pattern expressing SBVR logical formulations | ![Example](image28)                 |
| P19 Atomic formulation: NN1 (subject) + transitive verb+ NN2 (direct object) | ![Example](image29)                 |
| P20 Instantiation formulation: Noun Concept NN1 + ‘to be’ = Noun Concept NN2. | ![Example](image30)                 |
| P21 Restricting Formulation: NN + Relative clause modifier (acl:relcl). | ![Example](image31)                 |
TABLE 2. (Continued.) Patterns to be identified and mapped to SBVR elements.

| Pattern | Description |
|---------|-------------|
| P22 | Conjunction/Disjunction: Verb Concept ‘p’ + ‘and’ or ‘or’ + Verb Concept ‘q’ |
| P23 | Implication: Verb Concept ‘p’ + Verb Concept ‘q’ as adverbial clause |
| P24 | Negation: Verb having the adverb ‘not’ as a negation. |
| P25 | Negation: Adjective having the adverb ‘not’ as adverbial modifier |
| P26 | Quantification: NN + cardinal number as a numerical modifier |
| P27 | Modal Formulation: Verb having modal (MD) as an auxiliary |

FIGURE 4. Example of compound clauses simplification.

In the other hand, clauses having more than one subject or object are dissected into simpler one, to extract the most atomic information (see an example in figure 4).

In the next steps, we detail how we could use extracted patterns to generate both of business vocabulary and business rules that respect our target SBVR metamodel.

B. STEP 2: GENERATION OF SBVR BASED BUSINESS VOCABULARY

In this sub section, we show how noun concepts and verb concepts are identified before adding additional specifications for each one of them to generate a more comprehensive SBVR based Terminological Dictionary.

1) VERB CONCEPT IDENTIFICATION

SBVR standard has defined two forms of verb concepts (facts): ‘Sentential form’ which includes a verb such in ‘rental incurs recovery charge’ and ‘Noun form’ such in ‘return branch of the rental’ which reflects the fact that ‘rental has return branch’. Accordingly, explicit clauses having subject and predicate identified at the linguistic analysis step will be mapped to verb concept in Sentential Form, while implicit clauses will be mapped to verb concepts in Noun Form.

2) NOUN CONCEPTS IDENTIFICATION

First, all nouns found in the subject and object part of all identified clauses will be mapped to a single word noun concept. Then, nouns having modifiers will be adjusted to create multi-word noun concepts, namely nouns in compound relations with other nouns such in ‘credit card’, or nouns having noun modifier such in ‘company in operating country’, or nouns having adjectival modifier such in ‘bad driver’ (see the algorithm in figure 5).

3) TERMINOLOGICAL DICTIONARY GENERATION

Instead of generating a list of nouns and facts, SBVR based business vocabulary defines additional specifications for each concept to create a Terminological Dictionary making business vocabulary reflecting more comprehensive meaning. Some specifications should be added manually to clarify the concept such as description, note and examples. While other specifications could be understood from statements, namely: definitions of concepts, synonyms, generalization relations and necessity/possibility statements.

4) NOUN CONCEPT DEFINITIONS

As previously cited, SBVR metamodel defines two sources of concept’s definitions: ‘Owned definition’ and ‘Adopted
A. Haj et al.: Semantic of Business Vocabulary and BR: An Automatic Generation From Textual Statements

5) GENERALIZATION RELATIONS
Two patterns (P17 and P18) were defined to identify the general concept of a given noun concept.

6) NECESSITY AND POSSIBILITY STATEMENTS
For each extracted concept, SBVR standard defines some statements that are necessary (or possible) to trace limits in which the concept could be used. Thus, for each extracted concept, we affect statements in which it was found as a necessity or possible statement for the concept according the modality of the statement.

7) SYNONYMS & PREFERRED DESIGNATION
Given that business rules are usually written by different intervenors, it is highly probable that writers use different expressions having same meaning or use abbreviations instead of the whole multi words noun. To deal with this issue, there is no single agreed method that is valid for all contexts, but existing approaches generally tend to ignore this problem such as Chittimalli et al. [26], or manually mark terms having same meaning such as Selway et al. [21], Roychoudhury et al. [22], Bajwa and Shahzada [23], or trust only dictionary database to identify them Skersys et al. [24], Danenas et al. [25].

In our approach, we propose two algorithms: the first one automatically extracts noun concepts having similar dictionary-based sense, while the second one relates abbreviations with their expansions. Finally, the most frequent noun concept is flagged as ‘preferred designation’.

To relate abbreviations with their expansions, we have defined the algorithm in figure 6 based in the following heuristics:
- An abbreviation is not a known (dictionary) word (for this reason, they are generally tagged as proper nouns).
- Letters composing the abbreviation should match the first letter (or letters) of each word in its expansion.
- Potential abbreviation will be compared with multi words terms.

Regarding synonyms identification, we have compared eight different algorithms for synonyms identification WUP [30], RES [31], JCN [32], LIN [33], LCH [34],
LESK [35] HSO [36] and PATH (the shortest path connecting senses in hyponym and/or hypernym taxonomy), and we have found that LIN algorithm is the most accurate one. Then, we have used this algorithm with different distances between pairs of synonyms, and we have found that terms having at least 0.5 as a distance are more likely to be synonym. Finally, heuristics are used to filter result using algorithm of figure 7 summarized in the following two main points:
- Synonym nouns should have similar context, and then have at least one common neighbor noun concept.
- The noun concept should not appear with his synonym in the same statement.

C. STEP 3: GENERATION OF SBVR BASED BUSINESS RULES
To communicate the meaning of BR statements, SBVR uses structured facts about the “Semantic Formulations” instead of a formal language. SBVR defines two kinds of Semantic Formulation: ‘Logical Formulations’ that structure propositions (business rules), and ‘Projections’ that formulate definitions, aggregations, and questions. The latter is out of the scope of this paper, thus, in this sub-section we detail how logical formulation are identified, namely: atomic formulation, instantiation formulation, modal formulation, restricting formulation, conjunctions, quantifications, and negation.

1) ATOMIC FORMULATION
All verb concepts that are written according to the pattern P19 are mapped to an Atomic Formulation, with NN1 is its first role and the NN2 is its second role.

2) INSTANTIATION FORMULATION
All verb concepts that are written according to the pattern P20 are mapped to Instantiation Formulation, with NN1 is the considered concept and NN2 is its bindable target.

3) RESTRICTING FORMULATION
Logical formulations are recursive. Several kinds of logical formulations embed other logical formulations. For example, some logical formulation could be used as “restricting formulation” that defines concepts of other atomic formulation such in ‘site that is used for rental business’ which is a restriction of the noun concept ‘site’. For this reason, we try to extract relative clause and map it as a restricting formulation of nouns according to pattern P21.

4) CONJUNCTION/DISJUNCTION
Each two verb concepts ‘p’ and ‘q’ that are related with a conjunction / disjunction relation (pattern P22) are mapped to a Binary Logical Formulation having ‘p’ as ‘logical operand 1’ and ‘q’ as ‘logical operand 2’.

5) IMPLICATION
Each clause ‘p’ having an ‘Adverbial Clause’ (advcl) ‘q’ such in pattern P23 will be mapped to ‘Implication’, with ‘p’ as ‘antecedent’ and ‘q’ as ‘consequent’.

6) NEGATION
Extracted clauses could contain a negation. For this purpose, we have defined three patterns P24, P25 and P26 to identify them.

7) QUANTIFICATION
Quantifications are logical formulations introducing exactly one variable included in a logical formulation. Table 3 contains some quantifications supported by our approach.

For example, the statement ‘each bad experience has at least one notification date-time’ is formulated by an atomic formulation having two “role binding”. The first one binds to the concept ‘bad experience’ that is introduced by a universal quantification ‘each’, while the second binds to the variable ‘notification date-time’ that is introduced by existential quantification ‘at least one’.

In our approach, we are based on the Named Entity Recognition (NER) of numerical modifiers (nummod) related to

| Expression | Logical Formulation |
|------------|---------------------|
| A, each, all | Universal quantification |
| at least n | at-least-n quantification |
| no less than | at-most-n quantification |
| at most n | at-most-n quantification |
| no more than | less-than-n quantification |
| more than n | Exactly-n quantification |

| Table 3. Example of expression-to-quantification mapping. |
extracted noun concepts (if exists), which is generated by the parse tool. (see pattern P27)

8) MODAL FORMULATION
Modal auxiliaries attached to verbs are mapped to modal formulations such in pattern P28. Otherwise, modality could be also found as a clause at the beginning of the statement such in “It is obligatory that…” . Table 4 includes some expressions and their equivalence in SBVR.

D. STEP 4: SBVR BASED XMI: THE OUTPUT FILE
As a final step, the semantic formulation of each BR statement as well as the terminological dictionary containing extracted concepts coupled with their specifications are saved in an SBVR based XMI file that respect patterns proposed by the official documentation.

Our generated file could be used to automatically generate a structured textual description of business vocabulary and rules understandable by all users as well as machines. Furthermore, the XMI file will make our approach easily integrable in other approaches dealing with textual BR.

Table 5 shows the mapping between SBVR element and the SBVR-SE captions proposed in the official documentation to present the Terminological Dictionary.

A comprehensive example about how extracted elements are saved in the XMI file could be found in the illustrative example of the next section.

V. ILLUSTRATIVE EXAMPLE
Different SBVR elements are extracted by our approach using different patterns in each step. Thus, to provide a complete example, we need a much greater number of BR statements to cover all steps of our approach. For reasons of space, we will take just the following complex statement, and show how its semantic could be identified using our patterns:

Example 1: ‘The local area is the organization unit that manages exactly three branches and some service depots if the local area is an active area’.

The parse tree and POS tagging performed by the parser tool in step 1 is shown in figure 8.

By applying our approach on this example, we were able to identify 8 noun concepts, 10 verb concepts, 3 general concepts, 2 atomic formulations, 6 instantiation formulations, 1 restricting formulation, 1 conjunction, 1 implication formulation, and 1 quantification. The example does not contain neither modality nor negation formulation. As well, the statement contains a dependent clause which may not be used as a definition for any concept. Finally, to show how synonym terms are identified, we need at least six statements (three statements for each term), because our algorithm of synonyms extraction is sensitive to term frequency. Thus, synonyms are also not identified from the example. Details about identified SBVR element as well as patterns used for this purpose are shown in table 6.

The semantic of the statement is then saved in an SBVR based XMI file using patterns cited in the official documentation (Figure 9 is an excerpt). Based on these generated XMI file, we can easily generate an MS-word file for the Terminological Dictionary.

VI. EXPERIMENTAL EVALUATION
To evaluate our approach, experiment was carried out on three different sets of textual BR statements taken from different sources according to the following:

- Case-1 [7]: 53 statements about client preferences.
- Case-2 [6]: 53 statements about flight booking.
To avoid problems that may result from natural language ambiguity which may affect results, our BR statements are intended to be well-formed and ambiguity free. Thus, we have reformulated some statements that do not fit recommendations proposed by [6] such as statements that use pronouns or both of conjunction and disjunction.

To get our truth-table, we have first manually extracted the Business Vocabulary coupled with specifications as well as the semantic of business rules included in each case according to SBVR standard. Then, we have compared results with others automatically generated with our method. Finally, we calculated the number of True Positive (TP), False Positive (FP), False Negative (FN), Precision (1), Recall (2) and F1-Score (3) using the following equations:

\[
\text{Precision} = \frac{TP}{TP + FP} \tag{1}
\]

\[
\text{Recall} = \frac{TP}{2 \times TP + FN} \tag{2}
\]

\[
F1 - \text{score} = \frac{\text{Precision} + \text{Recall}}{\text{Precision} \times \text{Recall}} \tag{3}
\]

In the rest of this section, we will show - step by step - to what extent our approach could generate both of business vocabulary and rules according to SBVR standard.

A. LINGUISTIC ANALYSIS STEP

Each set of BR statements is saved in a separate file text, in which each statement is written in a separate line ended by a full stop punctuation mark. Then, each file was linguistically processed before generating the XML file containing POS tagging as well as the dependency parse tree of each statement. As a result, 96% of statements (all cases included) had been linguistically analyzed correctly by the Stanford CoreNLP tool, divided as follow: 96% in case-1, 93% in case-2, and 98% in case-3. However, 4% of erroneous statements is considered as very interesting. The source of errors at this first step was either a misplaced punctuation mark, or some words that are usually ambiguous such as “requests” and “references” which could be used both as a verb and as a noun, or some statements that are formulated using a semantic formulation that is not supported by our approach (e.g. whether-or-not formulation).

The second part of this linguistic analysis step concerns the dissection of clauses that compose statements to extract implicit ones. As previously cited, we try to dissect clauses having more than one subject noun or object noun related by conjunctions. Although all defined patterns were correctly identified and dissected, some clauses containing more than one verb with same subject and object were not identified. For example, the following statement “A passenger may check-in and board a flight” has two verbs in conjunction relation that...
TABLE 6. SBVR element identified from the statement of example 1.

| SBVR  | Id  | Output                                                                 |
|-------|-----|-------------------------------------------------------------------------|
| gc-1  |     | Using Algorithm of figure 5: (NN + modifier)                            |
| gc-2  |     | “Local area”                                                            |
| gc-3  |     | “Active area”                                                           |

General concept

| gc-4  |     | Using Algorithm of figure 5: (Compound NN)                              |
| gc-5  |     | “Organization unit”                                                     |
| gc-6  |     | “Service depot”                                                         |

| gc-7  |     | Using Algorithm of figure 5: NN (Single word)                           |
| gc-8  |     | “branch”                                                                |
| nec-x |     | The BR statement will be added as a necessity statement for each identified noun concepts. |

TABLE 6. (Continued.) SBVR element identified from the statement of example 1.

| Implication | id  | Output                                                                 |
|-------------|-----|-------------------------------------------------------------------------|
| Implication | imp-1 | Using pattern 23: The implication ‘imp-1’ has: antecedent: ‘if-1’ consequence: ‘if-2’ |

Quantification

| qtf-1 | Using pattern 27: The quantification “exactly three” introduces a variable ranging over the concept ‘gc-5’ The quantification scopes over an “exactly-n” quantification. exactly-n quantification has cardinality ‘3’ |

Definition | No |
Modality | No |
Synonym | No |
Negation | No |

have the same subject and object. At the linguistic analysis, we got two clauses: the first one has only a subject + verb “passenger check-in” and a second clause has a verb+object “board flight”. The source of error at this level is that the statement is not as simple as our approach requires. Although this is a kind of issues that was found less than three times, and a pattern taking into consideration this form of compound clauses must be added, or clauses must be more simple by adding explicit subject and object for each verb regardless the words redundancy.

The output result at this step is saved in an XML file that will be used in the next step to generate both of business vocabulary and business rules.

B. EXTRACTION OF BUSINESS VOCABULARY AND RULES

Based on statements linguistically analyzed in the first step, we extract all necessary elements needed to generate the semantic of business vocabulary and rules.

Table 7 summarizes obtained result related to our automatic identification of SBVR elements from natural language BR statements, namely: noun concepts, verb concepts, generalization relations, implications, quantifications, and synonyms. Other elements are not discussed in the table as they are obtained by a direct mapping from identified elements, namely atomic formulations and instantiation formulations which are mapped from verb concepts, and binary logical formulations which in turn are obtained from dissected clauses.

1) BUSINESS VOCABULARY EXTRACTION

In this sub-section, elements needed for the SBVR-based Terminological Dictionary are extracted, containing business vocabulary coupled with further specifications.

a: NOUN CONCEPTS

General concepts were identified with an F1-score between 89% and 96%. General concepts composed of a single word
TABLE 7. Obtained result related to the automatic generation of the business rules semantic.

|                   | Cases | Manually Extracted | Automatically Identified | TP | FP | FN | Precision | Recall | F1 Score |
|-------------------|-------|--------------------|-------------------------|----|----|----|-----------|--------|----------|
| **General Concept** |       |                    |                         |    |    |    |           |        |          |
| case1             | 273   | 271                | 261                     | 10 | 12 | 96% | 96%       | 96%    |          |
| case2             | 212   | 209                | 201                     | 8  | 11 | 96% | 95%       | 95%    |          |
| case3             | 138   | 135                | 121                     | 14 | 17 | 90% | 88%       | 89%    |          |
| **Verb Concept**  |       |                    |                         |    |    |    |           |        |          |
| case1             | 282   | 276                | 274                     | 2  | 8  | 99% | 97%       | 98%    |          |
| case2             | 181   | 178                | 175                     | 3  | 6  | 98% | 97%       | 97%    |          |
| case3             | 96    | 91                 | 87                      | 4  | 9  | 96% | 91%       | 93%    |          |
| **Generalization**|       |                    |                         |    |    |    |           |        |          |
| case1             | 19    | 17                 | 15                      | 2  | 4  | 88% | 79%       | 83%    |          |
| case2             | 33    | 30                 | 24                      | 6  | 9  | 80% | 73%       | 76%    |          |
| case3             | 27    | 27                 | 22                      | 5  | 5  | 81% | 81%       | 81%    |          |
| **Implication**   |       |                    |                         |    |    |    |           |        |          |
| case1             | 53    | 49                 | 49                      | 0  | 4  | 100%| 92%       | 96%    |          |
| case2             | 7     | 6                  | 6                       | 0  | 1  | 100%| 86%       | 92%    |          |
| case3             | 20    | 18                 | 18                      | 0  | 2  | 100%| 90%       | 95%    |          |
| **Quantification**|       |                    |                         |    |    |    |           |        |          |
| case1             | 12    | 11                 | 9                       | 2  | 3  | 82% | 75%       | 78%    |          |
| case2             | 13    | 11                 | 8                       | 3  | 5  | 73% | 62%       | 67%    |          |
| case3             | 14    | 13                 | 12                      | 1  | 2  | 92% | 86%       | 89%    |          |
| **Synonym**       |       |                    |                         |    |    |    |           |        |          |
| case1             | 22    | 21                 | 16                      | 5  | 6  | 76% | 73%       | 74%    |          |
| case2             | 20    | 20                 | 17                      | 3  | 3  | 85% | 85%       | 85%    |          |
| case3             | 15    | 16                 | 14                      | 2  | 1  | 88% | 93%       | 90%    |          |

were less efficiently identified compared to multi words ones, as they are more exposed to ambiguity such as the term “Gold” that was tagged as a noun in some statements and as an adjective in others.

Individual concepts are not discussed because we have not defined any strategy to identify them, except what Stanford tools has detected.

b: VERB CONCEPTS

Verb concept were identified with an F1-score between 93% and 98%. The main problem at this level is that some verb tenses are not supported by our approach such in “An applicant must be having adequate funding” which should be “An applicant must have adequate funding” to be identifiable by our approach. Further, some statements have used linking verbs that are not included in our generic metamodel such as the verb “become” in the statement “A shipment must be assigned to a packer, when it becomes fillable”.

Another key element to mention is that 52% of all identified verb concepts were extracted from compound clauses and multi word noun concept (54% in case-1, 65%, in case-2, and 24% in case-3). This reflects the amounts of information that are missed in other approaches not dealing with implicit clauses.

c: DEFINITIONS

In case 1, all statements are expressed using implication formulation. Thus, no one of them respect our patterns for noun concepts. For this reason, we were not able to affect any definition to any noun concept in this case. While 11 noun concepts were given a definition in case-2, against 8 in case 3. In fact, finding definitions for noun concepts is based on how much definitional BR statements is defined instead of Behavioral ones.

d: GENERALIZATION RELATIONS

Generalization/specialization relations were identified with an F1-score between 81% and 83%. Some identified relations were wrong regarding that some statements contain more than one expression for the same concept, such in “client preference” and “preference” which generated the fact that the second is the general concept of the first. In the other hand, some wrong multi word nouns have generated wrong generalization relation between noun concepts, such as “same flight” which were detected as a specialization of the concept ‘flight’.

e: SYNONYMS

BR statements are usually written by different intervenors. Thus, to imitate this context, we have taken about 30% of statements from each case and sent them to two different engineers to reformulate some of BR statements by injecting synonyms and abbreviations.

As previously cited, most approaches dealing with textual specifications trusted only dictionary databases to identify synonyms. While in our approach, we use LIN algorithm to query the WordNet database and identify potential synonym terms, before refining result using heuristics. Figure 10 shows the positive impacts of our heuristics in each case.

Our approach was able to extract synonyms with an F1 score between 74% and 90%. The main issue at this level is that our approach is intended to detect only synonym nouns composed of same number of words. In the other hand, nouns having different dictionary base sense are out of the scope of our approach such in “account balance” and “current balance”.

Another key element that has affected our result is that our heuristics are sensitive to term frequency. More terms are frequent, more they have change to detect their synonyms,
because they reflect their real context. Figure 11 shows that case 3 uses more frequent terms compared with other cases. For this reason, we could detect 90% of injected synonyms because terms were used many times in different statements, unlike the case 1 and case 2.

2) EXTRACTION OF LOGICAL FORMULATION COMPOSING BUSINESS RULES

a: MODALITY

All modality formulations were perfectly extracted, except the following statement “if the flight booking confirmation is international, then it is obligatory that the flight booking confirmation have exactly one set of passport details for each passenger of the flight booking confirmation.”, which includes the clausal modality “it is obligatory that” at the middle of the statement instead of the beginning as our approach requires. Thus, this form of modality formulation should be also taken into consideration in future works. At last, 40 statements containing modality formulation were found in case-3, and 39 statements in case-2, while no modality expression was included in case-1.

Extracted statements at this step, have been attached as necessity/possibility captions to 18 noun concepts in case-2, and 26 noun concepts in case-3.

b: IMPLICATION

Our method was able to identify implication formulations with 100% of precision, and 95% of F1-score. The main issue at this level was a wrong dependency parsing of complex statements having many clauses in conjunction relation. Some other statements have used some special expressions that were not detected as adverbial clause such as ‘p on condition that q’.

c: QUANTIFICATION

All quantification formulations defined in our method were correctly identified. However, there was some expressions that should be also taken into consideration such as “exceeding” and “x to y”. as a result, quantifications were identified with an F1 score between 67% and 89%.

d: NEGATION

Our patterns defined to detect negation formulations were sufficient to perfectly identify all the 57 situations of negation found in all statements with an F1-score of 100% (24 in case-1, 13 in case-2 and 16 in case-3). This is because patterns used in negation formulations are practically exhaustive as they are often based only on the term “NOT” related to the main verb. Given the above, negation formulation was not included in the table of results.

As a last note, even though there were some patterns and expressions that are not covered by our approach, we were able to obtain encouraging results. A few enhancements at this level will without a doubt produce more accurate results.

VII. THREATS TO VALIDITY

Despite our optimistic results, the following threats to validity need to be considered:

- Threats to Internal Validity: examines the impact of internal factors on the results.
- Threats to External Validity: limits the generalization of the results beyond the experiment setting.
- Threats to Conclusion Validity: examines factors leading to incorrect conclusion.

A. THREATS TO INTERNAL VALIDITY

Our approach is designed to analyze BR statements that are well-formed and ambiguity free. For this reason, textual BR that are not written respecting good practices have less chance to be correctly processed. This threat was minimal in our experiments, because some BR statements have been reformulated by authors of this paper to remove ambiguity before to be injected to our approach pipeline.

In the other hand, our approach is intended to deal with (informal) natural language statements which makes it hard to guess their grammatical structure. This threat was also minimal in our experiments, because the most used statement structures found in different books [5]–[7] specialized in the field are injected in our samples, except those previously cited as not supported by the approach.
B. THREATS TO EXTERNAL VALIDITY

In our approach, each step is based on result generated by its previous step, except the first step which is totally based on the result generated by the parser tool. Fortunately, we have adopted Stanford CoreNLP tool as one of modern, regularly updated with high quality text analytics. This is clearly shown in the fact that only 4% of statements were incorrectly analyzed.

WordNet is another tool used in our approach to identify synonyms. It is considered as the most popular database at this filed. However, generated results have been refined using heuristics.

C. THREATS TO CONCLUSION VALIDITY

The most concern of our experiments is our samples size which appears to be not sufficient to draw accurate conclusion. Our samples contain three sets of BR statements taken form three different sources and containing about 50 statements in each one of them. We tried to randomly select statements that represent all kind of sentence structures (simple, compound, and complex); thus, we believe that any increase in statements number will not affect the result too much.

VIII. CONCLUSION

We have presented an approach to automatically transform textual business rules to an SBVR based business vocabulary and rules, in which an in dept natural language processing have been used to identify explicit and implicit information from BR statements. Comparing to other approaches dealing with Text-to-SBVR, our approach is more complete as we extract not only noun concepts and verb concepts with some logical formulations, but also a more comprehensive Terminological Dictionary that contains further specifications about extracted concepts such as definitions and synonyms. As well, our approach does not use any assistant or pre-existing information, which makes it fully automatic. Our focus was more on BR providing relevant information or pre-existing information, which makes it fully automatic.

As a conclusion, the findings of our work support the following ideas:

- Providing a more comprehensive meaning of textual business rules statements may remove the gap between business experts and business IT and speed up the software development process, especially in its early phases.
- Trusting only textual business rules statements to automatically generate their real semantic without any human intervention is possible.
- Respecting best practices in writing textual business rules statements is the key to a successful Text-to-Model transformations.
- Generating the SBVR model of a textual specifications may not be limited to only concepts and entity-relation elements but may be also extended to cover other specifications included in the SBVR metamodel such as definitions of concepts, synonyms, and semantic formulations.

REFERENCES

[1] OMG | Object Management Group. Accessed: Dec. 29, 2019. [Online]. Available: https://www.omg.org/
[2] (Dec. 2019). Semantics of Business Vocabulary and Rules, Version 1.5. Accesee: Dec. 29, 2019. [Online]. Available: https://www.omg.org/spec/SBVR/About-SBVR/
[3] RuleSpeak || Let the Business People Speak Rules! Accessed: Dec. 2, 2020. [Online]. Available: http://www.rulespeak.com/ets/
[4] B. Van Halle, Business Rules Applied: Business Better Systems Using the Business Rules Approach. New York, NY, USA: Wiley, 2002.
[5] A. J. Morgan, Business Rules and Information Systems: Aligning IT With Business Goals. Boston, MA, USA: Addison-Wesley, 2002.
[6] G. C. Witt, Writing Effective Business Rules a Practical Method. San Mateo, CA, USA: Morgan Kaufmann, Jan. 2012.
[7] I. Graham, Business Rules Management and Service Oriented Architecture: A Pattern Language. Hoboken, NJ, USA: Wiley, 2006.
[8] T. Yue, L. C. Briand, and Y. Labiche, “A systematic review of transformation approaches between user requirements and analysis models,” Require-
ments Eng., vol. 16, no. 2, pp. 75–99, Jun. 2011, doi: 10.1007/s00766-010-0111-y.
[9] M. G. Ilieva and O. Ormandjieva, “Automatic transition of natural lan-
guage software requirements specification into formal presentation,” in Proc. Int. Conf. Appl. Natural Lang. Inf. Syst., 2005, pp. 392–397.
[10] C. Aurora, M. Sabotzadeh, L. Briand, and F. Zimmer, “Extracting domain models from natural-language requirements: Approach and indus-
trial evaluation,” in Proc. ACM/IEEE 19th Int. Conf. Model Driven Eng. Lang. Syst., Saint-Malo, France, Oct. 2016, pp. 250–260, doi: 10.1145/2976767.2976769.
[11] H. M. Harmain and R. Gaizauskas, “CM-builder: A natural language-
based case tool for object-oriented analysis,” Automated Softw. Eng., vol. 10, no. 2, pp. 157–181, 2003.
[12] V. B. R. V. Sagar and S. Abirami, “Conceptual modeling of natural
language functional requirements,” J. Syst. Softw., vol. 88, pp. 25–41, Feb. 2014.
[13] R. Sharma, P. K. Srivastava, and K. K. Biswas, “From natural language requirements to UML class diagrams,” in Proc. IEEE 2nd Int. Workshop Artif. Intell. Requirements Eng. (AIRE), Ottawa, ON, Canada, Aug. 2015, pp. 1–8, doi: 10.1109/AIRE.2015.7373625.
[14] W. Ben Abdessalem Karaa, Z. Ben Azzouz, A. Singh, N. Dey, A. S. Ashour, and H. Ben Ghazala, “Automatic builder of class diagram (ABCD): An application of UML generation from functional requirements,” Softw., Pract. Exper., vol. 46, no. 11, pp. 1443–1458, Nov. 2016.
[15] A. Al-Hroob, A. T. Imam, and R. Al-Heisa, “The use of artificial neural networks for extracting actions and actors from requirements document,” Inf. Softw. Technol., vol. 101, pp. 1–15, Sep. 2018.
[16] S. Tiwari, D. Ameta, and A. Banerjee, “An approach to identify use case scenarios from textual requirements specification,” in Proc. 12th Innov.
Softw. Eng. Conf. (Formerly Known India Softw. Eng. Conf.), Feb. 2019, pp. 1–11.
[17] F. Friedrich, J. Mendling, and F. Puhlmann, “Process model generation from natural language text,” in Proc. Int. Conf. Adv. Inf. Syst. Eng., 2011, pp. 482–496.
[18] D. Sadoun, C. Dubois, Y. Ghalmi-Doudane, and B. Grau, “From natural language requirements to formal specification using an ontology,” in Proc. IEEE 25th Int. Conf. Tools Artif. Intell., Nov. 2013, pp. 755–760.

[19] R. J. Kate, Y. W. Wong, and R. J. Mooney, “Learning to transform natural to formal languages,” in Proc. AAAI, 2005, pp. 1062–1068.

[20] J. L. Martínez-Fernández, J. C. González, J. Villena, and P. Martínez, “A preliminary approach to the automatic extraction of business rules from unrestricted text in the banking industry,” in Proc. Int. Conf. Appl. Natural Lang. Inf. Syst., 2008, pp. 299–310.

[21] M. Selway, G. Grossmann, W. Mayer, and M. Stumptner, “Formalising natural language specifications using a cognitive linguistic/configuration based approach.” In Inf. Syst., vol. 54, pp. 191–208, Dec. 2015, doi: 10.1016/j.is.2015.04.003.

[22] S. Roychoudhury, S. Sunkle, D. Kholkar, and V. Kulkarni, “From natural language to SBVR model authoring using structured English for compliance checking,” in Proc. IEEE 21st Int. Enterprise Distrib. Object Comput. Conf. (EDOC), Oct. 2017, pp. 73–78, doi: 10.1109/EDOC.2017.19.

[23] I. S. Bajwa and M. A. Shahzada, “Automated generation of OCL constraints: NL based approach vs pattern based approach,” Mehran Univ. Res. J. Eng. Technol., vol. 37, no. 2, pp. 243–254, Apr. 2017, doi: 10.22581/muet1982.1702.04.

[24] T. Skersys, P. Danenas, and R. Butleris, “Extracting SBVR business vocabularies and business rules from UML use case diagrams,” J. Syst. Softw., vol. 141, pp. 111–130, Jul. 2018, doi: 10.1016/j.jss.2018.03.061.

[25] P. Danenas, T. Skersys, and R. Butleris, “Natural language processing-enhanced extraction of SBVR business vocabularies and business rules from UML use case diagrams,” Data Knowl. Eng., vol. 128, Jul. 2020, Art. no. 101822, doi: 10.1016/j.datak.2020.101822.

[26] P. C. Chittimalli, C. Prakash, R. Naik, and A. Bhattacharyya, “An approach to mine SBVR vocabularies and rules from business documents,” in Proc. 13th Ind. Innov. Softw. Eng. Conf. Formerly Known India Softw. Eng. Conf., Jabalpur, India, Feb. 2020, pp. 1–11, doi: 10.1145/3385032.3385046.

[27] J. L. Martínez-Fernández, J. C. González, J. Villena, and P. Martínez, “A preliminary approach to the automatic extraction of business rules from unrestricted text in the banking industry,” in Natural Language and Information Systems, Berlin, Germany: Springer, 2008, pp. 299–310, doi: 10.1007/978-3-540-69858-6_29.

[28] B. Hnatkowska and A. Walkowiak-Gall, “Towards definition of a unified domain meta-model,” in Proc. KKIO Softw. Eng. Conf., 2018, pp. 86–100.

[29] C. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. Bethard, and D. McClosky, “The Stanford CoreNLP natural language processing toolkit,” in Proc. 52nd Ann. Meeting Assoc. Comput. Linguistics, Syst. Demonstrations, 2014, pp. 55–60.

[30] Z. Wu and M. Palmer, “Verbs semantics and lexical selection,” in Proc. 32nd Ann. Meeting Assoc. Comput. Linguistics, 1994, pp. 133–138.

[31] P. Resnik, “Using information content to evaluate semantic similarity in a taxonomy,” 1995, arXiv:cmp-lg/9511007. [Online]. Available: https://arxiv.org/abs/cmp-lg/9511007.

[32] J. J. Jiang and D. W. Conrath, “Semantic similarity based on corpus statistics and lexical taxonomy,” 1997, arXiv:cmp-lg/9709008. [Online]. Available: https://arxiv.org/abs/cmp-lg/9709008.

[33] D. Lin, “Extracting collocations from text corpora,” in Proc. 1st Workshop Comput. Terminol., 1998, pp. 57–63.

[34] C. Leacock and M. Chodorow, Combining Local Context and WordNet Sense Similarity for Word Sense Identification. WordNet, An Electronic Lexical Database. Cambridge, MA, USA: MIT Press, 1998.

[35] S. Banerjee and T. Pedersen, “An adapted Lesk algorithm for word sense disambiguation using WordNet,” in Proc. Int. Conf. Intell. Text Process. Comput. Linguistics, 2002, pp. 136–145.

[36] C. Fellbaum and G. Miller, “Lexical chains as representations of context for the detection and correction of malapropisms,” in WordNet: An Electronic Lexical Database. Cambridge, MA, USA: MIT Press, 1998, pp. 305–332.

ABDELLATIF HAJ received the master’s degree in computer sciences from the Faculty of Sciences and Technologies, Hassan I University, Settat, Morocco, in 2013. He is currently pursuing the Ph.D. degree with the Laboratory of Mathematics, Computer and Engineering Sciences, Department of Mathematics, Hassan I University. His research interests include software engineering, model-driven development, data mining, natural language processing, and artificial intelligence.

ABDESSAM JARRAR received the Ph.D. degree in computer science from the Faculty of Science and Technologies, Hassan I University, Settat, Morocco, in 2020. He has written many scientific articles in the domain of theory of computation, programming languages, and software engineering. His research interests include software development, theoretical computer science, formal methods, model checking, and logical reasoning.

YOUSSEF BALOUKI is currently a Professor of computer science with the Faculty of Science and Technologies, Hassan I University, Settat, Morocco. He has conducted many Ph.D. theses and has written a fifty of scientific articles in the domain of artificial intelligence, software engineering, model-driven development, data mining, and formal methods.

TAOUFIQ GADIR has been the Director of the Informatics, Imaging, and Modeling of Complex Systems Laboratory, since 2014. He is currently a Professor of computer science with the Faculty of Science and Technologies, Hassan I University, Settat, Morocco. He has conducted more than twenty Ph.D. theses and has written a seventy of scientific articles in the domain of model driving architecture, data mining and database analysis, 3D models analysis, modeling of complex systems, and machine learning.

***