Text2Dec: Extracting Decision Dependencies from Natural Language Text for Automated DMN Decision Modelling

Vedavyas Etikala[0000−0002−5184−3812],
Ziboud Van Veldhoven[0000−0001−6013−7437], and
Jan Vanthienen[0000−0002−3867−7055]

Leuven Institute for Research on Information Systems (LIRIS), KU Leuven
vedavyas.etikala@kuleuven.be

Abstract. Decisions are of significant value to organisations. Business decisions are often written down in textual documents, and modelling them is a tedious and time-consuming task. Although decision modelling has seen a surge of interest since the introduction of the Decision Model and Notation (DMN) standard, limited research has been conducted regarding automatically extracting decision models from the text. In this paper, we propose a text mining technique to automatically extract the decisions and their dependencies from natural language text to build the decision requirements diagram. A case-based evaluation is shown for the proposed mining approach with promising results. This approach can serve as a groundwork for further research in the field of decision automation.

Keywords: Decision Model and Notation (DMN), Business Decision Management, Natural Language Processing (NLP).

1 Introduction

Efficient decision modelling adds significant value to organisations in managing their recurrent yet essential business decisions such as granting a loan, determining credit card eligibility, or diagnosing a patient. Representing business knowledge as decision models not only increases the interpretability of otherwise complex decision processes but also paves the road towards automation of business decision management (BDM).

BDM, which concerns the entire process of modelling, managing, and enacting decisions present in the organisation, has gained increased interest in recent years. Since the introduction of Decision Model and Notation (DMN) as a decision modelling standard by the Object Management Group (OMG) in 2015, modelling decision knowledge at a higher level of abstraction has been made possible and reliable [11, 22]. Successful decision modelling requires understanding the business knowledge and learning the modelling technique [24]. Both of these tasks are time-consuming. Despite DMNs recent popularity [5, 8, 11, 17, 18, 23], little research has been conducted to improve these areas. Furthermore, most

of the business decision knowledge is still stored and shared in textual documents [13, 20].

Hence, we propose a novel Text2Dec framework that uses state-of-the-art text mining and natural language processing (NLP) to extract decision dependencies as Decision Requirements Diagrams (DRD) from text-based documents. As far as we know, this is the first attempt to address the transformation of text into DMN decision requirements. We follow a three-step methodology inspired by similar works in other modelling domains [1, 10, 21]: (i) understanding the textual descriptions of decisions, (ii) coding an ensemble of tailored NLP techniques for detecting patterns and identifying the concepts of decision models, and (iii) generating the DRD. The method is evaluated with a real-world case of "Employee Health Assessment".

This paper is structured as follows. Section 2 introduces related work on decision modelling and DMN. Next, we explain the scope of the proposed approach in section 3. Section 4 presents the methodology of the Text2Dec framework for automated decision dependency extraction. In section 5, we present a case study to evaluate the proposed approach. Section 6 contains the discussion, challenges, and future work. Finally, section 7 concludes the research.

2 Motivation and Related Work

2.1 Decision Modelling and DMN

Decision models, represented in DMN, are knowledge structures that capture not only business decisions but also their dependencies and logic. The models are intended to be both executable and understandable by all stakeholders. A DMN decision model consists of two levels. First, the decision requirements level in the form of a DRD (or graph) is used to model the requirements of the decisions and the dependencies between the different constructs in the decision model. Secondly, the decision logic level is used to specify the detailed logic for each decision, usually in the form of decision tables. The DRD consists of a small set of constructs: rectangles to depict decisions, corner-cut rectangles for business knowledge models, and ovals to represent data input as shown in Fig. 1. Requirements are depicted with arrows, e.g. an information requirement indicates that a decision requires the information from input data or the result of another decision.

Definition 1. An *entity* is the business object paying a pivot role in the given decision scenario. e.g: person, weather or discount.

Definition 2. A *concept* is an attribute of an entity, e.g: loyalty of a customer, height of a patient, loan qualification (an information item in DMN).

Definition 3. A *dependency* is either an information requirement or an authority requirement or a knowledge requirement in DMN, which is represented as a link between components of the DRD. A dependency link is mathematically represented as a tuple $dep = (\mathcal{A}, \mathcal{B}, \mathcal{D})$, where $\mathcal{A}$ is an action verb, $\mathcal{B}$ is a base concept, and $\mathcal{D}$ is a derived concept.

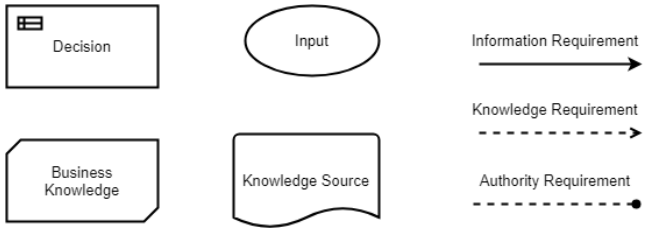

Fig. 1: Components and requirements of DMN

Definition 4. A *base concept* is an input information item in DMN. It is not the result of a decision.

Definition 5. A *derived concept* is a concept derived from one or more other concepts through a decision. The derived concept is dependent on these other concepts. E.g, *"Body Mass Index value"* is the derived concept from the base concepts *"height"* and *"weight"* of the patient.

Definition 6. A *DMN DRD* is a tuple in the form of DRD $= (\mathcal{D}, \mathcal{I}, \mathcal{R})$ where $\mathcal{D}$ is a set of decisions, $\mathcal{I}$ is a set of input information items identified as concepts and $\mathcal{R}$ is set of requirements identified as dependencies.

2.2 DMN Modelling

Decision models are usually constructed manually by domain modellers or business analysts based on the provided documentation and/or the knowledge acquired from the interaction with domain experts. Several modelling guidelines have already been proposed [2, 16, 24] to obtain complete, consistent and implementable decision models. However, the process remains a manual effort, even with tool support, and is therefore difficult and time-consuming.

Recently, some knowledge discovery approaches have been proposed for extracting parts of the decision model from structured sources, such as process event logs, historical data or process model flows:

- Decision rules can be discovered from historical cases as part of the large area of business analytics. In the context of process discovery, mining the decision logic at decision points in a process is called decision mining.
- The structure of the decisions can be derived too [4, 6].
- When a business process model is available, a decision model can be extracted from the process by identifying the decision points and data dependencies in the process flow. The result is a decision model (including the DRD) and a more simple process model [2, 3].
- Methodologies to mine decision models together with process models from extensive decision-process logs are also proposed, producing separate but integrated decision and process models [4, 9].

Despite these works on automating modelling from structured sources, limited research has been conducted on extracting decision models from textual sources. Decisions, however, are usually described in text. Hence, (semi)-automatic

extraction of decision models from text would be a highly beneficial endeavor. In related business modelling domains, extraction from text is not a new field:

- A number of techniques and tools have been suggested to extract process models from text [1, 12, 19].
- Rule extraction from legal and business texts has been researched in e.g. [7, 10]. Extracting the attributes and their values for correct logical statements is the point of focus in this domain.

Existing model extraction approaches detect sentence patterns to identify tasks and control structures such as "approve the claim" or "mail the client" [15]. The decision logic and requirements extractors, however, need to identify the relevant information that exists both explicitly in declarative statements and also implicitly in conditional statements. Conceptually, decision information is not just limited to a single sentence. Therefore, adequate paragraph-level mining is needed to analyse semantic and syntactic clues for language.

3 Scope of the Proposed Approach

Because deriving the entire detailed decision logic from a complex text would be an immense task and would produce a long list of logical rules without an appropriate structure, we follow the DMN guidelines and separate the decision requirements level from the decision logic level. Decision logic, because it has to deal with specific values for input and outcome information items, has to be much more precise than dependencies between concepts. Once the decisions and their requirements are identified, it will be easier to isolate and construct the decision logic in a (semi)-automatic way. The exact value of the outcome of a decision is not the first concern but rather the observation that a decision depends on, e.g. one specific input information item is the result of another decision. In this stage of the research, only information requirements are considered. Consider the following running example to understand the practical benefit of being able to automatically structure textual knowledge as a DRD:

Example: *"The health risk level of a patient should be assessed from the obesity level, waist circumference and the sex of the patient. Furthermore, the degree of obesity should be determined from the BMI value and sex of the patient. Patient's height and weight are considered to calculate his BMI value. If the weight of the patient given in kgs and height of patient given in meters, then the BMI value is weight/(height*height)."*[1]

The concepts need to identified, extracted, and categorised into base concepts (height, weight, sex, waist circumference) and derived concepts (BMI value, obesity level, health risk level) based on the semantic role using syntactic patterns of nouns around the action verbs (assess, consider, determine, is). Each pair of base and derived concept can then be represented as dependency tuples (action, base concept, derived concept). Finally, using the domain-independent heuristics, the tuples can be converted into a DRD as shown in Fig. 2.

[1] taken from `https://www.nhlbi.nih.gov/files/docs/guidelines/prctgd_c.pdf`

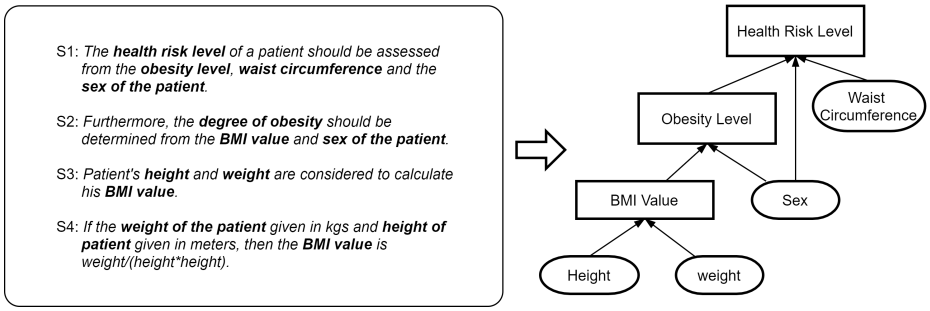

Fig. 2: The DRD of the health risk level description

4 Text2Dec Framework

To identify, extract and represent the decision requirement knowledge into a DRD from natural language documents, we have developed a novel Text2Dec framework. This approach follows three stages and is shown in Fig. 3. We use the above given *'health risk level'* example to illustrate the proposed methodology.

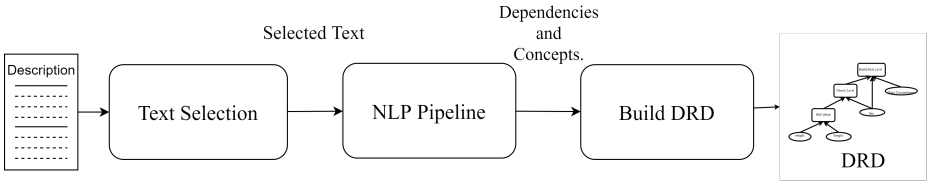

Fig. 3: Text2Dec framework

4.1 Stage I: Requirement Text Selection

In the first stage, the software automatically identifies the sentences containing decision dependency patterns as shown in Table 1 from the given input text using regular expressions and a list of predefined verbs that is used to identify requirement statements. Only sentences in these formats are considered for further processing. Where A and B are concepts, and an arrow <= indicates dependency.

4.2 Stage II: NLP Pipeline

In the second stage, using the open-source tool kits standford's coreNLP[2], NLTK [3], neuralcoref [4], and SpaCy libraries [5], an NLP pipeline has been built in python

[2] https://nlp.stanford.edu/software/
[3] https://www.nltk.org/
[4] https://github.com/huggingface/neuralcoref
[5] https://spacy.io/usage/

Table 1: Sentence patterns considered to extract dependencies.

Pattern	Example	Base Concepts	Derived Concepts
Passive A<=B	Patient's BMI value is determined from his height	height	BMI value
Active A=>B	Patient's height determines his BMI value	height	BMI value
Conditional A=>B	Unless the season is summer, do not plan a barbeque	season	plan a barbeque
Conditional A<=B	A customer is loyal, if his annual sales are high	annual sales	customer

code and applied on the running example. This pipeline consists of six steps to extract the dependencies from the text as shown in Fig. 4.

Step a: Preprocessing. The algorithm reads the selected text and preprocesses it by removing determinants such as *the*, *a*, and *an*. For example, the sentence *"The Risk Level is assessed from the BMI Level and waist circumference"* is converted to *"Risk level is assessed from BMI level and waist circumference."*

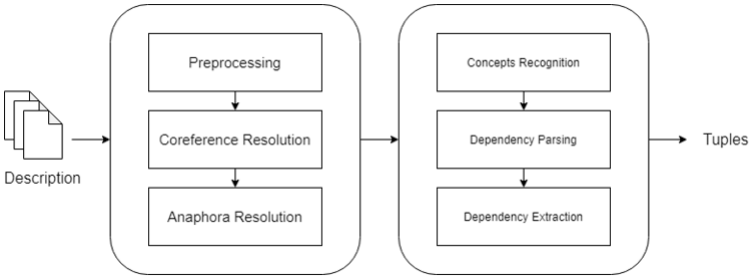

Fig. 4: NLP pipeline

Step b: Coreference Resolution. In this step, coreferences are identified and mapped based on semantic equivalence following the coreference resolution technique. Coreferences are the synonymous terms and phrases that occur in the text. They are resolved by replacing all coreferences with their first occurrence's term. For the running example, *degree of obesity* is replaced with *obesity level*.

Step c: Anaphora Resolution. Anaphoras such as cross-referred pronouns are detected and fixed by replacing pronouns with their referring owners. The ownership of the noun terms is also propagated over the conjunctions. This step is important because both the entity or the attribute of an entity could be the concept of interest. For the running example, the pronoun *his* is mapped with *patient's*. Also, the ownerships are simplified, e.g. *"of the patient"* is replaced with *"patient's"*. These both resolutions also help to correctly identify the intermediate decisions.

The result of these steps is: ***patient's*** *health risk level should be assessed from* ***patient's obesity level***, ***patient's*** *waist circumference and* ***patient's*** *sex. Furthermore,* ***patient's obesity level*** *should be determined from* ***patient's*** *BMI value and* ***patient's*** *sex ...*

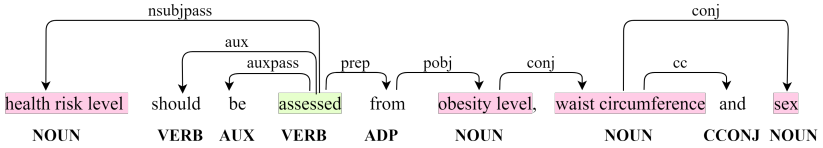

Fig. 5: Dependency parse tree for the example statement

Step d: Concept Recognition. At sentence level, the syntactic details are exploited by breaking each sentence into tokens. We process each sentence to recognise different concepts based on noun phrases, e.g. *health risk level*. To convert the noun phrases into concepts, a chunking technique has been used to merge noun tokens based on the parts-of-speech (POS) [6] tags such as NOUN for a noun, and PRON for a pronoun and PROPN for a proper noun. The result is that *[health:NOUN][risk:NOUN][level:NOUN]* is converted into *[health risk level:NOUN]*. In our risk level example, this step automatically generates a concept list c1: health risk level, c2: obesity level, c3: waist circumference, c4: sex from the statement "health risk level should be assessed from obesity level, waist circumference and sex".

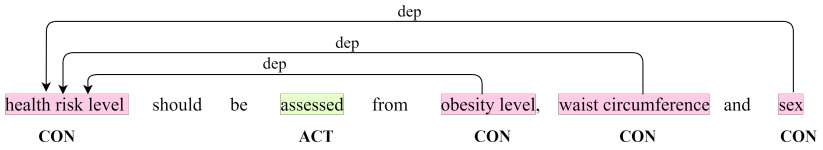

Fig. 6: Compacted parse tree for the example statement

Step e: Dependency Parsing. In this step, a verb-based dependency parse tree is built. To confirm dependency between concepts, we match the detected action verb with a predefined list of action verbs such as *require, decide, select determine, asses, calculate, consider ....* For the conditional statements when the main action verb is not detected, auxiliary verbs such as *"is"* are considered as action verbs. For example, a parse tree formed for the example statement is shown in Fig. 5. Even though *should* is a *verb* along with the root verb *assessed*, it is not considered as an action verb due to the lack of dependent concepts. Afterwards, a compact dependency parse tree is formed, highlighting only the concepts and action verbs for each sentence as shown in Fig. 6. Hence, the output of this step is a compact dependency parse tree for each sentence.

[6] https://universaldependencies.org/u/pos/

Step f: Dependency Extraction. In this step, the concepts are linked with the identified dependencies from the parse tree. Concepts are classified into base concepts and derived concepts depending on the role they are playing in the statement. In the example statement, the concept *health risk level* is playing a passive subject role (*nsubjpass*) with the action verb *assessed*. Therefore, it will be labelled as a derived concept. The concepts from step d and the dependencies from step e are joined, and a set of dependency tuples in the format *(relation, base concept, derived concept)* is generated based on the labelled concepts. The result of this example is a set of tuples consisting of {*(assess, obesity level, health risk level), (assess, waist circumference, health risk level), (assess, sex, health risk level),...*}. By using set data structures, duplication of tuples is avoided.

Once tuple sets for all sentences from the text are formed, a unified set of tuples is generated. This process depends on the order of the statements in the given description. Here, base concepts are filtered by removing the concepts that are identified as derived concepts in the later sentences and labelled as initial inputs to the DRD. A decision set is constructed where each decision contains a set of inputs and outputs. The final result is a DRG. In the running example, the DRG formed is {*(decisions, inputs, requirements)*} with decisions = {*([obesity level, waist circumference, sex],[health risk level]),([BMI value, sex],[obesity level]),([weight, height],[BMI value])*}, inputs = {*waist circumference, sex, weight, height*}, requirements = {*(assess, obesity level, health risk level), (assess, waist circumference, health risk level), (assess, sex, health risk level),....*}.

4.3 Stage III: DRD construction

In the final step of our framework, an XML file is generated based on the extracted concepts and their dependency relationships mentioned in the DRG. Only the connected graph component of the obtained DRG of the target decision is converted into a DRD. The XML file can be read by popular DMN tools such as Camunda[7]. For the running example generated DRD is shown in Fig.2. Other DMN constructs such as knowledge sources, business knowledge models, knowledge or authority requirements are not included yet.

5 Evaluation

We evaluated the Text2Dec framework on a simple "Prepayment" example and a larger "Employee Health Evaluation" case inspired from the example DRD given in [14]. The decision descriptions are designed to be natural and to contain various patterns of implicit and explicit dependencies and pose multiple challenges. The first description reads *"Prepayment is not required for loyal customers when the OrderAmount is small. A loyal customer is defined as such if his AnnualSales is high and his Customeryears is more than 5."*. The Text2Dec

[7] https://camunda.com/dmn/

approach generates the dependency tuples and corresponding DRD as shown in Fig. 7.

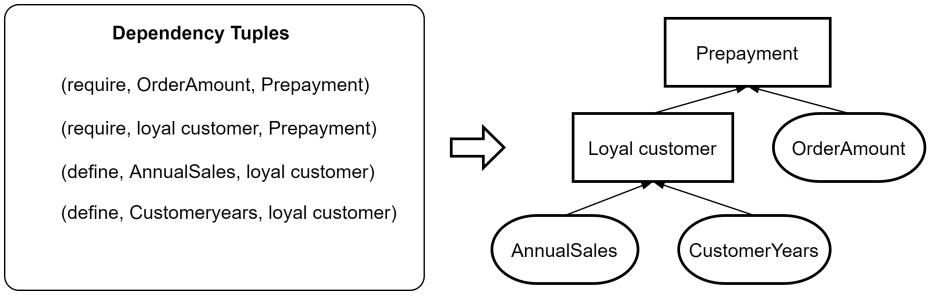

Fig. 7: The Dependencies and DRD of the Customer prepayment

The second description reads: *"Health evaluation criteria consider both physical health score and mental health score of a patient. The physical health score is determined from the physical fitness score, BMI based health risk level and healthiness of senses. Physical fitness score is calculated from the sex of a patient and results of various tests such as strength test, coordination test, agility test, stamina test and speed test. The patient's risk level should be assessed by determining the level of obesity based on BMI value, also on the waist circumference. The obesity level or degree of overweight should be assessed by determining the BMI value. If the weight of the patient given in kgs and length of patient given in meters, then the BMI value is weight/(length*length). The healthiness of the senses is calculated from the results of eye and hearing tests. Health evaluation also depends on the score of Mental health, which is determined from the EQ test result and the IQ test score. An IQ of a patient is assessed from testing his verbal, math and abstract levels."*

Table 2: Extracted dependencies from the health evaluation case

SNO	Base Concept	Action Verb	Derived Concept
1	physical health score	consider	Health evaluation criteria
2	mental health score	consider	Health evaluation criteria
3	physical fitness score	determined	physical health score
4	BMI based health risk level	determined	physical health score
5	healthiness of senses	determined	physical health score
6	sex	calculated	physical fitness score
...	...	...	...
22	math	assessed	IQ test score
23	abstract	assessed	IQ test score

By applying the Text2Dec framework on the second description, the following dependencies are extracted, shown in Table 2. Next, these dependencies are automatically transformed into an XML file which can be read by Camunda as shown in Fig. 8.

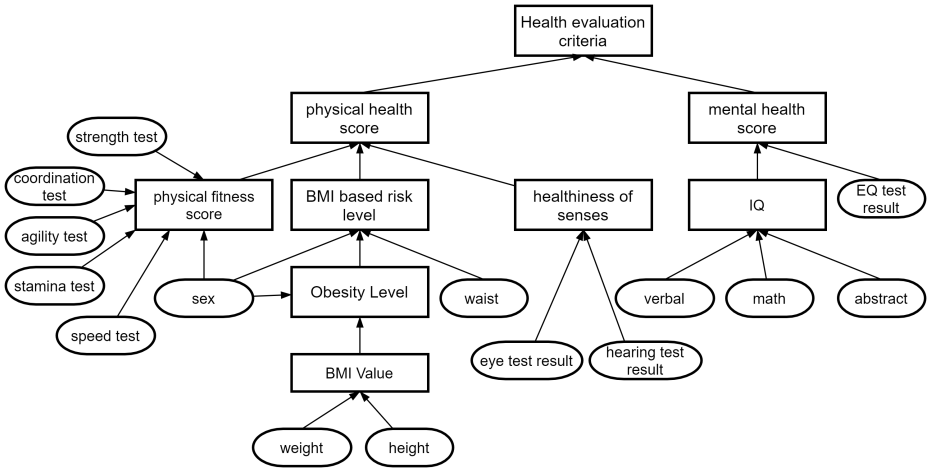

Fig. 8: The DRD of the health evaluation example

To assess the quality of the generated DRDs, we compared them with manually modelled DRDs. The generated DRDs have promising results: the structure of the DRD, and the number of nodes, information items, and information requirements stay the same. However, the obtained decision labels are slightly different. For example, the "math test result" is detected as "math" in our approach resulting in a small loss of semantic meaning.

6 Discussion

In this paper, we presented the Text2Dec framework for extracting DRDs from textual descriptions. The obtained DRDs correspond with those manually modelled, given the assumptions stated below, showcasing the promising applicability of our work. Automatically extracting DRDs can be useful for rapid prototyping. These prototype models can then be manually completed into full models with the information needed for the execution (e.g. decision logic). The presented framework can serve as a cornerstone for automatic decision model extraction from text.

This methodology faces several challenges. First, the ambiguity in language is hard to grasp. Therefore, we assume that the input decision descriptions consist of non-ambiguous and grammatically sound full English sentences. Nevertheless, our methodology allows for expanding the vocabulary and detection of patterns. Secondly, the main decision must be correctly identified for the DRD construction which can be hard in long texts. Thus, the Text2Dec framework relies on the assumption that the description is sequential, contains no irrelevant information or redundancies, and contains only one main decision. Otherwise, this could lead to unconnected components in the intermediate decisions in the DRD. The applied coreference resolution is a crucial step in this regard to link the cross-referred concepts in the text. It is used to identify the links between different

implicit dependencies between the components of the DRD. Thirdly, not all derived inputs are inputs in the real models. Some of these are knowledge sources or business knowledge models. This distinction is not yet implemented and is needed to make models more clean and consistent.

For future work, we will extend the Text2Dec framework to automatically extract the decision logic found in textual documents in the form of decision tables. This way, we can automatically derive a fully functional decision model from the text. Moreover, we would like to extend our work towards full end-to-end automation for decision knowledge and decision support applications. Secondly, we plan to evaluate the performance of the proposed methodology quantitatively. To analyse the complexity of the generated models in realistic decision support systems, we will apply the Text2Dec framework on a series of real-world cases which also includes the challenges of ambiguity and redundancy. Opportunities exist for fellow researchers to investigate automatic conformance checking with the textual guidelines, conversational decision support agents and explainability in decision making.

7 Conclusion

In this study, we contribute to the BDM and DMN research by presenting a novel Text2Dec framework for automating the extraction of decision dependencies from business decisions' descriptions. The evaluation shows that the generated DRDs correspond with the manually developed models with in terms of structure and semantics. This work paves the way towards automated decision modelling and logic extraction.

References

1. van der Aa, H., Di Ciccio, C., Leopold, H., Reijers, H.A.: Extracting declarative process models from natural language. In: International Conference on Advanced Information Systems Engineering. pp. 365–382. Springer (2019)
2. van der Aa, H., Leopold, H., Batoulis, K., Weske, M., Reijers, H.A.: Integrated process and decision modeling for data-driven processes. In: International Conference on Business Process Management. pp. 405–417. Springer (2015)
3. Batoulis, K., Meyer, A., Bazhenova, E., Decker, G., Weske, M.: Extracting decision logic from process models. In: CAiSE. pp. 349–366. LNCS, Springer (2015)
4. Bazhenova, E., Buelow, S., Weske, M.: Discovering decision models from event logs. In: International Conference on Business Information Systems. pp. 237–251. Springer (2016)
5. Calvanese, D., Dumas, M., Laurson, Ü., Maggi, F.M., Montali, M., Teinemaa, I.: Semantics and analysis of DMN decision tables. In: International Conference on Business Process Management. pp. 217–233. Springer (2016)
6. Campos, J., Richetti, P., Baião, F.A., Santoro, F.M.: Discovering business rules in knowledge-intensive processes through decision mining: an experimental study. In: International Conference on Business Process Management. pp. 556–567. Springer (2017)

7. Danenas, P., Skersys, T., Butleris, R.: Natural language processing-enhanced extraction of SBVR business vocabularies and business rules from UML use case diagrams. Data & Knowledge Engineering p. 19 (2020)
8. Dasseville, I., Janssens, L., Janssens, G., Vanthienen, J., Denecker, M.: Combining DMN and the knowledge base paradigm for flexible decision enactment. Supplementary Proceedings of the RuleML 2016 Challenge 1620 (2016)
9. De Smedt, J., Hasić, F., vanden Broucke, S.K., Vanthienen, J.: Holistic discovery of decision models from process execution data. Knowledge-Based Systems 183, 15 (2019)
10. Dragoni, M., Villata, S., Rizzi, W., Governatori, G.: Combining NLP approaches for rule extraction from legal documents. In: MIREL (2016)
11. Figl, K., Mendling, J., Tokdemir, G., Vanthienen, J.: What we know and what we do not know about DMN. Enterp. Model. Inf. Syst. Archit. Int. J. Concept. Model. 13, 2:1–16 (2018)
12. Friedrich, F., Mendling, J., Puhlmann, F.: Process model generation from natural language text. In: International Conference on Advanced Information Systems Engineering. pp. 482–496. Springer (2011)
13. Froelich, J., Ananyan, S.: Decision support via text mining. In: Handbook on Decision Support Systems (2008)
14. Hasic, F., Vanthienen, J.: Complexity metrics for DMN decision models. Comput. Stand. Interfaces 65, 15–37 (2019)
15. Honkisz, K., Kluza, K., Wisniewski, P.: A concept for generating business process models from natural language description. In: KSEM (2018)
16. Janssens, L., Bazhenova, E., Smedt, J.D., Vanthienen, J., Denecker, M.: Consistent integration of decision (DMN) and process (BPMN) models. In: CAiSE Forum. CEUR Workshop Proceedings, vol. 1612, pp. 121–128. CEUR-WS.org (2016)
17. Janssens, L., De Smedt, J., Vanthienen, J.: Modeling and enacting enterprise decisions. In: International Conference on Advanced Information Systems Engineering. pp. 169–180. Springer (2016)
18. Kluza, K., Honkisz, K.: From SBVR to BPMN and DMN models. proposal of translation from rules to process and decision models. In: International Conference on Artificial Intelligence and Soft Computing. pp. 453–462. Springer (2016)
19. Sànchez-Ferreres, J., Burattin, A., Carmona, J., Montali, M., Padró, L.: Formal reasoning on natural language descriptions of processes. In: BPM (2019)
20. Silver, B.: DMN Method and Style. 2nd Edition: A Business Pracitioner's Guide to Decision Modeling. Cody-Cassidy Press (2018)
21. Sintoris, K., Vergidis, K.: Extracting business process models using natural language processing (NLP) techniques. 2017 IEEE 19th Conference on Business Informatics (CBI) 01, 135–139 (2017)
22. Taylor, J., Fish, A., Vanthienen, J., Vincent, P.: Emerging standards in decision modeling. In: Intelligent BPM Systems: Impact and Opportunity. pp. 133–146. BPM and Workflow Handbook series, iBPMS Expo (2013)
23. Valencia-Parra, Á., Parody, L., Varela-Vaca, Á.J., Caballero, I., Gómez-López, M.T.: DMN for data quality measurement and assessment. In: International Conference on Business Process Management. pp. 362–374. Springer (2019)
24. Vanthienen, J., Dries, E.: Illustration of a decision table tool for specifying and implementing knowledge based systems. International Journal on Artificial Intelligence Tools 3(2), 267–288 (1994)