Supporting Process Model Validation through Natural Language Generation

Henrik Leopold[1] Jan Mendling[2] Artem Polyvyanyy[3]

The design and development of process-aware information systems is often supported by specifying requirements as business process models. Although this approach is generally accepted as an effective strategy, it remains a fundamental challenge to adequately validate these models given the diverging skill set of domain experts and system analysts. As domain experts often do not feel confident in judging the correctness and completeness of process models that system analysts create, the validation often has to regress to a discourse using natural language. In order to support such a discourse appropriately, so-called verbalization techniques have been defined for different types of conceptual models. However, there is currently no sophisticated technique available that is capable of generating natural-looking text from process models. The reason why a proper process model verbalization technique is still missing might be a result of the difficulty to meet this challenge. A process model verbalization technique has to serialize the non-sequential structure of a process model into sequential, yet execution-order preserving, text. In addition, it must be capable of analyzing the short and grammatically varying labels of process model elements and of annotating them with their semantic components like action or business object. Furthermore, the verbalization technique needs to handle optionality of certain pieces of information. In the paper [LMP14], we address this research gap and propose a technique for generating natural language texts from business process models.
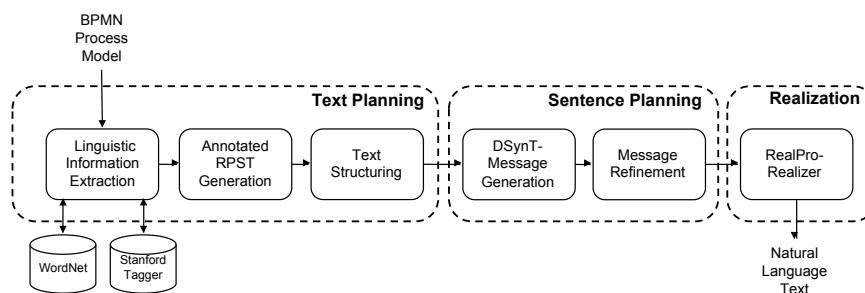

Abb. 1: Architecture of our Natural Language Generation System

The architecture of our text generation technique is building on the traditional pipeline concept from natural language generation systems. The basic rationale of the technique is to utilize the existing information from the model to generate a text. In order to derive

---

[1] VU University Amsterdam, De Boelelaan 1081, 1081HV Amsterdam, The Netherlands, h.leopold@vu.nl

[2] WU Vienna, Welthandelsplatz 1, 1020 Vienna, Austria, jan.mendling@wu.ac.at

[3] Queensland University of Technology, Brisbane, QLD 4001, Australia

a sequence of sentences, we linearize the model via the creation of a tree structure. In particular, the text generation technique comprises six components (see Figure 1):

1. *Linguistic Information Extraction*: Extraction of linguistic components from the process model element labels.

2. *Annotated RPST Generation*: Linearization of process model through the generation of a tree structure. In addition, each node is annotated with the linguistic information from the previous step.

3. *Text Structuring*: Application of text structuring techniques, such as the insertion of paragraphs and bullet points, based on the computed tree structure.

4. *DSynT-Message Generation*: Generation of an intermediate linguistic message structure for each node of the tree. This component represents the core of the generation technique.

5. *Message Refinement*: Refinement of the generated messages through aggregation or the introduction of referring expressions and discourse markers.

6. *RealPro-Realizer*: Transformation of intermediate message structures to grammatically correct sentences.

To demonstrate the capability of the proposed technique for generating natural language texts from process models, we conducted a two-step evaluation. First, we applied our technique to real-world process models and investigated how the generated texts compare to textual descriptions created by humans. Second, we studied in how far humans are capable of making sense of the generated texts. To this end, we asked humans to transform the generated texts back into process models. The first evaluation step showed that the generated texts convey the model semantics in a more compact and also syntactically less complex manner. Due to the design of the technique, the generated texts are closer to the model and describe the model content and control flow explicitly. The second evaluation step demonstrated that the generated texts are very informative and can successfully be interpreted by humans.

Based on our findings, we conclude that the proposed text generation technique has the potential to facilitate the validation discourse between domain experts and process analysts. First, the generated texts support domain experts in understanding the details of process models even if they are not familiar with process modeling. Second, the text generation may also train domain experts in reading and interpreting process models. As long as text and model are presented together, readers can see and learn about the connection between model and text. Thus, their overall familiarity with process models can be expected to increase in the long term.

Literatur

[LMP14] Leopold, Henrik; Mendling, Jan; Polyvyanyy, Artem: Supporting process model validation through natural language generation. Software Engineering, IEEE Transactions on, 40(8):818–840, 2014.