WoPeD goes NLP: Conversion between Workflow Nets and Natural Language

Thomas Freytag, Cooperative State University (DHBW) Karlsruhe, Germany
thomas.freytag@dhbw-karlsruhe.de
Philip Allgaier, bpcompass GmbH, Karlsruhe and Hamburg, Germany
philip.allgaier@bpcompass.com

Abstract. WoPeD (**Wo**rkflow **Pe**trinet **D**esigner) is an open-source Java software for designing business processes in terms of workflow nets, a common extension of Petri nets. This demo lays the focus on two recently added features making use of Natural Language Processing (NLP) algorithms in order support the conversion of a graphical process model into a textual process description and vice versa.

Keywords: Petri net tool, workflow nets, NLP, Process2Text, Text2Process.

1 Introduction

Having started in 2003 as a simple Petri net editor and simulator, WoPeD has been continuously extended over the last fifteen years towards a rich software tool covering a variety of BPM topics. Based on the well-defined language of workflow nets as introduced in [1], WoPeD has been frequently serving as an educational tool for a hands-on experience to model, analyse and simulate business process as well as to understand the underlying algorithms and verification methods. For example, WoPeD can check a process model for soundness and other semantic properties, can interactively construct the coverability graph, and perform stochastic computations like capacity planning. Another feature allows the transformation between WoPeD's workflow net models and other common process model languages like YAWL, BPMN, or EPC via an AProMoRe[1] process repository. Additionally, WoPeD has served serveral researchers in the BPM community as an experimental platform to implement innovative project ideas [4] [5]. Most of these features have been presented at past BPM conference demo sessions (e.g. [2,3]). This demo paper focuses on the most recently added features of WoPeD building the bridge between Natural Language Processing and BPM.

2 NLP and BPM

Graphical process modeling languages have become an industry standard, allowing various stakeholders to collectively create visual process descriptions on a conceptual level. However, graphical models frequently co-exist with informal textual process descriptions, be it for lack of skills or tools or time to create

[1] www.apromore.org

F. Casati et al. (Eds.): Proceedings of the Dissertation Award and Demonstration, Industrial Track at BPM 2018, CEUR-WS.org, 2018.

a graphical model or because a text-based description is required for a process handbook as sometimes mandated by QA agencies. This has brought up the question in BPM research transforming these two process representations into each other, i.e. to automatically generate the textual equivalent of a given graphical process model and to automatically generate an appealing visual process model from a given textual representation. There are interesting practical use cases for this functionality, e.g. to check the mutual compliance between a textual and graphical version of a business process or to facilitate the inclusion of visually impaired persons in process modelling activities by providing them with a text-based interface for understanding and describing a business process. The following two sections give an overview of how NLP algorithms have been integrated into WoPeD: Firstly, to export the current process editor contents into readable English text ("Process2Text") and secondly to synthetize an workflow net from a given verbal process specification ("Text2Process").

3 Process2Text: Verbalization of a Workflow Net

In [6], an NLP-based approach has been introduced taking a BPMN diagram as input and generating a naturual language text as output describing the modelled behavior. Figure 1 shows the outline of the underlying algorithm. In the first step, called "Text Planning", the relevant linguistic information is extracted from the workflow-net-based process model, mainly by analyzing labels of model elements, control flow dependencies and task resource assignments.

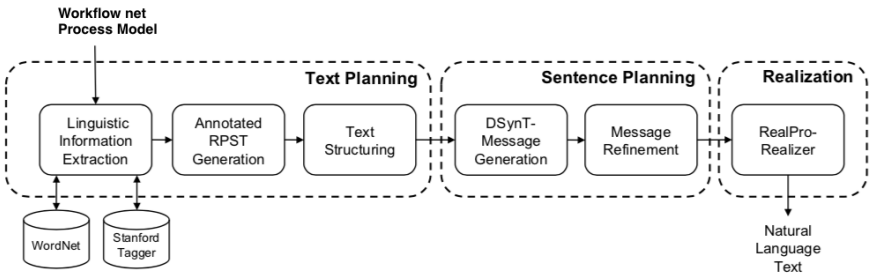

Fig. 1. The Process2Text algorithm (adapted from [6])

The analysis is done by two freely available standard NLP tools, Stanford Tagger[2] and WordNet[3]. The result is converted into a Refined Process Structure Tree (RPST), a generic data structure to represent components of a process

[2] http://nlp.stanford.edu/software/lex-parser.shtml
[3] http://wordnet.princeton.edu

graph being introduced in more detail e.g. in [7]. After applying various heuristics to restructure and fine-tune the RPST, the second major step of the algorithm "Sentence Planning" is applied. Here, the leaves of the RPST are processed and enriched with semantic information derived from the control flow in the graph. After an additional refinement of the text fragments, a realizer tool (used here: RealPro [8]) is applied in order to generate a naturally-sounding English text.

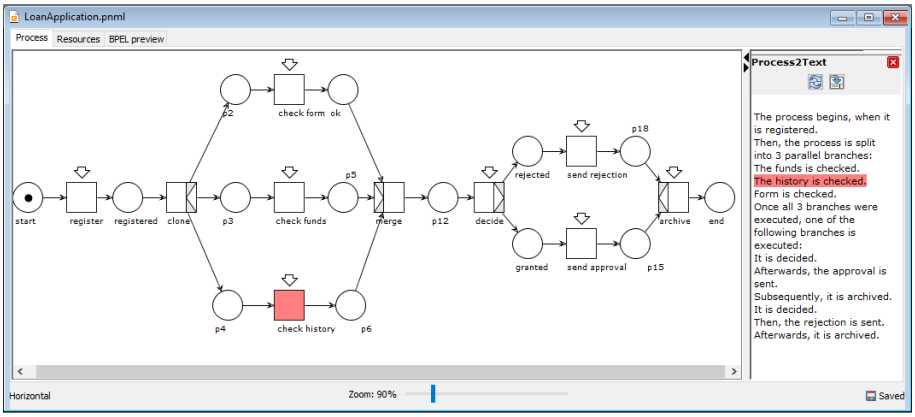

Fig. 2. Process2Text sidebar in WoPeD

The implementation in WoPeD takes major parts of the solution presented in [6], however making the transformation process applicable to workflow nets instead of BPMN diagrams and embedding the algorithm into the WoPeD process editor by displaying the generated text together with the process graph. Figure 2 shows a screenshot of the WoPeD editor with a graphical model and a "Process2Text" sidebar containing the auto-generated text. The associated process elements are highlighted once the user clicks on a specific text item line in the sidebar (shown in red).

4 Text2Process: Generate a Workflow Net from Text

The opposite direction of the transformation is based on an algorithm well-documented e.g. in [9] and [10]. The transformation uses the same freely-available NLP tools as the Process2Text algorithm above. As a first step, the input string is analyzed by the Stanford parser tool and split up into sentences and words according to the underlying grammar rules. The subsequent steps make use of two semantical word databases WordNet (see above) and FrameNet [4].

[4] http://framenet.icsi.berkeley.edu/fndrupal

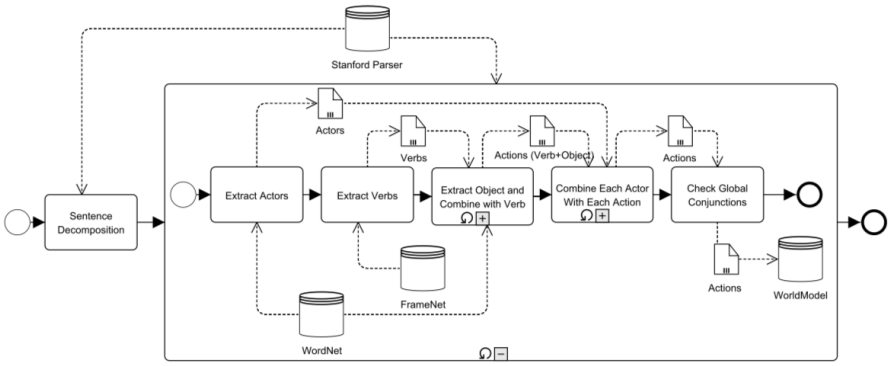

Fig. 3. The Text2Process Algorithm (taken from [9])

This mainly identifies the semantically relevant elements of the textual process model (e.g. actors, actions, business objects). The result is written into a data structure called *WorldModel* where all contributing model elements and their interrelations are stored. The WorldModel is converted into PNML, the standard XML-based representation for workflow nets. The last step is to open up a new WoPeD editor window and create coordinates for the automatically generated model by calling WoPeD's built-in layout beautifier. Figure 3 roughly visualizes the algorithm. Figure 4 shows a screenshot of the dialog for entering the textual process description (either by typing in or by uploading a text file) and below the resulting auto-generated and auto-layouted workflow net.

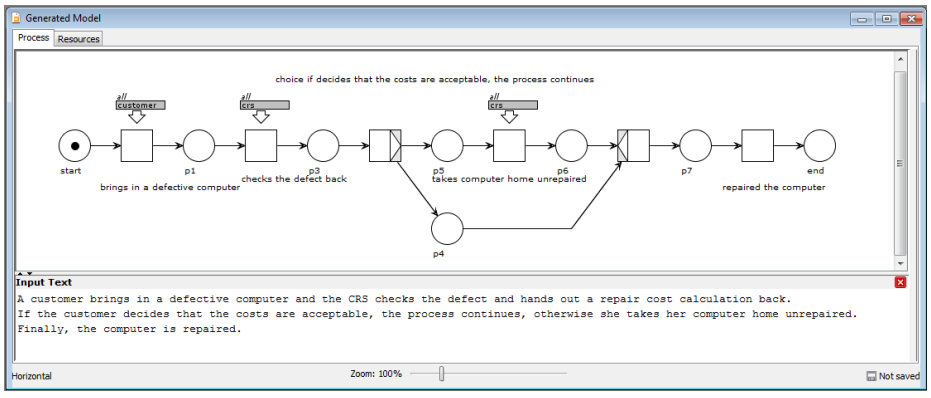

Fig. 4. The Text2Process dialog in WoPeD

5 Conclusion and Outlook

The above features are still in an experimental stage suffering from the general problem of semantical ambiguities in natural language processing in both directions. However, the plan is to improve the transformation quality in future releases of WoPeD. The software is open source and can be downloaded from the homepage[5] or directly from Sourceforge[6].

References

[1] Van der Aalst, W.M.P.; van Hee, K.M.: Workflow Management: Models, Methods, and Systems MIT Press 2002.

[2] Freytag, T.; Sänger, M.: WoPeD - An Educational Tool for Workflow Nets. BPM demo session 2014. http://ceur-ws.org/Vol-1295/paper3.pdf, accessed June 2018.

[3] Freytag, T.; Allgaier, P. Burratin A.; Danek-Bulius, A.: WoPeD - A "Proof-of-Concept" Platform for Experimental BPM Research Projects. BPM demo session 2017, http://ceur-ws.org/Vol-1920/BPM_2017_paper_190.pdf, accessed June 2018.

[4] Reijers, H.A.; Freytag, T.; Mendling, J.; Eckleder, A.: Syntax highlighting in business process models. Decision Support Systems 51(3): 339-349 (2011).

[5] Mendling, J.: Metrics for Process Models, Springer Verlag, Berlin, 2008.

[6] Leopold, H.; Mendling, J.; Polyvyanyy, A.: Generating Natural Language Texts from Business Process Models. In: Ralyté, J., Franch, X., Brinkkemper, S., Wrycza, S. (eds) Advanced Information Systems Engineering. CAiSE 2012. LNCS, vol 7328. Springer, Berlin, Heidelberg 2012.

[7] Polyvyanyy A.; Vanhatalo J.; Völzer H.: Simplified Computation and Generalization of the Refined Process Structure Tree. In: Bravetti M.; Bultan T. (eds) Web Services and Formal Methods. Lecture Notes in Computer Science, vol 6551, Springer, Berlin, Heidelberg, 2010.

[8] Lavoie, B.; Rambow, O.: A Fast and Portable Realizer for Text Generation Systems. 5th Conference on Applied Natural Language Processing, 1997, www.aclweb.org/anthology/A97-1039, accessed June 2018.

[9] Friedrich, F.; Mendling, J.; Puhlmann, F.: Process Model Generation from Natural Language Text. In: Mouratidis H., Rolland C. (eds) Advanced Information Systems Engineering. CAiSE 2011. Lecture Notes in Computer Science, vol 6741, Springer, Berlin, Heidelberg 2011.

[10] Riefer, M.; Ternis, S.; Thaler, T: Mining Process Models from Natural Language Text: A State-of-the-Art Analysis. Processings of Multikonferenz Wirtschaftsinformatik (MKWI-16), March 9-11, Illmenau, Germany, 2016.

[5] www.woped.org

[6] www.sourceforge.net/projects/woped/