Automatic detection and resolution of lexical ambiguity in process models

Mendling, Jan; Pittke, Fabian; Leopold, Henrik

Published in:
Software Engineering 2016

Published: 01/02/2016

Document Version
Publisher's PDF, also known as Version of record

Jens Knoop, Uwe Zdun (Hrsg.): Software Engineering 2016,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2016 75

Automatic Detection and Resolution of Lexical Ambiguity in Process Models (Extended Abstract)

Fabian Pittke[1], Henrik Leopold[2], and Jan Mendling[3]

Process models play an important role in various system-related management activities including requirements elicitation, domain analysis, software design as well as documentation of databases, business processes, and software systems. However, it has been found that the correct and meaningful usage of process models appears to be a challenge in practical settings requiring the usage of automatic model analysis techniques. Up until now, such automatic quality assurance is mainly available for checking formal properties of process models. For instance, there is a rich set of analysis techniques to check control-flow-related properties of process models, such as soundness. There are only a few techniques available for checking guidelines on text labels with regard to terminological ambiguity. Moreover, the terminology problem is even more serious in process models since the process model gives an abstract view of the business process and provides only limited context to detect and resolve ambiguity issues. It is thus the goal of [PLM15] to address the need for automatic techniques as well as to define detection and resolution technique for textual ambiguities that improve the terminological quality of a process models and repositories thereof.

For that purpose, our approach addresses three important issues, i.e. the manual effort, the missing focus on process model text fragments, and the focus on single models. First, the *required manual effort*, refers to the extensive amount of manual work that is required to detect and resolve ambiguities in process models. The human effort can be tremendous since organizations tend to maintain several hundreds or even thousands of process models. Second, the *missing focus on process model text fragments* refers to the fact that many approaches of ambiguity detection and resolution are tailored to deal with sentences and phrases taken from a grammatically and syntactically correct natural language text. However, the elements of process models contain only short textual fragments that do not exhibit a complete or a correct sentence structure impeding the direct application of such approaches. Third, the *focus on single models* relates to the observation that available techniques consider only single models or smaller units thereof. Hence, these techniques address ambiguities within a single document or process model. However, since we assume a repository of several process models, the correction of ambiguities on document level might introduce an inconsistency in another document or model.

Our approach introduces the notion of semantic vectors that represents all the possible meanings of a term in the context of a process model. A semantic vector interprets the

[1] WU Vienna, Welthandelsplatz 1, 1020 Vienna, Austria, fabian.pittke@wu.ac.at
[2] VU University Amsterdam, De Boelelaan 1081, 1081HV Amsterdam, The Netherlands, h.leopold@vu.nl
[3] WU Vienna, Welthandelsplatz 1, 1020 Vienna, Austria, jan.mendling@wu.ac.at

vector dimensions as word meanings and quantifies the occurrence of a particular word meaning with a score that is non-zero. The higher the score, the more prevailing the respective meaning in the process model. Furthermore, our approach operationalizes lexical ambiguity by defining necessary and sufficient ambiguity conditions. While the necessary ambiguity conditions focus on the basic characteristics of lexical ambiguities, the sufficient ambiguity conditions explicitly include the usage context of the respective term. The rationale is motivated by the fact that only because a term *might* be ambiguous, this does not necessarily mean that it actually is. A word is ambiguous only, if the context of the word is not sufficient to infer the correct meaning.

This logic has been used to define ambiguity detection and resolution techniques. Accordingly, the detection technique makes use of the previous conceptualization and combines it with the lexical database BabelNet and its integrated word sense disambiguation method to instantiate the semantic vectors and to evaluate the ambiguity conditions. Furthermore, the approach groups such process models, in which a specific term is used with a similar intention, i.e. models with semantic vectors close to each other. The resolution techniques employs different strategies based on semantic relations that suggest alternative terms for replacement. These strategies are, for instance, based on the hypernym and hyponym relations between terms or the specificity of alternative terms. Again, the resolution strategies make use of the lexical database BabelNet, which provides a rich knowledge base of possible word meanings and semantic relations between them.

These techniques have been evaluated by using three process model collections from practice varying in size, domain, and degree of standardization. In particular, the performance of the detection technique was evaluated by conducting an extensive user experiment. The experiment involved six native English speakers who provided their interpretation of a term in a given model. The performance of the resolution technique has been assessed by quantifying the degree of ambiguity and comparing it before and after applying the resolution strategies to the test collections. The evaluation with the English native speakers illustrates that the detection technique uncovers a relevant share of ambiguous terms within the test collections. Moreover, the introduced metrics highlight the positive effect of the resolution approach, which has lead to a significant reduction of ambiguity.

The results of this research underline the importance of terminological consistency of process models and other conceptual models that are affected by terminological inconsistencies, such as goal models, use case models, or feature models. The results also provide support for the revision of such inconsistent models and repositories and for sustaining the consistency of language and terminology over a longer period of time. Moreover, the techniques make an important contribution to existing quality assurance techniques and represent an important step towards the automated quality assurance of process models.

References

[PLM15] Pittke, Fabian; Leopold, Henrik; Mendling, Jan: Automatic Detection and Resolution of Lexical Ambiguity in Process Models. IEEE Transactions on Software Engineering, 41(6):526–544, 2015.