How To Detect Semantic Business Process Model Variants?

Agnes Koschmider
koschmider@aifb.uni-karlsruhe.de

Andreas Oberweis
oberweis@aifb.uni-karlsruhe.de

Institute of Applied Informatics and Formal Description Methods
Universität Karlsruhe (TH)
76187 Karlsruhe, Germany

ABSTRACT

Precise modeling of business processes has paved the way to realize process aware information systems that include allocations of resources, communication services, or hardware devices to users. Changes in business strategies or new business opportunities may result in modifications of implemented functionalities of information systems and their underlying business process models. The result of these modifications are business process model variants. In this paper, we propose an algorithm for determining linguistic similarities between business process model variants in order to faciliate process redesign.

Categories and Subject Descriptors

D.2.8 [**Software Engineering**]: Metrics/Measurement—*Process Metrics*

Keywords

Business Process Models, Ontologies, Petri nets, Similarity

1. INTRODUCTION

Changes in business strategies or new business collaborations lead to modifications of implemented functionalities of information systems and their underlying business process models. The result of these modifications are business process model variants that have to be distinguished when business processes are redesigned.

We aim at providing a solution for easier detection of process model variants and faster redesign of process models. Our approach can faciliate rapid adoptions to a changing environment due to reduced process modeling efforts. Our solution is based upon the calculation of similarities of process element names without considering structural properties of process models. The decision about the similarity of process model variants is easier if the occurrence of synonyms, homonyms, and different abstraction levels of element names can be recognized. We describe Petri nets with an Ontology Language-based format (OWL). These so-called semantic business process models (SBPMs) facilitates (semi-) automatic similarity measurements. Furthermore, they make it possible to implement an efficient algorithm for (semi-)automatic similarity computation between process model variants. A SBPM model corresponds to the instantiation of the Predicate/Transition net (Pr/T) ontology (as described in [2]), which can be represented in OWL syntax. The extraction of identifiers of business process elements and the mapping to the Pr/T net ontology is being carried out automatically and is not directly visible to the modeler. The result of mapping efforts are OWL files, which can afterwards be (semi-) automatically manipulated.

2. SIMILARITY MEASURES

The next subsection classifies linguistic structures of element names, which can be used to determine similarities between business process model variants.

2.1 Classification

In order to classify similarity measures we exploit the lexical taxonomic structure of WordNet[1], which provides synonym- (two terms have an identical meaning) and hyperonym/ hyponym- (two terms have an *is-a* relationship; subclass/ class relationship) sets consisting of nouns, verbs, adjectives, and adverbs. We extend this given structure of WordNet with homonyms (two terms have same pronounciation, but different meaning) and use this extended structure of terms as our classification scheme for similarity measurements. Similarity calculation of synonyms and homonyms is presented in [1]. The following subsection presents a formal similarity measure in order to detect hyperonyms/hyponyms.

2.2 Abstraction Level Similarity

To detect hyperonyms/hyponyms we compute abstraction level similarities, which take into account the depth of terms in lexical reference systems such as WordNet. The abstraction of terms correlates negatively with the depth of terms. The WordNet taxonomy can be represented as a rooted tree with the `root` c_0. Figure 3 depicts as an example a WordNet tree where `paper` and `document` have the common superconcept `writing` (one synonym for `writing` is `writing material`). In order to compute the abstraction level similarity of two concept instances we first integrate the processes

[1] http://wordnet.princeton.edu/

SAC'07, March 11-15, 2007, Seoul, Korea.

to be compared into one process. The integration approach is adopted from [3]. To compute correct abstraction level similarities it is useful to consider process element names in refined processes as well. Element names used on an abstract process level may be modeled in detail in a process on a lower level of abstraction. Our implemented similarity measurement system compares all instances of the same concept and returns only the pair with the highest similarity degree. Without the process integration procedure the system may propose incorrect pairs due to insufficient information. For the integration procedure we assume that processes are modeled top-down. If a transition that has to be refined has a cycle, then the cycle is broken into two subprocesses.

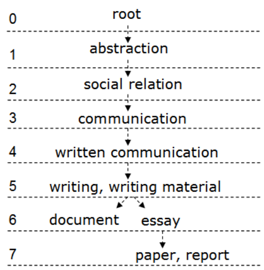

Figure 1: Term structure in WordNet

The abstraction level similarity requires to compute first the depth of terms (sim_{dep}), which is calculated as defined in [4]. Consequently, the calculation of sim_{dep} requires to find the most specific common superconcept of c_1 and c_2 called c_3. Then *N1* is the number of nodes on the path from c_1 to c_3 and *N2* is the number of nodes on the path from c_2 to c_3. *N3* is the number of nodes on the path from c_3 to *root*:

$$sim_{dep}(c_1, c_2) = \frac{2*N3}{N1+N2+2*N3}$$

sim_{dep} for (*paper* vs. *document*) equals 0.77.

It may be possible that a modeler e.g. does not subsume a *software product* to *products* but rather considers it as a *development project*. For this reason the *context* of concept instances as defined in [1] has to be regarded. Then the abstraction level similarity sim_{abs} is a combination of sim_{dep} and a structural similarity as defined in [1]:

$$sim_{abs}(c_1, c_2) := \frac{sim_{dep}(c_1, c_2) + sim_{str}(c_1, c_2)}{2}$$

Finally, we combine all similarity measures to sim_{com}[2]:

$$sim_{com}(c_1, c_2) :=$$

$$\frac{w_{c_{syn}} * sim_{syn}(c_1, c_2) + w_{c_{ling}} * sim_{ling}(c_1, c_2) + w_{c_{str}} * sim_{str}(c_1, c_2) + w_{c_{abs}} * sim_{abs}(c_1, c_2)}{w_{c_{syn}} + w_{c_{ling}} + w_{c_{str}} + w_{c_{abs}}}$$

3. IMPLEMENTATION

We have implemented a Petri net editor called SemPeT[3] that offers SBPM export. In order to support similarity measurement between SBPMs we have integrated FOAM, an alignment and mapping framework for ontologies, into SemPeT. On top of FOAM we have implemented the features of measuring combined similarity. In the first iteration of similarity measurements, instances of the same concept are compared. After several iterations (by default the system iterates three times) the system shows only the instances with the highest combined similarity degree. Extraction of instances of same concept and display of pairs with highest syntactical similarity are the first two steps of our similarity algorithm for detection of business process model variants (see Figure 2). The linguistic similarity is only computed for concept instances with $sim_{syn} \neq 1.0$. The next step in our algorithm is the calculation of structural similarities. In case of linguistic or structural similarity of 0.0 the abstraction level similarity is computed.

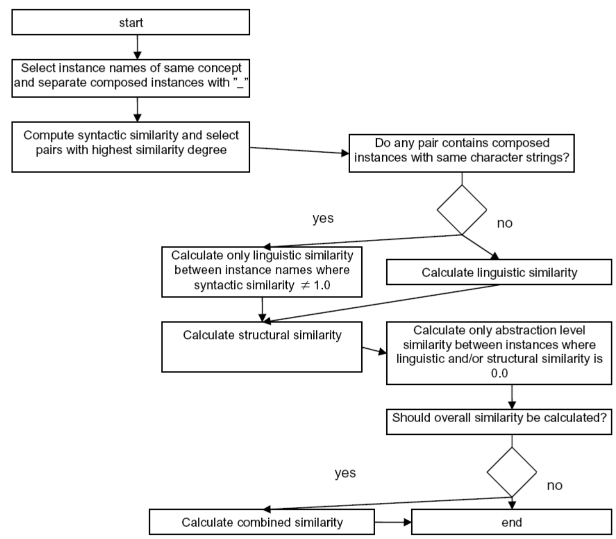

Figure 2: Process Variant Similarity Algorithm

4. CONCLUSION

In this paper we have sketched a similarity measure that detects different abstraction levels of process element names. The benefit of our approach is to facilitate process reuse by comparing process element names with process model variants and to (semi-) automatically detect process variants instead of purely manual comparison.

5. REFERENCES

[1] M. Ehrig, A. Koschmider, and A. Oberweis. Measuring Similarity between Semantic Business Process Models. In *Proc. of the Fourth Asia-Pacific Conference on Conceptual Modelling*, Australia, 2007. (to appear).

[2] A. Koschmider and A. Oberweis. Ontology based business process description. In *Proc. of the CAiSE-05 Workshops*, pages 321–333, Portugal, 2005. Springer.

[3] G. Lausen. Modeling and Analysis of the Behavior of Information Systems. *IEEE Transactions on Software Engineering*, 14(11):1610–1620, 1988.

[4] Z. Wu and M. Palmer. Verb semantics and lexical selection. In *32nd. Annual Meeting of the Association for Computational Linguistics*, pages 133 –138, 1994.

[2] w, sim_{syn} (syntactic similarity), sim_{ling} (linguistic similarity) and sim_{str} are defined in [1].

[3] http://aifbserver.aifb.uni-karlsruhe.de/sempet/index.htm