Triple-S: A Matching Approach for Petri Nets on Syntactic, Semantic and Structural level

Ugur Cayoglu, Andreas Oberweis, Andreas Schoknecht, and Meike Ullrich

Karlsruhe Institute of Technology (KIT), Institute of Applied Informatics and Formal Description Methods (AIFB)
first_name.surname@kit.edu

1 The Approach to Process Model Matching

1.1 Overview

So far, a handful contributions have been made to the problem of process model matching. The Triple-S matching approach adheres to the KISS principle by avoiding complexity and *keeping it simple and stupid*. It combines similarity scores of independent levels as basis for a well-founded decision about matching transition pairs of different process models. The following three levels and scores are considered:

- **Syntactic level -** $SIM_{syn}(a, b)$**:** For the syntactic analysis of transition labels we perform two preprocessing steps: (1) tokenization and (2) stop word elimination. The actual analysis is based on the calculation of Levenshtein distances between each combination of tokens (i.e. words) from the labels of transitions a and b. The final syntactic score is the minimum distance over all tokens divided by the number of tokens, i.e. the minimum average distance between each token.
- **Semantic level -** $SIM_{sem}(a, b)$**:** Prior to analysis, we perform the same preprocessing steps as above mentioned. Subsequently, we apply the approach of Wu & Palmer [1] to calculate the semantic similarity between each token of labels of transitions a and b based on path length between the corresponding concepts. The final semantic score is the maximum average similarity, i.e. it is calculated in an analogous manner to the final syntactic score.
- **Structural level -** $SIM_{struc}(a, b)$**:** Here, we investigate the similarity of transitions a and b through a comparison of (i) the ratio of their in- and outgoing arcs and (ii) their relative position in the complete net.

These three scores are combined to the final score $SIM_{total}(a, b)$ which represents the matching degree between two transitions a and b from different process models. It is calculated according to the following formula:

$$SIM_{total}(a, b) = \omega_1 * SIM_{syn}(a, b) + \omega_2 * SIM_{sem}(a, b) + \omega_3 * SIM_{struc}(a, b)$$

The three parameters ω_1, ω_2 and ω_3 define the weight of each similarity level. A threshold value θ is used to determine whether transitions actually match, i.e. iff $SIM_{total} \geq \theta$, two transitions positively match.

1.2 Specific Techniques

Compared to [2], the Triple-S approach makes several adjustments. Firstly, stop words are eliminated and the Levenshtein distance is calculated on the level of single tokens instead of complete sentences. Secondly, for the semantic level an established NLP approach is introduced. Finally, on the structural level TripleS performs contextual analysis by investigating local similarity only.

2 Application

2.1 Implementation

The Triple-S approach has been implemented using Java. For the calculation of the semantic score with the apporach of Wu & Palmer, the *WS4J Java API*[1] has been used to query Princeton's English *WordNet* 3.0 lexical database [3]. Relative positions of transitions are calculated using the implementation of Dijkstras algorithm by Vogella[2]. The code can be obtained from `http://code.google.com/p/bpmodelmatching/wiki/Download?tm=4` under *GNU GPL v3* license.

2.2 Evaluations

During our experiments we tried to approximate optimal results based on the gold standard examples. For the contest, we have used the following values: $\omega_1 = 0.45$, $\omega_2 = 0.3$, $\omega_3 = 0.25$ and $\theta = 0.6$. With those values we achieve values of 0.49 and 0.35 for precision and recall for the given gold standard examples. The Triple-S approach is currently developed as part of the ongoing SemReuse research project addressing business process model reuse. This contest on business process similarity presents a welcome possibility for first experiments. We are planning on refining the current measures for the individual levels, especially the semantic and structural level and improved detection of 1:n matches.

Acknowledgement: This work has been developed with the support of DFG (German Research Foundation) under the project SemReuse OB 97/9-1.

References

1. Zhibiao Wu and Martha Palmer. Verbs semantics and lexical selection. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, ACL '94, pages 133–138, 1994.
2. Marc Ehrig, Agnes Koschmider and Andreas Oberweis. Measuring similarity between semantic business process models. In *Proceedings of the fourth Asia-Pacific conference on Comceptual modelling*, APCCM '07, Vol. 67, pages 71–80, 2007.
3. Christiane Fellbaum. WordNet: An Electronic Lexical Database. Cambridge, MA, MIT Press, 1998.

[1] `https://code.google.com/p/ws4j/`
[2] `http://www.vogella.com/articles/JavaAlgorithmsDijkstra/article.html`