SIMILARITY OF ACTIVITIES IN PROCESS MODELS: TOWARDS A METRIC FOR DOMAIN-SPECIFIC BUSINESS PROCESS MODELING LANGUAGES

Hendrik Scholta

University of Münster - ERCIS, hendrik.scholta@ercis.uni-muenster.de

Follow this and additional works at: http://aisel.aisnet.org/ecis2016_rip

Recommended Citation

http://aisel.aisnet.org/ecis2016_rip/63

This material is brought to you by the ECIS 2016 Proceedings at AIS Electronic Library (AISeL). It has been accepted for inclusion in Research-in-Progress Papers by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.
SIMILARITY OF ACTIVITIES IN PROCESS MODELS: 
TOWARDS A METRIC FOR DOMAIN-SPECIFIC BUSINESS 
PROCESS MODELING LANGUAGES

Research in Progress

Scholta, Hendrik, University of Muenster - ERCIS, Leonardo-Campus 3, 48149 Muenster, 
Germany, hendrik.scholta@ercis.uni-muenster.de

Abstract

In the course of business process management, processes can be represented by models in order to 
document, analyze and improve processes. Many analyses of business process models require the cal-
culation of similarity values between activities of different process models. Process models can be cre-
ated using domain-specific business process modeling languages. These languages provide predefined 
domain-specific sets of activities and their properties and allow the annotation of property values based 
on glossaries. Since this information describes activities in detail, it enables a precise calculation of 
similarities between process model activities. However, the literature does not provide a suitable simi-
larity calculation for two activities in different process models created with the same domain-specific 
process modeling language. Consequently, relying on design science research the contribution of this 
paper is to propose such a similarity metric. The metric incorporates annotated property values of ac-
tivities and considers the properties’ different relevance, takes into account the characteristics of quan-
titative properties and analyzes the contexts of activities. The metric is conceptualized and demonstrated 
by an example. Future research requires its technical implementation and evaluation.

Keywords: Similarity Measure, Similarity Metric, Process Model Matching, Process Model Similarity, 
Process Management.

1 Introduction

Business Process Management (BPM) is the discipline of “supporting business processes using meth-
ods, techniques, and software to design, enact, control, and analyze operational processes involving 
humans, organizations, applications, documents and other sources of information” (van der Aalst et al., 
2003, p. 4) and has become an important domain in the research field of Information Systems (van der 
Aalst, 2013). Consequently, BPM aims at documenting, investigating and optimizing business pro-
cesses. In order to enable further analyses, business processes can be captured in process models.

Process models can be represented as directed graphs and can be developed using process modeling 
languages such as Business Process Model and Notation (BPMN) (Object Management Group, 2011) 
and Event-driven process chains (EPC) (Keller et al., 1992; Scheer et al., 2005). These notations are 
general-purpose process modeling languages and provide a high flexibility to create models with arbi-
trary complexity. However, this is not desirable in all modeling scenarios (Becker, Algermissen, et al., 
2012). Instead, domain-specific process modeling languages such as the PICTURE method and icebricks 
are tailored to the specifics of a domain. They provide a set of standardized domain-specific activities 
and make use of glossaries for annotations of standardized properties to activities such as executing 
roles and IT systems (Becker, Probandt, et al., 2012; Pfeiffer, 2008). Consequently, they restrict the 
degrees of freedom during the model creation. This results in a higher comprehensibility, standardization 
and comparability of business process models (Becker, Algermissen, et al., 2012).
In order to analyze process models, the literature offers a wide range of methods and algorithms for different purposes. These include among others model merge (e.g. La Rosa et al., 2013), model abstraction (e.g. Smirnov et al., 2012), model similarity (e.g. Dijkman, Dumas, et al., 2011), model matching (e.g. Dijkman et al., 2009), model refactoring (e.g. Dijkman, Gfeller, et al., 2011), inductive reference model construction (e.g. Scholta, 2016) and weakness detection (e.g. Delfmann and Höhenberger, 2015). Many of these topics rely on the calculation of similarities between two activities from different models.

However, a suitable similarity metric for activities in process models created with domain-specific process modeling languages is missing in the literature. Hence, this paper targets the following research aim: Development of a similarity metric for a pair of activities in different business process models that were created with the same domain-specific process modeling language. The inputs of the metric are two process model activities and the output is their similarity score.

This paper applies the design science research methodology as proposed by Peffers et al. (2007). Therefore, it is structured as follows. After performing the Problem identification in the introduction, section 2 deals with the Definition of objectives, i.e. provides requirements for the solution. In section 3, existing similarity calculations are focused in order to illustrate the innovative character of the proposed similarity metric. Section 4 presents a conceptualization of the proposed solution and therefore targets the Development phase. The Demonstration is presented in section 5. Evaluation is part of future work. Finally, a conclusion is drawn and an outlook on future work is presented in section 6.

2 Requirements

The following requirements for the metric were derived based on characteristics of domain-specific business process modeling and similarity metrics:

REQ1: Return one similarity score between 0 and 1 for a pair of activities. Since the similarity score is used to determine the similarity of two activities, it returns one single value indicating their similarity instead of two or more separated values (Richter, 1992). To ensure comparability of similarity scores, they are normalized (Han et al., 2012) and range from 0 (low similarity) to 1 (high similarity).

REQ2: Use annotated information of activities. As stated above, domain-specific process modeling languages deliver a set of predefined activities and allow for the enrichment of activities with values for predefined domain-specific properties (Becker, Probandt, et al., 2012; Pfeiffer, 2008). The metric makes use of this additional information. It does not only compare the labels of activities by using string distance measures or thesauri.

REQ3: Enable an easy comprehensibility of different degrees of relevance. Properties are not equally relevant for the similarity metric (Ehrig et al., 2007). The relevance of a property varies from domain to domain and from case to case. Therefore, domain experts are involved in the evaluation of properties with regard to their relevance. However, domain experts typically do not have comprehensive modeling expertise (Leopold, Mendling, et al., 2012). Thus, a metric is required that enables an easy understandability of its components for domain experts. Therefore, it firstly defines different degrees of relevance (categories) and secondly represents their semantics by mathematical constructs. When assigning properties to the degrees of relevance, domain experts only need to understand the textually described semantics and not the mathematical representations.

Contrasting approaches determine mathematical constructs such as weights representing the differing relevance of properties first. In order to assign weights to properties in a second step of such approaches, these constructs have to be understood by domain experts who may struggle to identify suitable semantics for the mathematical constructs. This may be indicated by questions such as “What does a weight of 0.4 mean?” and “In which cases is a property twice as relevant as another one?”.

REQ4: Take property value sets into account. In addition to single values, sets may be assigned as values to properties, e.g. an activity may process more than one data object. We refer to these sets as property value sets in the following. The metric uses a suitable comparison mechanism for sets.
**REQ5:** Consider specifics of quantitative properties. In contrast to qualitative properties, which may have different values, quantitative properties additionally represent measurable quantities (Han et al., 2012). Therefore, the similarity of quantitative properties can be quantified.

**REQ6:** Take the context of an activity into account. An activity is not only characterized by its own properties but it is also connected to other activities via control flow relations. Therefore, the surroundings of activities affect their similarity (Dijkman, Dumas, et al., 2011; van Dongen et al., 2008; Klinkmüller et al., 2014).

### 3 Related Work

Recently, a comprehensive amount of work has been published with regard to similarities between process model activities especially with the focus on process model matching which aims at identifying a set of suitable correspondences between elements of different process models (Branco et al., 2012; Dijkman et al., 2009; Klinkmüller et al., 2013; Leopold, Niepert, et al., 2012; Weidlich et al., 2010, 2013). Most of the similarity metrics are based on lexical matchings of activity labels. The lexical matching can be performed either syntactically, semantically or as combination of both (Dijkman et al., 2009). Whereas syntactic string metrics analyze the number of common substrings and symbols, semantic string metrics compare the meanings of the strings using thesauri. Some activity similarity calculations only consider the labels of activities (Dijkman et al., 2009; Klinkmüller et al., 2013). Instead, WEIDLICH ET AL. (2013) consider that activities may be further described with additional textual descriptions. Hence, they consider annotated information but do not distinguish between different properties and their degrees of relevance. BRANCO ET AL. (2012) differentiate name and type of a process model element and LEOPOLD, NIEPERT, ET AL. (2012) distinguish between an activity’s action, business object and additional fragments. However, they do not take further properties and their differing weights into account. WEIDLICH ET AL. (2010) consider that an activity may be characterized by its label, roles, input and output data and textual description. However, they integrate all terms of these annotations in one set called virtual document and thereby lose the possibility to differentiate between properties.

Also papers from other areas than process model matching propose or make use of similarity metrics (Dijkman, Dumas, et al., 2011; Dijkman, Gfeller, et al., 2011; van Dongen et al., 2008; Ehrig et al., 2007; Humm and Fengel, 2012; La Rosa et al., 2013). Dijkman, Dumas, et al. (2011) define several metrics including an attribute similarity. Since it is an unweighted average of the similarities of property values, it does not take into account the differing relevance of properties. EHRIG ET AL. (2007) distinguish between the model element types Place, Attribute, Value and Transition. Each type has different properties which are considered with different weights in their metric. However, it specifies the weights in advance and does not predefine different degrees of relevance and their semantics first.

Consequently, there are some approaches which consider annotated information in addition to the activities’ labels in order to refine the exactness of similarity metrics (REQ2). However, all these approaches do not define degrees of relevance at first (REQ3) and do not consider domain-specific business process modeling languages with their set of predefined activities and properties.

### 4 Conceptualization

The similarity of two activities is calculated by comparing their property values (REQ2). For simplicity reasons, this approach assumes that the values have been standardized beforehand (e.g. Delfmann et al., 2009). That means that no two strings may occur that refer to the same entity, e.g. “Invoice” and “Bill”.

Based on (Smirnov et al., 2011), this paper defines a business process model $m_i$ that was created with a domain-specific process modeling language as tuple $m_i = (A_i, F_i, P_i, v_i, props_i)$ with:

- $A_i$: A finite non-empty set of activities
- $F_i \subseteq A_i \times A_i$: The set of control flow relations
- $P_i$: A finite non-empty set of properties
• \( V_i \): A finite non-empty set of all occurring property values
• \( \text{props}_i : A_i \times P_i \rightarrow V_i \) is a mapping that assigns a property value to a pair of one activity and one property. If a property exists for an activity but is not filled, this function’s value is “Null”. The function has the value “Not Existing”, if an activity does not have a property.

### 4.1 Degrees of Relevance

#### 4.1.1 Concept

In order to fulfill \( \text{REQ}_2 \) and \( \text{REQ}_3 \), three degrees of relevance are defined for the incorporation of properties. In order to specify degrees of relevance, we searched for words that express importance, relevance or necessity. We identified the class of “Modal verbs”. A modal verb is an “auxiliary verb that expresses necessity or possibility. English modal verbs include must, shall, will, should, would, can, could, may, and might” (Oxford University Press, 2015). In order to avoid duplicates, we selected \text{Must}, \text{Shall} and \text{May} as degrees of relevance with the following semantics: a) \text{Must} properties must be equal when two activities are considered similar, b) \text{Shall} properties provide valuable information for the metric but different values of \text{Shall} properties do not necessarily mean that the according activities are not similar, i.e. they are less relevant than \text{Must} properties but shall be equal, c) \text{May} properties may be equal or not when two activities are considered similar, i.e. they are not relevant.

In order to represent the semantics by mathematical constructs, the similarity metric splits up the calculation into two parts: One part for \text{Must} properties and one part for \text{Shall} properties. \text{May} properties do not influence the similarity score. Like the attribute similarity of Dijkman, Dumas, et al. (2011), the score for the \text{Shall} properties is the average similarity between the property values. The score for the \text{Must} properties is calculated as the product of the similarity values between the property values indicating whether all \text{Must} properties fit or not. Since the two scores have to be integrated into one score (\( \text{REQ}_1 \)), the overall score is calculated as weighted sum of the \text{Must} and \text{Shall} score and ranges from 0 to 1. The similarity score \( \text{sim} : A_1 \times A_2 \rightarrow [0,1] \) between two activities \( \text{act}_1 \) of model \( m_1 = (A_1, F_1, P_1, V_1, \text{props}_1) \) and \( \text{act}_2 \) of model \( m_2 = (A_2, F_2, P_2, V_2, \text{props}_2) \) is defined as follows:

\[
\text{sim}(\text{act}_1, \text{act}_2) = \text{weight} \text{must} \sum_{p_k \in \text{props}_1} \text{sim}_\text{prop}(\text{act}_1, \text{act}_2, p_k) + \text{weight} \text{shall} \sum_{p_l \in \text{props}_2} \text{sim}_\text{prop}(\text{act}_1, \text{act}_2, p_l)
\]

\( \text{weight} \text{must} \) (\( \text{weight} \text{shall} \)) denotes the weight of the score of the \text{Must} (\text{Shall}) properties. \( P_{\text{act}_1, \text{act}_2}^{\text{must}} \) (\( P_{\text{act}_1, \text{act}_2}^{\text{shall}} \)) denotes the set of \text{Must} (\text{Shall}) properties for which \( \text{act}_1 \) and \( \text{act}_2 \) both have an assigned value and their values are not “Not Existing” and not “Null”. Such values can either result from an actual null value (e.g. an activity is not supported by an IT system) or a non-null value exists but it is not captured in the model (e.g. an activity is supported by an IT system but the property does not exist or the system is not modeled). Whereas the first reason influences the similarity of activities, the second reason does not necessarily indicate differences between activities. Without deeper investigation it is not possible to determine which of these two reasons is valid. Therefore, we decided to only consider properties for which both activities provide an actual value. \( \text{sim}_\text{prop}(\text{act}_1, \text{act}_2, p_k) \) denotes the similarity between \( \text{act}_1 \) and \( \text{act}_2 \) with regard to property \( p_k \), i.e. \( \text{sim}_\text{prop} : A_1 \times A_2 \times (P_1 \cap P_2) \rightarrow [0,1] \).

Hence, the two property values are compared. The calculation of \( \text{sim}_\text{prop} \) is provided in the next subsection. In order to represent whether at least one \text{Must} property does not fit, \( \text{sim} \) returns a Boolean value for \text{Must} properties indicating whether all \text{Must} properties are equal (return value 1) or not (return value 0), \( |S| \) represents the quantity of elements in a set \( S \).

If there is no \text{Shall} property for which both activities have an assigned value, i.e. \( |P_{\text{act}_1, \text{act}_2}^{\text{shall}}| = 0 \), then \( \text{sim} \) is defined as follows:

\[
\text{sim}(\text{act}_1, \text{act}_2) = \prod_{p_k \in \text{props}_1} \text{sim}_\text{prop}(\text{act}_1, \text{act}_2, p_k)
\]
To represent the semantics of the degrees of relevance, the weights are selected in a way that \( \text{weight}_{\text{must}} > \text{weight}_{\text{shall}} \) and \( \text{weight}_{\text{shall}} \neq 0 \) in order to give \textit{Must} properties more influence on the similarity than \textit{ Shall} properties. Beyond these constraints to represent semantics of the degrees of relevance, the weights can be selected depending on the user’s preferences and the individual case. In this paper, we select that \( \text{weight}_{\text{must}} = 0.75 \) and \( \text{weight}_{\text{shall}} = 0.25 \). Hence, if two activities have equal values for all \textit{Must} properties, a value of 0.75 or greater is returned. If the activities are different regarding at least one \textit{Must} property, \( \text{sim}_{\text{prop}} \) returns 0 for this property and \( \text{weight}_{\text{must}} \prod_{p_k \in \text{prop}_{\text{must}}} \text{sim}_{\text{prop}}(\text{act}_1, \text{act}_2, p_k) \) becomes 0, i.e. a value of 0.25 or less is returned for \( \text{sim} \). Consequently, based on their semantics, the metric incorporates the degrees of relevance (REQ3) in a way that \textit{Shall} properties are relevant for the metric but \textit{Must} properties are more important than \textit{Shall} properties (b). Since \textit{Must} properties must be equal, it is clearly indicated by returning a value of \( \text{weight}_{\text{shall}} \) or less if at least one \textit{Must} property does not fit (a). Properties belonging to the class \textit{May} are not considered by the metric (c).

4.1.2 Exemplary Application

We exemplarily defined degrees of relevance for the PICTURE method (Becker, Algermissen, et al., 2012). The PICTURE method is a domain-specific business process modeling language that is dedicated to public administrations and widely applied in the German governmental practice (Becker, Algermissen, et al., 2012; Detemple et al., 2014). It provides 24 domain-specific types of activities such as “sign document” and allows the enrichment of activities with properties such as processed documents and external participants. The degrees of relevance of the properties for the similarity metric were derived based on practical experiences with the PICTURE method and discussions with modeling experts. The following classification is made without taking purposes into account. Thus, the classification is a general suggestion which is to be adapted to individual cases. Table 1 contains the degrees of relevance for exemplary properties.

<table>
<thead>
<tr>
<th>Must</th>
<th>Shall</th>
<th>May</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Type of activity</td>
<td>• Software</td>
<td>• Name</td>
</tr>
<tr>
<td>• Documents</td>
<td>• Executing role</td>
<td>• Description</td>
</tr>
<tr>
<td>• Sender/Receiver of</td>
<td>• Input/Output channels</td>
<td>• Remarks and improvements suggestions</td>
</tr>
<tr>
<td>documents</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1. Degrees of Relevance of Exemplary Properties

When two properties are similar, their types must be equal since the type defines the activity that is performed. Additionally, properties related to documents are ranked as \textit{Must} as they describe the affected business objects and drive processes in public administrations (Klischewski, 2006). Similarly important are senders and receivers of documents since they indicate the addresssees of administrative services. As mentioned above, \textit{Must} properties can either be equal or not, i.e. \( \text{sim}_{\text{prop}} \) either returns 1 or 0. Since this is a qualitative comparison (Ott and Longnecker, 2010), it is not well applicable to quantitative properties. Therefore, quantitative properties are not ranked as \textit{Must} properties but can be \textit{Shall} or \textit{May} at most. A quantitative distance is to be calculated in \textit{Shall} cases. The percentages regarding the input and output channels such as “Telephone”, “In person” or “Mail” are rated as \textit{Shall} since a high difference may indicate different activities but the values do not have to be completely equal. Moreover, “Used software” and “Executing role” are classified as \textit{Shall} properties since they may point to similar activities if two values of these properties are equal. However, different values of these properties do not generally mean dissimilarity regarding the activities. Thus, they are rated as \textit{Shall}. Besides, two entire texts can hardly be completely equal. In contrast to quantitative properties, a quantitative distance cannot be computed for textual properties. Thus, texts cannot belong to the classes of \textit{Shall} or \textit{Must} properties but can only be \textit{May}. The properties “Name”, “Description” and “Remarks and improvements suggestions” are ranked accordingly.
4.2 Property Value Sets

\( \text{sim}\_\text{prop} \) compares the contents of two sets in order to meet \( \text{REQ4} \). A well-known and simple measure is the Jaccard index (Jaccard, 1901). It divides the number of elements two sets C and D have in common by the size of the union of these two sets. It is defined as follows (Naumann and Herschel 2010, p. 24):

\[
\text{Jacc}(C, D) = \frac{|C \cap D|}{|C \cup D|}
\]

\( \text{sim}\_\text{prop} \) can be defined as Jaccard index since the property values are standardized:

\[
\text{sim}\_\text{prop}(act_1, act_2, p_k) = \frac{|\text{props}_1(\text{act}_1, p_k) \cap \text{props}_2(\text{act}_2, p_k)|}{|\text{props}_1(\text{act}_1, p_k) \cup \text{props}_2(\text{act}_2, p_k)|}
\]

This way of calculating \( \text{sim}\_\text{prop} \) considers the quantity of matching elements of the property value sets. Hence, if 5 out of 10 elements match, then a higher value is returned in comparison to cases where only 2 out of 10 elements are considered equal. However, using this way of calculating \( \text{sim}\_\text{prop} \) may lead to an inappropriate representation of the semantics of the degrees of relevance (\( \text{REQ3} \)). If \( \text{sim} \) returns a score of 0.2 based on this way of calculating \( \text{sim}\_\text{prop} \), the score may result from different situations. The first option is that at least one \( \text{Must} \) property does not fit and this property receives a value of 0 for \( \text{sim}\_\text{prop} \). The returned value of 0.2 for \( \text{sim} \) is determined by the \( \text{Shall} \) score. A second option is that all \( \text{Must} \) properties are equal except for one property which receives the score 0.2 for \( \text{sim}\_\text{prop} \) and all \( \text{Shall} \) property values are completely different. This ambiguity would violate semantic a) since it cannot be recognized by the value of \( \text{sim} \) whether all \( \text{Must} \) properties fit or not. In order to ensure the proper representation of semantic a), the second option has to be avoided. Therefore, the calculation of \( \text{sim}\_\text{prop} \) stated above is used for \( \text{Shall} \) properties and an alternative calculation is used for \( \text{Must} \) properties. This calculation of \( \text{sim}\_\text{prop} \) returns 1 (at least one value fits) or 0 (no value fits). Therefore, \( \text{sim}\_\text{prop} \) is defined for a \( \text{Must} \) property \( p_k \) as follows:

\[
\text{sim}\_\text{prop}(act_1, act_2, p_k) = \begin{cases} 
1 & \text{if } |\text{props}_1(\text{act}_1, p_k) \cap \text{props}_2(\text{act}_2, p_k)| \geq 1 \\
0 & \text{if } |\text{props}_1(\text{act}_1, p_k) \cap \text{props}_2(\text{act}_2, p_k)| = 0 
\end{cases}
\]

A disadvantage of this calculation is that the quantity of equal elements is not taken into account. However, ambiguity is avoided and the user can be sure that all \( \text{Must} \) properties at least partially fit if \( \text{sim} \) returns a value of \( \text{weight}_{\text{must}} = 0.75 \) or higher.

4.3 Quantitative Properties and Surroundings

In order to fulfill \( \text{REQ5} \), the similarity between two activities \( \text{act}_1 \) and \( \text{act}_2 \) regarding a quantitative property \( p_k \) is defined as follows:

\[
\text{sim}\_\text{prop}(act_1, act_2, p_k) = \max(0, 1 - \frac{|\text{props}_1(\text{act}_1, p_k) - \text{props}_2(\text{act}_2, p_k)|}{\min(|\text{props}_1(\text{act}_1, p_k)|, |\text{props}_2(\text{act}_2, p_k)|)})
\]

In this case, \( |x - y| \) denotes the absolute distance between two numbers \( x \) and \( y \) and \( |x| \) is defined as the absolute value of a number \( x \). The absolute distance is calculated in the numerator. The denominator sets this figure in relation to the minimal number. Thus, e.g. a distance of 5 is more relevant for values around 10 than 100. 0 is returned if the distance is equal to or greater than the minimal property value. If the minimal value is 0, \( \max(0, 1 - |\text{props}_1(\text{act}_1, p_k) - \text{props}_2(\text{act}_2, p_k)|) \) is the result.

To meet \( \text{REQ6} \), the properties “Preceding activities” and “Succeeding activities” are classified as \( \text{Shall} \) properties. When calculating \( \text{sim}\_\text{prop} \) for these properties, directly adjacent activities of \( \text{act}_1 \) and \( \text{act}_2 \) are considered. An alternative option is to consider a wider surrounding, i.e. adjacent activities of adjacent activities and so forth. However, in the proposed similarity metric only directly adjacent activities are analyzed since other activities are indirectly connected to \( \text{act}_1 \) and \( \text{act}_2 \). Therefore, based on the definition of a process model at the beginning of section 4 they form no properties of \( \text{act}_1 \) and \( \text{act}_2 \) since a control flow relation connects two activities.
To incorporate “Preceding activities”, the similarity scores $sim_{adjacent}$ between all preceding activities of $act_1$ and all preceding activities of $act_2$ are calculated. The same applies to the succeeding activities for “Succeeding activities”. $sim_{adjacent}$ is calculated in the same way as $sim$. However, to avoid an infinite loop due to recursion, the properties “Preceding activities” and “Succeeding activities” are ignored when calculating $sim_{adjacent}$. A mapping between adjacent activities is simulated following a greedy procedure. The pair of preceding or succeeding activities with the highest value for $sim_{adjacent}$ is mapped until $act_1$ or $act_2$ does not have an unmapped adjacent activity or the threshold $weight_{must} = 0.75$ is reached. Then, the values of $sim_{prop}$ between $act_1$ and $act_2$ regarding “Preceding activities” and “Succeeding activities” are calculated as average scores considering each pair of mapped adjacent activities of $act_1$ and $act_2$ as 1 and each pair of unmapped activities as 0.

5 Demonstration

The similarity metric is demonstrated with the fictional PICTURE models A and B in Figure 1. The types of activities are visualized as proposed in (Algermissen et al., 2012; Becker, Algermissen, et al., 2012) and are mentioned in italic letters below the symbols. As the type of activity is a Must property, all activities with different types have a similarity score of 0.25 at most. Table 2 displays all similarity scores greater than 0.25. The second column contains the similarity score used for calculating “Preceding activities” and “Succeeding activities”. The third column provides the actual similarity score.

![Figure 1. Two Exemplary Process Models A (left) and B (right)](image)

The Must properties “Paper-based documents”, “Sender of documents (external)” and type of An1 and Bn1 are equal. The values of quantitative Shall properties “Mail” and “In person” differ leading to $sim_{prop}(An1,Bn1, Mail) = 1 - \frac{1 - 0.9}{0.9} = 1 - 0.1 = 0.89$ and $sim_{prop}(An1,Bn1, In person) = 1 - |0 - 0.1| = 0.9$. Their adjacent activities can be mapped since An1 and Bn1 are the start activities of the processes (“Preceding activities”) and An2 and Bn2 have a score for $sim_{adjacent}$ that is greater than $weight_{must} = 0.75$ (“Succeeding activities”). The overall similarity score is $sim(An1,Bn1) \approx 0.99$. An2 and Bn2 are equal regarding their Must properties and their preceding activities can be
mapped. However, their succeeding activities differ. Hence, \( \text{sim}(An2, Bn2) \approx 0.88 \). Since “Confirmation” occurs in both document-related property value sets and their types are equal, the values of the Must properties of \( An3 \) and \( Bn4 \) are equal. The property “Used software” is not considered since only \( Bn4 \) provides a value. Their adjacent activities are different which leads to \( \text{sim}(An3, Bn4) = 0.75 \).

Since the Must properties fit, \( \text{sim}_{prop}(An4, Bn3, \text{Software (input system)}) = \frac{1}{2} \) for the Shall property “Software (input system)” and their adjacent activities differ, the similarity score 0.79 is assigned to \( An4 \) and \( Bn3 \).

<table>
<thead>
<tr>
<th></th>
<th>( \text{sim}_{\text{adjacent}} )</th>
<th>( \text{sim} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( An1, Bn1 )</td>
<td>( 0.75 \cdot 1 + 0.25 \cdot \frac{0.89 + 0.9}{2} \approx 0.97 )</td>
<td>( 0.75 \cdot 1 + 0.25 \cdot \frac{0.89 + 0.9 + 1 + 1}{4} \approx 0.99 )</td>
</tr>
<tr>
<td>( An2, Bn2 )</td>
<td>1</td>
<td>( 0.75 \cdot 1 + 0.25 \cdot \frac{1 + 0}{2} \approx 0.88 )</td>
</tr>
<tr>
<td>( An3, Bn4 )</td>
<td>1</td>
<td>( 0.75 \cdot 1 + 0.25 \cdot \frac{0 + 0}{2} = 0.75 )</td>
</tr>
<tr>
<td>( An4, Bn3 )</td>
<td>( 0.75 \cdot 1 + 0.25 \cdot 0.5 \approx 0.88 )</td>
<td>( 0.75 \cdot 1 + 0.25 \cdot \frac{0.5 + 0 + 0}{3} \approx 0.79 )</td>
</tr>
<tr>
<td>( An5, Bn5 )</td>
<td>1</td>
<td>( 0.75 \cdot 1 + 0.25 \cdot \frac{0 + 1}{2} \approx 0.88 )</td>
</tr>
</tbody>
</table>

Table 2. Similarity Scores between Activities of A and B

6 Conclusion and Outlook

The contribution of this paper is to propose a similarity metric between two activities in process models created with the same domain-specific process modeling language. The metric returns a similarity score between 0 and 1 (REQ1), uses annotated information (REQ2), differentiates degrees of relevance for properties (REQ3), considers characteristics of property value sets (REQ4) and quantifiable properties (REQ5) and analyzes the surroundings of activities (REQ6). The approach relies on a prior standardization of property values. Besides, the properties have to be assigned to degrees of relevance. Although an assignment from previous cases may be reused, both tasks are time-consuming. Hence, it has to be decided in each case individually whether the usage of the proposed metric is worth those efforts. Moreover, the metric only returns meaningful results if the activities that are inputted to the similarity metric provide values for the same properties. Otherwise, only their types can be compared by the metric.

Future research requires the technical implementation of the similarity metric and its evaluation. The evaluation is going to include a comparison to existing approaches using PICTURE process modeling languages. We will apply a manual classification constructed by domain and modeling experts to define correct similarity values. These values serve as basis for the comparison to other approaches to measure accuracy.

Additionally, we consider to incorporate more sophisticated calculations, for instance based on causal footprints (van Dongen et al., 2006) for surroundings of activities and min-max normalization (Han et al., 2012) for quantifiable properties. In case of the PICTURE method, the assignment of the properties to the degrees of relevance can be validated using a survey with experts from the governmental practice. Since the requirements for the solution are sufficiently generic, we aim at investigating its applicability to general-purpose business process modeling languages such as BPMN. These languages also provide different properties for process model activities such as their label and information on IT systems and roles. Therefore, the concepts of the proposed similarity metric may be transferred to these languages. In the course of the evaluation, a comparison of the proposed metric to existing similarity metrics for general-purpose process modeling languages is possible.

Acknowledgment

I would like to thank my colleagues – especially Michael Räckers and Patrick Delfmann – for their support during the creation of this paper.
References


Scholta / Similarity Metric for Process Model Activities


