An Improved Structure-based Approach to Measure Similarity of Business Process Models

Jimin Ling, Li Zhang, Qi Feng
Software Engineering Institute, School of Computer Science and Engineering
Beihang University (BUAA), Beijing, China
lingjin@cse.buaa.edu.cn, lily@buaa.edu.cn, miffy_feng513@cse.buaa.edu.cn

Abstract—It is common for large enterprises or organizations to maintain repositories of process models. This paper focuses on how to measure the similarity and construct matching relations more effectively between process models. To resolve exponential time complexity to match node compositions, we proposed a graph-edit distance similarity metric based on SESE process fragments, and a greedy algorithm is employed to construct the optimal matching relations. Then a method to construct matching relations of fragments based on process structure tree is proposed. Finally, a comparative experiment based on real-world process models from BPM AI repository is conducted to evaluate the effectiveness and efficiency of our approach.

Keywords - Business process management; Process similarity; Process model matching; Process fragment

I. INTRODUCTION

Various techniques have been used to manage the large collections of process models. The model search technique returns the models from the collection that are most similar to a specified input process model. The similarity measurement between two process models is the key basis of this model search technique. Besides, matching relations between the process activities need to be constructed when organizations want to resolve the differences between their own processes and industry-wide standards. In short, this paper focuses on how to measure the similarity and construct matching relations more effectively between two process models.

This topic has been studied by many researchers in recent years. They always build matching relations between nodes in process models first and then evaluate the global similarity of two models in a structural [1][2][3][4] or behavioral way [1][5][6]. The similarity evaluation is based on node matching between two models, either by unique activity name or lexical similarity between node labels. However, one business logic can be realized as different amount of activities or even different structures by various modelers. Therefore, it is not enough to only consider the node matches, matching relations between node compositions need to be taken into account when the similarity is measured [7].

It has exponential time complexity to construct appropriate matching relations between node compositions and some arbitrary compositions do not make a sense. Consider the complete and independent business logic of single-entry-single-exit (SESE) process fragment [8][9], we proposed a graph-edit distance similarity metric based on the fragments, and a greedy algorithm is employed to compute the similarity value and construct the optimal matching relations. As a matching relation candidate set of fragments is required for similarity evaluation, we present a method to further construct matching relations of fragments by process structure tree. A comparative experiment is performed to evaluate the approach in this paper. The result shows that our approach have a positive impact on the accuracy of similarity evaluation comparing with current node matching based methods, and the efficiency of our technique is acceptable.

The rest of the paper is organized as follows. Section II introduces some basic concepts of business process model and process fragment. In section III we illustrate the key approach to measure the similarity of process models based on process fragments. Section IV presents how to construct matching relations between process fragments. Section V provides an experimental evaluation of our technique. Section VI discusses related work of our research. Section VII concludes the article and presents the future work.

II. PRELIMINARIES

This section introduces the notion of process model graphs (PMG) and then we illustrate the key concept of process fragments which is used in this paper.

A. Business Process Model

In this paper, we try to apply our method to various kinds of notations, so we will illustrate our method based on process model graph (PMG) rather than a specific modeling language. A PMG is simply a graph that captures nodes and edges whose properties such as names, types or resources used are treated as attributes of them. We can define the PMG and some other related concepts formally as follows.

Definition 1 (Process Model Graph). A process model graph (PMG) is a tuple \((N, E, T, \Omega, \alpha)\), in which:
- \(N\) is a set of nodes;
- \(E \subseteq N \times N\) is a set of directed edges;
- \(T\) is a set of attribute names, e.g. TYPE, LABEL, RESOURCE, INPUT, OUTPUT, etc.;
- \(\Omega\) is a set of text string values;
- \(\alpha: (N \cup E) \rightarrow (T \rightarrow \Omega)\) is a function that maps nodes or edges to attributes, where an attribute is a mapping from an attribute name to a text string value.
To limit the scope, we assume that the process models are block-structured. A process model is block-structured if the sequences, branches, and loops are represented as blocks with well-defined start and end nodes. The block-structured process models are more understandable for users with less possibility of errors than the non-block-structured models. The proportion of block-structured models in a model repository is always very high and a non-block-structured model can be transformed to a block-structured one in most cases [10].

B. Process Fragment

A block-structured process model can be decomposing hierarchically to a set of process fragments. We will introduce the single-entry-single-exit (SESE) process fragment [8][9] which is a basic concept of our approach.

Definition 2 (SESE process fragment). A node \( x \) is said to \textit{dominate} node \( y \) in a directed graph if every path from start to \( y \) includes \( x \). A node \( x \) is said to \textit{postdominate} a node \( y \) if every path from \( y \) to end includes \( x \). A SESE process fragment (process fragment for short) in a process graph \( G \) is an ordered edge pair \((a, b)\) of distinct control flow edges \( a \) and \( b \) where:

- \( a \) dominates \( b \),
- \( b \) postdominates \( a \), and
- every cycle containing \( a \) and contains \( b \) and vice versa.

We refer to \( a \) as the entry edge and \( b \) as the exit edge of the process fragment. The processes in Figure 1 gives an example of the dividing result of process fragments, and the example process is illustrated in Figure 2. Note that a node \( c \) is a skipped node if there are fragments \( X, Y, Z \) with \( X \) and \( Y \) are in sequence, \( F = X \cup Y \), and \( F \) and \( Z \) are in sequence; otherwise \( F \) is called canonical fragment.

III. FRAGMENT-BASED STRUCTURAL SIMILARITY ANALYSIS

This section deals with that given two process models and a candidate set of matching relations between their fragments, how we can evaluate the similarity value of these two models. The method to construct matching relationship between process fragments will be illustrated later in the next section.

We use the concept of graph-edit distance [11] to evaluate the global matching score of two process models. The graph-edit distance between two graphs is the minimal number of edit operations that is necessary to get from one graph to the other. The existing graph-edit similarity metric do NOT support the consideration of matching relations between process fragments, so we extend the original definition based on the above edit operations we defined.

Definition 3 (Extended graph-edit distance similarity). Let \( G_1 = (N_1, E_1, T_1, \Omega_1, \alpha_1) \) and \( G_2 = (N_2, E_2, T_2, \Omega_2, \alpha_2) \) are two PMGs and let \( M \) be the subset of fragment matching relations. Let \( P_1 \subseteq G_1 \) be a process fragment in \( G_1 \), and \( n_1, e_1 \) be a node and an edge in fragment \( P_1 \), then \( P_1 \) is a substituted fragment (\( n_1 \) and \( e_1 \) is substituted node and substituted edge respectively) iff \( \exists P_2 \subseteq G_2, M(P_1) = P_2 \). The set of nodes and edges in fragment \( P \) is denoted as \( N^P \) and \( E^P \) respectively. A node is a skipped node iff it’s not a substituted node. Let process fragments \( P_{11}, P_{12} \subseteq G_1; P_{21}, P_{22} \subseteq G_2 \), and \( e_1 \in E_1 \) is the link between \( P_{11} \) and \( P_{12} \), i.e. be the exit edge of \( P_{11} \) and also the entry edge of \( P_{12} \), denoted as \( (P_{11}, P_{12}) \). \( e_1 \) is a matched edge iff \( \exists (P_{21}, P_{22}) \subseteq E_2, M(P_{11}) = P_{21} \land M(P_{12}) = P_{22} \). Note that we consider that the matched edges also exist in the other model. Any other edge except for substituted edge and matched edge is called skipped edge. Let \( \text{subn} \), \( \text{skipn} \), \( \text{sube} \) and \( \text{skipe} \) be the sets of each kinds of nodes and edges, and \( \omega_{\text{subn}} \), \( \omega_{\text{skipn}} \), \( \omega_{\text{sube}} \) and \( \omega_{\text{skipe}} \) be the weights of substituted nodes, skipped nodes, substituted edges and skipped edges respectively. The extended graph-edit distance similarity of \( G_1 \) and \( G_2 \) induced by the mapping \( M \) is:

\[
\text{EGSim}(G_1, G_2, M) = 1.0 - \frac{\sum_{\text{subn}} \omega_{\text{subn}} + \sum_{\text{skipn}} \omega_{\text{skipn}} + \sum_{\text{sube}} \omega_{\text{sube}} + \sum_{\text{skipe}} \omega_{\text{skipe}}}{\sum_{\text{subn}} + \sum_{\text{skipn}} + \sum_{\text{sube}} + \sum_{\text{skipe}}}
\]

\[
\text{subn} = \frac{\sum_{P_{11}, P_{12}} \mid 1.0 - \text{Sim}(P_{11}, P_{12}) \mid (|E_1| + |N_1|)}{|\text{subn}|} \quad \text{skipn} = \frac{|\text{skipn}|}{|N_1| + |N_2|}
\]

\[
\text{skipe} = \frac{|\text{skipe}|}{|E_1| + |E_2|} \quad \text{sube} = \frac{\sum_{P_{11}, P_{12}} \mid 1.0 - \text{Sim}(P_{11}, P_{12}) \mid (|E_1| + |N_1|)}{|\text{sube}|}
\]

Here we illustrate the basic procedure based on a greedy strategy to compute the maximum metric score by selecting an optimal subset. Initially, all candidate matching relations are added to the CanPairs and the optimal set \( M \) is empty. In each
iteration, a new EGSim is computed for adding every pair in CanPairs into M to see which pair lead to a highest EGSim. This pair is added to the optimal mapping M and any other pair overlapping with this pair is removed from the CanPairs. The algorithm terminates when there is no group pair in CanPairs that can increase the metric score EGSim any more.

IV. CONSTRUCTING MATCHES OF PROCESS FRAGMENTS

Since there also exists some complex matching relations (i.e. matching between more than one node in each model) between process models, these matching relations need to be identified based on the node matches. Details about similarity metric between nodes can be achieved in [1].

It is clear that not all types of compositions of nodes have a sense, so we propose the prerequisite of grouping elements based on the concept of single-entry-single-exit (SESE) process fragments [8][9]. A canonical process fragment or consecutive canonical fragments may express complete and independent business logic to a great extent. Therefore, all process fragments are regarded as candidates of node combination. Considering the process in Figure 1, the candidate set of fragments includes: \{J\}, \{K\}, \{L\}, \{M\}, \{P\}, \{Q\}, \{R\}, \{I\}, \{N\}, \{I, R\}, \{R, N\}, \{I, R, N\}.

Based on the candidate set obtained by the above condition, we need a criterion of whether there exists a matching relation between two fragments in distinct process models. The matching relations between leaf fragments have been achieved in node matching relations set. For these non-leaf fragments, we hold that the similarity of two fragments should at least higher than the similarity of matching relations between their children fragments. Besides, to reduce searching space, one basic prerequisite is proposed, i.e. there should be at least one matching relation among their containing fragments.

The criterion requires a definition of similarity evaluation between two process fragments. Referring to node similarity, the similarity between groups, based on their attributes and the context of adjacent nodes, can be defined analogously. The major challenge of attribute similarity part is that different attributes may have different composing policies, e.g. the attribute LABEL of a group can be simply merged by the name label of the containing nodes while the merging of attribute INPUT or OUTPUT should ignore the input or output data produced or consumed inside the group itself.

Here we can summarize the algorithm of discovering fragment matching relations. First of all, the process structure trees (PST) of two process models are constructed by the cyclic equivalent algorithm [9], while the detail of this algorithm is not presented here. Then we can get the candidate set of fragments for each process model in a bottom-up order of the tree. Finally, the fragments in two candidate sets are checked one by one according to their sequence, and if the pair satisfies the criterion of building matches, the fragment pair with their similarity value is added to the fragment matching relations set.

V. EXPERIMENTAL EVALUATION

We evaluate our approach by the real-world process models from BPM Academic Initiative (BPM AI) [12] which is a joint effort of academic and industry partners that offers a process modeling platform for teaching and research purposes. We selected 40 pairs of process models captured in BPMN and these model pairs are in various degree of similarity. We obtained the similarity assessment value of each process model pair by using a questionnaire that was distributed to 10 academic process modelers, including researchers and post-graduate students in the realm of process modeling. In the questionnaire, the modelers were asked to decide the similar degree from score 1 to 6. We obtained the manual similarity assessment value by a weighted average of the scores provided by the 10 participants according to their modeling experience.

We create two different analyzers to conduct a comparative experiment. The Analyzer A measures graph-edit distance similarity based on node matching relations which is the method in [1][3]. The Analyzer B applies our fragment-based similarity measuring method. Our experimental evaluation is conducted by computing the similarity value of each process model pair using these two analyzers we defined and then their computing efficiency and correlation with the manual analysis results can be obtained. We will apply the weight parameters achieved by our earlier research work [13] to these analyzers.

<table>
<thead>
<tr>
<th>Analyzer name</th>
<th>Correlation coefficient</th>
<th>Time cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analyzer A</td>
<td>0.805</td>
<td>628 ms</td>
</tr>
<tr>
<td>Analyzer B</td>
<td>0.868</td>
<td>1817 ms</td>
</tr>
</tbody>
</table>

The results are shown in Table 1. Note that all the I/O operations (e.g. reading process models) are excluded from the time cost statistics. From the results we can see that the Analyzer B which fully applied our approach shows a better correlation coefficient result and the time efficiency of the method is also acceptable, i.e. about 45 milliseconds for each model pair in average. Figure 3 shows the correlation between the similarity degree (using the optimal correlation of analyzer B) and the similarity assessment as obtained from the questionnaire.

VI. RELATED WORK

Our approach is related to two closely interrelated topics in the research domain of managing large collections of business process models [7], i.e. business process model alignment and business process similarity search.
The business process model alignment determines which elements in one business process model correspond to which elements in another [14]. Three approaches based on lexical or graph matching are compared in [3], and the result showed that a greedy graph matching technique produces the best effectiveness. The ICoP framework [2] enables the optional creation of matchers from the reusable components which can support complex matching detecting, but the algorithms of the components are not illustrated clearly. Our earlier research work [13] proposes a matching technique to support fast detecting complex matches. Our fragment-based method could probably improve these techniques.

Business process similarity search is another closely related topic. A previous paper [15] of our research group utilizes the idea of similarity propagation to measure the process similarity, but the method is restricted to a specific modeling language and only node matches are supported. Remco Dijkman etc. presented and evaluated three similarity metrics and the result showed that the structural similarity slightly outperforming the presented and evaluated three similarity metrics and the result shows that the greedy algorithm produces a better result with best efficiency. Some novel efficient algorithms for similarity search have been proposed recently. There is an initiative to conduct similarity measures based on a tree-based index organized structure of process models in a collection [17] and clustering techniques were used to group similar process models in [18] to enable the comparison of process clusters rather than each individual process in isolation.

VII. CONCLUSION

Various techniques have been used to manage the large collections of process models. This paper focuses on the problem of how to measure the similarity and construct matching relations more effectively between two process models. To resolve the problem of exponential complexity to construct appropriate matching relations between node compositions, we proposed a graph-edit distance similarity metric based on the SESE process fragments, and a greedy algorithm is employed to compute the similarity value and construct the optimal matching relations. As a matching relation candidate set of fragments is required for similarity evaluation, we present a method to construct matching relations of fragments by process structure tree. A comparative experiment is performed to evaluate the approach in this paper. The result shows that our approach have a positive impact on the accuracy of similarity evaluation comparing with current methods based on node matching, and the efficiency of our technique is acceptable.

In future research, we aim to improve the effectiveness of our approach by conducting a more comprehensive experiment evaluation by huge amount of process models in various model repositories. Secondly, the further accuracy and efficiency improvement for the existing structural process similarity evaluation is another direction.

VIII. ACKNOWLEDGMENTS

This work is supported by Chinese National Natural Science Foundation (Grant No. 61170087 and Grant No. 61370058). Thanks for the valuable comments of Professor Xavier Blanc from University Bordeaux 1.

REFERENCES