Heuristic Methods for Searching and Clustering Hierarchical Workflows

Michael Kastner, Mohamed Wagdy Saleh, Stefan Wagner, Michael Affenzeller, Witold Jacak

Heuristic and Evolutionary Algorithms Laboratory
School of Informatics, Communications and Media - Hagenberg
Upper Austria University of Applied Sciences
Softwarepark 11, A-4232 Hagenberg, Austria
mkastner@fh-hagenberg.at, mohamed-wagdy.saleh@isi-hagenberg.at,
{swagner,maffenze,jacak}@fh-hagenberg.at

Abstract. Workflows are used nowadays in different areas of application. Emergency services are one of these areas where explicitly defined workflows help to increase traceability, control, efficiency, and quality of rescue missions. In this paper, we introduce a generic workflow model for describing fire fighting operations in different scenarios. Based on this model we also describe heuristics for calculating the similarity of workflows which can be used for searching and clustering.

1 Introduction

In the last years, process models were frequently used to describe workflows of business processes in the economic field [10, 7]. Following the idea of process-centric management, the explicit definition of workflows helps to increase traceability, control, efficiency, and quality. However, describing tasks by using formal process models is not restricted to the area of business process modeling. Especially in the area of emergency services, explicit process definitions are also of major importance to reduce the risk of human errors and to improve effectiveness [9].

In this paper, we focus on the scenario of a decentralized fire fighting organization. In this organization, hierarchical workflows (i.e. workflows consisting of actions, transitions, and sub-workflows) are used to describe actions to be executed in different kinds of emergency situations. As emergency situations are hardly ever exactly the same, most workflows have to take specific characteristics of a concrete emergency scenario into account. Therefore, individual workflows are defined by many users and a priori generalization is hard to achieve.

In order to keep a large number of workflows manageable, two strategies may be applied: First, performing a fuzzy search on all existing workflows is helpful for defining a new workflow. By this means, the user is enabled to check, if similar workflows have been created already, and to compare and adapt the new workflow iteratively. Second, clustering algorithms can be used to group

all workflows in order to identify which workflows describe similar tasks, which workflows can act as representatives for a whole cluster, and to perform an a posteriori simplification or standardization.

As a foundation for both approaches (searching and clustering), we introduce a generalized workflow model. Based on the work of Jung and Bae [8], we describe several heuristics for calculating the similarity of workflows on the semantic (actions) as well as on the structural (transitions) level. Furthermore, we develop a new method of combining different similarity measurements at a time. This approach allows us to take advantage of the individual characteristics of the proposed measurements and to apply them in combination in order to achieve satisfying clustering results. Finally, the suitability of this approach is evaluated by clustering artificially generated sets of workflows using classical clustering algorithms (k-Means, DBSCAN, and Expectation Maximization).

2 Workflow Model

In cooperation with fire fighters of different fire departments we defined a generalized workflow model. Each activity/task carried out by a fire fighter in an operation is called an action. Each action contains an ID, a name, a status, a description, and a set of keywords. The concept of keywords was introduced to facilitate the comparison of actions. Similar keywords in multiple actions indicate that these actions might deal with a similar topic. The status of an action can either be *to do*, *in progress*, or *done* and is set by a fire fighter when carrying out a workflow. Furthermore, an additional status *irrelevant* was also defined to mark actions which are not applicable in a concrete emergency situation.

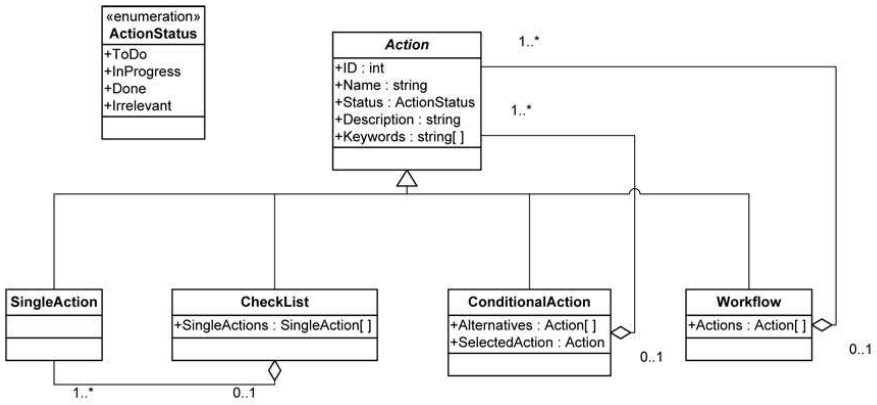

Fig. 1. UML chart of the workflow model

In order to facilitate different combinations of actions, four action subtypes were identified: The first type is a *single action* which represents a single and

atomic activity. The second type is called *checklist* and describes a collection of single actions that can be carried out in any order. For modeling multiple alternatives, the type *conditional action* was introduced which contains multiple actions of any type from which one action has to be chosen. Finally, the last type called *workflow* contains arbitrary many actions of any type which have to be executed in a predefined linear sequence. Obviously, as a whole workflow is considered as a type of action, workflows can be structured hierarchically. An UML representation of the workflow model is shown in Figure 1.

3 Similarity Measures

Based on the workflow model described above, we introduce several heuristics to calculate the similarity of actions in this section. At first, basic similarity measures are presented that turned out to be suitable for comparing workflows in the domain of fire fighting operations. Then it is described how these similarity measures can be combined in order to apply different similarity measures at a time depending on the types of actions that are compared. This approach allows the flexible definition of complex comparison heuristics that benefit from the individual characteristics of the basic measurement values.

3.1 Activity Similarity Measure (ASM)

As presented by Jung and Bae in [8], the activity similarity measure calculates how many activities are commonly shared between two actions by using the Cosine measure. It is assumed that the degree of similarity of two actions increases, as the number of shared activities increases. The transitions between activities are not taken into account. Furthermore, the ASM only considers top level actions; actions contained on lower hierarchy levels as for example the actions of a sub-workflow are not compared.

3.2 Transition Similarity Measure (TSM)

The transition similarity measure [8] is a more strict measure than the ASM, as it considers the transitions of activities in each action (i.e. the sequence in which activities are performed). It is assumed that the degree of similarity of two actions only increases, if activities in both actions are carried out in the same order. For both actions a transition vector is calculated which indicates each pairwise sequence of activities included in an action. Then the similarity of these transition vectors is again calculated by using the Cosine measure. Similarly to the ASM, also the TSM only considers activities on the top level of each action.

3.3 Single Action Activity Similarity Measure (SAASM)

As the two similarity measures described above (ASM and TSM) do not take multiple hierarchy levels into account, we propose another similarity measure

called single action activity similarity measure. It is assumed that the similarity of actions can be described by the similarity of all contained single actions regardless of the hierarchy level on which they occur. Therefore, all single actions of both actions are compared pairwise and the maximum similarity value for each single action is calculated. Then all maximum similarity values are averaged to get the similarity value for the two actions.

3.4 Single Action Transition Similarity Measure (SATSM)

The single action transition similarity measure is based on the SAASM but additionally considers the sequence in which single actions appear on all hierarchy levels of an action. It calculates the average similarity of all transitions contained in two actions, whereby the similarity of a transition is defined as the similarity of its source and destination single action.

3.5 Keyword Similarity Measure (KSM)

The keyword similarity measure is a simple way of measuring the similarity of two actions just by considering their keywords. It is defined as the relative number of identical keywords contained in both actions. For example, if three keywords out of seven are contained in both actions, the KSM gives a similarity value of 3/7.

3.6 Combination of Similarity Measures

The basic similarity measures described above can be applied to calculate the similarity of arbitrary actions contained in a workflow. However, it is reasonable to apply different similarity measures depending on the type of actions that are compared. For example, for comparing two checklists ASM can be applied as the single actions contained in a checklist can be executed in any order. Consequently, the sequence of these single actions (i.e. the transitions) do not have to be considered. In contrast, for comparing conditional actions or sub-workflows a similarity measure that also takes the sequence of actions into account might be more reasonable.

Table 1. Combination matrix of similarity measures

	Single Action	Checklist	Conditional Action	Workflow
Single Action	KSM	ASM	ASM	ASM
Checklist		ASM	SAASM	SAASM
Conditional Action			TSM	SATSM
Workflow				SATSM

Therefore, a specific combination of similarity measures can be defined in a matrix as shown exemplarily in Table 1. When comparing the actions contained in a workflow, the respective similarity measure is selected depending on the compared actions. This approach allows for a high degree of flexibility. By this means, workflow comparison can be easily tuned according to the structure and content of workflows in order to reflect different application scenarios.

4 Experiments

4.1 Setup

In order to evaluate the similarity measures proposed above with respect to their suitability for clustering workflows, we applied them on a set of artificially generated workflows. The creation of these benchmark workflows was performed in two steps: At first, 49 workflows were randomly generated. Thereby, for each workflow the maximum depth was randomly chosen between 3 and 6 and the maximum number of child actions per hierarchy level was chosen randomly between 3 and 10. In the second step, 8 workflows were randomly chosen out of the 49 generated workflows. For each of these selected workflows, 10 new workflows were created by applying few modifications. These modifications were done either by changing the keywords, adding additional child actions, swapping the order of child actions, or removing child actions. As a result, we obtained a set of 88 workflows representing 8 clusters, each containing 11 workflows. The remaining 41 workflows generated in the first step were kept as noise. This led to a total of 129 workflows. For all these workflows the pairwise similarity was calculated, resulting in a matrix of 16641 similarity values, where each line represents the similarities of a workflow to all other workflows.

Clustering of the similarity data set was performed using WEKA[1] which contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization [11]. In order to compare the results of different classical clustering algorithms, we applied k-Means [2, 3], DBSCAN (Density Based Spatial Clustering of Applications with Noise) [6, 1], and EM (Expectation Maximization) [4, 5]. As discussed in Section 3, different combinations of similarity measures can be used. In order to show the different characteristics of the proposed similarity measures, we carried out four experiments: in the first experiment we applied only SAASM, in the second only SATSM, in the third only ASM, and in the last experiment we used a combination of similarity measures as described in Table 1. For example, a typical result of SAASM and k-Means is shown in Figure 2.

4.2 Results

Table 2 shows the clustering results. For each experiment and for each clustering algorithm the number of workflows contained in each cluster is listed.

[1] `http://www.cs.waikato.ac.nz/ml/weka/`

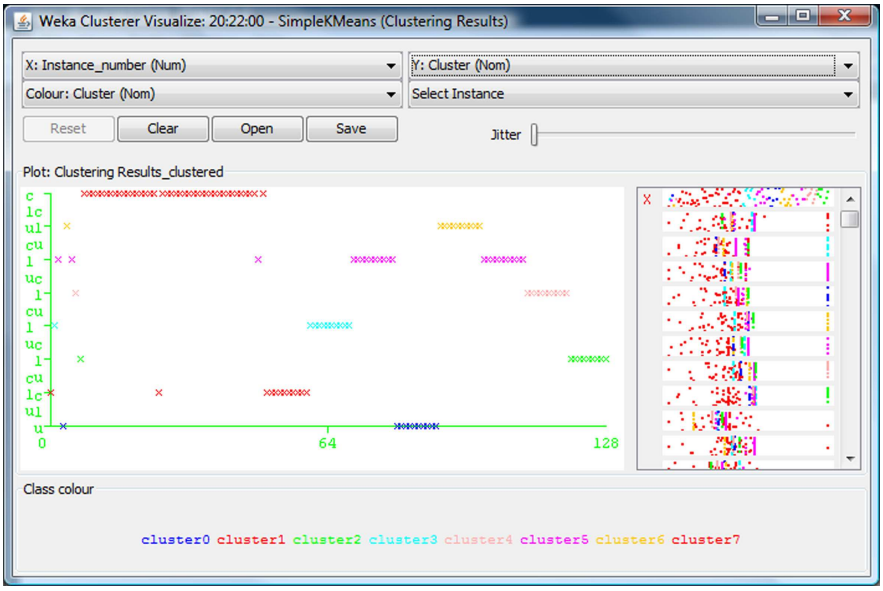

Fig. 2. Typical clustering result of SAASM and k-Means visualized in WEKA

Analyzing the results it can be noticed that the identified workflows vary significantly depending on which kind of similarity measure is used. This highlights the suitability of combined similarity measures in order to be able to tune workflow comparison for different application scenarios. k-Means showed the most robust results as the clusters had almost a fair distribution of workflows. After experimenting with different parameters, DBSCAN with $epsilon = 1.4$ and $minPts = 2$ showed ideal results in experiment 3 and experiment 4 as the workflows were clustered almost exactly in the way they were intended to be due to the generation procedure. The results of EM were not satisfying, as the number of clusters ranged from 2 to 5 clusters.

5 Conclusion and Future Work

In this paper we described a generic workflow model for representing different sequences of activities that have to be carried out in fire fighting operations. In order to support clear structuring and reuse, workflows are modeled as hierarchical structures. Each workflow contains a sequence of actions, whereby each action can either be a single activity, a checklist containing multiple single activities, a conditional action representing multiple alternatives, or a workflow itself.

Based on this workflow model, we presented several measures for calculating the similarity of workflows on the semantic (actions) as well as on the structural (transitions) level. These similarity measures represent the foundation for

Table 2. Clustering results

	k-Means	DBSCAN	EM		k-Means	DBSCAN	EM
Exp. 1	62	73	54	Exp. 2	30	99	10
	7	4	38		7	30	40
	19	2	24		17		30
	12	2	13		16		19
	6	3			23		30
	6	2			22		
	6	2			9		
	11	2			5		
Exp. 3	11	11	11	Exp. 4	11	11	54
	7	11	95		6	11	75
	10	11	11		5	11	
	4	11	11		5	11	
	64	10	1		5	10	
	11	11			11	11	
	11	11			75	11	
	11	11			11	11	

searching and clustering large numbers of workflows in order to identify workflows representing similar tasks and to enable simplification and standardization in the domain of fire fighting. Furthermore, we developed a new approach for combining different similarity measures depending on the type of actions that are compared. By this means, specialized similarity measures can be easily defined in a flexible way in order to reflect different application scenarios.

The suitability of the proposed similarity measures has been evaluated by applying classical clustering algorithms (k-Means, DBSCAN, Expectation Maximization) on a set of artificially generated workflows.

Future work will concentrate on the application of workflow similarity measures on different workflows in the domain of fire fighting operations as well as in the area of emergency services in general. As the experiments described in this paper were done using randomly generated workflows, the evaluation of the proposed similarity values on real workflows is still an open task. Furthermore, we will also continue experimenting with different combinations of similarity measures and clustering algorithms in order to identify a configuration that is best suited for clustering and analyzing workflows in the fire fighting domain.

6 Acknowledgements

The work described in this paper was done within the project "emergency mission control center" (emc^2) sponsored by the Austrian Research Promotion Agency (FFG).

References

1. D. Arlia and M. Coppola. Experiments in parallel clustering with dbscan. In *Proceedings of the 7th International Euro-Par Conference Manchester on Parallel Processing*, pages 326–331. Springer-Verlag, 2001.
2. D. Arthur and S. Vassilvitskii. How Slow is the k-Means Method? In *Proceedings of the twenty-second annual symposium on Computational geometry*, pages 144–153. ACM Press, 2006.
3. D. Arthur and S. Vassilvitskii. k-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035. Society for Industrial and Applied Mathematics, 2007.
4. S. Borman. The expectation maximization algorithm – a short tutorial. http://www.seanborman.com/publications/, July 2004.
5. F. Dellaert. The expectation maximization algorithm. Technical Report GIT-GVU-02-20, Georgia Institute of Technology, February 2002.
6. M. Ester, H. P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining*, pages 226–231. AAAI Press, 1996.
7. L. Fischer, editor. *2008 BPM & Workflow Handbook - Spotlight on Homan-Centric BPM*. Future Strategies, 1st edition, 2008.
8. J.-Y. Jung and J. Bae. Workflow clustering method based on process similarity. In *Computational Science and Its Applications - ICCSA 2006*, volume 3981 of *Lecture Notes in Computer Science*, pages 379–389. Springer, 2006.
9. J. Lundberg. *Principles of Workflow Support in Life Critical Situations*. Number 2 in Blekinge Institute of Technology Licentiate Dissertion Series. Blekinge Institute of Technology, 2007.
10. W. van der Aalst and K. van Hee. *Workflow Management: Models, Methods, and Systems*. Cooperative Information Systems. MIT Press, 2004.
11. I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, San Francisco, 2nd edition edition, June 2005.