On the Use of Scientific Workflows for Digital Humanities Research

Technical Report

Hess, Jan
hess@uni-trier.de
Trier Center for Digital Humanities (TCDH)
Universität Trier

Zeyen, Christian
zeyen@uni-trier.de
Center for Informatics Research and Technology (CIRT)
Universität Trier

Introduction

In recent years, research in the digital humanities (DH) increasingly focuses on enhancing transparency, reproducibility, and reusability of results with underlying processes. There have been proposals to consider modularization when developing new methods (Kuhn und Reiter) or to document research processes using process models (Kuras und Eckart 2017; Hug et al. 2011). Ongoing research (Liu et al. 2017) targets a methodology that allows for the reuse of research processes and required resources in the field of topic modeling. To this end, applied tools and scripts are stored in virtual environments and queryable metadata descriptions are created.

In e-Science, many communities rely on workflow technology as an established means for enabling scholars to develop their own computational methods in a reproducible and reusable way (Gil et al. 2007). Scientific Workflow Management Systems (SciWMS) enable researchers to create, execute, and manage scientific workflows without the need of having extensive programming skills. Those workflows particularly capture expert knowledge of how to solve a concrete problem in terms of required data, sequence of processing steps, and parameter settings. By reusing workflows or by experimenting with different combinations non-expert users are able to perform data analysis tasks more efficiently (Hauder et al. 2011).

As part of *eXplore!* (Lebherz et al., 2018), we focus on using scientific workflows for performing text analysis in literary studies. For this purpose, the project is testing RapidMiner (formerly named Yale) (Mierswa et al. 2006) as a SciWMS, which supports the visual programming of workflows for text and data mining tasks. To facilitate the familiarization with the workflow system and the underlying workflow language and to support the creation of workflows, we are developing a modeling assistance that supports the retrieval of already available workflows.

In our presentation, we report on the development and application of scientific workflows and highlight the challenges and opportunities we identified. Moreover, we present a first prototype of the modeling assistance that addresses some of these challenges and we show results of an experimental evaluation.

Application in the *eXplore!* project

The main goals of *eXplore!* are to research computer simulation and scientific workflows as new digital methods for literary studies. Therefore, the project conducts a pilot study on factors influencing the literary productivity of the German writer Klaus Mann based on his diaries. Text and data mining methods help to identify relevant patterns and relations and are also applied to support the semantic annotation of entities such as persons, places, or literary works. Due to the large number of entities and their various spellings in the diaries, this annotation process is demanding and error-prone and thus requires the interaction between manual and computational methods. For this purpose, different semi-automatic scientific workflows are constructed and executed using the RapidMiner SciWMS. These workflows are collected in a repository that functions as an extensible source for the workflow modeling assistance. In the following, the application of workflows is described based on the example of the semi-automatic annotation of persons in the diaries of Klaus Mann.

Different interactive workflows are executed repeatedly in the course of the manual annotation. Three documents are processed by the editor and by the workflows: a transcription of the diaries in XML/TEI, a register of persons based on the printed edition of the diaries, and a negative list of false annotations to be deleted and ignored in the further processing steps. The initial person annotation workflow executes consecutively four subprocesses (cf. Figure 1). In a first step, regular expressions are created for every entry in the register of persons (e.g., to detect and annotate inflected personal names or spelling variants). The second subprocess combines the Stanford CoreNLP NER with the RegexNER annotator that is trained with

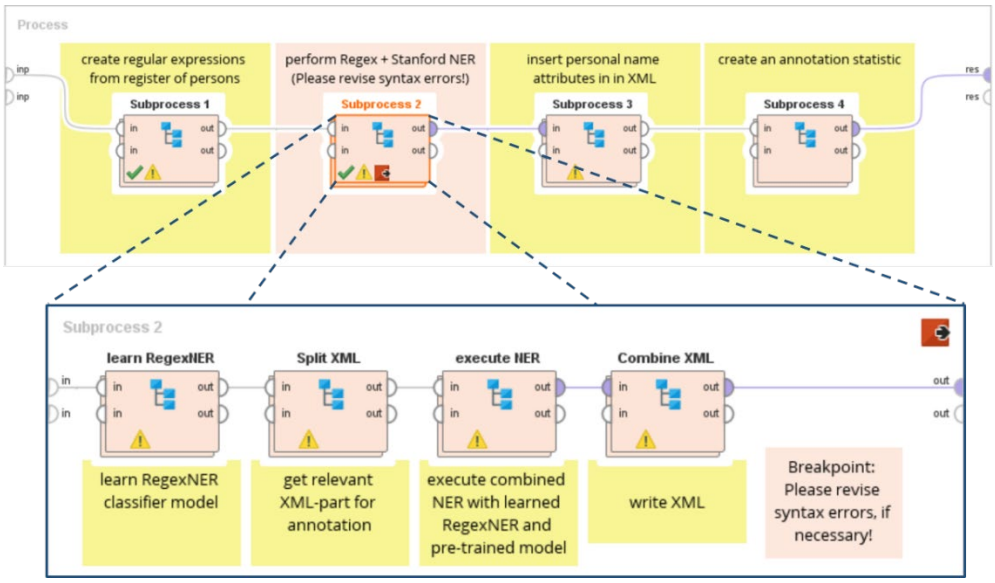

Figure 1: RapidMiner Workflow for the initial Annotation of Persons

the regular expressions built in the first step to detect and automatically annotate personal names in the XML document. This process may cause syntax errors that have to be manually corrected.

Subsequently, the third subprocess assigns the annotated personal names to corresponding entries in the register of persons and inserts the full personal names as attributes to each annotation to facilitate a manual examination of the automatic annotation. In the fourth and last subprocess an annotation statistic is created to summarize the applied modifications.

Certainly, the initial annotation is not faultless and has to be revised by a literary scientist. To support this step, further workflows are defined, which use the manual revisions to improve and complete the automatic annotation. For example, automatically tagged text passages that are actually no personal names, can be added to the negative list and automatically untagged. Subsequently, another workflow extracts the full personal name attributes added to the XML document during the manual revision and appends them to the register of persons. With a completion workflow these new entries can be processed analogously to the initial annotation.

Besides the aspect of time savings, which is particularly relevant when working with large corpora containing many recurring persons like the Klaus Mann diaries, SciWMS like RapidMiner allow for tracing the automatic steps of the annotation. The described workflows can easily be customized to annotate other named entities. To customize available workflows and to create new ones, it is not necessary to have extensive programming skills, although it is possible to integrate and run scripts within a scientific workflow.

Nonetheless, it is necessary to become familiar with the workflow paradigm. Especially for non-programmers, it is still difficult to overview all the available operators with their intended purposes. Despite the modularity of workflows and depending on the complexity of the research question, the creation of an adequate scientific workflow solution can be challenging.

Workflow Modeling Assistance

To support non-workflow experts in developing their own workflows, we propose a modeling assistance based on Case-Based Reasoning (CBR) (Aamodt und Plaza 1994) (cf. Figure 2).

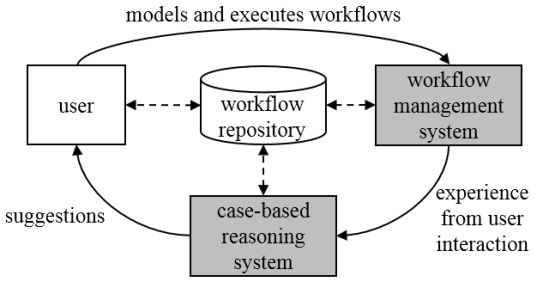

Figure 2: Modeling Assistance by Case-Based Reasoning

In a nutshell, CBR is a problem-solving method that suggests adapted solutions to users based on previously gathered experience from similar problems in the past. The core assumption is that similar problems

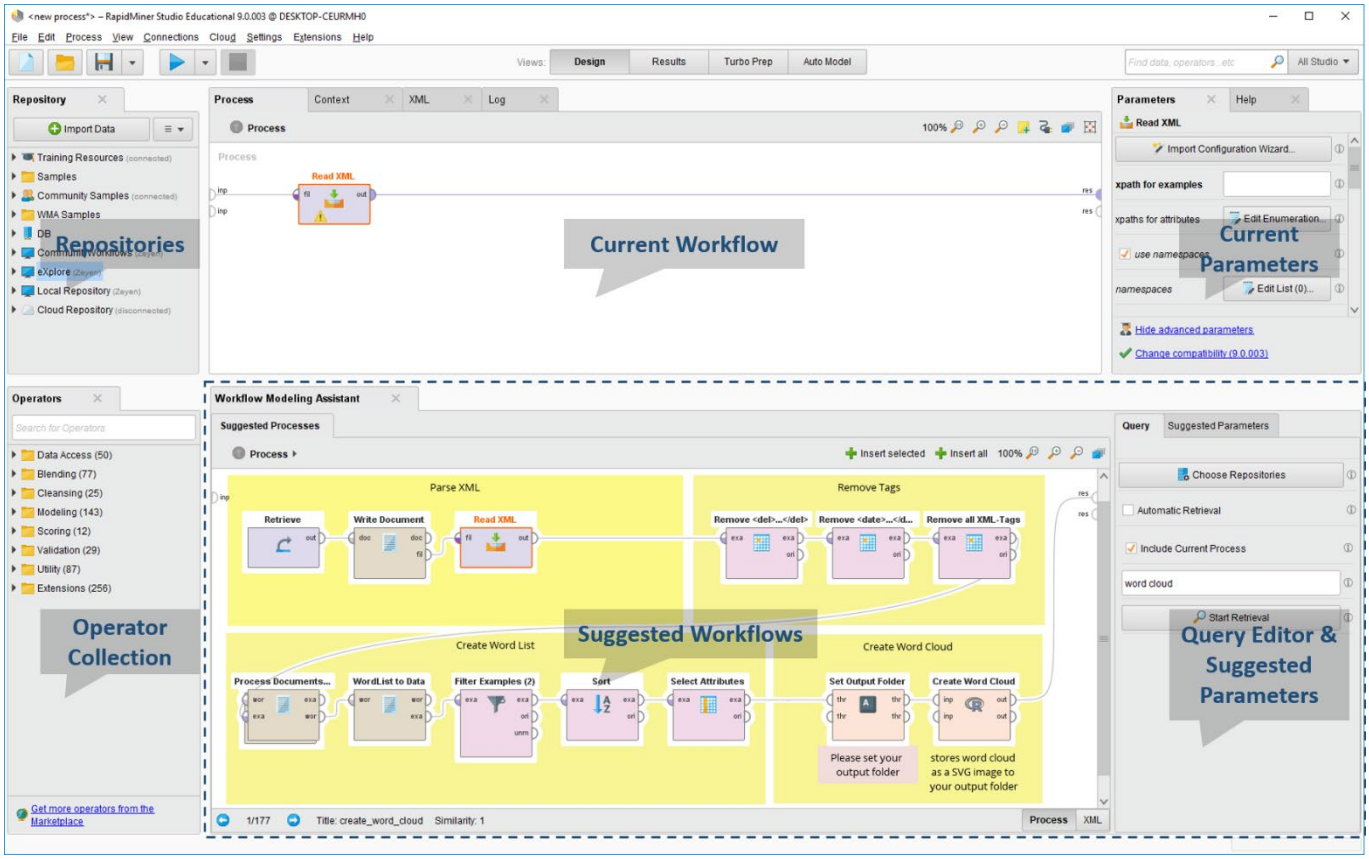

Figure 3: User Interface of RapidMiner Studio with the Workflow Modeling Assistant

tend to have similar solutions. When applied to workflow modeling, the CBR system provides suggestions to the user for completing the workflow under construction based on similar workflows stored in a repository (Bergmann und Müller 2018).

Consequently, the modeling assistance facilitates the reuse of workflows by retrieving and suggesting the most similar workflows from the past. In previous work (Malburg, Münster, Zeyen, & Bergmann, 2018), we investigate a suitable query model and retrieval mechanism for this purpose. A query can consist of the current (incomplete) workflow and/or a metadata description (describing features, goals, etc. with keywords). Similar workflows are determined based on their structural similarity using subgraph-matching, their semantic similarity of workflow components based on ontological knowledge, and their bag-of-words similarity of the metadata description.

We have implemented a first prototype as a plugin for RapidMiner that provides the basic functionality, i.e., the workflow retrieval. Figure 3 gives a screenshot of the RapidMiner user interface for modeling and executing workflows. The basic components comprise the workflow editor, the parameter settings, the repository list, and the operator collection. The dashed rectangle marks the modeling assistant that consists of an interactive workflow display, a parameter view, and a query editor. In the given example, the current workflow contains a single processing step 'Read XML' and 'word cloud' is inserted into the keyword description of the query. With these information, a workflow is retrieved that parses an XML document, preprocesses the word list, and creates a word cloud by executing a small R script.

Given that suitable workflows are available, the assistant can provide support in several ways. For instance, single operators can be viewed in the context of a complete and executable workflow as well as their specific parameter settings. Suitable operators can then be inserted into the current workflow. To reuse existing workflows entirely or in parts, they can be retrieved based on their metadata description only or in combination with an already modeled (yet incomplete) workflow. Moreover, alternative implementations of the current workflow that fulfill similar tasks can be retrieved.

To assess the general usefulness of the assistance we are currently conducting an experimental evaluation with students having limited experience in working with the RapidMiner workflow system. The evalution

builds up upon a preliminary study study (Malburg et al., 2018)we conducted with eleven students who tried to solve simple data mining tasks using the assistance. Although the assistance was provided as a separate application that only presented screenshots of retrieved workflows, the study has shown promising results regarding the feedback of the participants and observed time-savings.

Conclusion

Our work deals with the use of scientific workflows in literary studies. We demonstrate their application for the task of person annotation, which is a reoccurring task in many projects. To facilitate the reuse of proven workflows, we build up a repository and propose a modeling assistance that allows for retrieving similar workflows for a current workflow under development.

In future work, we will further investigate the use of scientific workflows for advanced research questions in the literary studies. Proven workflows will be further revised and optimized regarding their reusability. We also plan to extend the modeling assistant. In a next step, a conversational approach will be integrated that enables a more interactive retrieval and that identifies workflow fragments for reuse.

Acknowledgments

This work is part of the *eXplore!* project that is funded by the German Federal Ministry of Education and Research (BMBF) under grant number 01UG1606.

References

Aamodt, Agnar; Plaza, Enric (1994): Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. In: *AI Communications* 7 (1), S. 39–59.

Bergmann, Ralph; Müller, Gilbert (2018): Similarity-Based Retrieval and Automatic Adaptation of Semantic Workflows. In: Grzegorz J. Nalepa und Joachim Baumeister (Hg.): Synergies Between Knowledge Engineering and Software Engineering, Bd. 626: Springer (Advances in Intelligent Systems and Computing), S. 31–54.

Gil, Yolanda; Deelman, Ewa; Ellisman, Mark; Fahringer, Thomas; Fox, Geoffrey; Gannon, Dennis et al. (2007): Examining the Challenges of Scientific Workflows. In: *IEEE Computer* 40 (12), S. 24–32.

Hauder, Matheus; Gil, Yolanda; Sethi, Ricky J.; Liu, Yan; Jo, Hyunjoon (2011): Making data analysis expertise broadly accessible through workflows. In: Ian J. Taylor und Johan Montagnat (Hg.): WORKS'11, Proceedings of the 6th Workshop on Workflows in Support of Large-Scale Science, 2011: ACM, S. 77–86.

Hug, Charlotte; Salinesi, Camille; Deneckere, Rebecca; Lamasse, Stéphane (2011): Process modeling for Humanities: tracing and analyzing scientific processes. In: Philip Verhagen (Hg.): Annual Conference of Computer Applications and Quantitative Methods in Archaeology (CAA 2011). Beijing, China: Amsterdam University Press, S. 245–255.

Kuhn, Jonas; Reiter, Nils: A Plea for a Method-Driven Agenda in the Digital Humanities. In: Digital Humanities 2015, DH 2015, Conference Abstracts, 2015.

Kuras, Christoph; Eckart, Thomas (2017): Prozessmodellierung mittels BPMN in Forschungsinfrastrukturen der Digital Humanities. In: Maximilian Eibl und Martin Gaedke (Hg.): INFORMATIK 2017: Gesellschaft für Informatik, Bonn, S. 1101–1112.

Lebherz, D. S., Zeyen, C., Hess, J., Bergmann, R., Timm, I. J., Burch, T.,. . . Moulin, C. (2018). Towards Scientific Workflows and Computer Simulation as a Method in Digital Humanities. In M. Burghardt & C. Müller-Birn (Eds.), *INF-DH-2018.* Bonn: Gesellschaft für Informatik e.V.

Liu, Alan; Champagne, Ashley; Douglass, Jeremy; Kleinman, Scott; Russell, Jamal; Thomas, Lindsay (2017): Open, Shareable, Reproducible Workflows for the Digital Humanities: The Case of the 4Humanities.org "WhatEvery1Says" Project. In: Rhian Lewis, Cecily Raynor, Dominic Forest, Michael Sinatra und Stéfan Sinclair (Hg.): Digital Humanities 2017, DH 2017, Conference Abstracts, 2017: Alliance of Digital Humanities Organizations (ADHO).

Malburg, L., Münster, N., Zeyen, C., & Bergmann, R. (2018). Query Model and Similarity-Based Retrieval for Workflow Reuse in the Digital Humanities. In R. Gemulla, S. P. Ponzetto, C. Bizer, M. Keuper, & H. Stuckenschmidt (Eds.): *CEUR Workshop Proceedings, Proceedings of the Conference "Lernen, Wissen, Daten, Analysen", LWDA 2018, Mannheim, Germany, August 22-24, 2018* (pp. 251–262). CEUR-WS.org.

Mierswa, Ingo; Wurst, Michael; Klinkenberg, Ralf; Scholz, Martin; Euler, Timm (2006): YALE: Rapid prototyping for complex data mining tasks. In: Tina Eliassi-Rad, Lyle H. Ungar, Mark Craven und Dimitrios Gunopulos (Hg.): Proceedings of the Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2006: ACM, S. 935–940.