



Towards a Rule-Based Recommendation Approach for Business Process Modeling

Diana Sola^{1,2}(✉)

¹ Intelligent Robotic Process Automation, SAP SE, Walldorf, Germany

² Data and Web Science Group, University of Mannheim, Mannheim, Germany
diana@informatik.uni-mannheim.de

Abstract. Business process modeling can be time-consuming and error-prone, especially for inexperienced users. For this reason, graphical editors for business process modeling should support users by providing suggestions on how to complete a currently developed business process model. We address this problem with a rule-based activity recommendation approach, which suggests suitable activities to extend the business process model that is currently edited at a user-defined position. Contrary to alternative approaches, rules provide an additional explanation for the recommendation, which can be useful in cases where a user might be torn between two alternatives. We plan to investigate how rule learning can be efficiently designed for the given problem setting and how a rule-based approach performs compared to alternative methods. In this paper we describe the basic idea, a first implementation and first results.

1 Introduction

A business process model is the graphical representation of an organization's business process and an important instrument for Business Process Management. When modeling a business process, it is essential to precisely label the individual elements such that the process is consistent and unambiguous. In the case of domain-specific processes, this might require using a specialized and sometimes technical vocabulary, which often turns out to be challenging. There are a lot of tools supporting the modeling of business processes in a graphical notation such as Business Process Model and Notation or Petri Nets. Usually, they are graphical editors providing the user with a repository of symbols, which represent the building blocks of the underlying modeling language [12]. However, business process modeling remains time-consuming and error-prone, especially for inexperienced users. The modeling task can be facilitated by providing features which assist users during modeling and make recommendations on how to complete a business process model that is being edited [8]. The basis for such a recommendation feature could be a repository of completed business process models.

Supervised by Heiner Stuckenschmidt (Data and Web Science Group, University of Mannheim, Mannheim, Germany), heiner@informatik.uni-mannheim.de.

© Springer Nature Switzerland AG 2021

H. Hacid et al. (Eds.): ICSOC 2020 Workshops, LNCS 12632, pp. 25–31, 2021.

https://doi.org/10.1007/978-3-030-76352-7_4

One possible recommendation approach in business process modeling is activity recommendation [17]. Given the business process model being worked on, the recommendation system makes suggestions regarding suitable activities to extend the model at a user-defined position. In other words, the system recommends proper activities to support the user modeling business processes in an iterative way. Figure 1 shows an example of a business process model that is currently developed. The user has just added the sequence flow with label ‘Yes’. The task of the recommendation system is to find a suitable activity at this position. Since the business process model that has been developed so far depicts a version of the order-to-cash process, the recommender system suggests activities that have been used in similar order-to-cash processes in the repository: ‘Submit purchase order’, ‘Analyze quotation’ and ‘Create and submit the quotation’.

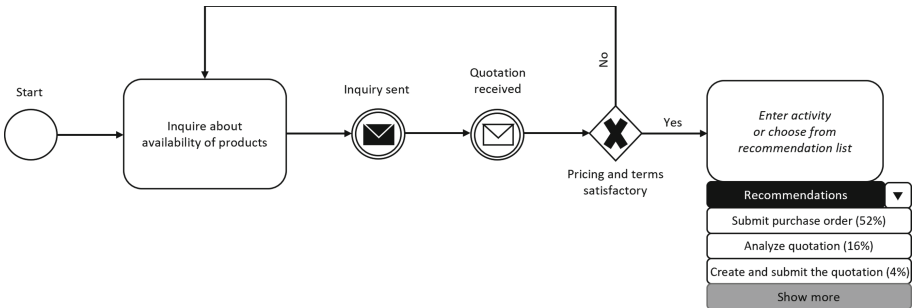


Fig. 1. A business process model under development

Structural patterns of activities (e.g. the order of activities) play an important role in the development of a recommendation method since some activities are more relevant at the current modeling phase than others. This poses a challenge in comparison to usual applications of recommender systems as they rarely need to consider structural patterns for the similarity of items or users.

Rules are a good option to model such patterns, therefore we want to address the activity recommendation problem by learning rules that capture the interrelationships between activities in processes of a repository. The rules can then be applied to a process under development. We intend to investigate how rule learning can be efficiently designed for the given problem setting. Moreover, we plan to analyse how a rule-based approach performs compared to alternative methods and what advantages can be derived in contrast to other methods. In this paper we propose a first, simple implementation and report about experiments where we compare it against one alternative approach.

2 Related Work

An overview of recommendation methods for business process modeling is presented in [11]. Kluza et al. distinguish between a subject-based classification,

which concentrates on what is actually suggested, and a complementary position-based classification, which focuses on the position where the suggestion is to be placed. According to their categorization, our work falls into the category of full-name suggestion for an element. The evaluation method that we use in the experimental studies corresponds to a forward completion approach.

There are several works that abstract a business process model to a directed graph and use graph-mining techniques to extract structural patterns from the process repository. While Cao et al. [3] calculate the distance between patterns and the partial business process based on graph edit distance [2], Li et al. [14] propose an efficient string edit distance [13] based similarity metric which turns the graph-matching problem into a string matching problem. Different distance calculation strategies are compared by Deng et al. [7].

An approach that involves semantic information and patterns observed in other users' preferences is proposed by Koschmider et al. [12]. They present a business process modeling editor with two features. First, the editor allows the user to search for process model fragments via a query interface, which is based on semantic annotations. Second, the system recommends appropriate process model fragments to the model being edited, which is based on the combination of several aspects as the frequency a process part has been selected by other users or its process design quality. In two experiments, the authors prove the usefulness and efficiency of their editor. However, the evaluation was based on a comparably small repository of process fragments that were developed particularly for the study's modeling exercise.

In [17], Wang et al. present their embedding-based activity recommendation method RLRecommender which extracts relations between activities of the process models and embeds both activities and relations into a continuous low-dimensional space. The training model used is based on TransE [1]. The embedded vectors for activities and relations and their distances in the space are then used to recommend an appropriate activity.

Jannach et al. [10] propose different recommendation techniques to provide modeling support for users in the specific area of data analysis workflows and evaluate them using a pool of several thousand existing workflows. The user support consists of recommending additional operations to insert into the currently developed machine learning workflow and is hence similar to activity recommendation. In a laboratory study, the authors show that their recommendation tool helps users to significantly increase the efficiency of the modeling process.

3 A Rule-Based Recommendation Approach

Following [17], we frame the activity recommendation problem in terms of a knowledge graph completion task (sometimes also referred to as link prediction). Within this framework, the processes of the repository and the incomplete process are represented as a (large) graph consisting of triples (*head activity, relation, tail activity*) and the recommendation of an appropriate activity has to be understood as the completion task. Within the last decade, the knowledge graph completion task has received lots of attention and the majority of

approaches uses embedding methods, where the knowledge graph is embedded into a low-dimensional space [18]. Thus, it is no surprise that this technique has also been used by Wang et al. in [17] to propose the embedding-based method RLRecommender as a solution for the activity recommendation problem.

While approaches that are based on embeddings dominate knowledge graph completion, more recently rule-based approaches, which have their origin in the field of inductive logic programming [5], have proven to be competitive [15]. As an additional benefit, these approaches offer an explanation for the given recommendation. Explainable recommendations have recently attracted more and more interest since they help improving the transparency, persuasiveness, effectiveness, trustworthiness, and satisfaction of recommendation systems [19].

We propose an approach which learns logical rules that describe how activities are used in the given process repository. These rules are used to give explainable recommendations for an appropriate activity at a given position. Our rule learner is based on the top-down search implemented in the association rule mining systems WARMR [6] and AMIE [9]. However, our implementation supports a specific language especially designed for predicting activities. The learned rules are Horn rules that predict the label u of an activity node X . In particular, they have the form $u(X) \leftarrow relation(X, Y), v(Y)$, where Y denotes another activity node in the process, v denotes the label of Y and $relation$ denotes the relation between the activities X and Y . For the relations between activities we make use of the definitions in [17]. The ‘Direct After’ relation depicts the connections of activities but makes it impossible to distinguish between an AND and an OR split. The ‘Direct Causal’ relations allow to capture the semantics of business process models more precisely. The concurrency of activities are described by ‘Direct Concurrent’ relations. The definition of these three relation families results in three rule learning strategies. Our first rule learning strategy (rules-after) is to only allow ‘after’ relations in the rule bodies. Analogously, we only allow ‘causal’ relations in the rule bodies for the second strategy (rules-causal). The third rule learning strategy (rules-concurrent) is to allow ‘causal’ and ‘concurrent’ relations in the rule body.

For the experiments, we made use of the two datasets that have also been used in the evaluation of RLRecommender in [17]. The first dataset (large dataset) consists of processes from the model collection of the Business Process Management Academic Initiative [16]. The second dataset (small dataset) consists of 221 processes collected from a district government in Hangzhou, China [4]. As evaluation metric we use the hit rate, which is the fraction of hits, where a hit is achieved if the generated recommendation list contains the activity that was actually chosen. We adopted the evaluation method from [17]. For the small dataset, we performed a fivefold cross-validation. For the large dataset, we chose the training and test split that we received from running the preprocessing code from RLRecommender on Github¹. As in [17], we report the hit rate for recommendation list lengths 1–5 and for the large dataset additionally for length 10.

¹ <https://github.com/THUBPM/RLRecommender> .

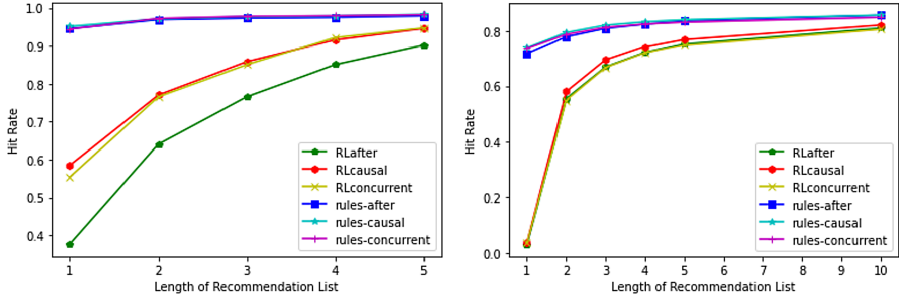


Fig. 2. Results on small (left) and large (right) dataset

The results of the experimental study in which we compared the embedding-based approach RLRecommender to our rule-based approach are depicted in Fig. 2. They show that our rule-based method outperforms the embedding-based approach on both datasets and for every recommendation list length.

4 Conclusion and Research Plans

This paper presents our ongoing work on a rule-based recommendation method for business process modeling which allows for explainable recommendations and outperforms an embedding-based approach.

In future work, we want to allow other forms of rules that involve more than one preceding activity of the process model. We also intend to conduct similar experimental evaluations with other existing methods and on other datasets. Jannach et al. [10] propose different approaches for predicting labels in the specific area of data analysis workflows. We plan to analyze whether these methods can also be used for the more general problem that we tackle, and if applicable, we will include their approach in a comprehensive experimental study.

The activity recommendation problem that we investigate is a multi-class classification problem. The learning of rules for multiple classes causes the problem of multiple rules firing. Until now we make use of a maximum strategy for the case that multiple rules make the same recommendation and assign the maximum confidence score of the rules to this recommendation. We plan to analyze other aggregation methods that are able to take entailment relations between rules into account.

In addition, we want to extend the problem to the possible case that there is no label in the process under development that has also been used in the process repository. In this case, we could try to match the labels to the labels in the repository and then apply the learned rules. However, this requires developing an approach that aggregates the confidence of generated mappings with the confidence scores of the rules to compute the ranking of the final recommendations.

Furthermore, we plan to investigate if a combined use of embeddings and rules can lead to further improvements in accuracy.

References

1. Bordes, A., Usunier, N., García-Durán, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: NIPS, pp. 2787–2795 (2013)
2. Bunke, H.: On a relation between graph edit distance and maximum common subgraph. *Pattern Recogn. Lett.* **18**(8), 689–694 (1997)
3. Cao, B., Yin, J., Deng, S., Wang, D., Wu, Z.: Graph-based workflow recommendation: on improving business process modeling. In: Proceedings of the 21st ACM International Conference on Information and Knowledge Management, pp. 1527–1531. Association for Computing Machinery, New York (2012)
4. Dataset consisting of processes collected from a district government in Hangzhou, China. <https://github.com/THUBPM/RLRecommender/tree/master/dataset/SRD>
5. De Raedt, L.: Logical and Relational Learning. Springer, Berlin (2008). <https://doi.org/10.1007/978-3-540-68856-3>
6. Dehaspe, L., Toivonen, H.: Discovery of relational association rules. In: Džeroski, S., LavračN. (eds.) *Relational data mining*, pp. 189–212. Springer, Berlin (2001). https://doi.org/10.1007/978-3-662-04599-2_8
7. Deng, S., et al.: A recommendation system to facilitate business process modeling. *IEEE Trans. Cybern.* **47**(6), 1380–1394 (2017)
8. Fellmann, M., Zarvic, N., Metzger, D., Koschmider, A.: Requirements catalog for business process modeling recommender systems. In: *Wirtschaftsinformatik*, pp. 393–407 (2015)
9. Galárraga, L.A., Teflioudi, C., Hose, K., Suchanek, F.: AMIE: association rule mining under incomplete evidence in ontological knowledge bases. In: Proceedings of the 22nd International Conference on World Wide Web, pp. 413–422 (2013)
10. Jannach, D., Jugovac, M., Lerche, L.: Supporting the design of machine learning workflows with a recommendation system. *ACM Trans. Interact. Intell. Syst. (TiiS)* **6**(1), 1–35 (2016)
11. Kluza, K., Baran, M., Bobek, S., Nalepa, G.J.: Overview of recommendation techniques in business process modeling. In: *KESE CEUR Workshop Proceedings*, vol. 1070. CEUR-WS.org (2013)
12. Koschmider, A., Hornung, T., Oberweis, A.: Recommendation-based editor for business process modeling. *Data Knowl. Eng.* **70**(6), 483–503 (2011)
13. Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions and reversals. *Sov. Phys. Dokl.* **10**(8), 707–710 (1966)
14. Li, Y., et al.: An efficient recommendation method for improving business process modeling. *IEEE Trans. Ind. Inform.* **10**(1), 502–513 (2014)
15. Meilicke, C., Chekol, M.W., Ruffinelli, D., Stuckenschmidt, H.: Anytime bottom-up rule learning for knowledge graph completion. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence, pp. 3137–3143. AAAI Press (2019)
16. Model collection of the Business Process Management Academic Initiative. <http://bpmmai.org/>

17. Wang, H., Wen, L., Lin, L., Wang, J.: RLRecommender: a representation-learning-based recommendation method for business process modeling. In: Pahl, C., Vukovic, M., Yin, J., Yu, Q. (eds.) ICSSOC 2018. LNCS, vol. 11236, pp. 478–486. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-03596-9_34
18. Wang, Q., Mao, Z., Wang, B., Guo, L.: Knowledge graph embedding: a survey of approaches and applications. *IEEE Trans. Knowl. Data Eng.* **29**(12), 2724–2743 (2017)
19. Zhang, Y., Chen, X.: Explainable recommendation: a survey and new perspectives. arXiv preprint [arXiv:1804.11192](https://arxiv.org/abs/1804.11192) (2018)