Content-based Filtering for Worklist Reordering to improve User Satisfaction: A Position Paper[a]

Sebastian Petter[1], Myriel Fichtner[1], Stefan Schönig[2] and Stefan Jablonski[1]
[1] *University of Bayreuth, Germany*
[2] *University of Regensburg, Germany*

Keywords: Business Process Management, Recommender Systems, Content-based Filtering, User-centered Process Improvement, User Satisfaction, Worklist Optimization, Process Activity Similarity.

Abstract: Business Process Management (BPM) is an approach to optimize business processes regarding certain company goals, e.g., the duration or quality of process outcome. Human resources are essential to business processes and are often neglected during process optimization. Considering domains focusing on users, it can be observed that recommender systems are often used to support user decisions and increase user satisfaction. This inspired us to use recommendation techniques in the context of BPM. Employee satisfaction significantly influences productivity, while employees are more satisfied when their preferences are taken into account during process execution. In this work, we propose to adopt the concept of content-based filtering to recommend worklist items to process participants they probably prefer. Since this work is part of a research project, we illustrate our approach on a simplified real-world business process from one of our application partners.

1 INTRODUCTION

Business processes can be understood as a series of activities to be completed to achieve a company goal. Optimizing such business processes is one of the primary purposes in research related to Business Process Management (BPM). Usual goals are to shorten processing time or to increase the quality of process outcomes (Jablonski and Bussler, 1996), (Koulopolous, 1995), (Lawrence, 1997). To reach optimization, companies need to gain insight into the structure of their processes. An established method is to visualize them using process modeling languages such as the Business Process Model and Notation (BPMN) (OMG, 2011). The resulting business process model contains the sequence of required steps and process entities involved, like process participants or data objects. Process-Aware Information Systems (PAIS) are used to manage and execute operational processes based on process models. Every executed activity is recorded in a so-called process event log during process execution. In addition, further details are included, like execution durations or resources executing activities (van der Aalst, 2016). Process mining techniques allow process optimization through the extraction of knowledge from event logs (Marin-Castro and Tello-Leal, 2021). Most approaches are activity-centered and focus on process control flow (Schonenberg et al., 2008), (Haisjackl and Weber, 2010). In contrast, many processes executed in small and medium enterprises are human-driven. Since human resources are an essential component of business process executions in companies, they have to be considered as a target for optimization.

In times of a shortage of skilled employees, companies have to distinctly focus on the needs and expectations of human workers. Consideration of such aspects increases employee satisfaction and decreases fluctuation. Furthermore, happiness and satisfaction with assigned work positively impact the employees' productivity. It is proved that "happy" workers are 13 % more productive (Bryson et al., 2015) (Bellet et al., 2019). Furthermore, human resources that have some degree of control over their work are more pro-

[a] The project is financed with funding provided by the Federal Ministry of Education and Research and the European Social Fund under the "Future of work" programme and managed by the Project Management Agency Karlsruhe (PTKA). The author is responsible for the content of this publication.

SPONSORED BY THE Federal Ministry of Education and Research
ESF European Social Fund for Germany
European Union
Designing. The Future. Together.

Petter, S., Fichtner, M., Schönig, S. and Jablonski, S.
Content-based Filtering for Worklist Reordering to improve User Satisfaction: A Position Paper.
DOI: 10.5220/0011092900003179
In *Proceedings of the 24th International Conference on Enterprise Information Systems (ICEIS 2022) - Volume 2*, pages 589-596
ISBN: 978-989-758-569-2; ISSN: 2184-4992

ductive and more motivated (Russell et al., 2016).

One way of increasing satisfaction and motivation of employees is to regard their preferences. Discovering preferences of human process participants can be adopted from the domain of recommender systems. Recommender systems are used to recommend items to users. For example, they are used in online shops to recommend products the user potentially prefers (Aggarwal, 2016). The use of such concepts has proven to be profitable for companies. These results inspired us to adopt the methods of recommender systems and using them in the context of BPM for user-centered process improvement. To the best of our knowledge existing research applies worklist reordering and optimization only for the purpose of improving key performance indicators. User and employee satisfaction, however, is neglected so far. In this work, we consider the concept of content-based filtering to recommend activities of a running business process instance to process participants. For this purpose, we introduce two ideas to enhance process activities and process event logs to adopt the concept of content-based filtering to BPM.

As a simple example, we consider a software development process. Besides other activities, it comprises the activities ”frontend development” and ”backend development”. Both must be executed by software developers. Some of them prefer to use JAVA to HTML. Due to missing knowledge about that preference, assigning activities to developers occurs randomly. The developers have to execute these activities independent of their preferences and the preferences of co-workers. This might lead to minor motivation to do the jobs.

Human resources have different preferences which may affect their motivation (Novak, 1999). By applying content-based filtering, we create user-centered process improvement by reordering the worklist items based on preferences of human resources. This creates a better working atmosphere, strengthens employee loyalty, and raises productivity. Furthermore, reordering the worklist has no negative impact on process performance (Pflug and Rinderle-Ma, 2015).

The main research question addresses the issue whether the approach of content-based filtering from recommender systems can be applied in BPM to increase user satisfaction. From this, two sub-tasks can be derived:

1. Which information can be exploited by a recommender system to infer user preferences?
2. How should these results be displayed to the users?

To realize user-centered process improvement combining BPM and recommender systems, an interdisciplinary consortium consisting of three research partners and five SMEs - three application partners and two implementation partners - collaborated in a joint project called PRIME[1] (Process-based integration of human expectations in digitalized work environments) funded by the Federal Ministry of Education and Research and the European Social Fund. In our work, we present an exemplary real-world process of one of our partners to show the relevance of this topic.

The remainder of the paper is structured as follows. In Section 2, we present the two main areas considered in this work: BPM and recommender systems in general and the content-based filtering method in particular. Furthermore, we show existing approaches combining these aspects. Section 3 describes the core idea of recommending activities to the current user of a PAIS. We conclude our work and give ideas for future work in Section 4.

2 BACKGROUND

This work considers the two fields of BPM and recommender systems to accomplish user-centered process improvement by means of worklist optimization. First, we introduce the two research areas, BPM and recommender systems. We then present related work that addresses both topics.

Business Process Management. BPM is used to increase efficiency and reduce costs of processes executed in enterprises of all industries and sizes. BPM considers modeling, executing, and analyzing business processes (Dumas et al., 2018). In order to execute a business process, first, a business process model is created. A Business Process Management System executes the model and displays activities that have to be executed. Activities that need to be executed are represented in a worklist. Since multiple users with the same roles are usually involved in the process execution, the worklist can be same for them. Users can select their preferred activities from the worklist as long as there are no further restrictions. Historical process executions of a business process are stored in so-called (process) event logs. Therefore each execution of an activity is recorded as event. Such events encapsulate all relevant information about the execution, e.g., execution timestamp or the involved resources. Other properties can arbitrarily extend this set of information. Events that refer to

[1]https://prime-interaktionsarbeit.de/

the same process instance are temporally ordered by their timestamp and build so-called ”traces” (van der Aalst, 2016).

Recommender Systems. Recommender systems predict results for alternative options based on user preferences. (Guo et al., 2019). Preference is one of the fundamental attributes to support user (Goldsmith and Junker, 2008). The value of recommender systems has been demonstrated in various application domains like E-commerce and other user-centric web applications (Manouselis and Costopoulou, 2007), (Nguyen and Haddawy, 1998). To calculate predictions, recommendation algorithms require information about users and items that should be recommended. Four main recommendation methods have been established, namely collaborative filtering, content-based filtering, knowledge-based filtering, and hybrid recommendation, which differ mainly in the input information used to calculate recommendations. In our work, we focus on content-based filtering. Content-based filtering aims to classify items with specific keywords, learn about user preferences, look up those keywords, and recommend similar items. Content-based filtering exploites two types of input data: (i) user feedback (explicit and implicit) and (ii) item attributes. User feedback is created while using, for example, web applications. Consider a webshop, where users can buy products and rate them afterwards. Buying a product is considered *implicit user feedback* (liking this product), whereas the rating is *explicit feedback*. Any product of a webshop has additional information about itself, so-called ”item attributes”, e.g., a book has an author and a genre. User preferences can be derived from feedback created by users. The item attributes are used to compare activities. Combining item attributes and user feedback, recommendation algorithms can calculate the probability of a user liking a specific product depending on the similarity to other, already rated products. The goal of content-based filtering methods is to predict the rating of each item considered using training data where items are rated by the user. Content-based filtering follows three main steps: (i) Preprocessing and Feature Extraction, (ii) Learning User Profiles, and (iii) Filtering and Recommendation (Aggarwal, 2016). During step (i) features are extracted from the considered item and are converted into a keyword-based vector space. In (ii), a user-specific model is created to predict user interest in items. The construction of this model is based on the history of a user buying or rating items. (iii) establishes recommendations on items for the user to consider while creating the model. A drawback of content-based filtering is the cold-start problem.It describes that it is impossible to create recommendations if we have no information about the user (e.g., in case of a new user). Furthermore, only apparent items are often recommended because the algorithm recommends items with similar attributes. In return, new items can be recommended because they are comparable to existing items due to the item attributes, and the recommendation is not based on preferences of other users. Furthermore, item attributes can be derived from the textual description of an item.

Related Work. Now, we provide an overview of work that combines the two areas Business Process Management and recommender systems. Most approaches consider process improvement through recommendations, whereas the focus is on optimizing process results or shortening processing time. In (Schonenberg et al., 2008) and (Haisjackl and Weber, 2010) the authors present an approach creating recommendations on possible next steps through prediction of a partial case. The recommendations are based on the current execution trace and consideration of similar process instances. The goal is to assist the user in optimal decision-making. A similar approach is presented in (van der Aalst et al., 2010). Based on event logs, recommendations are made on which activity should be executed next to receive a certain goal like shortened processing time. To support users in flexible processes, business process models are enhanced by estimations about runtime and availability of resources in (Barba et al., 2012). Based on this information, an optimized execution plan can be generated. To reduce risks (e.g., exceeding deadlines during process execution), recommendations, on which activity has to be executed next, are made to the user in (Conforti et al., 2015). Such recommendations are also made in (Huber et al., 2015). Considering the successors, activities are recommended to the user having the shortest runtime. In (Bidar et al., 2019) the authors make a suggestion on how to derive user-preferences from event logs. The optimal allocation of resources at process level is computed in (Cabanillas et al., 2013). In this work, a priority-based resource allocation is proposed using the Semantic Ontology of User Preferences (García et al., 2010). Every time a new activity has to be executed, it is allocated to the resource with the highest preference. Reordering worklist items to improve key performance indicators is considered in (Pichler and Edre, 2019). Furthermore, (Pflug and Rinderle-Ma, 2015) shows, that reordering the worklist of process participants does not have negative impact on temporal parameters like throughput time. Summing up, existing research mainly focuses on recommending next activities and resource allocation for performance optimization.

3 CONCEPTUAL APPROACH

To support the comprehensibility of our approach, we introduce a running example of a business process modeled in the modeling language Business Process Model and Notation (BPMN). The simplified business process shown in Figure 1 is part of a real-world example from one of our application partners. It consists of five activities that must be executed by human resources with the role *Team leader*. The first step is to accept an incoming order the *Team leader* receives as an E-mail from a customer. Afterward, the Roster for the technician must be created in Excel and dispatched to the technician by E-mail. During the step "Preparation of order (ID4)", the *Team leader* communicates with the customer (getting detailed order information) and with the technician (send detailed order information as Excel-file by E-mail). The last step of this process is the treatment of special requests the technicians send via E-mail.

As described in Section 2.2, user feedback and item attributes are used as input information for content-based filtering. In BPM, we can consider activities as items.

Thus, we need to get user feedback related to activities and attributes describing activities. Implicit user feedback is created during process execution and is stored in an event log. This information is declared as implicit feedback since it relates to execution parameters (e.g., execution time) and not to information directly provided by executors. However, to identify user preferences, explicit feedback in term of user rating is still missing. Therefore, we need to introduce an additional step after executing user tasks: the rating of an executed activity. In our example, we simply consider *like* and *dislike* as activity ratings. In general, the ratings can be of any complexity.

Another preprocessing step for applying content-based methods in BPM is tagging activities. In process models, activities have labels interpretable as textual instructions but do not contain information about semantics. To get more information about an activity and to be able to compare activities, we need to add tags or labels to each activity. Our concept consists of three basic elements: (i) Preprocessing and Tagging, (ii) Content-based Filtering, and (iii) Worklist Reordering.

Preprocessing and Tagging. The input data for preprocessing are business process models, process descriptions, an event log, and the current state of process execution provided by a PAIS instance, e.g., the current users and their worklists. As process model the introduced example from above is taken. During execution of this process model an event log is generated. Every event log entry must include a rating of whether the executing process participant likes or dislikes the activity.

To create recommendations based on content-based filtering, we also need activity tags containing semantic information. These tags can be generated from the process model automatically or manually by a process expert during modeling. Furthermore, a process expert can manually adapt these keywords (adding new or removing irrelevant ones). An advantage of this approach is the possibility of considering and therefore interpretively connecting activities of multiple processes. Thus, executions of different processes by the same user can be combined through these activity tags. This enables to adopt preferences for similar activities from other processes.

In our running example, we derive five tags from the business process model: *E-mail*, *Excel*, *communication with customer*, *communication with technician*, and *roster*. None of the five activities has all tags. For example, "Accept order (ID1)" considers the activity tags *communication with customer* and *e-mail*. The "Preparation of order (ID4)" activity considers all tags but *roster*.

Content-based Filtering. To generate recommendations, we consider training data which is defined as a set of rated activities A_R. These activities are rated by the process participant the items are recommended to. Since the event log is enhanced by explicit user feedback, it contains all information to calculate recommendations with content-based filtering methods. Therefore we consider it as training data. Every item of the event log contains an activity ID, the information which user executed what activity, and a set of activity tags.

Besides the training set A_R for content-based filtering, an unrated set of activities A_U (worklist) is considered. As described in Section 2, the goal of content-based filtering methods is to predict the rating of each activity in A_U using the training data. Both, A_R and A_U, are user-specific sets of activities.

The issue we have to solve is similar to that of classification. An established technique to classify data sets is the nearest neighbor classifier (Chomboon et al., 2015). We will use this classification method in our running example, while it is interchangeable and

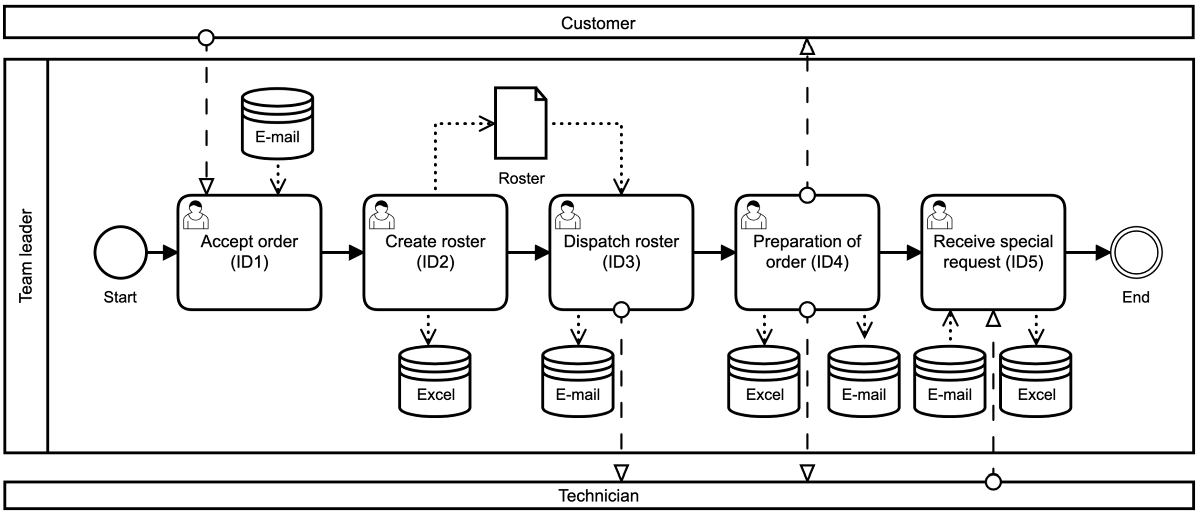

Figure 1: Example: Technician service management process.

can be replaced by any other classification method.

We define a similarity function based on the nearest neighbor classifier. Similarity/distance functions like the Euclidean distance or the Manhattan distance are used for structured or multidimensional data. In this work, we use the Euclidean distance to calculate the distance between two vectors to compare two activities. Afterward, we transform the results to a similarity measure due to better comparability.

The Euclidean distance is calculated as follows:

$$d(p,q) = \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2} \tag{1}$$

Where p and q are a pair of vectors representing activities with n attributes. Furthermore, a similarity metric is needed to improve comparability of the distances:

$$sim(p,q) = \frac{1}{1+d(p,q)} \tag{2}$$

For each activity in A_U, the similarity metric to each activity from A_R is calculated to identify the nearest neighbor. Considering the rating of the nearest neighbor activity, the rating of the related activity in A_U can be deduced.

In our running example, the training set A_R is based on the event log and the activity tags created in the Preprocessing and Tagging step. In Table 1 the training set A_R is displayed. The activities with ID1 – ID4 have been executed several times and rated by the current user. Activity ID5 is not displayed at the table because the team leader has never executed it until now. Furthermore, the tags for every activity are considered. In our example the values 0 and 1 show whether an attribute is present or not. The vector-representation of activity ID4 is $(1,1,1,1,0)$ since it is related to the tags *E-mail*, *Excel*, *communication with customer* and *communication with technician*. The set of unrated activities A_U is the worklist with tasks that team leaders must execute. We assume that the worklist of the team leaders contains currently five tasks in the following order where every task can be executed independent of others:

1. Create roster (ID2)
2. Dispatch roster (ID3)
3. Receive special request (ID5)
4. Accept order (ID1)
5. Preparation of order (ID4)

List 1: Worklist of team leader (FIFO).

To calculate the similarity between "Receive special request (ID5)" and "Preparation of order (ID4)" we have to define p as the activity "Receive special request (ID5)" in A_U and q as the activity "Preparation of order (ID4)" in A_R. The two vectors $v_p = (1,1,0,1,0)$ and $v_q = (1,1,1,1,0)$ are representations of the activities. Then $d(p,q) = 1$ and $sim(p,q) = 0.5$. Calculating all values for p of any activity in A_R, we notice that 0.5 is the highest value, whereas 1 means the activities are similar and 0 means that the activities are completely different. Considering the user rating of all "Preparation of order (ID4)" activities, we notice that the team leader always likes this activity. Thus, the team leader possibly likes "Receive special request (ID5)" too. Applying this calculation to all data considered, we achieve the following predictions: the first item of the team leader's worklist (List 1),"Create roster (ID2)", is rated as *dislike*. The activity has been executed by the team leader four times. Thus, the similarity metric $sim(p,q)$ equals 1.

Table 1: Training data A_R of one team leader for running example.

ActivityId	E-mail	Excel	Com. Customer	Com. Technician	Roster	...	Like/Disklike
ID1	1	0	1	0	0	...	DISLIKE
ID2	0	1	0	0	1	...	DISLIKE
ID3	1	0	0	1	1	...	DISLIKE
ID4	1	1	1	1	0	...	LIKE
ID1	1	0	1	0	0	...	DISLIKE
ID2	0	1	0	0	1	...	DISLIKE
ID3	1	0	0	1	1	...	LIKE
ID2	0	1	0	0	1	...	LIKE
ID2	0	1	0	0	1	...	DISLIKE
ID3	1	0	0	1	1	...	LIKE
ID4	1	1	1	1	0	...	LIKE

Three of four times, the user did not like the activity though the prediction for a feature rating is *dislike*. In the same way, the rating for ”Create roster (ID2)” is calculated, the rating for ”Dispatch roster (ID3)”, ”Accept order (ID1)”, and ”Preparation of order (ID4)” can be calculated because the user has already executed these activities. The only task the user did not execute yet is ”Receive special request (ID5)”. To predict this task, we need to calculate the nearest neighbor by calculating the Euclidean distance and this similarity metric shown in Formula (1) and (2). The calculation results in a similarity metric of 0.5 regarding the activity ”Preparation of order (ID4)”. This activity has been executed twice, and the user liked it both times, thus with a high probability the user likes the new activity.

Worklist Reordering. Most PAIS organize a worklist according to the FIFO principle. Taking the results from step (ii) (Content-based Filtering), we can rearrange the worklist and adapt it to the requirements of a current user. The manipulation of the items only affects the order of the worklist items. We do not omit any tasks as well as we do not add any new tasks. The active user can still decide which activity to execute next.

While displaying the reordered, customized worklist, it is required to present the reason for reordering so the user can reproduce why specific activities are recommended.

Taking the results of step (ii) (Content-based Filtering) and adapting the worklist according to these recommendations, the following reordered worklist can be generated:

1. Dispatch roster (ID3)
2. Receive special request (ID5)
3. Preparation of order (ID4)
4. Create roster (ID2)
5. Accept order (ID1)

List 2: Preference-based reordered worklist of team leader.

In this case, the content-based filtering method recommends executing the tasks ”Dispatch roster (ID3)”, ”Receive special request (ID5)”, and ”Preparation of order (ID4)” because these are the tasks the team leader potentially likes, whereas the last two tasks in List 2 are not among the user’s favorite tasks. Perhaps another team leader with opposite preferences performs the latter tasks, thus increasing that user’s satisfaction.

The example considers a worklist in first-in-first-out order. However, some approaches exist to reordering the worklist due to better process performance or similar (e.g., (Schonenberg et al., 2008)). These preordered worklists must be considered too. One possibility of including user preferences into preordered worklists is to reorder only tasks with the same priority, not falsifying previous calculations.

Considering the real-world example, we notice that the activities ”Receive special request (ID5)” and ”Preparation of order (ID4)” are similar, indeed. One team leader available during process documentation rated the activities ID1 - ID4 (Table 1). Activity ID5 has not been rated since we want to predict whether the team leader likes it or not. Presenting the result of the calculation, the team leader confirmed that he likes the activity ”Receive special request (ID5)”.

4 CONCLUSION AND FUTURE WORK

In this work, we provide a novel approach to increase user satisfaction during business process execution by worklist optimization through recommender systems and illustrate it with a running example. We apply the method of content-based filtering to a given worklist and the event log and present the result, a worklist, ordered according to potential user preferences, to the active user of the PAIS. An advantage of this approach is that existing data and information like the process model or the event log can be used to calculate the user's preferences. Furthermore, activities of the process model can be enhanced by other tags for more accurate predictions. We evaluated our work prototypically with our project partners. The idea received a lot of positive feedback, and the project partners confirmed that the presented idea leads to an improvement in user satisfaction. Initial prototypical implementations and evaluations showed the success of the approach. Manual calculations of recommendations provided promising results demonstrating the need for implementation.

In future work, we will extend our approach by alternate activity rating approaches as well as more approaches to generating activity tags. Furthermore, other methods known from recommender systems can be considered to tackle drawbacks, like cold-start problem for new users, or recommendation of apparent items, of content-based filtering method, e.g., collaborative filtering or knowledge-based filtering. Finally, we plan to implement a prototype and to include a real-world evaluation of our approach.

REFERENCES

van der Aalst, W. M. P. (2016). *Process Mining: Data Science in Action*. Springer, 2 edition.

van der Aalst, W. M. P., Pesic, M., and Song, M. (2010). Beyond process mining: From the past to present and future. In *Proc. of the 22nd International Conference on Advanced Information Systems Engineering (CAiSE '10)*, volume 6051.

Aggarwal, C. C. (2016). *Recommender Systems - The Textbook*. Springer.

Barba, I., Weber, B., and Valle, C. D. (2012). Supporting the optimized execution of business processes through recommendations. In *Proc. BPI*, volume 99, pages 135–140.

Bellet, C. S., Neve, J.-E. D., and Ward, G. (2019). Does employee happiness have an impact on productivity? *Saïd Business School WP*.

Bidar, R., ter Hofstede, A. H. M., Sindhgatta, R., and Ouyang, C. (2019). Preference-based resource and task allocation in business process automation. In *OTM 2019*, volume 11877, pages 404–421.

Bryson, A., Forth, J., and Stokes, L. (2015). Does worker wellbeing affect workplace performance? Technical report, Department for Business, Innovation & Skills.

Cabanillas, C., García, J. M., Resinas, M., Ruiz, D., Mendling, J., and Cortés, A. R. (2013). Priority-based human resource allocation in business processes. In *ICSOC 2013*, volume 8274, pages 374–388.

Chomboon, K., Chujai, P., Teerarassammee, P., Kerdprasop, K., and Kerdprasop, N. (2015). An empirical study of distance metrics for k-nearest neighbor algorithm. In *Proc. of the 3rd International Conference on Industrial Application Engineering*, pages 280–285.

Conforti, R., de Leoni, M., Rosa, M. L., van der Aalst, W. M. P., and ter Hofstede, A. H. M. (2015). A recommendation system for predicting risks across multiple business process instances. *Decis. Support Syst.*, 69:1–19.

Dumas, M., Rosa, M. L., Mendling, J., and Reijers, H. A. (2018). *Fundamentals of Business Process Management*. Springer, 2nd edition.

García, J. M., Ruiz, D., and Ruiz-Cortés, A. (2010). A model of user preferences for semantic services discovery and ranking. In *The Semantic Web: Research and Applications*, volume 6089, pages 1–14.

Goldsmith, J. and Junker, U. (2008). Preference handling for artificial intelligence. *AI Magazine*, 29:9–12.

Guo, Z., Zeng, W., Wang, H., and Shen, Y. (2019). An enhanced group recommender system by exploiting preference relation. *IEEE Access*, PP:1–1.

Haisjackl, C. and Weber, B. (2010). User assistance during process execution - an experimental evaluation of recommendation strategies. In *Proc. of BPI Workshop*, volume 66, pages 134–145.

Huber, S., Fietta, M., and Hof, S. (2015). Next step recommendation and prediction based on process mining in adaptive case management. In *7th International Conference on Subject-Oriented Business Process Management*.

Jablonski, S. and Bussler, C. (1996). *Workflow Management: Modeling Concepts, Architecture, and Implementation*.

Koulopolous, T. M. (1995). *The Workflow Imperative: Building Real World Business Solutions*.

Lawrence, P. (1997). *Workflow Handbook*.

Manouselis, N. and Costopoulou, C. (2007). Analysis and classification of multi-criteria recommender systems. *World Wide Web*, 10(4):415–441.

Marin-Castro, H. M. and Tello-Leal, E. (2021). Event log preprocessing for process mining: A review. *Applied Sciences*, 11.

Nguyen, H. and Haddawy, P. (1998). Diva: Applying decision theory to collaborative filtering. In *Proc. of the AAAI Workshop on Recommender Systems*.

Novak, J. M. (1999). Choice theory: A new psychology of personal freedom. *Brock Education: a Journal of Educational Research and Practice*, 9.

OMG (2011). Business Process Model and Notation (BPMN), Version 2.0.

Pflug, J. and Rinderle-Ma, S. (2015). Analyzing the effects of reordering work list items for selected control flow patterns. In *IEEE 19th International Enterprise Distributed Object Computing Workshop*, pages 14–23.

Pichler, H. and Edre, J. (2019). Towards look-ahead strategies for work item selection. *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, pages 752–757.

Russell, N., van der Aalst, W. M. P., and ter Hofstede, A. H. M. (2016). *Workflow Patterns: The Definitive Guide*. MIT Press.

Schonenberg, H., Weber, B., van Dongen, B. F., and van der Aalst, W. M. P. (2008). Supporting flexible processes through recommendations based on history. In *International Conference on Business Process Management (BPM '08)*, volume 5240, pages 51–66.