# Prescriptive Process Monitoring: Quo Vadis?

Kateryna Kubrak<sup>1</sup>, Fredrik Milani<sup>1</sup>, Alexander Nolte<sup>1,2</sup>, and Marlon Dumas<sup>1</sup>

<sup>1</sup> University of Tartu, Tartu, Estonia {kateryna.kubrak,milani,alexander.nolte,marlon.dumas}@ut.ee
<sup>2</sup> Carnegie Mellon University, Pittsburgh, PA, USA

Abstract. Prescriptive process monitoring methods seek to optimize a business process by recommending interventions at runtime to prevent negative outcomes or poorly performing cases. In recent years, various prescriptive process monitoring methods have been proposed. This paper studies existing methods in this field via a Systematic Literature Review (SLR). In order to structure the field, the paper proposes a framework for characterizing prescriptive process monitoring methods according to their performance objective, performance metrics, intervention types, modeling techniques, data inputs, and intervention policies. The SLR provides insights into challenges and areas for future research that could enhance the usefulness and applicability of prescriptive process monitoring methods. The paper highlights the need to validate existing and new methods in real-world settings, to extend the types of interventions beyond those related to the temporal and cost perspectives, and to design policies that take into account causality and second-order effects.

**Keywords:** Prescriptive process monitoring  $\cdot$  Process optimization  $\cdot$  Business process

## 1 Introduction

Process mining is a family of techniques that facilitate the discovery and analysis of business processes based on execution data. Process mining techniques use event logs extracted from enterprise information systems to, for instance, discover process models or to check the conformance of a process with respect to a reference model [42]. In this setting, an event log is a dataset capturing the execution of a process step by step, including timestamps, activity labels, case identifiers, resources, and other contextual attributes related to each case or each step within a case.

Over time, the scope of process mining has extended to encompass techniques that predict the outcome of ongoing cases of a process based on machine learning models constructed from event logs [24]. Predictions, however, only become useful to users when they are combined with recommendations [28]. In this setting, prescriptive process monitoring is a family of methods to recommend interventions during the execution of a case that, if followed, optimize the process with respect to one or more performance indicators [35]. For instance, an intervention might improve the probability of the desired outcome (e.g. on-time delivery) or

mitigate negative outcomes (e.g. delivery delays) [27]. Different prescriptive process monitoring methods have been proposed in the literature. These methods vary in relation to – among others – their predictive modeling approach and the interventions they prescribe. In some cases, two different methods aim at achieving the same objective but in different ways. For instance, to avoid an undesired outcome, one method prescribes assigning resources for the next task [36], whereas another recommends the next task to be executed [23].

The benefits of prescriptive process monitoring can only be fully realized if these methods prescribe effective interventions and if these prescriptions are followed [9]. At present, though, the variety and fragmentation of prescriptive monitoring methods makes it difficult to understand which method is likely to be most effective or more likely to be accepted by end users in a given business situation. There is no overview that captures existing prescriptive monitoring methods, what objectives they pursue, which interventions they prescribe, which data they require, or the extent to which these methods have been validated in real-life settings. Research overviews and classification frameworks have been put forward in the related field of predicative monitoring [11,28] and automated resource allocation [4,34], but not in the field of prescriptive process monitoring.

To address this gap, we study three research questions:

- Given that prescriptive process monitoring methods aim at prescribing interventions that produce business value, i.e., achieve an objective, we formulate the first research question as RQ<sub>1</sub>. What is the objective for using prescriptive process monitoring methods in the process?
- The second research question aims at discovering how the objectives can be achieved:  $\mathbf{RQ}_2$ . What are the interventions prescribed by prescriptive process monitoring methods?
- Finally, we explore the data required by the proposed methods: RQ<sub>3</sub>. What data do prescriptive process monitoring methods require?

To answer these questions, we conduct a Systematic Literature Review (SLR) following the guidelines proposed by Kitchenham et al. [22]. We identified 36 papers that we analyze these papers to develop a multi-dimensional framework to characterize prescriptive monitoring methods. The contribution of the paper is twofold. We first develop a framework that classifies the prescriptive process monitoring methods according to their objective, metric, intervention types, techniques, data inputs, and policies to trigger the interventions. Second, we provide insights into potential areas for future research in this field.

The rest of this paper is structured as follows. Section 2 discusses related work. In Section 3, we elaborate on the protocol of the SLR, while in Section 4 we present our findings. In Section 5 we present the proposed framework, followed by concluding remarks in Section 6.

### 2 Related Work

Methods for prescriptive process monitoring prescribe interventions that can change the outcomes of an ongoing process case. For instance, if a method detects

that an undesired outcome is probable to unfold, an alarm is raised that can lead to an intervention [37]. This intervention could come in the form of an action performed by a process worker, such as calling a customer, that helps to mitigate or prevent the negative outcome from materializing [37]. Interventions often entail an intervention cost (e.g., time spent executing an action) and a cost of undesired outcome (e.g., the order is canceled) [13]. Thus, it is essential to define a policy for when the prescription is generated. The example above [13] considers the probability of a negative outcome and evaluates the cost model and the mitigation effectiveness before triggering interventions.

A few previous studies focus on areas that are related to prescriptive process monitoring. Di Francescomarino et al. [11] introduce a value-driven framework that allows companies to identify when to apply predictive process monitoring methods. Another classification of predictive process monitoring methods is that of Márquez-Chamorro et al. [28], where the focus is on methods to train predictive models. In Mertens et al. [26], the authors evaluate predictive methods used to recommend follow-up activities in the healthcare domain. While prescriptive process monitoring methods incorporate predicted outputs, our work only focuses on prescriptive methods and prescribed interventions.

Pufahl et al. [34] present an SLR on automatic resource allocation. Similarly, Arias et al. [4] give an overview of resource allocation methods, but with a particular focus on human resources. However, the former work reviews prescriptive methods in general, and the latter two focus on a specific intervention, namely, resource allocation. In this paper, we enrich such works by considering all types of potential interventions in process-aware methods.

## 3 SLR Method

We aim to review the existing body of work on prescriptive process monitoring methods. More specifically, what the objectives for using such methods are  $(\mathbf{RQ}_1)$ , what interventions the methods prescribe  $(\mathbf{RQ}_2)$ , and what data the methods require  $(\mathbf{RQ}_3)$ . Therefore, we use the systematic literature review (SLR) method as it aids us to identify relevant literature in a specific research area [22]. We follow the guidelines proposed by Kitchenham et al. [22], who proposes three main steps: (1) planning the review, (2) conducting it, and (3) reporting the findings.

For the first step (planning), we identified research questions and developed the review protocol [22]. The research questions were defined and motivated above. We developed a search string for the review protocol, identified suitable electronic databases, and defined inclusion and exclusion criteria. Finally, we defined the data extraction strategy. In the search string, we included "process mining" to scope the study to methods that rely on event logs. We derived the term "prescriptive" from the research questions. We also included the terms "recommender" and "decision support", as we found these to be sometimes used instead of "prescriptive". Accordingly, we formulated the search string:

(recommender OR "decision support" OR prescriptive) AND "process mining"

While conducting the first search, we noted that the term "prescriptive process monitoring" is not consistently used. Using this search string only might thus lead us to miss relevant studies. We addressed this by examining the papers identified with the first search string to identify other used terms. We noted that terms such as "next-step recommendation", "next best actions", "proactive process adaptation" are used as synonyms for prescriptive process monitoring. We also noticed that the phrase "business process" often appeared in titles and keywords. Therefore, we formulated the second search string as:

(recommender OR "next activity" OR "next step" OR "next resource" OR proactive) AND "business process"

We searched using both strings on ACM Digital Library, Scopus (includes SpringerLink), Web of Science, and IEEE Xplore. The databases were selected based on coverage of publications within the field of process mining. We also ran the search strings on Google Scholar to capture potentially relevant works not yet published (such as arXiv). Finally, we conducted backwards referencing (snowballing) [32] to identify additional relevant papers.

Next, we defined the exclusion and inclusion criteria. We excluded papers not digitally accessible (EC1), not in English (EC2), duplicates (EC3), and shorter than six pages (EC4). Exclusion criteria EC1 and EC2 ensure that the paper can be generally accessed and understood by other researchers. Papers with open access or accessible via institutional access are considered accessible. Criterion EC3 removes duplicates that can appear since several digital libraries are used. We applied criterion EC4, as papers with less than six pages are likely not to contain enough information for our analysis. We also defined three inclusion criteria: (IC1) the paper is relevant to the domain of prescriptive process monitoring, (IC2) the paper presents, reviews, discusses, or demonstrates a method or a case for prescriptive process monitoring, (IC3) the paper describes at least one way to identify candidate interventions for an ongoing process case. IC1 aims to filter out the papers outside of the scope of the prescriptive process monitoring domain. With IC2, the studies that represent any theoretical discussion or practical application of a method are included. Inclusion criterion IC3 ensures that the paper contains enough information to address the research questions.

Finally, we defined the data extraction strategy. We first captured the metadata of all papers (title, author, publication venue, year). Then we defined the data required to address the research questions. Thus, for  $\mathbf{RQ}_1$ , we defined the data required to identify the objective of using the prescriptive process monitoring technique and the performance metric(s) targeted in each of the papers. Next, we defined the data to elicit the interventions prescribed, the process perspective, and the users the prescribed interventions are presented to  $(\mathbf{RQ}_2)$ . Finally, we defined the required data input for the techniques described in the different papers  $(\mathbf{RQ}_3)$ . Additionally, we added modeling techniques and policies used to trigger interventions.

We ran the search  $^3$  and identified a total of 1367 papers. We filtered them using the exclusion criteria EC1 and EC2. This resulted in the removal of 97

 $<sup>^3</sup>$  The first search was conducted on 22 Sep 2021, the second on 12 Oct 2021.

papers. Thus, 1270 papers remained and were filtered based on EC3. From the remaining 1010 papers, we removed short papers (EC4). This resulted in 900 papers remaining. These were filtered by title, thus removing papers that were clearly out of scope. The remaining 171 papers were filtered by abstract, resulting in 66 papers remaining. Finally, we applied the inclusion criteria by reading the whole paper and removing 44 papers. As a result, 22 papers remained. A total of 14 papers were added through backwards referencing, resulting in the final list of 36 relevant papers<sup>4</sup>.

Search	First		Second		Aggregat	ed
Selection criteria	# found	# left	# found	# left	# found	# left
Search results	572		795		1367	
Data cleaning	60	512	37	758	97	1270
Filtering by duplicates	116	396	144	614	260	1010
Filtering by # of pages	31	365	79	535	110	900
Filtering by paper title	252	113	477	58	729	171
Filtering by paper abstract	64	49	41	17	105	66
Filtering by full paper	34	15	10	7	44	22
Backward referencing	12		2			
Total		27		9		36

Table 1. Paper Selection Process

To derive the framework, we started with clustering the methods according to what they were aiming to improve  $(\mathbf{RQ_1})$ , e.g., "cycle time minimization", "cost optimization". We then noted that the methods formed two more prominent groups that, in the end, served as the main categorization of the framework, i.e., the objectives. Within the groups, we followed the research questions to classify the methods further, such as according to the interventions they trigger  $(\mathbf{RQ_2})$ , the input data they require  $(\mathbf{RQ_3})$ .

## 4 SLR Results

In the following sections, we present the results of our review. First, we describe the objectives of prescriptive process monitoring methods that we found  $(\mathbf{RQ}_1)$ . Then, we present the interventions prescribed to achieve the objectives  $(\mathbf{RQ}_2)$ , and, finally, we outline the data used to do so  $(\mathbf{RQ}_3)$ .

## 4.1 Prescriptive Process Monitoring Objectives

From our review, we identified two main objectives that prescriptive process monitoring methods aim to achieve. The first objective reduces the defect rate, whereas the second relates to optimizing quantitative case performance. The

<sup>&</sup>lt;sup>4</sup> Full review protocol: https://doi.org/10.6084/m9.figshare.17091554.v3

objective of reducing defect rate is expressed with binary metrics. For instance, the objective is achieved by reducing the risk of cost overrun [8]. The objective of optimizing is expressed as, for instance, reducing cycle time [45].

As to the objective of reducing the defect rate, five papers discuss undesired temporal outcomes, such as the violation of the planned cycle time or deadline [16,36,43,20,23]. For instance, Gröger et al. [16] describe the example of a manufacturing process, where the target is to avoid exceeding the allowed limits for cycle time. Another set of studies focuses on avoiding or mitigating an undesired categorical outcome [37,13,27,35,14,39,25,17]. For example, Ghattas et al. [14] try to avoid the customer rejecting the delivery in a bottle manufacturing process. In the domain of healthcare, examples of undesired outcomes are patients entering a critical stage [39], or medical mistakes due to patient restrictions [25]. One paper aims to eliminate or mitigate process risks, i.e., faults in the process that may arise if not addressed [8].

The second main objective considers optimizing quantitative case performance. Most papers consider optimizing the temporal perspective (15 out of 20), such as cycle or processing time. More specifically, in [45,21,31,40], reducing cycle time is defined as the main objective. For instance, Thomas et al. [40] describe a method to minimize the cycle time of an environmental permit application process. Others focus on processing time, i.e., time spent by a resource resolving a task [12]. For instance, in Park et al. [33] the aim is to reduce the processing time of manual tasks in a loan application process. Another set of methods aim at increasing quality. For example, a method seeks to increase perceived service quality for the users of a financial web service [44]. Finally, two papers [15,38] describe methods that aim to improve revenues, e.g., by increasing customer lifetime value [15]

#### 4.2 Prescribed Interventions

Prescriptive process monitoring methods prescribe actions to take, i.e., interventions. These interventions can be categorized according to the process perspective of the prescribed intervention. Our review indicates that interventions commonly concern control flow and resource perspectives.

A common intervention perspective is control flow, such as prescribing the next task to perform [23,18,6,29]. More specifically, in de Leoni et al. [23], the next best task is prescribed to the professional who helps a customer in finding a new job, whereas in Weinzierl et al. [44], the next step is presented to the end-user. Following the prescribed intervention can improve execution time, customer satisfaction or service quality. In other studies, a sequence of next steps is prescribed as an intervention. For instance, in one method, the appropriate treatment of a blood infusion is prescribed for patients based on their personal information [10], whereas another method prescribes steps to be taken in a trauma resuscitation process [47]. Such interventions aim to improve treatment quality.

Another group of methods focuses on the resource perspective, e.g., which resource should perform the next task. For instance, Wibisono et al. [45] prescribe which police officer is best suited for the next task in a driving license application

process based on their predicted performance. In another method, a mechanic is recommended to carry out car repairs because s/he is predicted to finish it within a defined time given their schedule and experience [36].

Some papers propose prescribing multiple interventions for one case [35,30,5]. For example, an intervention to make an offer to a client is prescribed together with a suggestion for a specific clerk to carry out the task [35]. Similarly, in an IT service management process, recommending the next task and the specialist to perform it can help to resolve open cases quicker [30].

When reviewing the identified papers, we noted that interventions could be divided into two aspects; intervention frequency and intervention basis. Intervention frequency captures when interventions are prescribed. In this sense prescriptive monitoring methods can be continuous or discrete. If the method is continuous, it prescribes an intervention for multiple or all activities of an ongoing case. For example, the best-suited resource for each next task is prescribed [45]. Discrete interventions, in comparison, prescribe actions to be taken only when a need is detected. For instance, in Metzger et al. [27], interventions are triggered only when it is detected that the probability of a negative outcome exceeds a defined threshold.

The intervention basis describes whether a method is *prediction*-based or *similarity*-based. Prediction-based methods predict the outcomes of an ongoing case and then prescribe an intervention. Similarity-based methods in comparison provide recommendations solely based on an analysis of historical traces. For instance, one method predicts the possible outcomes if a task is performed as the next step [23]. The next step is prescribed based on which option leads to a greater metric improvement. In contrast to this method, a set of actions are prescribed based on the similarity rate of a current ongoing case and similar previous cases in another method [41].

### 4.3 Required Data Input

Prescriptive process monitoring methods we identified use control flow, resource, temporal, and domain-specific data. Some methods focus on a single type of data, but other methods combine data input from different types.

As expected, methods that prescribe interventions impacting control flow, such as the next task to execute, commonly use control flow data. For example, in Conforti et al. [8], the authors apply decision trees on data, such as task duration and frequency, to predict the risk of a case fault, e.g., exceeding the maximum cycle time and costs overrun. Goossens et al. [15] prescribe the next task by using the sequence of events as a key feature.

Data on resources are used to trigger interventions related to different prescription perspectives. For instance, one method predicts the execution time of past resource performance [46]. The data on resource performance is used to reallocate pending work items to the resources with higher efficiency. In another method, the authors use resource roles and capabilities combined with domainspecific features, such as vehicle type, to recommend which mechanic should be assigned the next task [36]. The data is used to predict which resource would improve the probability of the vehicle repair being finished within a defined time.

Temporal data, e.g. day of the week, is also used to prescribe interventions. Such data is commonly used in combination with other data, such as control flow or resources. For instance, the best-suited resource to execute the next task is recommended utilizing the period of the day (morning, afternoon, or evening), inter-arrival rate, and task queue data as input [45]. In another method, temporal information (month, weekday, hour) of the last event and the inactive period before the most recent event in the log are used to evaluate the effectiveness of an intervention to reduce cycle time [7].

Domain-specific features, such as materials used in a manufacturing process [14], patient demographics and treatment attributes in a patient treatment process [47], are also utilized as data input. For instance, data about previously treated patients and data on the current patient is used to assess the predicted outcome of alternative next tasks [25]. This method recommends the task that has the best-predicted outcome for the patient to reduce the risk of medical mistakes, such as prescribing the wrong medication.

## 5 Framework and Implications

In this section, we present a framework for characterizing existing work on prescriptive process monitoring. Furthermore, we discuss the implications of our review and conclude with the limitations of the study.

#### 5.1 Framework Overview

The proposed framework (Figure 1) characterizes prescriptive process monitoring methods from ten aspects, derived from the review results (cf. Section 4). The framework<sup>5</sup> reads from left to right and begins with the aspect of objective. Other aspects include the interventions to reach the objective, the input required, the techniques used in the methods and the policies used to trigger interventions.

The main aspect of the framework is the *Objective*. Our analysis shows that the identified methods can be divided into two categories according to the objective they pursue (cf. Section 4.1). The first category aims to reduce the percentage of cases with a negative outcome (i.e. the defect rate). Methods in the second category aim to optimize a performance dimension captured via a quantitative performance metric defined at the level of each case (e.g. cycle time). The next aspect of the framework is the *Target*: the metric used to assess if the performance is improved by the prescribed interventions. For the objective of reducing the defect rate, the target may be a count of a categorical case outcome (e.g., customer complaints) or of a temporal outcome (deadline violations). Quantitative performance targets include cycle time, labor cost, or revenue.

The next two columns (*Prescription perspective, Intervention*) capture the interventions that the methods prescribe to pursue the defined objective. Thus,

<sup>&</sup>lt;sup>5</sup> Link to the full framework: https://doi.org/10.6084/m9.figshare.17091521.v1

Objective	Target	Prescription	Intervention	Input	Feature encoding	Modeling technique	Policy	-	Interventio	Referen
		herspective	Resource settings	All available	All case features		Exceeding metric limits	Continuous	Prediction	
		Resource			Resource experience, preference, collaboration,		2			2
	Temnoral		Resource allocation	R; T; D	utilization; domain-specific case features; time of day	Decision trees (prediction	Decision trees (prediction Highest predicted metric value	Continuous	Prediction	36
	outcome			T; C	Timestamps; activity sequence	Composite classification	Multiple options (ume-based, deadiline-based, decision-based, goal-based)	Continuous	Prediction	20
		Control flow		၁	Activity sequence; activity duration	LSTM	Exceeding metric limits, maximum metric improvement	Discrete	Prediction	43
			Next task to perform	Metric specific	Metric specific features	Random Forest, SVM, de	Random Forest, SVM, of Maximum metric improvement	Continuous	Prediction	23
			An alarm to trigger an intervention	All available	All data features, hour, weekday, month, time since case start, time since last event	Random forest, gradient	Probability of a negative outcome above a threshold (for k consecutive events), cost model (intervention, undesired outcome, compensation costs), mitigation effectiveness	Discrete	Prediction	13; 37
Reduce			Various interventions	C: T		Ensemble deep supervis	Probability of a negative outcome above a threshold, cost model Ensemble deep supervisi (adaptation, penalty, compensation costs), reliability estimate	Discrete	Prediction	27
		Varies (multiple)	Various interventions and resources for them	All available	All data features, number of offers, hour, day of the month, month	Probability of a nega XGBoost (prediction), Of resource availability	Probability of a negative outcome above a threshold, intervention cost, resource availability	Discrete	Prediction	35
	Categorical outcome		Various process decisions	D; C	Past process performance, task sequence, domain-specific case attributes	Decision trees	Highest predicted metric value	Continuous	Prediction	14
			Next task to perform and the resource	D; C; R	Task sequence, past resource performance, domain-specific case attributes	ModCNN, longest comm	Similarity based	Discrete	Similarity	39
			Task that can, should, or should not be executed next	٥	Domain-specific case features	Nearest neighbor	Similarity based	Continuous	Similarity	25
		Control flow		O	Past process executions, enabled activities	Nearest neighbor	Similarity based	Discrete	Similarity	17
	Risk of the fault		Next task to perform	S, R	Task durations and frequencies, resources	Decision trees (prediction	Decision trees (prediction Probability and severity of risk, scheduling algorithm	Discrete	Prediction	00
			Resource for the next	C; T	Queue, inter-arrival rate, timeframe	Naive Bayes model (pred	Naive Bayes model (pre∮ Highest predicted resource performance	Continuous	Prediction	45
	Cycle time		task	R; T	Co-working history, past resource performance	Model-less	Set of rules (resource availability, resource compatibility, processing time)	Continuous	Similarity	1
			acitacolla conscaed	R; C; T	Activity sequence, processing time, start time, finish time, instance weight	LSTM	Scheduling algorithm	Continuous	Prediction	33
	Processing time	Resource		R	Resource experience, past resource performance, resource workload	Nearest neighbor	Set of rules (resource availability, resource experience, processing time, processing cost)	Continuous	Similarity	4
			Team for the incoming task	R	Resource experience, past resource performance, resource workload, resource capabilities	Nearest neighbor	Set of rules (individual and in-team performance)	Continuous	Similarity	8
	Labor cost		Resource for the next task	R; C	Activity sequence, processing time, labor cost	Decision trees	Highest predicted resource performance	Continuous	Prediction	21; 31
	:		Pending work items reallocation from one resource to another	Ŗ. C; ⊤	Activity sequence, arrival rate, resource responsibilities, resource workload, past resource performance	Statistical analysis	Scheduling algorithm	Continuous	Prediction	46
	Cycle time			o	Activity sequence, past process performance	Model-less	Maximum metric improvement	Continuous	Prediction	29
Optimize				C; T; D	Activity sequence occurence, activity duration, amounts of damage	State model	User-defined function	Continuous	Similarity	18
quantitative	Dougonio		Next task to perform	C	Activity sequence	SVM	Maximum metric improvement + similarity	Continuous	Prediction, s	15
performance		Control flow		D; C	Domain-specific case features, activity sequence	KNN, ALS, BPR	Similarity based	Continuous	Similarity	38
	Processing time			D; R	Domain-specific case features, resource experience, processing time	Probabilistic model	Set of rules (resource availability, type of repair, processing time and cost)	Continuous	Prediction	9
				D	Domain-specific case features	Decision trees	Similarity based + internal and external rules	Continuous	Similarity	10
	Onality			D; C	Domain-specific case attributes	Nearest neighbor	Similarity based	Continuous	Similarity	47
	diam's		Set of tasks to perform	D; C	Domain-specific case attributes	Nearest neighbor	Similarity based	Continuous	Similarity	41
			Next clicks in the process	D; C	Activity sequence, domain-specific case features	LSTMs	Probability above a thershold	Continuous	Prediction	44
				R; C; T	Activity sequence, resources, domain-specific case features (case type, priority, status, tags)	Nearest neighbor	Similarity based	Continuous	Similarity	30
			Next task to perform	C; R	Activity frequency, processing time	Decision trees	Similarity based	Continuous	Similarity	40
	Cycle time	(multiple)	and resource to perform it	C; R	Activity duration, resource availability, number of instances	Constraint programming	Set of rules (optimized plans, resource availability)	Continuous	Similarity	2
			Various interventions	F. R	Temporal information of the last event, workload, difference between start time of the case and start time of the timeframe	Orthogonal random fores User-defined policy	User-defined policy	Discrete	Prediction	7

Fig. 1. Prescriptive process monitoring framework

Prescription perspective describes to which process aspect the intervention concerns, e.g., resource or control flow. We also included the category "multiple" for methods that describe several interventions. Then, the aspect *Intervention* lists the actual interventions (cf. section 4.2). For instance, actual intervention can be which resource to assign to the next task.

The following four columns define the data (Input perspective, Feature encoding), techniques (Modeling technique), and policies (Policy) to trigger interventions. Namely, *Input perspective* describes the types of features, i.e., input data, required for a specific method (cf. section 4.3). Thus, the categories we elicited are (C) control flow (e.g., activities, sequence, and frequencies), (R) resources (e.g., performers of activities), (T) temporal features (time-related), and (D) domain-specific (features that depend on the domain or type of process). The aspect *Feature encoding* explains how features are further refined by a prescriptive method. For instance, resource-perspective features can be encoded as resource experience, resource performance, or resource workload.

The aspect *Modeling technique* relates to the technique used in the method to predict the outcome of the process or the metric performance based on the input. Next, *Policy* relates to the conditions under which an intervention is prescribed. For example, under a similarity-based policy, an intervention is prescribed based on the similarity of the current case to previous cases. Some methods take a set of rules as their policy. For example, the need for an intervention is detected when the probability of a negative outcome exceeds a defined threshold, but also the effectiveness of the intervention is assessed before prescribing it.

The aspect *Intervention frequency* shows whether a method is continuous (prescribes actions at every step) or discrete (only when the need is detected). Additionally, *Intervention basis* describes whether a method builds on the prediction of how the current case will continue or its similarity to the past cases (cf. section 4.2). Finally, the aspect *Example* (see full version of the framework) can be used as a reference to how the introduced method with its inputs, technique, and policies was used to trigger interventions to reach the objective of a process in a specific domain.

Existing methods could be explored from the framework by objective, target, and prescribed process perspective. As such, if one seeks to minimize the cycle time of a process, the aim is to optimize quantitative case performance (Objective), more specifically, cycle time (Target). The framework shows that this can be achieved by prescribing interventions related to the control-flow or resources of the respective process (Prescription perspective). If we follow the resource perspective, the framework shows that a set of methods can recommend, for example, resources for the next task or a whole team for a specific request (Intervention). For instance, two methods [45,1] propose to recommend resources for the next task as an intervention. However, they have different input perspectives and techniques. Continuing with selecting control flow and temporal as the input perspectives, it leads to the method which uses the predicted (Intervention basis) highest resource performance (Policy) to continuously (Intervention frequency) prescribe a resource for the next task.

### 5.2 Research Gaps and Implications

The presented framework provides an overview of existing prescriptive process monitoring methods by categorizing them according to their objectives and targets. The framework also presents different methods available and different ways these methods enable reaching particular objectives. The overview unveils several gaps and associated implications for future research.

First, we observe that in the vast majority of previous studies, the proposed prescriptive process monitoring methods have been tested using synthetic and/or real-life event logs. However, the validation of the method is done using a real-world or synthetic observational event log, but not in real-life settings. An attempt to test the effectiveness of interventions in real-life settings was made by Dees et al. [9]. Their study showed that the predictions were rather accurate, but the interventions did not lead to the desired outcomes. Thus, validations of methods should be done in real-life settings to ensure their usefulness in practice, as also previously highlighted by [28].

Second, the current state-of-the-art in the field is focused on identifying cases in which interventions should be applied and finding the point in time an intervention should be triggered during the execution of a case. In contrast, little attention has been given to the problem of discovering which interventions could be applied to optimize a process with respect to a performance objective. Discrete methods leave it up to the users (stakeholders) to define the possible intervention(s) a priori (e.g., [23,37]). Those methods that use observational event logs from BPI challenges<sup>6</sup>, rely on winner reports to identify the possible interventions (e.g., [35,37]). Continuous methods, on the other hand, focus on recommending the next task(s) (e.g., [15,6,47]) or resource allocation (e.g., [3,1,21]). Thus, one direction for further research is designing methods to support the discovery of interventions from business process event logs, or from textual documentation, or other unstructured or structured metadata about the process.

Related to the above problem of discovering interventions, another gap in existing research relates to the problem of designing and tuning policies for prescriptive process monitoring. Existing prediction-based methods (e.g., [36,16]) prescribe an intervention when the probability of a negative outcome exceeds a defined threshold. However, because the predictive models upon which methods rely are based on correlation (as opposed to causal relations), the prescriptions produced by this method might not address the cause of the negative outcome or poor performance (e.g. the cause of delay). In this respect, we note that only a few existing methods take into account causality in policy design (e.g., [35,7]). Thus, developing policy design techniques that take into account causality is another direction for further research.

As discussed in [9,23], the choice of whether or not to apply an intervention or which intervention to apply often depends on contextual factors. Some interventions may lead to prove ineffective or counter-productive, for example, due to second-order effects. For example, an intervention wherein a customer is

<sup>6</sup> https://www.tf-pm.org/competitions-awards/bpi-challenge

contacted pro-actively in order to prevent a customer complaint may actually heighten the probability of a complaint [9]. Similarly, assigning a resource to a case that is running late might lead to other cases being neglected, thus creating delays elsewhere and thus leading to a higher ratio of delayed cases. Detecting such second-order effects requires human judgment and iterative policy validation (e.g. via AB testing). In this respect, it is striking that existing prescriptive process monitoring methods do not take into account the need to interact with human decision-makers. A crucial step in this direction is the ability to explain why the prescriptive monitoring system recommends a given intervention. There are two aspects to this question. First, explaining the prediction that underpins a given prescription (prediction explanation), and second, explaining the policy that is used to trigger the prescription (policy explanation). A possible direction to enhance the applicability of prescriptive monitoring methods in practice is to integrate explainability mechanisms into them. While several proposals have been made to enhancement the explainability of predictive process monitoring methods [19], the question of policy explanation in the area of prescriptive process monitoring is unexplored. In other words, current methods do not incorporate a mechanism to explain why an intervention is recommended by the method for a given case and in a given state.

Besides the above gaps, the SLR highlights that the majority of methods in the field aim to improve processes along the temporal perspective (e.g. cycle time, processing time, deadline violations). In comparison, other performance dimensions are represented in only a few examples (quality in [10,44,47], revenue in [15,38]). Thus, another research direction is to investigate other performance objectives that could be enhanced via prescriptive process monitoring.

Finally, our review also highlights a lack of common terminology in the field. This might be due to the novelty of the research field of prescriptive process monitoring. The literature refers to methods with a wide range of names. As such, the terms "proactive process adaptation", "on-the-fly resource allocation", "next-step recommendation" are all used to describe the development and application of prescriptive process monitoring methods. This highlights the need for common terminology.

### 5.3 Threats to Validity

SLRs have a number of typical pitfalls and threads to validity [22,2]. First, there is a potential risk of missing relevant publications during the search. We mitigated this risk by conducting a two-phase search that included a broad range of key terms, as well as backward referencing. Another potential threat is to exclude relevant publications during screening. We mitigated this threat by using explicitly defined inclusion and exclusion criteria. Furthermore, all unclear cases were examined and discussed by all authors of this paper. Third, there is a threat of data extraction bias as this step involves a degree of subjectivity. We discussed each paper in the final list and refined the data extraction when needed to minimize this risk.

### 6 Conclusion

This paper provided a snapshot of the research field of prescriptive process monitoring via an SLR and outlined a framework for characterizing methods in this field. The framework characterizes existing methods according to their objective, target metric, intervention type, technique, data input, and policy used to trigger interventions. The framework was derived from and used to characterize the 36 relevant studies identified by the SLR.

Based on the SLR, we identified research gaps and associated research avenues. In particular, the SLR highlighted: (i) a lack of *in vivo* validation of the proposed methods; (ii) a lack of methods for discovering suitable interventions and assessing their potential effectiveness; (iii) little emphasis on explainability and feedback loops between the prescriptive monitoring system and the endusers; and (iv) a narrow focus on temporal metrics and comparatively little work on applying prescriptive monitoring to other performance dimensions.

**Acknowledgments.** This research is funded by the Estonian Research Council (PRG1226) and the European Research Council (PIX Project).

### References

- Abdulhameed, N., Helal, I., Awad, A., Ezat, E.: A resource recommendation approach based on co-working history. International Journal of Advanced Computer Science and Applications 9(7), 236–245 (2018)
- Ampatzoglou, A., Bibi, S., Avgeriou, P., Verbeek, M., Chatzigeorgiou, A.: Identifying, categorizing and mitigating threats to validity in software engineering secondary studies. Information and Software Technology 106, 201–230 (2019)
- 3. Arias, M., Munoz-Gama, J., Sepúlveda, M.: A multi-criteria approach for team recommendation. In: Business Process Management Workshops. pp. 384–396 (2016)
- Arias, M., Saavedra, R., Marques, M.R., Munoz-Gama, J., Sepúlveda, M.: Human resource allocation in business process management and process mining: A systematic mapping study. Management Decision (2018)
- 5. Barba, I., Weber, B., Valle, C.D.: Supporting the optimized execution of business processes through recommendations. In: Business Process Management Workshops (1). pp. 135–140. Springer (2011)
- Batoulis, K.: Proactive decision support during business process execution. In: EMoV+MinoPro@Modellierung. CEUR Workshop Proceedings, vol. 1185, pp. 35–41. CEUR-WS.org (2014)
- 7. Bozorgi, Z.D., Teinemaa, I., Dumas, M., La Rosa, M., Polyvyanyy, A.: Prescriptive process monitoring for cost-aware cycle time reduction. In: 2021 3rd International Conference on Process Mining (ICPM). pp. 96–103. IEEE (2021)
- 8. Conforti, R., de Leoni, M., Rosa, M.L., van der Aalst, W.M.P., ter Hofstede, A.H.M.: A recommendation system for predicting risks across multiple business process instances. Decis. Support Syst. **69**, 1–19 (2015)
- 9. Dees, M., de Leoni, M., van der Aalst, W.M.P., Reijers, H.A.: What if process predictions are not followed by good recommendations? In: BPM (Industry Forum). CEUR Workshop Proceedings, vol. 2428, pp. 61–72. CEUR-WS.org (2019)
- Detro, S.P., Santos, E.A.P., Panetto, H., Loures, E.D.F.R., Lezoche, M., Barra, C.M.C.M.: Applying process mining and semantic reasoning for process model customisation in healthcare. Enterp. Inf. Syst. 14(7), 983–1009 (2020)

- 11. Di Francescomarino, C., Ghidini, C., Maggi, F.M., Milani, F.: Predictive process monitoring methods: Which one suits me best? In: International Conference on Business Process Management. pp. 462–479. Springer (2018)
- Dumas, M., Rosa, M.L., Mendling, J., Reijers, H.A.: Fundamentals of Business Process Management, Second Edition. Springer (2018)
- Fahrenkrog-Petersen, S.A., Tax, N., Teinemaa, I., Dumas, M., de Leoni, M., Maggi, F.M., Weidlich, M.: Fire now, fire later: alarm-based systems for prescriptive process monitoring. arXiv preprint arXiv:1905.09568 (2019)
- Ghattas, J., Soffer, P., Peleg, M.: Improving business process decision making based on past experience. Decision Support Systems 59, 93–107 (2014)
- 15. Goossens, J., Demewez, T., Hassani, M.: Effective steering of customer journey via order-aware recommendation. In: ICDM Workshops. pp. 828–837. IEEE (2018)
- Gröger, C., Schwarz, H., Mitschang, B.: Prescriptive analytics for recommendationbased business process optimization. In: International Conference on Business Information Systems. pp. 25–37. Springer (2014)
- 17. Haisjackl, C., Weber, B.: User assistance during process execution an experimental evaluation of recommendation strategies. In: Business Process Management Workshops. pp. 134–145. Springer (2010)
- Heber, E., Hagen, H., Schmollinger, M.: Application of process mining for improving adaptivity in case management systems. In: DEC. LNI, vol. P-244, pp. 221–231. GI (2015)
- 19. Hsieh, C., Moreira, C., Ouyang, C.: Dice4el: Interpreting process predictions using a milestone-aware counterfactual approach. In: 3rd International Conference on Process Mining, ICPM. pp. 88–95. IEEE (2021)
- Huber, S., Fietta, M., Hof, S.: Next step recommendation and prediction based on process mining in adaptive case management. In: S-BPM ONE. pp. 3:1–3:9. ACM (2015)
- 21. Kim, A., Obregon, J., Jung, J.Y.: Constructing decision trees from process logs for performer recommendation. In: International Conference on Business Process Management. pp. 224–236. Springer (2013)
- 22. Kitchenham, B.A., Charters, S.: Guidelines for performing systematic literature reviews in software engineering. Tech. Rep. EBSE 2007-001 (07 2007)
- de Leoni, M., Dees, M., Reulink, L.: Design and evaluation of a process-aware recommender system based on prescriptive analytics. In: 2nd International Conference on Process Mining, ICPM. pp. 9–16. IEEE (2020)
- 24. Maggi, F.M., Di Francescomarino, C., Dumas, M., Ghidini, C.: Predictive monitoring of business processes. In: International conference on advanced information systems engineering. pp. 457–472. Springer (2014)
- Mertens, S.: Personalized process and decision support in emergency care by mining electronic health records and sensor data. In: PervasiveHealth. pp. 415–418. ACM (2020)
- 26. Mertens, S., Gailly, F., Sassenbroeck, D.V., Poels, G.: Comparing strategies to generate experience-based clinical process recommendations that leverage similarity to historic data. In: 2019 IEEE International Conference on Healthcare Informatics (ICHI). pp. 1–11 (2019)
- 27. Metzger, A., Kley, T., Palm, A.: Triggering proactive business process adaptations via online reinforcement learning. In: International Conference on Business Process Management. pp. 273–290. Springer (2020)
- Márquez-Chamorro, A.E., Resinas, M., Ruiz-Cortés, A.: Predictive monitoring of business processes: A survey. IEEE Transactions on Services Computing 11(6), 962–977 (2018)

- Nakatumba, J., Westergaard, M., van der Aalst, W.M.: A meta-model for operational support. BPM Center Report BPM-12-05, BPMcenter. org pp. 16-32 (2012)
- 30. Nezhad, H.R.M., Bartolini, C.: Next best step and expert recommendation for collaborative processes in IT service management. In: BPM. Lecture Notes in Computer Science, vol. 6896, pp. 50–61. Springer (2011)
- 31. Obregon, J., Kim, A., Jung, J.Y.: Dtminer: A tool for decision making based on historical process data. In: Asia-Pacific Conference on Business Process Management. pp. 81–91. Springer (2013)
- 32. Okoli, C., Schabram, K.: A guide to conducting a systematic literature review of information systems research. SSRN Electronic Journal 10 (2010)
- 33. Park, G., Song, M.: Prediction-based resource allocation using LSTM and minimum cost and maximum flow algorithm. In: ICPM. pp. 121–128. IEEE (2019)
- Pufahl, L., Ihde, S., Stiehle, F., Weske, M., Weber, I.: Automatic resource allocation in business processes: A systematic literature survey. CoRR abs/2107.07264 (2021)
- 35. Shoush, M., Dumas, M.: Prescriptive process monitoring under resource constraints: A causal inference approach. arXiv preprint arXiv:2109.02894 (2021)
- 36. Sindhgatta, R., Ghose, A.K., Dam, H.K.: Context-aware analysis of past process executions to aid resource allocation decisions. In: CAiSE. Lecture Notes in Computer Science, vol. 9694, pp. 575–589. Springer (2016)
- 37. Teinemaa, I., Tax, N., de Leoni, M., Dumas, M., Maggi, F.M.: Alarm-based prescriptive process monitoring. In: BPM (Forum). pp. 91–107. Springer (2018)
- 38. Terragni, A., Hassani, M.: Optimizing customer journey using process mining and sequence-aware recommendation. In: SAC. pp. 57–65. ACM (2019)
- Thomas, L., Kumar, M.M., Annappa, B.: Recommending an alternative path of execution using an online decision support system. In: Proceedings of the 2017 International Conference on Intelligent Systems, Metaheuristics & Swarm Intelligence (2017)
- 40. Thomas, L., Kumar, M.M., et al.: An online decision support system for recommending an alternative path of execution. In: Intl. Conference on Electrical, Computer and Communication Technologies (ICECCT). pp. 1–9. IEEE (2017)
- 41. Triki, S., Saoud, N.B.B., Dugdale, J., Hanachi, C.: Coupling case based reasoning and process mining for a web based crisis management decision support system. In: WETICE. pp. 245–252. IEEE Computer Society (2013)
- 42. Van Der Aalst, W.: Process mining. Commun. ACM 55(8), 76-83 (Aug 2012)
- 43. Weinzierl, S., Dunzer, S., Zilker, S., Matzner, M.: Prescriptive business process monitoring for recommending next best actions. In: BPM (Forum). pp. 193–209. Springer (2020)
- 44. Weinzierl, S., Stierle, M., Zilker, S., Matzner, M.: A next click recommender system for web-based service analytics with context-aware lstms. In: HICSS. pp. 1–10. ScholarSpace (2020)
- 45. Wibisono, A., Nisafani, A.S., Bae, H., Park, Y.: On-the-fly performance-aware human resource allocation in the business process management systems environment using naïve bayes. In: AP-BPM. pp. 70–80. Springer (2015)
- 46. Yaghoibi, M., Zahedi, M.: Cycle time reduction and runtime rebalancing by real-locating dependent tasks. International Journal of Engineering **30**(12), 1831–1839 (2017)
- 47. Yang, S., Dong, X., Sun, L., Zhou, Y., Farneth, R.A., Xiong, H., Burd, R.S., Marsic, I.: A data-driven process recommender framework. In: KDD. pp. 2111–2120. ACM (2017)