Expertise versus Data: Comparison of Expertise Based Process Models to Process-Mining Models of Surgery Under General Anesthesia

Hugo Boisaubert[1][a], Chloé Grivaud[1], Antoine Bouchet[1], Corinne Lejus-Bourdeau[2,3] and Christine Sinoquet[1]

[1]*Nantes Université, CNRS, LS2N UMR 6004, 2 Chemin de la Houssinière, 44300 Nantes, France*

[2]*Nantes Université, LESiMU, 9 rue Bias, 44000 Nantes, France*

[3]*Department of Anaesthesia and Intensive Care, CHU Nantes, 1 place Alexis Ricordeau, 44000 Nantes, France*

Keywords: Process-Mining, Anesthesia, Healthcare, Model Comparison, Machine Learning, Expertise, Personal Data.

Abstract: Digital tools accessible for healthcare are often based on models representing a medical process and learned from medical data. Unfortunately, those data are protected by privacy regulation and therefore are quite rare. This rarity leads to process models mainly based on the expertise of caregivers. Those expertise-based model and data-based models are rarely compared to show their common characteristics and differences. When both model can be produced for the same situation multiple questions arise. Should the expertise-based model be invalidated if it is not in full conformity to the data-based model ? Are those models' characteristics the same? In this article, we present a comparison of expertise-based models and data-based models produced for a surgery under general anesthesia with 204 real cases. We conducted a process mining algorithm performance comparison on our specific real data to identify the most promising learning method. Then we compared the produced data-based models to the expertise-based models with some metrics. The comparison results show strong differences between the two types of models, the expertise-based model is very much smaller than the data-based model, but we have noticed that the expert-based model is included in a data-based model. Therefore, the main difference between the two models appears to be on a level of abstraction.

1 INTRODUCTION

Anesthesia is a common practice in occidental medicine to facilitate specific care or medical gestures. More than 300 million anesthesias are induced every year in the world (Gao et al., 2022).

The development in digital sciences plays a role in this improvement with new tools for caregivers, first with decision aid software (O'Connor et al., 1999) for anesthesia consultation, with the involvement of digital simulation of patients in training of caregivers (Kononowicz et al., 2019), and most recently with digital twins (Katsoulakis et al., 2024).

Many of those tools are based on process models that represent the medical situation where the tool is intended to be used, from organization process models for a full hospital to medical treatment process models for the care of a patient (Kaymak et al., 2012).

However, before giving access to caregivers of this kind of tool, those models must be accurately produced. This can be easily achieved by process mining and machine learning techniques based on real medical data. Unfortunately, those kinds of data are quite rare, as they are highly sensitive patient data and therefore protected by privacy regulations like the GPDR[1]. Consequently, models are usually built with the help of caregivers and experts based on their expertise.

This work is born inside a collaboration between the Laboratory of Digital Sciences of Nantes University and the Nantes University Hospital. During this collaboration, aiming to develop new methods for caregivers in general anesthesia training with the help of digital tools, medical data should have been used to learn process models by machine-learning techniques and process mining approaches.

However, because of the private and sensitive nature of those data, access was not easily granted. Thus, a modeling effort, involving anesthesia experts, trainers, and caregivers, has been carried out, re-

[a] https://orcid.org/0000-0002-1855-0727

[1]The *General Data Protection Regulation* (GPDR), is the European Union regulation on the usage of data.

Boisaubert, H., Grivaud, C., Bouchet, A., Lejus-Bourdeau, C. and Sinoquet, C.
Expertise versus Data: Comparison of Expertise Based Process Models to Process-Mining Models of Surgery Under General Anesthesia.
DOI: 10.5220/0013262300003911

In *Proceedings of the 18th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2025) - Volume 2: HEALTHINF*, pages 742-749
ISBN: 978-989-758-731-3; ISSN: 2184-4305

sulting in expertise-based process models of general anesthesia.

When access to the precious health data was, at least, granted, the question of the conformity of expertise-based process models to real data produced by the real medical procedure has arisen.

The aim of this work is to compare different expertise-based process models to data-based process models in healthcare situations, more precisely in a general anesthesia situation. To our knowledge, this work of comparison has not yet been done.

This article will first present our medical situation, then our application context and approach, and finally our results and an analysis where we will try to determine how both types of models compare to, hopefully, lead us to a proposition of usage.

2 A SPECIFIC MEDICAL CONTEXT

Application of process mining approaches for health is now a full research domain (known as *PM4H*) with distinct characteristics and challenges (Munoz-Gama et al., 2022). Experimental results and specific approaches to process mining for health have been published in the last decade (Rojas et al., 2016)(Guzzo et al., 2022), including multiple results on anesthesia (Kaymak et al., 2012).

In the case of surgery performed under general anesthesia, the process is directly dependent on the specificities of the patient. Although there is a common framework for all interventions under general anesthesia, there is no reference process model to serve as the ground truth, just recommendations for good practices from health authorities or national anesthesia society.

2.1 General Anesthesia

General anesthesia consists of the use of powerful *analgesics*, such as drugs from the opioid family, like morphine, on an unconscious patient. It is recommended when the patient cannot consciously endure a care. The loss of consciousness in a patient involves the use of specific medications, such as *hypnotics* - substances capable of inducing and/or maintaining sleep - and specialized medical procedures to initiate and maintain it.

Depending on the depth of sedation - the loss of vigilance and consciousness through the use of medication - the patient may lose their respiratory reflex. It is therefore necessary to intubate them and connect them to a mechanical respirator to provide respiratory assistance. Throughout the operation, anesthesia is maintained by continuous or regular injection of drugs.

A general anesthesia involves many risks. The use of curare is the cause of the majority of allergic risks for that kind of care, but the use of certain hypnotics can lead, in very rare cases, to serious and generally unpredictable allergic reactions.

The high variability of these procedures leads to high variability in the data they produce and therefore hinder the ability of process mining algorithms to generate models. Handling this variability is one of the challenges for PM4H (Munoz-Gama et al., 2022).

2.1.1 A Semi-Rigid Structure

General anesthesia is administered in four main steps:

- **Patient Entry:** the patient arrives in the operating room and is prepared for the procedure.
- **Anesthesia Induction:** analgesia is performed and loss of consciousness is induced. The patient is placed on respiratory assistance.
- **Surgical or Medical Procedure:** the actions planned during the procedure are performed. Loss of consciousness and analgesia are maintained.
- **Patient Exit:** maintenance of anesthesia is stopped and the patient is prepared for awakening.

The four major steps of an anesthesia procedure correspond to different moments of activity of the intervention and are always present. For each of these major steps, specific steps are carried out by caregivers in a determined and fixed order. This is the **rigid** component of the structure of anesthesia. Steps for the first main step are presented in Table 1.

Table 1: First Main Step, Steps and Sub-steps of a general anesthesia.

Main step	Step	Sub-step
Patient entry	Patient set up	Enter in room
		Set up on table
	Monitoring set up	Heart rate
		Blood pressure
		O2 saturation
		Curare level
	Preparation	Bair Hugger
		Venous route
		Prophylaxis

Each of these actions, relating to care, is composed of sub-steps that correspond to different activities dedicated to medical care. They are carried out by the medical team and vary according to the intervention and the patient's profile. This is the **variable** component of the structure of anesthesia.

2.2 Anesthesia Event Logs

The various actions carried out by the medical team during the main steps, steps and sub-steps of an anesthesia procedure are recorded as *event logs* during the monitoring of the intervention.

Since 1994 (Journal Officiel, 1994), in France all anesthesia procedures must be monitored, and all events and physiological time series of the patients must be archived for forensic reasons.

Since 2000, forensic archiving has been increasingly digitalized in more and more hospitals. Those archived data form an important database of real-world anesthetic data. For this work, only the event log of the surgery has been used as raw data. Those event logs are composed of the timestamp of the event, the event name, and sometimes a comment added by the medical team. Some examples of event log tuples are shown in Table 2.

Table 2: Example of an anesthetic event log.

Timestamp	Event name
21/02/2022 09:30:27	Patient set up
21/02/2022 09:30:57	Heart rate monitoring
21/02/2022 09:31:27	Blood pressure monitoring

3 COMPARED MODELS

In this work, we seek to represent general anesthesia situations during surgery in the form of a process model. Whether these process models come from machine learning techniques or from the expertise of caregivers, they aim to represent the same situations.

3.1 Process Models

A process model is a representation of a process. This kind of modeling is used for multiple tasks, such as discussing processes, document workflow, workflow verification, performance analysis, etc. (Van Der Aalst, 2016).

3.1.1 Concepts

Each process model is composed of a set of *activities* A. With a process model, we aim to represent which activities will be executed and the sequential order of execution during the process.

Therefore, a process model is also composed of a set of *transitions* T that define the sequential order of execution. Process models are represented as an oriented graph where activities and transitions are nodes linked by arcs. Some activities or chains of activities may be optional in the processes, or concurrent; we can therefore identify specific transitions into two categories: *splits* and *gateway*. Split transitions define multiple possible split-choices after an activity:

- **AND-split** imposes multiple next activities;
- **XOR-split** imposes to select one of the multiple next activities (the choice is made after the evaluation of specifically defined conditions);
- **OR-split** OR-split permits to select one or more in the multiple next activities (here too, the choice is made after the evaluation of specifically defined conditions).

The join transitions (i.e., gateways) work in a symmetric way to define an activity joining multiple possible preceding activities:

- **AND-join** requires multiple preceding activities to be completed before the next activity can be executed;
- **XOR-join** XOR-join requires one of the previous activities to be completed before executing the next activity;
- **OR-join** requires one or more of the previous activities to be executed before the next activity can take place.

3.1.2 Notation

Process model notation is usually a graphical representation of the model; those notations are plethorous, and it is relatively easy to translate a process model in a specific notation to another one (van Der Aalst et al., 2003) with tools dedicated to this task in reference process mining toolkit ProM (Verbeek et al., 2010).

The most used notations are Petri Net and BPMN (*Business Process Model and Notation*), as process mining has a good proximity with business modeling.

3.2 Expertise-Based Process Models

The models obtained from experts were constructed based on various discussion sessions with experts combined with reference sources, such as bibliographic sources, recommendations from anesthesia societies, or internal good practice references from hospitals.

The models were iteratively improved from session to session using a top-down approach. A first model corresponding to a generic anesthesia situation was formulated, and this first model was then specified by successively adding cases specific to certain patients, certain medical procedures, or medications.

3.3 Data-Based Process Models

Data-based processes are process models constructed using machine learning techniques from real data, specifically the event log of an anesthetic procedure. Those techniques are part of the process mining field.

3.3.1 Datasets

All the process models have been learned on anesthetic data from the Nantes University Hospital.

The used dataset is composed of the event log extracted from a forensic archive. The cohort of patients used to produce this dataset is composed of 204 male patients, aged between 30 and 50 years, with no specific medical history or comorbidities, receiving a surgery under general anesthesia for the curation of an inguinal hernia under laparoscopy.

A dataset consisting of 204 cases may seem too small for any machine learning approach. However, this is a constraint when using real medical datasets. Synthetic data could be used, but that would have undermined the value of our comparative approach. Handling ”small” datasets, in regard to habitual datasets in machine learning, is one of the challenges of PM4H algorithms (Munoz-Gama et al., 2022). Different characteristics of the dataset are shown in the Table 3

Table 3: Multiple characteristics of the used dataset.

Characteristic	Values
Number of Cases	204
Number of Activities	144
Mean Activities per event log	34.8
Min/Max Activities per event log	27/45
Average Trace Time (seconds)	8166
Min/Max Trace Time (seconds)	2048/18654

3.3.2 Selection of Some Process Mining Methods

There are many algorithmic approaches to learn process models by process mining techniques. Algorithms dedicated to process discovery are numerous. In order to determine which approaches of process discovery would be most suitable for anesthetic event logs, a small comparative study was carried out. We selected 6 different process discovery algorithms, based on their usage in literature or performance in healthcare process mining (Guzzo et al., 2022)(Munoz-Gama et al., 2022). We will introduce them in more detail in the next subsections.

All those algorithms are implemented in Java, mostly inside the reference tool ProM (Verbeek et al., 2010), except for the Split Miner algorithm. For some of them, the PM4PY(Berti et al., 2023) library has been used as a Python interface.

The **Alpha+ algorithm** (De Medeiros et al., 2004) is an upgrade of the original alpha algorithm, with the ability to mine short loops (De Medeiros et al., 2004) . The Alpha+ algorithm is a formal approach to process mining. This approach presupposes a perfect event log (Weijters et al., 2006) where the log is complete, which means that if an activity can follow another activity directly, the log should contain an example of this behavior. Besides, the log contains no noise, meaning everything is correctly recorded. Real-life event logs are very rarely noise-free and/or complete. The use of this algorithm with our data, if computation is possible, will probably show its lack of performance with real data. However, as it is a reference algorithm, we have selected it for this study.

The **Heuristics Miner** algorithm (Weijters et al., 2006) is an upgrade of the alpha+ algorithm, using the frequency of a behavior in an event log to be able to deal with noise and low-frequent behavior. The main advantage of the heuristics miner is its ability to deal with noise. However, it is only able to express the main behavior registered in an event log, not all details and exceptions.

The **Inductive Miner** (Leemans et al., 2014)uses a divide-and-conquer approach to discover a process model from an event log. The log is split into sublogs in order to discover an operator between them (Leemans et al., 2014). The process model discovered is a process tree, easily convertible into a Petri-Net or a BPMN notation. For this algorithm, a noise threshold must be selected; the values chosen for this parameter are discussed in the result section.

The **ILP Miner** (*Integer Linear Programming Miner*) (Van der Werf et al., 2008) uses a region-based approach. The main goal of this algorithm is to search for as many places as possible, such that the resulting Petri net is consistent with the log (i.e able to replay the log).

The **ETM Miner** (*Evolutionary Tree Miner*) (Buijs, 2014) uses an evolutionary approach to create, evaluate, and change candidate solutions. The evaluation is always done using the four quality dimensions used in process discovery (see 3.3.3). This approach is quite flexible because it allows for the selection of evaluation criteria that we want to maximize.

The **Split Miner** (Augusto et al., 2017) algorithm is designed to generate understandable and accurate

Table 4: Comparison of the selected process discovery algorithms.

Algorithms	Noise sensitivity	Duplicate tasks detection	Hidden tasks detection	No-free choice construct	Loops detection
Alpha+ Algorithm	Sensitive	No	No	No	Small loops
Heuristic Miner	Less sensitive	No	No	Partially	Yes
Inductive Miner	Robust	Yes	No	Yes	Yes
ILP Miner	Very sensitive	No	Partially	Yes	Yes
ETM Miner	Robust	Yes	Yes	Yes	Yes
Split Miner	Robust	Yes	No	Yes	Yes

process models. It stands out for its ability to handle complex transitions and concurrent behaviors by splitting the event log into partitions based on dependency relations, then creating local models for each partition, and finally integrating the local models into a global process model using logical operators.

A comparison of the process discovery algorithms selected for this study is presented in Table 4.

3.3.3 Process Mining Model Evaluation Criteria

The models generated using the different algorithms presented earlier were evaluated using four different metrics : fitness, precision, generalization, and simplicity.

The **fitness** metric calculates how many behaviors from the event log are covered by the model. The token-based replay fitness method returns the percentage of traces that are complete in the model (Berti and van der Aalst, 2019).

The **precision** metric involves replaying different parts of the event log on the model to check if the model is underfitting the log. To compute precision, we used the heuristic-based (Munoz-Gama and Carmona, 2010) method, faster but not exact.

The **generalization** metric measures whether the elements of a model are sufficiently visited when the event log is replayed on the model. The generalization measure is the one given in (Buijs et al., 2014).

The **simplicity** metric is calculated using an inverse arc degree method described by (Blum, 2015).

3.4 Comparison of Process Models

In this work, we are taking a similar approach to the search for conformity. However, it is not a question of comparing a learned model with a reference model assumed to be true (ground truth) but of comparing two models in order to highlight their differences and deduce the impact on the possible use of this model.

There are several approaches to checking the conformance of a process model: alignments, comparing footprints (Van der Werf et al., 2008). However, those approaches are mainly used to find commonalities and discrepancies between a process model and an event log to then improve the process model.

Other approaches are more user-friendly to compare two different process models. In this work we will use two metrics of distinguishability : Support and Confidence (Kuo and Chen, 2012). Those metrics are complementary and aim to quantify the similarity of the compared models.

The **Support** metric aims to quantify the resemblance, from a relational point of view, of the activities of both process models; For two process models a and b, this metric is computed as:

$$Support(a,b) = \frac{A_{a\cap b}}{A_{a\cup b}} \times \frac{T_{a\cap b}}{T_{a\cup b}} \quad (1)$$

where :

- $A_{a\cap b}$ is the number of the common activities in process models a and b ;
- $A_{a\cup b}$ is the number of activities in process models a and b ;
- $T_{a\cap b}$ is the number of the common transitions in process models a and b ;
- $T_{a\cup b}$ is the number of transitions in process models a and b.

The **Confidence** metric computes a ratio of sequential activities (i.e a transition) common to both models with the number of transitions in a specific model. For two process models a and b, this metric is computed as:

$$Confidence(b|a) = \frac{T_{a\cap b}}{T_a} \quad (2)$$

where :

- $T_{a\cap b}$ is the number of the common transitions in process models a and b ;
- T_a is the number of transitions in process models a.

4 RESULTS

This section presents the results of our comparative process mining study. We then introduce the process models produced, and we present the results of their quality evaluation. Lastly, the expert-based process model and the data-based process model are compared.

4.1 Comparative Study of Algorithms

All the selected algorithms have been applied to the inguinal hernia dataset presented in 3.3.1.

4.1.1 Algorithm Parameters

Most of the process discovery algorithms we selected do not require specific parameter settings. However, the Inductive Miner and the ETM Miner do require specific parameters.

For the Inductive Miner algorithm, a noise threshold must be selected. By default, it should be set to 0.2 (Leemans et al., 2014). However, depending on the quality of the data, it is noted by the authors that it might be useful to increase this threshold. This is why we have decided to test three noise threshold values when testing on real data: 0.2, 0.25, and 0.3. Our goal with these settings is to determine whether these variations can improve the metrics' values and lead to the best possible model, as medical data are known to be noisy (Munoz-Gama et al., 2022).

The ETM miner algorithm requires many parameters. Since this is a model that takes a long time to run, compared to the other approaches, we have chosen to keep the default parameters to not increase the computing time (Buijs, 2014).

4.1.2 Computing and Result Transformation

As the dataset used is composed of personal medical data, the computing of all selected process discovery algorithms has been done inside the digital infrastructure of the University Hospital.

A computing pipeline has been created with Python to select data from the dataset, then compute the different process models in one job with the help of PM4PY Python library to interface the ProM tool. However, this was not possible for the ETM algorithm and Split Miner; those computations have been made manually. Computing times for all the algorithms are presented in Table 5.

Some of the selected process mining algorithms produce a process model using the Petri Net notation; others use the BPMN notation. All the produced process models have been converted to BPMN using the conversion tools of ProM(Verbeek et al., 2010) to ensure uniform notation.

4.1.3 Evaluation of the Obtained Process Models

Two of the selected process discovery algorithms, the Alpha+ and ILP miner, did not produce results due to too long computation times. 6 models were produced, with 3 models for Inductive Miner, one for each noise threshold. These models have been evaluated using the different metrics described in 3.3.3, and the results are shown in Table 6.

Table 5: Computing times (in seconds) on the dataset for the selected process mining algorithms.

Algorithm	Computing Time
Inductive Miner ($Th = 0.20$)	2.723
Inductive Miner ($Th = 0.25$)	1.582
Inductive Miner ($Th = 0.30$)	1.419
Heuristic Miner	2.237
Split Miner	2.178
ETM Miner	6060

Table 6: Evaluation result for the models obtained with Inductive Miner, Heuristics Miner, Split Miner and ETM Miner.

Metric	Inductive Miner			Heur.	ETM	Split
	0.2	*0.25*	*0.3*			
Fitness	0.983	0.883	0.841	0.785	0.353	0.764
Precision	0.387	0.816	0.867	0.797	0.995	0.997
Generalization	0.798	0.759	0.745	0.344	0.400	0.549
Simplicity	0.562	0.595	0.617	0.407	0.530	0.58

As we can see, of all produced process models, fitness and precision values are lower than what may be expected. The performance of the ETM Miner is quite disappointing given the computing time involved. If the Heuristics Miner and Split Miner produce good models from the point of view of fitness and precision, those models lack simplicity and generalization. It may be the result of noise in the dataset.

As predicted, the multiples value of the noise threshold for the inductive miner shows an association with the relaxation of this value.

Based on these results, we have selected the process model produced by the Inductive Miner algorithm ($th = 0.3$), to be used as our data-based process model as it fits with the best equilibrium for the different evaluated metrics.

4.2 Comparison of Models

The expert-based process model obtained after multiple iterations with the expert and caregivers for the cure of an inguinal hernia under laparoscopy has been validated by a group of external experts. This process model has been then compared with the selected data-based process model.

The two process models are too big to be integrated in this article. The main characteristics of those models are presented in Table 7.

The two metrics of distinguishability (Kuo and

Table 7: Characteristics of the two compared process models: data-based process model (*dd*) and expertise-drive process model (*ed*).

	Expertise-based process model	**Data-based process model**
Activities	35	95
Transitions	45	138
Arcs	74	210

Chen, 2012) presented in 3.4 have been computed between the data-based process model (*db*) and expertise-based process model (*eb*) and are presented in table 8.

Table 8: Metrics of distinguishability for the two compared models : data-based process model (*db*) and expertise-based process model (*eb*).

Metric	**Value** (%)
$Support(db, eb)$	8.56 %
$Confidence(db\|eb)$	10.24 %
$Confidence(eb\|db)$	83.58 %

A comparison of the characteristics of the two models involved shows their differences in terms of the number of activities, transitions, and arcs, which inevitably leads to a rather weak *support* metric.

If it is consequently expected that the confidence between the data-based model and the expert-based model is low, it is interesting to note that the opposite, the confidence between the expert-based and data-based models, is very high.

We can easily deduce that the expert-based model is included in the data-based model, but that they still differ in some parts. It may be due to the impact of noise in the data (whose impact we see in the evaluation metrics) and by the intrinsic character of the expertise-based model in terms of level of abstraction.

5 DISCUSSION

These elements lead us to several analyses and critiques concerning the results obtained, the biases they may highlight as well as the generalizable and repeatable nature of this work.

5.1 Some Criticisms of the Results

As often described in the literature, medical data are often very noisy (Munoz-Gama et al., 2022), and this has an impact on the values of the different evaluation metrics, where one could expect higher fitness and precision values. Therefore, the process models obtained with state-of-the-art algorithms are subperforming with our dataset.

In addition, the technical constraints related to carrying out the experiments in the digital environment of the hospital added a constraint in the implementation of the experiment and led us to use a Python library as a roundabout way to carry out the study. This constrained experiment did not allow us to use the different settings to their full potential. Work on refining the parameters should be considered.

5.2 Experts' Reflexive Analysis

The expert-based and data-based models are conventionally opposed; on the one hand, one comes from a reflexive analysis process carried out by experts during their gain in experience; on the other hand, the data-based models are based exclusively on data, and the current approaches to process discovery do not allow this reflexive analysis.

Process mining approaches using expert knowledge for the learning of model may be a solution to explore.

5.3 Repeatability and Generalization

If this work focuses on a specific intervention, the treatment of an inguinal hernia, in a specific context, general anesthesia, it can be extended beyond this medical context. This comparative approach of expert-based and data-based models can be applied to more situations. If an expert is capable of creating a model, and if a process mining algorithm can learn it from traces, this approach can be employed.

Unfortunately, our specific result cannot be reproduced easily as the real data are unavailable to the public for privacy reasons. However, this approach applied to situations where it can be assumed that traces and the experts' point of view differ in their levels of abstraction, should give similar results.

6 CONCLUSIONS

To conclude, we can consider different elements. The experimental results that we have produced indicate that an expert-based model can be included in a data-based model. The differentiation between the two models appears to be on a level of abstraction.

Thus, the use of these two types of models can be interesting for health in that they provide two different points of view on the same situation. The expert-based model brings a high-level abstraction point of view and therefore a rationalization of the process, whereas the data-based model brings a point of view oriented by the data and therefore by the execution of the process, which inherently implies all the variations and anomalies that can occur in the real world.

The models have different characteristics that oppose realism and rationalization. A relevant use of these two types of models could be to evolve the expert point of view towards more realism and to evolve the learning models with inclusions of expertise.

6.1 Future Work and Improvement

Those results are quite preliminary and need more work to consolidate and refine our conclusion.

The critiques of our result must be addressed in future work to reduce noise in the event log and improve the quality of the models produced.

As stated by (Munoz-Gama et al., 2022) process mining for healthcare is confronted with multiple challenges, and using classical approaches of process discovery on this kind of data leads to still-interpretable but sub-performing results. Therefore, a future intended work is to propose new approaches and algorithms for process discovery in medical data and expertise-based process model comparison with data-based process models and inclusion of expertise to enhance the process discovery.

Process models learned from traces of different periods of times can show changes in healing practices and therefore contribute to evidence-based medicine. In the same way, process models combined with patient clustering can offer a precious decision aid tool for personalized medicine to choose the best process for a patient.

Better process model learning algorithm for health data is the common goal of solving those challenges.

REFERENCES

Augusto, A. et al. (2017). Split miner: Discovering accurate and simple business process models from event logs. In *2017 IEEE international conference on data mining (ICDM)*, pages 1–10. IEEE.

Berti, A. and van der Aalst, W. M. (2019). Reviving token-based replay: Increasing speed while improving diagnostics. *ATAED@ Petri Nets/ACSD*, 2371:87–103.

Berti, A., van Zelst, S., and Schuster, D. (2023). Pm4py: A process mining library for python. *Software Impacts*, 17:100556.

Blum, F. R. (2015). Metrics in process discovery. *Tech. Rep. Technical Report TR/DCC-2015–6, Computer Science Dept., University of Chile*.

Buijs, J. C. (2014). *Flexible evolutionary algorithms for mining structured process models.* PhD thesis, Technische Universiteit Eindhoven. Research TU/e / Graduation TU/e), Mathematics and Computer Science.

Buijs, J. C., van Dongen, B. F., and van der Aalst, W. M. (2014). Quality dimensions in process discovery. *International Journal of Cooperative Information Systems*, 23(01):1440001.

De Medeiros, A. A. et al. (2004). Process mining: extending the alpha-algorithm to mine short loops. *Working papers*, 113.

Gao, M., Liu, W., et al. (2022). Global trends in anesthetic research over the past decade: A bibliometric analysis. *Annals of Translational Medicine*, 10(10).

Guzzo, A., Rullo, A., and Vocaturo, E. (2022). Process mining applications in the healthcare domain: A comprehensive review. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 12(2):e1442.

Journal Officiel, R. F. (1994). Décret n°94-1050 du 5 décembre 1994.

Katsoulakis, E., Wang, Q., et al. (2024). Digital twins for health: a scoping review. *NPJ Digital Medicine*, 7(1):77.

Kaymak, U., Mans, R., Van de Steeg, T., and Dierks, M. (2012). On process mining in health care. In *2012 IEEE international conference on Systems, Man, and Cybernetics (SMC)*, pages 1859–1864. IEEE.

Kononowicz, A. A., Woodham, L. A., et al. (2019). Virtual patient simulations in health professions education. *Journal of medical Internet research*, 21(7):e14676.

Kuo, M.-H. and Chen, Y.-S. (2012). A method to identify the difference between two process models. *J. Comput.*, 7(4):998–1005.

Leemans, S. J. et al. (2014). Discovering block-structured process models from event logs containing infrequent behaviour. In *Business Process Management Workshops: BPM 2013*, pages 66–78. Springer.

Munoz-Gama, J. and Carmona, J. (2010). A fresh look at precision in process conformance. In *International Conference on Business Process Management*, pages 211–226. Springer.

Munoz-Gama, J., Martin, N., Fernandez-Llatas, C., et al. (2022). Process mining for healthcare: Characteristics and challenges. *Journal of Biomedical Informatics*, 127:103994.

O'Connor, A. M. et al. (1999). Decision aids for patients facing health treatment or screening decisions: systematic review. *Bmj*, 319(7212):731–734.

Rojas, E., Munoz-Gama, J., et al. (2016). Process mining in healthcare: A literature review. *Journal of biomedical informatics*, 61:224–236.

Van Der Aalst, W. (2016). *Data science in action*. Springer.

van Der Aalst, W. M., Ter Hofstede, A. H., Kiepuszewski, B., and Barros, A. P. (2003). Workflow patterns. *Distributed and parallel databases*, 14:5–51.

Van der Werf, J. M. E. et al. (2008). Process discovery using integer linear programming. In *International conference on applications and theory of petri nets*, pages 368–387. Springer.

Verbeek, H., Buijs, J., Van Dongen, B., and van der Aalst, W. M. (2010). Prom 6: The process mining toolkit. *Proc. of BPM Demonstration Track*, 615:34–39.

Weijters, A. J., van Der Aalst, W. M., and De Medeiros, A. A. (2006). Process mining with the heuristicsminer algorithm. *Working papers*, 166.