Mathias Weske, Judith Michael (Hrsg.): Modellierung 2024,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2024 119

Process Mining for Unstructured Data: Challenges and Research Directions

Agnes Koschmider[1], Milda Aleknonytė-Resch[2], Frederik Fonger[2], Christian Imenkamp[1], Arvid Lepsien[2], Kaan Apaydin[2], Maximilian Harms[1], Dominik Janssen[1], Dominic Langhammer[1], Tobias Ziolkowski[3], Yorck Zisgen[1]

Abstract: The application of process mining for unstructured data might significantly elevate novel insights into disciplines where unstructured data is a common data format. To efficiently analyze unstructured data by process mining and to convey confidence into the analysis result, requires bridging multiple challenges. The purpose of this paper is to discuss these challenges, present initial solutions and describe future research directions. We hope that this article lays the foundations for future collaboration on this topic.

Keywords: Process Mining; Unstructured Data; Challenges; Directions

1 Introduction

The volume of data is continuously increasing and the ability and demand to efficiently analyze the data has become even more crucial. Machine learning and data mining are suitable techniques and tools to efficiently process and analyze the data. Complementary to both techniques is *process mining* [Aa16]. Process mining is a promising approach to find additional patterns (e.g., in terms of causal effects or bottlenecks) in data and in that way to give new insights into the data that could not be directly found with techniques like machine learning or data mining. The insights from processes are given by means of events that have been tracked by information systems. Then, this event data that is structured within a log (i.e., an event log), is used as input to any process mining algorithm. Process mining allows both an analysis based solely on event logs as well as a comparison between (manually generated or as-is) process models and an event log reflecting the to-be processes. By means of process mining, patterns can be uncovered in data with the objective to reveal comprehensive insights into the end-to-end processes aiming to answer questions like

[1] University of Bayreuth, Group Business Informatics and Process Analytics, Wittelsbacherring 10, 95444 Bayreuth, Germany
{firstname.lastname}@uni-bayreuth.de
[2] Kiel University, Group Process Analytics, Hermann-Rodewald-Str. 3, 24118 Kiel, Germany
{mar,ffo,ale,kap}@informatik.uni-kiel.de
[3] GEOMAR Helmholtz Centre for Ocean Research, Wischhofstr. 1-3, 24148 Kiel, Germany
tziolkowski@geomar.de

"When did what happen and in what order?", "When will something happen?", "Are there deviations from how it should have happened?" or "Will there be any unforeseen events?". A plethora of process discovery algorithms exist, mostly focusing on structured data (e.g., [GA07; LFA13; WAM06]) and relying on three requirements for the event log: a case ID, an activity name and a timestamp. However, many application scenarios that are based on unstructured data would benefit from process mining.

In this paper, data is referred to as unstructured when it is not organized in a scheme that enables the retrieval of information as required by the desired application [BH06], i.e., it does not fulfill the requirements of an event log for process mining. Note that this does not mean that unstructured data lacks any structure at all [FS07]. Rather, data can be unstructured for process mining while being structured for other applications – e.g., unprocessed video data lacks case IDs and activity names, but is highly suitable and structured for the task of determining the color of a pixel at a specific timestamp.
Unstructured data is a common data format, for instance in disciplines like engineering or life and natural sciences. These disciplines have a high demand to identify anomalies and causalities in the data and thus, to receive answers to the questions that process mining can respond to. However, unstructured data such as images, audio, video, documents, social media, or sensor data has not been curated in a computer-accessible form fit for process mining and thus to be directly used for these disciplines. This raises the need for processing methods that transform these unstructured data into a format that is structured for process mining. For instance, video data can be transformed into data structured for process mining through the choice of an appropriate data collection, activity extraction, and event abstraction techniques [LKK23]. The overarching challenge for process mining on unstructured data is that such data collections are on a lower level of abstraction as commonly used data for process mining, which is close to the business level. Classical process mining assumes that event data is totally ordered, discrete, correct, and accurate, in an isomorphic relation with individual activity executions, and, finally, at a symbolic level. Meeting these requirements is challenging for unstructured data. Thus, a direct application of process mining is not possible or would not lead to useful results. Instead, a number of intermediary steps are necessary before applying process mining on unstructured data.

The purpose of this paper is to discuss challenges and research directions related to process mining on unstructured data, similarly as conducted for other fields like modeling [Mi23]. Generally, on one hand, the challenges relate to the quality of the data, and on the other hand to the process of the data analysis (i.e., how is and could the data be processed?). Against this background, the paper is structured as follows. The next section summarizes a use case to which we will refer throughout the paper. Section 3 describes the analysis pipeline for process mining on unstructured data, which we commonly apply in our research. The challenges are presented in Section 4 and are followed by research directions, which are summarized in Section 5. The paper ends with a conclusion.

2 Illustration of Process Mining on Unstructured Data on a Use Case

This section introduces a smart factory use case, as seen in Fig, 1. We will refer to this use case throughout the paper to illustrate the challenge and future research directions for process mining on unstructured data. The use case refers to a production process with four assembly lines. One assembly line puts raw components together into a product. Subsequently, the next line divides them into a drilling and welding step, accordingly. Then, the product is colored and packaged. Table 1 shows an excerpt of an exemplary event log with products, timestamps, and activities from such a production. Several sensors and video cameras are installed in the smart factory for automatic control. For instance, temperature, humidity, and carbon dioxide are tracked, while cameras monitor the tasks of the assembly line. Finally, the quality of the product is inspected.

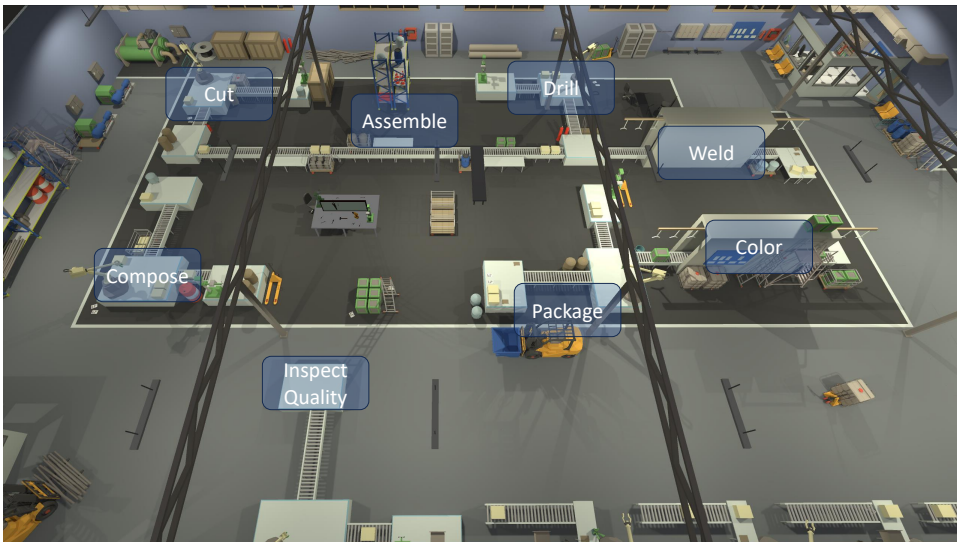

Fig. 1: Smart factory use case with four assembly lines where each line is defined by tasks.

Using unstructured data for process mining for this use case requires efficient pre-processing techniques, which we will summarize in the next section. In the following, we distinguish between unstructured data, which has been captured regularly (like time series data) or irregularly (e.g., event-based). Time series data might be captured either as a continuous, irregular, or periodic stream within specific time frames. Sensors of a smart factory generate time series data through temperature or humidity sensors. The data has a low level of abstraction as raw sensor readings are stored in the data collection, as exemplary shown in Table 2. This is different from business data which is typically used for process mining, which has activities at a high level of abstraction. Event-based data is captured when triggered by events, e.g., an alarm that is activated in response to an event (e.g., exceeding a temperature value).

Product ID	Timestamp	Activity	...
1234	2021-01-01, 10:00	Compose	
1567	2021-01-01, 10:00	Color	
1567	2021-01-01, 10:05	Inspect Quality	
1567	2021-01-01, 10:10	Package	
1234	2021-01-01, 10:10	Color	
...	...	...	

Tab. 1: Exemplary event log for the smart factor production use case (Fig. 1).

Timestamp	Sensor Type & ID	Sensor Value
...	...	...
2021-01-01 12:59:57	`MotionSensor22`	ON
2021-01-01 13:06:21	`MotionSensor22`	OFF
2021-01-01 13:06:22	`MotionSensor14`	ON
2021-01-01 13:06:23	`TemperatureSensor04`	22
2021-01-01 13:06:24	`MotionSensor23`	OFF
2021-01-01 13:06:23	`TemperatureSensor07`	65
...	...	...

Tab. 2: Example of stationary sensor event data

Process mining can give valuable insights into the smart factory use case in terms of discovering reasons for, e.g. late packaging and deliveries, identifying bottlenecks, or improving predictions. In addition to the smart factory use case, process mining on unstructured data might provide also benefits in domains such as healthcare, logistics, finance, and education.

3 The Process Analytics Pipeline

To efficiently analyze data, requires a systematic approach. Fig. 2 shows such an approach in terms of a process analytics pipeline. The pipeline consists of five subsequent steps, which generally apply to all data formats, structured and unstructured. However, applying this pipeline requires a prior evaluation if new insights can be gained for the specific use case, particularly when unstructured data is involved. Additionally, each step in the pipeline needs to be adjusted to the case and its specific characteristics (e.g., when the data is distributed and distributed analysis is an efficient analysis solution). The subsequent paragraphs summarize

each step of the pipeline, focusing on the handling of unstructured data in the smart factory use case.

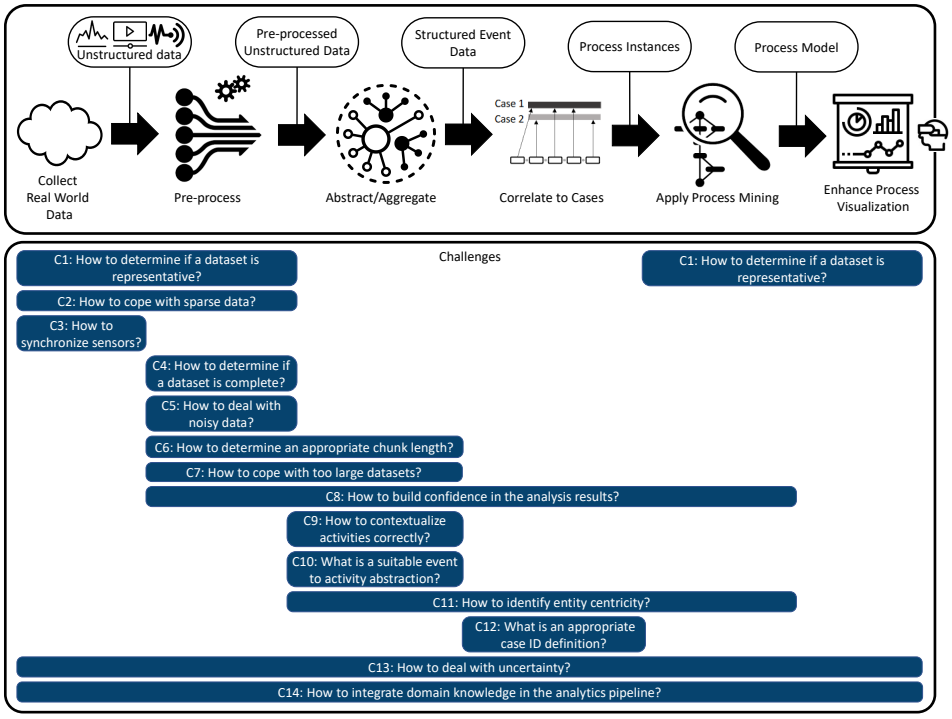

Fig. 2: The way from unstructured data to process visualization: the steps to get there.

Pre-processing Data pre-processing is a time-consuming and error-prone task due to extensive tasks involved like data integration, data enhancement and enrichment, data transformation, data reduction, data discretization and data cleaning [Te23]. Data pre-processing is necessary before the data from the raw format can be used (i.e., time series data must be transformed into an appropriate data format), noise and outliers (e.g., erroneous or missing values) must be removed, and representative data has to be extracted. In the smart factory, video data could be pre-processed in order to delete images of low resolution, adjust frame rates, and label data with bounding boxes and activity names [LKK23]. The processed data must be divided into subsequences representing time periods (like seasons, or weeks). Clustering is one appropriate technique to find similarities between subsequences. As a result, one subsequence can be assigned to one cluster, while one cluster is mapped onto a process activity. We call the step from a cluster to an activity an event/activity abstraction, which is summarized next. The data shown in Tab. 2 might be used as input for the pre-processing step.

Abstraction Next, raw data has to be enriched with semantics. Assume that the raw data shown in Tab. 2 has been used as input for the abstraction phase, then aggregation techniques are needed to leverage the semantics of raw data to a higher level of abstraction. For this, several techniques exist to abstract high-level events (e.g., Table 1) from pre-processed raw data (e.g., Table 2). A comprehensive survey on event abstraction for structured data was conducted by van Zelst et al. [Ze21]. High-level events for process mining can be abstracted through thresholds between data entries (e.g., between temperature measurements for machines in the smart factory to define activities such as "temperature increases", therefore classification of time series data can be use for the abstraction of activities), by applying knowledge gathered from domain experts [JKM; LKK23] or by using machine learning to uncover more complex patterns [BN06; MD15].

Case Correlation In this step, the abstracted (high-level) events have to be correlated to process instances by assigning case IDs. Several properties have been used to relate one record to an instance of a process and often techniques must be designed allowing to involve domain knowledge or make assumptions about the start and end of process instances [Ja21; JKM; Le22; LKF20]. In the smart factory, activities could be assigned to cases by setting the case ID equal to the product ID of the product (see Fig. 1) that is being produced such that one instance of the process begins with the collection of raw material and ends with the packaging of the final product. Different process discovery algorithms exist to extract process models from event data, each with advantages and disadvantages [LKF20], some of the most popular being the Heuristic Miner [WAM06], Fuzzy Miner [GA07] and Inductive Miner [LFA13]. To analyze the event data of the smart factory, discovery algorithms could be applied to mine models of the most frequent production steps or get a holistic overview of all process variations.

Visualization The visualization of event data, as well as the visualization of process graphs can be approached with different diagrams and models for analysis. The visualization of the process graphs with e.g., Petri nets allow the analysis of the control flow, i.e. the order of activities, choices, concurrency, and loops within the process. Further, process graphs can be augmented in order to analyze how often activities follow one another or to analyze the duration of activities in terms of time differences between activity executions. Moreover, process mining tools are able to simulate the discovered process models, which allow additional insights [Aa22]. Futhermore, initial studies show that the adaption of 3D visualizations, e.g., virtual or augmented reality, can have benefits due to the visualization space and depth [WK22]. Besides process graphs, additional visualization techniques exist, such as dotted charts [SA07], sequence diagrams [LVV18], heatmaps [LKK23] and decision trees [DVD14]. In the smart factory use case dotted charts might be used to visualize when production steps are executed.

4 Challenges

This section presents challenges that need to be addressed to advance the area of process mining on unstructured data. These challenges stem from various use cases we have encountered e.g. [JKM; LKK23; Me23; Zi22]. The challenges lay the foundation to formulate research directions that we will describe in Section 5. To align the challenges with the process analytics pipeline, we cross-reference the challenges with steps of the pipeline (Fig. 2).

Challenge 1. How to determine if a dataset is representative? □

The representative selection of a dataset impacts the goal of the analysis. For example, if the goal of the analysis is anomaly detection, then the anomalous behavior results from the definition of anomalous vs. behavior that is within the range. The analysis goals should be fixed before gathering data and thus, the data collection is planned in such a way as to collect representative data. In practice process mining is rarely the primary goal for gathering data, but rather, already existing data needs to be pre-processed to fit the requirements for process mining. Thus, it is imperative to assess the suitability of the dataset for the intended analysis. An exploratory data analysis might help to get initial insights whether the gathered data aligns with the analysis goals and is not a "garbage-in-garbage-out" approach [WS19]. Assume that the analysis goal is to find anomalies in the first assembly line, then the observation period would have to be long enough to collect data affecting all steps of the first assembly line. If the data set is considered to be non-representative, it will not necessarily impact the data abstraction and case correlation pipeline steps, but will affect the conclusions made from the analysis (i.e., lead to misleading conclusions).

Challenge 2. How to cope with sparse data? □

Process data has to be captured for processes that are frequently repeated so that variations are detected and can be analyzed [Gr20]. For example, in a smart factory, sensors that are attached to machines could be installed for a limited time period, which could lead to too sparse data for process mining. To overcome this challenge, related data could be synthetically generated [ZJK22]. Depending on the type of data, this could be done by using algorithms that generate additional data, or by applying noise to a given data set [GBC16] during the pre-processing step (Fig. 2). Alternatively, more real-world data may need to be collected.

Challenge 3. How to synchronize sensors? □

Time synchronization is an issue in case multiple sensors are in use and interact with each other [ESA16]. If sensors are not synchronized in the data collection step (Figure 2), it might result in temporal discrepancies in the recorded data as well as incorrect representations of the behavior of events and finally, lead to erroneous interpretations of the

process models derived from the event log. For example, multiple motion detection sensors monitoring different areas of the smart factory must be calibrated for the same timestamp in order to correctly identify moving entities from one area to another.

Challenge 4. How to determine if a data set is complete? □

A data set may not be complete due to missing data. Reasons for missing values are unreliable sensors that do not always work as intended, sensors that were not used properly, or if data recording was not possible. For instance, assume that wearable sensors are used, but have not been turned off while wearing and thus continuously generate data. Also, it is challenging to identify rare events in the data since it is not clear whether the event is missing or rare. To counteract this, counting event frequencies or applying rare event detection techniques might be a solution [HSR16]. Referring to the smart factory example, let's assume that location detection is an analysis purpose. Then external data about power failures could be included beside the data from the movement detection sensors to determine data set completeness. Incomplete data can lead to erroneous data logs and thus impact not only the abstraction/aggregation pipeline step, but all other following steps.

Challenge 5. How to deal with noisy data? □

Logged data can have different types of imperfections [Su17], especially incorrect data [BMA13]. Therefore, the quality of the data should be improved during the pre-processing step (see Fig. 2) using data cleaning techniques [JKM]. Outliers can lead to the creation of models that contain a significant number of infrequent execution paths or that do not accurately reflect the behavior [CRH17]. For example, imagine tracking the location of an object in the smart factory. Even though the object is still at an intermediate production step, i.e., further steps within the production line are mandatory, the captured location records could indicate a transport to the warehouse. This occurrence might be tracked back to software or sensor failure. Filtering such cases becomes necessary. Otherwise, the final process model might contain paths that are not possible in the real world. Furthermore, there are also cases when filtering techniques are not applicable. For example, when a video camera in the smart factory captures data that contains scenes where the images are blurred due to unexpected light, movement, or a malfunctioning focus. In such a case it might be more efficient to proceed with the recording of another data set instead of handling the erroneous data. Eventually, the step of the process analytics pipeline "pre-processing" is often a labor-intensive task but has a significant impact on the process mining result.

Challenge 6. How to determine an appropriate chunk length? □

To determine the chunk length of unstructured data is crucial for several reasons. Firstly, the chunk length determines the granularity at which the data is analyzed. Depending on the goal of the analysis, a smaller chunk length allows detailed insights into the analysis goal. Contrary, a larger chunk length allows a more general view. Also, the chunk length

has an impact on clustering, classification, and pattern recognition [Ja21]. A sliding window technique in the pre-processing step (Fig. 2) can be used to divide the data into overlapping chunks of varying sizes, allowing for analysis of different resolutions and improved identification of patterns and trends. Statistical techniques such as power analysis and sample size calculation can also be employed to determine an appropriate chunk length. Alternatively, machine learning algorithms, including clustering and dimensionality reduction techniques such as Principal Component Analysis, can be applied to extract meaningful information from high-dimensional time series data and improve interpretability for further analysis. In a smart factory, sensors that are installed in machines and in the facility produce big data. This amount of data needs to be segmented into a chunk size appropriate for the research question. Finally, processing large chunks of unstructured data can be computationally intensive. By determining an appropriate chunk length, data processing times can be reduced. Thus, the chunk length selection is a trade-off between computational complexity and information accuracy related to the analysis goal.

Challenge 7. How to cope with too large datasets? □

If the data volume is too high, computational complexity is an issue affecting all steps of the process analytics pipeline (see Challenges 6 and 10). Re-scaling or sampling allows reducing the quantity of the data [LKK23; Sa17], which in our case would be part of the pre-processing step. However, this could impact the amount and accuracy of activities abstracted from the events. Thus, when reducing the amount of data, not only the processing time but also the meaningfulness of abstraction needs to be considered [LKK23]. As an example, video data gathered from cameras in the smart factory may require rescaling as a specific measure to manage data volume. The resolution of the videos might need to be scaled down to reduce file size until processing time becomes adequate while activities, e.g., distinct manual assembly steps, remain detectable [ERH02; HHL23; Ma19]. Thus, given such scenarios, the applicability of process mining techniques becomes limited.

Challenge 8. How to build confidence in the analysis results? □

While process mining on unstructured data can produce valuable insights, its potential to realize this value is inherently limited by the confidence that decision-makers place in it [MF21]. Providing confidence in the process mining results, which rely on machine learning techniques (e.g., for object and activity recognition from video recordings in the smart factory) is challenging due to the black-box nature of these models [MF21]. These concerns are further amplified when multiple pipeline steps are required to prepare the data, raising the need for methods to communicate the results in a way that builds trust in the analysis and confidence in the decisions based on the results [KOH22].

Challenge 9. How to contextualize activities correctly? □

The methods available for pre-processing unstructured data for process mining usually do not integrate the context of the analyzed process. This can be problematic, as two activities that are distinct in a process may be difficult to distinguish at the raw data level [REF19]. For instance, in the smart factory example, wearable sensors would produce very similar data for lifting a workpiece from one workstation and putting it down at the next station. Because the pre-processing steps miss contextual information, they are unable to efficiently distinguish between these activities during the abstraction/aggregation step. Thus, in the pipeline steps after pre-processing, the data needs to be explicitly contextualized into the realm of the analyzed process.

Challenge 10. What is a suitable event to activity abstraction or aggregation? □

The challenge of abstraction or aggregation particularly when processing unstructured data is that semantics must be put into input data that is generally not understandable at all (i.e., the input data is at a lower level of abstraction than business data). The abstraction of activities to events has been extensively studied [Ze21]. A too fine-grained abstraction leads to the overfitting of the discovered process model, while a too coarse-grained abstraction results in underfitting. To illustrate this, consider the activity of cutting wood. Then the question arises if the start of the activity is initiated when the wood enters the area of the machine, when the machine or the lid starts running or is closed, or when the cutting process starts. These questions need to be answered before starting the analysis and the same requirement applies to all data anylsis and data sets. The abstraction of activities from events (in terms of granularity) depends on the analysis purpose [Ta18; Ze21].

Challenge 11. How to identify entity centricity? □

Most stationary sensors operating in the event-based capture mode and video cameras share the challenge of identifying the observed entities, especially if a lack of identification tags is present. Regarding video data, in the case of observing humans, entities could be re-identified using facial recognition [KKR22] during pre-processing, abstraction, and aggregation as well as case correlation. However, there are monetary and sensor management trade-offs when implementing these strategies. Similarly with video data, if the entities exit the field of view, it is difficult to determine which entity has re-entered the observation field first. For a different example, a smoke detector placed in the middle of the smart factory cannot identify the machine or entity triggering it. By increasing the number of sensors or video cameras present, it may be easier to identify the entities, e.g., in the case of a smoke detector, a number of smoke detectors could be placed above each factory machine, and thus the smoke detector right above the machine emitting smoke will be triggered first.

Challenge 12. What is an appropriate case ID definition? □

A further challenge when preparing the data for process mining is the definition of the case ID, which is necessary for process mining algorithms to extract process models from event logs. The case notion might be ambiguous and thus selecting and assigning an appropriate case ID during correlation is challenging since no distinction of cases exists, e.g., in time series data from natural and life science [Me23]. Also, applications exist where the systems are not able to log the case IDs [BDM23]. Within the smart factory, sensors attached to some production steps may generate a continuous data stream (e.g. torque, temperature, humidity) which does not include identifiers to correlate their output with steps in the production cycle of this specific production step. After defining activities and events through an abstraction process (see Challenge10), case IDs have to be defined artificially.

Challenge 13. How to deal with uncertainty? □

Generally, the collection and processing of unstructured data are subject to errors, which introduces uncertainty to the analysis. Because this uncertainty is inherent in every analysis using process mining on unstructured data, uncertainty-aware process mining techniques, i.e., techniques that explicitly handle the uncertainty attached to an event log, and quantify or visualize it, are required [Pe22; PUA21]. For instance, inadequate data collections (e.g., due to too low video frame rates, incorrect sensor placements), and the probabilistic mappings applied during the abstract/aggregate and correlate to case steps, lead to event logs not being fully significant [KMH19].

Challenge 14. How to integrate domain knowledge in the analytics pipeline? □

While many steps of the analytics pipeline can be fully automated (e.g., collecting real-world data, pre-processing), the main challenge is understanding the practical application contexts where domain knowledge is crucial. Domain knowledge can be provided in many different forms, e.g., textual descriptions of a process or relationships between activities, formal constraints, or normative process models. For example, domain knowledge can be used to facilitate event abstraction [Ba18], in case correlation [BAE16] or to repair missing events in traces [BCG23]. The focus here is recognizing and defining the areas where domain knowledge integration can enhance analysis quality.

5 Future Directions

This section summarizes research directions that should be tackled in the future in order to significantly advance the field of process mining on unstructured data.

Direction 1. Integration of domain knowledge. □

Related challenges: 1, 9, 11, 12, 14

Domain knowledge is pivotal for understanding the context of processes and events. Integrating this knowledge into the analytics pipeline is crucial (see Fig. 1). A core aspect of this direction is to set the focus on the development of *generic* methods and models for integrating domain knowledge into the analytics pipeline. Techniques such as natural language processing and sentiment analysis facilitate the extraction of vital information from textual data. Analyzing language in documents and other unstructured sources give additional insights and uncover patterns not evident in structured data alone.

A key aspect of this direction is designing hybrid models that combine data science methods with human input, which can integrate this domain knowledge into specific use cases. Here, humans give domain-specific insights to enhance the analysis's accuracy. For instance, factory workers can provide additional information about the production process and the relationships between different activities that can be used to improve the accuracy of the analysis, while domain experts can shed light on particular challenges related to smart factories.

Future studies should emphasize hybrid models for seamless domain knowledge integration. This entails designing techniques to capture and weave into domain-specific insights, implementing algorithms to assimilate this knowledge efficiently, and establishing validation protocols. Moreover, integrating feedback loops will be the key to refining these models sustainably.

Direction 2. Exploration of data fusion techniques for comprehensive analysis. □

Related challenges: 2, 3, 4, 7, 10

Data fusion combines structured and unstructured data sources and gives additional insights into the analysis. This involves integrating data from distributed sources to create a more complete and accurate picture of the underlying processes. For instance, in the smart factory use case, data from sensors and video cameras are collected to monitor the tasks of the assembly line. By fusing this data, and other structured data sources such as production logs and quality control reports, it is possible to identify patterns that may not be apparent from solely one data source alone. This requires the development of new techniques for data pre-processing, feature extraction, and data integration for heterogeneous data. To enable applications such as predictive monitoring and support during execution, these data

fusion techniques need to be developed with a focus on scalability to enable integration and analysis of the data in real-time.

Direction 3. Use of advanced visualization techniques for unstructured data. □

Related challenges: 5, 6, 8

New visualization techniques are required to efficiently convey the results of process mining on unstructured data to stakeholders. Traditional process mining techniques often rely on visualizations in terms of process models. These visualizations may not always be suitable for unstructured data sources, which require different kinds of visualization views. It is imperative to strike a balance between complexity and clarity: it must be complex enough to convey the insights and at the same time simple enough to be understandable. For instance, in the smart factory use case, the data collected from sensors and video cameras can be visualized using interactive dashboards, and 3D visualizations, among others.

Furthermore, the scalability of advanced visualization techniques will emerge as a paramount requirement. As the volume and variety of data sources increase, it becomes more challenging to visualize the data in real-time. Therefore, future research should focus on developing scalable visualization techniques that can handle large volumes of data from different sources.

Direction 4. Use of machine learning to make process mining on unstructured data more explainable. □

Related challenges: 5, 7, 8, 11

As process mining on unstructured data becomes more prevalent, it is important to ensure that the results are explainable and trustworthy. This is particularly important when using unstructured data in critical fields, such as medicine or law. Otherwise, reliable decisions could be compromised, which could lead to incorrect medical treatments or unfair legal outcomes. Future research should tackle explainability by developing machine learning models that enhance data accuracy (e.g., via feedback mechanisms) while incorporating explainability.

Direction 5. Research and develop frameworks for ethical and legal implications. □

Related challenges: 8, 13

Collecting and processing unstructured data such as camera images or sensor readings raises issues related to data protection and privacy, and possibly bias and discrimination if the data is not collected properly, undermining confidence in the results. Furthermore, when decisions from the analysis are drawn, people in less represented cases could be discriminated against [Ma22].

Also, process mining currently lacks clear guidelines and regulations for using unstructured data. Therefore, future research should focus on developing comprehensive frameworks for ethical, legal, and transparent data governance. This involves examining the ethical and legal issues involved in using unstructured data for process mining and developing guidelines and regulations to ensure that the data is collected, stored, and analyzed in a responsible and accountable manner. To guarantee privacy, techniques need to be developed that can anonymize the data while preserving its utility. This is an area that requires further research [El22].

A collaboration between researchers, practitioners, and policymakers is needed to develop these frameworks. This requires a deep understanding of the ethical and legal issues involved in process mining and the ability to transfer this knowledge into practical guidelines and regulations.

6 Conclusion

This paper presented challenges and research directions related to process mining on unstructured data. The quality of the data and the process of data analysis are two major challenges that need to be addressed. To overcome these challenges, an analysis pipeline consisting of five subsequent steps has been presented. The challenges presented arise from the large number of our practical experiences with process mining on unstructured data.

Process mining on unstructured data is a challenging but promising area of research. By addressing the challenges and exploring new research directions, the potential of unstructured data can be unlocked, and new insights into processes can be gained that traditional techniques might not fully respond.

Acknowledgments

This project has received funding from the State of Schleswig-Holstein under the Datencampus project grant no. 220 21 016, the German Research Foundation (DFG) SPP 2422 and FOR 5495, the Federal Ministry for Digital and Transport under the CAPTN-Förde 5G project grant no. 45FGU139 H, the Federal Ministry for Economic Affairs and Climate Action under the MARISPACE-X project grant no. 68GX21002E and the German Federal Ministry of Education and Research (BMBF) for the ABBA project grant no. 16DHBKI002, 16DHBKI003, 16DHBKI004, 16DHBKI005.

References

[Aa16] van der Aalst, W. M. P.: Process Mining: Data Science in Action. Springer, Berlin, Heidelberg, 2016.

[Aa22] van der Aalst, W. M. P.: Process Mining: A 360 Degree Overview. In (van der Aalst, W. M. P.; Carmona, J., eds.): Process Mining Handbook. Springer International Publishing, pp. 3–34, 2022.

[Ba18] Baier, T.; Di Ciccio, C.; Mendling, J.; Weske, M.: Matching events and activities by integrating behavioral aspects and label analysis. Software & Systems Modeling 17/2, pp. 573–598, 2018.

[BAE16] Bayomie, D.; Awad, A.; Ezat, E.: Correlating Unlabeled Events from Cyclic Business Processes Execution. In: CAiSE 2016. LNCS, Springer, pp. 274–289, 2016.

[BCG23] Bogdanov, E.; Cohen, I.; Gal, A.: SKTR: Trace Recovery from Stochastically Known Logs. In: 2023 5th International Conference on Process Mining (ICPM). Pp. 49–56, 2023.

[BDM23] Bayomie, D.; Di Ciccio, C.; Mendling, J.: Event-case correlation for process mining using probabilistic optimization. Information Systems 114/, p. 102167, 2023.

[BH06] Boulton, D.; Hammersley, M.: Analysis of unstructured data. Data collection and analysis 2/, pp. 243–259, 2006.

[BMA13] Bose, R. J. C.; Mans, R. S.; van der Aalst, W. M.: Wanna improve process mining results? In: 2013 IEEE Symposium on Computational Intelligence and Data Mining (CIDM). Pp. 127–134, 2013.

[BN06] Bishop, C. M.; Nasrabadi, N. M.: Pattern recognition and machine learning. Springer, New York, 2006.

[CRH17] Conforti, R.; Rosa, M. L.; Hofstede, A. H. T.: Filtering Out Infrequent Behavior from Business Process Event Logs. IEEE Transactions on Knowledge and Data Engineering 29/2, pp. 300–314, 2017.

[DVD14] De Leoni, M.; Van Der Aalst, W. M. P.; Dees, M.: A General Framework for Correlating Business Process Characteristics. In: BPM 2014. Vol. 8659, LNCS, Springer, Cham, pp. 250–266, 2014.

[El22] Elkoumy, G. et al.: Privacy and Confidentiality in Process Mining: Threats and Research Challenges. ACM Trans. Manag. Inf. Syst. 13/1, 11:1–11:17, 2022.

[ERH02] Egmont-Petersen, M.; de Ridder, D.; Handels, H.: Image processing with neural networks—a review. Pattern recognition 35/10, pp. 2279–2301, 2002.

[ESA16] van Eck, M. L.; Sidorova, N.; van der Aalst, W. M. P.: Enabling process mining on sensor data from smart products. In: 2016 IEEE Tenth International Conference on Research Challenges in Information Science (RCIS). IEEE, pp. 1–12, 2016.

[FS07] Feldman, R.; Sanger, J.: The text mining handbook: advanced approaches in analyzing unstructured data. Cambridge university press, 2007.

[GA07] Günther, C. W.; van der Aalst, W. M. P.: Fuzzy Mining – Adaptive Process Simplification Based on Multi-perspective Metrics. In: BPM 2007. LNCS, Springer, Berlin, Heidelberg, pp. 328–343, 2007.

[GBC16] Goodfellow, I.; Bengio, Y.; Courville, A.: Deep learning. MIT press, 2016.

[Gr20] Grisold, T.; Mendling, J.; Otto, M.; vom Brocke, J.: Adoption, use and management of process mining in practice. Business Process Management Journal 27/2, pp. 369–387, 2020.

[HHL23] Hojjat, A.; Haberer, J.; Landsiedel, O.: ProgDTD: Progressive Learned Image Compression with Double-Tail-Drop Training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. IEEE, pp. 1130–1139, 2023.

[HSR16] Harrison, D. C.; Seah, W. K. G.; Rayudu, R.: Rare Event Detection and Propagation in Wireless Sensor Networks. ACM Comput. Surv. 48/4, 2016.

[Ja21] Janssen, D.; Mannhardt, F.; Koschmider, A.; van Zelst, S. J.: Process Model Discovery from Sensor Event Data. In: ICPM 2020 Workshops. Vol. 406. LNBIP, Springer, pp. 69–81, 2021.

[JKM] Janssen, D.; Koschmider, A.; Mannhardt, F.: Process Mining on Sensor Location Event Data. In: BPM 2023 Workshops. To appear.

[KKR22] Kratsch, W.; König, F.; Röglinger, M.: Shedding light on blind spots – Developing a reference architecture to leverage video data for process mining. Decision Support Systems 158/, p. 113794, 2022.

[KMH19] Koschmider, A.; Mannhardt, F.; Heuser, T.: On the contextualization of event-activity mappings. In: BPM 2018 Workshops. Vol. 342. LNBIP, Springer, pp. 445–457, 2019.

[KOH22] Koschmider, A.; Oppelt, N.; Hundsdörfer, M.: Confidence-driven communication of process mining on time series. Informatik Spektrum 45/4, pp. 223–228, 2022.

[Le22] Lepsien, A.; Bosselmann, J.; Melfsen, A.; Koschmider, A.: Process Mining on Video Data. In: ZEUS 2022. Vol. 3113, CEUR-WS.org, pp. 56–62, 2022.

[LFA13] Leemans, S. J. J.; Fahland, D.; van der Aalst, W. M. P.: Discovering Block-Structured Process Models from Event Logs - A Constructive Approach. In: Application and Theory of Petri Nets and Concurrency. Springer, Berlin, Heidelberg, pp. 311–329, 2013.

[LKF20] Laue, R.; Koschmider, A.; Fahland, D.: Prozessmanagement und Process-Mining. De Gruyter Oldenbourg, 2020.

[LKK23] Lepsien, A.; Koschmider, A.; Kratsch, W.: Analytics Pipeline for Process Mining on Video Data. In: BPM 2023 Forum. Vol. 490. LNBIP, Springer, pp. 196–213, 2023.

[LVV18] Leemans, M.; Van Der Aalst, W. M. P.; Van Den Brand, M. G. J.: Hierarchical performance analysis for process mining. In: Proceedings of the 2018 International Conference on Software and System Process. ACM, pp. 96–105, 2018.

[Ma19] Ma, S. et al.: Image and video compression with neural networks: A review. IEEE Transactions on Circuits and Systems for Video Technology 30/6, pp. 1683–1698, 2019.

[Ma22] Mannhardt, F.: Responsible Process Mining. In (van der Aalst, W. M. P.; Carmona, J., eds.): Process Mining Handbook. Vol. 448, LNBIP, Springer, Cham, pp. 373–401, 2022.

[MD15] Murty, M. N.; Devi, V. S.: Introduction to pattern recognition and machine learning. World Scientific, 2015.

[Me23] Melfsen, A. et al.: Describing Behavior Sequences of Fattening Pigs Using Process Mining on Video Data and Automated Pig Behavior Recognition. Agriculture 13/8, p. 1639, 2023.

[MF21] Mehdiyev, N.; Fettke, P.: Explainable Artificial Intelligence for Process Mining: A General Overview and Application of a Novel Local Explanation Approach for Predictive Process Monitoring. In: Interpretable Artificial Intelligence: A Perspective of Granular Computing. Studies in Computational Intelligence, Springer, pp. 1–28, 2021.

[Mi23] Michael, J.; Bork, D.; Wimmer, M.; Mayr, H. C.: Quo Vadis modeling? Software and Systems Modeling/, 2023.

[Pe22] Pegoraro, M.: Probabilistic and Non-deterministic Event Data in Process Mining: Embedding Uncertainty in Process Analysis Techniques. In: CAiSE 2022 Doctoral Consortium. Vol. 3139, CEUR-WS.org, pp. 37–46, 2022.

[PUA21] Pegoraro, M.; Uysal, M. S.; van der Aalst, W. M. P.: PROVED: A Tool for Graph Representation and Analysis of Uncertain Event Data. In: Application and Theory of Petri Nets and Concurrency. Vol. 12734, LNCS, Springer, pp. 476–486, 2021.

[REF19] Rebmann, A.; Emrich, A.; Fettke, P.: Enabling the Discovery of Manual Processes Using a Multi-modal Activity Recognition Approach. In: BPM 2019 Workshops. LNBIP, Springer, pp. 130–141, 2019.

[SA07] Song, M.; van der Aalst, W.: Supporting Process Mining by Showing Events at a Glance. WITS 2007 - Proceedings, 17th Annual Workshop on Information Technologies and Systems, 2007.

[Sa17] Sayood, K.: Introduction to data compression. Morgan Kaufmann, 2017.

[Su17] Suriadi, S.; Andrews, R.; ter Hofstede, A. H. M.; Wynn, M. T.: Event log imperfection patterns for process mining: Towards a systematic approach to cleaning event logs. en, Information Systems 64/, pp. 132–150, 2017.

[Ta18] Tax, N.; Sidorova, N.; Haakma, R.; van der Aalst, W. M. P.: Event Abstraction for Process Mining Using Supervised Learning Techniques. In: IntelliSys 2016. Vol. 15, LNNS, Springer, pp. 251–269, 2018.

[Te23] Ter Hofstede, A. H. M. et al.: Process-Data Quality: The True Frontier of Process Mining. Journal of Data and Information Quality 15/3, 2023.

[WAM06] Weijters, A.; Aalst, W.; Medeiros, A.: Process Mining with the Heuristics Miner-algorithm. BETA publicatie : working papers 166/1/, Jan. 2006.

[WK22] Wetzel, M.; Koschmider, A.: Entwicklung einer VR-Umgebung zur Exploration von Process-Mining. HMD Praxis der Wirtschaftsinformatik 59/1, pp. 37–53, 2022.

[WS19] Wynn, M. T.; Sadiq, S.: Responsible Process Mining - A Data Quality Perspective. In: BPM 2019. LNCS, Springer, pp. 10–15, 2019.

[Ze21] van Zelst, S. J.; Mannhardt, F.; de Leoni, M.; Koschmider, A.: Event abstraction in process mining: literature review and taxonomy. Granular Computing 6/3, pp. 719–736, 2021.

[Zi22] Ziolkowski, T.; Koschmider, A.; Schubert, R.; Renz, M.: Process Mining for Time Series Data. In: BPMDS 2022/EMMSAD 2022. Vol. 450. LNBIP, Springer, pp. 347–350, June 2022.

[ZJK22] Zisgen, Y.; Janssen, D.; Koschmider, A.: Generating Synthetic Sensor Event Logs for Process Mining. In: CAiSE Forum 2022. Vol. 452. LNBIP, Springer, pp. 130–137, 2022.