

Process Mining

Applying data science to discover workflows while combining data mining and process analytics

Camino Yuste Laura Maria
Hannover Hochschule

Universidad Rey Juan Carlos
Madrid, Spain
lm.camino.2020@alumnos.urjc.es

Vieilledent Mario
Hannover Hochschule

Institut d'Ingénierie Informatique de Limoges
Rodez, France
mario.vieilledent@gmail.com

Al Nasouh Mohammad
Hannover Hochschule

University of applied sciences and arts
Hannover, Germany
mohamadalnasoh123@gmail.com

Wolf Wolf Andreas
Hannover Hochschule

Universidad Rey Juan Carlos
Madrid, Spain
andreaswolfwolf41202@gmail.com

El Moustamide el Ouazzani Usama
Hannover Hochschule

Institut d'Ingénierie Informatique de Limoges
Limoges, France
usama.el-mousatmide-el-ouazzani@stud.hs-hannover.de

Index Terms—Process Mining, Data Mining, BI, BMN, BPM, PM, Business Process

I. ABSTRACT

This paper will examine Process Mining under all of its angles, history, state of the art methods as well as real case applications under the form of study cases and demonstrations. The methodology used in this paper is a summary of the knowledge behind Process Mining incorporating scientific papers, foundational documents and real use cases. It concludes that Process Mining is a great tool to analyse business processes and that it will be increasingly applied in the business world due to the its growth potential and relative novelty.

II. INTRODUCTION

A. What Is Process Mining

In our increasingly complex world, there is a need to optimize and improve business processes in organizations to keep up with market forces as well as to increase general efficiency and reduce the potential loss (monetary or in other resources) due to those inefficiencies. As an answer to this challenge Process Mining emerged in the early 2000s as a potential answer. The objective of Process Mining is to analyze and improve real-life processes by using data from process execution systems and workflow management systems (an example being **Fig. 1**). Process Mining (PM) analyzes and tracks processes inside businesses while using existing data to automatically display the real process. It's like an X-ray of our business processes. The idea of Process Mining is to *discover*, *monitor* and *improve* real processes by extracting knowledge from event logs (van der Aalst et al., 2012) It works by extracting knowledge from event logs which are generated from business data and then, it makes a model with all that data, so it is easy to discover inefficiencies, revealing their causes, quantifying their impact in the overall business, and giving insights of how to fix them. Basically, when we

have a business in which several operations and events are tracking place simultaneously, it is very hard to know what is exactly happening in every step of the process. So, Process Mining takes all that data recorded in the logs, analyze it, and transform it in a more visual model in which it becomes simple to know what is going on, which process are affecting others and how are they all related. As a result we find that Process Mining is a good intersection between data science and process science (shown in **Fig. 3**).

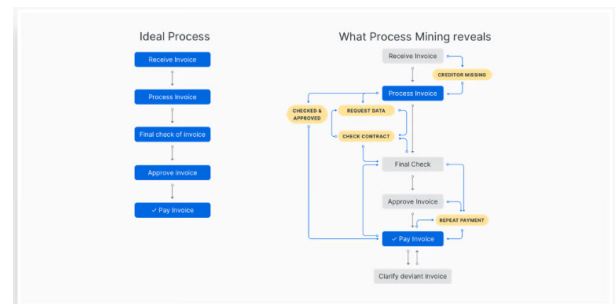


Fig. 1: What we think is the flow of the process VS the real flow of the process

B. Event Log

Set of events, events are data elements composed on the event itself (what is it, and what is the data related to it), what or who is the source of the event, its timestamp and an ID. Each event logs is produced by one activity of a business process and is used as main resource for Process Mining.

“Starting point for Process Mining is an event log”

Process Mining: Overview and Opportunities

C. Process Model

A Process Model is a model built thanks to an event log (like seen in **Fig. 2**). The Process Model helps understanding

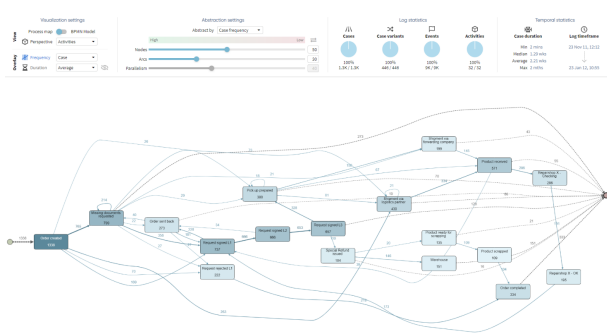


Fig. 2: Example of a data process model

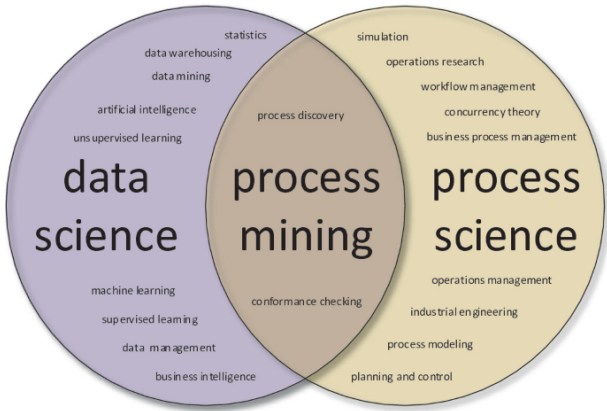


Fig. 3: Process Mining, data science and process science

the real end-to-end business process it describes. The model is the graphical representation of the event log and it aims to be simple so it is understandable. Making, managing, improving process models is in the scope of Process Mining.

There are algorithms used to build process models based on event logs, such as the α -algorithm. Those are considered intelligent since they are not just simple algorithms that produce a well defined result given an input, but can produce different results related on which criteria is considered most important to discover a process model.

Regarding to the GP3 of the Manifesto, process models should support concurrency, choice points and loops (and so do the algorithms for process discovery).

D. Activity

An activity or a task in a company is an operation made by a human or a system, at a specific date and time, and that can be done one other activities are done, and can as well allowing other activities to start.

III. RELATED WORK

For the study of the subject and the writing of this article, we based ourselves on scientific articles describing Process Mining, in particular the Process Mining Manifesto.

These articles present in detail the concepts and technologies related to the field of Process Mining, give guidelines for its future, and provide a basis for further research in this field.

This article relies mainly on the Manifesto to define the types of Process Mining, to define terms, algorithms, etc.

This article provides simple definitions of the basic elements of Process Mining, and attempts to analyse its functioning, impact, benefits, and future prospects through case studies. This article also presents the different algorithms of Process Mining and It contains a manual implementation of an Event Log with the application of an open-source Tool to test the Alpha Miner algorithm.

Reinkemeyer Editor, L. (2020). Process Mining in action : Principles, use cases and outlook.

WIL VAN DER AALST, Eindhoven University of Technology. Process Mining: Overview and Opportunities

IV. HISTORY

The need to have a sort of X-ray of how a business process is being carried out is not new. Since the 19th century we have try to obtain a complete overview of our processes to have a better understanding of the business we own. That desire brings us back to the 21th century. In this era, people still trying to create pen and paper models reflecting the reality of their business processes, but a Computer Scientist of the Eindhoven university of Technology, Wil Van der Aalst, decided that it was about time to make this process more automatic and efficient and that is how Process Mining was born (in the **Fig. 4** we can see the principles). At first, the methods of Van der Aalst did not arise too much interest in the community, but at the beginning of the 21st century, people starting to show more and more interest in this new Process Mining concept. As it was gaining more and more recognition along the years, the IEEE publishes the Process Mining manifesto in 2012, promoting it. Nowadays, Process Mining is considered a require tool for every company that operates in this digital era since businesses have evolve and become much more complex, with a lot more processes going on simultaneously. For the future, researchers are trying to add AI techniques to Process Mining in order to expand it from traditional process logs to predictive models which can study and decipher human processes patterns and integrate them alongside the logs, making Process Mining completer and more accurate.

A. Goals

Process Mining seek to improve efficiency, compliance and to identify process improvement opportunities. Here are it's main goals:

- Discover the actual processes from event logs so we can get a good insight of what is actually happen with each process.
- Be able to compare the actual flow of process against a model to discover deviations and ensuring compliance.
- Improve the process by identifying bottlenecks and inefficiencies.

- Predict the likelihood of certain events
- Do performance measures

GP1	Event Data Should Be Treated as First-Class Citizens
	Events should be <i>trustworthy</i> , that is, it should be safe to assume that the recorded events actually happened and that the attributes of events are correct. Event logs should be <i>complete</i> , that is, given a particular scope, no events may be missing. Any recorded event should have well-defined <i>semantics</i> . Moreover, the event data should be <i>safe</i> in the sense that privacy and security concerns are addressed when recording the event log.
GP2	Log Extraction Should Be Driven by Questions
	Without concrete questions it is very difficult to extract meaningful event data. Consider, for example, the thousands of tables in the database of an ERP system like SAP. Without questions one does not know where to start.
GP3	Concurrency, Choice and Other Basic Control-Flow Constructs Should Be Supported
	Basic workflow <i>patterns</i> supported by all mainstream languages (e.g., BPMN, EPCs, Petri nets, BPML, and UML activity diagrams) are <i>sequence</i> , <i>parallel routing</i> (AND-splits/joins), <i>choice</i> (XOR-splits/joins), and <i>loops</i> . Obviously, these patterns should be supported by process mining techniques.
GP4	Events Should Be Related to Model Elements
	Conformance checking and enhancement heavily rely on the relationship between <i>elements in the model</i> and <i>events in the log</i> . This relationship may be used to “replay” the event log on the model. Replay can be used to reveal discrepancies between event log and model (e.g., some events in the log are not possible according to the model) and can be used to enrich the model with additional information extracted from the event log (e.g., bottlenecks are identified by using the timestamps in the event log).
GP5	Models Should Be Treated as Purposeful Abstractions of Reality
	A model derived from event data provides a <i>view on reality</i> . Such a view should serve as a purposeful abstraction of the behavior captured in the event log. Given an event log, there may be multiple views that are useful.
GP6	Process Mining Should Be a Continuous Process
	Given the dynamical nature of processes, it is not advisable to see process mining as a one-time activity. The goal should not be to create a fixed model, but to breathe life into process models such that users and analysts are encouraged to look at them on a daily basis.

Fig. 4: Six principles of Process Mining from the Process Mining Manifesto (van der Aalst et al., 2012)

B. Benefits - Why To Use it?

All in all, there have to be a good reason to create a Process Model using mining techniques, in this case, the reason why so many companies are applying them is because of its benefits. The main benefit of Process Mining techniques is that information is objectively compiled. In other words, Process Mining techniques are helpful because they gather information about what is actually happening according to an event log of an organization, and not what people think that is happening in the organization (Gupta & Gupta, 2014). This way of creating a model for our processes also provide us with the following advantages:

- High Scale analysis throughout the entire organization
- Accuracy based on facts
- No bottleneck, deviation or inefficient processes that should be rethought
- Continuously monitoring processes
- Use in ANY industry
- Applicable in EVERY area

C. How Does it Work?

Process Mining can be divided into the following steps:

- 1) **Reads digital footprint:** Every system generates a data footprint that Process Mining reads and transform in a process log to work with (In case it has not being previously provided).
- 2) **Model Generation:** Every system generates a data footprint that Process Mining reads and transform in a process log to work with (In case it has not being previously provided).

- 3) **Work over the model:** Once you know what is going on with your business, you can begin to fix all the problems. Process Mining allows you to focus on a specific area of business and you can start to prioritize and fix concrete problems from them.
- 4) **Monitoring:** Know that you have finish building your model and you have finish with all the necessary repairs, it's time to see if it works! For a Process Mining to be successful is key to keep monitoring the business in case new inefficiencies appears.

D. Use Cases of Process Mining

As you can deduce, Process Mining is very flexible and can be use in several scenarios, some of the main ones are:

- A business process in a company is growing, maybe scaled (e.g. vertically, horizontally) and getting more and more complex to understand, the company needs metrics and better abstraction in order to understand the process.
- A company is split in teams, services, departments, or especially geographical location, also the system computing and the data storage might be separated in different geographical locations. Then the teams want to have a unique and clear vision of the business process that is understandable by everyone (i.e. for diverse point of views)
- A business needs to form new contributors in an efficient way (no waste of time in rediscovering and trying to understand the process)
- A business use an old or complex workflow that need to be improved for diverse reasons (e.g. the workflow is not sufficient enough or new rules or policies induce to change it)

V. THE THREE TYPES OF PROCESS MINING

There are 3 main types of Process Mining that you can apply (or mix)

A. Discovery

One of the central tasks of Process Mining is Process Discovery, where knowledge is extracted from event logs (Ghawi & Pfeffer, 2016)). It Consist of going over the logs to automatically construct a model based on observable events from the logs, without outside influence. Since we want to apply Process Mining to workflow management systems to improve our businesses, we should develop algorithms able to cope with concurrency and complexity while avoiding overfitting or underfitting. Discovery is mainly use for discussing problems among stakeholders, generate improvement ideas form the model, enhance the model, or configure a Business Process Management System (BPM System).

Discovery is mainly used for:

- To find and discuss issues or weaknesses among stakeholders
- To improve the system (especially its process)
- To enhance the process models

- To configure a Business Process Management System (BPM System)

Since one want to apply Process Mining to workflow management systems to improve our businesses, we have to use algorithms that should be able to cope with concurrency and complexity while avoiding over-fitting or under-fitting.

There's 4 criteria (i.e., dimensions) that describe the quality of a process model:

- Fitness (ability to describe all cases, number from 0 to 1)
- Simplicity or Occam's Razor (simplest = minimal working process model)
- Precision (precise = does not allow to much behavior)
- Generalization (one model to describe all cases = not over-fitting)

B. Conformance

Conformance checking techniques relates events in the log to activities in the model (van der Aalst, 2012). It uses a model and an event log as input and checks whether the actual model that have been provided is in fact reflected in practice. It identifies any deviation of the model from real data and looks for overfitting or underfitting cases. It is mainly used to check the accuracy of a model, to identify deviation cases, to judge the quality of a model and as a starting point for model enhancement.

It is mainly used to:

- To check the conformance i.e., the accuracy of a model
- To check the quality of existing documentation
- To identify deviating cases
- To find fragments of the process
- To judge of the quality of a discovered process model
- To improve discovery algorithms
- Used as a starting point for model enhancement

C. Enhancement

It consists of extend or improve an existing model with the event log and additional information. It extends the previous model and could make it useful for predictions and recommendations.

VI. COMPARISON OF PROCESS MINING, BUSINESS ACTIVITY MANAGEMENT (BAM) AND DATA MINING

As we stated before, Process Mining is usually confuse with other techniques such as data mining or BPM. It's important to note that each one should be use in different use cases and therefore they should not be confuse or mistake. To understand this is important to define them and highlight their differents respect to Process Mining.

A. Business Process Management

BPM is a disciplined approach to identify, design, execute, document, measure, monitor and control both automated and

nonautomated business processes to archive consistent, targeted result align with organization's strategic goals. (Reinke-meyer Editor, 2020) The main different between BPM and Process Mining is that BPM focuses on "To be processes in order to describe how processes should work" (Reinkemeyer Editor, 2020) while Process Mining focus on modeling the processes that have already happened to get knowledge from then. One focus on describing how future processes should work (BPM) while the other focus on analyzing what already happened to learn from it (Process Mining).

B. Data Mining

On the other hand, people also tends to confuse Data Mining with Process Mining. Data mining is the process of discovering meaningful new correlations, patterns, and trends by sifting through large amounts of data stored in repositories, using pattern recognition technologies as well as statistical and mathematical techniques (Larose, 2014) whereas Process Mining is focusing on processes and not only the data that have been gathered. So the main difference between Process Mining and Data mining is that one is focusing only in data analysis where the other take a wider look and considered all the processes that have been carried on.

VII. DISCOVERY ALGORITHMS

The primary aim of Process Mining is to investigate a scalable solution that can evaluate, compare and renk these Process Mining algorithms efficiently. (Gupta & Gupta, 2014)

A. Alpha Miner

This algorithm helps us connect observed data and event logs in the discovery phase of a process model. It takes a set of events as input and return a model which represent the sequence of events that are more likely to happen. To do that, the alpha miner takes the set of events and transform them into a relationship with which it creates a petri net which describes the process model. For that, the alpha miner algorithm studies how the events have happened in the past, trying to find a pattern between them, like, for example, an event B that always occur after another event A have a dependency relation with A ($A \rightarrow B$, if A have not been execute, B could not be executed either).

There are 4 main relationships that alpha miner algorithms look for:

- 1) **Directly follows:** If $a > b$ (a is directly follow by b)
- 2) **Sequence:** $a \rightarrow b$ (a must be done before b ($a > b$) and not $b > a$)
- 3) **Parallel:** $a||b$ (a can be done before b and vice versa. Both, $a > b$ and $b > a$)
- 4) **Non direct relationship:** $a \# b$ (Neither $a > b$ or $b > a$)

With that information about relationships, we can create a footprint matrix as this one (Fig. 5):

All in all, the alpha miner algorithms follow these steps:

- 1) Define all events
- 2) Define all possible Start events
- 3) Define all possible End events

	a	b	c	d	e	f	g
<a,b,c,d,e,g>	#	→	→	→	#	#	#
<a,b,c,d,f,g>	#	←	#			#	→
<a,c,d,b,f,g>	#	←		#		→	#
<a,b,d,c,e,g>	#	←			#	→	#
<a,b,d,c,e,g>	#	#	←	←	#	#	→
<a,d,c,b,f,g>	#	#	←	#	←	#	→
<a,d,c,b,f,g>	#	#	#	#	←	←	#

Fig. 5: Example of a footprint matrix obtained by alpha mining

- 4) Calculate sets of independent events. All events inside a set are independent from each other and the sets are casually related among themselves
- 5) Drop Non maximum sets (avoid repetition)
- 6) Create places for all derived sets and add start and end states to them
- 7) Draw the connections
- 8) Return the petri net (as seen in **Fig. 6**)

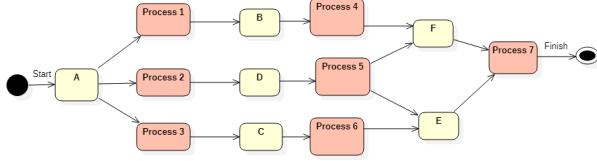


Fig. 6: Example of petri net from previous Matrix

B. Heuristic Miner

It can cope well with noise, and it's used to show the main behavior, which means that it does not repair on small details or and exceptions of the log but focus only on the main processes. Since it is a noise tolerant algorithm, it is applied to demonstrate how the model works in the presence of noise. Heuristic miner can be considered an extension of alpha miner which takes the frequency of traces in the log into account, mining the control flow of a process model. To do that, it only considers the order of events within a case, which means that it only takes into account the fields case id, time stamp and activity while doing the mining.

Heuristic miner follows these steps:

- 1) Read the log (like the one in **Fig. 7**)
- 2) Get the set of tasks
- 3) Infer the ordering relations based on their frequency
- 4) Build the net based on the inferred relations
- 5) Output the net (**Fig. 8**)

This algorithm uses the same basic order relationships that the alpha miner but including a frequency metric to indicate how certain we are about a relation. Thanks to its characteristics, heuristic miner is particularly useful to describe scientific workflows in which we have lots of tasks being executed in parallel, some of which have a long distance dependency which discovery is really important.

case id	activity id	originator	time stamp
case 1	activity A	John	9-3-2004:15.01
case 2	activity A	John	9-3-2004:15.12
case 3	activity A	Sue	9-3-2004:16.03
case 3	activity B	Carol	9-3-2004:16.07
case 1	activity B	Mike	9-3-2004:18.25
case 1	activity C	John	10-3-2004:9.23
case 2	activity C	Mike	10-3-2004:10.34
case 4	activity A	Sue	10-3-2004:10.35
case 2	activity B	John	10-3-2004:12.34
case 2	activity D	Pete	10-3-2004:12.50
case 5	activity A	Sue	10-3-2004:13.05
case 4	activity C	Carol	11-3-2004:10.12
case 1	activity D	Pete	11-3-2004:10.14
case 3	activity C	Sue	11-3-2004:10.44
case 3	activity D	Pete	11-3-2004:11.03
case 4	activity B	Sue	11-3-2004:11.18
case 5	activity E	Clare	11-3-2004:12.22
case 5	activity D	Clare	11-3-2004:14.34
case 4	activity D	Pete	11-3-2004:15.56

Fig. 7: Example of an event log

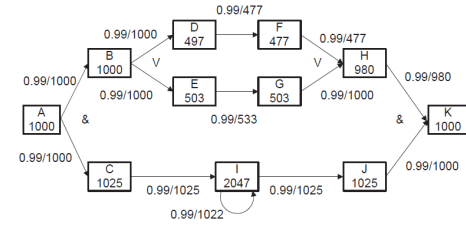


Fig. 8: Example of a petri net from the previous data

C. Genetic Miner

When we are working with processes inside a whole business environment, we can find very well-defined processes which are complete and perfectly related to each other. Nevertheless, we can also find incomplete processes with noise and not so much cohesion. This is when the generic algorithm kicks in. A Generic Miner algorithm is used when we must tackle with incompleteness, noise, duplicate or hidden activities... To deal with all of that, Generic miner generates a petri net which represents the process model from execution traces. It calculates the quality of fitness of a candidate by comparing the model itself with the event log which can be very useful when we are working on a globally optimal process but very dangerous when we are trying to use it in a local approximation, given that an erroneous example could mess up the entire model.

This algorithm has 3 main concerns:

- 1) **First:** Define the internal representation (Search space).
- 2) **Second:** Define the fitness measure (Quality of a point in the search space considering the logs).
- 3) **Third:** Selection of generic operators (Crossover and mutation) to ensure that all points may be reach.

The steps to create a generic algorithm are:

- 1) **Read the event log:** We start by reading and processing the event log that we have been provided with.
- 2) **Build the initial population:** Once we have read the log, we have to select the initial population. Since it is the first population, it can contain all individuals of the search space (defined by the number of tasks of the log) or we could use an initial selection criterion.
- 3) **Calculate the fitness of the individuals:** Now that we have the initial population selected, we need to calculate the degree of fitness of each individual so we can find the fittest. We will assign a fittest range from 0 to 1 to each individual. If a concrete individual is correctly describing the registered behavior of the log, the higher its fitness will be.
- 4) **Return the fittest:** To find the fittest individuals, we applied the following strategies:
 - a) **Elitism:** We copy a certain percentage of the fittest individuals to the next population.
 - b) **Crossover:** We generate new individuals (offsprings) from the fittest individuals (parents), recombining the fittest material hoping that the combination will result in a fittest population.
 - c) **Mutation:** It consists of applying small changes on the fittest individuals hoping that these changes will return a fittest population.
 - d) **Tournament Selection:** Consist of randomly selecting 5 individuals and returning the fittest of them all.
- 5) **Create next Population:** We create a new population to be tested based on the results of the previous selection. This algorithm will stop if we have find an individual which fitness is 1, we have compute the maximum number of generations or the fittest individuals have not change in $n/2$ generations in a row. If none of this conditions have been fulfilled, then the new population will be selected by following these steps:
 - a) Copy elitism rate X population Size of the best individuals of the current population.
 - b) While we have individuals to be created do:
 - i) Tournament selection on both parents
 - ii) Select a random number between 0 and 1 (If it is less than the crossover rate, do a crossover with the parents)
 - iii) Mutate the offsprings and copy then to the new population.
 - c) Return the new population and keep going until it ends (and gives a result like **Fig. 9**).

D. Inductive Miner

This algorithm is an improvement of alpha miner and heuristic miner, given that it provides a soundness process model with good values of fitness. This algorithm does not work on a petri net but on process trees (**Fig. 10**). It follows these steps:

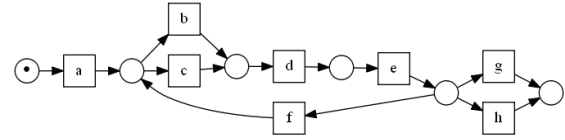


Fig. 9: Example of a petri net from a generic miner algorithm

- 1) Find a prominent split in the event log (For example, the initialization event).
- 2) Keep recursively going through the splits, searching for base cases and doing more splits.

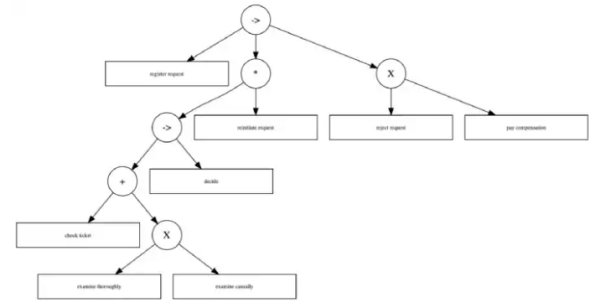


Fig. 10: Example of a process tree

E. Fuzzy Miner

This algorithm is especially useful when we want to mine less structured processes with a large amount of unstructured data. It also allows us to look at the process from different abstractions levels. It is used to avoid spaghetti like plots given that the fuzzy model it returns is a simplified version of the model or the unstructured data you have given it.

VIII. CASE STUDY

A. Manufacturing Process Analysis: a Case Study

A perfect use case study for us is the Samsung Electro-Mechanics Equipment Optimization Technology Group. The Samsung company along with the Ulsan National Institute studied how the data flows of the company are managed and how one could apply Process Mining to analyze manufacturing processes in Samsung Electro-Mechanic.

The company, Samsung Electro-Mechanics is one of the largest manufacturing companies in the East Asian region. They produce and manufacture high-tech integrated components and mechanical parts for electronic devices. All the data and MES of the company comes from the 30 divisions operated in Korea and overseas. The main data useful for exploitation is stored in event logs from the manufacturing execution systems (MES) in the company being analyzed with several Process Mining techniques.

A MES is a system that keeps track of the manufacturing processes and takes process-oriented information automatically, so it helps operation managers to make better decisions in manufacturing processes. Many manufacturing companies have adopted the MES system to provide the right information at the right time to control multiple elements of production processes. The MES uses a system where every lot (a basic unit in manufacturing) has a case ID number used to track the quantity or material. For this case study only lots from May to October 2012 have been tracked, giving us a total number of 11,226 cases and almost a million events. The study case was conducted with real event logs extracted from the Samsung MES to examine the validity of the framework that uses Process Mining techniques to help improve the manufacturing process with all the information processed by the framework.

In order to analyze all this data, the companies use conventional data analysis techniques such as statistics, data mining. These techniques have difficulties to provide overall results, such as overall performance, manufacturing patterns, etc. MES generates a lot of data and makes it more difficult to understand overall processes with such many data. Process Mining techniques make it easier to extract all the information from the logs of the MES, so it can exploit the data of it and provide an accurate view of the manufacturing process.

The framework used to extract, exploit and interpret the data is the following one, consisting of 4 major steps (Fig. 11):

- 1) **Data preparation:** Raw data is extracted from the MES system databases.
- 2) **Data preprocessing:** The raw data is filtered, refined and converted into a standard form (MXML)
- 3) **Manufacturing Process Mining and Analysis:** Several Process Mining techniques are applied according to the two perspectives of the model:
 - a) **Process:** A conduct process performance analysis (Bottleneck analysis, pattern analysis, conformance checking)
 - b) **Resource:** Resource performance analysis to find machine utilization
- 4) **Evaluation and Interpretation:** Results are evaluated and interpreted by decision makers. After that, the existing processes are improved based on the interpreted results.

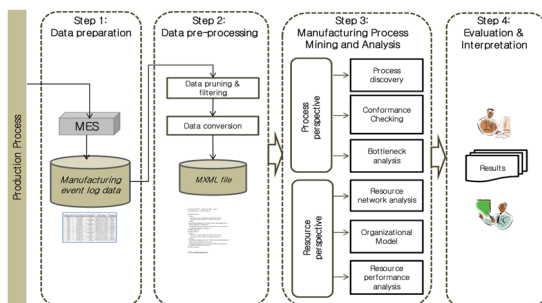


Fig. 11: Production Process

Process perspective analysis extracts a manufacturing process model and provides performance analysis results of the model. The Fig. 12 shown down shows the process model for one of Samsung Electro-Mechanics production processes, using the Heuristic mining algorithm [ref]. The model starts with the Input task and outputs the Packaging task and separates the manufacture process in three: preprocessing, main processing and inspection. The once the model is discovered, a conformance check is done. The objective of the conformance check is to provide a value (called fitness value) of the derived model. For this model the value found is 0.9962 and indicated that almost all the traces fit the model (being close to 1). But as it is not perfect another analysis was performed using an LTL (linear temporal logic) checker that verifies the validity of paths as well as a Bottleneck analysis that checks where bottlenecks happen. It found abnormalities on the flow (Fig. 13) going from Delivery Inspection to the Final Test meaning that some unexpected issues occurred from the delivery to the inspection.

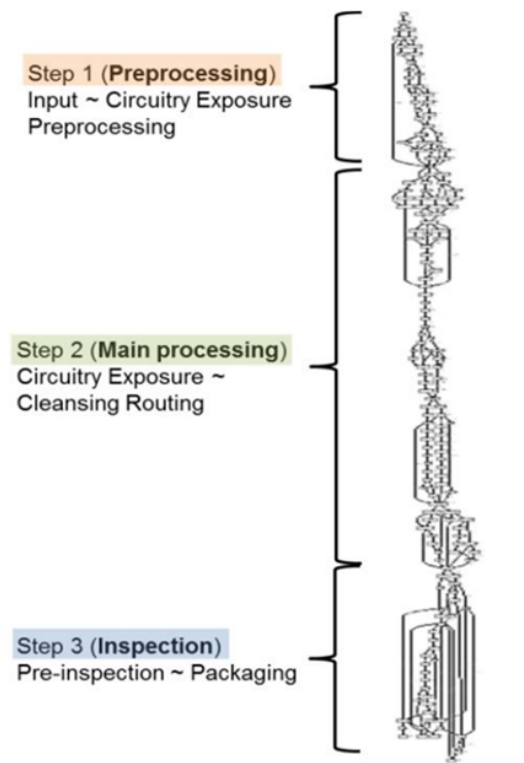


Fig. 12: Steps

Then again using the model and the raw data of the event log a machine performance analysis is done. That means we analyze what machine produces what lot and created a matrix of those (Fig. 14).

There was found to be a huge imbalance between the different machine with half the machines producing less than 600 lots and one machine producing 5612. Added to that another analysis of the average working time for each machine was realized and found that relative to production the N28026

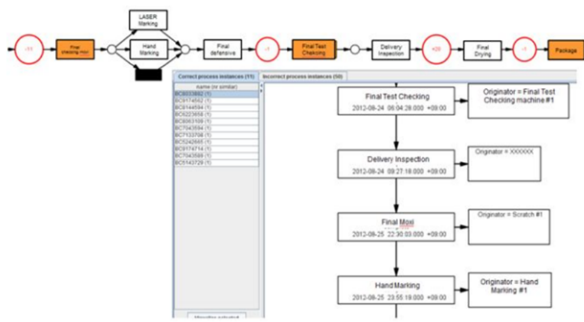


Fig. 13: Flow Diagram

Machine	Package
N28026	5,612
N28027	2,965
N20031	2,143
N20030	500
N14074	2
N28025	2
N28146	2

Fig. 14: Table

is the most efficient and N20031 has the largest average per production meaning the machine is the least efficient. To investigate the inefficiency of the N20031 machine, the event log was used, and it showed that 90% of the cases are finished in under 5 hours but some fringe cases could take up to a day thus reducing enlarging the production costs and reducing the machine efficiency.

B. Conclusion

The main contribution of this case study is that the proposed framework can be applied to any manufacturing process analysis and can provide insight into the current manufacturing processes. The derived discovered process model (due to the heuristic algorithm) shows the flows of processed in the factory and can be used as a basis for further analysis of how to reduce bottlenecks, how to mainstream some superfluous flows or which machines of the flow are more efficient than others. In this case the event log showed an already streamlined flow of the processes of Samsung Electro-Mechanics but in another company the process model derived from the event log with a simple analysis could lead to huge organizational gains as it would easily show the inefficiencies and how to tackle them.

IX. IMPLEMENTATION AND TESTS OF DISCOVERY

An idea to complete our analysis was to produce an event log, feed the the event log as the data input and use an open source discovery algorithm in order to discover the model. So accordingly, we developed a small concurrent GoLang application that generates a random event log following some specific rules that were chosen by us.

Before implementing the event log generator, we defined the rules and drew a business process model (as can be seen

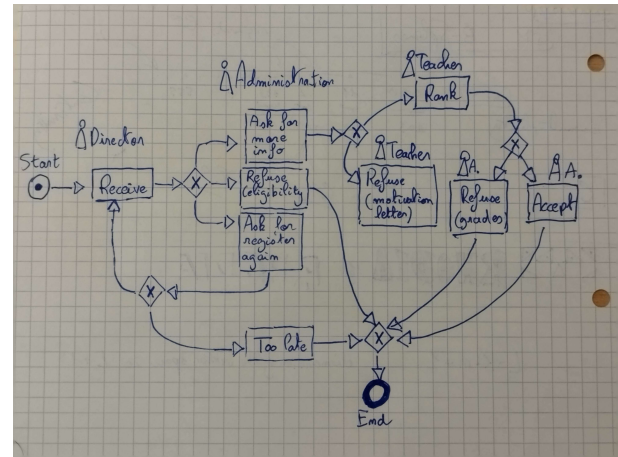


Fig. 15: Process model drawn before implementing code

in Fig. 15).

This application creates a number of random students (here the number being defined by n) who want to register in an university. The people in charge of the student registration (called Responsible people) will ask students to provide more information if they did not sent enough, refuse or accept their candidacy. Also, after a deadline, all activities are stopped and the whole process is terminated.

The program works concurrently and takes in account the fact that some students take a random amount of time to provide their information. The people in the university, with specific roles, do their tasks and take some time to do it. Therefore, the tasks are stored in a FIFO (First In, First) list before being processed.

It's for this need of concurrency and easy management of channels that we decided to use the programming language Go, a simple compiled programming language that is very powerful for concurrent programs.

When running the program, a new event log is generated in JSON and in CSV. This event log (Fig. 16) contains 1 process with 100 different students, and many more activities, each related to one of the 100 students. Each activity has a timestamp.

With several event logs generated (that are not the same because of the random generation in the application), we want to test Alpha Miner algorithm to produce a process model.

We chose a popular open-source tool for Process Mining: **Prom Lite 1.3**, <https://www.promtools.org/doku.php>. With this tool, we parsed the event log data into a so called XES format that knows which column is grouping activities together, and which column is defining the tasks (Fig. 17).

With the XES format, we can use the **AlphaMiner algorithm** to discover algorithm to produce a BPM (Fig. 18) . The

Configure CSV Parser Settings

CSV Parser: Settings

Charset: windows-1252
Separator Character: Comma (,)
Quote Character: DOUBLE QUOTE (")

title	description	idRelatedTo	nameUser	roleResp...	timestamp	date
Receive	Receive re...	31	Finn	Director	5	2022-12-1...
Refuse (eli...	The univer...	31	Finn	Administra...	8	2022-12-1...
Receive	Receive re...	99	Sean	Director	12	2022-12-1...
Ask more l...	Your regist...	99	Sean	Administra...	13	2022-12-1...
Receive	Receive re...	25	Jay	Director	18	2022-12-1...
Refuse (eli...	The univer...	25	Jay	Administra...	19	2022-12-1...
Receive	Your regist...	3	Zayn	Director	22	2022-12-1...
Ask more l...	Your regist...	3	Zayn	Administra...	23	2022-12-1...
Rank	We are ha...	99	Sean	Teacher	25	2022-12-1...
Refuse (m...	We are de...	3	Zayn	Teacher	26	2022-12-1...
Receive	Receive re...	49	Luke	Director	30	2022-12-1...
Ask regist...	The regist...	49	Luke	Administra...	31	2022-12-1...
Receive	Receive re...	76	Cash	Director	35	2022-12-1...
Refuse (eli...	The univer...	76	Cash	Administra...	39	2022-12-1...
Receive	Receive re...	51	Myles	Director	43	2022-12-1...
Ask more l...	Your regist...	51	Myles	Administra...	44	2022-12-1...
Receive	Receive re...	53	Luke	Director	47	2022-12-1...
Ask regist...	The regist...	53	Luke	Administra...	48	2022-12-1...
Rank	We are ha...	51	Myles	Teacher	49	2022-12-1...
Receive	Receive re...	82	Cruz	Director	54	2022-12-1...
Refuse (eli...	The univer...	83	Cruz	Administra...	56	2022-12-1...

Fig. 16: Process model drawn before implementing code

Configure Conversion from CSV to XES

Mapping to Standard XES Attributes

Case Column (Optional)
Groups events into traces, and is mapped to 'concept:name' of the trace. Select one or more columns, re-order by drag & drop.

Event Column (Optional)
Mapped to 'concept:name' of the event. Select one or more columns, re-order by drag & drop.

Start Time (Optional)
Mapped to 'time:timestamp' of a separate start event

Completion Time (Optional)
Mapped to 'time:timestamp'

Cancel Previous Next

Fig. 17: Process model drawn before implementing code

generation is surprisingly close to the process drawn at first (Fig. 15). We can see that Alpha Miner produces a rather high quality model.

- 1) **Fitness.** We can find that there is the process going from *Ask register again* to *Receive* that allows a student to register again before deadline is reached. However it turns out this was a mistake in the source code we developed, therefore we can consider this model to have a perfect fitness.
- 2) **Simplicity,** the model seem to be as simple as possible, each activity is shown once in the model, we can see that there is no superfluous activity or extra arrows.
- 3) **Precision,** our workflow being very simple, it's quite hard for a model to generate a non-precise model, the one generated has a good precision, that we can consider ideal, so it is not under-fitting.
- 4) **Generalization,** again, the workflow we imagined is simple enough to be easily generalized, so the model generated is not over-fitting.

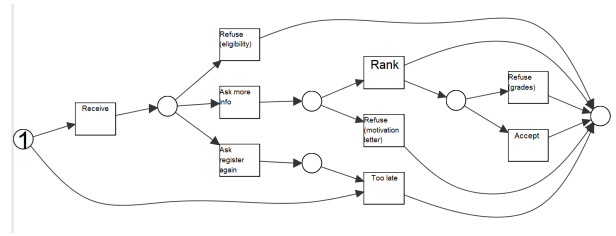


Fig. 18: Process model drawn before implementing code

This specific implementation, the test of a Process Mining tool and the use of Alpha Miner algorithm let us put our knowledge into practice and discover the potential of these tools.

A. Source Code

All the source code of the Go application is available in this GitHub repository: <https://github.com/MarioVieilledent/ProcessSimulator---Process-Mining>

X. CONCLUSION - THE POTENTIAL OF PROCESS MINING

In summary, Process Mining is a great strategy to analyse and monitor business processes in order to get a better understanding of them. In contrast to data mining, the focus is on the processes and not the data.

Process Mining pursues various goals, on the one hand, an analysis of a business model is carried out with regard to accuracy on the basis of facts, on the other hand, the processes are continuously monitored and over-folded so that bottlenecks and deviations can be discovered.

Process Mining also brings many advantages, it can reveal it very useful for any company to apply Process Mining ideas to improve quality and productivity. Firstly, it can be applied in any industry and in any field. Secondly, companies can save costs, increase productivity and improve performance with the help of Process Mining.

In addition, there are various open-source tools and algorithms that support and promote Process Mining. Although a lot of papers have been published, a lot of Process Mining algorithms are developed and it exists already several Process Mining tools, there is still a lot to discover and to research.

Keeping this in mind, it still has a lot of room to grow and evolve. It is believed that it will lead us towards great improvements in the near future.

REFERENCES

- [1] Ghawi, R., & Pfeffer, J. (2016). Mining social networks from linked open data Springer International Publishing. doi:10.1007/978-3-030-23182-8_16
- [2] Gupta, E., & Gupta, A. E. (2014). Process Mining algorithms Process Mining algorithms
- [3] Larose, D. T. (2014). In Larose C. D. (Ed.), Discovering knowledge in data : An introduction to data mining / daniel T. larose, chantal D. larose (2nd ed. ed.) Hoboken, New Jersey : IEEE.
- [4] Reinkemeyer Editor, L. (2020). Process Mining in action : Principles, use cases and outlook. Cham: Springer International Publishing. doi:10.1007/978-3-030-40172-6

- [5] WIL VAN DER AALST, Eindhoven University of Technology. Process Mining: Overview and Opportunities
- [6] van der Aalst, W. (2012). Process Mining. *ACM Transactions on Management Information Systems*, 3(2), 1-17. doi:10.1145/2229156.2229157
- [7] van der Aalst, W., Adriansyah, A., de Medeiros, A., Karla Alves, Arcieri, F., Baier, T., Blickle, T., . . . Wynn, M. (2012). Process Mining manifesto. Paper presented at the , 99(1) 169-194. doi:10.1007/978-3-642-28108-2_19
- [8] Sangsu Choi, Sookyoung Son, Minseok Song, Yong Jang (2014). Process Mining for manufacturing process analysis: A case Study.