Business Process Intelligence Challenge 2019 - Hierarchical process deviation analysis using evolutionary model discovery

Max Adaloudis[1], Veronika Cucorova[1], Koen van der Leij[1], and Koen Minartz[1]

Eindhoven University of Technology, Eindhoven, The Netherlands

Abstract. Process Mining nowadays is an extensively used field in process management to extract insights from event logs. The Business Process Intelligence Challenge of 2019 poses a problem and provides a real-life event log from a large multinational company in the area of coatings and paints. This paper uses process mining to find multiple measures of deviations within four purchase order types. Deviations are measured with respect to process flow, invoice matching and rework of purchase orders. To describe the four processes, we used evolutionary model discovery techniques and multi-level data analysis. For deviations regarding the process flow, we consider the three hierarchical levels event, case, and purchase order to analyze structural performance changes within the event logs. Our findings suggest that multivariate data analysis combining trace attributes with event data used to discover process deviations can provide insights to improve current process flow.

Keywords: Business Process · Intelligence · BPIC2019 · Process Mining · Evolutionary Model Discovery · Hierarchical Process Mining · Process Deviation Analysis · ProM

1 Introduction

A BPI challenge uses real life event-log and challenges participants to solve the process owner's questions or find unique insights into the process. There is no restriction on tools used by the participants. The participants will present their findings to a jury in the form of a paper, after which the winner is selected for each of the categories. The Business Process Intelligence Challenge 2019 (BPIC 2019) consists of two categories, student and non-student, where the student category is expected to answer one or multiple of the process owner's questions in depth and the non-student category are expected to focus on a broader range of aspects.

1.1 List of definitions

Purchase document	Also called **Purchase order**, represents a document that is issued to order one or more purchase items.
Purchase item	Also called **Line item**, represents one case (also named trace) within an event log.
Event	An activity occurring within a process logged within an event log.
Event log	Also called **Dataset**, represents the database containing process data composed of traces containing events.
ProM	Open source software platform allowing various process mining techniques.
Disco	Process mining tool from Fluxicon.
Petri net	Modelling language describing distributed systems using nodes called places to describe pre-conditions and post-conditions and transitions between places to describe activities.
Precision	Measure of how much of the behaviour present in the log is covered by the model. The higher value, the better, low value implies the model does not explain the data well.
Fitness	Measure of how much of the behaviour allowed by the model is actually present in the log. The higher, the better, low value implies that the model is too general and not many insights can be derived.

1.2 Data BPIC 2019

The 2019 BPIC uses real life data collected from a large multinational in the Netherlands. From now on this multinational will be referred to as the process owner. The challenge description specifies four process flows, into which the items can be partitioned according to the way their order is handled and which documents are required. A purchase document contains one or multiple line items. Moreover, multiple goods receipt messages and corresponding invoices for each item can exist. The process is defined to be compliant when for each line item the amounts of the line item, goods receipt messages and corresponding invoices are equal. The four flows for line items are following:3-way matching, invoice after goods receipt, 3-way matching, invoice before goods receipt, 2-way matching (no goods receipt needed), Consignment.

The provided dataset is anonymized and consists of over 1,5 million events for purchase orders submitted in 2018. This data contains the process of handling the purchase orders and information about the item categories, vendors and documents.

- The resources are split between fully anonymized batch users and normal users. The batch users are automated processes executed by different systems. The normal users refer to human actors in the process.
- The values of each event are fully anonymized from the original data using a linear translation respecting 0, i.e. addition of multiple invoices for a single item should still lead to the original item worth (although there may be small rounding errors for numerical reasons).
- Company, vendor, system and document names and IDs are again fully anonymized in a consistent way throughout the log. The process owner has the anonymization key, so any result can be translated by them to business insights about real customers and real purchase documents.

The event-log is in .XES format and is IEEE compliant. A total of 76,349 purchase documents containing 251,734 items are in the event-log. These items or cases consist of 1,595,923 events relating to 42 activities performed by 607 human users and 20 batch users. If the user field is empty or NONE, this indicates there is no user recorded in the system. For each case, i.e. item, several features are recorded. The features and their explanation can be found on the website that hostes the challenge [1].

The features can be split among the documents, items and events as can be seen in the entity-relationship diagram 1.

1.3 Challenge

The process owner has several questions that can be answered in the challenge. The following questions are of interest to the process owner [1]:

- Is there a collection of process models which together properly describe the process in this data. Based on the four categories above, at least 4 models are needed, but any collection of models that together explain the process well is appreciated. Preferably, the decision which model explains which purchase item best is based on properties of the item.
- What is the throughput of the invoicing process, i.e. the time between goods receipt, invoice receipt and payment (clear invoice)? To answer this, a technique is sought to match these events within a line item, i.e. if there are multiple goods receipt messages and multiple invoices within a line item, how are they related and which belong together?
- Finally, which Purchase Documents stand out from the log? Where are deviations from the processes discovered in (1) and how severe are these deviations? Deviations may be according to the prescribed high-level process flow, but also with respect to the values of the invoices. Which customers produce a lot of rework as invoices are wrong, etc.?

The decision of our team was to put focus on the third question. As for this question it is necessary to at least partially solve the first question as well, the following report will cover the identification of the deviations within the process preceded by creation of a suitable set of models. The following section closely

reflect the approaches taken in chronological order.
Section 2 describes the cleaning of the event log performed before the model creation. In section 3 various insights and relationships of different concepts in the data are described. In section 4 we provide an elaborate overview of the model creation and a thorough description of the models themselves. After the models were created, we aimed to determine the documents that do not conform to the defined standard processes by the models, which is described in section 5.

2 Data cleaning

The provided event log had a number of problems which needed to be addressed before performing further analysis. The following section provides a clear overview of the steps taken in in order to maximize the value of future findings and models constructed using the cleaned event log.

2.1 Missing values

Of all the features, only three had any missing values summing up to total of 3289 cases with missing values for 16294 events. This corresponds to about 1% of all cases. The features with missing values for these cases are:

- case Spend area text
- case Sub spend area text
- case Spend classification text

Interestingly, for each activity in a case these features are missing. To indicate so, the values is replaced with *unknown* as we do not consider it necessary to discard these cases, because the problem is not very severe. Moreover in the future these values might be traced back if their absence provides any insights.

2.2 Dates

Secondly, we examine the time frame and duration of the purchasing documents. Since the data description [1] indicates all data should occur in 2018 the events, cases and purchasing documents occurring outside of 2018 have to be looked into.The distribution of all the events, the starting events and ending events of purchasing documents is shown in Table 2. There are purchasing documents starting and ending before and after 2018, whereas the data should only contain purchasing documents created in 2018. For the purchasing documents starting after 2018, the assumption is made that these are future orders that are placed by clients that order ahead. For the cases that end after 2018 but start in 2018, a similar assumption is made, namely that activities are scheduled in the future necessary for the completion of the order and can only be done after certain time elapsed. For the cases that have a date earlier than 2018, it is assumed that this data is wrongly registered in the log, based on the challenge description. To investigate these purchasing documents further, the events that happen before

Table 2. Events, starting and ending starting purchasing documents per year

Year	Events	Purchasing Documents starting	Purchasing Documents ending
1948	10	1	0
1993	9	1	0
2001	22	15	0
2008	45	1	0
2015	3	2	0
2016	6	2	0
2017	223	75	0
2018	1550468	76241	65268
2019	45135	11	11079
2020	2	0	2

2018 are looked into. Only three out of forty-two event types occur before 2018, these event types can be seen in Table 3, with the value count and the time of the event occurring.

Table 3. Events, starting and ending starting purchasing documents per year

event concept:name	count	unique	time of events
Create Purchase Requisition Item	7	1	00:00:00
Vendor creates debit memo	27	1	23:59:00
Vendor creates invoice	284	1	23:59:00

As wrongly entered data and time might imply wrong order of the activities within a trace the constructed models might be based on a wrong order of the activities and also wrongly computed. Sub-optimal performance of a document corresponding to unusually long duration. Thus, it is decided to drop purchasing documents that start before 2018. All events connected to this purchasing document, including the ones happening in 2018 or later, will be dropped to guarantee the validity of the data.

3 Data analysis

Regardless of the purpose for which the data is used in the future, after the data cleaning we perform an initial data analysis to get general statistics and insights about the event log. Firstly the relationships between different concepts are investigated, after which we look at the distribution of different variables in the whole event log and their distributions within a trace. We analyze the frequency of different activities and dedicate a whole subsection to the time analysis.

3.1 Entity relationships

From Figure 1, which was constructed examining the given data, it can be derived that in one purchase document multiple items can be ordered. For all of these items the attributes specific to the purchase document are the same. It is always the same vendor delivering all the items ordered within one document. Interestingly, the Goods receipt flag is specific to purchase document, but GR-Based Inv. Verif. is not. From this it also follows that one purchasing document might contain items of different categories. One item, or case, contains one or multiple events, which differ in their timestamp, user that generated the event and other attributes. Surprising is that Document Category name is an attribute that varies within a single item within a single document.
The diagram also shows the updated counts for all of the three hierarchical entities present in the event log after the data cleaning.

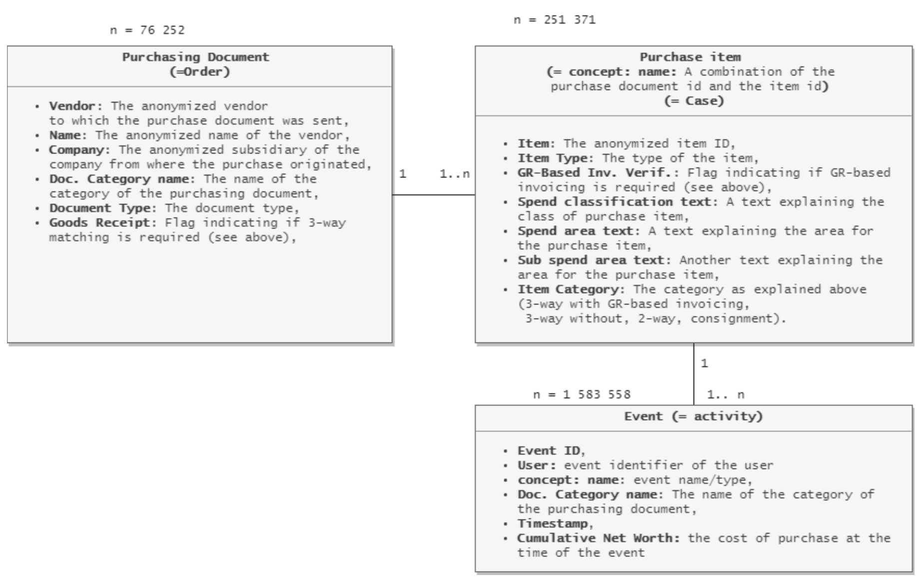

Fig. 1. Purchase documents, Items and Events shown in an entity relationship diagram showing their hierarchical relationship.

3.2 Activity frequencies

In the preprocessed log there are 42 different activities, of which following seven: *Record Goods Receipt*, *Create Purchase order Item*, *Clear Invoice*, *Record Invoice Receipt*, *Vendor creates invoice*, *Record Service Entry Sheet*, *Remove payment block* together create 90% of all events.

3.3 Attribute distributions

Firstly, the company distribution is very uneven. Out of the 4 companies present in the log the company with *ID_ 0000* is attributed to 99.66% of the events within the event log. Similarly, the Document Type is unevenly distributed, 97% of the events having *Standard_ PO* and *Framework order* and *EC Purchase order* only making small percentage of all. Of item categories *3-way match, invoice before GR* makes 77.9% of all events, followed by around 19.5% of event attributed to *3-way match, invoice after GR* and the rest split among *Consignment* and *2-way match*, see figure 2. For Item Type, the most occurring type is *Standard*, followed by *Service*, while the rest occur very little as shown in figure shown in Figure 2 as well. The most occurring values for Spend area text are *Packaging*, *Sales*, *Logistics* and *Trading & end products*.

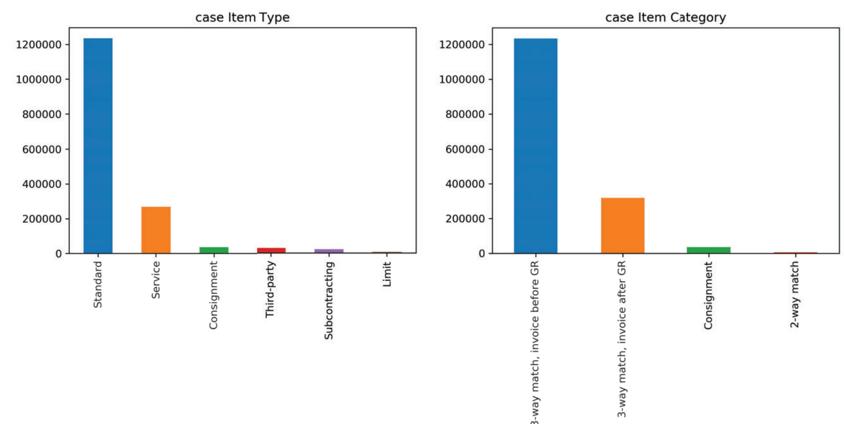

Fig. 2. Histogram showing the amount of events for each value for Item type and Spend area text

3.4 Time analysis

The median case duration after removal of the purchase documents with events from before 2018 is 64.3 days. The average lies at 70.6 days implying there is a small number of outliers that take much longer, which are probably the 5 cases that end after 2018 due to the assumed scheduled events in the future. The distribution of events generated on a certain day follows are regular interval of a work week, implying much higher number during a working day and a drop of the number over the weekends. Besides these drops, the number of events also drop heavily around Christmas period. Last regular pattern is a peak number of events generated on a single day, which happens monthly from March until November always between the 25^{th} and 30^{th} day of the month.

The number of cases open grows linearly from the beginning of the year until the 26^{th} of March at which point there are around 32000 case open. From April onward, the number of cases does not grow significantly anymore, but there is always a monthly peak of the number cases open between the 25^{th} and 30^{th} day of the month. The sudden drop after this day correlating with the peak in the number of events on that day suggest the events are closing events for these cases. On the 1^{st} of January 2019, there are still 20000 cases open which decreases to 5 until the 19^{th} of January.

Table 4. Events, starting and ending starting purchasing documents per year

Variable	Purchasing Document (max unique)	case concept:name (max unique)	Attribute of
case Goods Receipt	1	1	Purchasing Document
case Company	1	1	Purchasing Document
case Document Type	1	1	Purchasing Document
case Purch. Doc. Category name	1	1	Purchasing Document
case Vendor	1	1	Purchasing Document
case Name	1	1	Purchasing Document
case Source	1	1	Purchasing Document
case Spend classification text	2	1	case concept:name
case GR-Based Inv. Verif.	2	1	case concept:name
case Item Type	2	1	case concept:name
case Item Category	3	1	case concept:name
case Spend area text	4	1	case concept:name
case Sub spend area text	7	1	case concept:name
case Item	429	1	case concept:name

4 Models

It is desirable to produce a process model in the form of a classical Petri net, allowing for conformance checking with the original data and extensive analysis.

Multiple quality measures are taken into account when constructing the model. Further on, the generated process model will define the standard behaviour that will allow for deviation identification.

4.1 Model creation methodology

We split the data among the four line-matching categories as mentioned in the challenge description [1]. This is done because the four categories represent distinct process flows, specifically when it comes to 2-way and 3-way matching. The four process models created from these matching categories can then be properly compared against the event log, analyzing how a case in a category deviates from its respective process model. Additionally, as shown in Figure 2, the case item categories specifying the type of matching used, are not distributed evenly among the cases.

For model discovery, we used the ETMd Pareto Front Miner [2] in Live Mode. The ETMd algorithm is an evolutionary miner that creates random variations on process models at each generation. These random variations can then be used to propagate different types of models, ranging from relatively simple models (accuracy) to extremely complex models (fitness). At each generation, a population of new process models is created with the results of the previous generation as input, combined with a random factor. The algorithm trains on the previous generations and keeps the best mutations as an elite population to be passed along to the next generation.
The ETMd miner's Pareto front shows process models against two performance measures, where we chose fitness and precision to be determinant. An example of the Pareto front can be seen in seen in Figure 10. As we wish to analyze deviations, we typically do not want extremely high fitness, as it would cover deviations in the event log. However, achieving an equally high precision would result in an overly simplistic model, resulting in non-deviating but less frequent cases as deviating.
Using the ETMd miner, we created several process models for each matching category, ranging from higher-fitness/lower-precision to lower-fitness/higher-precision and compared these against the event log. For the ETMd parameters, we used a *population size* of 50 and an *elite count* of 5. We ran the ETMd miner for 500 generations for each of the four matching categories, enough for the Pareto front to converge. After a certain number of generations, the Pareto fronts were near-optimal and a marginally better model would have been found after an extremely long time or not at all in the case that there are no better models.
We used the Multi-perspective Process Explorer [3] plugin within ProM to do conformance checking of the process models generated from the ETMd miner, with the results denoted in table 5. More importantly, we used the Multi-perspective Process Explorer to export the alignments results between the created models and the event log, to be analysed in more detail in the following sections.

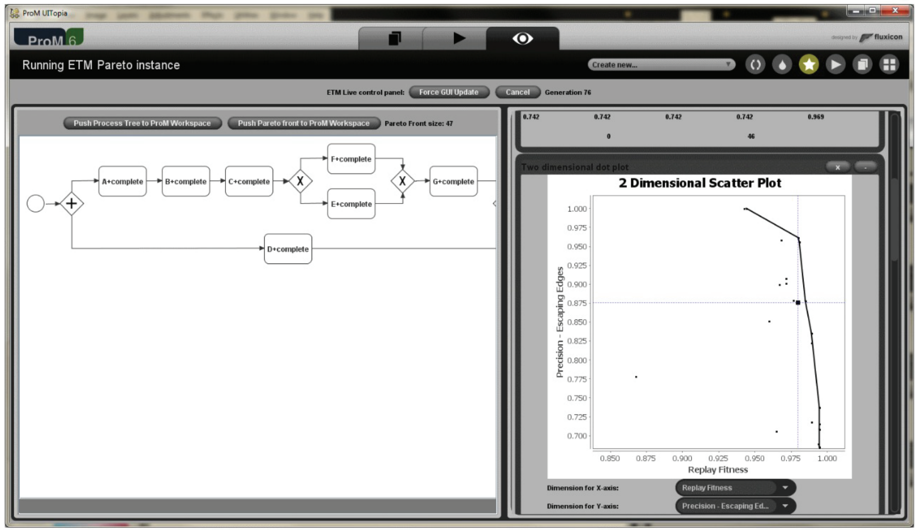

Fig. 3. Live Mode View view of the ETMd miner - Buijs, J. C. A. M. (2014). Flexible evolutionary algorithms for mining structured process models.

Category	Fitness	Precision
2-way match	87.2	92.6
Consignment	94.4	97
3-way-match Invoice after GR	87.3	84.6
3-way-match Invoice before GR	93.6	91.9

Table 5. Fitness and precision for Petri net models created against the event log of the corresponding category

4.2 Final models

Following section presents Petri nets corresponding to the process flow of one of the Item Categories. We provide a figure exported from the Multi-perspective Process Explorer with no modifications, despite the fact that automatically generated Petri net might contain redundant transitions or places that do not allow for extended behaviour. Instead we aim to interpret each of them and reflect on whether the model allows for behaviour that it should allow. The Petri nets contain black transitions with no name which shall be interpreted as silent (tau) transitions, which change state and correspond to no real activity of the system. Despite the fact, we use the Petri nets to check for conformance with the data, we also provide a process map from Disco that shows the absolute number of cases following certain paths.

2-way match The model for 2-way match4 contains a split initially, in which either *Vendor creates debit memo* happens, or *Vendor creates invoice* happens, or an arbitrary number of *Change Approval for Purchase Order* happens. These three options are structured in an or-split which mean only one of them shall happen in the model. Afterwards, *Create Purchase Order Item* should always happen, followed by again arbitrary amount of *Change Approval for Purchase Order*. Afterwards, both *Clear Invoice* or *Record Invoice Receipt*.

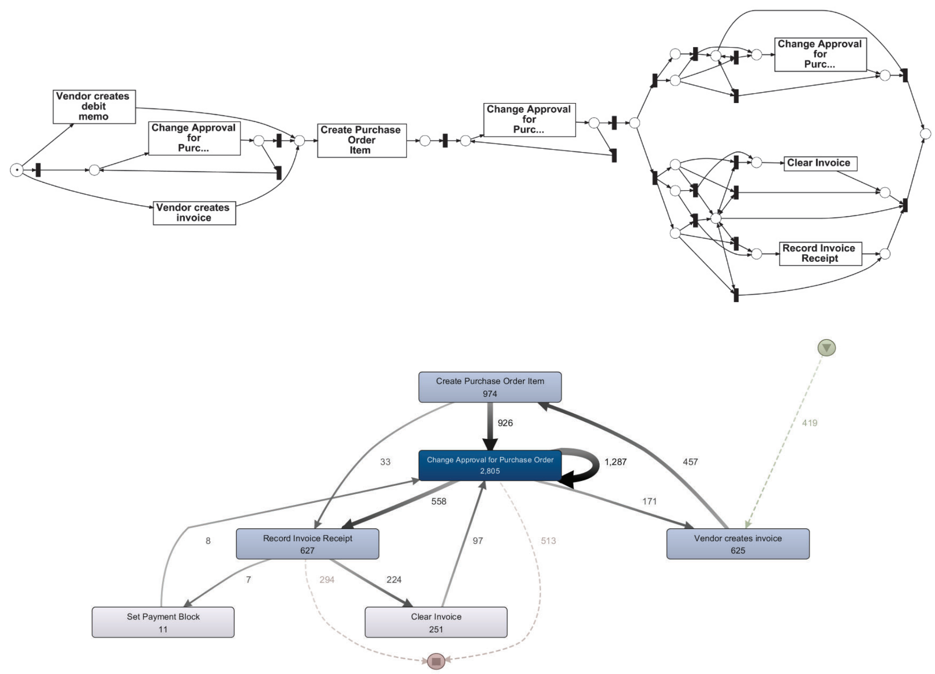

Fig. 4. Petri net used for the 2-way match flow and a process map reflecting the data.

Consignment The Petri net 5 is considerably straightforward in its initial part and the or-split can be interpreted as the process starting either with *Create Purchase Order Item* or *Change Price*, exclusively. Interestingly, no case within 2-way match data set after cleaning starts with the *Change Price*, neither there are cases in which both would be never followed by one another(or in another words exclude each other) as the model presents it. This is however not an issue

as the *Change Price* has relative frequency of 0.12%, so even if the model does not reflect the process perfectly, placing the *Change Price* activity in or-split with *Create Purchase Order Item* does not cause a problem.
Second part of the Petri net, following the silent transition has 5 activities: *Change quantity*, *Receive Order Confirmation*, *Delete Purchase Order Item*, *Record Goods receipt* and *Change Delivery Indicator*, which are in or-split, but the silent transition backwards allow for any order and any number of repetitions of these activities. This raises a question whether the model is not too loose and allowing for behaviour not present in the data, but the precision of the model is 97%, which ensures that the behaviour in the data is indeed closely represented by the model. The process map suggests that a few activities might happen after the order is created, in most of the cases, the last activity is *Record Goods Receipt*.

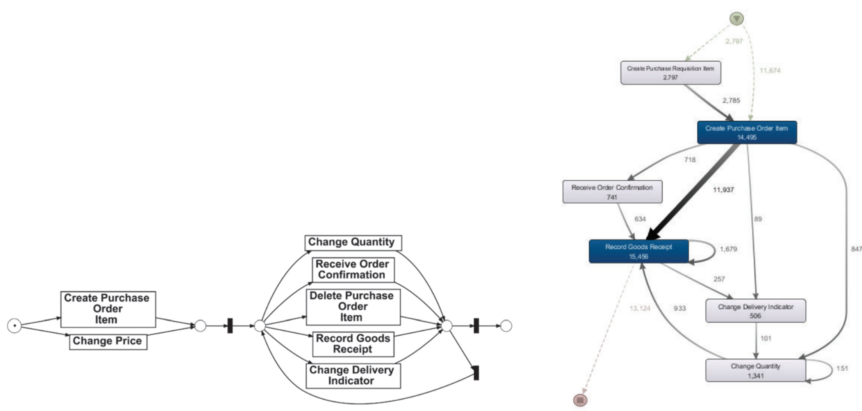

Fig. 5. Petri net model used for the consignment flow and a process map reflecting the data.

3-way match - Invoice after GR *Create Purchase Order Item* activity is optionally followed by *Receive Order Confirmation*. Afterwards there is an and-split, which implies that Vendor creates invoice should always happen at some point in the process before *Record Invoice Receipt*. In parallel, either different combinations of *Record Service Entry Sheet*, *Record Goods Receipt* and *Change Quantity* may happen, varying in order and the number of activities. An alternative to this is the combination of *Change Price* and *Vendor creates invoice* in any order. After *Record Invoice Receipt*, *Clear Invoice* and *Remove Payment Block* happen in arbitrary order.
Considering the process map 6, it is expected that the automatically generated SRM processes will be marked as non-conforming to our created model. The process map also shows many times after clearing an invoice, new invoice was created, which is also not accepted by the created model and will likely be marked

as non conforming in the deviations analysis. After examining the process map, it also becomes clear why the loop in the central part of the Petri net was added by the mining algorithm, considering the large number of repetitions of *Record Service Entry Sheer*, highlighted in dark blue.

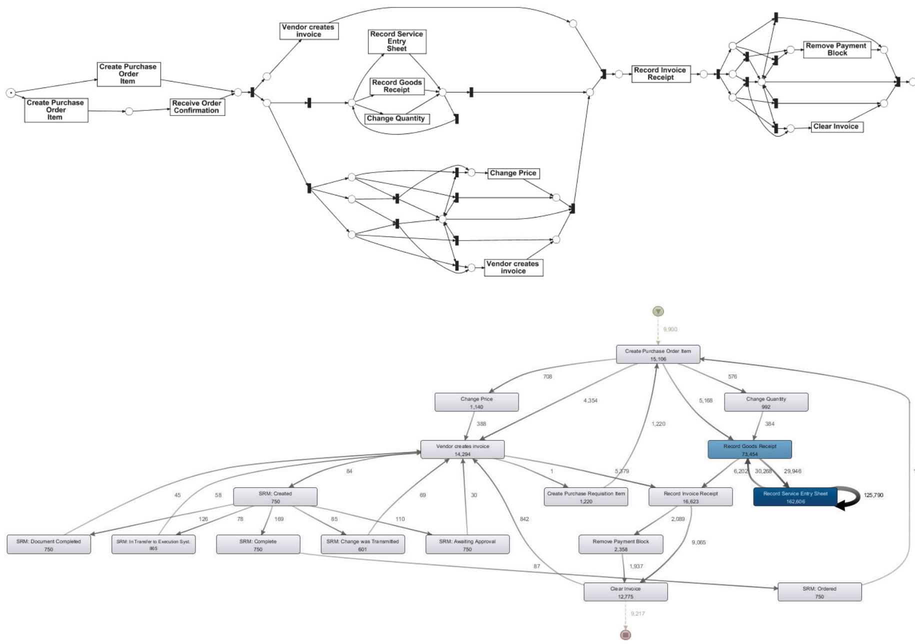

Fig. 6. Petri net used for the 3-way match - invoice after GR flow and a process map reflecting the data.

3-way match - Invoice before GR The process model 7 for this Item category is again rather straightforward, considering that this is the largest category of the four. The process model requires *Create Purchase Order Item* to happen first, after which it is necessary that *Record Goods receipt* happens sometime in the process before *Clear Invoice*. Interestingly, the model does not require *Vendor creates Invoice* to happen before the goods are receipt as the name of the item category suggests. Optionally, before *Clear Invoice*, *Remove Payment block* may happen.

5 Deviations identification

The data is analyzed to identify deviations, which is done to give an insight into which features create these deviations. The deviations are distinguished from

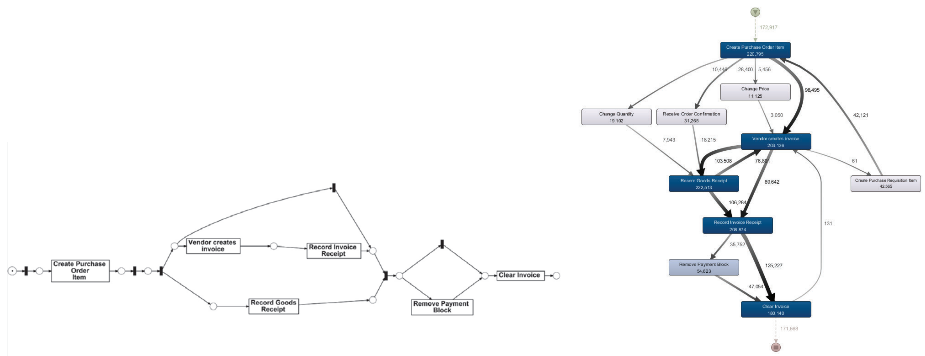

Fig. 7. Petri net used for the 3-way match - invoice before GR flow and a process map reflecting the data.

the standard process based on the models previously described for each Item Category. The process owner can use this information to look further into these features and see if improvements can be made to solve the deviations or find underlying reasons for the deviations.

5.1 Alignment fitness analysis

For the first of the analyses, we use the trace fitness calculated based on the models introduced in section 4.2 to see which factors have influence on the degree to which a case complies with the business process. In order to do so, it is assumed that the model represents the true process of the company. This assumption is reasonable as the fitness and precision of the models are quite high.

First, the mean case fitness is evaluated over the course of 2018. This is done for cases in general, but also for each of the four flow types. The result is depicted in Figure 8.

For all categories, the fitness remains more or less constant up until August; however, the case fitness rapidly decreases after that month. This suggests that the process flows have changed from those of the models depicted of section 4.2.

Besides time, other factors could be correlated with the fitness of a case. It was found that the case fitness exhibits significant correlations with the throughput time of a case and the cumulative net worth of the events of that case. The results are displayed in Table 6.

Clearly, cases that have a longer throughput time tend to comply less with the models, as one could have expected. interestingly, cases that tend to have events with a high cumulative net worth attribute, also tend to have a lower fitness.

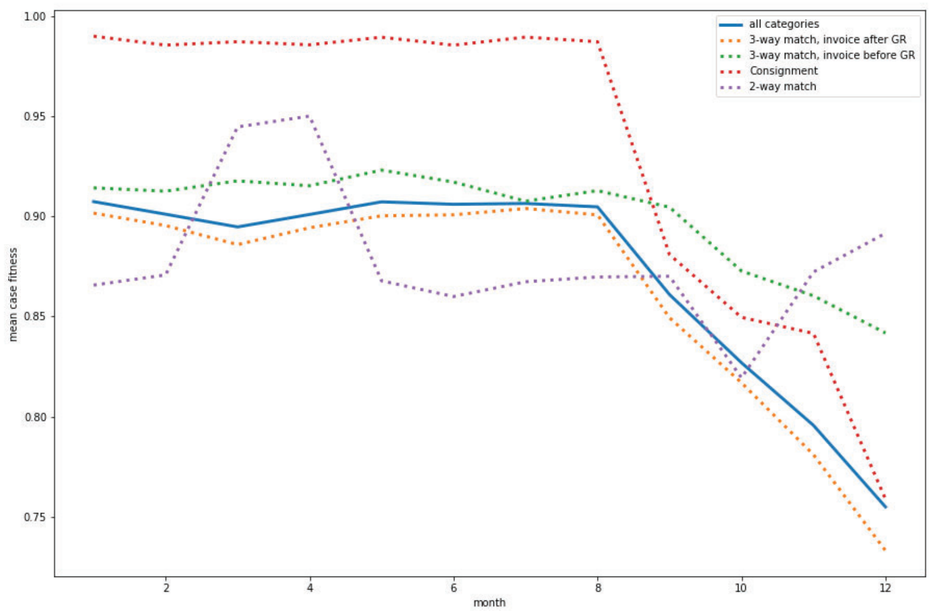

Fig. 8. Mean fitness of events plotted by the months of 2018

Also, the vendors of which the cases have the worst fitness were investigated. To do so, the mean fitness per vendor was calculated. The worst percent of vendors in terms of fitness are shown in Table 7. These vendors should be looked into to find why the fitness is so low. Special attention needs to be given to the vendor with Vendor ID 0582 since this vendor has over 1000 cases and is in the lowest percentile of fitness.

	Fitness	Throughput time	Cumulative net worth
Fitness	1	-0.624	-0.113
Throughput time	-0.624	1	0.339
Cumulative net worth	-0.113	0.339	1

Table 6. Correlations of Fitness, Throughput time and Cumulative net worth

Vendor ID	fitness	Vendor ID	fitness
0039	0.083	1592	0.0833
0046	0.083	1593	0.0833
0055	0.147	1594	0.1470
0065	0.147	1595	0.1470
0090	0.135	1596	0.1351
0582	0.067	1598	0.0674
1254	0.147	1599	0.1470
1572	0.147	1600	0.1470
1576	0.090	1604	0.0909
1583	0.147	1608	0.1470
1586	0.1470	1613	0.1470

Table 7. Users that stand out relative to the mean case fitness

5.2 Case duration analysis

In order to see what variables can delay a case the mean duration per variable are compared and outliers are identified. The duration is used instead of the throughput since duration it is the inverse of the throughput and easier to understand. Outliers are defined as cases that are more than two standard deviations higher than the mean case duration over the variable. The first interesting finding is that there are a lot of vendors with a high mean case duration. These vendors might be worth looking into to find an explanation for the long mean case duration. The vendors in Table 9 have a mean case duration of more than two standard deviations over the mean. The event users that stand out are shown in Table 8.

The previously mentioned users, vendors and sub spend area text are useful to investigate further. The process could be improved by reducing the mean case duration for each of the groups mentioned in Table 8 and Table 9.

In Figure 8 it is seen that the mean case duration for cases containing subcontracting as item type is higher than for the other item types. Further investigation into why this is higher might be useful to improve the process.

User ID	Mean case duration
u153	153.316667
u203	134.000000
u252	146.118881
u279	141.392857
u388	134.124031

Table 8. Users that stand out relative to the mean case duration

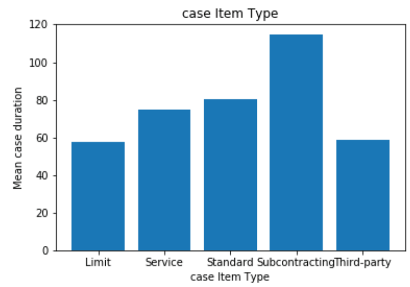

Fig. 9. Mean case duration per item type

Vendor ID	Mean case duration	Vendor ID	Mean case duration	Vendor ID	Mean case duration
0402	344.0	0780	214.684210526	1192	177.666666667
1914	328.0	0824	213.0	1430	176.0
1324	303.0	1024	211.0	1365	176.0
1318	297.0	1522	211.0	1542	175.0
1703	266.0	0532	201.0	0714	175.0
0335	263.0	1101	200.444444444	1386	174.0
1404	263.0	1750	197.0	1107	173.0
1413	260.0	1263	196.5	0025	171.0
0208	254.0	1534	196.0	1543	171.0
1175	247.0	1359	192.0	1472	170.333333333
1261	233.0	1162	192.0	1089	168.0
1194	227.0	1090	191.09375	1346	168.0
0002	224.0	1457	189.0	1523	167.0
0898	223.0	1382	187.310344828	1464	166.5
1428	218.5	0318	183.5	0555	166.458874459
0507	217.0	1489	183.0	1314	165.666666667
1085	216.045454545	1501	179.0	0871	164.5
0789	216.0	1023	178.222222222	1258	163.0

Table 9. Vendors that stand out relative to the mean case duration.

5.3 3-way Matching Analysis

Framework The 4 categories of matching as specified in section 1.2 require that each Purchase Document with line items follows a matching standard, with 3-way matching being dominant in the dataset. Thus, in this section, we provide further analysis on the verification and analysis of deviations in matching in the two 3-way matching categories. In both of these categories, the value goods receipts messages should be matched against the value of the invoice receipt messages and the value during creation of the item.

As both the number of Goods Receipt and Invoice receipt messages vary greatly among cases that themselves vary greatly in terms of duration, complexity and type, we use a framework to distinguish between 'matching' and 'non-matching' cases. To avoid taking into account duplicate entries for Goods Receipt or Invoice Receipt messages, we drop activities if they are duplicates regarding the following variables:

- Activity
- Case ID
- Complete Timestamp
- event Cumulative net worth (EUR)
- event User

Here, 'event Cumulative net worth (EUR)' represents the aforementioned 'value' that should be matched among (1) Goods Receipt messages, (2) Invoice Receipt messages and the creation of the line item, denoted as a (3) 'Create Purchase Order Item' message. These duplicate entries are do not occur frequently ($< 1\%$), but since the matching is dependent on an exact match of the cumulative values of the aforementioned three messages, a single entry that is duplicate within a case would result in a 'no-match'.

After removing duplicate entries, we calculated the values of the three messages (Goods Receipt, Invoice Receipt, Create Purchase Order) as follows, with V_e denoting the event Cumulative net worth (EUR) of an event with name e.

- Item Creation:
$$V_0 = V_{\text{Create Purchase Order Item}}$$
- Goods Receipt:
$$V_1 = \sum V_{\text{Record Goods Receipt}} - \sum V_{\text{Cancel Goods Receipt}}$$
- Invoice Receipt:
$$V_2 = \sum V_{\text{Record Invoice Receipt}} + \sum V_{\text{Record Subsequent Invoice Receipt}} + \sum V_{\text{Record Service Entry}}$$

We specify the following outcomes:

- Complete 3-way matching : *FULL_MATCH* ($V_0 = V_1 = V_2$)
- Complete 3-way mis-match *NO_MATCH*
- One or more messages missing, and as each of the three messages can be present or absent, there are $2^3 = 6$ possible missing outcomes: (e.g. *MISSING_GOODS_RECEIPT_INVOICE_RECEIPT*
- A partial match (e.g. *GOODS_RECEIPT_INVOICE_RECEIPT*).

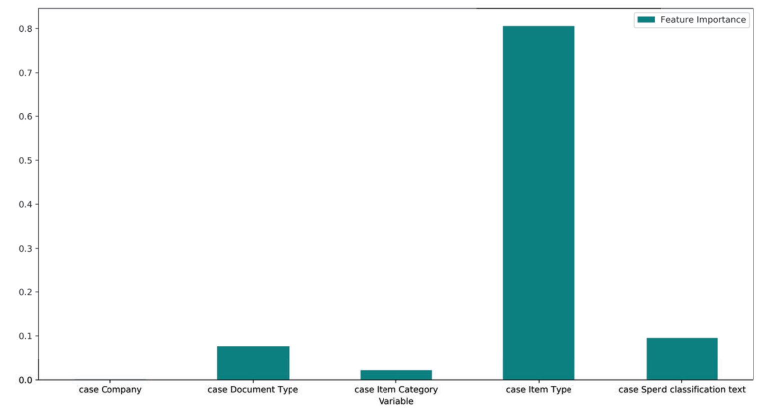

Fig. 10. Feature importance of variables [0,1] predicting matching outcomes using a Decision Tree Classifier

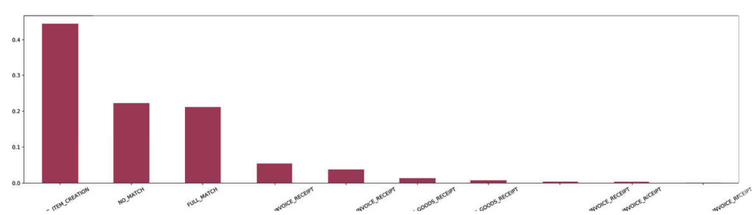

Fig. 11. Matching Results for item Type Service

Performance When calculating the matching on the entire dataset, we find that 87.26% of all cases are fully matching, requiring further analysis. To analyze what influences matching deviation, we used a Decision Tree Classifier using a train/test split of 80%/20% with the Match result as dependent variable, resulting in a test-accuracy of 88.07%. As seen in Figure 10, the *case Item Type* is the most important variable in predicting the matching result, and so we filtered on these cases to investigate further.

When strictly looking at a full matches, we find that the service type cases are considered outliers using the specified framework, this is seen in Figure 12. More than two-thirds of these mis-matching items are the result of these cases not having a *Goods Receipt* message, and many lack a *Create Invoice Receipt* message in their case or parent Purchasing Document, even though they occur completely in 2018.

Since these service-type items do not occur frequently (2.48%), we are interested in looking at the matching results for the rest of the dataset. Using the same train/test split of 80%/20%, the result is 95.16% test score, and now the

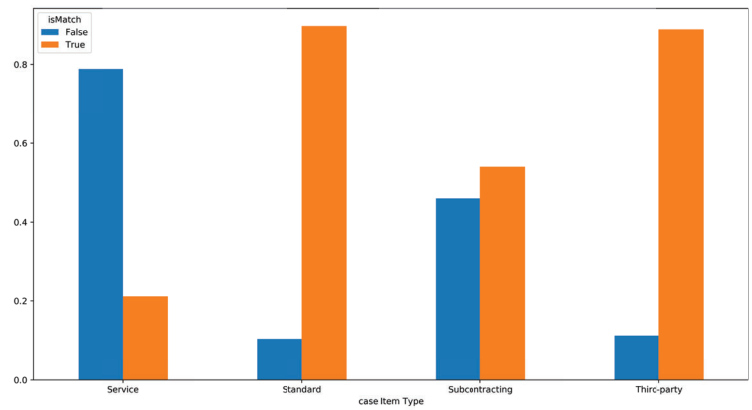

Fig. 12. Fraction of items completely matching (orange) for each case item type

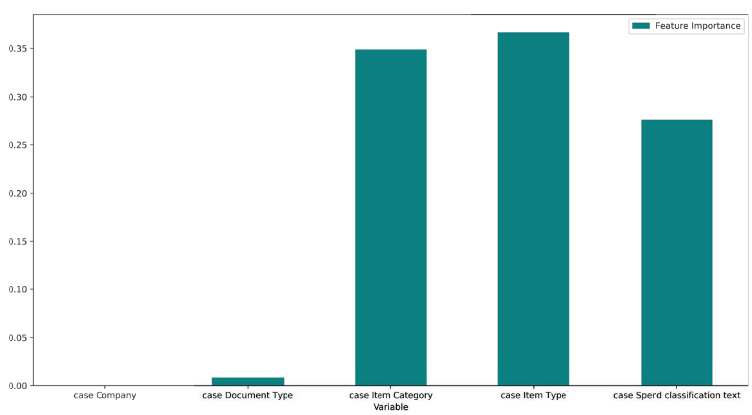

Fig. 13. Fraction of items completely matching (orange) for each case item type, without service items

case Item Category and Spend classification text together predict 99.17% of the matching outcome, as seen in 13. Without the service items, 90.94% of cases are fully-matching, and the following differences are observable for the non-matching cases:

– case Item Type: the type Subcontracting cases had less full-matches (53.69%) compared to Standard (89.36%) and Third-Party (85.04%) .

– case Item Category: The matching type Invoice-after-GR cases had less full-matches (80.93%) compared to Invoice-before-GR (88.77%).
– case Spend classification text: The cases with a case Spend classification text of 'Unknown' matches much less (62.20%) compared to the NPR, OTHER and PR cases. This 'Unknown' value is exactly what we used to fill the missing case Spend classification text values.

5.4 Conclusion

It is found that the case fitness changes from august onward. This might be due to a change in the process. It is also found that the fitness is negatively correlated with the throughput time. It is found that certain vendors tend to have a low case fitness. These vendors should be investigated further to see why this low fitness occurs. One vendor stands out due to the amount of cases this vendor handles. This vendor is the most important to look into and find an explanation for the low fitness.

It is found that there are a lot of vendors that have a high mean case duration compared to the mean. Looking into these vendors could be useful whilst trying to reduce the case duration. Users who have a high mean case duration are also identified, looking into these users is also useful in trying to reduce the mean case duration. Furthermore it is found that the item type subcontracting results in a higher mean case duration.

References

1. https://icpmconference.org/icpm-2019/contests-challenges/bpi-challenge-2019/
2. Buijs, J. C. A. M. (2014). Flexible evolutionary algorithms for mining structured process models Eindhoven: Technische Universiteit Eindhoven DOI: 10.6100/IR780920
https://research.tue.nl/en/publications/flexible-evolutionary-algorithms-for-mining-structured-process-mo
3. Mannhardt, Felix & de Leoni, Massimiliano & A. Reijers, Hajo. (2015). The Multi-perspective Process Explorer. CEUR Workshop Proceedings. 1418.
https://www.researchgate.net/publication/282283719_The_Multi-perspective_Process_Explorer