Test On Real life data

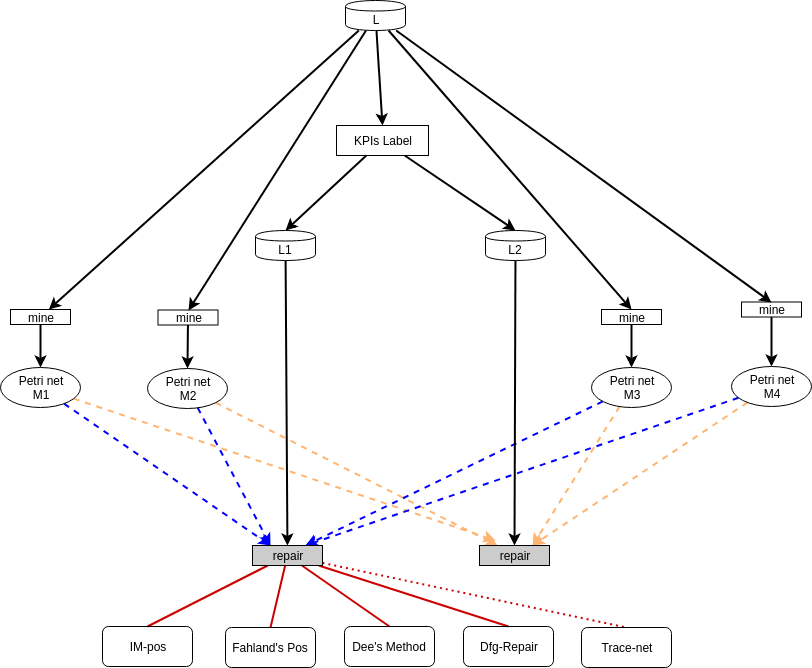
The goal to test on real life data::

– to prove the use of our method with comparison with Dee’s ones, Fahhland’s method

<1> model simplicity and precision

<2> model KPI improvement, how to validate this?? By checking the positive output percentage, it is a measurement.

Test data description::

The first data set is from BPI15\_1.xes, <https://data.4tu.nl/repository/uuid:a0addfda-2044-4541-a450-fdcc9fe16d17>

The information about it :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Data ID | File name | process | cases | events | Event class |
| 1 | BPIC15\_1.xes.xml | Real life data from BPI15 | 1199 | 52217 | 398 |
| 2.1 | BPI15\_1\_50\_filter.xes | Heuristic filtering by setting 50% for all from data 1 | 592 | 14307 | 28 |
| 2.2 | BPI15\_1\_50\_filter\_tt.xes | Add trace attribute of throughput time on the traces | 592 | 14307 | 28 |
| 3.1 | BPI15\_1\_50\_filter\_sum\_neg.xes | Filter data on trace attribute of SUMleges, if it is over the 70% sum, then it is assigned with negative. | 166 | 4149 | 28 |
| 3.2 | BPI15\_1\_50\_filter\_sum\_pos.xes | Filter data on trace attribute of SUMleges, if it is below the 70% sum, then it is assigned with positive. | 426 | 10158 | 28 |
| 3.3 | BPI15\_1\_50\_filter\_sum\_labels.xes | Union 3.1 and 3.2 to get an event log with sum labels for our methods | 592 | 14307 | 28 |
| 4.1 | BPI15\_1\_50\_filter\_tt\_neg.xes | Filter data on trace attribute of throughput time, if it is over the 70%, then it is assigned with negative. | 179 | 4360 | 28 |
| 4.2 | BPI15\_1\_50\_filter\_tt\_pos.xes | Filter data on trace attribute of throughput time, if it is over the 70%, then it is assigned with negative. | 413 | 9947 | 28 |
| 4.3 | BPI15\_1\_50\_filter\_tt\_labels.xes | Union 3.1 and 3.2 to get an event log with throughput time labels for our methods | 592 | 14307 | 28 |
| 5 | BPI15\_1\_50\_filter\_18\_classes.xes | Filter 2.1 again with heuristic filter , setting: 80-80-70, | 567 | 10050 | 18 |

Besides of the event logs, we need multiple models as our reference models. We can generate them in the following ways:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Data file | Setting | Model file | Model figures | Description |
| M1 | BPI15\_1\_50\_filter\_18\_classes.xes | Inductive Mine:  IM-infrequent : 0.2  concept: name | BPI\_1\_M1\_18\_classes.pnml | BPI\_1\_M1\_figure.pdf | Have several changes but not so much |
| M2 | BPI15\_1\_50\_filter\_18\_classes.xes | Inductive Mine:  IM-infrequent : 0.5  concept: name | BPI\_1\_M2\_18\_classes.pnml | BPI\_1\_M2\_figure.pdf | It is already like a sequential model  So not go deeper into it |
| M3 | BPI15\_1\_50\_filter.xes | Inductive Mine:  IM-infrequent : 0.2  concept: name | BPI\_1\_M3\_28\_classes.pnml | BPI\_1\_M3\_figure.pdf | Almost in linear but with silent transitions and parallel |
| M3 | BPI15\_1\_50\_filter.xes | Inductive Mine:  IM-infrequent : 0.5  concept: name | BPI\_1\_M4\_28\_classes.pnml | BPI\_1\_M4\_figure.pdf | n linear with few silent transitions and parallel |

After obtaining logs and models, we conduct the test with several techniques.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Log | Model | Techniques | Result |
| T1.1 | 3.2 | M1 | IM\_pos: only on positive, test based on the train data D3.3 | TP: 137 FP: 48 TN:118 FN:289  Recall: 0.3215962441314554  Precision: 0.7405405405405405  Accuracy: 0.43074324324324326  F-score: 0.44844517184942717 |
|  | 3.2 | M1 | Fahland’s method: only on positive, test based on train data |  |
|  | 3.3 | M1 | Dees method: on whole data |  |
|  | 3.3 | M1 | Dfg method: on whole data |  |
|  | 3.2 | M1 | Trace nets |  |
| T1.2 | 3.2 | M2 |  |  |
|  | ... | ... |  |  |
| T1.3 | 3.2 | M3 |  |  |
|  | ... | ... |  |  |
| T1.4 | 3.2 | M4 |  |  |
|  | ... | ... |  |  |
| T2.1 | 4.2 | M1 |  |  |
|  | ... | ... |  |  |
| T2.1 | 4.2 | M2 |  |  |
|  | ... | ... |  |  |
| T2.3 | 4.2 | M3 |  |  |
|  | ... | ... |  |  |
| T2.4 | 4.2 | M4 |  |  |
|  | ... | ... |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

The total number of tests for one event logs BPI\_1.xes is 2\*4\*5=40; we can repeat the method on the other logs BPI\_2.xes, BPI\_3.xes,BPI\_4.xes,BPI\_5.xes. So total number is 40\*5 =200… If we can conduct this tests.

But first make sure that this methods work…

Because the repair elapsed time is around 60 mins, so we change to another methods with KPI information and do the experiments.

So reduce the classes and generate new data sets :

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Data ID | File name | process | cases | events | Event class | Min | Mean | Max |
| D1 | BPIC15\_1.xes.xml | Real life data from BPI15 | 1199 | 52217 | 398 | 2 | 44 | 101 |
| D2 | BPI15\_1\_40\_filter.xes | Heuristic filter with 40 and get event logs;  [with filter out low-frequency traces, we still have the same event classes, so we will keep it here] | 495 | 9565 | 20 | 14 | 19 | 25 |
| D3 | BPIC15\_1\_filter\_40\_filter\_60.xes | Apply heuristic filter again on the data D2 with 60, get a model less than it | 378 | 4566 | 12 | 11 | 12 | 17 |
| D4.1 | BPI15\_1\_40\_filter\_sum\_neg.xes | Use 0.7 as sum threshold on SumLedges, above is negative | 146 | 2811 | 20 | 15 | 19 | 25 |
| D4.2 | BPI15\_1\_40\_filter\_sum\_pos.xes | Use 0.7 as sum threshold on SumLedges, below is positive | 349 | 6744 | 20 | 14 | 19 | 22 |
| D4.3 | BPI15\_1\_40\_filter\_sum\_labels.xes | Union of D4.1 and D4.2 | 495 | 9565 | 20 | 14 | 19 | 25 |
| D5.1 | BPI15\_1\_40\_filter\_tt\_neg.xes | Use 0.7 as threshold on throughput time, above is negative | 149 | 2846 | 20 | 14 | 19 | 25 |
| D5.2 | BPI15\_1\_40\_filter\_tt\_pos.xes | Use 0.7 as sum threshold on throughput time, below is positive | 346 | 6719 | 20 | 15 | 19 | 25 |
| D5.3 | BPI15\_1\_40\_filter\_tt\_labels.xes | Union of D5.1 and D5.2 | 495 | 9565 | 20 | 14 | 19 | 25 |

Besides of the event logs, we need multiple models as our reference models. We can generate them in the following ways:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Data file | Setting | Model file | Model figures | Description |
| M1 | BPI15\_1\_40\_filter.xes  convert to data Petri net, | Inductive Mine:  IM-infrequent : 0.2  concept: name | BPI\_1\_40\_M1\_IM0\_classes.pnml | BPI\_1\_40\_M1\_figure.pdf | Contain 20 classes, xor one, one parallel, silent transitions;  CM:  TP: 112  FP: 40  TN: 106  FN: 237  Recall:  0.3209169054441261 Precision:  0.7368421052631579 Accuracy:  0.4404040404040404 F-score:  0.4471057884231537 |
| M2 | BPI15\_1\_40\_filter.xes | Inductive Mine:  IM-infrequent : 0.5  concept: name | BPI\_1\_40\_M2\_IM50\_classes.pnml | BPI\_1\_40\_M2\_figure.pdf | 20 event classes, compared to M1, less silent transitions |
| M3 | BPIC15\_1\_filter\_40\_filter\_60.xes | Inductive Mine:  IM-infrequent : 0.2  concept: name | BPI\_1\_40\_M3\_IM20\_classes.pnml | BPI\_1\_40\_M3\_figure.pdf | With one xor, one parallel, with 12 classes |
| M3 | BPIC15\_1\_filter\_40\_filter\_60.xes | Inductive Mine:  IM-infrequent : 0.5  concept: name | BPI\_1\_40\_M3\_IM50\_classes.pnml | BPI\_1\_40\_M4\_figure.pdf | Sequential model, 12 classes |

After obtaining logs and models, we conduct the test with several techniques.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Log | Model | Techniques | Result |
| T1.1.1 | 4.2 |  | IM\_pos: only on positive, test based on the train data D4.3 | TP: 137 FP: 48 TN:118 FN:289  Recall: 0.3215962441314554  Precision: 0.7405405405405405  Accuracy: 0.43074324324324326  F-score: 0.44844517184942717 |
| T1.1.2 | 4.2 | M1 | Fahland’s method: only on positive, test based on train data with all labels | We get models with a lot of complexity to fulfill the needs in the data. But TP=FP=0, don’t know why.. |
|  | 4.3 | M1 | Dees method: on whole data, but it demands a discrete KPI outcome, so I need to add one KPI outcomes according to it.. | Get exceptions about the null pointer..  doubts at the Petri net importer. |
|  | 4.3 | M1 | Dfg method: on whole data:: |  |
|  | 4.2 | M1 | Trace nets |  |
| T1.2 | 3.2 | M2 |  |  |
|  | ... | ... |  |  |
| T1.3 | 3.2 | M3 |  |  |
|  | ... | ... |  |  |
| T1.4 | 3.2 | M4 |  |  |
|  | ... | ... |  |  |
| T2.1 | 4.2 | M1 |  |  |
|  | ... | ... |  |  |
| T2.1 | 4.2 | M2 |  |  |
|  | ... | ... |  |  |
| T2.3 | 4.2 | M3 |  |  |
|  | ... | ... |  |  |
| T2.4 | 4.2 | M4 |  |  |
|  | ... | ... |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

The total number of tests for one event logs BPI\_1.xes is 2\*4\*5=40; we can repeat the method on the other logs BPI\_2.xes, BPI\_3.xes,BPI\_4.xes,BPI\_5.xes. So total number is 40\*5 =200… If we can conduct this tests.

Data set from Dee’s

Features of this data set: with KPI, but unfortunately, no context for me to write the information about it. So I can not really use it directly. No public available..

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Data ID | File name | process | cases | events | Event class |
| 1 | Train-Data-In-XES.xes | Real life data from Dee’s | 1199 | 52217 | 398 |
| 2.1 | BPI15\_1\_50\_filter.xes | Heuristic filtering by setting 50% for all from data 1 | 592 | 14307 | 28 |
| 2.2 | BPI15\_1\_50\_filter\_tt.xes | Add trace attribute of throughput time on the traces | 592 | 14307 | 28 |
| 3.1 | BPI15\_1\_50\_filter\_sum\_neg.xes | Filter data on trace attribute of SUMleges, if it is over the 70% sum, then it is assigned with negative. | 166 | 4149 | 28 |
| 3.2 | BPI15\_1\_50\_filter\_sum\_pos.xes | Filter data on trace attribute of SUMleges, if it is below the 70% sum, then it is assigned with positive. | 426 | 10158 | 28 |
| 3.3 | BPI15\_1\_50\_filter\_sum\_labels.xes | Union 3.1 and 3.2 to get an event log with sum labels for our methods | 592 | 14307 | 28 |

Data set from BPI::

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Data set | description | feasibility |
| 1 | BPI12 | <https://data.4tu.nl/repository/uuid:3926db30-f712-4394-aebc-75976070e91f>  Loan system, main goal check the resource and workflow distribution  event classes 21  3.19MB | Not fit! Because no significant KPI, also throughputtime is in different cases, like decline instantly is not what we wishes!!  Possibility: use the throughput time??  use the decline time as neg? |
| 2 | BPI17 | <https://data.4tu.nl/repository/uuid:5f3067df-f10b-45da-b98b-86ae4c7a310b>  Similar to BPI12 but with more data classes 66  28.3MB |  |
| 3 | BPI18 | <https://data.4tu.nl/repository/uuid:3301445f-95e8-4ff0-98a4-901f1f204972>  Farmer payment project,  XES 1.9GB  41 classes, 43809 cases, 2514266 events, min 24 -2973 | Big data,  but with one potential KPI of throughput time  Not fit!! |

Or in a simple way, to reduce the scale of this data!!

When we have the result, it doesn’t have significant changes with the values except all the zero values after this.

Even after we do experiments on the real life data, due to the filtering stuff, we can have good model, but the weight changes trend, not so clear.. So we need to see more, and find the pattern…

After this, we should find some models from the result and shows them?? Do we ?? What do you want to prove?? One is to show the ability to create good models, the other is about the tendency.

For the demo example, find a way this night.. I suggest, in which way that the model can accept deviations but let the model running…

The drawback exists in directly-follows relation, but how to balance them?? Some is existing ones, and the new generated ones..

In one compressed example, show the result on them… Transitions system not handle parallelsim well… But should we keep them both in a regular way?? Yes, actually, we have done this, but it is sensitive about the changes?? Can we say this?

Because of the memory, so we can not apply to all data just in one round!!

Repeat it again with small sets..

After testing on M1-D53, we have ext-plot, neg-plot and pos-plot with overlapped line of recall:

ext-plot: accuracy with recall

neg-plot: accuracy with recall

pos-plot: accuracy with recall

Thought sth different, but ok, the same