Process Oriented Data Science

UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

Campus d'Excel·lència Internacional

Josep Carmona
Computer Science Department

Outline

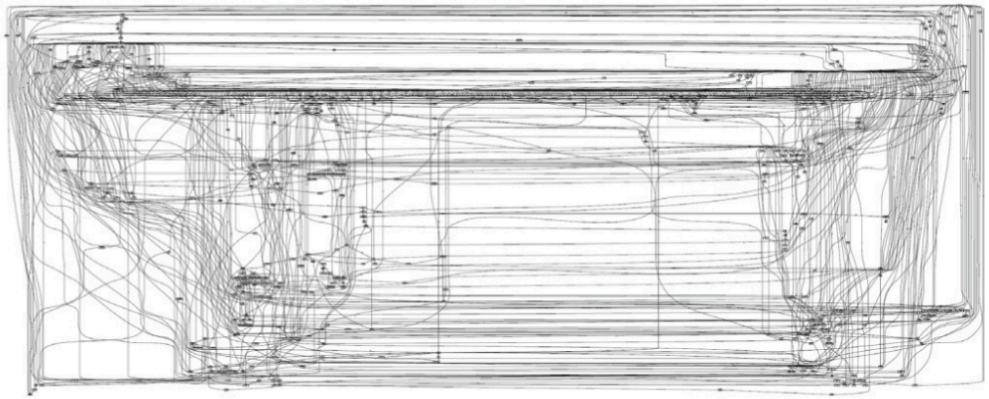
- M1: Process Mining Overview, Positioning & Preliminaries (Event data & Process Models)
- **M2: Process Discovery**
- M3: Conformance Checking
- M4: Process Enhancement

Disclaimer

- Most of the material of this course is taken from my colleagues:
  - RWTH Aachen (Prof. Wil van der Aalst)
  - **Humboldt University zu Berlin (Prof. Matthias Weidlich)**
  - Technishe Universiteit Eindhoven (Prof. Boudewijn van Dongen)
  - University of Tartu (Prof. Marlon Dumas)
  - University of Melbourne (Prof. Marcello La Rosa)
  - Technical University of Denmark (Prof. Andrea Burattin)
- Hence, this material is only provided for your learning, please do not share nor publish

Issues with the α-family

- Properties of the α(+/++) – algorithms
  - Each (!) direct successorship is processed
  - All tasks and successorship relations are considered to be equally important
- Consequences:
  - α – algorithms are not robust against noise
  - α – algorithms provide no means for abstraction

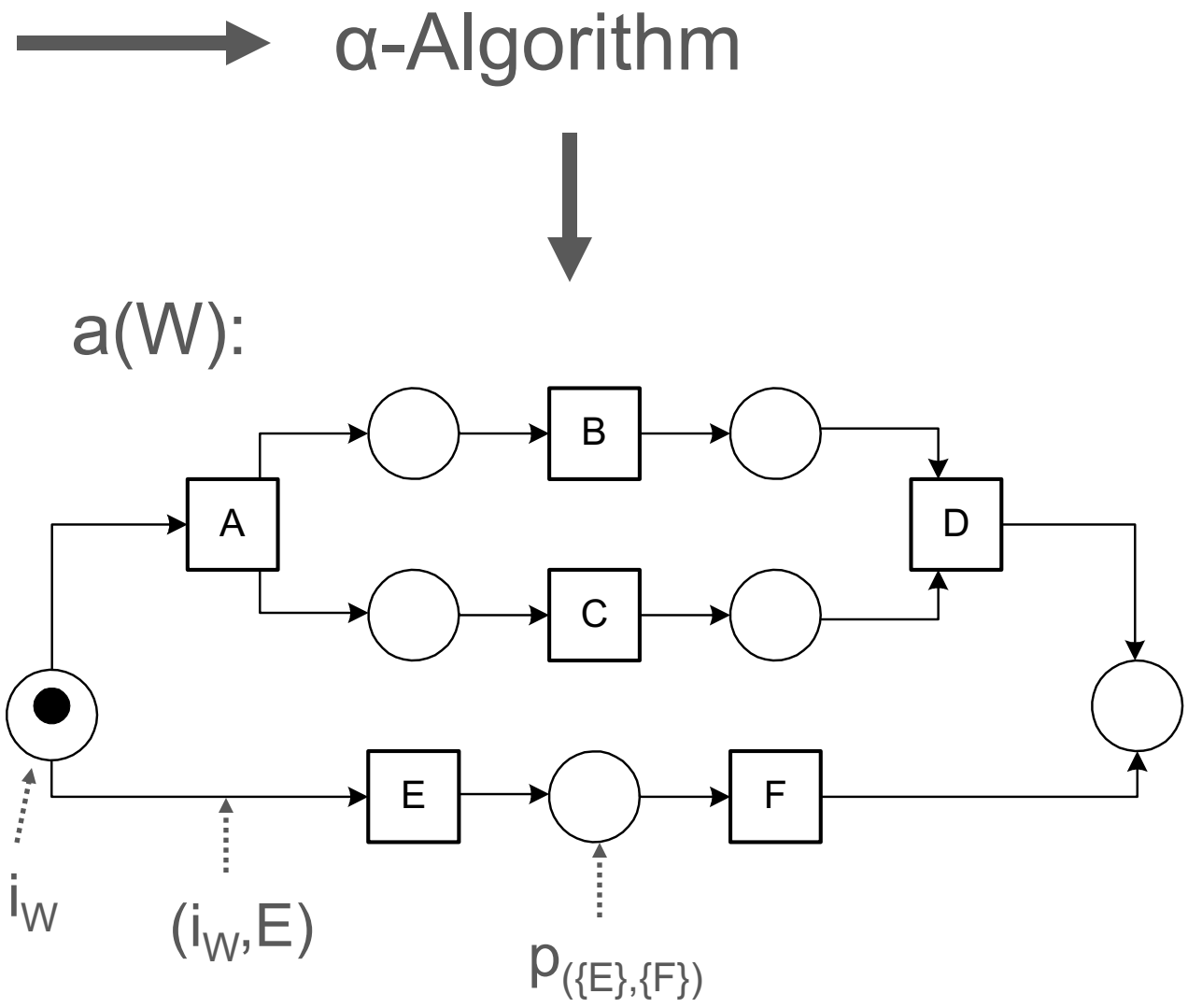


Quality of Event Logs

- "Clean event logs"
  - Event relates to one activity and one process instance
  - All traces are valid execution sequences of the process
- Not realistic in practice, there is "noise"
  - Erroneous logging mechanisms
  - Not all log entries are written, some are lost or inserted in a wrong order
- Think about the example scenarios again….

Impact of Noise

```
Case 1: ABCD
Case 2: ACBD
Case 3: EF
Case L: ABD
Case M: CBD
Case N: CADB
```

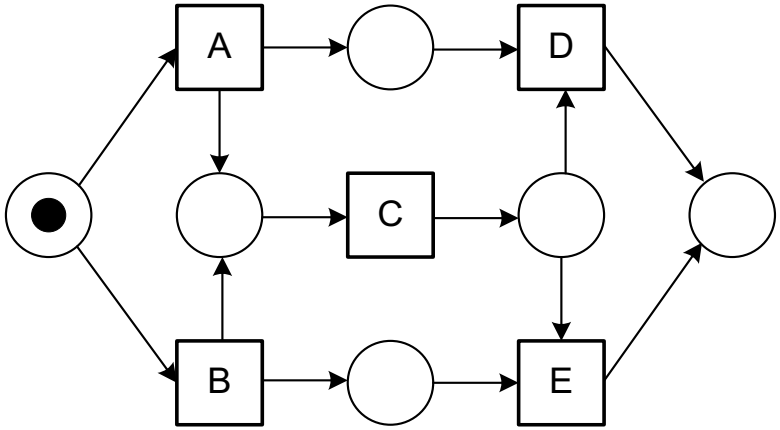


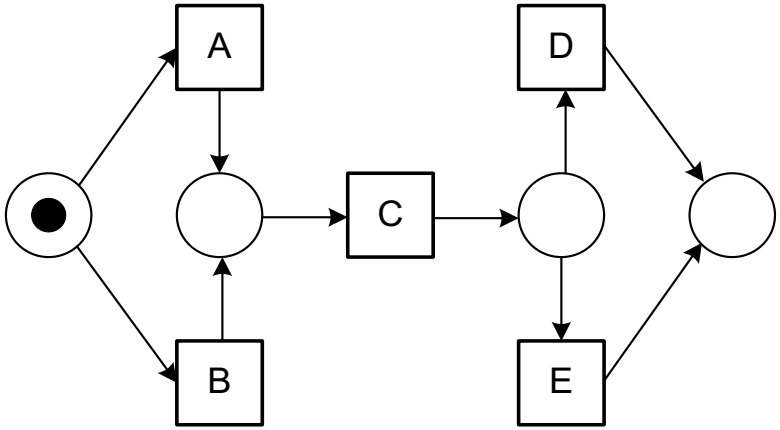
Consequences of Noise

- Massive impact on discovery, conformance, and enhancement techniques – we will get back to this
- Already an issue in the construction of event logs
- Major issue: *what* is noise is close to impossible to characterise without domain knowledge

Noise Example

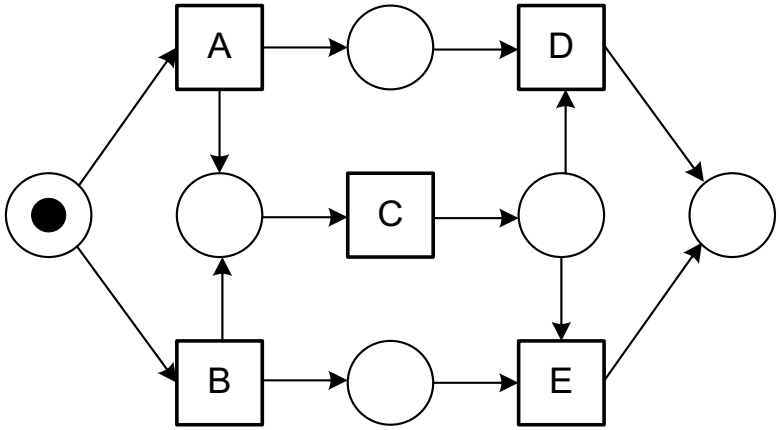
ACD	99
ACE	0
BCE	85
BCD	0

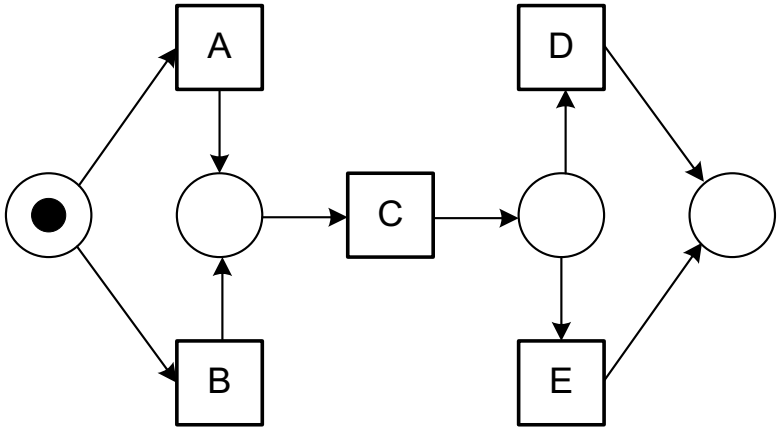




Noise Example

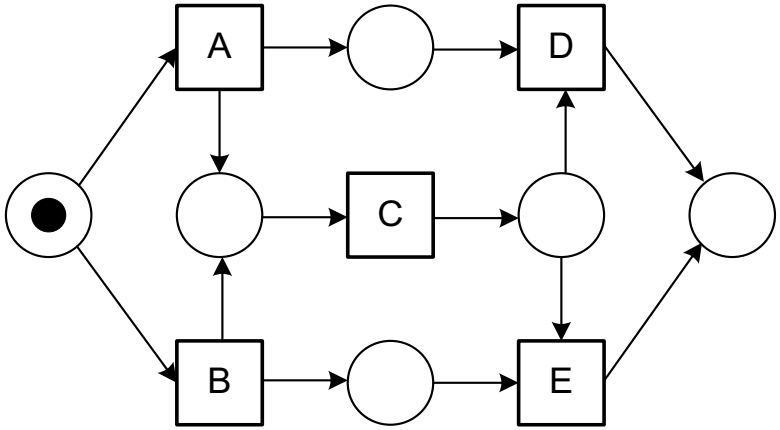
ACD 99
ACE 88
BCE 85
BCD 78

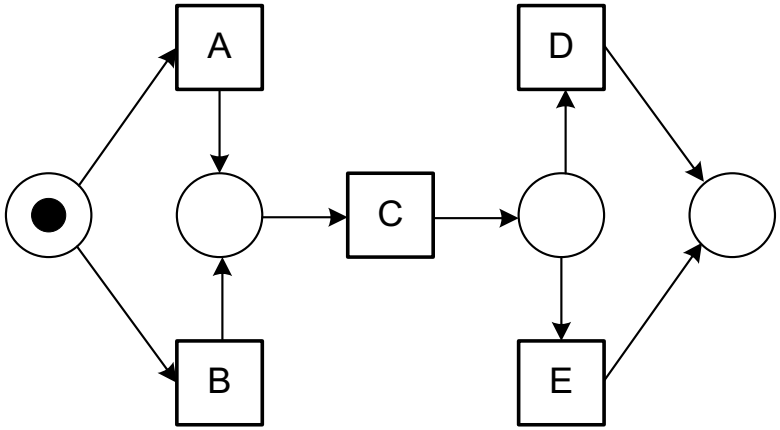




Noise Example

ACD	99
ACE	2
BCE	85
BCD	3





Heuristics & Fuzzy Miners

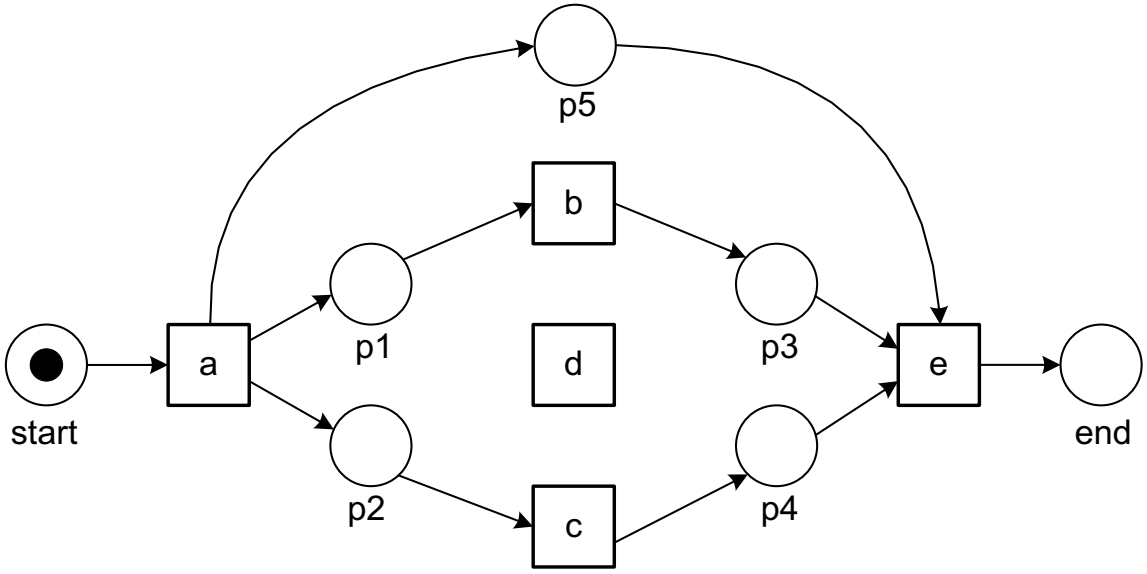
- More practical approaches to process discovery
  - *Heuristic Miner*: exploits occurrence frequencies to estimate flow probabilities
  - *Fuzzy Miner*: introduces measures for significance and correlation to create abstract views of the process model
- Common basis:
  - Ordering relations of the α – algorithm are used as the foundation
  - Relations provide a model to reason about frequencies
- Details:
  - A. J. M. M. Weijters, Wil M. P. van der Aalst: Rediscovering workflow models from event-based data using little thumb. Integrated Computer-Aided Engineering (ICAE) 10(2):151-162 (2003)
  - Christian W. Günther, Wil M. P. van der Aalst: Fuzzy Mining - Adaptive Process Simplification Based on Multi-perspective Metrics. BPM 2007:328-343

Construction of a Net System

- General net structure is directly given by the dependency graph
- Transformations needed for splits and joins, i.e., tasks with multiple incoming or outgoing edges
  - Rely on the relations between succeeding (split) or preceding tasks (join) to determine type of split/join
  - Rely on the frequencies for the split (join) and its succeeding (preceding) tasks
- But: Splits and joins are only considered locally (thus most of the discovered model are not sound and require repair actions)
- Examples: C-Nets, Petri nets, BPMN ...

Example log; problem alpha algorithm

$$L = [\langle a,e\rangle^{5}, \langle a,b,c,e\rangle^{10}, \langle a,c,b,e\rangle^{10}, \langle a,b,e\rangle^{1}, \langle a,c,e\rangle^{1}, \langle a,d,e\rangle^{10}, \langle a,d,d,e\rangle^{2}, \langle a,d,d,d,e\rangle^{1}]$$



$$L = [\langle a,e\rangle^5, \langle a,b,c,e\rangle^{10}, \langle a,c,b,e\rangle^{10}, \langle a,b,e\rangle^1, \langle a,c,e\rangle^1,$$
$$\langle a,d,e\rangle^{10}, \langle a,d,d,e\rangle^2, \langle a,d,d,d,e\rangle^1]$$

$$|a >_L b| = \sum_{\sigma \in L} L(\sigma) \times |\{1 \leq i < |\sigma| \mid \sigma(i) = a \wedge \sigma(i+1) = b\}|$$

$\|>_L\|$	a	b	c	d	e
a	0	11	11	13	5
b	0	0	10	0	11
c	0	10	0	0	11
d	0	0	0	4	13
e	0	0	0	0	0

$$|a >_L b| = \sum_{\sigma \in L} L(\sigma) \times |\{1 \leq i < |\sigma| \mid \sigma(i) = a \wedge \sigma(i+1) = b\}|$$

$|a \Rightarrow_L b|$ is the value of the dependency relation between a and b:

$$|a \Rightarrow_L b| = \begin{cases} \dfrac{|a >_L b| - |b >_L a|}{|a >_L b| + |b >_L a| + 1} & \text{if } a \neq b \\ \dfrac{|a >_L a|}{|a >_L a| + 1} & \text{if } a = b \end{cases}$$

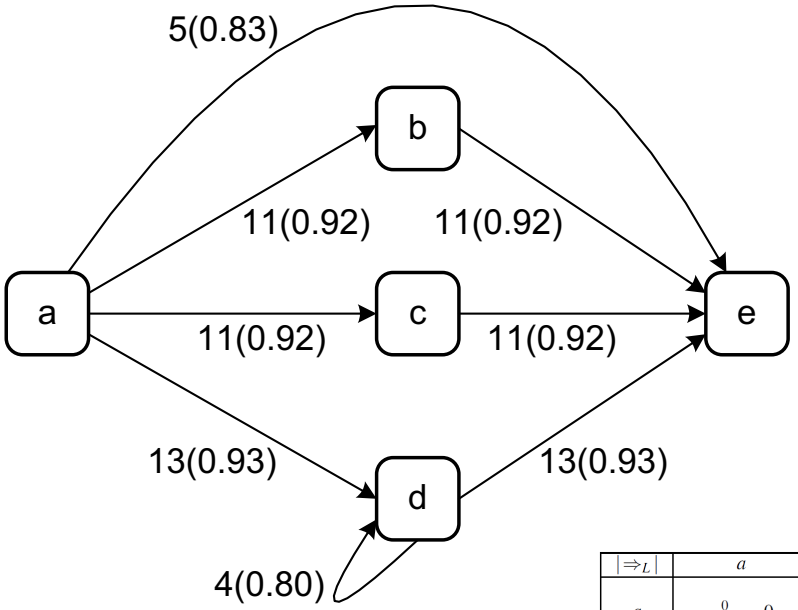
$\lvert \Rightarrow_L \rvert$	a	b	c	d	e
a	$\frac{0}{0+1} = 0$	$\frac{11-0}{11+0+1} = 0.92$	$\frac{11-0}{11+0+1} = 0.92$	$\frac{13-0}{13+0+1} = 0.93$	$\frac{5-0}{5+0+1} = 0.83$
b	$\frac{0-11}{0+11+1} = -0.92$	$\frac{0}{0+1} = 0$	$\frac{10-10}{10+10+1} = 0$	$\frac{0-0}{0+0+1} = 0$	$\frac{11-0}{11+0+1} = 0.92$
c	$\frac{0-11}{0+11+1} = -0.92$	$\frac{10-10}{10+10+1} = 0$	$\frac{0}{0+1} = 0$	$\frac{0-0}{0+0+1} = 0$	$\frac{11-0}{11+0+1} = 0.92$
d	$\frac{0-13}{0+13+1} = -0.93$	$\frac{0-0}{0+0+1} = 0$	$\frac{0-0}{0+0+1} = 0$	$\frac{4}{4+1} = 0.80$	$\frac{13-0}{13+0+1} = 0.93$
e	$\frac{0-5}{0+5+1} = -0.83$	$\frac{0-11}{0+11+1} = -0.92$	$\frac{0-11}{0+11+1} = -0.92$	$\frac{0-13}{0+13+1} = -0.93$	$\frac{0}{0+1} = 0$

$\lvert a \Rightarrow_L b \rvert$ is the value of the dependency relation between a and b:

$$\lvert a \Rightarrow_L b \rvert = \begin{cases} \frac{\lvert a >_L b \rvert - \lvert b >_L a \rvert}{\lvert a >_L b \rvert + \lvert b >_L a \rvert + 1} & \text{if } a \neq b \\ \frac{\lvert a >_L a \rvert}{\lvert a >_L a \rvert + 1} & \text{if } a = b \end{cases}$$

$\lvert >_L \rvert$	a	b	c	d	e
a	0	11	11	13	5
b	0	0	10	0	11
c	0	10	0	0	11
d	0	0	0	4	13
e	0	0	0	0	0

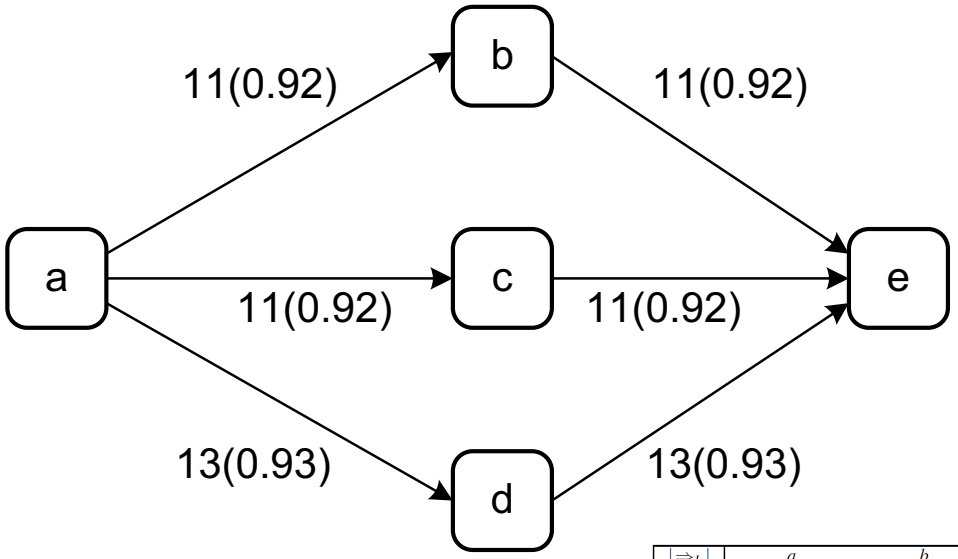
Lower threshold (2 direct succ. and a dep of >= 0.7)



$\lvert >_L \rvert$	a	b	c	d	e
a	0	11	11	13	5
b	0	0	10	0	11
c	0	10	0	0	11
d	0	0	0	4	13
e	0	0	0	0	0

$\lvert \Rightarrow_L \rvert$	a	b	c	d	e
a	$\frac{0}{0+1}=0$	$\frac{11-0}{11+0+1}=0.92$	$\frac{11-0}{11+0+1}=0.92$	$\frac{13-0}{13+0+1}=0.93$	$\frac{5-0}{5+0+1}=0.83$
b	$\frac{0-11}{0+11+1}=-0.92$	$\frac{0}{0+1}=0$	$\frac{10-10}{10+10+1}=0$	$\frac{0-0}{0+0+1}=0$	$\frac{11-0}{11+0+1}=0.92$
c	$\frac{0-11}{0+11+1}=-0.92$	$\frac{10-10}{10+10+1}=0$	$\frac{0}{0+1}=0$	$\frac{0-0}{0+0+1}=0$	$\frac{11-0}{11+0+1}=0.92$
d	$\frac{0-13}{0+13+1}=-0.93$	$\frac{0-0}{0+0+1}=0$	$\frac{0-0}{0+0+1}=0$	$\frac{4}{4+1}=0.80$	$\frac{13-0}{13+0+1}=0.93$
e	$\frac{0-5}{0+5+1}=-0.83$	$\frac{0-11}{0+11+1}=-0.92$	$\frac{0-11}{0+11+1}=-0.92$	$\frac{0-13}{0+13+1}=-0.93$	$\frac{0}{0+1}=0$

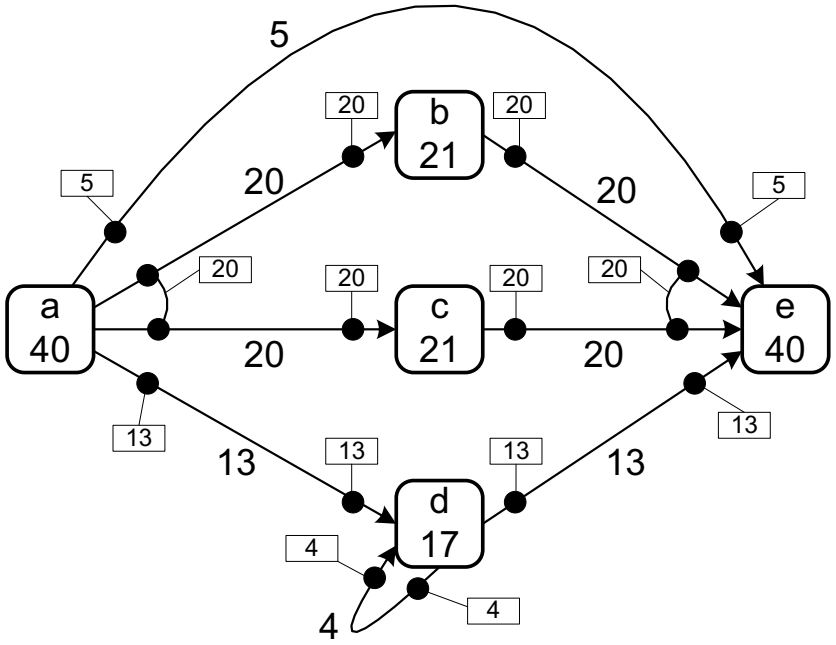
Higher threshold (5 direct succ. and a dep of >= 0.9)



$\lvert >_L \rvert$	a	b	c	d	e
a	0	11	11	13	5
b	0	0	10	0	11
c	0	10	0	0	11
d	0	0	0	4	13
e	0	0	0	0	0

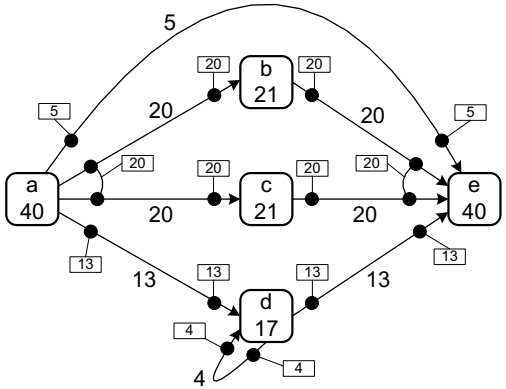
$\lvert \Rightarrow_L \rvert$	a	b	c	d	e
a	$\frac{0}{0+1}=0$	$\frac{11-0}{11+0+1}=0.92$	$\frac{11-0}{11+0+1}=0.92$	$\frac{13-0}{13+0+1}=0.93$	$\frac{5-0}{5+0+1}=0.83$
b	$\frac{0-11}{0+11+1}=-0.92$	$\frac{0}{0+1}=0$	$\frac{10-10}{10+10+1}=0$	$\frac{0-0}{0+0+1}=0$	$\frac{11-0}{11+0+1}=0.92$
c	$\frac{0-11}{0+11+1}=-0.92$	$\frac{10-10}{10+10+1}=0$	$\frac{0}{0+1}=0$	$\frac{0-0}{0+0+1}=0$	$\frac{11-0}{11+0+1}=0.92$
d	$\frac{0-13}{0+13+1}=-0.93$	$\frac{0-0}{0+0+1}=0$	$\frac{0-0}{0+0+1}=0$	$\frac{4}{4+1}=0.80$	$\frac{13-0}{13+0+1}=0.93$
e	$\frac{0-5}{0+5+1}=-0.83$	$\frac{0-11}{0+11+1}=-0.92$	$\frac{0-11}{0+11+1}=-0.92$	$\frac{0-13}{0+13+1}=-0.93$	$\frac{0}{0+1}=0$

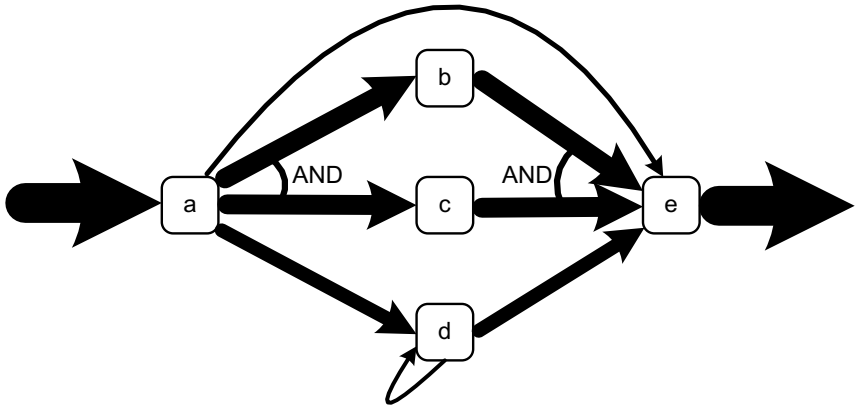
Learning splits and joins



$$L = [\langle a,e\rangle^{5}, \langle a,b,c,e\rangle^{10}, \langle a,c,b,e\rangle^{10}, \langle a,b,e\rangle^{1}, \langle a,c,e\rangle^{1},$$
$$\langle a,d,e\rangle^{10}, \langle a,d,d,e\rangle^{2}, \langle a,d,d,d,e\rangle^{1}]$$

Alternative visualization





$$L = [\langle a,e\rangle^{5}, \langle a,b,c,e\rangle^{10}, \langle a,c,b,e\rangle^{10}, \langle a,b,e\rangle^{1}, \langle a,c,e\rangle^{1},$$
$$\langle a,d,e\rangle^{10}, \langle a,d,d,e\rangle^{2}, \langle a,d,d,d,e\rangle^{1}]$$

Exercise

- Consider more traces for the original log and some noise :

W = [ABCD^{10}, ACBD^{8}, EF^{20}, ABCED, AD]

- Some frequencies for task C:
  - $\#C = 19$
  - $\#A > C = 8,\ \#D > C = 0,\ \#B > C = 11,\ \#E > C = 0$
  - $\#C > D = 10,\ \#C > B = 8,\ \#C > E = 1$
  - $C \rightarrow^{L} D \approx 0.9,\ C \rightarrow^{L} B \approx -0.15$

- Compute the resulting dependency graph with threshold 0.7.

Heuristic Miner Characteristics

- Can deal with noise and therefore quite robust.
- Improved representational bias.
- Split and join rules are only considered locally (therefore most of the discovered model are not sound and require repair actions).

Fuzzy Miner

- Creation of views on process model that are driven by significance and correlation
  - Abstract from undesired details
  - Provide high-level view
- Means to cope with processing complexity
  - Aggregation: cluster similar elements
  - Abstraction (aka Projection): remove low-level information
  - Emphasis: highlight significant information
  - Customisation: adapt to context of model use

The Map Analogy

Abstraction

insignificant roads are not shown.

Aggregation

parts of the city are merged.

Customization

Focuses on the intended use and level of detail.

Emphasis

Highways are highlighted by size, contrast and color.

Significance and Correlation

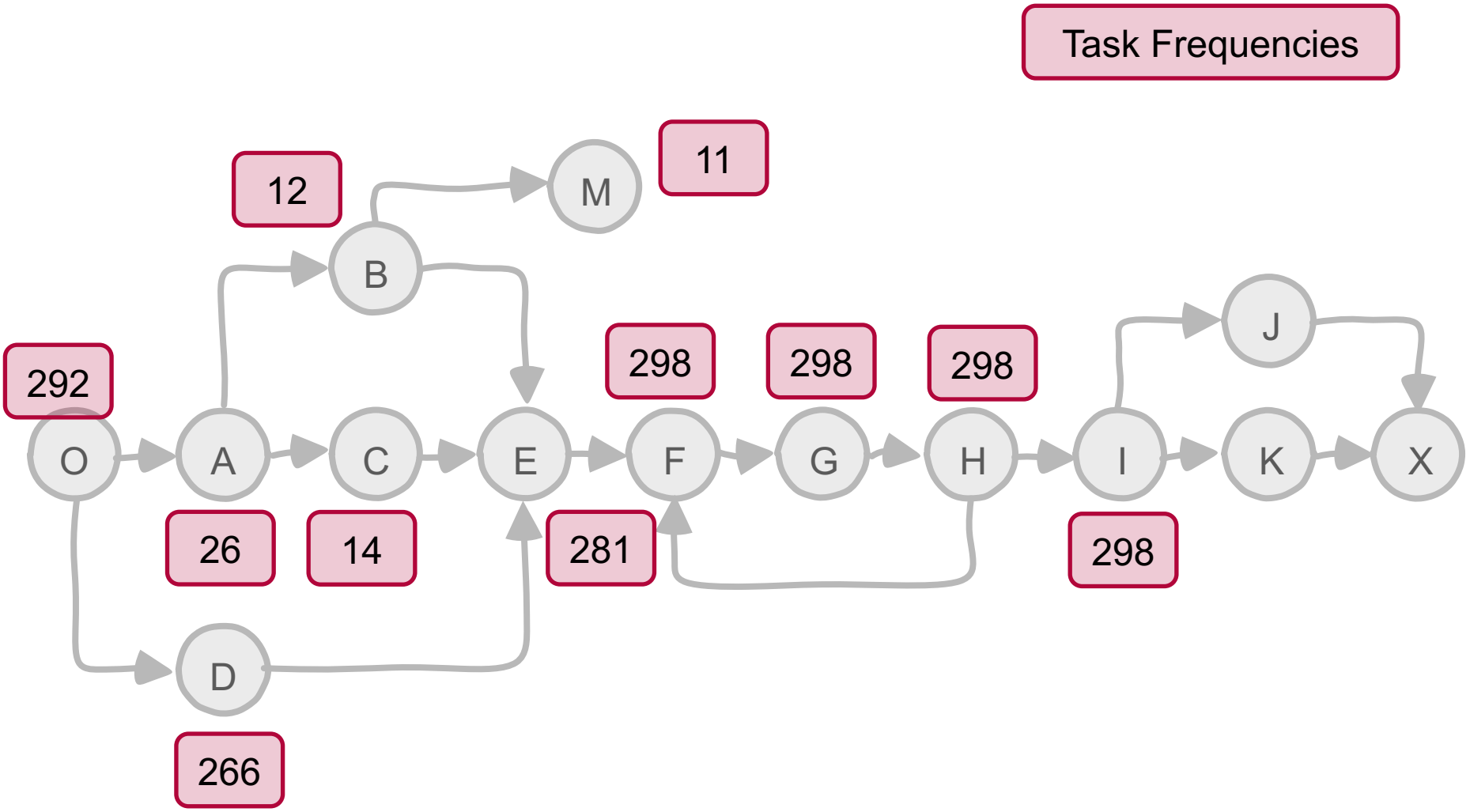
- Metrics to control the creation of views
  - Significance of individual tasks or binary ordering relations (direct successor, causality)
  - Correlation of pairs of tasks

- Idea for simplification of process model:
  - Highly significant behaviour is *preserved*
  - Less significant, but highly correlated behaviour is *aggregated* into clusters
  - Less significant and less correlated behaviour is *abstracted*

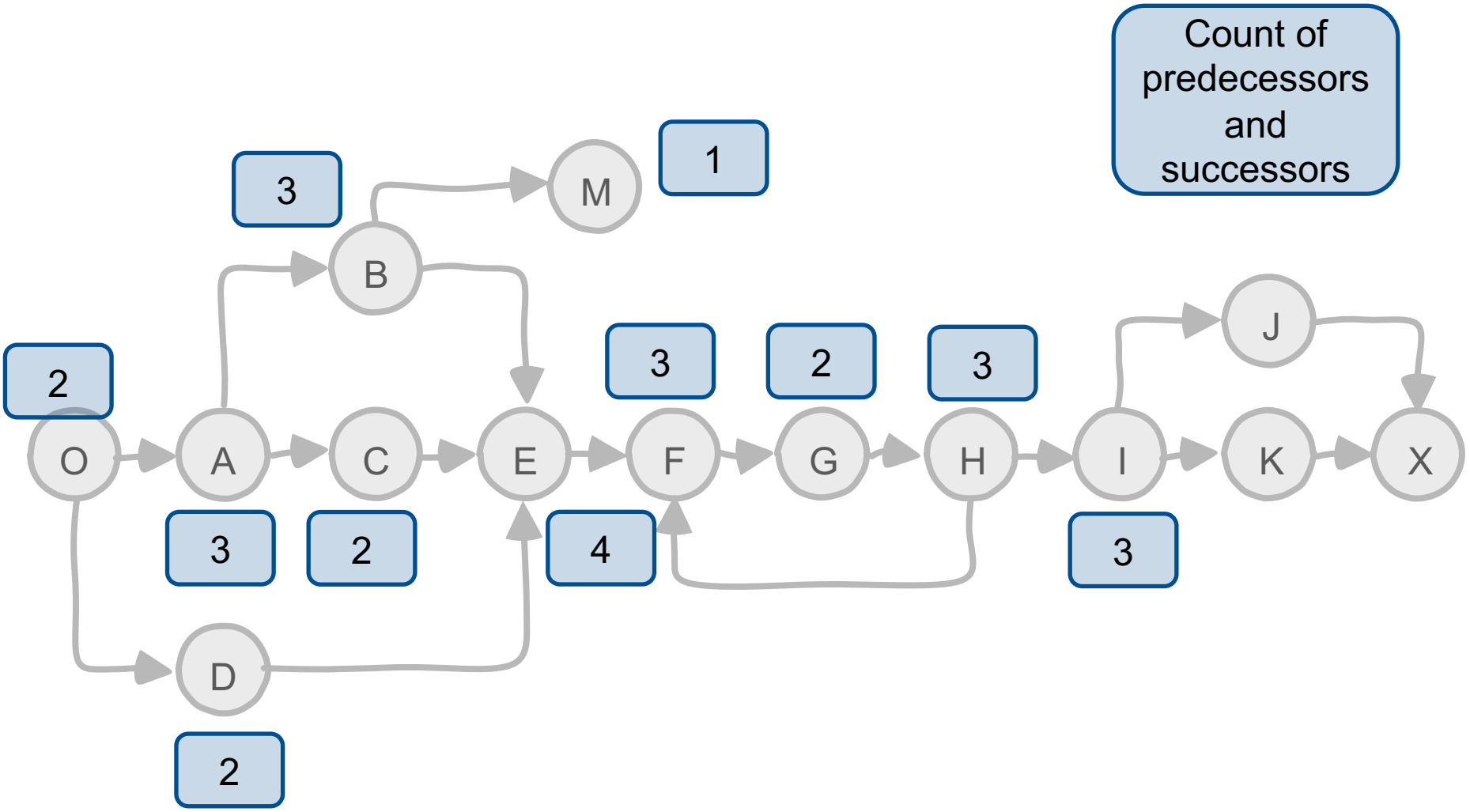
Significance Metrics

- Frequency significance: Frequency of task occurrence
  - But: Can be misleading
  - "Housekeeping" tasks like archiving, storing documents in regular intervals
- Routing significance: Count of distinct predecessors and successors along with their significance
  - Splits and joins typically important to understand process logic
- Relation significance: Frequency of direct successor relation for pairs of tasks
  - In addition, consider difference between relation frequency and frequency significance of source and target nodes

Examples


Task Frequencies
12
M
11
B
292
298
298
298
J
O
A
C
E
F
G
H
I
K
X
26
14
281
298
D
266

Examples


Count of predecessors and successors
O
A
B
C
D
E
F
G
H
I
J
K
X
M
2
3
3
2
3
2
3
4
3
2
1

Correlation Metrics

- Answer the question: abstract or aggregate?
- Proximity correlation: avg distance between execution of tasks in cases
- Many more based on attribute values:
  - Originator correlation: tasks conducted by the same roles
  - Data type correlation: tasks have been executed for partially overlapping data (e.g., the same data objects)

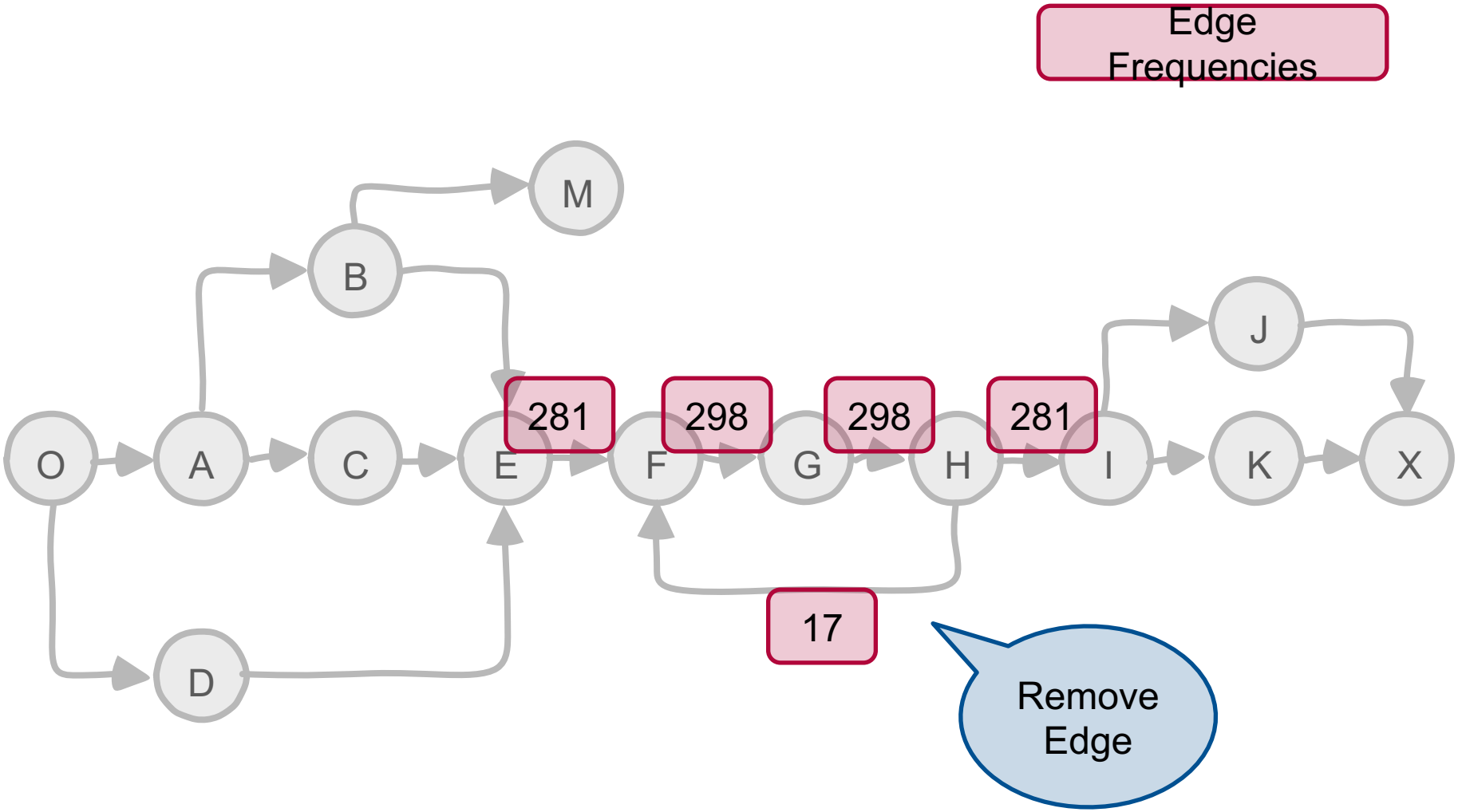
Fuzzy Miner Technique

- Start with basic dependency graph induced by causality relation of α – algorithm
- Based on significance and correlation metrics, transform graph by:
  - Edge filtering
  - Aggregation and abstraction
- Transformed graph can be further processed as discussed earlier (e.g., used to construct a WF-system)

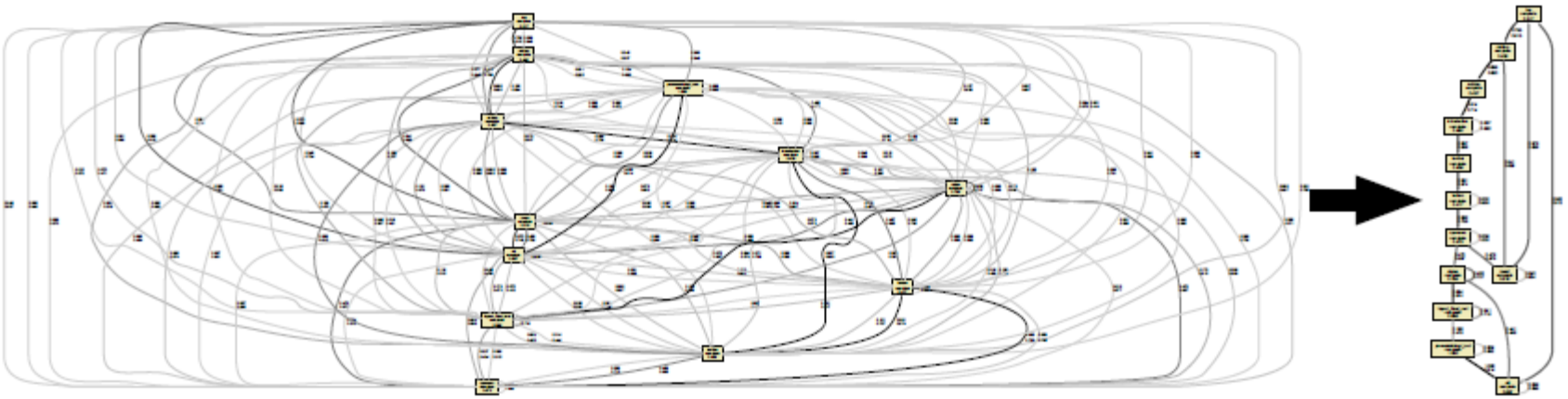
Edge Filtering

- Baseline solution: Remove insignificant edges
  - But: Tendency to create unconnected clusters of frequent behaviour
- Thus: Compute edge utility as weighted sum of its significance and the source-target correlation
  - For each task, maintain incoming and outgoing edge with highest utility value
  - Apart from that, filter based on threshold

Example


Edge Frequencies
M
B
J
281
298
298
281
O
A
C
E
F
G
H
I
K
X
17
D
Remove Edge

Edge Filtering in Practice



Aggregation and Abstraction

- Idea: Tasks with significance below thresholds are *victims* and are aggregated or abstracted

First phase:

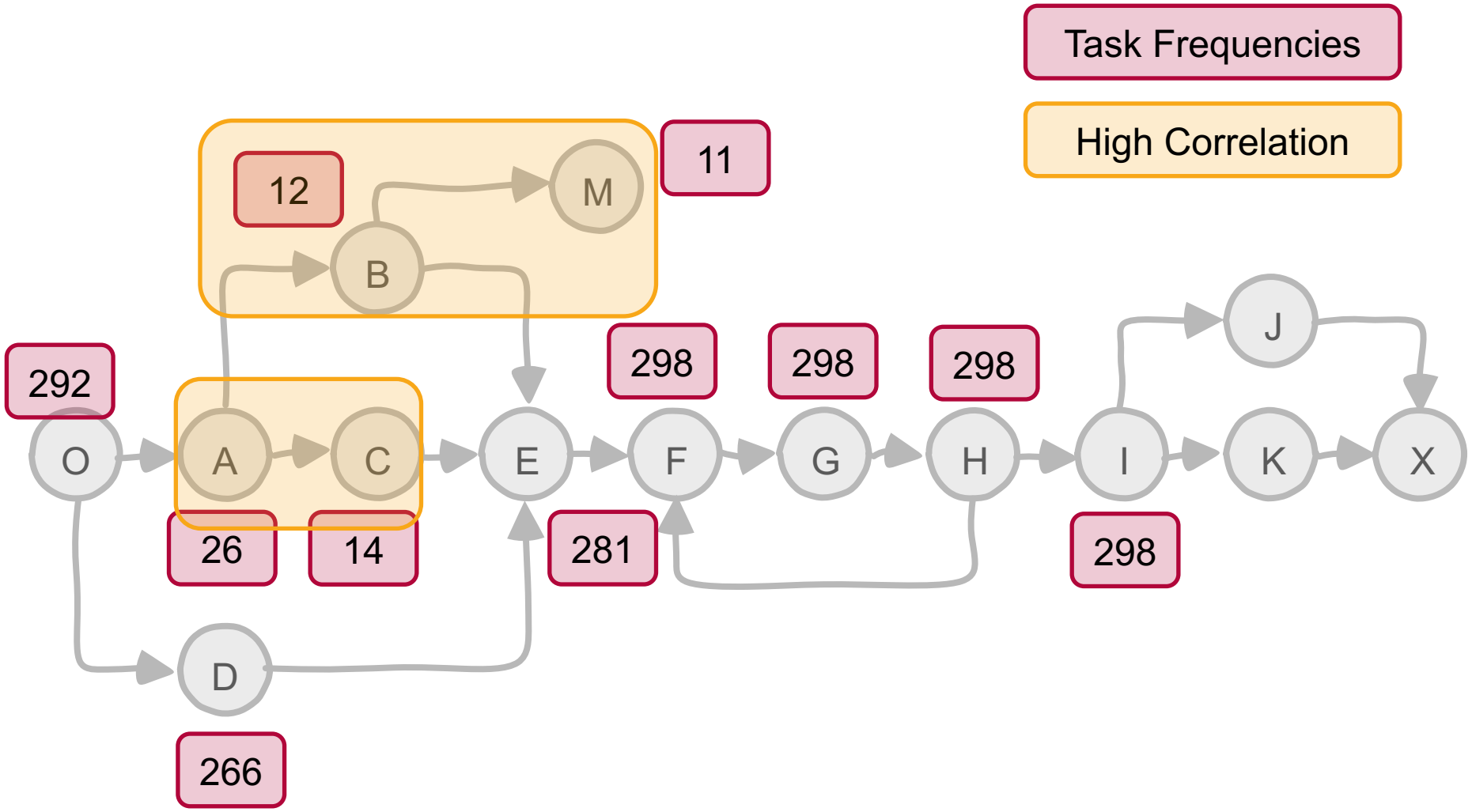
- For each victim, find most highly correlated neighbour
- If neighbour is cluster, add victim
  - Cluster "inherits" incoming and outgoing edges of victim
- Otherwise, create singleton cluster with victim

Second phase:

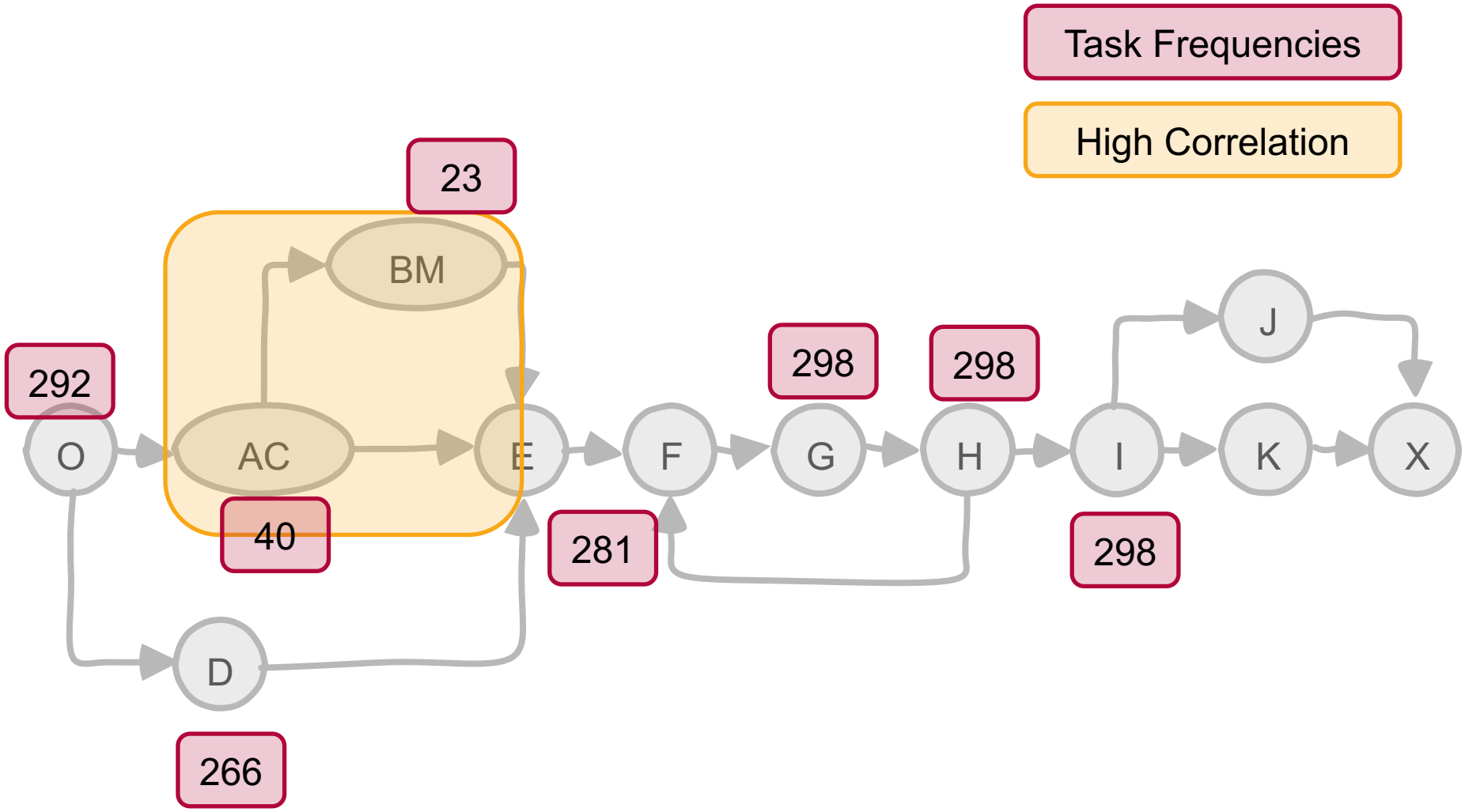
- For each cluster, check whether all predecessor and successors are also clusters
- If so, merge with most highly correlated predecessor or successor cluster, respectively

Third phase: remove isolated clusters and singleton clusters (edges are preserved transitively)

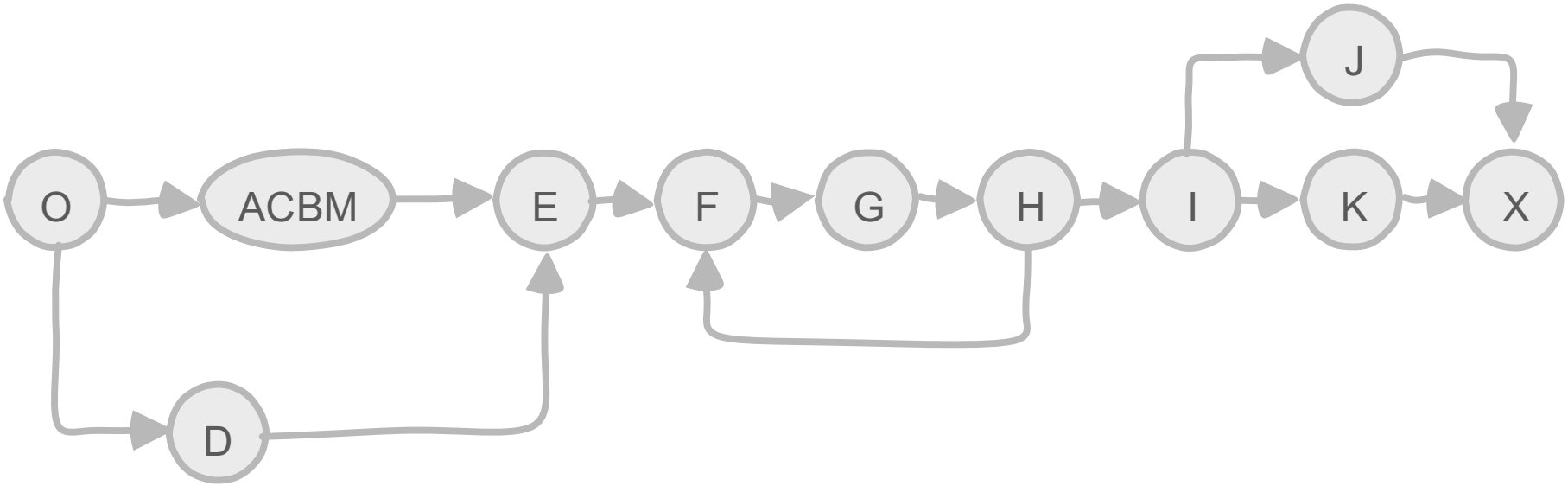
Example


Task Frequencies
High Correlation
12
M
11
B
292
298
298
298
J
O
A
C
E
F
G
H
I
K
X
26
14
281
298
D
266

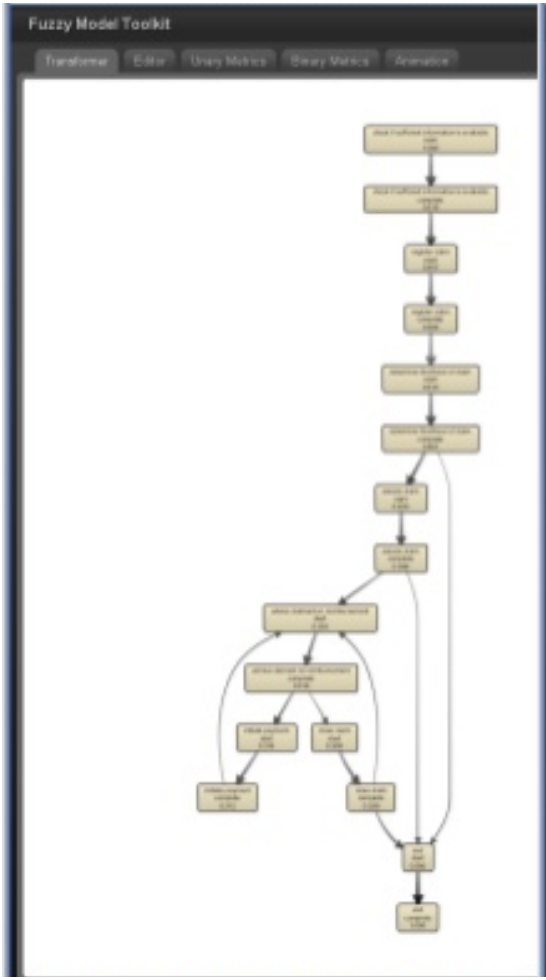
Example

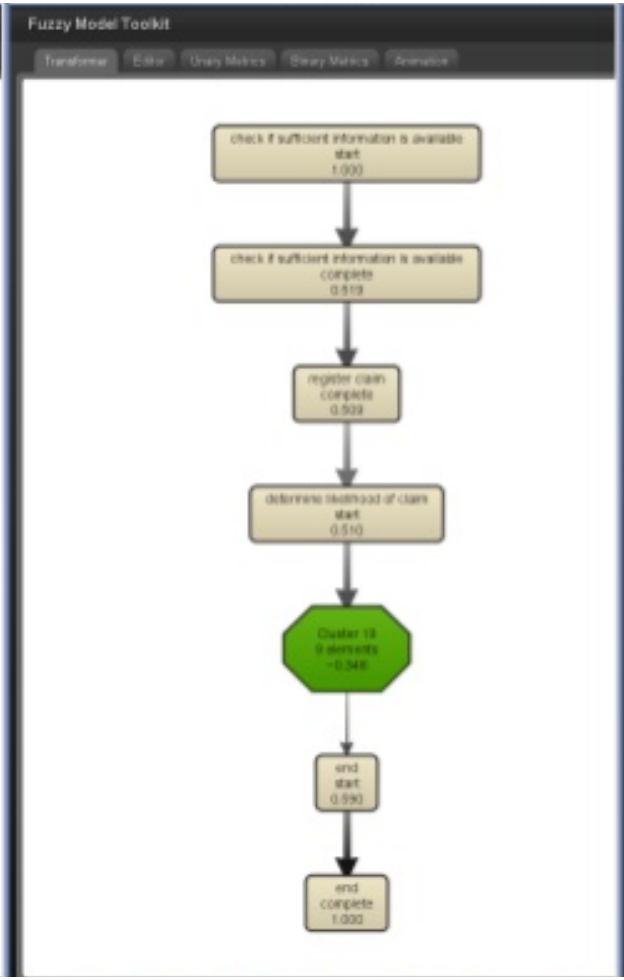

Task Frequencies
High Correlation
23
BM
292
O
AC
E
F
298
G
298
H
J
I
K
X
40
281
298
D
266

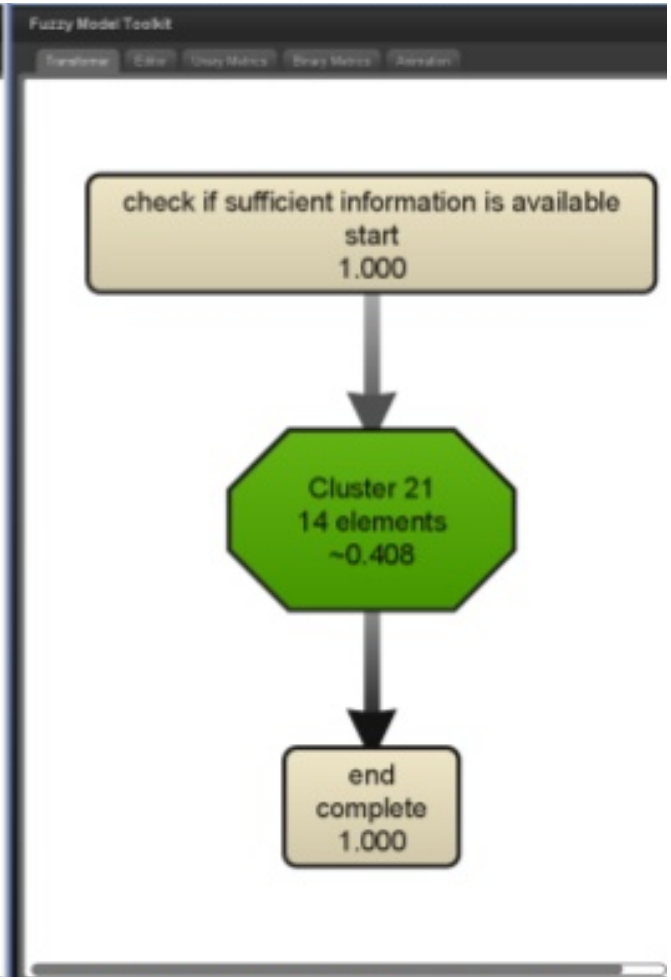
Example



Aggregation and Abstraction in Practice







Take Away

Process Mining needs to be robust against noise and abstraction capabilities

Heuristics to consider occurence freqs to cope with noise

Dependency graph nice intermidiate representation from which models can be generated (challenging often)

Heuristic miner simple yet practical

Fuzzy miner based on map metaphor (abstraction and aggregation)