Process Mining

Part V – Evaluation and verification of models

Conformance check Log-based property verification





Outline

- Part I Introduction to Process Mining
 - Context, motivation and goal
 - General characteristics of the analyzed processes and logs
 - Classification of Process Mining approaches
- Part II Workflow discovery
 - Induction of basic Control Flow graphs
 - Other techniques (α-algorithm, Heuristic Miner, Fuzzy mining)
- Part III Beyond control-flow mining
 - Organizational mining
 - Social net discovery
 - Extension of workflow models

Part IV – Evaluation and validation of discovered models

- Conformance Check
- Log-based property verification

Part V – Clustering-based Process Mining

- Discovery of hierarchical process models
- Discovery of process taxonomies
- Outlier detection



Conformance check and evaluation

- Conformance Check
 - assess how much a process model matches process instances
- Verification of rules/properties against event logs





Conformance Checker

Aim

 Assess how much a process model matches a given set of process instances

Method

Replay process instances in the workflow model

Types of diagnosis

- Fitness
- Structural Appropriateness
- Behavioral Appropriateness



Conformance Checker – Fitness

• Can the model replay the log?





Conformance Checker – Fitness (2)

Can the model replay the log?



Conformance Checker – Structural Appropriateness

Is the model overly complex?





Behavioral Appropriateness



Is the model precise enough?





Behavioral Precision/Recall

- It is an analysis plug-in, which takes as input two models and a log
- Measures (from 0% until 100%) how much behavior two process models have in common w.r.t. the log



LTL Checker

Aim

Verify if process instances fulfill certain properties

Driving force

 Specification of properties in a language based on Linear Temporal Logics

Example

Four-eyes principle



LTL Checker - Example





LTL Checker - Example

ProM [4.2]			
le Mining Analysis Conversion Exports Window Help			
			ProM
Analysis - LTL Checker			r ⊠ ⊠
Select formula :			4
eventually_activity_A_then_B	-	Check options	Open LTL file
		Check untill first failure	Save LTL file
<u>C</u> heck formula		○ Check untill fir <u>s</u> t success	Save LTL file as
		Skip if result is known	26
Description :			-21
Compute if there is an activity with name A and then, eventually ther Arguments:	e is an activity with n	ame B	
• A of type set (ate. WorkflowModelElement)			
B of type set (ata WorkflowModalFlamout) Valuate the parameters :			<u> </u>
A set	Repair (Sir	mple) Repair (Complex)	Set values as <u>d</u> efault
	houriepa		Delete formula
01:19:41 [M] Process mining finished			



LTL Checker – Example

ProM [4.2] e Mining Analysis Conversion E may a the organization of the organization o	cports Window Help	ProN
Checked formula : eventually_activity_ Parameters: A = ["Repair (Complex	A_then_B ", "Repair (Simple)"]	
B = Test Repair Correct process instances (1102) name (nr similar)	Incorrect process instances (2)	
1095 (1) 1127 (1)	Repair (Complex) start 1970-01-24 09:05:00.000 +01:00	
	Repair (Complex) complete 1970-01-24 09:16:00.000 +01:00	
<u>V</u> isualize selected		Zoom: 174 %



LTL – Defining Formulae



<u>IAR</u>

LTL – Defining Formulae







formula accept_or_reject_but_not_both() := {} (<>(activity == "accept") <-> !(<>(activity == "reject")));

🎎 The ProM-framework		. II X
File Mining Analysis Conversion	xports Window Help	
Analysis - LTL Checker Plugin		
Checked formula : accept_or_reject	out_not_both	0000
Correct process instances (400)	hannan hannan (0)	00000
name (id)		
1 (0)		00000
10 (1)	Process Instance Data I	00000
11 (3)		399999
12 (4)	Inuite reviewers	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
14 (6)	2006-01-04 0000000-00100	
15 (7) 16 (8)		33333
17 (9)	gerreview 3 complexe 2006-01-06 2006-00 2006-01-06 2006-00 2000	00000
18 (10)		000000
2 (12)		33333
20 (13) 21 (14)	complete 2006-01-07 00:000-01:00 2006-01-07 00:000:000 -01:00	00000
22 (15)		
23 (16) 24 (17)	dire-out 2 complexe — Criginator =	88888
25 (18)	2006-01-10 000000.000 -0100	
26 (19) 27 (20)		38888
28 (21)	Comprise 2006-01-10 000000.000 -01100 Cfg[nator = Anne	
3 (23)		00000
30 (24)		00000
31 (25) 32 (26)	2006-01-14 00/00/000 -01/00	00000
33 (27)		
35 (29)	accept complete 2006-01-14 00000000 - 01-00 Citylinator = Vike	
36 (30)		
Visualize selected		
		16
A		000000000



formula action_follows_decision() :=
{Is every decision followed by one of the three possible actions? }
[]((activity == "decide" -> _O(((activity == "accept" \/ activity == "reject") \/
activity == "invite additional reviewer"))));

The ProM-framework	
e Mining Analysis Conversion Exports Window Help	
Analysis - LTL Checker Plugin	
checked formula : action_tollows_decision	
Carroct pracose instances (400) [hearrost pracose instances (0)]	
correct process instances (i)	
1 (0)	
10 (1)	
100 (2)	Process Instance Data (
11 (3)	
12 (4)	Indito realewers
13 (5)	Complete Diginator = Wike
14 (6)	
15 (7)	+
15 (8)	get review 1 Originator = Wary
19 (10)	2006-03-20 0010000.000 -01100 result = accept
19 (11)	
2 (12)	ŧ
20 (13)	dime-out 2 complete Diginator =
21 (14)	2006-03-23 00:00:000 -01:00
22 (15)	
23 (16)	dime-out 3
24 (17)	complete 2006-03-30 01/00:000 -02:00
25 (18)	
26 (19)	+
27 (20)	collectroviews
28 (21)	2006-04-01 0100/000 -02/00
3 (23)	
30 (24)	V
31 (25)	complete Criginator = Wil
32 (26)	2006-04-06 0100/000 -02/00
33 (27)	Ļ
34 (28)	accept
35 (29)	complete Diginator = Wike
36 (30)	
VISUALIZE SELECTED	

17



_ 8 ×







subformula accept(a : activity) :=
{Is the result a some activity an accept?}
<> ((activity == a / ate.result == "accept"));

formula dont_reject_paper_unjustified() :=
{Are all papers with strong support indeed accepted.}
(((accept("get review 1") /\ accept("get review 2")) /\ accept("get review 3"))
-> <> (activity == "accept"));









🚜 The ProM-framework		
File Mining Analysis Conversion Exports A	Window Help	
Che ProM-framework		_ 8 ×
File Mining Analysis Conversion Exports W	Mindow Help	
🔲 Analysis - LTL Checker Plugin		막다 🗵
Checked formula : four_eyes_principle		
Correct process instances (99) Incorrect pr	process instances (1) Origination white	
41 (36)		EU
	get review 2	1000
	complete Uriginator = John result = reject	
	get review 1 Originator = John	
	2006-07-14 01:00:00.000 +02:00	
	get review 3 complete Driginator = Pete result = reject	
	collect reviews	
	oomplete Originator = Mike 2006-07-22 01:00:00.000 +02:00	
	decide complete Driginator = Will	
	2006-07-27 01:00:000 +02:00	
Visualize selected		_



Semantic Process Analysis: Motivation

Business Process Management Systems

 Paral (2)
 \$7.50

 In Mark Andre (merrors (1400) Mode He
 \$7.50

 In Mark Andre (1400) Mode He
 \$7.50

 In Mark Andre (1400) Mode He
 \$7.50

How many requested were for pre-defined bundles?



Inform me whenever running instances for predefined bundles are taking longer than 15 days.

Best Effort: A,B,C Silver: C,D,E Gold: S,W



Semantic Process Analysis: Motivation

Semantic Business Process Management Systems





Semantic LTL Checker

Semantic analysis

- Match based on concepts, not labels
- Audit at the concept level





🕌 ProM [4.2]		
File Mining Analysis Conversion Exports Window Help		
		ProM
Analysis - Semantic LTL Checker		r 5 🛛
Select formula : eventually_activity_A	Check options	Open LTL file
	Check <u>w</u> hole process Check untill <u>f</u> irst failure	Save LTL file
<u>Check formula</u>	○ Check untill fir <u>s</u> t success	Save LTL file as
	Skip if result is known	
Description :		
Does activity A occur?		
Compute if there is an activity with name A		=
Arguments:		
A of type set (ate. WorkflowModelElement)		
		Set values as default
A set O Instance: Prepare Best Effort Prepare Gold Prepare Silver		Set values as <u>u</u> erault
Optologer Nexcomulativities Concent: PredefinedBundle	Include super concents	Delete formula
Ē	Include sub concepts	
► 15:01:13 [T]		





Analysis - Semantic LTL Checker			ວ້ ຫ້ 🗵
select formula : overbuilly_actwity_A	-	Check options	Open I. TI. file
Check formula		Check until first failure	Save LTL file as
		Ship if result is known	In the second se
onscription : Does activity A occur? Compute if there is an activity with name A			
Description: Description: Compute if there is an activity with name A. Argements: • A of type in (ate Workfloodd/Element)			
Descriptions: Descriptions: Compute If there is a schoolty with name A Argements:			soft values as gefault
Precipion: Does activity A occur? Computer I have a schooly with means A. Agamente: A of type and (also life/schoold/life/sond) A of type and (also life/schoold/life/sond) A of the parameters : A of Declarge Recommendation (Precipion and Precipion		ł	Sof values as globall Delete formula