Analisis dan Implementasi Process Mining dengan Algoritma Heuristic Miner studi kasus: event logs Rabobank Group ICT Netherlands

Analysis And Implementation Of Process Mining with Heuristic Miner Algorithm Case study: Event logs of Rabobank Group ICT Netherlands

Rendy Setiadi Mangunsong [1], Angelina Prima Kurniati[2], Mira Kania Sabariah [3]

[123]Fakultas Informatika, Universitas Telkom, Bandung
[1]rendy.mangunsong91@gmail.com, [2] angelina@telkomuniversity.ac.id, [3]mirakania@telkomuniversity.ac.id

Abstrak

Process mining dapat meningkatkan analisis proses bisnis pada perusahaan dengan lebih akurat. Ide dasarnya adalah dengan memanfaatkan informasi berupa event log yang terekam dalam sistem informasi perusahaan. Berdasarkan event log tersebut, process mining dapat menyajikan struktur proses bisnis perusahaan yang ada beserta berbagai elemen dan masalah yang mungkin ada dan terjadi di dalam perusahaan.

Pada tugas akhir ini process mining dilakukan terhadap event log dari Rabobank Group ICT, sebuah bank yang sedang meningkatkan pelayanannya dengan menelusuri aktifitas yang sudah berlangsung dan memberikan rekomendasi berdasarkan hasil penelusuran tersebut. Process mining dilakukan dengan menggunakan framework process mining, yaitu PROM 5.2 dan plugins Heuristic Miner. Algoritma heuristic miner dipilih karena kemampuannya untuk menangani event logs dengan noise, dan dapat menampilkan main behavior dari proses bisnis yang ada[1]. Dengan melakukan process discovery terhadap event logs yang ada, diharapkan proses bisnis yang ada dapat dimodelkan secara utuh dan dapat dilakukan analisis lebih lanjut. Hal ini kemudian dapat digunakan untuk melakukan rekomendasi yang bermanfaat untuk peningkatan kualitas layanan Rabobank ICT ke depannya.

Kata kunci: ***process mining, event log, proses bisnis, heuristic miner, process discovery, rekomendasi***

Process mining can enhance the analysis of business processes within the companies with more accuracy. The basic idea is to take advantages from the information in the event logs recorded in the enterprise information system. Based on the event logs, process mining can present the structure of existing business processes, their various elements, and the problems that happened in the company.

In this final project, process mining conducted on the event logs of the Rabobank Group ICT, a bank which have to increasing its service with discovers the activities that have taken place and provide recommendations based on the results. Process mining conducted by using the ProM 5.2 framework and heuristic miner plugins. Heuristic miner algorithms chosen for its ability to handle the event logs with noise, and can show the main behavior from the business processes [1]. From the process discovery done with event logs, the business processes expected to be modeled and analyzed as a whole process. After that, the results will give recommendations to improve the services quality of Rabobank ICT in the future.

Keyword: process mining, event logs, business processes, heuristic miner, process discovery, recommendation

1. Pendahuluan

1.1. Latar Belakang

Perkembangan dunia ICT yang sangat pesat pada era modern saat ini, juga berdampak cukup signifikan pada sistem informasi perusahaan. Salah satu fungsi penting dari sistem informasi perusahaan yaitu menyimpan data mengenai proses bisnis yang berjalan di dalamnya. Pada umumnya, data tersebut tersimpan dalam bentuk event logs. Event logs ini merupakan rekaman kronologis dari aktivitas-aktivitas di dalam sistem yang kemudian disimpan agar nantinya daat dilihat kembali oleh administrator untuk mengidentifikasi user actions terhadap sistem atau proses-proses yang terjadi[4]. Informasi penting yang tersimpan pada event logs tersebut dapat dimanfaatkan melalui metode process mining. Process mining pada dasarnya adalah sebuah teknik untuk mengambil informasi penting yang terdapat pada event logs yang direkam oleh sistem informasi yang ada[1] dan memodelkannya ke dalan bentuk visual yang lebih mudah dipahami [5].

Pada tugas akhir ini dilakukan process mining pada event logs Rabobank Group ICT, Belanda. Rabobank Group ICT merupakan sebuah divisi Information and Communications Technology (ICT) pada perusahaan layanan keuangan yang berbasis di Belanda, yaitu Rabobank Group. Saat ini, Rabobank Group ICT sedang berusaha mengimplementasikan peningkatan software pada perusahaan untuk mendukung layanan pada bagian

Service Desk dan IT Operations. Hal ini dilakukan berdasarkan analisis terhadap proses yang sudah berlangsung dalam kurun waktu Januari 2013 sampai dengan Maret 2014. Data tersebut diberikan dalam bentuk event logs sehingga dapat dilakukan process mining untuk menemukan dan melakukan analisis terhadap model proses yang berlangsung pada Service Desk Agent (SDA)..

Process mining dilakukan dengan menggunakan framework process mining, yaitu PROM 5.2 dan Algoritma Heuristic Miner. Algoritma Heuristic Miner dipilih karena kemampuannya untuk menangani event logs dengan noise, dan dapat menampilkan main behavior dari proses bisnis yang ada[1]. Selain itu, dengan algoritma heuristic miner dapat dilihat secara jelas hubungan keterkaitan antar proses yang berlangsung sehingga analisis dapat dilakukan secara objektif berdasarkan proses-proses yang berlangsung. Dengan melakukan process discovery terhadap event logs yang ada, diharapkan proses bisnis yang ada dapat dimodelkan secara utuh dan dapat dilakukan analisis lebih lanjut. Hal ini kemudian dapat digunakan untuk melakukan peningkatan terhadap proses bisnis yang ada sehingga bermanfaat untuk peningkatan kualitas layanan Rabobank ICT ke depannya..

1.2. Perumusan Masalah

Berdasarkan hal-hal yang dijabarkan pada latar belakang, penulis menyimpulkan adanya permasalahan utama sebagai berikut :

RQ 1. Bagaimana melakukan preprocessing event logs Rabobank Group ICT sebagai input dari process mining?

RQ 2. Bagaimana analisis dan performansi algoritma Heuristic Miner dalam melakukan pemodelan proses bisnis Rabobank Group ICT?

RQ 3. Bagaimana rekomendasi berdasarkan model proses hasil process mining dengan algoritma Heuristic Miner terhadap event logs Rabobank Group ICT?

1.3. Tujuan

Sesuai dengan permasalahan yang telah dirumuskan, tujuan yang ingin dicapai dari tugas akhir ini adalah:

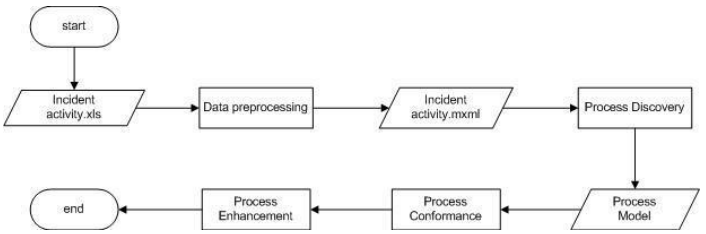
1. Menghasilkan *output* dari preprocessing *event logs* Rabobank Group ICT yang dapat digunakan sebagai input dari *process mining* dengan algoritma *heuristic miner*
2. Mendapatkan model proses terbaik sebagai hasil dari process mining dengan algoritma heuristic miner pada event logs Rabobank Group ICT
3. Menghasilkan rekomendasi kepada Rabobank Group ICT berdasarkan analisis model proses terbaik setelah dilakukan process mining dengan algoritma heuristic miner pada event logs Rabobank Group ICT

1.4. Batasan Masalah

Penulis membatasi masalah pada tugas akhir ini sebagai berikut:

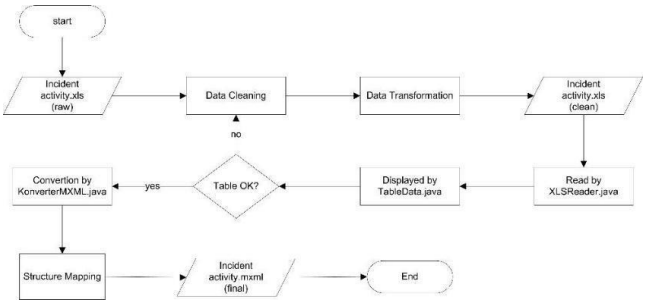
1. Proses bisnis yang akan dimodelkan dan dianalisis dengan process mining menggunakan tools ProM 5.2 dan plug-ins Heuristic Miner dengan masukan berupa event logs berbentuk MXML
2. Melakukan Process mining dengan algoritma heuristic berdasarkan event logs pada Rabobank Group ICT.
3. *Process mining* yang dilakukan sampai *process discovery* dan *conformance checking*, dengan hasil berupa *recommendation* yang diberikan kepada Rabobank Group ICT

2. Deskripsi dan Gambaran Umum Sistem



Gambar 1: flowchart penelitian

2.1. Preprocessing event log



Gambar 2 : flowchart preprocessing

Preprocessing event logs dilakukan melalui proses data cleaning. Tahap pertama yang dilakukan adalah dengan membuang atribut yang tidak digunakan.

Yang digunakan dalam penelitian ini adalah empat buah kolom yang masing-masing kolomnya memiliki peran sebagai berikut:

a. Case ID, merupakan identitas yang menentukan ruang lingkup proses. Pada penelitian ini ruang lingkup prosesnya adalah aktivitas pada bagian incident, sehingga yang digunakan sebagai Case ID adalah kolom Incident ID. Setiap Case ID mewakilkan satu buah proses yang berbeda dari proses lainnya.

b. Activity, merupakan atribut dengan nilai yang menggambarkan tahap-tahap dari proses yang berjalan. Pada tabel 3.1 di atas, kolom yang merepresentasikan activity adalah kolom IncidentActivity_Type. Dapat dilihat bahwa setiap case ID memiliki beberapa activity

yang berlangsung seperti Reassignment, Update from customer, Assignment, dsb.

c. Timestamp, merupakan atribut yang menjadi penanda waktu terjadinya activity dalam setiap case. Setiap activity memiliki timestamp masing-masing. Timestamp sangat penting untuk menetapkan urutan terjadinya activity mulai dari start sampai end dari satu case ID. Pada tabel 3.1 di atas terlihat yang menjadi timestamp adalah kolom datestamp yang memiliki nilai waktu berupa tanggal dan jam terjadinya suatu activity.

d. Originator, merupakan atribut yang menunjukkan pelaku atau penanggungjawab dari setiap activity yang berlangsung. Pada event log yang digunakan sebagai originator adalah kolom Assignment Group yang berisi nama tim sebagai pelaku dari setiap activity.

Tabel 1 : event log setelah data yang tidak deiperlukan dihilangkan

Incident ID	DateStamp	IncidentActivity_Type	Assignment Group
IM0000004	2013-01-07 08:17	Reassignment	TEAM0001
IM0000004	2013-11-04 13:41	Reassignment	TEAM0002
IM0000004	2013-11-04 13:41	Update from customer	TEAM0002
IM0000004	2013-11-04 12:09	Operator Update	TEAM0003
IM0000004	2013-11-04 12:09	Assignment	TEAM0003
IM0000004	2013-11-04 13:41	Assignment	TEAM0002
IM0000004	2013-11-04 13:51	Closed	TEAM0003
IM0000004	2013-11-04 13:51	Caused By CI	TEAM0003
IM0000004	2013-11-04 12:09	Reassignment	TEAM0003
IM0000004	2013-09-25 08:27	Operator Update	TEAM0003
IM0000005	2013-06-03 11:15	Update	TEAM9999
IM0000005	2013-04-03 11:29	Operator Update	TEAM0003
IM0000005	2013-01-07 08:17	Reassignment	TEAM0001
IM0000005	2013-09-05 08:58	Operator Update	TEAM0003
IM0000005	2013-04-12 11:03	Operator Update	TEAM0003
IM0000005	2013-04-23 08:22	Status Change	TEAM0003
IM0000005	2013-12-02 12:00	Update from customer	TEAM0002
IM0000005	2013-12-02 12:32	Reassignment	TEAM0002
IM0000005	2013-12-02 12:32	Assignment	TEAM0002

Tahap selanjutnya yang dilakukan adalah dengan mengubah nama kolom menjadi Case ID, Activity, Timestamp, dan Originator supaya sesuai dengan deskripsi dan peran dari setiap kolom. Selanjutnya, dilakukan proses sorting terhadap data. Hal ini bertujuan supaya terlihat jelas urutan-urutan berlangsungnya activity dalam suatu case. Sorting hanya dilakukan berdasarkan timestamp, karena secara default data sudah terurut berdasarkan case ID. Pada tabel 2 berikut dapat dilihat secara jelas event log yang sudah dilakukan proses sorting dan cleaning sehingga siap untuk digunakan dalam prcess mining.

Tabel 2 : event log setelah dilakukan cleaning dan sorting

IncidentID	Timestamp	Activity	Originator
4	2007-01-13 08:17	Reassignment	TEAM0001
4	2025-09-13 08:27	Operator Update	TEAM0002
4	2004-11-13 12:09	Operator Update	TEAM0002
4	2004-11-13 12:09	Assignment	TEAM0003
4	2004-11-13 12:09	Reassignment	TEAM0003
4	2004-11-13 13:41	Reassignment	TEAM0002
4	2004-11-13 13:41	Update from customer	TEAM0003
4	2004-11-13 13:41	Assignment	TEAM0003
4	2004-11-13 13:51	Closed	TEAM0003
4	2004-11-13 13:51	Caused By CI	TEAM0003
5	2007-01-13 08:17	Reassignment	TEAM9999
5	2027-03-13 08:57	Operator Update	TEAM0003
5	2027-03-13 08:57	Description Update	TEAM0001
5	2003-04-13 11:29	Reassignment	TEAM0003
5	2003-04-13 11:29	Operator Update	TEAM0003
5	2003-04-13 11:29	Assignment	TEAM0003
5	2012-04-13 11:03	Operator Update	TEAM0002
5	2012-04-13 11:03	Status Change	TEAM0002
5	2016-04-13 14:19	Status Change	TEAM0002

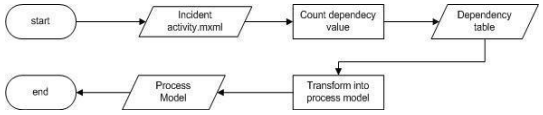
Tahap selanjutnya adalah konversi dengan menggunakan aplikasi yang dibuat. Aplikasi berbasis java dibangun dengan menggunakan netbeans dan memiliki fungsi untuk mengkonversi file xls menjadi file mxml yang digunakan sebagai input dari process mining. Aplikasi dapat membaca empat kolom dari file excel (caseID, activity, timestamp, dan originator) yang telah didefinisikan pada Datalog.java. XLSReader.java akan membaca keseluruhan event logs berupa file excel yang diinputkan ke dalam aplikasi.

Proses konversi dilakukan oleh KonverterMXML.java. Proses pertama yaitu aplikasi membaca inputan berupa file xls. Jika yang dipilih bukan file xls, maka akan ditampilkan pesan error. Setelah aplikasi mendapatkan inputan data yang benar berupa file xls, aplikasi membaca dan menyalin seluruh kolom yang akan dikonversi mulai dari baris pertama sampai baris terakhir dari file xls. Jika sudah semua maka program akan berhenti menyalin.

Tahap selanjutnya yaitu program melakukan pemecahan dan pemetaan masing-masing caseID dari file excel ke dalam struktur file mxml. Struktur awal yang berupa baris-baris file excel akan dikelompokkan menjadi baris-baris syntax mxml. Syntax ini dicatat secara berurutan mulai dari case ID terkecil sampai terbesar yang ada pada log, dengan rincian detail activity, timestamp, dan originator yang dicatat juga. Program akan berhenti jika sudah semua baris dalam log selesai dikonversi ke dalam syntax mxml. Setelah selesai, event log yang berhasil dikonversi disimpan secara default pada file project Netbeans pada

C:\Users\John\Documents\NetBeansProjects\Konverter MXML\res.

2.2. Process Discovery



Gambar 3 : Flowchart Process Discovery

Untuk menggambarkan model proses, terlebih dahulu dilakukan perhitungan terhadap nilai dependency dari setiap activity yang ada. Sampel event log yang digunakan yaitu:

L=[(a, j)^4, (a, j, i)^7, (a, b, j, i)^{10}, (a, c, b, j, i)^{10}, (a, b, c, j, i)^{10}, (a, e, d, b, j)^{10}, (a, b, c, f, f, i, j)^4, (a, c, b, g, h, i, j)^5]

Selanjutnya, untuk menghitung nilai dependency akan digunakan persamaan berikut:

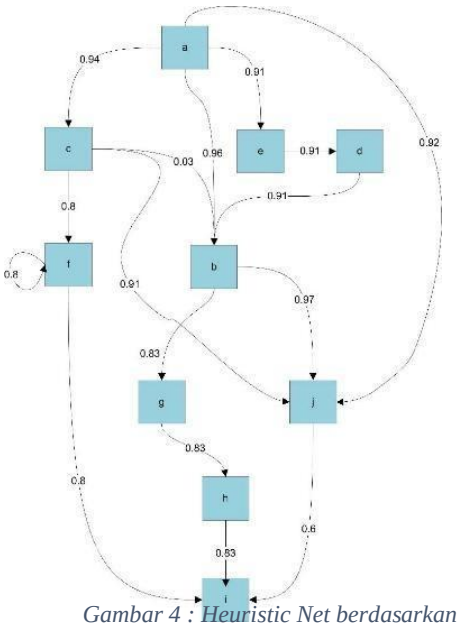
$$|a \Rightarrow_L b| = \begin{cases} \frac{|a>_L b| - |b>_L a|}{|a>_L b| + |b>_L a| + 1} & \text{if } a \neq b \\ \frac{|a>_L a|}{|a>_L a| + 1} & \text{if } a = b \end{cases}$$

Dimana $|a \Rightarrow_L b|$ menunjukkan nilai dependency relation antara a dan b, sedangkan $|a >_L b|$ menunjukkan jumlah dari activity a diikuti langsung oleh b pada event log . Nilai dari $|a \Rightarrow_L b|$ berada antara -1 dan 1. Jika mendekati 1, maka kedua activity memiliki dependency relation yang kuat atau dengan kata lain activity a seringkali diikuti oleh activity b, tetapi activity b jarang atau hamper tidak pernah mengikuti activity a. Jika nilainya mendekati -1, maka dapat disimpulkan bahwa b adalah activity yang selalu mendahului a. Hasil perhitungan nilai dependency pada sampel ditunjukkan pada tabel 3 berikut.

Tabel 3 : Nilai Dependency antar activity

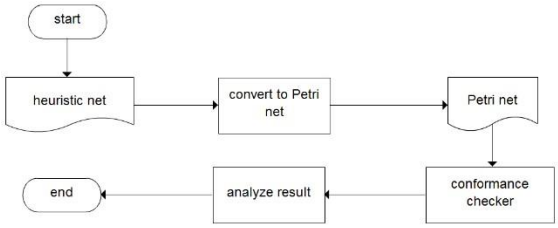
	a	b	c	d	e	f	g	h	i	j
a	0	0.96	0.94	0	0.91	0	0	0	0	0.92
b	-0.96	0	-0.03	-0.91	0	0	0.83	0	0	0.97
c	-0.94	0.03	0	0	0	0.8	0	0	0	0.91
d	0	0.91	0	0	-0.91	0	0	0	0	0
e	-0.910	0	0	0.91	0	0	0	0	0	0
f	0	0	-0.8	0	0	0.8	0	0	0.8	0
g	0	-0.83	0	0	0	0	0	0.83	0	0
h	0	0	0	0	0	0	-0.83	0	0.83	0
i	0	0	0	0	0	-0.8	0	-0.83	0	-0.6
j	-0.92	-0.97	-0.91	0	0	0	0	0	-0.6	0

Nilai dependency yang menunjukkan angka nol, misalnya pada relasi antara a-d, d-f, d-j, dan lainnya menunjukkan tidak adanya hubungan antar kedua activity. Hal ini mengakibatkan tidak terbentuk koneksi antar kedua activity pada heuristic net. Sedangkan nilai dependency yang menunjukkan angka lebih besar dari nol, menunjukkan adanya koneksi yang akan tergambar pada heuristic net. Untuk lebih jelasnya dapat dilihat pada gambar 4 di bawah. Kotak berisi huruf a-j adalah activity yang ada pada event log, sedangkan panah menunjukkan dependency antar dua activity dengan nilai dependency yang tercantum pada setiap panahnya.



Gambar 4 : Heuristic Net berdasarkan tabel 3

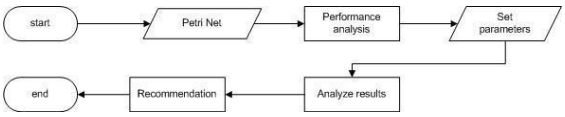
2.3. Process Conformance



Gambar 5 : flowchart process conformance

Setelah process mining dilakukan dan hasil model proses didapatkan, tahap selanjutnya adalah melakukan conformance checking. Proses ini merupakan tahap validasi terhadap process model yang dihasilkan sebelumnya, apakah sesuai dengan event log yang ada dan juga sebaliknya, apakah event log yang ada sudah sesuai dengan process model nya. Pada tahap ini akan dilakukan transformasi dari model proses yang berbentuk heuristic net menjadi Petri net dengan bantuan ProM. Setelah itu, model proses yang sudah berbentuk Petri net dianalisis dengan menggunakan aplikasi Conformance checker. Hasilnya adalah berupa nilai fitness, precision, dan structure yang menunjukkan tingkat kecocokan antara event log dan process model dan juga sebaliknya.

2.4. Process Enhancement



Gambar 6 : process enhancement

Pada tahap ini dilakukan analisis berdasarkan waktu terhadap model proses yang ada dan juga rekomendasi berdasarkan hasil analisis tersebut. Dengan melakukan identifikasi bottleneck, waktu tempuh antar aktifitas yang berlangsung dapat diketahui. Proses ini masih menggunakan Petri net sebagai model utama yang akan dianalisis. Sebagai hasilnya, diperoleh analisis performansi berupa informasi mengenai waktu tempuh dengan skala rendah, sedang, dan tinggi. Adanya bottleneck ditunjukkan oleh informasi token berwarna ungu pada petri net hasil analisis performansi.

Tahap selanjutnya adalah proses analisis berdasarkan informasi yang dikumpulkan berdasarkan tahap-tahap sebelumnya. Informasi dari hasil analisis conformance checking dan bottleneck digunakan untuk merumuskan recommendation yang ditujukan kepada perusahaan. Recommendation berisi kumpulan saran-saran sebagai perbaikan terhadap proses bisnis yang berjalan pada perusahaan dan bertujuan untuk meningkatkan performansi perusahaan di masa yang akan datang.

3. Skenario Pengujian

3.1. Skenario Satu

Pada skenario ini akan dilakukan pengujian terhadap data sebanyak 65.000 event yang diambil dari event log. Event log dengan jumlah 65.000 tersebut merupakan event log yang bebas dari low case–variants atau variasi activity dengan jumlah case rendah, sebagai contoh varian dengan 1 atau 2 case saja. Banyaknya jumlah variasi dengan jumlah case rendah dapat menyebabkan jalur antar case atau arc yang tergambar pada model proses menjadi sangat banyak sehingga menghambat penemuan model proses utama pada event log tersebut.

Log yang berupa file .mxml diproses dengan ProM 5.2 dengan berbagai kombinasi dari parameter dependency threshold, relative to the best threshold, dan positive observation threshold. Hasilnya akan ditampilkan dalam tabel dan diuraikan dalam bentuk analisis pengaruh ketiga parameter tersebut terhadap model proses yang dihasilkan.

3.2. Skenario Dua

Pada skenario ini model proses terbaik yang didapatkan dari skenario pengujian satu akan dianalisis secara lengkap melalui conformance analysis with petri net. Hasilnya yang berupa nilai fitness, precision, dan structure akan diuraikan melalui analisis lebih lanjut

3.3. Skenario Tiga

Pengujian yang akan dilakukan adalah baseline Pada skenario ini model proses terbaik yang didapatkan dari skenario pengujian satu akan dianalisis dengan menggunakan performance analysis with petri net untuk mengidentifikasi adanya botlleneck dan juga analisis dari sudut pandang performansi lainnya

4. Hasil Pengujian dan Analisis

4.1. Hasil Pengujian dan Analisis Skenario Satu

Tabel 4.1: Hasil pengujian terhadap parameter dependency threshold

dependency threshold	relative-to-best threshold	positive observation threshold	number of connections	wrong observations	fitness
0.9	0.05	1000	36	328	0.8494
0.7	0.05	1000	36	328	0.8494
0.5	0.05	1000	36	328	0.8494
0.3	0.05	1000	36	328	0.8494
0.1	0.05	1000	36	328	0.8494

Tabel 4.2 : Hasil pengujian terhadap parameter relative-to-best threshold

dependency threshold	relative-to-best threshold	positive observation threshold	number of connections	wrong observations	fitness

0.9	1	1000	36	328	0.8494
0.9	0.75	1000	36	328	0.8494
0.9	0.5	1000	36	328	0.8494
0.9	0.25	1000	36	328	0.8494
0.9	0.05	1000	36	328	0.8494

Tabel 4.3 : Hasil pengujian terhadap parameter positive observation threshold

dependency threshold	relative-to-best threshold	positive observation threshold	number of connections	wrong observations	fitness
0.9	0.05	10	62	333	0.6593
0.9	0.05	100	47	333	0.6725
0.9	0.05	200	40	333	0.7101
0.9	0.05	500	38	328	0.8374
0.9	0.05	1000	36	328	0.8494
0.9	0.05	6500	34	6046	0.7592

Berdasarkan hasil pengujian seperti yang ditunjukkan pada tabel di atas, nilai parameter *positive observation threshold* yang diubah memiliki pengaruh terhadap nilai fitness dan juga nilai dari number of connections dan wrong observations. Nilai *default* pada parameter *positive observation threshold,* yaitu 10 menyebabkan seluruh aktifitas yang memiliki frekuensi rendah diikutsertakan dalam model proses. Hal ini menyebabkan nilai dari number of connections menjadi paling tinggi pada pengujian ini. Selain itu juga nilai fitness yang dihasilkan menjadi paling rendah. Semakin kecil nilai dari *threshold* ini, maka semakin banyak *low-frequent activity* yang ikut serta pada model proses dan menyebabkan meningkatnya dari number of connections dan wrong observations yang terbentuk.

Sebaliknya jika nilai *positive observation threshold* ditingkatkan, akan menyebabkan meningkatnya nilai fitness yang diikuti juga dengan menurunnya number of connections dan wrong observations yang terbentuk pad amodel proses. Hal ini disebabkan seiring peningkatan dari parameter *positive observation threshold,* semakin sedikit *low-frequent activity* yang ikut serta pada model proses. Tetapi, ada sat dimana nilai *positive observation threshold* yang terlalu besar juga menghasilkan number of connections dan wrong observations yang meningkat dan nilai fitness yang semakin menurun. Hal itu disebabkan activity yang memiliki frekuensi di atas nilai threshold yang terlalu besar menjadi sangat sedikit.

Hasil nilai fitness terbaik pada pengujian parameter *positive observation threshold* ini didapatkan ketika nilai *positive observation threshold* 1000, dependency threshold 0.9, dan *relative-to-best threshold* 0.05. Dapat disimpulkan bahwa dari ketiganya, yang berpengaruh hanya parameter *positive observation threshold* saja.

4.2. Hasil Pengujian dan Analisis Skenario Dua

Model proses yang digunakan merupakan hasil dari process mining dengan algoritma heuristic miner dengan setelan parameter yang digunakan yaitu positive observation threshold 1000, relative to the best 0.05, dan nilai dependency threshold pada setelan default yaitu 0.9. Model proses ini merupakan model proses terbaik yang dihasilkan dari pengujian skenario satu

4.3. Hasil Pengujian dan Analisis Skenario Tiga

Berikut ini adalah tabel relasi antar proses yang memiliki waiting time tinggi atau terjadi bottleneck pada token yang menghubungkan proses-proses tersebut

Tabel 4.4 : Identifikasi Bottleneck antar proses

waiting time (hours) / relations	average	min	max
assignment-closed	19.20	0.0	3869.90
assignment-communication with customers	6.5363	0.0	102.03
pending vendor-closed	31.19	0.0	212.34
update from customers-closed	12.99	0.02	44.91
update-closed	33.16	0.0	2181.91
reassignment-closed	33.95	0.0	3944.47
operator update-closed	33.16	0.0	2753.10
resolved-closed	12.74	0.0	934.49
status change-closed	20.81	0.0	481.91

4.4. Rekomendasi

Berdasarkan hasil dari conformance checker dan performance analysis yang telah dilakukan, terdapat beberapa rekomendasi yang dapat dijadikan bahan pertimbangan oleh Rabobank Group ICT untuk meningkatkan kualitas dari proses bisnis dan dalam hal ini sekaligus juga meningkatkan kualitas pelayanan terhadap customer. Beberapa rekomendasi tersebut adalah:

1. Dengan membandingkan model proses terbaik yang didapatkan dari hasil pengujian dan model proses yang menampilkan keseluruhan proses (gambar 4.1), dapat dilihat bahwa ada koneksi antar proses yang banyak berkurang. Sehingga dapat direkomendasikan bahwa perlu adanya pengurangan koneksi antar proses secara nyata. Hal ini dilakukan untuk mengurangi timbulnya variasi rangkaian aktifitas yang dapat menjadi outlier. Sebagai contoh, pada model proses yang menampilkan keseluruhan proses terdapat koneksi antara proses Description Update dengan Caused by CI, Closed, dan Assignment. Sedangkan pada model proses terbaik hasil pengujian hanya terdapat koneksi antara Description Update dan Assignment. Hal ini menunjukkan terjadinya pengurangan koneksi antar proses, seperti juga terjadi pada proses-proses lainnya.

2. Perlu dilakukan mapping yang benar-benar baik terhadap human resources yang ada pada Rabobank Group ICT. Hal ini untuk menghindari adanya team yang overwork dan juga workless, sehingga dapat menyebabkan terjadinya penumpukan antrian proses yang harus dikerjakan dan menghasilkan bottleneck

seperti pada analisis subbab sebelumnya. Setelah pembagian kerja yang tepat, diharapkan fenomena bottleneck tidak akan terulang lagi pada proses-proses dari Rabobank Group ICT di masa mendatang.

5. Kesimpulan dan Saran

5.1. Kesimpulan

Berdasarkan hasil penelitian Tugas Akhir ini, ada beberapa kesimpulan yang didapatkan oleh penulis, diantaranya sebagai berikut:

1. Algoritma heuristic miner dapat diterapkan pada event log proses incident activity Rabobank Group ICT.

2. Algoritma heuristic miner dapat memodelkan event log Rabobank Group ICT ke dalam model proses dengan cukup baik, dilihat dari nilai fitness yang mencapai 0.85.

3. Model proses yang ideal didapatkan dengan setting parameter positive observation threshold 1000, relative to the best 0.05, dan nilai dependency threshold 0.9. Dengan parameter yang memiliki pengaruh terbesar adalah positive observation threshold. Hal ini dikarenakan parameter tersebut memiliki kemampuan untuk menyaring data dengan frekuensi trace sangat kecil untuk tidak diproses dalam process mining.

4. Terdapat beberapa titik terjadinya pemampatan proses atau bottleneck setelah dilakukan performance analysis pada model proses. Hal ini dapat dikurangi dengan menerapkan rekomendasi secara tepat.

5.2. Saran

Adapun saran untuk pengembangan yang berkaitan dengan penelitian ini adalah:

1. Perlu dilakukan penelitian dengan metode atau algoritma berbeda, dan dilakukan perbandingan terhadap hasil dari process mining dengan algoritma heuristic miner ini.

2. Penelitian ini berdasarkan data event log dari Rabobank Group ICT, sehingga setiap saran pada bagian rekomendasidapat dijadikan bahan pertimbangan untuk meningkatkan kualitas layanan

3. Penentuan parameter pada parameter positive observation threshold, relative to the best, dan nilai dependency threshold sebaiknya ditentukan secara otomatis pada pengembangan framework ini di masa yang akan datang. Metode pencarian model proses yang ada saat ini hanya dengan melakukan percobaan setting nilai parameter secara manual sehingga memakan waktu process mining yang cukup lama.

DAFTAR PUSTAKA

[1] Weijters, M. Der Aalst, Van, de Medeiros, Alves. (2008). Process Mining with Heuristic Miner Algorithm

[2] Der Aalst, Van. Process Mining – Discovery, Conformance and Enhancement of Business Process. New York:Springer, 2011

[3] Gunther, Christian W.(2009). Process Mining in Flexible Environments. Eidhohen University of Technology. Netherlands.

[4] der Aalst, Van. Verbeek, HMW.(2012).Process Discovery and Conformance Checking Using Passages.

[5] der Aalst, Van. Weijters. and Maruster, L.(2004).Workflow Mining: Discovering process models from event logs.

[6] der Aalst, Van. Adriansyah, Arya. and Van Dongen, Boudewijn.(2011).Causal Nets: A Modeling Language Tailored towards Process Discovery

[7] Jans, Mieke. Alles, Michael. and Vasarhely, Miklos.(2011).Process Mining of Event logs in Internal Auditing: A Case Study.

[8] Rozinat, A. and W.M.P. van der Aalst (2006). Decision Mining in Business Process : Eindhoven, Netherlands.

[9] Li, Chen. Reichert, Manfred. and Wombacher, Andreas .(2010).A Heuristic Approach for Discovering Reference Models by Mining Process Model Variants

[10] W.M.P. van der Aalst, H.A. Reijers, A.J.M.M. Weijters, B.F. van Dongen, and friends. Business Process mining : An Industrial Application, Eidhohen University of Technology. Netherlands.

[11] R.S.Mans, M.H Schonenberg, M.Song, W.M.P van der Aalst. Process mining in Healthcare a case study. Eidhohen University of Technology. Netherlands

[12] http://www.win.tue.nl/bpi/2014/challenge diakses pada 20 Desember 2014.

[13] http://www.processmining.org diakses pada 25 Januari 2015.

[14] http://www.promtools.org/pmtv/ diakses pada 25 Januari 2015.