Penggunaan Prinsip Apriori untuk *Outlier Cleaning* pada *Process Mining* dengan Algoritma α

Zainiyah Rizkita Arief[1], Imelda Atastina[2], Angelina Prima K[3]
[1]Program Studi Magister Teknik dan Manajemen Industri,
Fakultas Teknologi Industri, Institut Teknologi Bandung.
E-mail : [1]zrizkita.arief@gmail.com

Abstrak

Process mining telah digunakan untuk membantu dalam penyelesaian masalah pada kehidupan sehari-hari. Discovery adalah salah satu tipe process mining yang membentuk model proses dari event log yang ada. Algortima α adalah salah satu algoritma yang dapat digunakan untuk melakukan discovery process. Algoritma α melakukan pengurutan proses yang terjadi pada event log dan membandingkan semua keterurutan tersebut. Maka dari itu, akan didapatkan informasi proses mana yang merupakan kausalitas dan proses mana yang bersifat paralel. Pada kenyataannya, kesederhanaan konsep ini memberikan masalah pada penerapannya pada data real life. Data real-life memiliki keragaman yang tinggi sehingga mengandung banyak outlier yang akan menjadi data yang mengganggu. Maka dari itu, data outlier tersebut perlu dihilangkan. Salah satu metode yang dapat menghilangkan outlier pada data mining, adalah dengan mengadopsi prinsip apriori. Sehingga kasus dan hubungan aktivitas yang tidak memenuhi syarat batas yang telah ditentukan tidak merusak model secara keseluruhan. Pengujian dilakukan pada data registrasi yang tersimpan dalam event log pada sistem informasi.

***Kata kunci**: registrasi mahasiswa, process mining, discovery, algoritma α, apriori*

1. Pendahuluan

Proses registrasi merupakan proses yang selalu dilaksanakan pada awal setiap semester. Pada saat itu, mahasiswa membuat rencana studi selama satu semester ke depan. Proses registrasi yang tidak dilaksanakan dengan serius akan berakibat kurang baik terhadap mahasiswa. Mahasiswa tersebut tidak akan maksimal menjalankan studinya satu semester ke depan. Maka dari itu, proses registrasi merupakan proses yang penting.

Institut Teknologi Telkom (IT Telkom) telah melakukan proses registrasi sejak IT Telkom berdiri. IT Telkom memiliki mekanisme standar yang harus dijalani setiap mahasiswa dan institusi. Namun, mekanisme tersebut tidak selalu dijalankan sesuai dengan aturan yang berlaku. Maka dari itu, perlu dilakukan permodelan terhadap proses yang sebenarnya terjadi. Institusi dapat melakukan evaluasi terhadap proses registrasi berdasarkan model tersebut.

Algoritma α akan menentukan hubungan dua aktivitas kausalitas antara satu aktivitas dengan aktivitas lainnya. Misalnya, suatu *event log* memiliki dua kasus dengan urutan {a, b, c, d} dan {a, b, d, c}. Algoritma α akan menentukan bahwa a dan b memiliki hubungan kausalitas. Selanjutnya, algoritma α akan mnentukan b dengan c atau d tidak miliki hubungan kausalitas. Hubungan tersebut ditentukan tanpa mempedulikan berapa kali c atau d saling bertukar urutan. Kesederhanaan konsep tersebut menyebabkan algoritma α memiliki keterbatasan. Keterbatasan tersebut meliputi kelemahan algoritma α dalam penerapannya pada data yang mengandung *noise, incompleteness* dan hubungan antar transisi yang kompleks (van der Aaslt, 2011).

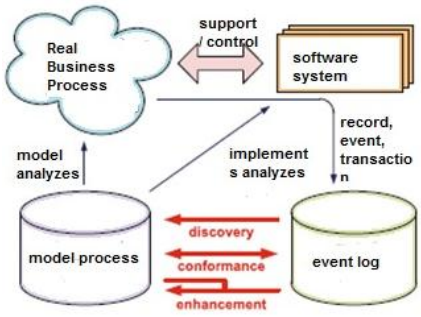
Algoritma α telah digunakan untuk melakukan *discovery* model proses pada kasus *student registrastion* sebuah universitas di Thailand. Data yang digunakan merupakan data *student registration event log* yang telah dilakukan *data preprocessing*. Pada penelitian tersebut tidak menjelaskan berapa akurasi dari model proses yang dihasilkan. Akan tetapi, diterangkan bahwa algoritma α dapat menghasilkan model yang mampu menunjukkan semua hubungan antar aktivitas yang ada (Weerapong 2012). Oleh karena itu, proses registrasi merupakan proses yang cukup sederhana sehingga dapat diproses menggunakan algortima α.

1. Process Mining

Van der Aalst menyatakan bahwa perkembangan sistem informasi dari hari ke hari semakin pesat. Organisasi akan memilih untuk menyimpan datanya dalam bentuk digital. Data yang disimpan bisa mencapai satuan *terrabyte*. Data yang banyak menyebabkan organisasi kesulitan untuk mendapatkan informasi yang terdapat dalam data

tersebut. *Process mining* memiliki keterkaitan yang cukup erat dengan *data mining*. Buku yang sama juga mengatakan bahwa *process mining* adalah *data mining* yang diterapkan pada *event log*. Hal ini dilakukan untuk mengetahui pola aktivitas yang terjadi pada suatu proses. Perbedaan mendasar antara *process mining* dan *data mining* jenis asosiasi terletak pada perhatian *process mining* terhadap urutan kejadian (van der Aaslt, 2011).

Process mining memiliki tiga tipe, yaitu *discovery*, *conformance* dan *enhancement*. *Discovery* akan membentuk sebuah model proses dari *event log* yang ada. *Conformance* akan membandingkan model proses dengan *event log* yang ada. Sementara *enhancement* dapat memperbaiki model proses yang sudah ada dengan membandingkannya dengan *event log* yang ada.



Gambar 1. Gambaran Umum Process Mining

2. Algoritma α

Algoritma α adalah salah satu algoritma pertama yang memadai menangani konkurensi. Algoritma ini dapat men-*generate* model dari proses yang memiliki terjadi dua aktivitas yang terjadi bersamaan. Algoritma α ini sangat sederhana. Algoritma α hanya memeriksa hubungan antar dua aktivitas.
Terdapat empat macam hubungan, yaitu

- ✓ follow ($>_L$), dimana *event* a $>_L$ b, jika dan hanya jika t_i=a dan t_{i+1}=b,
- ✓ causal ($\rightarrow_L$), dimana *event* a $\rightarrow_L$ b, jika dan hanya jika a $>_L$ b dan b $\not>_L$ a,
- ✓ paralel ($\|_L$), dimana *event* a $\#_L$ b, jika dan hanya jika a $>_L$ b dan b $>_L$ a, dan
- ✓ unrelated ($\#_L$), dimana e*vent* a $\|_L$ b, jika dan hanya jika a $\not>_L$ b dan b $\not>_L$ a.

Secara matematis algoritma α dituliskan dalam rumus-rumus berikut,

1) $T_L = \{t \in T \mid \exists_{o} \underline{C}_L\ t \in \sigma\}$
2) $T_I = \{t \in T \mid \exists_{o} \underline{C}_L\ t = first(\sigma)\}$
3) $T_O = \{t \in T \mid \exists_{o} \underline{C}_L\ t = last(\sigma)\}$
4) $X_L = \{(A,B) \mid A \subseteq T_L \wedge A \neq \emptyset \wedge B \subseteq TL \wedge A \neq \emptyset \wedge \forall_{a \in A} \forall_{b \in B} a \rightarrow_L b \wedge \forall_{a1,a2 \in A} a_1 \#_L a_2 \wedge \forall_{b1,b2 \in B} b_1 \#_L b_2$
5) $Y_L = \{(A,B) \in X_L \mid \forall_{(A',B') \subseteq XL}, A \subseteq A', B \subseteq B' \rightarrow (A,B) = (A',B')\}$
6) $P_L = \{p_{(A,B)} \mid (A,B) \in Y_L\} \cup \{i_L, o_L\}$
7) $FL = \{(a, p_{(A,B)}) \mid (A,B) \in Y_L \wedge a \in A\} \cup \{(p_{(A,B)},b) \mid (A,B) \in Y_L \wedge b \in B\} \cup \{(i_L, t) \mid t \in T_i\} \cup \{(t, o_L) \mid t \in T_o\}$

$\alpha(L) = \{P_L, T_L, F_L\}$

(Weerapong, 2012)
Algoritma α merupakan metode yang sangat sederhana sehingga memiliki keterbatasan dalam penerapannya dalam kehidupan sehari-hari. Keterbatasan algoritma ini adalah sebagai berikut,

- tidak bisa menangani data dengan *noise*,
- kemungkinan aktivitas yang *incomplete* dan
- data dengan hubungan antar transisi yang kompleks.

(van der Aaslt, 2011).

3. Prinsip Apriori

Kelemahan yang dimiliki oleh algoritma α menyebabkan data yang dimasukkan ke dalam algoritma harus sudah bersih dari noise. *Outlier* bisa pula memiliki sifat-sifat noise yang mengganggu data. *Outlier* sulit dibedakan dengan data yang bukan *outlier*. Bahkan, untuk menentukan *outlier* diperlukan metode tersendiri. *Infrequent*

pattern adalah salah satu representasi dari *outlier*. Salah satu metode yang efisien untuk menghilangkan *infrequent pattern* adalah metode apriori (Nadimi-Shahraki,2009).

Tujuan dari penggunaan prinsip apriori ini adalah menghasilkan *rule association* yang optimal (Tan, 2006). Hal ini dilakukan dengan memangkas *rule* yang tergambar dalam *infrequent itemset*. *Infrequent itemset* ini dapat dilihat dari nilai *support* dan *confidence* yang rendah. Maka dari itu, prinsip apriori ini juga bisa digunakan untuk melakukan *cleaning outlier* (Nadimi-Shahraki,2009).

Langkah-langkah yang dilakukan pada implementasi prinsip apriori adalah sebagai berikut,
1) *Generate frequent itemset* sampai jumlah maksimal *itemset* yang diinginkan.
2) *Generate rule* dari *frequent itemset* tersebut.
3) Hitung *support* dan *confidence* setiap *rule* dari *frequent itemset*.
4) Hilangkan *frequent itemset* dengan nilai *support* dan nilai *confidence* dibawah *threshold* yang telah ditentukan (Han, 2001; Tan, 2006).

4. Pengolahan Data

Data yang tercatat dalam event log proses registrasi adalah aktivitas siap ACC, ACC dan cetak KSM. Selain itu juga tercatat aktivitas tersebut dilakukan oleh mahasiswa atau institusi. Untuk memudahkan mengolahan data dilakukan transformasi data berupa perubahan data aktivitas sebagai berikut

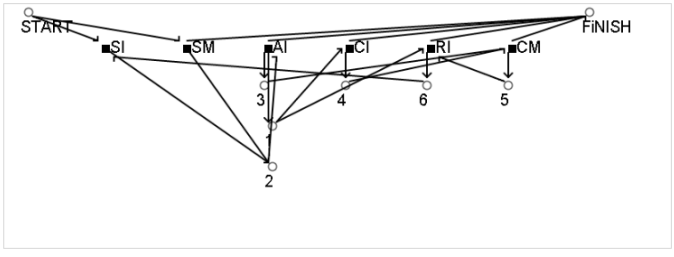
Tabel 1: Singkatan Data

Aktivitas	Singkatan
Siap ACC dilakukan oleh mahasiswa	SM
ACC dilakukan oleh mahasiswa	AM
Cetak KSM dilakukan oleh mahasiwa	CM
Siap ACC dilakukan oleh institusi	SI
ACC dilakukan oleh institusi	AI
Cetak KSM dilakukan oleh institusi	CI

Tabel 2 : Contoh Data Training

NIM	Tanggal	Jam	Aktivitas
BGfffdcd13f	2013-02-05	9:19:23	SM
BGfffdcd13f	2013-02-05	16:36:12	AI
BGfffdcd13f	2013-02-06	9:33:59	CM
BGfffdcd13f	2013-02-06	9:34:59	CM
…			
BG139871	2013-02-06	15:31:39	SM
BG139871	2013-02-06	17:31:50	RI
BG139871	2013-02-07	14:47:27	SM
BG139871	2013-02-07	20:02:04	AI
BG139871	2013-02-07	22:23:43	CM

Event log tersebut diolah dengan algoritma alpha. Sebelum dioleh oleh algoritma alpha, data outlier telah dihilangkan oleh prinsip apriori. Berikut adalah contoh model proses yang dihasilkan.



DAFTAR PLACE

id	dari		ke	
1	AI		CI	RI
2	SI	SM	AI	
3	AI		CM	
4	CI		CM	
5	CM		RI	
6	RI		SI	

Gambar 2 : Ilustrasi Model Proses yang Dihasilkan

5. Pengujian

Pengujian dilakukan untuk mengetahui berapa banyak outlier yang harus disingkirkan untuk mendapatkan model proses yang optimal. Maka dari itu, pengujian dilakukan dengan mengubah nilai minimal support dan confidence untuk selanjutnya model proses dilakukan penghitungan perfomansi. Perhitungan performasi dihitung dengan F-measure, precision dan recall.

Terdapat dua jenis data yaitu data aktual dan data prediktif. Data aktual merupakan data kausalitas dari data latih atau *data testing* yang belum diolah sedangkan data prediktif adalah data kausalitas urut menurut model yang telah dihasilkan. Selain itu, aktivitas pun terdiri dari dua jenis aktivitas yaitu, aktivitas positif dan aktivitas negatif. Aktivitas positif adalah aktivitas yang benar apabila ada pada posisi tersebut berdasarkan acuannya. Maka, aktivitas negatif adalah aktivitas yang tidak benar apabila ada di posisi tersebut berdasarkan acuannya. Aktivitas negatif tersebut didapatkan dari *generate* ANE (*artificial negative event*).

Tabel 3 : Pengelompokan Data

	ACTUAL POSITIVE	ACTUAL NEGATIVE
PREDICTIVE POSITIVE	TRUE POSITIVE (TP)	FALSE POSITIVE (FP)
PREDICTIVE NEGATIVE	FALSE NEGATIVE (FN)	TRUE NEGATVE (TN)

Perhitungan performansi model proses dapat dilakukan dengan menghitung *F-Measure* dengan cara berikut,

$$\text{FMeasure} = \frac{2 * precision * recall}{precision + recall}$$

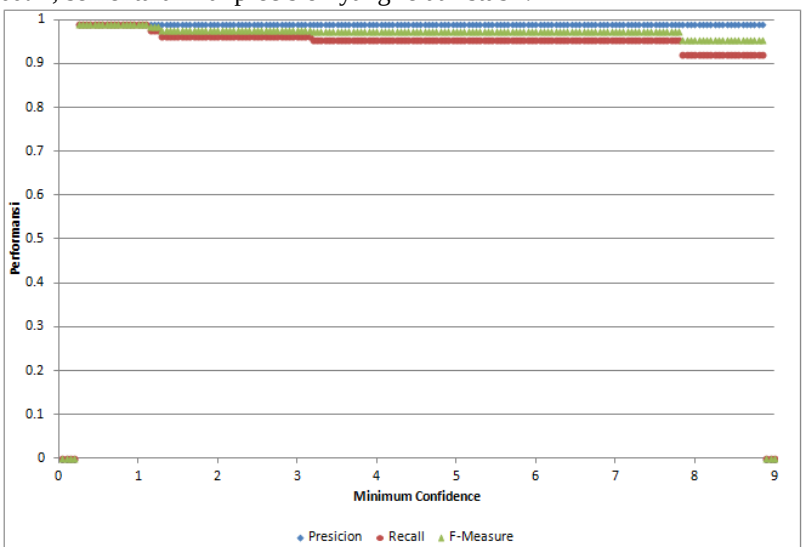
$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

(De Weerdt).

6. Hasil Pengujian dan Analisis

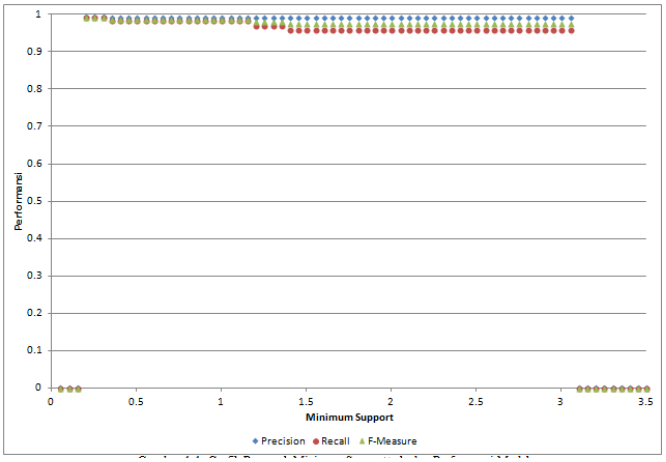
Nilai *F-Measure* menurun seiring bertambahnya nilai minimum *confidence*. Penurunan ini disebabkan oleh menurunnya nilai *recall*, sementara nilai precision yang relatif stabil.



Gambar 3 : hasil pengujian pengubahan nilai confidence

Nilai *minimum confidence* yang semakin besar akan menyebabkan hilangnya hubungan yang seharusnya digambarkan pada model. Apabila nilainya terlalu tinggi tidak akan ada hubungan yang dapat digambarkan. Namun, nilai *confidence* yang terlalu rendah juga dapat menyebabkan kompleksitas hubungan antar transisi yang juga berakibat hilangnya hubungan sehingga menyebabkan adanya transisi yang tidak dapat mencapai *finish*.

Nilai *F-Measure* menurun seiring bertambahnya nilai minimum *support*. Penurunan ini disebabkan oleh menurunnya nilai *recall*, sementara nilai precision yang relatif stabil.



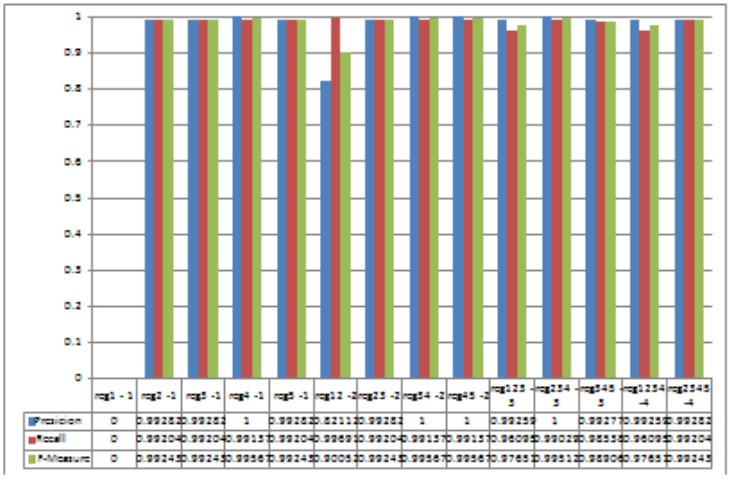
Gambar 4 : Hasil pengujian pengubahan nilai support

Nilai *minimum support* yang semakin besar akan menyebabkan hilangnya hubungan yang seharusnya digambarkan pada model. Apabila nilainya terlalu tinggi model tidak dapat digambarkan. Hal ini disebabkan *data training* yang mengalami *incompleteness*. Namun, nilai minimum *support* yang terlalu rendah akan meningkatkan keragaman kasus yang menyebabkan hubungan antar transisi menjadi kompleks sehingga tidak ada model yang dapat dihasilkan.

Nilai maksimum *F-Measure* yang dicapai pada setiap kelompok data yang berbeda jumlah ini tidak jauh berbeda. Hal ini disebabkan oleh kasus yang terjadi pada setiap semester hampir serupa sehingga tidak berpengaruh terhadap keragaman data. Maka dari itu, data yang berjumlah satu semester pun tidak mengalami ***incompleteness***.

Tabel 3 : Jumlah data

Data	Jumlah Kasus
reg1	1994
reg2	6734
reg3	7909
reg4	7163
reg5	8636
reg12	8728
reg23	14643
reg34	15072
reg45	15799
reg123	16637
reg234	21806
reg345	23708
reg1234	23800
reg2345	30442
reg12345	32436



Gambar 5 : Hasil pengujian pengubahan jumlah data training

Oleh karena itu, model yang dihasilkan oleh *data training* yang terdiri dari satu semester pun memiliki kemampuan yang tinggi dalam me-*replay* data yang berjumlah empat semester. Kesamaan model yang dihasilkan oleh data yang lebih sedikit juga disebabkan oleh sifat algoritma α yang tidak memperdulikan berapa kali suatu kasus terjadi hubungan relasi yang dihasilkan akan tetap sama.

Namun, terdapat beberapa pengecualian pada model yang dihasilkan oleh kelompok data reg1 dan kelompok data yang mengandung data reg1. Model yang dihasilkan oleh semua kelompok data yang mengandung reg1 tidak memenuhi aturan. *F-Measure* model yang dihasilkan oleh kelompok data yang mengandung kelompok data reg1 memiliki nilai yang lebih rendah daripada kelompok data lainnya dengan jumlah data yang sama. Maka dari itu, hal tersebut semakin menunjukkan bahwa data reg1 adalah data yang mengalami ***incompleteness***.

7. Kesimpulan

Berdasarkan analisis yang telah dilakukan, terdapat beberapa hal penting yang dapat disimpulkan. Pertama, pada kelompok data yang berbeda akan memiliki nilai *minimum support* dan nilai *minimum confidence* optimal yang berbeda. Parameter nilai *minimum confidence* yang tinggi, akan menurunkan performansi model. Parameter nilai *minimum support* meningkat akan menyebabkan performansi turun. Namun, ketika nilai *minimum support* dan *minimum confidence* yang terlalu rendah, performansi model yang dihasilkan akan bernilai 0. Selain itu, algortima α akan menghasilkan model dengan performansi yang tinggi pada *data training* yang ***complete***, walaupun jumlah *data training* yang digunakan sedikit.

8. Saran

Untuk penelitian selanjutnya, akan lebih baik apabila melakukan penerapan algoritma α+ pada proses registrasi agar hubungan a-b-a dapat digambarkan dalam model. *Running time* evaluasi model sangat kompleks sehingga dibutuhkan metode lain yang lebih efektif dari segi kompleksitas waktu. Selain itu, akan lebih baik apabila pencarian nilai *minimum support* dan nilai *minimum confidence* optimal dilakukan dengan metode *evolutionary computation*, khususnya dengan menggunakan *genetic algorithm* atau *evolution strategies*. Hal ini dilakukan agar proses pencarian menjadi lebih cepat.

9. Referensi

[1] Weerapong, Sawitree, Parham Porouhan, Wichian Premchaiswadi. 2012. Process Mining Using α-Algorithm as a Tool (A case study of Student Registration). *IEEE Tenth International Conference on ICT and Knowledge Engineering* : 213 – 220.

[2] van der Aalst, Wil M.P. 2011. *Process Mining : Discovery, Confermance and Enhancement of Bussiness Processes*. New York : Springer.

[3] Nadimi-Shahraki, M.H, Norwati Mustapha, Md Nasir B Sulaiman, Ali B Mamat. 2009. Efficient Candidacy Reduction For Frequent Pattern Mining. *International Journal of Computer Science and Information Security 2009, 6 : 230-237.*

[4] Tan, Pang-Ning, Micheal Steinbach, Vipin Kumar. 2006. *Introduction to Data Mining*. Boston : Pearson Education.

[5] Han, Jiawei, Micheline Kamber. 2001. *Data Mining : Concept and Technique*. San Fransisco : Morgan Kaufmann Publisherss.

[6] De Weerdt, Jochen, Manu De Backer, Jan Vanthienen, and Bart Baesens. A Robust F-Measure for Evaluating Discovered Process Models. *Computational Intelligence and Data Mining (CIDM), 2011 IEEE Symposium.*

[7] Buijs, J.C.A.M, van Dongen, W.M.P van der Aaslt. 2012. On the Role of Fitness, Precision, Generalization and Simplicity in Process Discovery. *Lecture Notes in Computer Science*, 7565 : 305-322.

[8] van der Aalst, Wil, Arya Ardiansyah, Boudenwijn van Dongen. Replaying Histroy on Process Model for Confermance Checking and Performance Analysis. 2012. *WIREs Data Mining Knowledge Discovery 2012, 2* : 182-192.

[9] Rozinat, A., W.M.P. van der Aaslt. 2008. Confermance Checking of Processes Based on Monitoring Real Behaviour. *Information Systems*, 33(1) : 64-95.