

PREDIKSI POLA KECELAKAAN KERJA PADA PERUSAHAAN NON EKSTRAKTIF MENGGUNAKAN ALGORITMA DECISION TREE: C4.5 DAN C5.0

Yolanda Rizkita Putri, Imam Mukhlash dan Nurul Hidayat
Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam,
Institut Teknologi Sepuluh Nopember (ITS)
Jl. Arief Rahman Hakim, Surabaya 60111 Indonesia
e-mail: ITS.imamm@matematika.its.ac.id

Abstrak— Kecelakaan kerja pada perusahaan sangat mungkin terjadi. Kemungkinan tersebut dapat muncul dari berbagai aspek, mulai dari aspek karyawan atau sistem perusahaan yang diterapkan. Salah satu tindakan preventif yang dapat dilakukan adalah memprediksi pola kecelakaan dari kecelakaan kerja yang terjadi. *Data mining* merupakan metode yang dapat memprediksi pola data menggunakan data historis. Salah satu metode yang dapat dipilih adalah klasifikasi. Klasifikasi bekerja untuk membentuk pola dari keterhubungan atribut yang ada pada data. Dalam Tugas Akhir ini, metode klasifikasi yang digunakan adalah *decision tree*. Pada *decision tree*, atribut pada data dapat membentuk suatu *rules* yang menggambarkan kemungkinan terjadinya kejadian. Saat ini telah berkembang algoritma *decision tree* yang dapat digunakan untuk memprediksi pola kecelakaan kerja, yaitu C4.5 dan C5.0. Hasil Algoritma C4.5 dan C5.0 berupa *rules* akan digunakan untuk proses klasifikasi. Selanjutnya proses testing digunakan untuk mengetahui performansi dari kedua algoritma. Performansi tersebut diukur dengan dua parameter yaitu akurasi dan waktu komputasi. Hasil pengujian menunjukkan akurasi Algoritma C4.5 lebih baik daripada Algoritma C5.0. Sedangkan waktu komputasi kedua algoritma hampir sama. Jadi dapat disimpulkan bahwa Algoritma C4.5 lebih baik dari Algoritma C5.0.

Kata Kunci— Kecelakaan kerja, Data mining, Klasifikasi, C4.5, C5.0

I. PENDAHULUAN

Kegiatan penggunaan teknologi dalam perusahaan (industri) di Indonesia beberapa tahun ini mengalami kemajuan. Aturan kerja yang berubah akan membawa dampak langsung bagi karyawan yang melakukan pekerjaan langsung dengan teknologi yang digunakan. Dengan demikian, resiko kerja yang dimiliki oleh perusahaan akan meningkat dan akhirnya akan menimbulkan insiden kerja atau kecelakaan kerja.

Istilah kecelakaan kerja menjadi umum bagi kalangan industri Indonesia saat ini. Menurut Coling David A (1990) kecelakaan kerja adalah setiap kejadian tak terencana dan tidak terkontrol yang disebabkan oleh manusia, faktor situasi atau lingkungan atau merupakan kombinasi dari faktor-faktor tersebut yang mengganggu proses kerja yang mungkin berakibat atau tidak berakibat cedera, penyakit, kematian, kerusakan harta benda atau kejadian lain yang tidak diharapkan tetapi berpotensi untuk terjadinya hal tersebut [8].

Tingkat kecelakaan kerja di Indonesia memang cukup tinggi dibanding dengan negara di Eropa dan Asia lainnya. Pada tahun 2010 tercatat tingkat kecelakaan kerja di Indonesia mencapai 98.711 kasus. Namun pada tahun 2011, angka tersebut mengalami penurunan, yaitu 86.386 kasus. Hal terpenting dalam masalah kecelakaan kerja ialah bagaimana cara meminimalisir tingkat kecelakaan kerja yang terjadi pada suatu perusahaan. Untuk itu, tindakan preventif dan pengetahuan tentang aturan kerja sangat penting bagi karyawan perusahaan. Salah satu hal penting dalam menentukan tindakan preventif ini dengan mengetahui pola kecelakaan kerja yang telah terjadi sebelumnya. Salah satu cara yang dapat dilakukan untuk membuat pola kecelakaan kerja tersebut dapat dilakukan dengan *data mining*.

Pada penelitian ini akan dibangun suatu perangkat lunak yang mampu menggali pola pada data menggunakan metode klasifikasi. Diharapkan keluaran yang dihasilkan dapat membentuk pola kecelakaan kerja sebagai referensi tindakan preventif yang digunakan oleh perusahaan. Selain itu, dapat mengetahui perbandingan tingkat keakuratan dari algoritma yang digunakan, yaitu C4.5 dan C5.0.

II. DASAR TEORI

2.1 Data Mining

Data mining merupakan proses ekstraksi pola yang penting dari data dalam jumlah besar. *Data mining* merupakan salah satu langkah dalam proses *Knowledge Discovery in Database* (KDD) untuk menemukan pola yang bermanfaat. *Data mining* juga didefinisikan sebagai suatu proses yang menggunakan berbagai perangkat analisis data untuk menemukan pola dan relasi data agar dapat digunakan untuk membuat prediksi dengan tepat [4].

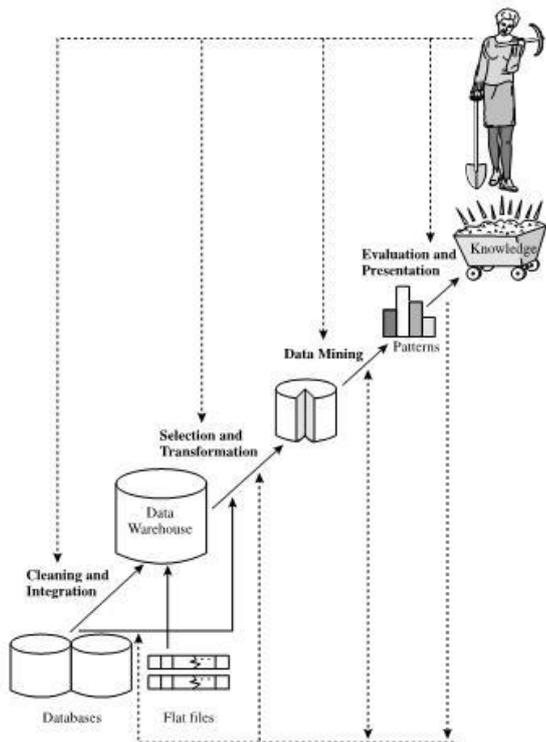
Beberapa karakteristik umum yang dimiliki data yang akan dianalisis adalah sebagai berikut [4]:

1. Ukuran data besar .
2. Data tidak lengkap, sehingga perlu proses *cleaning*.
3. Struktur data kompleks.
4. Merupakan kumpulan data yang heterogen.

2.2 KDD (*Knowledge Discovery in Database*)

KDD merupakan sebuah proses yang terdiri dari serangkaian proses iteratif yang terurut dan *data mining* merupakan salah satu langkah dalam KDD [4].

Tahapan-tahapan proses KDD secara berurut dapat dilihat pada Gambar 2.1 berikut ini:



Gambar 2.1 Proses Knowledge Discovery in Database

2.3 Klasifikasi

Klasifikasi adalah proses penemuan model (atau fungsi) yang menggambarkan dan membedakan kelas data atau konsep yang bertujuan agar bisa digunakan untuk memprediksi kelas dari objek yang label kelasnya tidak diketahui [4].

Klasifikasi data terdiri dari 2 langkah proses. Pertama adalah *learning* (fase *training*), dimana algoritma klasifikasi dibuat untuk menganalisa data *training* lalu direpresentasikan dalam bentuk *rules* (pola kecelakaan kerja). Proses kedua adalah klasifikasi, dimana data testing digunakan untuk memperkirakan akurasi dari *rules* [6].

2.4 Decision Tree

Decision tree adalah salah satu metode klasifikasi yang paling populer karena mudah untuk diinterpretasi oleh manusia [12]. *Decision tree* adalah model prediksi menggunakan struktur pohon atau struktur berhirarki. Konsep dari *decision tree* adalah mengubah data menjadi pohon keputusan atau aturan-aturan keputusan.

Metode ini digunakan untuk memprediksi nilai diskrit dari fungsi target, yang mana fungsi pembelajaran dipresentasikan oleh sebuah *decision tree* [12]. *Decision tree* merupakan himpunan IF...THEN.

2.5 Algoritma C4.5

Algoritma C4.5 merupakan algoritma yang digunakan untuk membentuk pohon keputusan [5][10]. Pohon keputusan merupakan metode klasifikasi dan prediksi yang sangat kuat dan terkenal.

Secara ringkas, langkah kerja Algoritma C4.5 dapat digambarkan sebagai berikut :

- Hitung setiap kemunculan kejadian pada tiap *value* dari atribut yang digunakan terhadap *value* dari atribut target.
- Hitung *entropy* total dan *entropy* untuk setiap *value* atribut. *Entropy* total berasal dari atribut target. Perhitungan *entropy* ini akan dibahas lebih lanjut pada subbab 2.7.
- Hitung *information gain* untuk tiap atribut. *Information gain* inilah yang akan digunakan untuk membuat *tree*. Pada iterasi pertama, atribut yang mempunyai nilai *information gain* tertinggi akan menjadi *root tree*. Perhitungan *information gain* ini akan dibahas lebih lanjut pada subbab 2.7.
- Bentuk simpul pertama yang berisi atribut yang memiliki *information gain* tertinggi.
- Ulangi perhitungan *information gain* sampai semua data telah termasuk dalam kelas yang sama. Atribut yang telah dipilih tidak diikuti lagi dalam perhitungan nilai *information gain* pada iterasi berikutnya.

2.6 Algoritma C5.0

Algoritma C5.0 adalah salah satu algoritma *data mining* yang khususnya diterapkan pada algoritma *decision tree* [2]. C5.0 merupakan penyempurnaan algoritma sebelumnya yang dibentuk oleh Ross Quinlan pada tahun 1987, yaitu ID3 dan C4.5. Dalam algoritma ini pemilihan atribut diproses menggunakan *gain ratio*. C5.0 menghasilkan *tree* dengan jumlah cabang per *node* bervariasi [2].

Langkah kerja pembangunan *tree* pada Algoritma C5.0 mirip dengan pembangunan Algoritma C4.5. Kemiripan tersebut meliputi perhitungan kemunculan keadian, perhitungan *entropy* dan *information gain* . Jika pada Algoritma C4.5 berhenti sampai perhitungan *information gain*, maka pada Algoritma C5.0 akan melanjutkannya dengan perhitungan *gain ratio* dengan menggunakan *information gain* dan *entropy* yang telah ada. Perhitungan *gain ratio* akan dibahas pada subbab 2.7.

2.7 Information Gain dan Gain Ratio

Ukuran *information gain* dan *gain ratio* digunakan untuk memilih atribut uji pada setiap *node* di dalam *tree*. Ukuran ini digunakan untuk memilih atribut atau *node* pada *tree*. Atribut dengan nilai *information gain* tertinggi akan terpilih sebagai *parent* bagi *node* selanjutnya pada Algoritma C4.5. Rumus yang digunakan untuk *information gain* adalah [6]:

$$Entropy(S) = - \sum_{i=1}^n p_i * \log_2 p_i \tag{1}$$

dengan :

S = himpunan kasus

n = jumlah partisi S

p_i = proporsi dari $|S_i|$ terhadap $|S|$

S adalah sebuah himpunan yang terdiri dari s data objek. Diketahui atribut *class* adalah m dimana mendefinisikan kelas-kelas di dalamnya, C_i (for $i= 1, \dots, m$), s_i adalah jumlah objek pada S dalam class C_i . Untuk mengklasifikasikan objek yang digunakan maka diperlukan informasi dengan menggunakan aturan seperti di atas. Dimana p_i adalah proporsi kelas dalam output seperti pada kelas C_i dan diestimasikan dengan s_i / s .

Atribut A memiliki nilai tertentu $\{a_1, a_2, \dots, a_v\}$. Atribut A dapat digunakan pada partisi S ke dalam v subset, $\{S_1, S_2, \dots, S_v\}$, dimana S_j berisi objek pada S yang bernilai a_j pada A. Jika A dipilih sebagai atribut tes (sebagai contoh atribut terbaik untuk split), maka subset ini akan berhubungan pada cabang dari *node* himpunan S. S_{ij} adalah jumlah objek pada class C_i dalam sebuah subset S_j .

Untuk mendapatkan nilai *information gain* selanjutnya digunakan persamaan dibawah ini :

$$Information\ Gain(S, A) = Entropy(S) - \sum_{i=1}^n \frac{|S_i|}{|S|} * Entropy(S_i) \quad (2)$$

dimana :

S = himpunan kasus

A = atribut

n = jumlah artisi atribut A

$|S_i|$ = jumlah kasus pada partisi ke-i

$|S|$ = jumlah kasus dalam S

Perhitungan *gain ratio* untuk Algoritma C5.0 akan berjalan setelah perhitungan *information gain* diatas dilakukan. Perhitungan *gain ratio* selanjutnya menggunakan persamaan dibawah ini :

$$GainRatio = \frac{Information\ Gain(S, A)}{\sum_{i=1}^n Entropy(S_i)} \quad (3)$$

Dengan adanya perhitungan *gain ratio* inilah yang menjadikan pembangunan *tree* pada C5.0 lebih ringkas dibanding *tree* pada Algoritma C4.5. Sehingga menyebabkan pola kecelakaan kerja yang dihasilkan lebih sedikit dibandingkan Algoritma C4.5.

Untuk menghitung keakuratan dari pola / *rules* yang dihasilkan oleh Algoritma diatas, maka digunakan persamaan dibawah ini:

$$Akurasi = \frac{Nilai\ Kecocokan}{Jumlah\ Seluruh\ Kejadian} * 100\% \quad (4)$$

2.8 Informasi Data

Atribut data yang akan dipakai dalam tugas akhir ini antara lain:

Tabel 1. Atribut Data Kecelakaan Kerja

Id Kejadian	Id kejadian kecelakaan kerja merupakan waktu kejadian, sekaligus kunci primer sebagai identitas kejadian.
Nama	Nama karyawan yang mengalami kecelakaan kerja.
Jenis Kelamin	Jenis kelamin karyawan yang mengalami kecelakaan kerja. Nilai dari atribut ini yaitu 1. Laki- laki (L) 2. Perempuan (P)
Umur	Umur karyawan saat mengalami kecelakaan kerja. Atribut ini akan dijadikan nilai kategorikal pada tahap transformasi, menjadi : 1. Muda 2. Tua

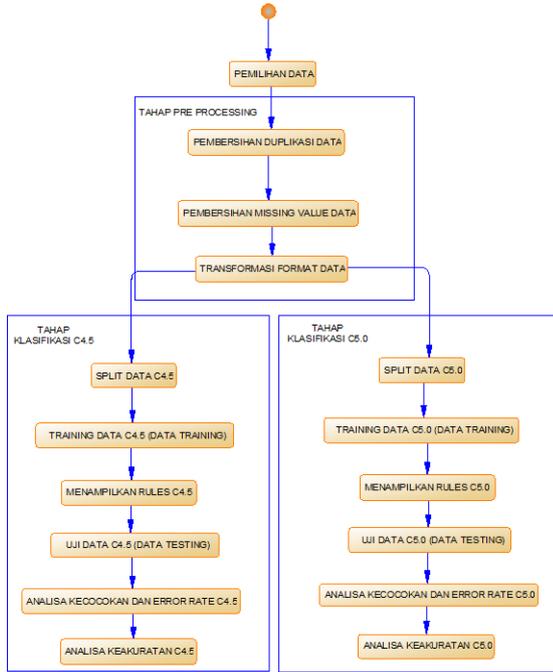
Kewarganegawaan	Asal negara karyawan yang mengalami kecelakaan kerja.
Tipe Pekerjaan	Departemen karyawan yang mengalami kecelakaan kerja. Nilai dari data ini yaitu : 1. GRD A 2. GRD B 3. GRD C
Waktu Perhari	Waktu per hari pada tiap departemen. Ada 2 nilai dalam atribut ini, yaitu : 1. 08:00:00,00 2. 09:00:00,00
Lama Bekerja	Waktu kerja karyawan selama di perusahaan. Nilai dalam atribut ini juga akan dikategorikan menjadi, 1. Junior 2. Senior
Status	Status karyawan saat ia menjadi bagian dari departemen, yaitu : 1. Regular 2. Fuji 3. Swabina 4. ISS 5. Kontraktor
Pelatihan	Riywayat pelatihan yang dimiliki oleh karyawan, yaitu YES atau NO
Kecelakaan Kerja	Kecelakaan kerja dibagi mejadi 3 jenis, yaitu : 1. On Work 2. Traffic On Commute 3. Traffic Inside Fact Atribut kecelakaan kerja juga akan menjadi atribut arget dalam proses <i>classification</i> .

Klasifikasi kecelakaan kerja diatas, berdasarkan data pada perusahaan. Definisi dari istilah tersebut yaitu :

- *On Work* : Kecelakaan kerja yang terjadi saat karyawan sedang dalam proses pekerjaan dan berada dalam area perusahaan
- *Traffic Inside Fact* : Kecelakaan yang terjadi selama di area perusahaan yang tidak termasuk *On Work*
- *Traffic On Commute* : Kecelakaan kerja yang terjadi saat karyawan berada di luar area perusahaan, contoh : waktu berangkat kerja, waktu pulang kerja, dinas luar.

III. PERANCANGAN SISTEM

3.1. Diagram Alir Sistem



Gambar 3.1. Diagram Alir Sistem

3.2. Gambaran Umum Sistem

Proses data mining berjalan saat pemilihan data dilakukan. Data kecelakaan kerja yang berada pada database akan dipanggil sebagai masukan utama. Setelah pemanggilan sukses, kemudian data akan di *pre-processing* untuk menghilangkan *noise* dan mengubah format pada atribut. Setelah data siap diolah terbentuk, tahap selanjutnya adalah pembangunan algoritma. Pembangunan C4.5 akan dilakukan terlebih dahulu menggunakan *information gain* sebagai tahap training. Setelah terbentuk pola kecelakaan kerja, tahap testing akan dilakukan untuk mengetahui keakuratan pada algoritma ini.

Berikut ini penjelasan dari beberapa tahap dalam sistem perangkat lunak ini:

a. Pre-processing

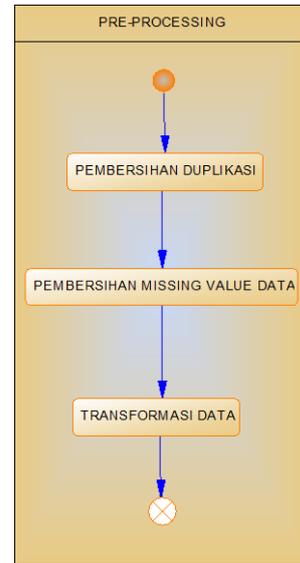
Awal dari proses ini, pemanggilan data dari database. Data yang ditampilkan merupakan data real yang kemungkinan masih memiliki kekurangan. Setelah data ditampilkan, maka *pre-processing* dapat dijalankan dengan 3 tahap, yaitu membersihkan duplikasi data, *missing value data*, dan transformasi data.

Pembersihan Duplikasi Data

Tahap ini akan ditangani dengan menggunakan *primary key* yang ada pada data. *Primary key* pada suatu tabel tidak diijinkan ada yang sama. Oleh karena itu, saat input *record* baru yang mempunyai *primary key* yang sama, maka *record* tersebut akan ditolak. Untuk data karyawan kecelakaan kerja ini, yang menjadi *primary key* adalah atribut “id kejadian”.

Pembersihan Missing Value Data

Tahap ini menggunakan 2 cara, yaitu mencari modus dan rata-rata pada atribut. Pencarian modus akan dikenakan pad atribut tipe pekerjaan. Tipe data dari atribut ini berupa string, yaitu GRD A, GRD B, dan GRD C. Dari semua *record* yang ada, akan dicari kemungkinan muncul terbesar. *Value* terbesar akan mengisi *record* yang kosong. Pencarian modus ini berlaku untuk atribut kategorikal.



Gambar 3.2. Swimlane Diagram pada Pre-Processing

Transformasi Data

Tranformasi data pada perangkat lunak ini akan mereduksi beberapa atribut dan mengubah format pada beberapa atribut. Atribut yang direduksi dianggap tidak memiliki pengaruh pada potensi terjadinya kecelakaan kerja.

Tabel 2. Perubahan Atribut Setelah Transformasi Data

Sebelum Transformasi	Sesudah Transformasi
1. Id	1. Id
2. Nama	2. Jenis Kelamin
3. Jenis kelamin	3. Umur
4. Umur	4. Tipe pekerjaan
5. Kewarganegaraan	5. Lama bekerja
6. Tipe pekerjaan	6. Status
7. Waktu per hari	7. Pelatihan
8. Lama bekerja	8. Kecelakaan Kerja
9. Status	
10. Pelatihan	
11. Kecelakaan Kerja	

b. Data Mining

Proses data mining adalah pembentukan *tree* sebagai awal dari Algoritma C4.5 dan C5.0. Langkah awal dari proses ini adalah menghitung kemunculan kejadian untuk setiap *value* tiap atribut. Dari tabel hasil transformasi, akan dihitung atribut untuk perhitungan *node 1 (root)*.

Setelah diketahui kemunculan setiap *value*, kemudian akan dicari nilai *entropy*. Perhitungan *entropy* pada Algoritma C4.5 sama dengan Algoritma C5.0 yaitu menggunakan persamaan (1).

Perhitungan *entropy* pada sedua algoritma berfungsi sebagai kontroler. Dalam pembuatan *tree*, tidak semua atribut digunakan untuk membuat cabang. Karena dikontrol oleh *entropy*, maka tiap cabang dalam *tree* tidak sama dan mempunyai panjang yang berbeda.

Setelah perhitungan *entropy*, maka dilakukan perhitungan *information gain*. Perhitungan *information gain* pada kedua algoritma juga sama. Perhitungan *information gain* ini menggunakan persamaan (2).

Tabel 3. Perhitungan Kemunculan Kejadian

Node		Jml Kasus	On Work	Traffic on Commute	Traffic inside fact	
?	Total	200	35	113	52	
	Jenis Kelamin	L	195	35	112	48
		P	5	0	1	4
	Umur	Umur1	42	6	31	5
		Umur2	69	19	36	14
		Umur3	61	6	33	22
		Umur4	28	4	13	11
	Tipe Pekerjaan	GRD A	88	14	61	13
		GRD B	66	8	40	18
		GRD C	46	13	12	21
	Lama Bekerja	Junior	111	27	65	19
		Senior	89	8	48	33
	Status	ISS	7	0	1	6
		Reguler	65	15	37	13
	Pelatihan	Fuji	60	7	47	6
		Swabina	52	11	26	15
		Kontraktor	16	2	2	12
	Pelatihan	Yes	101	5	64	32
		No	99	30	49	20

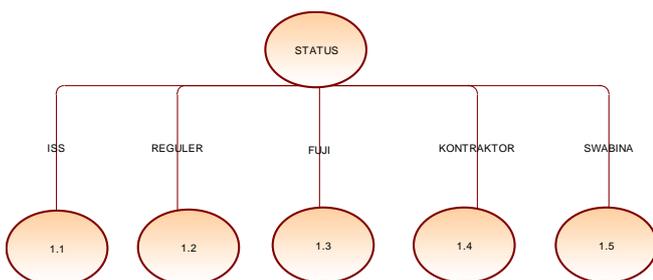
Perhitungan *information gain* inilah yang akan digunakan untuk membuat *node 1 (root)* pada Algoritma C4.5.

Perhitungan *information gain* dan *gain ratio* akan berhenti saat tidak ada lagi atribut yang dapat dipecah menjadi lebih kecil. Pengambilan *node 1 (root)* akan memilih atribut dengan nilai *gain ratio* tertinggi.

Tabel 4. Perhitungan *Entropy*, *Informaton Gain* dan *Gain Ratio* pada *node 1*

Node		Entropy	Information Gain	Gain Ratio	
1	Total	1.4107			
	Jenis Kelamin	L	1.4021	0.0437	0.0313
		P	NaN		
	Umur	Umur1	1.0899	0.0644	0.0121
		Umur2	1.4689		
		Umur3	1.3392		
		Umur4	1.4445		
	Tipe Pekerjaan	GRD A	1.1960	0.0959	0.0237
		GRD B	1.3181		
		GRD C	1.5374		
	Lama Bekerja	Junior	1.3841	0.0535	0.0198
		Senior	1.3236		
	Status	ISS	NaN	0.1871	0.0338
		Reguler	1.4153		
		Fuji	0.9698		
		Swabina	1.4914		
Pelatihan	Kontraktor	1.0613	-0.1503	-0.0198	
	Yes	1.1571			
	No	1.5508			

Tabel diatas akan dicetak setiap akan menambahkan cabang baru. Tahap inilah yang dinamakan *fase training* dengan *decision tree*. Setelah *tree* terbentuk, maka pola kecelakaan kerja yang berupa *rule* dapat direpresentasikan dengan bentuk *list*. *List rules* inilah yang nantinya akan diuji keakuratannya menggunakan data testing.



Gambar 3.3 Root yang dihasilkan pada Algoritma C4.5

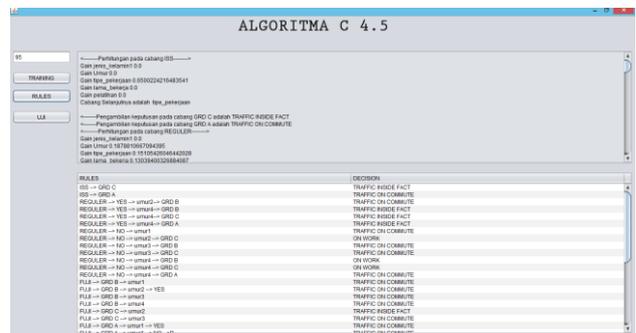
IV. PENGUJIAN DAN PEMBAHASAN

Uji coba pada perangkat lunak dalam penelitian tugas akhir ini dilakukan dengan beberapa pembagian data *testing* dan data *training*. Data *training* adalah data yang digunakan untuk membentuk pola kecelakaan kerja, sedangkan data *testing* adalah data yang digunakan untuk menguji keakuratan pola yang dihasilkan. Pembagian data *training* dan data *testing* dilakukan saat sebelum melakukan tahap *training*. Dalam perangkat lunak ini dibagi beberapa jenis pembagian seperti tabel 5.

Tabel 5. Pembagian Data yang Digunakan.

No	Data Training	Data Testing
1	95%	5%
2	90%	10%
3	85%	15%
4	80%	20%
5	75%	25%

Hasil uji coba dengan Algoritma C4.5 dan C5.0 menghasilkan jumlah pola yang berbeda, tergantung pada data *training* yang digunakan.



Gambar 4.1 Hasil *Training* pada C4.5 dengan 95% Data



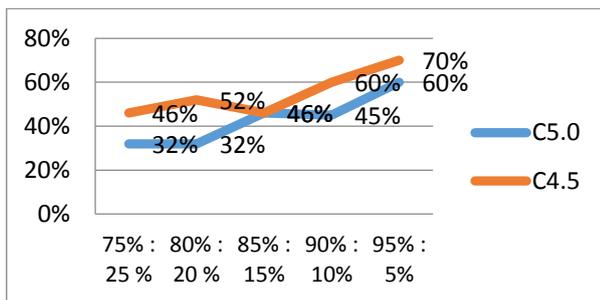
Gambar 4.2 Hasil *Training* pada C5.0 dengan 95% Data

Pola diatas dihasilkan dari relasi atribut yang membentuk *tree*. Relasi tersebut mulai dari *root* sampai *node* terakhir atau *leaf node*. *Leaf node* merepresentasikan keputusan atau jenis kecelakaan kerja yang terjadi. Salah satu pembacaan pola kecelakaan kerja seperti dibawah ini :

IF “Status” = “REGULER” && “Pelatihan” = “YES” && “Tipe Pekerjaan” = “GRD B” THEN “Kecelakaan Kerja” = “TRAFFIC ON COMMUTE”

Pada uji coba yang dilakukan pada kedua algoritma, terlihat bahwa prosentasi keakuratan cenderung menurun jika pembagian data pada data *training* lebih diperbesar. Semakin besar nilai prosentasi data *training*, maka

prosentasi keakuratan semakin kecil. Perbandingan keakuratan kedua algoritma dapat terlihat pada Gambar 4.3.



Gambar 4.3 Perbandingan Angka Keakuratan Pada Algoritma C4.5 dan C5.0

Dari pola yang dihasilkan pada uji coba Algoritma C4.5 dan C5.0, atribut yang paling berpengaruh terhadap terjadinya kecelakaan kerja adalah atribut "Status". Atribut "Status" memiliki nilai *information gain* dan *gain ratio* tertinggi. Nilai tertinggi pada atribut ini dipengaruhi oleh kemunculan atribut pada jumlah kasus yang terjadi. Selain itu, banyaknya *predictor* pada atribut ini juga mempengaruhi nilai dari *entropy*. Semakin tinggi nilai *entropy* pada suatu atribut, maka semakin berpengaruh atribut tersebut pada pembangunan *tree*. Jadi dapat disimpulkan bahwa atribut "Status" pada 200 data kecelakaan kerja ini adalah atribut yang paling berpengaruh dalam terjadinya kecelakaan kerja.

V. KESIMPULAN DAN SARAN

Berdasarkan analisis terhadap hasil pengujian yang telah dilakukan terhadap Algoritma C4.5 dan C5.0 dengan 200 data kecelakaan kerja, maka dapat diambil beberapa kesimpulan sebagai berikut:

1. Perangkat lunak dapat melakukan *pre processing*, pembangunan *tree* dan pengujian algoritma sehingga didapatkan pola prediksi kecelakaan kerja beserta angka keakuratannya.
2. *Rules* sebagai representasi dari pola kecelakaan kerja yang dihasilkan oleh Algoritma C5.0 lebih ringkas daripada *rules* yang dihasilkan oleh C4.5, yaitu 32 pola kecelakaan kerja.
3. Hasil keakuratan pola yang dihasilkan oleh Algoritma C4.5 lebih tinggi dibanding dengan Algoritma C 5.0, yaitu 70%.

Berdasarkan hasil yang dicapai pada tugas akhir ini, ada beberapa hal yang penulis sarankan untuk pengembangan selanjutnya yaitu:

4. Data yang dipakai dalam perangkat lunak ini masih sedikit dan bervariasi. Dalam perhitungan data mining, sebaiknya data harus dalam jumlah yang banyak karena mempengaruhi pola dan hasil keakuratan yang didapatkan. Untuk penelitian selanjutnya dapat menggunakan data yang lebih banyak.
5. Perangkat lunak ini belum bisa menampilkan hasil *decision tree* yang menarik. Untuk penelitian selanjutnya dapat menggunakan representasi data dalam bentuk *tree* atau *list rules* yang lebih baik sehingga memudahkan dalam testing data.

Pengembangan perangkat lunak ini dapat dilakukan dengan menambah menu input data baru demi manfaat yang lebih baik.

DAFTAR PUSTAKA

- [1] Bujlow, Tomasz, Tahir Riaz dkk. (2012). "A Method for Classification of Network Traffic on C5.0 Machine Learning Algorithm". **ICNC'12 : 2012 International Conference on Computing, Networking and Communication (ICNC)**. (pp 237-241). IEEE
- [2] Dunham, MH. (2003). "Data Mining Introductory and Advanced Topics". **Upper Saddle River, NJ : Pearson Education, Inc**
- [3] Gorunescu, Florin. (2011). "Data Mining: Concepts, Models, and Techniques". **Verlag Berlin Heidelberg, Springer**
- [4] Han, J, Micheline K, Jian Pei. (2012). "Data Mining Concept and Techniques 3rd Edition". **Morgan Kaufman Publisher, USA**
- [5] I. H. Witten and E. Frank. (2005). "Data Mining: Practical Machine Learning Tools and Techniques, 2nd Edition". **Morgan Kaufmann, San Francisco**
- [6] Kantardzic, M. (2003). "Data Mining Concept Models, Methods, and Algorithm". **New Jersey, USA : A John Wiley & Sons**
- [7] Moore, Samuel. A, Daniel M. D Addario, dkk. (2011). "Are Decision Trees Always Greener on the Open (Source) Side of the Fence?". **Sciweaver**
- [8] Permatasari, Anggun. (2009). "Investigasi Kecelakaan Penyebrangan Perlintasan KRL UI-Margonda Depok". **Tugas Akhir Fakultas Kesehatan Masyarakat, Universitas Indonesia**
- [9] Polat, Kemal, Gunes, Salih. (2009). "A Novel Hybrid Intelligent Method Based on C4.5 Decision Tree Classifier and One-Against-All Approach for Multi-Class Classification Problem". **Elsevier**. (pp 1587-1592).
- [10] Quinlan, J. R. (1993). "C4.5: Programs for Machine Learning. San Mateo" **CA : Morgan Kaufmann**
- [11] Rivas, T, M. Paz, dkk. (2011). "Explaining and Predicting Workplace Accident Using Data-Mining Techniques". **Elsivier**. (pp 739-747).
- [12] Tan, Pang-Ning, Michael Steinbach, Vipin Kumar. (2006). "Introduction of Data Mining". **Boston : Pearson - Addison Wesley**
- [13] Wu, Xindong, Kumar, Vipin, dkk. (2008). "Top 10 Algorithms in Data Mining". **Knowl Inf Systm (2008) 14 : 1-37. Springer-Verlag London Limited**