



BAB II

TINJAUAN PUSTAKA

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Telah ada beberapa penelitian yang dilakukan terkait Penerapan Metode C4.5 untuk memprediksi penerimaan pegawai *outsourcing*, diantaranya adalah :

Penelitian yang dilakukan oleh Fandy Ferdian Harryanto dan Seng Hansun (2017), yang berjudul **“Penerapan Algoritma C4.5 untuk Memprediksi Penerimaan Calon Pegawai Baru di PT. WISE”**. Hasil dari penelitian ini adalah Perusahaan pada umumnya memerlukan pegawai yang memiliki kemampuan yang baik, perilaku yang baik serta dapat menyelesaikan pekerjaan yang diberikan kepadanya. Namun terdapat beberapa kesulitan untuk mengetahui kualitas orang-orang yang memiliki potensi baik sebagai pegawai pada suatu perusahaan. Oleh karena itu diperlukan cara atau metode untuk mengidentifikasi calon pegawai suatu perusahaan. Algoritma C4.5 dapat digunakan untuk melakukan prediksi dan klasifikasi terhadap calon pegawai yang berpotensi untuk masuk ke dalam perusahaan dengan cara membuat pohon keputusan berdasarkan data-data yang sudah ada dan melakukan prediksi terhadap calon pegawai baru yang ingin masuk ke perusahaan. Berdasarkan metode pengukuran *akurasi ten-fold cross validation* telah didapatkan hasil pengukuran tingkat keberhasilan prediksi calon pegawai baru sebesar 71% dengan menggunakan aplikasi prediksi calon pegawai yang menerapkan C4.5.

Penelitian yang dilakukan oleh Siti Nurlela, Lestari Yusuf, Hermanto, Elah Nurlelah dan Mahmud Syarif (2018), yang berjudul “ **Penerapan Algoritma Decision Tree C4.5 Dalam Penerimaan Guru Pada SMK Sirajul Falah Parung**”. Hasil dari penelitian ini adalah Sekolah Kejuruan Sirajul Falah adalah salah satu sekolah menengah kejuruan swasta yang berlokasi di Jakarta Bogor yang selalu berusaha untuk meningkatkan kualitas baik secara internal maupun eksternal pada suatu secara berkelanjutan untuk bersaing dengan sekolah menengah kejuruan lainnya. Salah satu upayanya dibuat adalah untuk memberikan guru yang siap mendidik siswa dari Sirajul Falah Sekolah Menengah Kejuruan. Namun, proses penerimaan guru di Sirajul Falah Sekolah Kejuruan masih subyektif. Algoritma C4.5 akan digunakan dalam penelitian ini. *Decision tree* adalah klasifikasi dan prediksi yang sangat kuat dan terkenal. C4.5 Algoritma adalah algoritma yang digunakan untuk membentuk pohon keputusan untuk memprediksi atau mengklasifikasikan sangat yang kuat. Dalam algoritma C4.5 untuk membangun pohon keputusan yang harus dilakukan adalah memilih atribut sebagai root, lalu buat cabang untuk setiap nilai di *root*.

Penelitian yang di lakukan oleh Melan Susanti dan Jefa (2018), yang berjudul “**Prediksi Pengangkatan Karyawan Kontrak Menjadi Karyawan Tetap Menggunakan Decision Tree Pada PT. Baskara Cipta Pratama**”. Hasil dari penelitian ini adalah Penentuan karyawan kontrak untuk menjadi karyawan tetap di PT. Baskara Cipta Pratama, masih terlihat kurang tepat dan membutuhkan waktu, karena penilaian dan perhitungan hasil penentuan karyawan kontrak menjadi

karyawan tetap dilakukan secara bertahap seperti penyeleksian berkas, tes lisan atau tes tertulis, wawancara dan lain sebagainya, sehingga kemungkinan kesalahan dalam hasil akhir dari penentuan karyawan kontrak menjadi karyawan tetap sering tidak memenuhi kriteria yang di butuhkan perusahaan dan menghambat kinerja perusahaan. Penelitian ini bertujuan untuk memprediksi karyawan kontrak menjadi karyawan tetap yang dapat membantu perusahaan dalam memilih karyawan yang tepat. Hasil dari penelitian ini membangun sebuah sistem pendukung keputusan menggunakan metode algoritma C.45 guna meningkatkan akurasi dalam penentuan karyawan kontrak menjadi karyawan tetap, Hasil dari pengklasifikasiannya divalidasi *dengan k-fold cross validation* dengan tingkat akurasi 90.83 %, presisi 91.18% dan *recall* 62,50 %.

Penelitian yang dilakukan oleh Febri Erwanto, ahmad Haidar dan Edi Supratman (2019), yang berjudul **“Penerapan Algoritma C4.5 Untuk Penentuan Kelayakan Penerimaan Pegawai Di Kemenkumham”**. Hasil dari penelitian ini adalah Kementerian Hukum dan Hak Asasi Manusia Republik Indonesia (disingkat Kemenkumham RI) adalah kementerian dalam Pemerintah Indonesia yang membidangi urusan hukum dan hak asasi manusia. Instansi pemerintah seperti Kemenkumham seringkali mengadakan penerimaan pegawai, maka pihak dari pegawai dapat melakukan tindakan-tindakan yang diperlukan untuk mendapatkan seorang pegawai yang benar-benar memenuhi kriteria yang diminta. Algoritma C4.5 merupakan metode berbasis pohon keputusan (*decision tree*). Dalam algoritma C4.5 pemilihan atribut dilakukan dengan menggunakan *Gain, Ratio*, dengan mencari nilai

Entropy. Algoritma C4.5 sendiri menggunakan pendekatan induksi dimana yaitu membagi data berdasarkan kriteria yang dipilih untuk membuat sebuah pohon keputusan yang menggunakan pendekatan secara *topdown*. Algoritma C4.5 mampu memberikan hasil yang efektif dalam mendukung suatu keputusan dengan kriteria yang dibuat secara random.

Penelitian yang di lakukan oleh Deni Anugrah Saputra,dkk (2020), yang berjudul **“ Penentuan Faktor Kelayakan Penerimaan Karyawan Mneggunakan Algoritma Decision Tree Pada Perusahaan PT. Personel Alih Daya “**. Berdasarkan hasil penelitian yang telah dilakuan dengan *software Rapidminer* terhadap dataset penentuan faktor kelayakan penerimaan karyawan pada PT. Personel Alih Daya yang diperoleh dari *Employee Recruitment System* (ERS) dengan jumlah data sebesar 19163 kandidat yang melamar kerja pada tahun 2019 dengan menggunakan algoritma *Decission Tree* menghasilkan *Accuracy* Sebesar 73,27% dan nilai *AUC* 0,789. Faktor kelayakan penerimaan karyawan pada PT. Personel Alih Daya yang paling berpengaruh yaitu pada referensi dari teman, umur dari 26 sampai dengan 30, jenis kelamin wanita, posisi yang di lamar *Office Services* , berdomisili Jabodetabek dan level berpendidikan sarjana. disimpulkan hasil penelitian ini dapat memberikan rekomendasi kepada PT. Personel Alih Daya dalam menentukan calon tenaga kerja sesuai dengan kriteria yang dibutuhkan oleh klien.

II.2. Data Mining

Data mining bukanlah suatu bidang yang sama sekali baru. Salah satu kesulitan untuk mendefinisikan data *mining* adalah kenyataan bahwa data *mining* mewarisi banyak aspek dan teknik dari bidang-bidang ilmu yang sudah mapan terlebih dahulu. Berawal dari beberapa disiplin ilmu, data *mining* bertujuan untuk memperbaiki teknik tradisional sehingga bisa menangani jumlah data yang sangat besar dimensi data yang tinggi dan data yang heterogen dan berbeda sifat. Data *Mining* merupakan proses ataupun kegiatan untuk mengumpulkan data yang berukuran besar kemudian mengekstraksi data tersebut menjadi informasi-informasi yang nantinya dapat digunakan (Oscario, dkk, 2019 : 2).

Sedangkan menurut (Heni Sulastri;2017:2) Data *Mining* merupakan disiplin ilmu yang mempelajari metode untuk mengekstrak pengetahuan atau menemukan pola dari suatu data. Data *Mining* adalah suatu metode pengolahan data untuk menemukan pola yang tersembunyi dari data tersebut. Hasil dari pengolahan data dengan metode data *mining* ini dapat digunakan untuk mengambil keputusan di masa depan. Data *mining* ini juga dikenal dengan istilah *pattern recognition*.

Menyebutkan bahwa KDD atau *Knowledge Discovery From Data*, merupakan proses terstruktur, yaitu sebagai berikut:

1. *Data Cleaning* adalah proses memberikan data dari data *noise* dan tidak konsisten.
2. *Data Integration* adalah proses untuk menggabungkan data dari beberapa sumber yang berbeda.

3. *Data Selection* adalah proses untuk memilih data dari *database* yang sesuai dengan tujuan analisis.
4. *Data Transformation* adalah proses mengubah bentuk data menjadi data yang sesuai untuk proses *mining*.
5. *Data Mining* adalah proses penting yang menggunakan sebuah metode tertentu untuk memperoleh sebuah pola dari data.
6. *Pattern Evaluation* adalah proses mengidentifikasi pola.
7. *Knowledge Presentation* adalah yang dapat merepresentasikan informasi yang dibutuhkan, proses dimana informasi yang telah didapatkan kemudian digunakan oleh pemilik data

II.3. Karyawan Outsourcing

Karyawan *outsourcing* (kontrak) adalah karyawan yang bekerja pada suatu instansi dengan kerja waktu tertentu yang di dasari atas suatu perjanjian atau kontrak yang dapat juga disebut dengan PKWT (Perjanjian Kerja Wktu Tertentu), yaitu perjanjian kerja yang di dasarkan suatu jangka waktu yang diadakan untuk paling lama 2 tahun dan hanya diperpanjang 1 kali untuk jangka waktu maksimal 1 tahun (UU RI Ketenagakerjaan 2003 Pasal 50 ayat 1).

Sedangkan menurut (Melan Susanti dan jefi,2018:1) Karyawan kontrak adalah karyawan yang diperbantukan untuk menyelesaikan pekerjaanpekerjaan rutin perusahaan, dan tidak ada jaminan kelangsungan masa kerjanya. Dalam kelangsungan masa kerja karyawan kontrak ditentukan oleh prestasi kerjanya. Semakin bagus prestasi kerjanya, karyawan kontrak akan dipertahankan oleh perusahaan, namun jika

prestasi kerjanya tidak ada peningkatan maka perusahaan akan memberhentikan karyawan tersebut.

II.4. Algoritma C4.5

Algoritma C4.5 merupakan salah satu algoritma yang digunakan untuk membentuk *decision tree* berdasarkan *training data*. Algoritma ini merupakan metode klasifikasi dan prediksi yang sangat kuat dan terkenal. *Decision tree* adalah model prediksi menggunakan struktur *tree* atau struktur berhirarki. Konsep dari *decision tree* adalah mengubah data menjadi *decision tree* dan aturan-aturan keputusan (*decision rules*). Algoritma C4.5 merupakan pengembangan dari ID3. Beberapa pengembangan yang dilakukan C4.5 adalah bisa mengatasi *missing value*, *continue data* dan *pruning*. Algoritma C4.5 mempunyai *input* berupa *training samples* dan *samples*, *training samples* berupa data contoh yang akan digunakan untuk membangun sebuah *tree* yang telah diuji kebenarannya, sedangkan untuk *samples* merupakan *field-field* data yang nantinya akan kita gunakan sebagai parameter dalam melakukan klasifikasi data. Algoritma C4.5 dan *decision tree* merupakan dua model yang tidak terpisahkan. Karena untuk membangun sebuah *decision tree*, dibutuhkan algoritma C4.5. Di akhir tahun 1970 hingga di awal tahun 1980, J. Ross Quinlan seorang peneliti dibidang mesin pembelajaran membuat pengembangan sebuah *decision tree* yang dinamakan ID3 (*Iterative Dichotomizer*), walaupun sebenarnya proyek ini telah dibuat sebelumnya oleh E.B. Hunt, J. Marin dan PT. Stone, kemudian Ross Quinlan membuat algoritma dari pengembangan ID3 yang dinamakan C4.5 yang berbasis *supervised learning* (Rismayanti, 2016 : 2).

Ada beberapa tahapan dalam Algoritma C4.5 untuk membangun pohon keputusan yaitu sebagai berikut (Rismayanti, 2016 : 2) :

1. Mempersiapkan data *training*. Data *training* biasanya diambil dari data histori yang pernah terjadi sebelumnya atau disebut data masa lalu dan sudah dikelompokkan dalam kelas-kelas tertentu.
2. Menghitung akar dari pohon. Akar akan diambil dari atribut yang akan dipilih, dengan cara menghitung nilai *Gain* dari masing-masing atribut, nilai *Gain* yang paling tinggi yang akan menjadi akar pertama. Sebelum menghitung nilai *Gain* dari atribut, hitung dahulu nilai *Entropy*.

Rumus untuk menghitung nilai *Entropy*:

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

Di mana:

S = himpunan kasus

n = jumlah partisi S

P_i = proporsi S_i terhadap S

3. Rumus untuk menghitung nilai *Gain* :

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

Di mana:

S = himpunan kasus

A = Atribut

n = jumlah partisi atribut A

$|S_i|$ = Jumlah Kasus Pada Partisi ke-i

$|S|$ = jumlah kasus dalam S

4. Ulangi langkah ke-2 dan langkah ke-3 hingga semua *record* terpartisi

5. Proses partisi *decision tree* akan berhenti saat:

- a) Semua *record* dalam simpul N mendapat kelas yang sama.
- b) Tidak ada atribut didalam *record* yang dipartisi lagi.
- c) Tidak ada *record* didalam cabang yang kosong.

II.4.1. Contoh Studi Kasus Algoritma C4.5

Untuk memudahkan dalam penjelasan algoritma C4.5, berikut ini penulis menyertakan contoh studi kasus penelitian yang di lakukan oleh Panji Bimo Nugroho Setio, dkk, (2020) dengan judul penelitian Klasifikasi Dengan Pohon Keputusan Berbasis Algoritma C4.5

Tabel II.1. Kasus Keputusan Algoritma

NO	Cuaca	Suhu	Bermain/Tidak
1	Cerah	Panas	Tidak
2	Cerah	Panas	Tidak
3	Berawan	Panas	Ya
4	Hujan	Sejuk	Ya
5	Hujan	Dingin	Ya
6	Hujan	Dingin	Ya
7	Berawan	Dingin	Tidak
8	Cerah	Sejuk	Ya
9	Cerah	Dingin	Ya
10	Hujan	Sejuk	Ya

(Sumber : Panji Bimo Nugroho Setio, dkk, 2020)

Dalam kasus yang ada pada **tabel II.1** akan di buat sebuah pohon keputusan untuk menentukan bermain atau tidak dengan melihat keadaan cuaca dan suhu.

Secara umum algoritma Decision tree C4.5 untuk membangun pohon keputusan adalah sebagai berikut :

1. Pilih atribut sebagai akar
2. Buat cabang untuk masing-masing nilai
3. Bagi atribut terpilih dalam cabang
4. Ulangi proses untuk masing-masing cabang sampai semua atribut terpilih pada cabang memiliki kelas yang sama.

Untuk memilih atribut sebagai akar, didasarkan kepada nilai gain tertinggi dari atribut yang ada. Untuk menghitung *gain* digunakan rumus sebagai berikut :

$$\text{Gain}(S, A) = \text{Entropy}(S) - \sum_{i=1}^n \frac{|S_i|}{|S|} * \text{Entropy}(S_i) \dots\dots\dots(3)$$

Keterangan :

S : himpunan kasus

A : atribut

n : jumlah partisi atribut A

|S_i| : jumlah kasus pada partisi ke-i

|S| : jumlah kasus dalam S

Sementara itu, penghitungan nilai entropy dapat dilihat pada persamaan berikut.

$$\text{Entropy}(S) = \sum_{i=1}^n -p_i * \log_2 p_i \dots\dots\dots(4)$$

Keterangan:

S : Himpunan kasus

A : Atribut

N : Jumlah partisi atribut A

$|S_i|$: Jumlah kasus pada partisi ke-i

$|S|$: Jumlah kasus dalam S

Berikut ini adalah penjelasan lebih terperinci mengenai tiap-tiap langkah dalam pembentukan pohon keputusan dengan menggunakan algoritma C4.5 untuk menyelesaikan permasalahan pada **Tabel II.1**.

Menghitung jumlah kasus untuk keputusan Ya dan jumlah kasus untuk keputusan Tidak

Tabel II.2. Perhitungan Node 1

Node			Jumlah Kasus	Tidak	Ya
1	Total		10	3	7
	CUACA				
		Cerah	4	2	2
		Berawan	2	1	1
		Hujan	4	0	4
	SUHU				
		Panas	3	2	1
		Sejuk	3	0	3
		Dingin	4	1	3

(Sumber : Panji Bimo Nugroho Setio, dkk, 2020)

Selanjutnya menghitung nilai *Entropy* dari semua kasus yang di bagi berdasarkan atribut **CUACA** dan **SUHU**. Setelah itu, lakukan perhitungan *gain* untuk

setiap atribut. Hasil perhitungan untuk nilai Entropy (total) dengan persamaan 4 sebagai berikut :

$$Entropy (Total) = \left(-\frac{3}{10} * \log_2 \left(\frac{3}{10} \right) \right) + \left(-\frac{7}{10} * \log_2 \left(\frac{7}{10} \right) \right)$$

$$Entropy (Total) = 0.8812909$$

Sementara untuk nilai *gain* pada baris **CUACA** dihitung dengan menggunakan persamaan 3 sebagai berikut :

$$Gain (Total, CUACA) = 0.8812909 - \left(\frac{4}{10} * 1 \right) + \left(\frac{2}{10} * 1 \right) + \left(\frac{4}{10} * 0 \right)$$

$$Gain (Total, CUACA) = 0.281290899$$

Berikut ini adalah tabel untuk hasil dari nilai *Entropy* dan *Gain*

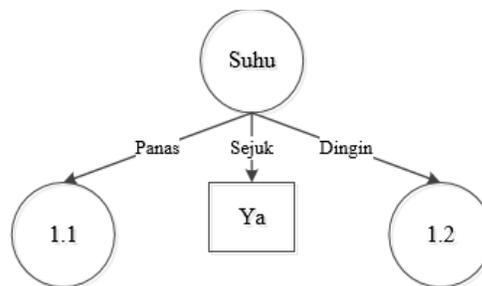
Tabel II.3. Hasil Penghitungan *Entropy* dan *Gain*

Node			Jumlah Kasus	Tidak	Ya	Entropy	Gain
1	Total		10	3	7	0.8812909	
	CUACA						0.281290899
		Cerah	4	2	2	1	
		Berawan	2	1	1	1	
		Hujan	4	0	4	0	
	SUHU						0.281290899
		Panas	3	2	1	0,9182958	
		Sejuk	3	0	3	0	
		Dingin	4	1	3	0,8112781	

(Sumber : Panji Bimo Nugroho Setio, dkk, 2020)

Berdasarkan perhitungan pada **Tabel II.3**, dipilih atribut yang memiliki nilai gain tertinggi untuk dijadikan sebagai akar. Pada **Tabel II.3**, terlihat kedua atribut memiliki nilai gain yang sama, oleh karena itu dipilih salah satu dari atribut tersebut.

Atribut yang dipilih yaitu suhu sebagai akar. Terdapat tiga nilai dari atribut suhu, yaitu panas, sejuk dan dingin. Dari ketiga nilai tersebut, nilai atribut sejuk sudah mengklasifikasikan kasus menjadi 1, yaitu keputusannya Ya, sehingga tidak perlu dilakukan perhitungan lagi. Untuk nilai atribut panas dan dingin masih perlu dilakukan perhitungan lagi. Dari hasil tersebut dapat digambarkan pohon keputusan sementara seperti pada **Gambar II.1**.



Gambar II.1. Pohon Keputusan Hasil Perhitungan Node 1
(Sumber : Panji Bimo Nugroho Setio, dkk, 2020)

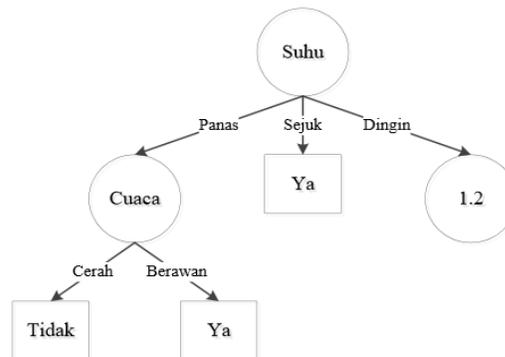
Selanjutnya menghitung nilai dari atribut cuaca untuk dijadikan node percabangan dari nilai atribut panas. Perhitungan dilakukan dengan cara yang sama seperti pada **Tabel II.2**. dan hasil perhitungan dapat dilihat pada **Tabel II.4**.

Node			Jumlah Kasus	Tidak	Ya	Entropy	Gain
1.1	SUHU (Panas)		3	2	1	0,9182958	
	CUACA						0,9182958
		Cerah	2	2	0	0	
		Berawan	1	0	1	0	
		Hujan	0	0	0	0	

(Sumber : Panji Bimo Nugroho Setio, dkk, 2020)

Berdasarkan perhitungan pada **Tabel II.2**, dipilih atribut cuaca sebagai node

percabangan dari nilai atribut panas. Pada **Tabel II.2.** terlihat ketiga nilai tersebut, nilai atribut cerah sudah mengklasifikasikan kasus menjadi 1, yaitu keputusannya tidak. Kemudian nilai atribut berawan sudah mengklasifikasikan kasus menjadi 1, yaitu keputusannya Ya sehingga tidak perlu dilakukan perhitungan lagi untuk atribut ini. Dari hasil tersebut dapat digambarkan pohon keputusan sementara seperti pada **Gambar II.2.**



Gambar II.2. Pohon Keputusan Hasil Perhitungan Node 1.1
(Sumber : Panji Bimo Nugroho Setio, dkk, 2020)

II.5. Rapid Miner

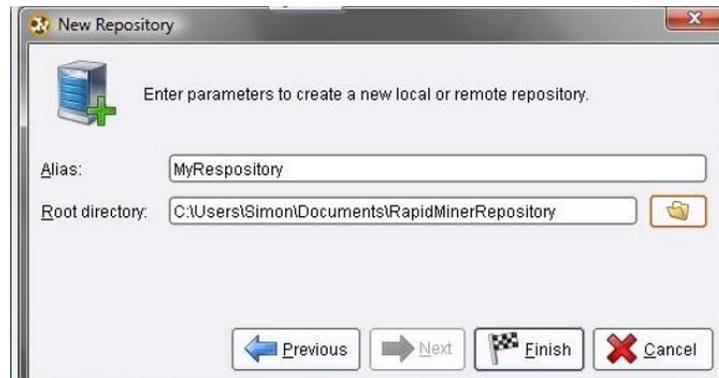
Rapid Miner merupakan perangkat lunak yang dibuat oleh Dr. Markus Hofmann dari *Institute of Technology Blanchardstown* dan Ralf Klinkenberg dari rapid-i.com dengan tampilan GUI (*Graphical User Interface*) sehingga memudahkan pengguna dalam menggunakan perangkat lunak ini. Perangkat lunak ini bersifat open source dan dibuat dengan menggunakan program Java di bawah lisensi GNU *Public Licence* dan *RapidMiner* dapat dijalankan di sistem operasi manapun. Dengan

menggunakan *RapidMiner*, tidak dibutuhkan kemampuan koding khusus, karena semua fasilitas sudah disediakan. *RapidMiner* dikhususkan untuk penggunaan data mining. Model yang disediakan juga cukup banyak dan lengkap, seperti Model *Bayesian*, *Modelling*, *Tree Induction*, *Neural Network* dan lain-lain. Banyak metode yang disediakan oleh *RapidMiner* mulai dari klasifikasi, klustering, asosiasi dan lainlain. Jika tidak ada model atau model algoritma yang tidak ada dalam Weka, pengguna boleh menambahkan modul lain, karena weka bersifat *open source*, jadi siapapun dapat ikut mengembangkan perangkat lunak ini *software* pilihan untuk melakukan ekstraksi data dengan metode-metode data mining (Siska Haryati, dkk, 2015 : 4).

II.6. Pengujian Rapid Miner

Menjalankan *RapidMiner* untuk pertama kali, akan menanyakan pembuatan repositori baru seperti yang terlihat pada **Gambar II.3**



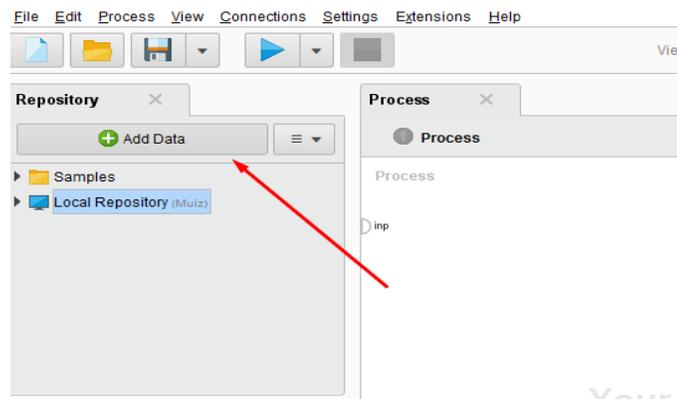


Gambar II.3 Pembuatan Repositori Baru

(Sumber : Abdl Muiz Khalimi, 2020)

Repositori ini berfungsi sebagai lokasi penyimpanan terpusat untuk data dan proses analisa.

Langkah ke 2 : Setelah semuanya sudah selesai, selanjutnya pada kotak dialog *Repository* pilih Add Data kemudian cari lokasi file data tersebut

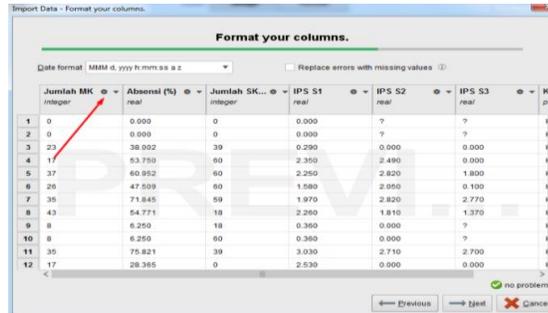


Gambar II.4. Kotak Dialog Untuk Mencari Lokasi File Data

(Sumber : Abdl Muiz Khalimi, 2020)

Add Data ini berfungsi untuk memasukan data yang telah kita sediakan sebelumnya .

Langkah ke 3 : Setelah sudah menemukan file datanya, pilih **Next**



Gambar II.5. Kotak Dialog Menentukan Type Data

(Sumber : Abdl Muiz Khalimi, 2020)

Pilih *Change Type* kriteria sebagai berikut :

Nominal : nilai secara kategori

Numeric : nilai numerik secara umum

Integer : bilangan bulat

Real : bilangan nyata

Text : teks bebas tanpa struktur

Binominal : nominal dua nilai

Polynomial : nominal lebih dari dua nilai

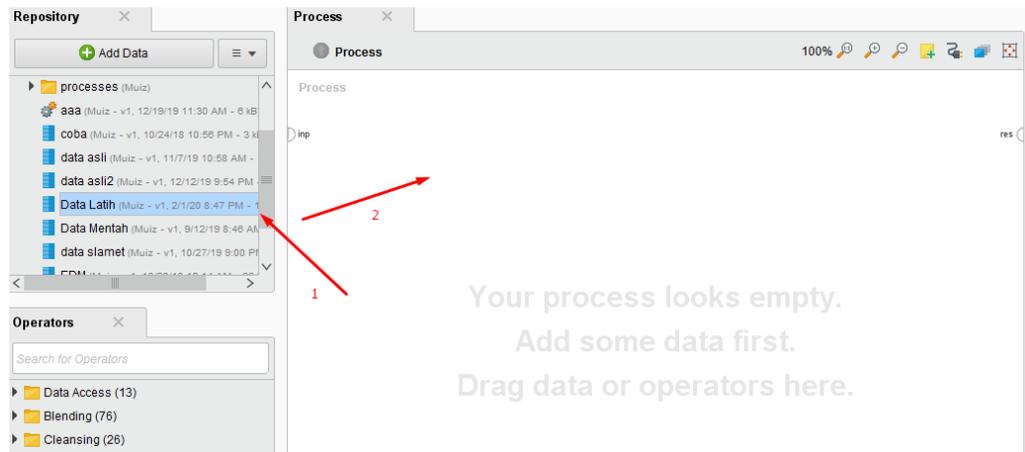
Date_time : tanggal dan waktu

Date : hanya tanggal

Time : hanya waktu

Setelah sudah menentukan kriteria, selanjutnya **klik Next** sampai dengan **finish**.

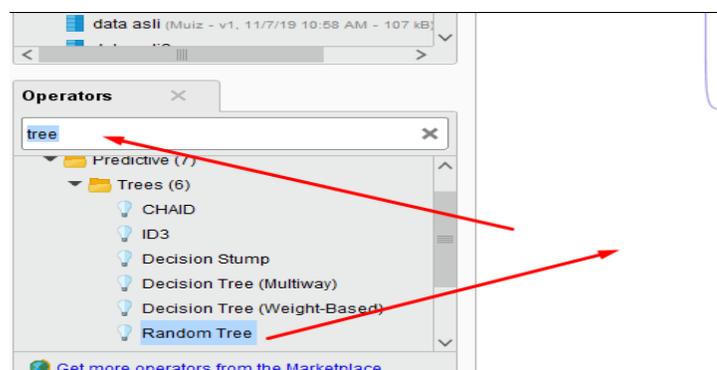
Langkah ke 4 : Pada halaman repository akan muncul data yang sudah *terimport* oleh sistem, kemudian *Drag* ke halaman lembar kerja *rapid miner*



Gambar II.6. Pemindahan Data ke Dalam Lembar Kerja

(Sumber : Abdl Muiz Khalimi, 2020)

Langkah ke 5 : Selanjut nya ada kotak dialog *Operators* , pilih menu-menu seperti **Random Tree , Split Data , Apply Model , Dan Permormance**. Kemudian *Drag* ke halaman kerja satu persatu. Cara mencari tulisan di kotak dialog pencarian pada kotak dialog tersebut

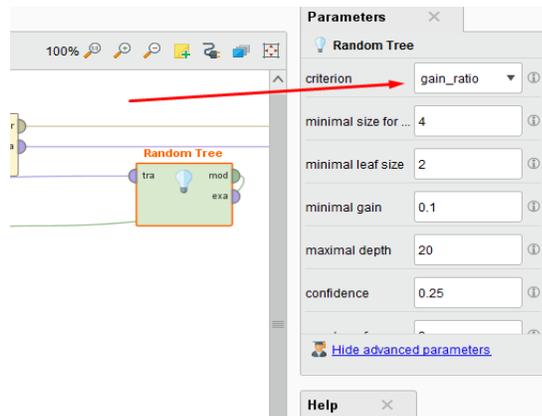


Gambar II.7. Penentuan Operator Modelling

(Sumber : Abdl Muiz Khalimi, 2020)

Operator Modeling ini dapat memproses set data yang berisi atribut nominal dan numerik. Atribut label harus nominal untuk klasifikasi dan numerik untuk menjalankan proses. Modeling yang digunakan disini adalah *modeling Random tree*.

Langkah ke 6 : Kemudian *Double* klik pada operator ***Random Tree*** dan atur parameter hitungannya. Kriteriaa hitungannya ada *gain ratio*, *info ratio*, *gini index* dan *akurasi*.



Gambar II.8. Mengatur Parameter Hitungan

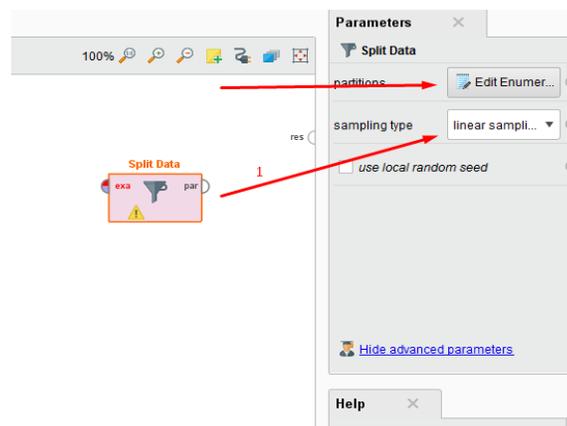
(Sumber : Abdl Muiz Khalimi, 2020)

Criterion, berguna memilih kriteria untuk menetapkan atribut sebagai akar dari *decision tree*. kriteria yang dapat dipilih, antara lain :

1. *Gain ratio*, merupakan varian dari information gain. Metode ini menghasilkan informasi gain untuk setiap atribut yang memberikan nilai atribut yang seragam.

2. *Information ratio*, dengan metode ini, semua *entropy* dihitung. Kemudian atribut dengan *entropy* minimum yang dipilih untuk dilakukan perpecahan pohon (*split*). Metode ini memiliki bias dalam memilih atribut dengan sejumlah besar nilai.
3. *Gini index*, merupakan ukuran ketidakadilan dari suatu *ExampleSet*. Metode ini memisahkan pada atribut yang dipilih memberikan penurunan indeks gini rata-rata yang dihasilkan subset
4. Akurasi. metode ini memilih beberapa atribut untuk memecah pohon (*split*) yang memaksimalkan akurasi dari keseluruhan pohon.

Langkah ke 7 : Selanjutnya double klik pada operator **Split Data**, kemudian atur *sampling type* pada kotak parameter, seperti gambar di bawah ini :



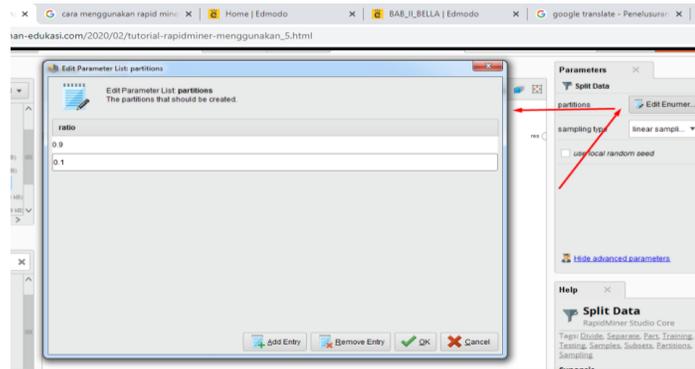
Gambar II.9. Kotak Dialog Split Data

(Sumber : Abdl Muiz Khalimi, 2020)

ini bertujuan untuk mengetahui performa dari model yang akan di bentuk oleh C4.5

Langkah ke 8 : Setelah itu atur porposisi atau kapasitas antara data latih dan data uji.

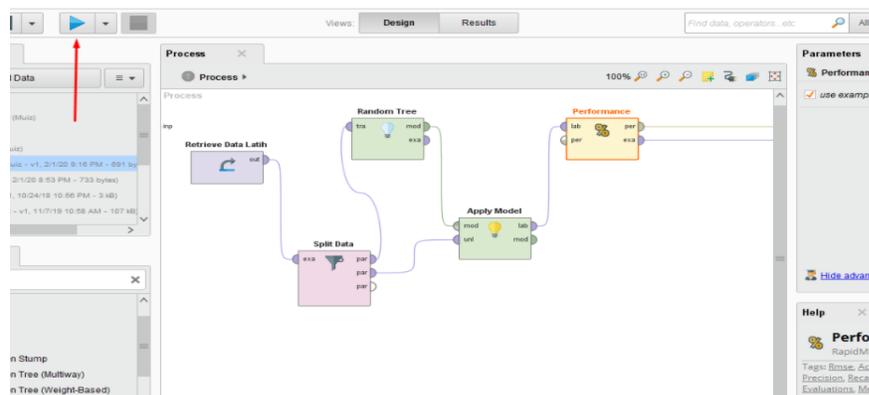
klik pada menu Edit *Enumerating*. Kemudian akan muncul kotak dialog seperti di bawah ini.



Gambar II.10. Kotak Dialog Edit Enumering

(Sumber : Abdl Muiz Khalimi, 2020)

Setelah itu **Add Entry** kemudian **Ok**. Maksud dari 0.9 adalah data yang di ambil 90% dan 0.1 adalah data uji di ambil 10% dari perbandingan seluruh Dataset
Langkah ke 9 : langkah terakhir adalah sambungkan semua operator seperti gambar di bawah ini :



Gambar II.11. Menyambungkan Semua Operator

(Sumber : Abdl Muiz Khalimi, 2020)

Pastikan garis yang dihubungkan sudah secara benar, sehingga tidak terjadi eror

Langkah ke 10 : Setelah semua sudah tersambung, **klik *Play* atau *Start*** , kemudian jika proses selesai akan muncul kotak dialog seperti berikut :

accuracy: 100.00%				
	true Keluar	true Lulus	class precision	
pred Keluar	46	0	100.00%	
pred Lulus	0	39	100.00%	
class recall	100.00%	100.00%		

Gambar II.12. Hasil Akhir Rapid Miner
(Sumber : Abdl Muiz Khalimi, 2020)

Berikut penjelasan arah panah dengan nomer :

1. Hasil total yang sudah di lakukan proses Split data
2. Hasil dari nilai Performa mneggunakan Algoritma c4.5
3. Total Dataset asli
4. Beberapa kriteria Evaluasi untuk mengukur Performa suatu algoritma