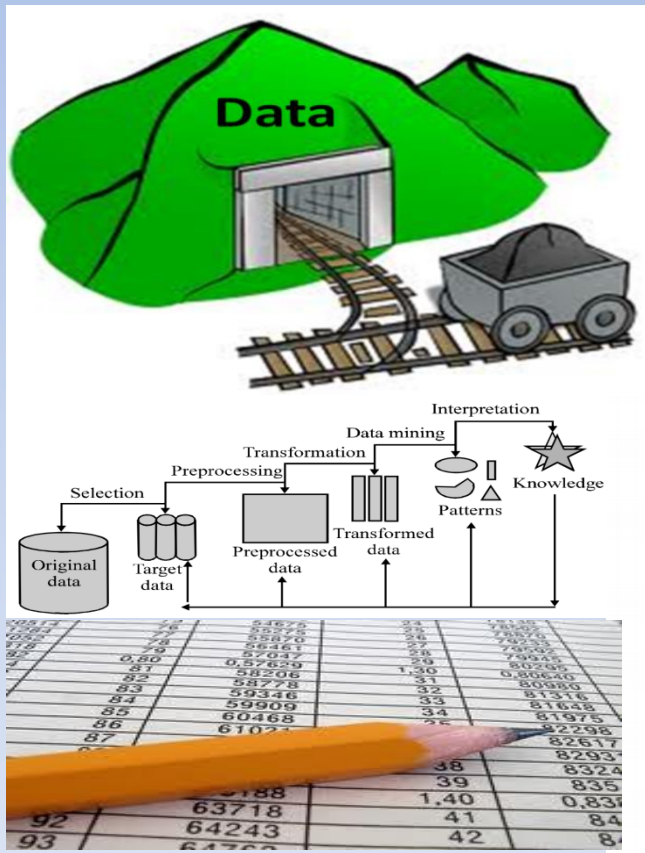
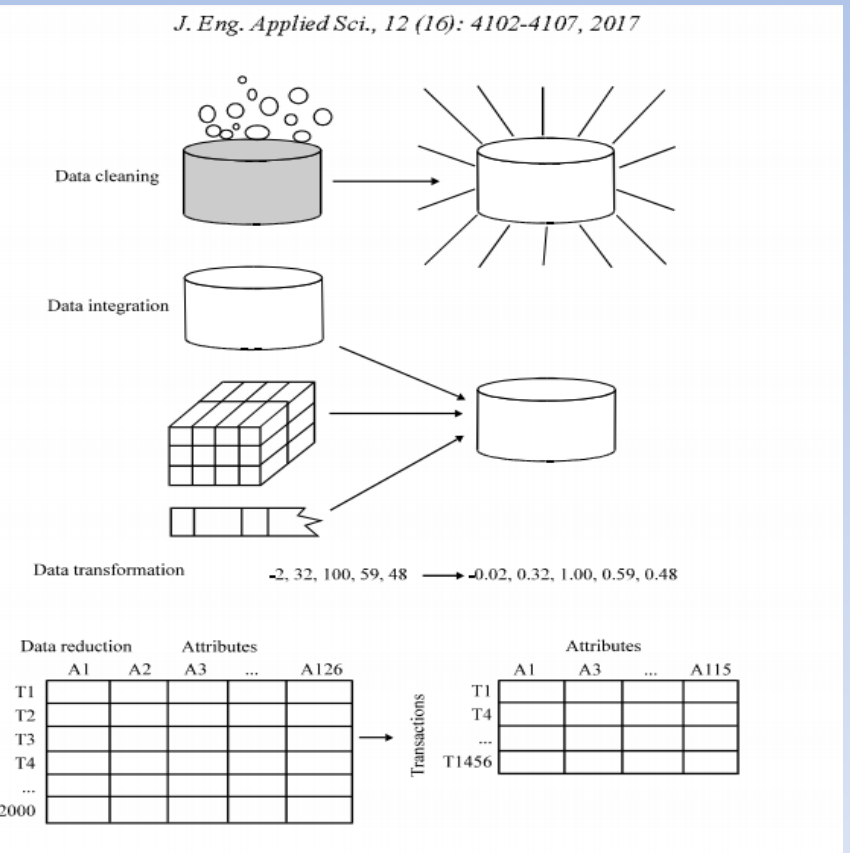
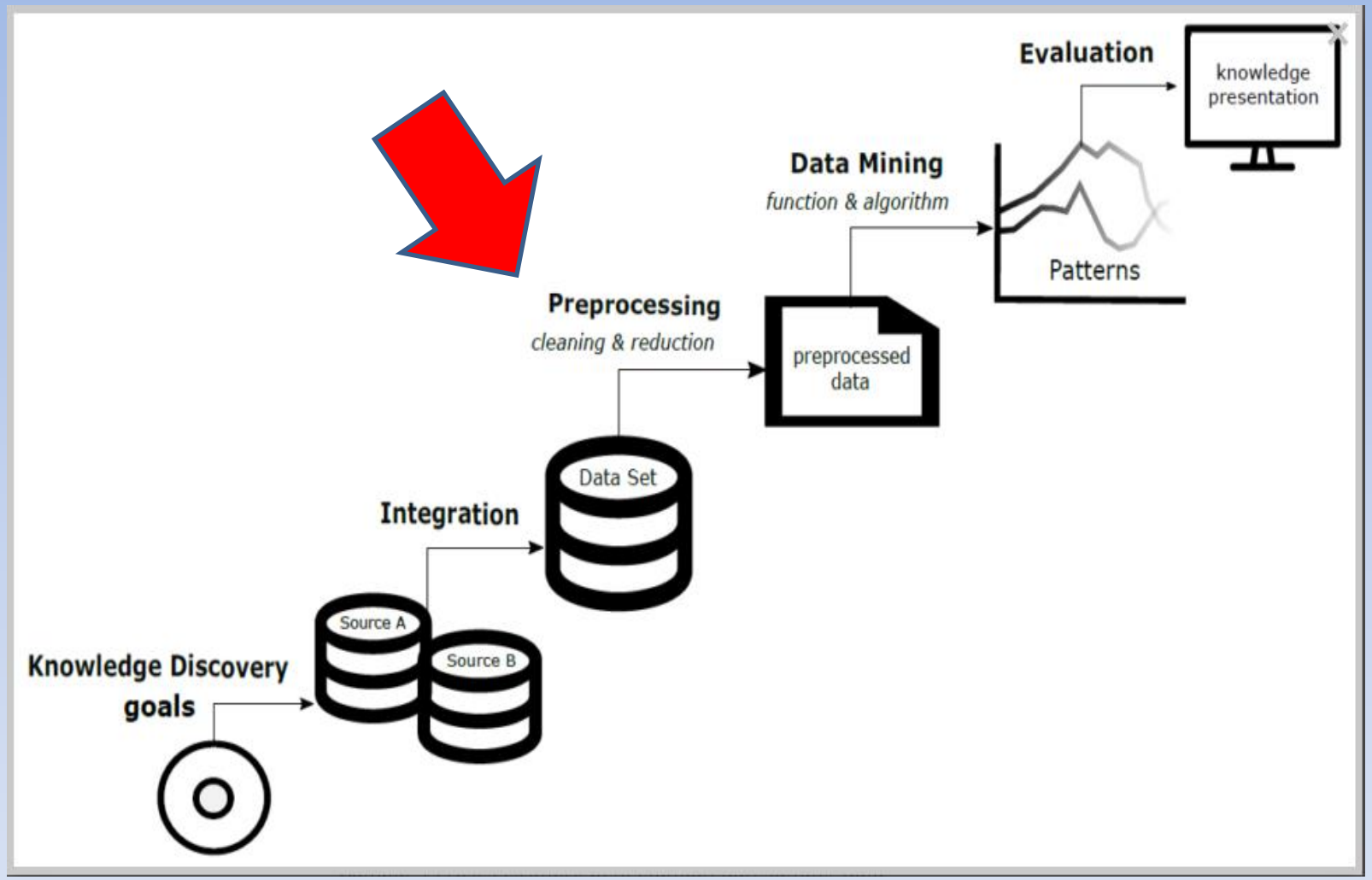
DATA PREPARATION





Pengertian Data Preparation

Data Preparation atau bisa disebut juga dengan data preprocessing adalah suatu proses/langkah yang dilakukan untuk membuat data mentah menjadi data yang berkualitas (input yang baik untuk data mining tools).


Evaluation
knowledge presentation
Data Mining
function & algorithm
Patterns
Preprocessing
cleaning & reduction
preprocessed data
Integration
Data Set
Source A
Source B
Knowledge Discovery goals

Mengapa data perlu di-preprocessing?

Karena dalam data mentah masih terdapat data yang : incomplete, yaitu

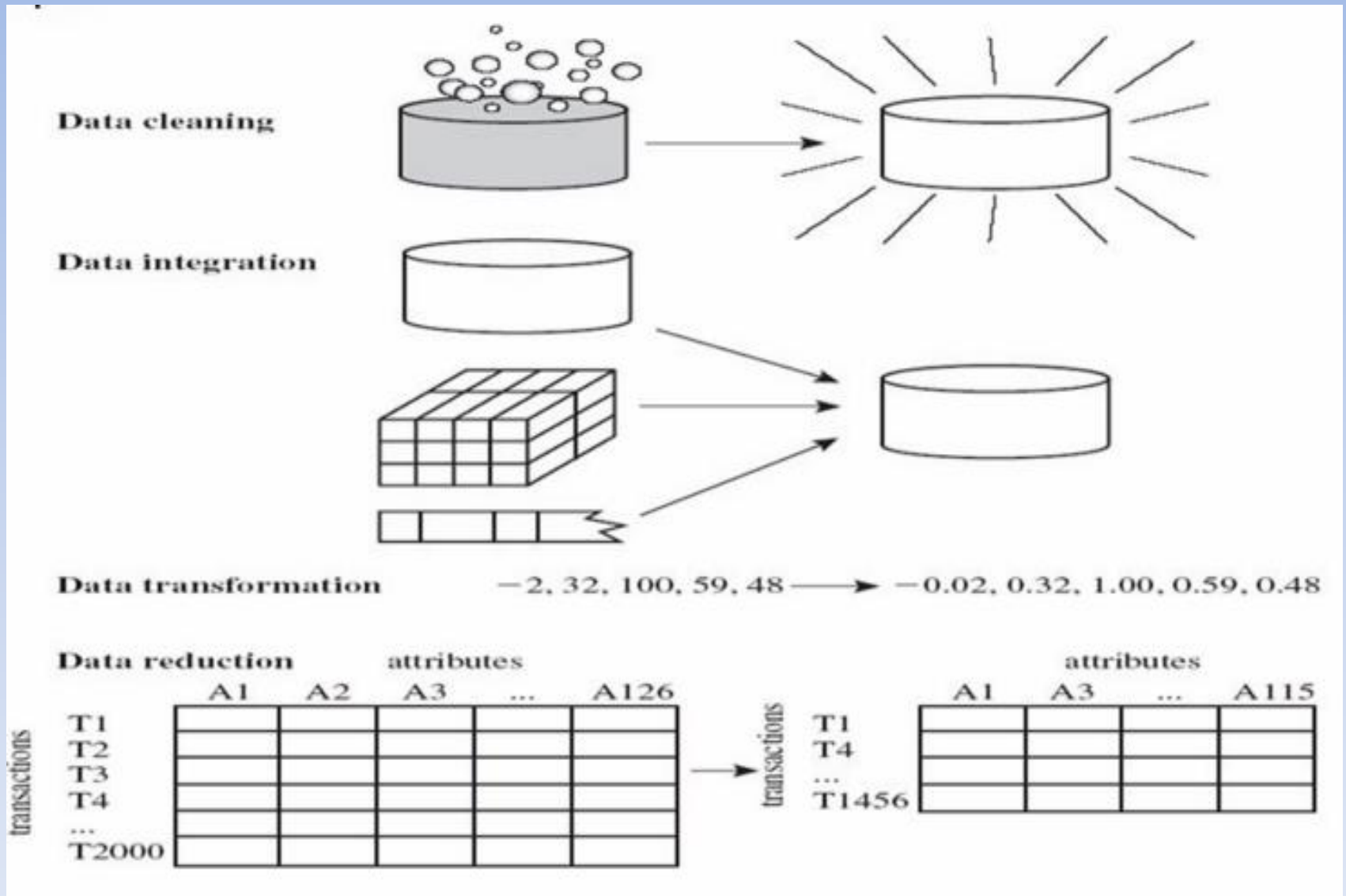
1. data yang kekurangan nilai atribut atau hanya mengandung agregat data (contoh : address = " ").
2. noisy, yaitu data yang masih mengandung error dan outliers (contoh : salary = -10).
3. inconsistent, yaitu data yang mengandung discrepansi dalam code dan nama atau singkatnya datanya tidak konsisten (contoh : dulu rating = 1,2,3 sekarang a,b,c).

Mengapa harus dilakukan data preparation?

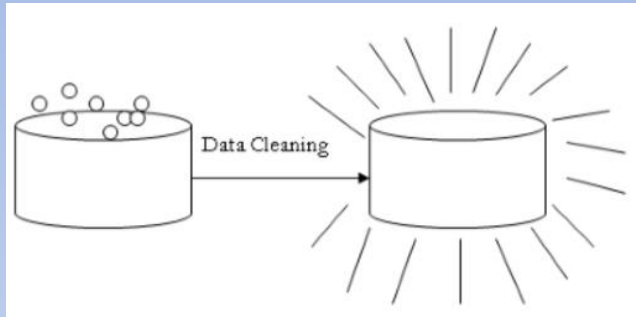
Jika data masukan tidak berkualitas, maka hasil data mining juga tidak akan berkualitas.

1. Keputusan yang berkualitas pasti berasal/berdasarkan data yang berkualitas.
2. Data Warehouse membutuhkan integrasi yang konsisten dari data yang berkualitas.

LANGKAH-LANGKAH DALAM DATA PREPARATION :



LANGKAH-LANGKAH DALAM DATA PREPARATION :



1. **Data Cleaning**

Dalam data cleaning yang akan kita lakukan antara lain mengisi missing value, mengidentifikasi outlier, menangani data noise, mengoreksi data yang tidak konsisten, dan menyelesaikan masalah redudansi data akibat integrasi data.

LANGKAH-LANGKAH DALAM DATA PREPARATION :

2. Data Integration

Data integration adalah suatu langkah untuk menggabungkan data dari beberapa sumber. Data integration hanya dilakukan jika data berasal dari tempat yang berbeda-beda (sumber data tidak hanya dari 1 tempat). Langkah yang dilakukan antara lain mengintegrasikan skema, mengidentifikasi masalah entitas, dan mendeteksi sekaligus menyelesaikan konflik pada nilai data.

LANGKAH-LANGKAH DALAM DATA PREPARATION :

3. Data Transformation

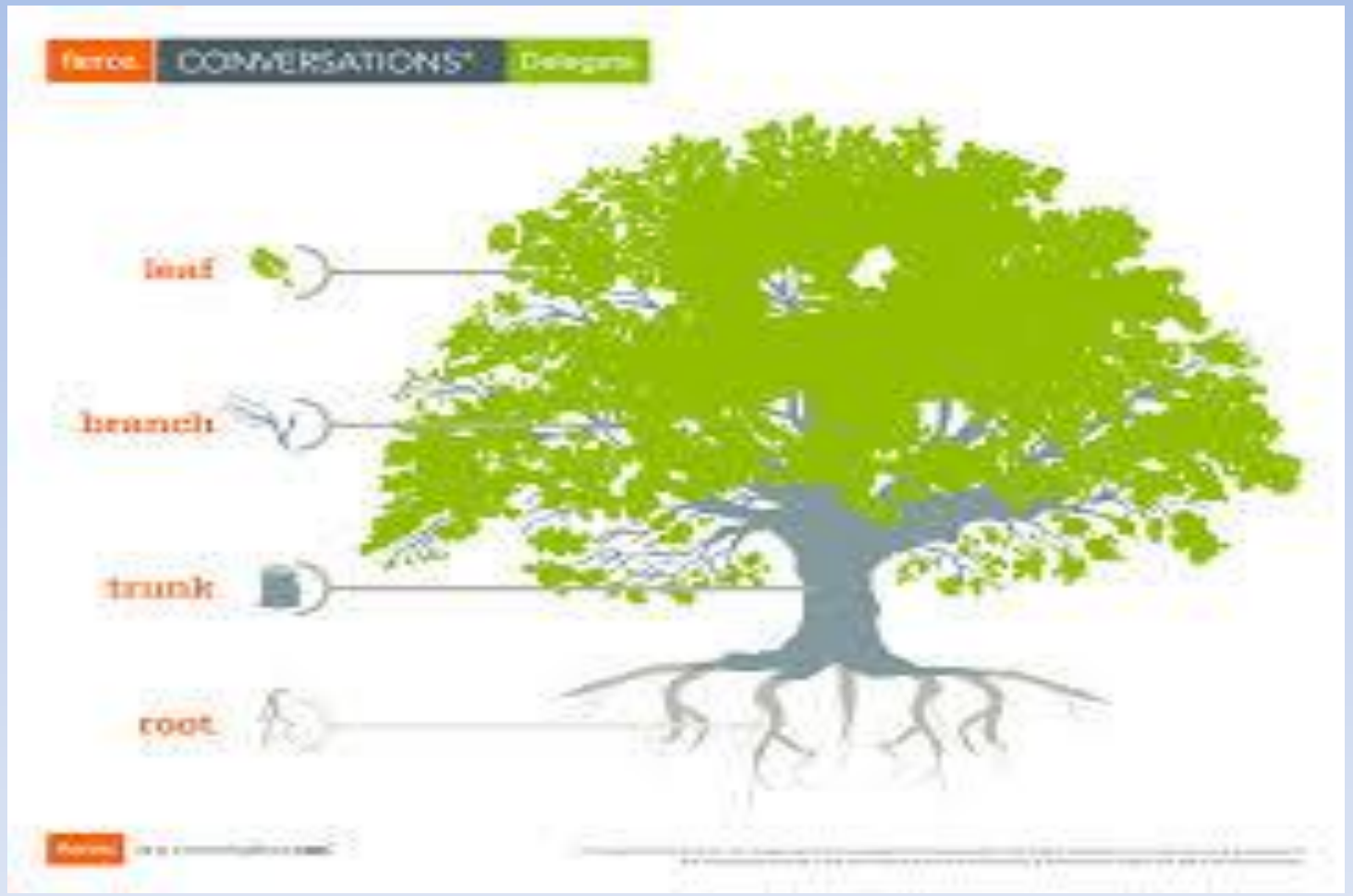
Data transformation yaitu mengubah suatu data supaya diperoleh data yang lebih berkualitas. Yang akan dilakukan antara lain menghilangkan noise dari data (smoothing), meng-agregasi data, generalisasi data, normalisasi data, dan pembentukan atribut/fitur.

LANGKAH-LANGKAH DALAM DATA PREPARATION :

4. Data Reduction

Data Reduction yaitu langkah untuk mereduksi dimensi, atribut ataupun jumlah data. Yang akan dilakukan antara lain agregasi data cube, reduksi dimensi, diskretisasi, dan kompresi data.

DATA PREPARITON
DGN DECISION TREE



PENGERTIAN POHON KEPUTUSAN

- Pohon dalam analisis pemecahan masalah pengambilan keputusan adalah pemetaan mengenai alternatif-alternatif pemecahan masalah yang dapat diambil dari masalah tersebut.

- Pohon tersebut juga memperlihatkan faktor-faktor kemungkinan/probablitas yang akan mempengaruhi alternatif alternatif keputusan tersebut, disertai dengan estimasi hasil akhir yang akan didapat bila kita mengambil alternatif keputusan tersebut.

MANFAAT POHON KEPUTUSAN

- Kemampuannya untuk mem-*break down* proses pengambilan keputusan yang kompleks menjadi lebih simpel sehingga pengambil keputusan akan lebih menginterpretasikan solusi dari permasalahan.
- Pohon Keputusan juga berguna untuk mengeksplorasi data, menemukan hubungan tersembunyi antara sejumlah calon variabel input dengan sebuah variabel target.
- Pohon keputusan memadukan antara eksplorasi data dan pemodelan, sehingga sangat bagus sebagai langkah awal dalam proses pemodelan bahkan ketika dijadikan sebagai model akhir dari beberapa teknik lain.
- Sering terjadi tawar menawar antara keakuratan model dengan transparansi model.

KELEBIHAN POHON KEPUTUSAN

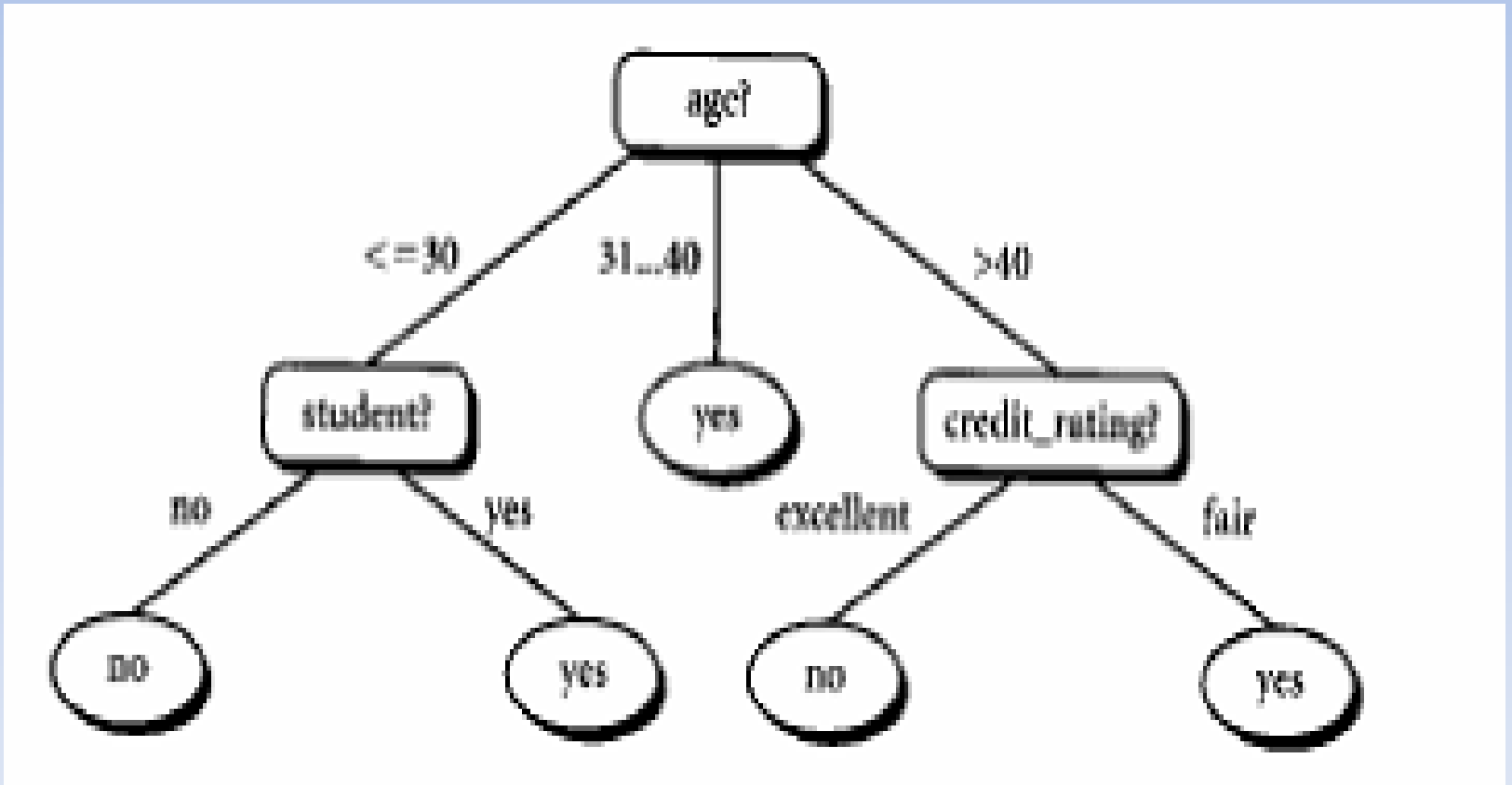
- Daerah pengambilan keputusan yang sebelumnya kompleks dan sangat global, dapat diubah menjadi lebih simpel dan spesifik.
- Eliminasi perhitungan-perhitungan yang tidak diperlukan, karena ketika menggunakan metode pohon keputusan maka sample diuji hanya berdasarkan kriteria atau kelas tertentu.
- Fleksibel untuk memilih fitur dari internal node yang berbeda, fitur yang terpilih akan membedakan suatu kriteria dibandingkan kriteria yang lain dalam node yang sama.

KEKURANGAN POHON KEPUTUSAN

- Terjadi overlap terutama ketika kelas-kelas dan criteria yang digunakan jumlahnya sangat banyak. Hal tersebut juga dapat menyebabkan meningkatnya waktu pengambilan keputusan dan jumlah memori yang diperlukan.
- Pengakumulasian jumlah eror dari setiap tingkat dalam sebuah pohon keputusan yang besar.
- Kesulitan dalam mendesain pohon keputusan yang optimal.
- Hasil kualitas keputusan yang didapatkan dari metode pohon keputusan sangat tergantung pada bagaimana pohon tersebut didesain.

MODEL POHON KEPUTUSAN

Pohon keputusan adalah model prediksi menggunakan struktur pohon atau struktur berhirarki. Contoh dari pohon keputusan dapat dilihat di Gambar 1 berikut ini.



Algoritma C4.5

Untuk memudahkan penjelasan mengenai algoritma C4.5 berikut ini disertakan contoh kasus yang dituangkan dalam Tabel 1:

Tabel 1. Keputusan Bermain Tenis

NO	OUTLOOK	TEMPERATURE	HUMIDITY	WINDY	PLAY
1	Sunny	Hot	High	FALSE	Np
2	Sunny	Hot	High	TRUE	No
3	Cloudy	Hot	High	FALSE	Yes
4	Rainy	Mild	High	FALSE	Yes
5	Rainy	Cool	Normal	FALSE	Yes
6	Rainy	Cool	Normal	TRUE	Yes
7	Cloudy	Cool	Normal	TRUE	Yes
8	Sunny	Mild	High	FALSE	No
9	Sunny	Cool	Normal	FALSE	Yes
10	Rainy	Mild	Normal	FALSE	Yes
11	Sunny	Mild	Normal	TRUE	Yes
12	Cloudy	Mild	High	TRUE	Yes
13	Cloudy	Hot	Normal	FALSE	Yes
14	Rainy	Mild	High	TRUE	No

Algoritma C4.5

Dalam kasus yang tertera pada Tabel 1, akan dibuat pohon keputusan untuk menentukan main tenis atau tidak dengan melihat keadaan cuaca (outlook), temperatur, kelembaban (humidity) dan keadaan angin (windy).

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

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

Algoritma C4.5

Untuk memilih atribut sebagai akar, didasarkan pada nilai gain tertinggi dari atribut-atribut yang ada. Untuk menghitung gain digunakan rumus seperti tertera dalam Rumus :

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

Dengan :

S : Himpunan kasus
A : Atribut
n : Jumlah partisi atribut A
|Si| : Jumlah kasus pada partisi ke i
|S| : Jumlah kasus dalam S

Sedangkan perhitungan nilai entropy dapat dilihat pada rumus 2 berikut:

$$Entropy(S) = \sum_{i=1}^{n} - pi * \log_2 pi \qquad (2)$$

dengan :

S : Himpunan Kasus
A : Fitur
n : Jumlah partisi S
pi : Proporsi dari S_i terhadap S

Algoritma C4.5

Berikut ini adalah penjelasan lebih rinci mengenai masing-masing langkah dalam pembentukan pohon keputusan dengan menggunakan algoritma C4.5 untuk menyelesaikan permasalahan.

a. Menghitung jumlah kasus, jumlah kasus untuk keputusan Yes, jumlah kasus untuk keputusan No,dan Entropy dari semua kasus dan kasus yang dibagi berdasarkan atribut OUTLOOK, TEMPERATURE, HUMIDITY dan WINDY. Setelah itu lakukan penghitungan Gain untuk masing masing atribut. Hasil perhitungan ditunjukkan oleh Tabel 2.

Algoritma C4.5

Tabel 2. Perhitungan Node 1

NODE			JUMLAH KASUS (S)	NO (S_1)	YES (S_2)	ENTROPY	GAIN
1	TOTAL		14	4	10	0.863120569	
	OUTLOOK						0.258521037
		CLOUDY	4	0	4	0	
		RAINY	5	1	4	0.721928095	
		SUNNY	5	3	2	0.970950594	
	TEMPERATURE						0.183850925
		COOL	4	0	4	0	
		HOT	4	2	2	1	
		MILD	6	2	4	0.918295834	
	HUMIDITY						0.370506501
		HIGH	7	4	3	0.985228136	
		NORMAL	7	0	7	0	
	WINDY						0.005977711
		FALSE	8	2	6	0.811278124	
		TRUE	6	4	2	0.918295834	

Algoritma C4.5

Baris TOTAL kolom Entropy pada Tabel 2 dihitung dengan rumus 2, sebagai berikut:

$$Entropy(Total) = (-\frac{4}{14} * \log_2(\frac{4}{14})) + (-\frac{10}{14} * \log_2(\frac{10}{14}))$$

$$Entropy(Total) = 0.863120569$$

Sedangkan nilai Gain pada baris OUTLOOK dihitung dengan menggunakan rumus 1, sebagai berikut:

$$Gain(Total, Outlook) = Entropy(Total) - \sum_{i=1}^{n} \frac{|Outlook_i|}{|Total|} * Entropy(Outlook_i)$$

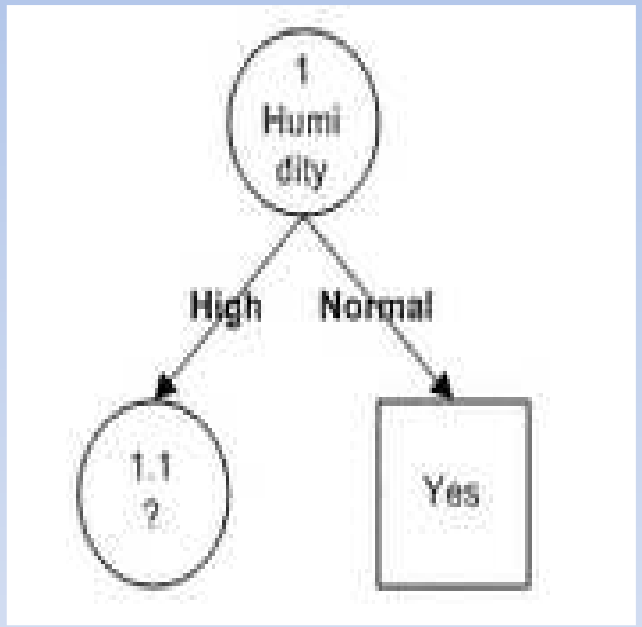
$$Gain(Total, Outlook) = 0.863120569 - ((\frac{4}{14} * 0) + (\frac{5}{14} * 0.723) + (\frac{5}{14} * 0.97))$$

Sehingga didapat Gain(Total, Outlook) = 0.258521037

Algoritma C4.5

Dari hasil pada Tabel 2 dapat diketahui bahwa atribut dengan Gain tertinggi adalah HUMIDITY yaitu sebesar 0.37. Dengan demikian HUMIDITY dapat menjadi node akar. Ada 2 nilai atribut dari HUMIDITY yaitu HIGH dan NORMAL. Dari kedua nilai atribut tersebut, nilai atribut NORMAL sudah mengklasifikasikan kasus menjadi 1 yaitu keputusan-nya Yes, sehingga tidak perlu dilakukan perhitungan lebih lanjut, tetapi untuk nilai atribut HIGH masih perlu dilakukan perhitungan lagi.

Dari hasil tersebut dapat digambarkan pohon keputusan sementara seperti Gambar disamping



Algoritma C4.5

NO	OUTLOOK	TEMPERATURE	HUMIDITY	WINDY	PLAY
1	Sunny	Hot	High	FALSE	Np
2	Sunny	Hot	High	TRUE	No
3	Cloudy	Hot	High	FALSE	Yes
4	Rainy	Mild	High	FALSE	Yes
5	Sunny	Mild	High	FALSE	No
6	Cloudy	Mild	High	TRUE	Yes
7	Rainy	Mild	High	TRUE	No

Algoritma C4.5

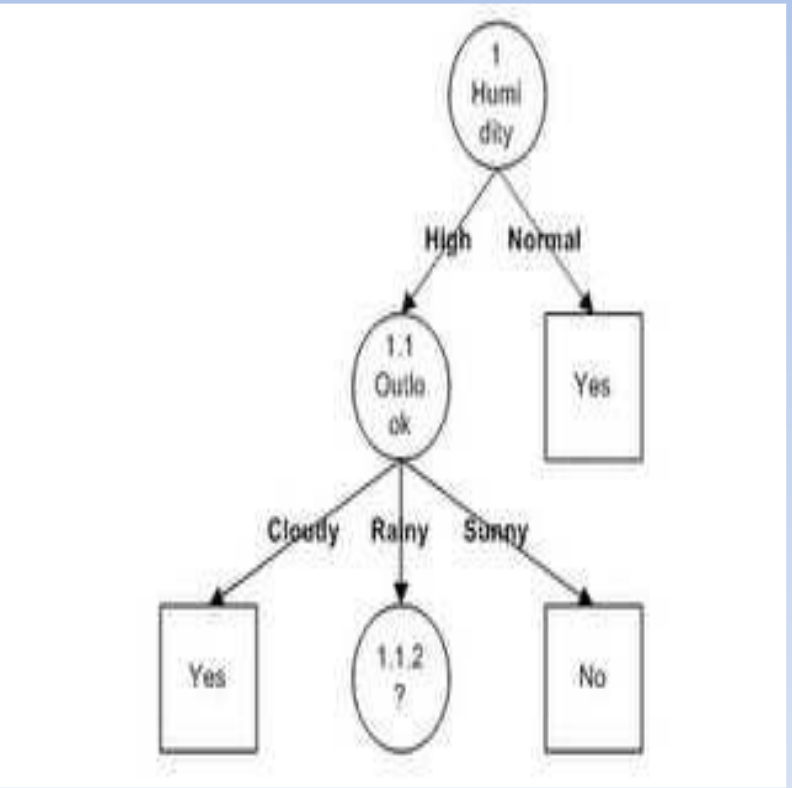
b. Menghitung jumlah kasus, jumlah kasus untuk keputusan Yes, jumlah kasus untuk keputusan No, dan Entropy dari semua kasus dan kasus yang dibagi berdasarkan atribut OUTLOOK, TEMPERATURE dan WINDY yang dapat menjadi node akar dari nilai atribut HIGH. Setelah itu lakukan penghitungan Gain untuk masing-masing atribut. Hasil perhitungan ditunjukkan oleh Tabel 3.

Algoritma C4.5

Node			Jml Kasus (S)	Tidak (S1)	Ya (S2)	Entropy	Gain
1.1	HUMIDITY-HIGH		7	4	3	0.985228136	
	OUTLOOK						0.69951385
		CLOUDY	2	0	2	0	
		RAINY	2	1	1	1	
		SUNNY	3	3	0	0	
	TEMPERATURE						0.020244207
		COOL	0	0	0	0	
		HOT	3	2	1	0.918295834	
		MILD	4	2	2	1	
	WINDY						0.020244207
		FALSE	4	2	2	1	
		TRUE	3	2	1	0.918295834	

Algoritma C4.5

Dari hasil pada Tabel 3 dapat diketahui bahwa atribut dengan Gain tertinggi adalah **OUTLOOK yaitu sebesar 0.67.** Dengan demikian OUTLOOK dapat menjadi node cabang dari nilai atribut HIGH. Ada 3 nilai atribut dari OUTLOOK yaitu CLOUDY, RAINY dan SUNNY. Dari ketiga nilai atribut tersebut, nilai atribut CLOUDY sudah mengklasifikasikan kasus menjadi 1 yaitu keputusan-nya Yes dan nilai atribut SUNNY sudah mengklasifikasikan kasus menjadi satu dengan keputusan No, sehingga tidak perlu dilakukan perhitungan lebih lanjut, tetapi untuk nilai atribut RAINY masih perlu dilakukan perhitungan lagi.



Algoritma C4.5

NO	OUTLOOK	TEMPERATURE	HUMIDITY	WINDY	PLAY
1	Rainy	Mild	High	FALSE	Yes
2	Rainy	Mild	High	TRUE	No

Algoritma C4.5

Menghitung jumlah kasus, jumlah kasus untuk keputusan Yes, jumlah kasus untuk keputusan No, dan Entropy dari semua kasus dan kasus yang dibagi berdasarkan atribut TEMPERATURE dan WINDY yang dapat menjadi node cabang dari nilai atribut RAINY. Setelah itu lakukan penghitungan Gain untuk masing-masing atribut. Hasil perhitungan ditunjukkan oleh Tabel 4.

Algoritma C4.5

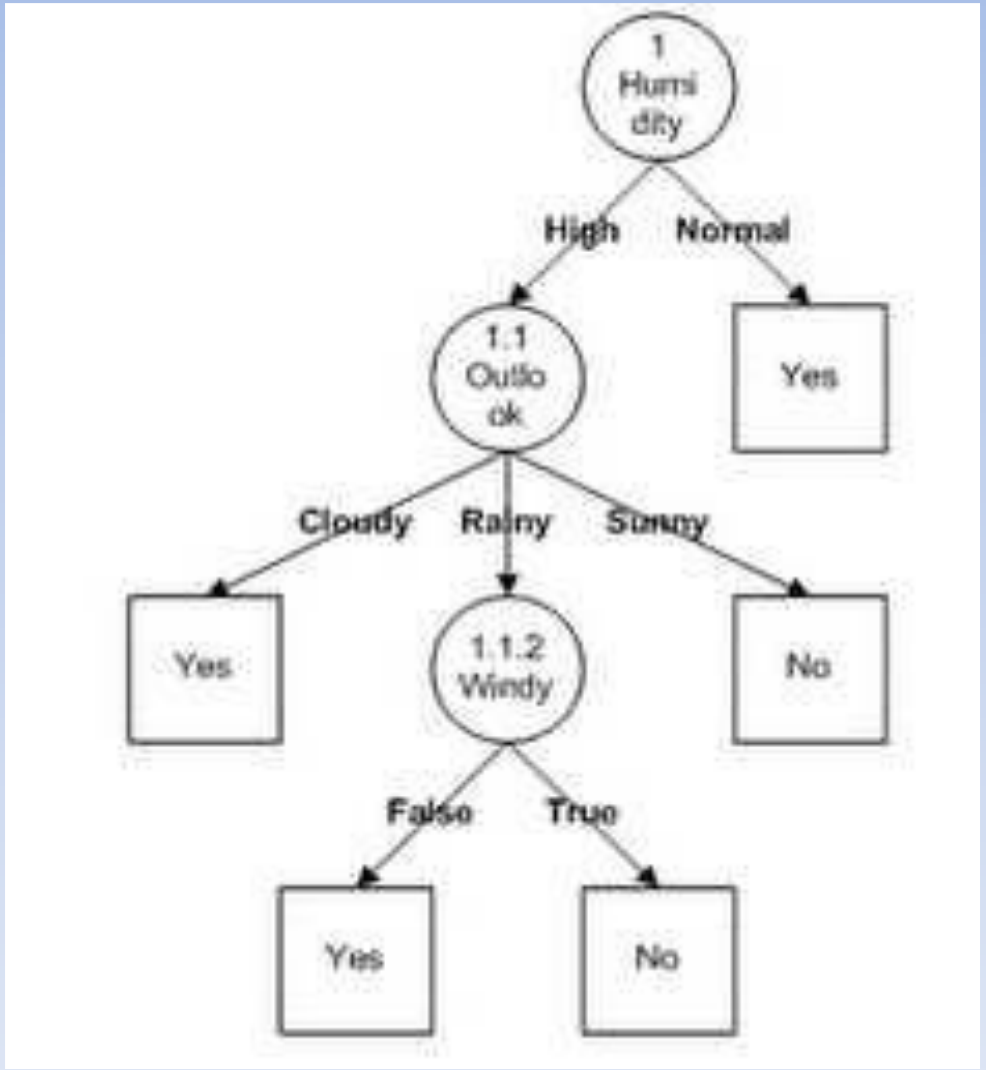
Node			Jml Kasus (S)	Tidak (S1)	Ya (S2)	Entropy	Gain
1.1.2	HUMIDITY-HIGH dan OUTLOOK-RAINY		2	1	1	1	
	TEMPERATURE						0
		COOL	0	0	0	0	
		HOT	0	0	0	0	
		MILD	2	1	1	1	
	WINDY						1
		FALSE	1	0	1	0	
		TRUE	1	1	0	0	

Algoritma C4.5

- Dari hasil pada tabel 4 dapat diketahui bahwa atribut dengan Gain tertinggi adalah WINDY yaitu sebesar 1.
- Dengan demikian WINDY dapat menjadi node cabang dari nilai atribut RAINY.
- Ada 2 nilai atribut dari WINDY yaitu FALSE dan TRUE. Dari kedua nilai atribut tersebut, nilai atribut FALSE sudah mengklasifikasikan kasus menjadi 1 yaitu keputusan-nya Yes dan nilai atribut TRUE sudah mengklasifikasikan kasus menjadi satu dengan keputusan No, sehingga tidak perlu dilakukan perhitungan lebih lanjut untuk nilai atribut ini.

Algoritma C4.5

Dengan memperhatikan pohon keputusan pada Gambar disamping, diketahui bahwa semua kasus sudah masuk dalam kelas. Dengan demikian, pohon keputusan pada Gambar disamping merupakan pohon keputusan terakhir yang terbentuk.



TUGAS

Buatlah Decision Tree dari table kasus di bawah ini :

NO	KEHADIRAN	LINGKUNGAN	KERJASAMA	PRAKARSA	REKOMENDASI
1	Rajin	Kurang Peduli	Mampu	Tidak Inisiatif	TIDAK
2	Cukup	Peduli	Tidak Mampu	Inisiatif	YA
3	Rajin	Kurang Peduli	Tidak Mampu	Kurang Inisiatif	TIDAK
4	Rajin	Peduli	Mampu	Inisiatif	YA
5	Rajin	Peduli	Mampu	Inisiatif	YA
6	Cukup	Peduli	Tidak Mampu	Inisiatif	YA
7	Kurang	Peduli	Tidak Mampu	Kurang Inisiatif	TIDAK
8	Rajin	Peduli	Tidak Mampu	Kurang Inisiatif	YA
9	Rajin	Peduli	Mampu	Inisiatif	YA
10	Kurang	Kurang Peduli	Tidak Mampu	Kurang Inisiatif	TIDAK
11	Cukup	Peduli	Mampu	Kurang Inisiatif	TIDAK
12	Rajin	Peduli	Mampu	Tidak Inisiatif	YA
13	Cukup	Peduli	Mampu	Inisiatif	YA
14	Rajin	Peduli	Mampu	Inisiatif	YA