Materi 4

DATA MINING

3 SKS | Semester 6 | S1 Sistem Informasi | UNIKOM | 2015

Nizar Rabbi Radliya | nizar.radliya@yahoo.com

Nama Mahasiswa	
NIM	
Kelas	

Kompetensi Dasar

Memahami pemrosesan awal data yang akan diproses dengan metode-metode data mining.

Pokok Bahasan

Pemrosesan Awal Data:
1. Agregasi (aggregation)
2. Penarikan contoh (sampling)
3. Diskretisasi dan binerisasi (discretization and binarization)
4. Pemilihan fitur (feature subset selection)
5. Transformasi atribut (attribute transformation)

Pada materi sebelumnya kita sudah membahas mengenai set data. Set data yang akan diproses dengan metode-motode data mining sering kali harus melalui pemrosesan awal. Langkah ini masuk ke dalam tahapan KDD sebelum proses data mining.

Beberapa permasalahan seperti jumlah populasi data yang besar, banyaknya data yang menyimpang (anomali data), dimensi yang terlalu tinggi, banyaknya fitur yang tidak berkontribusi besar, dan lain-lain merupakan pemicu munculnya pemrosesan awal data (*pre-processing*) yang harus diterapkan pada set data sebelum akhirnya digunakan dalam proses data mining. Beberapa pekerjaan yang umum dilakukan sebagai pemrosesan awal pada set data akan dibahas pada bab-bab di bawah ini.

I. Agregasi

Agregasi (*aggregation*) adalah proses mengkombinasikan dua atau lebih objek ke dalam sebuah objek tunggal. Aggregasi data sangat berguna ketika pada set data ada sejumlah nilai dalam satu fitur yang sebenarnya satu kelompok, yang tidak akan menyimpang dari deskripsi fitur tersebut jika nilainya digabungkan. Agregasi yang dapat dilakukan adalah *sum* (jumlah), *average* (rata-rata), *min* (terkecil), *max* (terbesar).

Sebagai contoh adalah data transaksi pembelian di beberapa cabang distributor. Setiah hari masing-masing cabang melakukan banyak sekali transaksi. Semua transaksi tersebut akan menghasilkan data yang besar dan komplek. Oleh sebab itu data tersebut

akan lebih sederhana tetapi tetap tidak menghilangkan deskripsinya apabila disajikan dalam bentuk gabungan setiap harinya di masing-masing cabang. Dengan begitu, pemrosesan data dalam data mining akan relatif lebih sederhana dan komputasinya menjadi lebih cepat. Selain itu dampaknya adalah penggunaan perangkat penyimpanan menjadi lebih sedikit atau kecil. Lebih jelasnya dapat dilihat pada tabel-tabel di bawah ini.

Tabel 1. Set Data Transaksi Pembelian Oleh Pelanggan

Cabang	IDT	Tanggal	Total
Bandung	B01001	30-01-2015	250.000
Bandung	B01002	30-01-2015	300.000
Tasikmalaya	T01001	30-01-2015	500.000
Tasikmalaya	T01002	30-01-2015	450.000
Tasikmalaya	T01003	31-01-2015	350.000

Misalnya kita menggunakan agregasi sum pada kolom total, dikelompokan berdasrkan kolom tanggal dan kolom IDT dapat dihilangkan sehingga hasilnya tampak seperti pada tabel 2 di bawah ini.

Tabel 2. Set Data Transaksi Pembelian Oleh Pelanggan Setelah Agregasi

Cabang	Tanggal	Total
Bandung	30-01-2015	550.000
Tasikmalaya	30-01-2015	950.000
Tasikmalaya	31-01-2015	350.000

Ada beberapa alasan mengapa sebaiknya melakukan agregasi, diantaranya adalah:

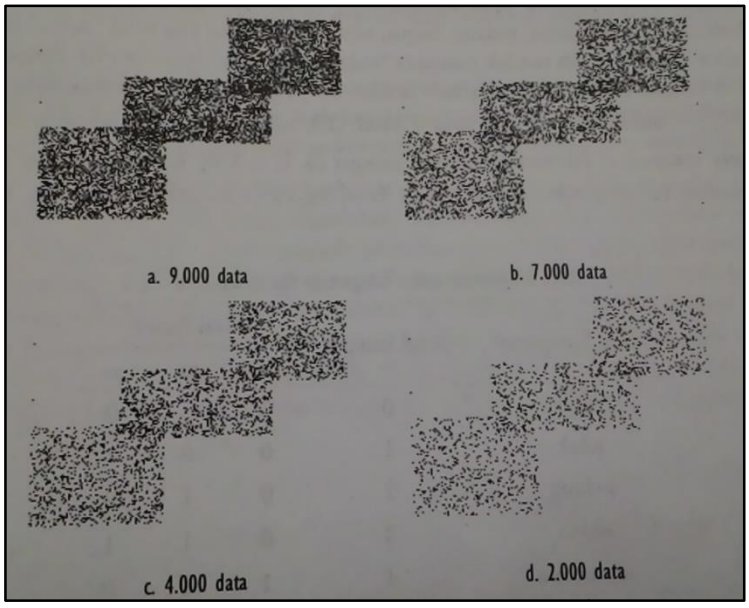
1. Set data yang lebih kecil akan membutuhkan memori penyimpanan yang lebih sedikit (pengurangan data atau perubahan skala).
2. Waktu pemrosesan dalam algoritma data mining menjadi lebih cepat.
3. Agregasi bertindak untuk mengubah cara pandang terhadap data dari level rendah menjadi level tinggi.
4. Perilaku pengelompokan objek atau atribut sering kali lebih stabil dari pada objek individu itu sendiri (lebih sedikit variasinya).

II. Penarikan Contoh

Kunci utama dalam penarikan contoh (*sampling*) adalah bahwa sampel data akan bekerja hampir sama dengan seluruh data jika sample tersebut mampu mewakili (representatif) seluruh data. Sample disebut representatif jika diperkirakan mempunyai sifat yang sama dengan seluruh data, biasanya diukur dengan rata-rata (*mean*) pada sample dan data asli. Jika sama atau sangat mendekati, sample tersebut bisa dikatakan

bagus. Tetapi, penggunaan sample yang baik juga tidak menjamin bahwa hasil pemrosesan data mining pada sample juga sama bagusnya dengan pemrosesan pada seluruh data asli.

Ada dua tipe penarikan contoh yang sering digunakan yaitu penarikan contoh tanpa pengembalian dan penarikan contoh dengan pengembalian. Pada teknik yang pertama, setiap data yang sudah terambil untuk digunakan sebagi sample tidak dikembalikan lagi ke data aslinya, sedangkan pada teknik kedua setiap data yang sudah terambil untuk digunakan sebagai sampel dikembalkan ke data asli. Akibatnya, sebuah data memiliki kemungkinan untuk muncul lebih dari satu kali dalam sampel. Sebagai contoh dapat dilihat pada gambar 1 di bawah ini.



Gambar 1. Struktur Data Yang Hilang Karena Penarikan Contoh

Pada gambar 1 di atas dapat dilihat contoh proses *sampling* secara acak pada set data dua dimensi yang berisi 9.000 data. Secara kasat mata penarikan contoh 7.000 data masih memberikan bentuk data yang menyerupai bentuk distribusi data yang asli. Ketika menggunakan 4.000 data, bentuk aslinya masih terlihat. Akan tetapi, ketika menggunakan 2.000 data, hasil penyampelan mulai terdistorsi dan bentuk asli data mulai tidak tampak.

III. Binerisasi dan Diskretisasi

Beberapa algoritma data mining, khususnya algoritma klasifikasi membutuhkan data dalam bentuk atribut kategorikal. Sedangkan algoritma asosiasi memerlukan data dalam bentuk atribut biner. Transformasi data dari tipe kontinu dan diskret ke atribut biner disebut binerisasi (*binarization*) sedangkan transformasi data dari atribut kontinu ke atribut kategoris disebut diskretisasi (*discretization*).

Cara pertama untuk melakukan binerisasi adalah dari M macam nilai kategoris, masing-masing diberikan nilai yang unik dengan nilai integer dalam jangkauan [0,M-1]. Jika atribut ordinal, urutan nilai kategorisnya harus diperhatikan. Misalnya untuk nilai kategoris kualitas = {rusak, jelek, sedang, bagus, sempurna}, nilai-nilai tersebut memeliki urutan nilai dari rendah ke tinggi (kalau dari contoh nilai kualitas tersebut dari kiri ke kanan). Jumlah bit yang dibutuhkan untuk binerisasi adalah N=$\lceil log_2(M) \rceil$.

Sebagai contoh dapat dilihat pada tabel 3 di bawah ini, dimana nilai kategoris kualitas = {rusak, jelek, sedang, bagus, sempurna} dikonversi menjadi nilai integer {0, 1, 2, 3, 4}. Karena ada lima macam nilai kategoris, jumlah bit yang dibutuhkan adalah N=$\lceil log_2(5) \rceil = 3$, yaitu menjadi tiga atribut biner x1, x2, x3.

Tabel 3. Konversi Atribut Kategoris ke Tiga Atribut Biner

Nilai Kategoris	**Nilai Integer**	**Nilai Biner**		
		X1	**X2**	**X3**
Rusak	0	0	0	0
Jelek	1	0	0	1
Sedang	2	0	1	0
Bagus	3	0	1	1
Sempurna	4	1	0	0

Sedangkan untuk melakukan diskretisasi terdiri atas dua langkah. Pertama, memutuskan berapa jumlah kategori yang harus digunakan. Langkah kedua, menentukan bagaimana memetakan nilai-nilai dari atribut kontinyu ke nilai kategori. Sebagai contoh nilai yang ada pada tabel 4 diubah menjadi atribut katarogikal dengan nilai: rendah, sedang tinggi.

Tabel 4. Contoh Atribut Kontinu Yang Akan Didiskretisasi

Atribut Kontinu
125
100
70
120
95
60
220
85
75
90

Pendekatan equal width: range data [60 - 220]:

Rendah: range [60-113]

Rendah: range [114-167]

Rendah: range [168-220]

IV. Pemilihan Fitur

Salah satu cara untuk mengurangi dimensi set data adalah dengan memilih fitur yang tepat atau hanya menggunaka atribut-atribut yang diperlukan. Secara konseptual, pemilihan sub set fitur merupakan suatu proses pencarian terhadap semua kemungkinan sub set fitur.

Dalam memilih fitur perlu diperhatikan fitur-fitur yang memiliki duplikasi informasi yang tergantung dalam satu dan lebih atribut lain. Selain itu fitur-fitur yang tidak relevan yaitu fitur-fitur yang tidak mengandung informasi berguna untuk tugas data mining secara langsung. Sebagai contoh NIM setiap mahasiswa sering tidak relevan untuk memprediksi IPK mahasiswa.

V. Transfromasi Atribut

Transformasi atribut berfungsi untuk memetakan keseluruhan himpunan nilai dari atribut yang diberikan ke suatu himpunan nilai-nilai pengganti yang baru sedemikian hingga nilai yang lama dapat dikenali dengan satu dari nilai-nilai baru tersebut.

Sebagain fungsi dari transformasi atribut adalah standarisasi dan normalisasi. Tujuan dari standarisasi dan normalisasi adalah untuk membuat keseluruhan nilai mempunyai suatu sifat khusus. Salah satu contoh transformasi standarisasi adalah dengan:

1. Hitung nilai tengah dengan median
2. Hidung absolute standard deviation dengan persamaan.

Rumus persamaan yang akan digunakan:

$$\sigma_A = \sum_{i=1}^{m} |x_i - \mu|$$

$$x' = \frac{(x - \mu)}{\sigma_A}$$

Sebagai contoh lakukan standarisasi dari data set berikut x = {2.5, 0.5, 2.2, 1.9, 3.1, 2.3, 2, 1, 1.5, 1.1}. Dari data tersebut dihitung median = μ=(1.9+2)/2=1.95. Maka dihasilkan hasil standarisasi pada tabel 5 di bawah ini.

Tabel 5. Contoh Standarisasi

x	x-μ	| x-μ |	x'
0.5	-1.45	1.45	-0.24

1.0	-0.95	0.95	-0.16
1.1	-0.85	0.85	-0.14
1.5	-0.45	0.45	-0.08
1.9	-0.05	0.05	-0.01
2.0	0.05	0.05	0.01
2.2	0.25	0.25	0.05
2.3	0.35	0.35	0.06
2.5	0.55	0.55	0.1
3.1	1.15	1.15	0.19
		$\sigma_A = 6.1$	

Sedangkan untuk transformasi atribut menggunakan normalisasi menggunakan pendekatan linear, yang pertama kita terlebih dahulu menghitung rata-rata (persamaan 1) dan varian (persamaan 2) dengan rumus:

$$x_k = \frac{1}{N}\sum_{i=1}^{N} x_{ik} \qquad \text{(persamaan 1)}$$

$$\sigma_k^2 = \frac{1}{N-1}\sum_{i=1}^{N}(x_{ik} - x_k) \qquad \text{(persamaan 2)}$$

Data hasil normalisasi dapat dihitung menggunakan cara pertama dengan persamaan berikut:

$$x_{ik} = \frac{x_{ik} - x_k}{\sigma_k} \qquad \text{(persamaan 3)}$$

Hasil normalisasi dengan cara persamaan 3 didapatkan fitur yang mempunyai sifat *zero-mean dan unit variance*. Teknik linear yang lain adalah dengan menskalakan jangkauan setiap fitur dalam jangkauan [0,1] (persamaan 4) atau [-1,1] (persamaan 5).

$$x_{ik} = \frac{x_{ik} - \min(x_k)}{\max(x_k) - \min(x_k)} \qquad \text{(persamaan 4)}$$

$$x_{ik} = \frac{2x_{ik} - (\max(x_k) + \min(x_k))}{\max(x_k) - \min(x_k)} \qquad \text{(persamaan 5)}$$

Sebagai contoh ada data $X = \{x_1, x_2, x_3, x_4, x_5\}^T$, dimana untuk $x_1 = \{0, 2, 1\}$, $x_2 = \{1, 7, 1\}$, $x_3 = \{2, 6, 3\}$, $x_4 = \{5, 1, 4\}$, $x_5 = \{3, 3, 4\}$. Jika diperhatikan, jangkauan nilai untuk fitur pertama adalah [0,5], fitur kedua [1,7], fitur ketiga [1,4]. Masing-masing fitur memiliki jangkauan yang tidak sama. Data tersebut disajikan pada tabel 6 di bawah ini.

Tabel 6. Contoh Data Belum Normal

Fitur 1	Fitur 2	Fitur 3
0	2	1
1	7	1
2	6	3
5	1	4

3	3	4

Jika dilakukan normalisasi menggunakan pendekatan linear yang pertama, dihitung terlebih dahulu rata-rata dan standar deviasi. Untuk fitur pertama, didapatkan:

$$x_1 = \frac{1}{5} \times (0 + 1 + 2 + 5 + 3) = 2.2$$

$$\sigma_1^2 = \frac{1}{5-1} \times ((0-2.2)^2 + (1-2.2)^2 + (2-2.2)^2 + (5-2.2)^2 + (3-2.2)^2) = 3.7$$

$$\sigma_1 = 1.9235$$

$$x_{11} = \frac{0-2.2}{1.9235} = -1.1437$$

$$x_{21} = \frac{1-2.2}{1.9235} = -0.6239$$

$$x_{31} = \frac{2-2.2}{1.9235} = -0.1040$$

$$x_{41} = \frac{5-2.2}{1.9235} = -1.4557$$

$$x_{51} = \frac{3-2.2}{1.9235} = -0.4159$$

Setelah dihitung pada 3 fitur didapatkan hasil normalisasi sebagai berikut:

Tabel 7. Hasil Normalisasi *Zero-Mean* dan *Unit Variance*

Fitur 1	Fitur 2	Fitur 3
-1.1437	-0.6954	-1.0550
-0.6239	1.236	-1.0550
-0.1040	0.8499	0.2638
1.4557	-1.0817	0.9231
0.4159	-0.3091	0.9231

Jika dinormalisasi menggunakan pendekatan linear dengan jangkau [0,1] didapatkan:

Tabel 8. Hasil Normalisasi Linear [0,1]

Fitur 1	Fitur 2	Fitur 3
0	0.1667	0
0.2000	1.0000	0
0.4000	0.8333	0.6667
1.0000	0	1.0000
0.6000	0.3333	1.0000

Jika dinormalisasi menggunakan pendekatan linear dengan jangkau [-1,1] didapatkan:

Tabel 9. Hasil Normalisasi Linear [-1,1]

Fitur 1	Fitur 2	Fitur 3
-1.0000	-0.6667	-1.0000
-0.6000	1.0000	-1.0000
-0.2000	0.6667	0.3333
1.0000	-1.0000	1.0000
0.2000	-0.3333	1.0000

VI. Daftar Pustaka

[1] Astuti, F.A. 2013. Data Mining. Yogyakarta: Andi.
[2] Kusrini & Taufiz, E.L. 2009. Algoritma Data Mining. Yogyakarta: Andi.
[3] Prasetyo, E. 2012. Data Mining: Konsep dan Aplikasi Menggunakan MATLAB. Yogyakarta: Andi.
[4] Prasetyo, E. 2014. Data Mining: Mengolah Data Menjadi Informasi Menggunakan MATLAB. Yogyakarta: Andi.

VII. Materi Berikutnya

Pokok Bahasan	Klasifikasi
Sub Pokok Bahasan	1. Konsep klasifikasi 2. Klasifikasi berbasis *decision tree* 3. Pembahasan algoritma