K- **Pembentukan cluster dalam *Knowledge Discovery in Database* dengan Algoritma K-Means**

Oleh:
Sri Andayani
Jurusan Pendidikan Matematika FMIPA UNY,email: andayani@uny.ac.id

Abstrak

Pembentukan cluster merupakan salah satu teknik yang digunakan dalam mengekstrak pola kecenderungan suatu data. Teknik ini ini digunakan dalam proses *Knowledge discovery in database* (KDD).

Salah satu algoritma pembentukan cluster data adalah algoritma K-Means. Algoritma bekerja dengan cara membagi data dalam k cluster. Setiap cluster ditentukan atas kedekatan jarak tiap-tiap data dengan titik tengahnya (*mean point*).

Sebuah basis data sangat mungkin berisi data non numerik, yang tidak dapat ditentukan titik tengahnya. Algoritma K-Means dapat dipergunakan untuk pembentukan cluster dalam sebuah basis data yang besar dengan menerapkan aturan *similarity* dan *dissimilarity* terhadap data dalam basis data terlebih dahulu.

Kata kunci: *Cluster, Knowledge Discovery in Database,* Algoritma K-Means,

Pendahuluan

Dewasa ini pengolahan data elektronik telah menjadi kebutuhan yang sangat utama. Perkembangan pesat dalam teknologi informasi yang menjadikan semua informasi dapat disimpan dalam jaringan komputer telah membuat munculnya sistem basis data yang sangat besar. Dalam hitungan detik, data-data dalam berbagai basis data akan senatiasa terbarukan, baik dikarenakan adanya *update* maupun penambahan data baru. Permasalahan yang kemudian muncul adalah bagaimana mengetahui informasi yang terdapat dalam basis data yang sangat besar.

Knowledge discovery in Database (KDD) didefinisikan sebagai ekstraksi informasi potensial, implisit dan tidak dikenal dari sekumpulan data. Proses *knowledge discovery* melibatkan hasil dari proses *data mining* (proses mengekstrak kecenderungan pola suatu data), kemudian mengubah hasilnya secara akurat menjadi informasi yang mudah dipahami.

Ada beberapa macam pendekatan berbeda yang diklasifikasikan sebagai teknik pencarian informasi/pengetahuan dalam KDD. Ada pendekatan kuantitif, seperti pendekatan probabilistik and statistik. Beberapa pendekatan memanfaatkan teknik visualisasi, pendekatan klasifikasi seperti logika induktif, pencarian pola, dan analisis pohon keputusan. Pendekatan yang lain meliputi deviasi, analisis kecenderungan, algoritma genetik, jaringan syaraf tiruan dan pendekatan campuran dua atau lebih dari beberapa pendekatan yang ada.

Pada dasarnya ada enam elemen yang paling esensial dalam teknik pencarian informasi/ pengetahuan dalam KDD ([7]), yaitu: (1) mengerjakan sejumlah besar data, (2) diperlukan efisiensi berkaitan dengan volume data, (3) mengutamakan ketepatan/keakuratan, (4) membutuhkan pemakaian bahasa tingkat tinggi, (5) menggunakan beberapa bentuk dari pembelajaran otomatis, dan (6) menghasilkan hasil yang menarik.

Clustering

Salah satu metode yang diterapkan dalam KDD adalah *clustering*. *Clustering* adalah membagi data ke dalam grup-grup yang mempunyai obyek yang karakteristiknya sama ([1]). Garcia-Molina et al. ([2]) menyatakan *clustering* adalah mengelompokkan item data ke dalam sejumlah kecil grup sedemikian sehingga masing-masing grup mempunyai sesuatu persamaan yang esensial.

Clustering memegang peranan penting dalam aplikasi data mining, misalnya eksplorasi data ilmu pengetahuan, pengaksesan informasi dan text mining, aplikasi basis data spasial, dan analisis web. *Clustering* diterapkan dalam mesin pencari di Internet. Web mesin pencari akan mencari ratusan dokumen yang cocok dengan kata kunci yang dimasukkan. Dokumen-dokumen tersebut dikelompokkan dalam cluster-cluster sesuai dengan kata-kata yang digunakan.

Kategori *clustering*

Tan, dkk.([4]) membagi *clustering* dalam dua kelompok, yaitu *hierarchical and partitional clustering*. *Partitional Clustering* disebutkan sebagai pembagian obyek-obyek data ke dalam kelompok yang tidak saling overlap sehingga setiap data berada tepat di satu cluster. *Hierarchical clustering* adalah sekelopok cluster yang bersarang seperti sebuah pohon berjenjang (hirarki).

William ([8]) membagi algoritma *clustering* ke dalam kelompok besar seperti berikut:

1. *Partitioning algorithms*: algoritma dalam kelompok ini membentuk bermacam partisi dan kemudian mengevaluasinya dengan berdasarkan beberapa kriteria.
2. *Hierarchy algorithms*: pembentukan dekomposisi hirarki dari sekumpulan data menggunakan beberapa kriteria.
3. *Density-based*: pembentukan cluster berdasarkan pada koneksi dan fungsi densitas.
4. *Grid-based*: pembentukan cluster berdasarkan pada struktur *multiple-level granularity*
5. *Model-based*: sebuah model dianggap sebagai hipotesa untuk masing-masing cluster dan model yang baik dipilih diantara model hipotesa tersebut.

Algoritma K-Means

Algoritma K-Means adalah algoritma *clustering* yang paling popular dan banyak digunakan dalam dunia industri [1]. Algoritma ini disusun atas dasar ide yang sederhana. Ada awalnya ditentukan berapa cluster yang akan dibentuk. Sebarang obyek atau elemen pertama dalam cluster dapat dipilih untuk dijadikan sebagai titik tengah (*centroid point*) cluster. Algoritma K-Means selanjutnya akan melakukan pengulangan langkah-langkah berikut sampai terjadi kestabilan (tidak ada obyek yang dapat dipindahkan):

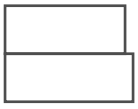
1. menentukan koordinat titik tengah setiap cluster,
2. menentukan jarak setiap obyek terhadap koordinat titik tengah,
3. mengelompokkan obyek-obyek tersebut berdasarkan pada jarak minimumnya.

Gambar 1 berikut menunjukkan diagram alir dari algoritma K-Means.

Berikut ini adalah ilustrasi penggunaan algoritma K-means untuk menentukan cluster dari 4 buah obyek dengan 2 atribut, seperti ditunjukkan dalam Tabel 1. *Clustering* akan dilakukan untuk membentuk 2 cluster jenis obat berdasarkan atributnya ([6]).

Langkah-langkah algoritma K-means adalah sebagai berikut :

1. Pengesetan nilai awal titik tengah. Misalkan obat A dan obat B masing-masing menjadi titik tengah (*centroid*) dari cluster yang akan dibentuk.

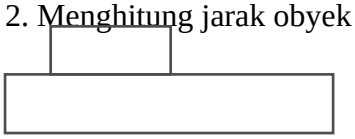


*Tentukan koordinat kedua centroid tersebut,*yaitu dan

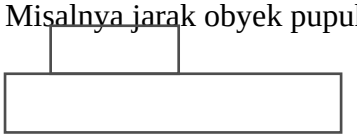
Tabel 1. Daftar obyek yang akan diolah dalam *clustering*

Obyek	atribut1 (X): indeks berat	atribut 2 (Y): pH
Obat A	1	1
Obat B	2	1
Obat C	4	3
Obat D	5	4

2. Menghitung jarak obyek ke centroid dengan menggunakan rumus jarak Euclid.



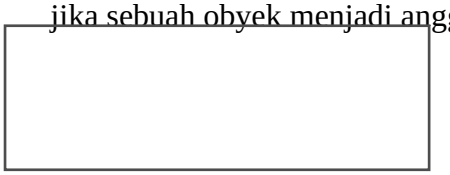
Misalnya jarak obyek pupuk C=(4,3) ke *centroid* pertama *adalah* dan jaraknya dengan centroid



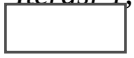
kedua adalah .

Hasil perhitungan jarak ini disimpan dalam bentuk matriks k x n, dengan k banyaknya cluster dan n banyak obyek. Setiap kolom dalam matriks tersebut menunjukkan obyek sedangkan baris pertama menunjukkan jarak ke *centroid* pertama, baris kedua menunjukkan jarak ke *centroid* kedua. Matriks jarak setelah iterasi ke-0 adalah sebagai berikut:

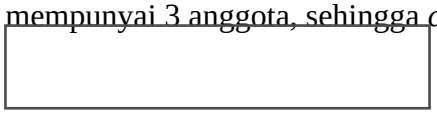
3. Clustering *obyek* : Memasukkan setiap obyek ke dalam cluster (grup) berdasarkan jarak minimumnya. Jadi obat A dimasukkan ke grup 1, dan obat B, C dan D dimasukkan ke grup 2. Keanggotaan obyek ke dalam grup dinyatakan dengan matrik, elemen dari matriks bernilai 1 jika sebuah obyek menjadi anggota grup.



4. *Iterasi-1*, *menetukan* centroid : Berdasarkan anggota masing-masing grup, selanjutnya

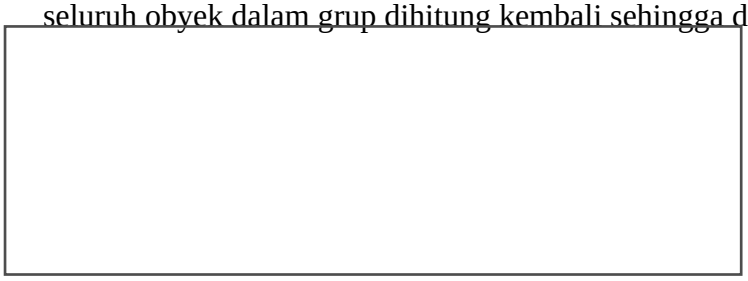


ditentukan *centroid* baru. Grup 1 hanya berisi 1 obyek, sehingga *centroid*nya tetap . Grup 2 mempunyai 3 anggota, sehingga *centroid*nya ditentukan berdasarkan rata-rata koordinat ketiga

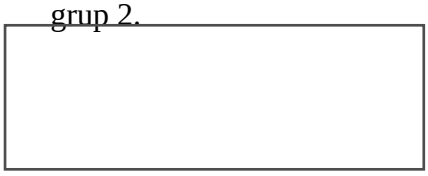


anggota tersebut: .

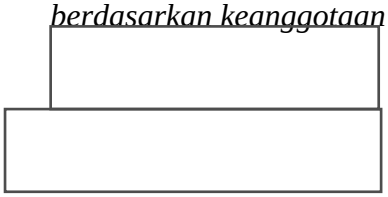
5. *Iterasi-1, menghitung jarak obyek ke* centroid: selanjutnya, jarak antara *centroid* baru dengan seluruh obyek dalam grup dihitung kembali sehingga diperoleh matriks jarak sebagai berikut:



6. *Iterasi-1*, clustering *obyek*: *langkah* ke-3 diulang kembali, menentukan keanggotaan grup berdasarkan jaraknya. Berdasarkan matriks jarak yang baru, maka obat B harus dipindah ke grup 2.



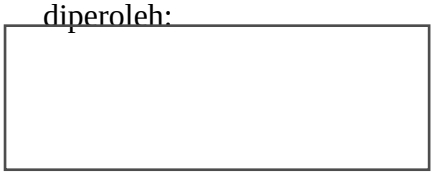
7. *Iterasi-2, menentukan* centroid: *langkah ke-4 diulang kembali untuk menentukan centroid baru berdasarkan keanggotaan grup yang baru.* Grup 1 dan grup 2 masing-masing mempunyai 2



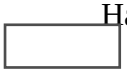
anggota, sehingga centroidnya menjadi dan

8. *Iterasi-2, menghitung jarak obyek ke* centroid : ulangi langkah ke-2, sehingga diperoleh matriks jarak sebagai berikut:

9. *Iterasi-2*, clustering *obyek*: mengelompokkan tiap-tiap obyek berdasarkan jarak minimumnya, diperoleh:



Hasil pengelompokkan pada iterasi terakhir dibandingkan dengan hasil sebelumnya,



diperoleh . Hasil ini menunjukkan bahwa tidak ada lagi obyek yang berpindah grup, dan algoritma telah stabil. Hasil akhir *clustering* ditunjukkan dalam Tabel 2.

Tabel 2. Hasil *clustering*

Obyek	atribut1 (X): indeks berat	atribut 2 (Y): pH	Grup hasil
Obat A	1	1	1
Obat B	2	1	1
Obat C	4	3	2
Obat D	5	4	2

Kelebihan dan Kelemahan algoritma K-means

Algoritma K-means dinilai cukup efisien, yang ditunjukkan dengan kompleksitasnya **O(tkn)**, dengan catatan **n** adalah banyaknya obyek data, **k** adalah jumlah cluster yang dibentuk, dan **t** banyaknya iterasi. Biasanya, nilai k dan t jauh lebih kecil daripada nilai n. Selain itu, dalam iterasinya, algoritma ini akan berhenti dalam kondisi optimum lokal ([8]).

Hal yang dianggap sebagai kelemahan algoritma ini adalah adanya keharusan menetukan banyaknya cluster yang akan dibentuk, hanya dapat digunakan dalam data yang *mean*-nya dapat ditentukan, dan tidak mampu menangani data yang mempunyai penyimpangan-penyimpangan (*noisy data* dan *outlier*). Berkhin([1]) menyebutkan beberapa kelemahan algoritma K-means adalah: (1) sangat bergantung pada pemilihan nilai awal centroid, (2) tidak jelas berapa banyak cluster k yang terbaik, (3) hanya bekerja pada atribut numerik.

Similarity* dan *Dissimilarity

Memperhatikan input dalam algoritma K-Means, dapat dikatakan bahwa algoritma ini hanya mengolah data kuantitatif. Hal tersebut juga diungkapkan oleh Berkhin ([1]), bahwa algoritma K-means hanya dapat mengolah atribut numerik.

Sebuah basis data, tidak mungkin hanya berisi satu macam type data saja, akan tetapi beragam type. William ([8]) menyatakan sebuah basis data dapat berisi data-data dengan type sebagai berikut: *symmetric binary, asymmetric binary, nominal, ordinal, interval* dan *ratio*. Sedangkan Pal dan Mitra menyebutkan sebuah basis data dapat berisi data-data teks, simbol, gambar dan suara ([3]).

Berbagai macam atribut dalam basis data yang berbeda type (dalam [5] disebut sebagai data multivariate, seperti nominal, ordinal, and kuantitatif) harus diolah terlebih dahulu menjadi data numerik, sehingga dapat diberlakukan algoritma K-means dalam pembentukan clusternya. Pengukuran *similarity* dan *dissimilarity* dapat digunakan untuk pengolahan data tersebut ([5]).

Atribut yang berbeda tipe sama artinya dengan adanya ketidaksamaan (*dissimilarity)* antar atribut tersebut. Ketidaksamaan (*dissimilarity)* antara dua obyek dapat diukur dengan menghitung jarak antar obyek berdasarkan beberapa sifatnya. Hubungan *dissimilarity* antara 2 buah data obyek a=(a1,a2,……,ap) dan b=(b1,b2, ….,bp) dapat dinyatakan dengan pengukuran jarak antara 2 obyek tersebut. Beberapa sifat jarak (*dissimilarity)* adalah sebagai berikut ([5] dan [8]):

. d(a, b) (0 , jarak kedua obyek selalu positif atau nol,
. d(a, a) = 0, jarak terhadap diri sendiri adalah nol,
. d(a, b) = d(b, a) , jarak kedua obyek adalah simetri,
. d(a, b) (d(a, c) + d(c, b), jarak memenuhi ketidaksamaan segitiga.

Misalkan *dissimilarity* antara obyek **i** dan obyek **j** dinyatakan dengan $\mathbf{d_{ij}}$ dan *similarity* dinyatakan dengan $\mathbf{s_{ij}}$. Hubungan antara *relationship* dissimilarity dengan *similarity* dinyatakan dengan $\mathbf{s_{ij.=1-}}$ $\mathbf{d_{ij}}$, dengan *similarity* terbatas pada 0 dan 1 ([5]). Jika *similarity* bernilai satu (benar-benar sama), maka *dissimilarity* nol, dan jika *similarity* bernilai nol (sangat berbeda), *dissimilarity* bernilai satu. Setelah perhitungan jarak atau *dissimilarity* dari setiap variabel, maka seluruh hasil dikumpulkan menjadi sebuah indeks *similarity* (atau

dissimilarity) antara dua obyek ([5]). Selanjutnya hasil tersebut dapat diolah menjadi obyek-obyek yang akan dikelompokkan dalam cluster-cluster oleh algoritma K-means.

Penutup

K-means adalah algoritma pembentukan cluster yang populer dan mengolah data numerik. Namun demikian, algoritma ini juga dapat digunakan untuk pembentukan cluster dari sebuah basis data yang atribut-atributnya berasal dari tipe yang berbeda-beda, dengan cara mengubah atribut-atribut tersebut ke dalam indeks *similarity* atau *dissimilarity*.

Referensi:

[1] Berkhin, Pavel. *Survey on clustering data mining techniques*, http://www.ee.ucr.edu/~barth/EE242/clustering_survey.pdf

[2] Garcia-Molina, Hector; Ullman, JD., & Widom, Jennifer. 2002. *Database systems the complete book, International edition.* New Jersey, Prentice Hall.

[3] Pal, Shankar K & Mitra, Pabitra. 2004. *Pattern Recognition algorithms for data mining.* CRC Press.

[4] Tan, Pang-Ning,; Steinbach,Michael; Kumar ,Vipin. *Data Mining Cluster Analysis: Basic Concepts and Algorithms.* www-users.cs.umn.edu/ ~**kumar** /dmbook/-16k.

[5] Teknomo, Kardi. ***Similarity Measurement*** http://people.revoledu.com/kardi/tutorial/Similarity/index.html

[6] Teknomo, Kardi. *Numerical Example of K-Means Clustering*, http://people.revoledu.com/kardi/tutorial/kMean/NumericalExample.htmNu

[7] Wright, Peggy , *Knowledge Discovery In Databases: Tools and Techniques*, http://www.acm.org/crossroads/xrds5-2/kdd.html#11

[8] William, Graham, *Data Mining Cluster*, http://datamining.anu.edu.au/student/math3346_2005/050809-maths3346-clusters-2x2.pdf

---

start

Banyaknya cluster K

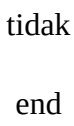
Tentukan pusat

Hitung jarak obyek ke pusat

Kelompokkan obyek berdasar jarak minimum

Ada obyek yang harus dipindah?

ya



Gambar 1. Flowchart algoritma K-Means